INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 05 MELBOURNE, AUGUST 15-18, 2005

MODEL-BASED DESIGN OF ACTUATION CONCEPTS: A SUPPORT FOR DOMAIN ALLOCATION IN MECHATRONICS

Sebastian Jansen and Ewald G. Welp

Abstract

This paper presents a methodical approach for the development of mechatronic product concepts. The task of domain allocation, i.e. the selection of the appropriate engineering domain (mechanics, electronics or software) is identified as a major step in mechatronic design. In order to develop a methodical approach for domain allocation, existing methods for the conceptual design of mechatronic products are analysed. Requirements on a domain allocation method are identified and a methodical framework is presented. The interfaces between the mechanical and the electrical domain are represented by actuators. Thus, actuator selection is strongly related to domain allocation. For this reason, a model-based approach for the design of actuation concepts is integrated into the domain allocation method.

Keywords: Mechatronics, product structure, component selection, actuator principles, modelling, simulation, software tool

1. Introduction

Mechatronic products fulfil their functions by a synergetic interaction of mechanics, electronics and software. In the context of mechatronics, the term "mechanics" is often used to subsume the disciplines which are usually assigned to the area of mechanical engineering, namely mechanics, hydraulics, pneumatics etc. In this paper, the term is used in the same manner for clarity reasons. Especially in the phase of conceptual design of mechatronic systems, tight cooperation between experts of the different disciplines is required [1], since a superior mechatronic product results from a global optimum taking into account all involved domains. However, the combination of design principles from different domains leads to a large number of possible solutions for a given design task. In order to achieve the desired optimum with a minimum of time-consuming iterations, a systematic approach to the design of mechatronic systems is desirable. In the later phases of the mechatronic development process like embodiment and detailed design, domain specific methods can be applied, because the solution principles as well as the interfaces between the different domains have been determined in the conceptual design phase. Hence, rather domain-specific tasks like circuit layout, microcontroller programming or optimisation of the mechanical structure have to be fulfilled and can be treated with the feasible methods from the respective disciplines. Although the efficient development of mechatronic products also requires an integration of domain-specific methods and tools, e.g. the coupling of multi-body-simulation and control design, in particular the early stages of the mechatronic design process need methodical support [2].

A major task in the conceptual design of a mechatronic product is to decide, which functions should be realised in which domain, i.e. which ones should preferably be realised by mechanics, which ones by electronics and which ones by software. In literature, different terms are used for this important step in the design process. For example, in [2] the term "technology allocation" is used, while other authors refer to this design step as "partitioning" of the mechatronic system [3], [4]. In this paper, the term "domain allocation" is used as a synonym.

The design and selection of the interfaces between different domains is strongly related to the issue of domain allocation. The interfaces between the mechanical and the electrical domain are represented by sensors and actuators. While sensors transform non-electrical properties into processable electrical output signals, (mechatronic) actuators transform electrical signals into non-electrical physical properties [2]. Consequently, the domain-structure of a mechatronic system, i.e. the part of the product-structure which describes the relations between functions and appropriate domains, can only be generated if feasible domain-interfaces (actuators and sensors) exist. Therefore, a systematic approach for the design and the selection of actuators and sensors is desirable. This paper describes a methodical support for the design of domain interfaces in mechatronic products. Exemplarily, the focus is set on the area of actuation principles. However it should be mentioned that the proposed method can be adapted to the design and selection of other system components such as sensors or mechanical converters like gear mechanisms. Thus, the approach is not only limited to the use in the context of mechatronics and can be adapted to a wider field of application.

This paper is organized as follows: In chapter 2, existing approaches and research results related to the problem of domain allocation and actuator selection are analysed. Chapter 3 presents a framework for a methodical support for the creation of domain-structures and identifies the task of actuator selection as an important subproblem. Chapter 4 concretises the developed method for the design and selection of actuation principles. Chapter 5 presents a prototypic software tool which is based on the proposed method. Furthermore, the application of the method is shown exemplarily. Chapter 6 concludes the results of the presented work.

2. Related work

2.1. Methods for conceptual mechatronic design and domain allocation

Due to the nature of mechatronic systems, namely the synergetic interaction of working principles from mechanics, electronics and software technology, existing methods from the involved disciplines cannot be directly transferred to the use in mechatronic design. This problem in particular arises in the early design phases where the product concept is generated in a domain-spanning, collaborative development process. The specific requirements on a methodical support for the conceptual design of mechatronic systems have led to a number of research activities aiming on the provision of methods and tools for the mechatronic development process.

Buur [2] defines a set of axiomatic statements that describe the general characteristics of mechatronic systems. Based on these axioms, he suggests a number of theorems for mechatronic design. The axioms and theorems can be considered as a theoretical basis for the development of design methods for mechatronic systems. In regard to domain allocation, the Law of Vertical Causality is adapted to mechatronics and the importance of the function/means tree is emphasized. Although the results cover some fundamental aspects that have to be considered in the development of design method for mechatronics, a holistic integrative approach is not presented.

Isermann [5] mentions that simultaneous engineering has to take place during the development of mechatronic systems, since the mechanical and electronic components have to be considered as an integrated overall system from the beginning. However, the proposed design methods are rather related to the control aspects of mechatronic systems. A more generic concept is presented in [6]. Based on the process that has to be carried out by the system, the functions which are preferable realized by information technology are identified as well as the ones which are fulfilled by energy-dominated principles. At this point also the interfaces like sensors or actuators are considered. In later steps, the initial allocation is checked for possible optimizations and improved if necessary. Thus, domain allocation can be considered as an iterative approach that starts in early phases of the development process and the domain-structure can vary during the development process due to optimizations.

The guideline VDI 2206 describes a flexible procedure model which is based on the V-shaped model on the macro-level and the cycle of problem solving on the micro-level. In addition, several predefined process modules for typical design tasks are provided. The partitioning of the system, i.e. the process of domain allocation, is mentioned in context of transition from functions to solution elements. However, the guideline does not give any detailed methodical support concerning the conceptual design and the task of domain allocation [7][3].

A more detailed description of domain allocation is presented in [8]. The domain-structure is generated in different steps. Before the functions are allocated to the mechanical, electrical or software domain, the system is divided into energy- and information-dominated areas. After that, the domain-structure is detailed and varied in order to obtain an appropriate solution. Although the proposed differentiation between energy- and information-dominated product functions is an interesting approach for domain allocation, a concrete method for the synthesis of the domain-structure is missing.

Salminen and Verho, who have analysed mechatronic design processes, emphasize the necessity for a tight collaboration of experts from mechanics, electronics and software especially in the conceptual design phase. They propose a combined application of feasible existing domain-specific methods. However, they mention that missing linkages between complementary methods complicate this procedure. [1]

In [4] the partition of functions in mechatronics and the choice of means to fulfil the functions is characterized as a problem of high complexity due to the large number of means and their relationships. The author emphasizes the need for a computer tool which can support the designer in creating the product concept and proposes an approach of function-costing as a methodical basis.

An object-oriented computer tool for conceptual design of mechatronic systems is presented in [9]. The tool is supposed to support the designer by providing solution principles for abstract functions which are stored in a knowledge base. Thus, the creation and representation of the product concept rather takes place on an abstract level and the problem of domain allocation is not explicitly addressed.

2.2. Creation and representation of product models and structures

Domain allocation is tightly related to the issue of modelling product concepts and the representation of product structures. All involved disciplines, mechanics, electronics and software, have their own ways of conceptual modelling (e.g. sketches of mechanisms, block diagrams or state diagrams). However, a total view of the product is important since the selection of a particular design principle influences the properties of the entire product [10]. Nevertheless, until today there exits no domain-independent modelling language that is capable of covering all aspects of the domain-specific approaches. Due to the diversity of the different aspects which have to be modelled (e.g. geometries in mechanics, signal flows in electronics), it seems difficult to define a common modelling language for all disciplines and it is questionable if such a language would be beneficial in terms of efficiency and unambiguousness.

While a number of approaches for product structures aims on the representation of existing products e.g. for the purpose of manufacturing or assembly, others are rather related to the synthesis of new products. In [11], a graphical language for the representation of product structures is presented. The approach enables the designer to create product structures on different levels of abstraction, since the graphical symbols for functions, working principles and parts can be modelled simultaneously. Furthermore, the concept also defines the representation of the relations between the elements and considers hierarchical relations as well as non-hierarchical ones. The approach is intended to cover domain-spanning design as it is required in mechatronics through the use of domain-independent graphical elements and domain-specific representations by providing different views on the product structure such as hydraulic circuits or block diagrams for control design. The presented concept seems to be a good basis for the representation of mechatronic product concepts. However, some important aspects like the process of generation of domain specific views are not described in detail. Furthermore, state transitions as they occur in mechatronic systems are barely considered.

2.3. Actuator selection

The interfaces between electronics and mechanics are represented by actuators. The selection of appropriate actuator principle and the location of the actuators within the mechatronic product strongly influence the properties of the whole system. For example, the properties of a system with a central actuator and gear-coupled mechanical movements will differ from those of a mechatronic system with independent actuators coupled by electronics. In mechatronic systems, different types of actuators are used. Many of them like DC motors, solenoids or stepper motors are based on the electromagnetic or electrodynamic principle. However, in recent years a number of alternatives, for example piezoelectric ceramics and polymers or shape memory alloys, have arisen. The properties of the actuator principles are different and for an advantageous mechatronic product concept, the selection of the best suited principle is a crucial factor.

For the selection of electromagnetic actuators like DC motors, a number of selection methods and tools exists. In most cases, a preselection of motors is carried out by the comparison of the required peak power and the maximum motor power. For motors which pass this test, the range of feasible transmission ratios is determined by speed and torque analysis and simulations. In [12] a selection criterion for servo motors is presented. The criterion is based on a load curve derived from the required motor torque, which is normalized with respect to the motor inertia. The load torque is represented by an average or peak value, which leads to a loss of precision regarding the dynamic behaviour of the movement. Thus, also this approach cannot supersede the use of simulations for the selection of an appropriate actuator.

A method for the selection of actuators that operate in linear fashion causing a finite change in length such as piezos and shape memory alloys is introduced in [13]. Different performance indices which measure the effectiveness of an actuator are defined. The performance indices are estimated from manufacturers' data and simple models of performance limitation. Furthermore, the authors provide graphical representations of actuator characteristics which allow the comparison of different actuation principles. A database prototype for the selection of actuators based on performance indices is presented in [14].

3. Framework of a methodical support for domain allocation

3.1. Requirements on a domain allocation method

In general, domain allocation is a problem of selecting the best suited technological domain for a specific product function or a set of interrelated product functions. An overview of the requirements on the domain allocation method and the main interrelations between them is shown in figure 1.

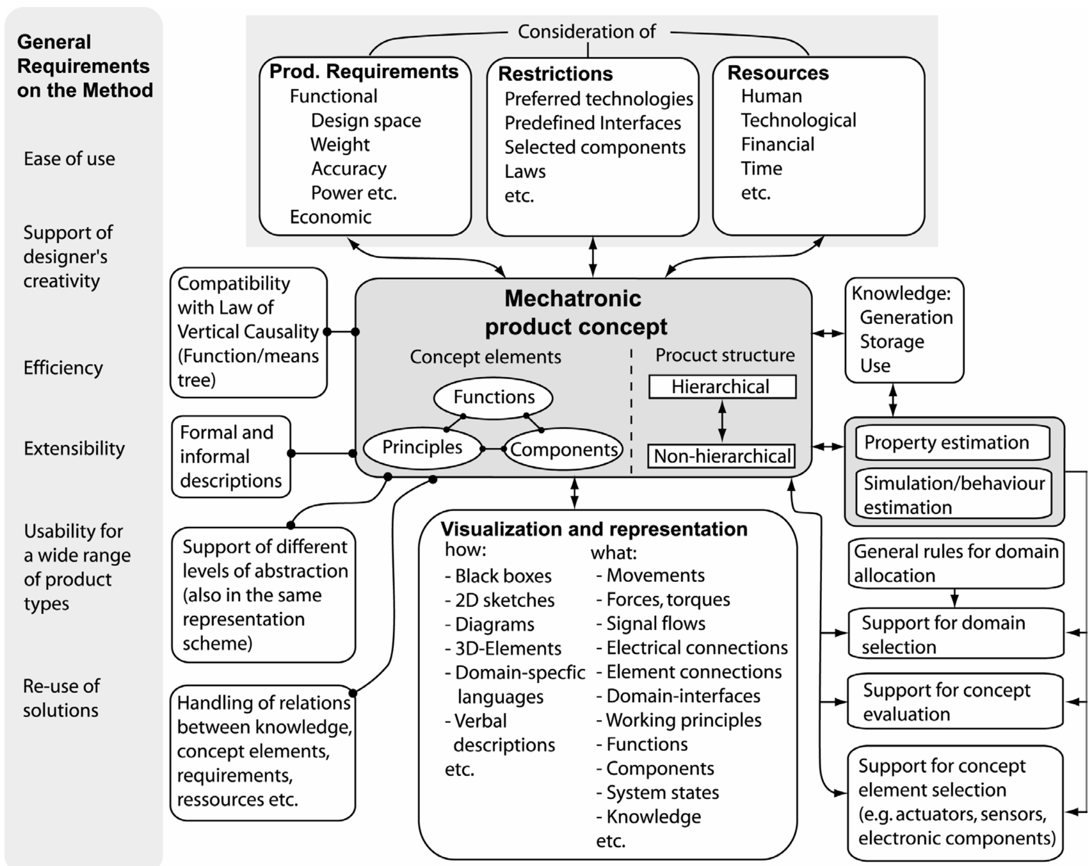

Figure 1. Requirements on the domain allocation method

As already mentioned, domain allocation is strongly related to the creation of the product concept and structure. In some cases the decision for a certain domain can be made without a detailed analysis of substituting working principles. This is for instance the case, if general restrictions like preferred technologies for certain functions exist (e.g. the use of a specific actuation concept due to modularization issues). However, in most cases the appropriate domain can only be selected if different alternatives of working principles and their interrelations with the product structure are considered. Therefore, a method which can support domain allocation has to support the creation and representation of the entire mechatronic product concept. In order to develop the concept, product requirements, restrictions and resources have to be considered. The method should provide appropriate means to manage these influencing factors. The concept should be represented in a flexible, task-oriented way. Therefore, different principles of visualization and different levels of abstraction should be used. For example, it should be allowed to mix 2D sketches and 3D representations of working principles with verbal descriptions or black-boxes that only define the inputs and outputs of a function. Furthermore, the properties of the product concept should be estimated as accurately as possible with respect to the level of concept concretization. The selection of domains should be supported by general rules which for example can help the designer to identify the advantages or disadvantages that result from the use of a certain domain in a certain context.

The creation of the domain-structure, especially the selection of domain-interfaces represented by sensors and actuators, should be supported by simulations of the system's behaviour and property estimations. The knowledge which is generated during the design process (e.g. reasons for design decisions, identified problems or advantages as well as simulation results) should be stored and provided if necessary.

3.2. Concept of a domain allocation method

Product requirements and restrictions can be considered as sources of influencing factors on the layout of appropriate domain structures [15]. Furthermore, the resources available for the design and manufacturing of a mechatronic product have to be considered as illustrated in figure 1. Thus, a methodical support should handle product requirements, restrictions and resources in order to guide the process of conceptual design in general and the task of domain allocation in particular. The proposed approach considers these aspects by assigning them to the concept elements. The procedure is exemplarily demonstrated in figure 2 for the development of a mechatronic robot leg.

In the first step, the overall function of the system is defined and the input and output variables are added. The influencing factors (requirements, resources, restrictions) are linked to the overall function. In the next step, the overall function can be divided into subfunctions, if it is too complex to serve as a basis for the search for appropriate working principles.

In order to decide which domain or domains should or have to be used to fulfil a certain function, an analysis of the function is necessary. In general, all functions which influence the movement of material are related to mechanics. This includes for example the acceleration, the deformation and the support of bodies. Consequently, the means required to fulfil a certain product function must contain working principles from the mechanical domain, if the movement of material has to be influenced by the function. This general result can be considered as an example for a general allocation rule and can support the process of domain allocation, since it can help to identify functions which have to be fulfilled by the use of mechanical means. Another example of a general allocation rule is the fact, that the use of software-based functions enhances the flexibility of the product with respect to its versatility on the one hand and its adaptability to specific requirements on the other hand.

In the example, the overall function requires to influence the movements of the leg. Thus, the mechanical domain is naturally involved. In the next step, a simple model of the leg's basic mechanical components is created. The model can be represented by a 2D sketch or a 3D model. Due to this flexibility, the designer can adapt the modelling approach to the requirements of the design task and the level of concept concretization. Furthermore, different types of concept elements like functions and working principles, can be combined in the same representation scheme, as suggested in [16], for example. Interrelations between different concept elements, influencing factors etc. can be added.

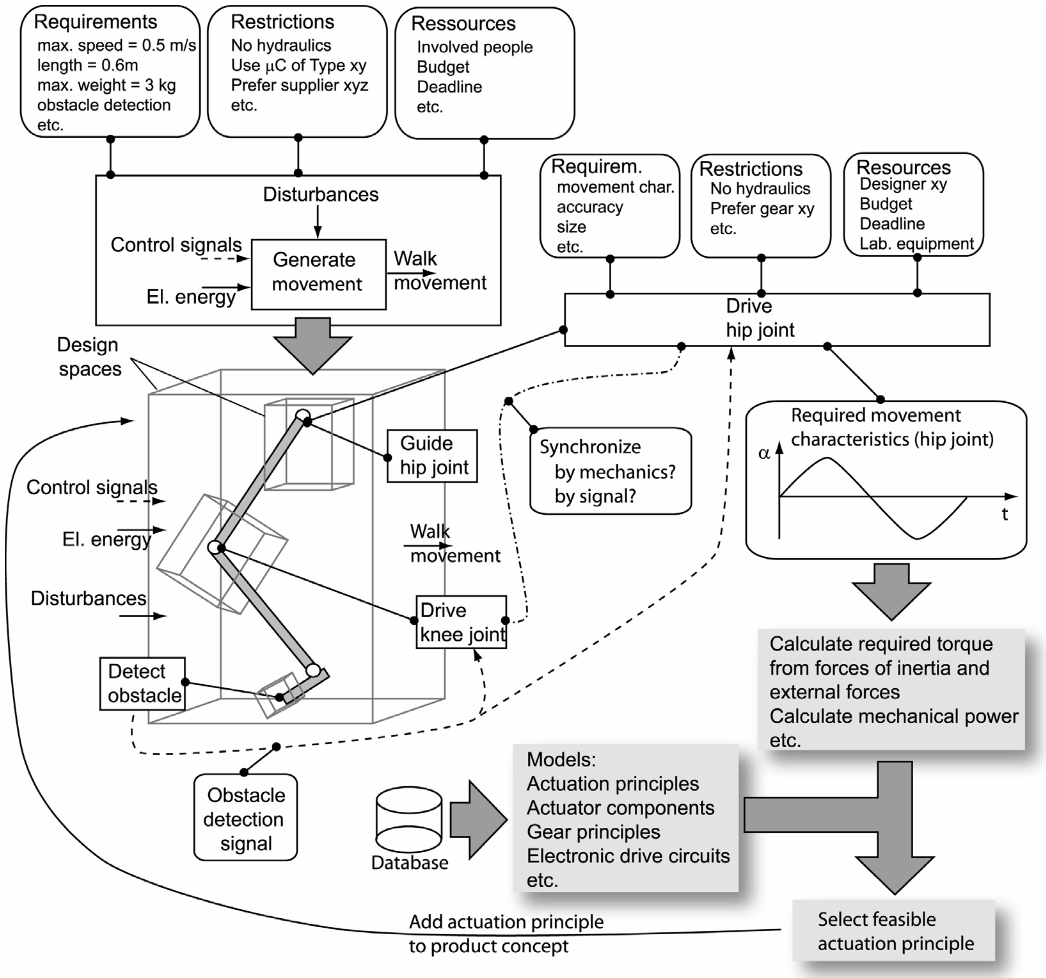

Figure 2. Design of a mechatronic leg as an example for the systematic creation of mechatronic product concepts

In figure 2 a signal-based interaction between the functions "detect obstacle" and "drive knee joint" / "drive hip joint" has been identified. In further design steps this interaction has to be concretized. There is also an interrelation between the two drive functions. In the illustrated state of concept, it has not yet been decided whether the synchronization of movements should be realised by mechanical principles or by signal, i.e. by electronics and software. Since the leg is supposed to react on obstacles in a flexible way, a mechanical coupling between the hip movement and the knee movement seems inappropriate. For this reason, two independent actuators synchronized by electronics and software should be preferred. In order to describe and analyze the interrelations between concept elements, an approach similar to the one presented in [17] seems promising, where the interactions are quantified with respect to their level of necessity and harmfulness.

Moreover, design spaces for the identified subfunctions can be defined. The usability of design spaces for product modelling is for example described in [18]. A similar approach seems reasonable in the context of domain allocation, since spatial aspects have a major impact on the selection of feasible working principles and technologies. The design spaces can be considered as black-boxes, if only a function description or input and output variables are determined. The black-boxes can be filled with subfunctions, working principles or components in order to establish the product concept step by step.

As already mentioned, the selection of working principles and components should be supported by a domain allocation method. In particular the selection and arrangement of domain interfaces is of major importance. Therefore, a systematic approach for the selection of actuator and sensor principles should be embedded into the proposed method. Exemplarily, figure 2 shows the basic procedure of actuator selection. Before an actuation principle can be designed, the functions which cover domain interfaces have to be identified. In the example, the function “drive hip joint” represents a domain interface between electronics and mechanics. Thus, the working principles of this function will include one or more actuation principles that transform electrical power into mechanical power. As illustrated, the characteristics of the required movements that have to be generated are linked to the function. From the movement characteristics, the inertial properties of the moved parts and the external forces, different values like the required mechanical power, the maximum speed and acceleration etc. can be calculated. The available design space and other requirements like maximum input voltage, weight or cost can be added to the interface function and can be used as input parameters for the selection of actuation principles, too. From a database, the models of different actuator and gear principles are selected and their feasibility for the specific actuation task is analysed by an algorithm. Furthermore, information on typical electronic drive circuits for different actuator principles can be provided. The so-gained information enable the designer to select a feasible actuation principle. The principle is added to the product concept and can be linked to other related concept elements. The next section provides a more detailed description of the method for the design and selection of actuator principles.

4. Model-based design and selection of actuator principles

In mechatronic systems, the interfaces between the electrical and the mechanical domain are represented by actuators. All actuators have in common, that they contain one or more physical effects which transform energy from one type into another type. Figure 3 gives an overview of the most common energy transformation effects that are used in mechatronic actuators.

Physical effect	Examples	Input		Output
Electrodynamic effect (Lorentz force)	DC motor Voice coil actuator	Electrical energy	→	Mechanical energy
Electromagnetic effect (reluctance force)	Stepper motor Electromagnet	Electrical energy	→	Mechanical energy
Piezoelectric effect	Single stroke / multiple stroke/ ultrasonic actuators	Electrical energy	→	Mechanical energy
Magnetostrictive effect	Single stroke / multiple stroke/ ultrasonic actuators	Magnetic energy	→	Mechanical energy
Shape-memory effect	Wire actuators, foil actuators	Thermal energy	→	Mechanical energy
Thermal expansion	Bimetal	Thermal energy	→	Mechanical energy
Effect of fluidic pressure	Hydraulic cylinder, hydraulic motor, pneumatic cylinder	Fluidic energy	→	Mechanical energy
Electrorheological effect	Electrorheological damper	Electrical energy	→	Change of fluid viscosity
Magnetorheological effect	Magnetorheological damper	Magnetic energy	→	Change of fluid viscosity

Figure 3. Energy transformation effects used in actuators

As shown in the figure, not all transformation effects used in actuators have an electrical input and a mechanical output. In order to use these effects within an actuator that couples the electrical and the mechanical domain, they have to be combined with other physical effects. A hydraulic cylinder for example transforms fluidic energy into mechanical energy. Nevertheless this effect can be used to couple electrical and mechanical variables, if an electrically controlled valve is added which influences the fluid pressure in the cylinder.

Figure 4 shows the basic structure of an actuator and exemplary specifications of the elements. The input signals, which in mechatronic systems usually are electrical, are combined with auxiliary energy in an amplifier. Thus, the output energy of the amplifier is a function of the input signal and represents the input of the transformer. By the transformer, the energy type is changed into the required output energy type. If the quantities of the transformer's output do not match the required quantities, they can be adapted by a converter.

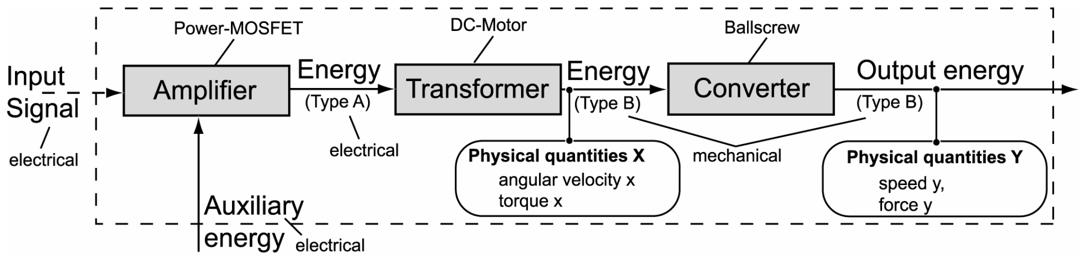

Figure 4. Basic structure of an actuator

The basic structure can be used as a starting point for the development of a method which supports the design and selection of actuation principles. In order to select a feasible transformation and conversion principle, the required output of the actuator has to be analysed with respect to the type and characteristics of the physical quantities. Since the output energy of a mechatronic actuator is usually of mechanical type, the methodical approach is described for this case. However, it should be mentioned that a similar procedure can be used if the required output energy is of non-mechanical type, e.g. thermal or fluidic.

The concept of the proposed method is shown in figure 5. The method starts with an analysis of the required mechanical movement that has to be driven by the actuator. The required characteristics of the movement derive from the considered product concept. Usually the designer can estimate the displacement, speed or acceleration of a specific movement already in early phases of the conceptual design phase. In order to provide an input for the method, an arbitrary movement can be created by specifying the time-dependent behaviour of the linear and rotatory components of the movement. In the next step, the estimated inertias of the moved bodies can be linked to the movement as well as springs, dampers or arbitrary external forces and torques. Based on these information, the time-dependent drive forces and torques as well as the required mechanical work and power output which have to be provided by the actuator are calculated. Furthermore, a number of characteristic values like maximum power or maximum displacement are determined. These characteristic values describe the basic properties of the movement and can be used for a first assessment of the applicability of a particular transformation component, for example.

In order to create an advantageous actuation concept, the properties as well as the behaviour of different transformation principles have to be considered. For this purpose, a generic model is provided for each transformation principle. The model describes the principle by defining the relevant geometric and physical properties as well as the input and output types of the transformation principle. For example, the generic model of a DC motor includes variables for the length and the diameter as geometric properties, the weight, the torque constant, the electrical resistance etc. as physical properties and furthermore defines the input as electrical power and the output as mechanical power. Following the notation of bond graph theory, the input and output power is expressed as a product of an effort variable (e.g. voltage, torque) and a flow variable (e.g. current, velocity). This differentiation is beneficial for the modelling of element couplings, since a coupling requires continuity of both, the flow and the effort variable.

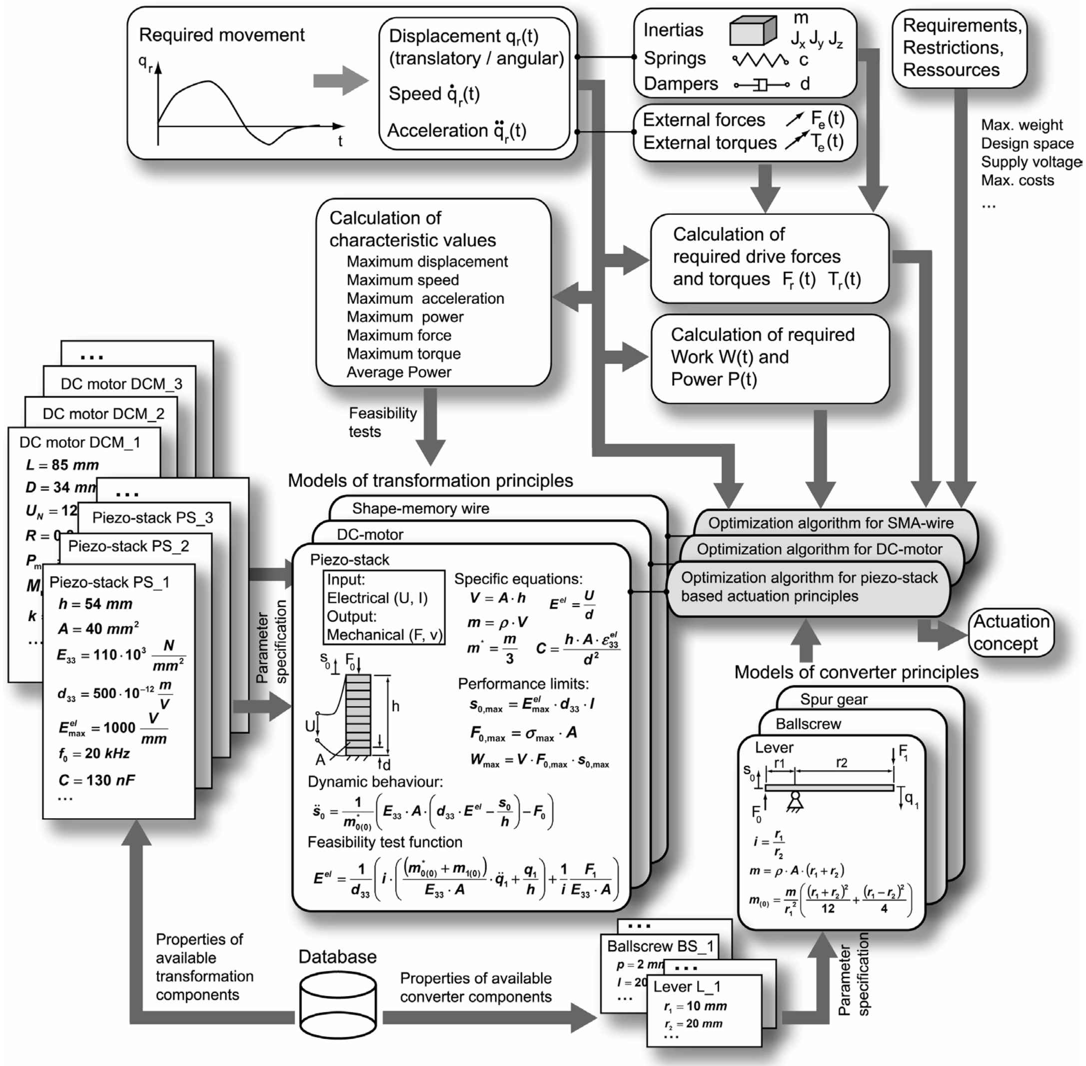

Figure 5. Model-based method for the design of actuation principles

Furthermore, the model comprises methods for the calculation of the values of dependent properties from the properties which serve as independent inputs. In the case of the piezo stack in figure 5, the length, the cross-section and the plate thickness could serve as independent geometric variables. In combination with material properties like elastic modulus, piezoelectric constant etc., the dependent variables like mass, capacity or stiffness can be calculated.

The specification of a model can be performed in two different ways. On the one hand, the data of available components stored in a database can be used to specify the variables of the model. In this case, the usability of an existing component for a particular actuation task is analysed. On the other hand, new components can be created by assigning arbitrary values to the model's variables. In this way, a synthesis process can be performed and usability of the created components can be analysed again.

The performance of working principles is limited by certain principle-specific factors. This is also true for transformation principles used in actuators. In the case of a piezo stack, for example, the maximum work output per unit volume that can be achieved is limited by the actuator material properties, namely the maximum permissible stress, the maximum permissible electric field and the piezoelectric constant. Thus, given a required work output per actuator stroke, the minimum material volume which is necessary for this actuation task can be calculated. The so-gained value can serve as a lower bound for the actuator volume. It represents the volume which is necessary if the force on the piezo stack is equal to the maximum permissible force during the entire stroke at maximum permissible electric field. In reality, the force of the actuator usually is not constant and the applied voltage should not be too close to the maximum permissible value. Thus, in most cases a larger volume of piezoelectric material will be required. Nevertheless, the performance limit can be used as a basis for first feasibility tests. Actuators with a smaller material volume than the lower bound do not have to be considered in further steps of analysis. Besides the discussed performance limit of maximum work per volume, other performance limits can be considered. For example, the first resonance frequency of a piezo stack which depends on the mass, the geometry and the stiffness, can be calculated from the independent parameters and can be used as an upper bound for dynamic operation.

As exemplarily described for a piezo stack, performance limits can be determined for other transformation principles, too. For example, in the case of actuators based on shape memory alloys, the maximum frequency usually is limited by the heat transfer between the actuator and the environment, since the actuator has to cool down from the austenite finish to the martensite start temperature in order to complete an actuation cycle. In contrast to the heating time which theoretically can be arbitrarily decreased by increasing the applied power, the cooling time depends on the shape of the actuator as well as on the surrounding medium and its flow. Thus, the cooling process usually takes considerably longer than the heating process. Similar to the piezo stack, the mechanical work which can be performed by a shape memory wire within one cycle is limited by the maximum permissible deformation, the maximum permissible stress and the material volume.

The maximum mechanical power that can be provided by a permanent magnet DC motor depends on the applied voltage and the armature resistance, while the maximum torque depends additionally on the torque constant. Figure 6 summarizes some examples of performance limits of different transformation principles.

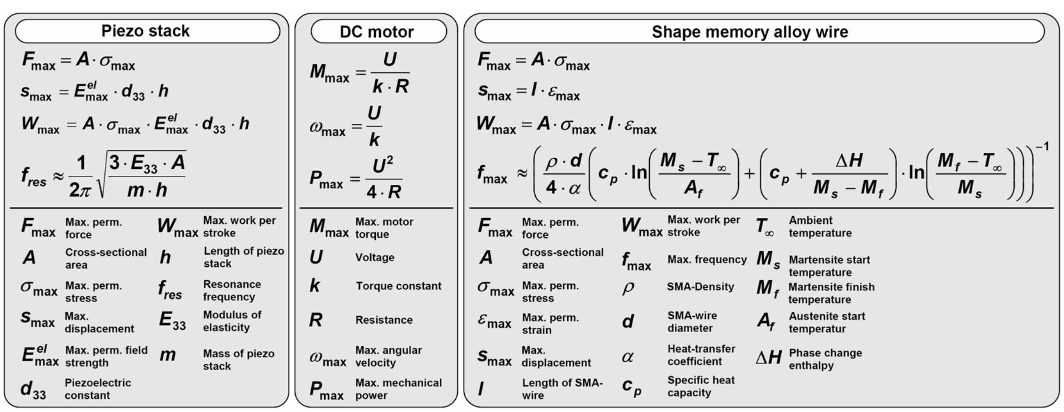

Figure 6. Examples of performance limits for piezo stack , permanent magnet DC motor and shape memory alloy wire

The performance limits can be compared to the characteristic values of the movement and thus can be used for the purpose of a rough dimensioning and feasibility tests in early phases. But in order to design and select an actuation concept that meets all requirements, further investigations are necessary. Especially the dynamic behaviour of the actuator has to be taken into account. In most cases, a converter principle with a certain transmission ratio has to be added. Thus, an optimization of the complete actuator requires the consideration of the interaction between transformation and converter principle. Therefore the transmission ratio is included as a parameter in the feasibility test function. This function is used to evaluate the usability of an actuation concept for the generation of the required movement. In the function, which is exemplarily shown in figure 5 for a piezo stack, the required electrical input of the actuator (e.g. electric field strength, voltage or current) is calculated from the dynamic movement characteristics (displacement, speed, acceleration, force etc.). An actuation concept is considered as feasible, if the calculated electrical input does not exceed a principle-specific critical value. Regarding the piezo stack, for example, this value is represented by the permissible maximum field strength.

In order to generate solutions which adequately fulfil the requirements, the actuation principles have to be optimized. For this purpose, specific optimization algorithms for the different transformation principles are provided. These algorithms vary the parameters of the model and evaluate the result with respect to the criteria derived from the requirements and restrictions. If, for example, the usability of a particular DC motor from the database has to be analyzed, the transmission ratio of the converter is varied in order to get the desired dynamic behaviour. However, also an optimization based on variation of more than one parameter (e.g. length and cross-sectional area of a piezo stack or a shape memory wire) is possible. For this purpose, genetic algorithms are used. The fitness functions of the genetic algorithms depend on the user-defined optimization criteria like volume or weight. In order to influence the optimization process, the designer can rank the criteria or combine them with weight factors.

5. Software prototype and validation of the method

The presented method for domain allocation in general and the model based approach for the design of actuation concepts in particular should be implemented in a computer tool in order to increase the efficiency of use. Especially for the appropriate representation of mechatronic product concepts, for the handling of interrelations between concept elements and for the execution of extensive calculations and optimizations, a software-based tool seems appropriate.

In order to investigate the usability of the developed methodical approch, a first software prototype which covers a number of the aspects described above, has been realized. For the implementation, an object-oriented approach has been chosen which is well suited to represent the structural aspects of domain allocation. Java has been chosen as programming language, since it is platform independent and strictly object-oriented. The implementation of three-dimensional aspects of the product concept is realized in Java3D, which provides powerful means for the representation of virtual worlds. The data of product concepts, available components, requirements etc. is stored in an object-relational database, namely PostgreSQL, which is coupled to the Java program via the Hibernate-Framework. This increases the flexibility of the design tool, since the database can be exchanged without changing the source code of the program. Figure 7 shows a screenshot of the user interface of the design tool.

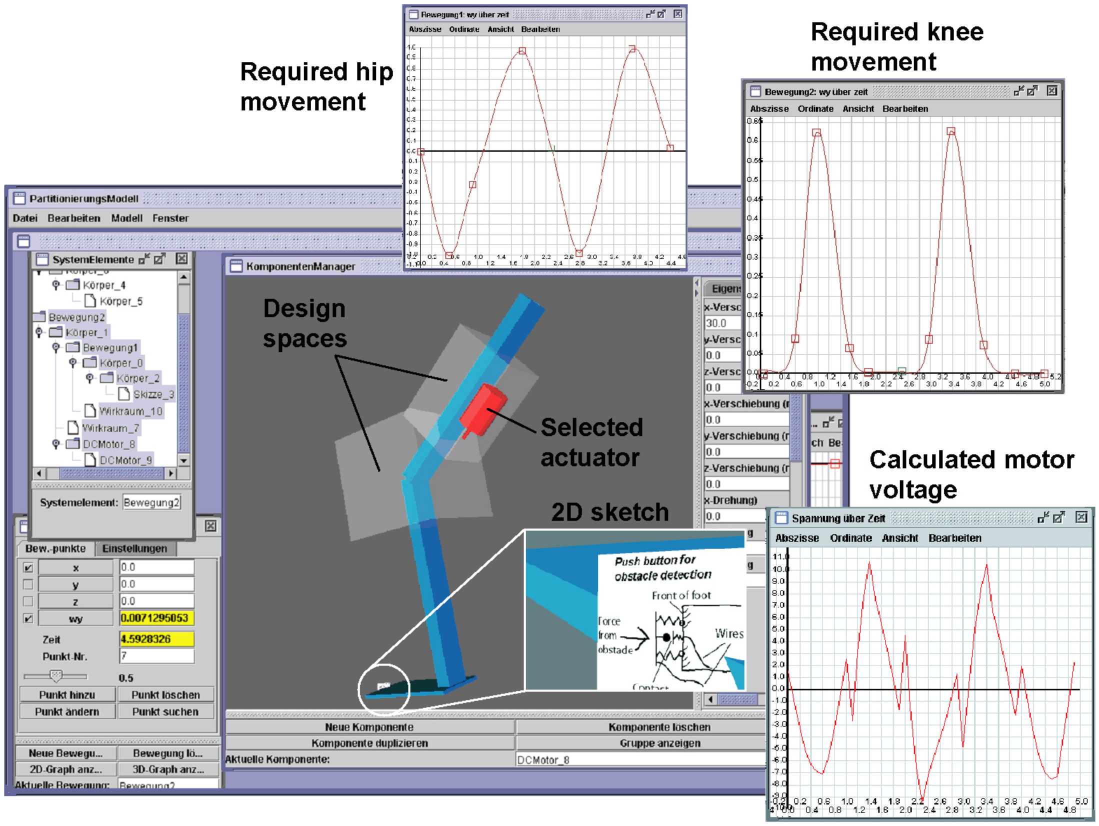

Figure 7. Screenshots of the design tool prototype based on the presented method

The software prototype has been tested in a mechatronic development process. The design task was the development of a mechatronic leg which is used for educational issues. In the first step, the required movements of the leg were estimated and designed. For this purpose, the computer tool is equipped with a graphical user interface which enables the designer to create arbitrary movement characteristics by defining translatory and rotatory displacement, speed or acceleration. As described above, the method is intended to support different ways of modelling and representation. For this reason, the software prototype is capable of representing three dimensional components and working principles as well as design spaces and 2D-sketches in the same model. For the representation of components, simple shapes are used. All components can be linked to each other in order to represent their interrelationships.

In the next step, the required drive forces and torques are calculated. For this purpose, the forces and moments of inertia of the moved components, the force of gravity etc. are calculated. The so-gained information is used for the dimensioning and the selection of an appropriate actuation concept. In the example, the use of a DC motor in combination with a transmission gear is analyzed. Since the calculated motor voltage does not exceed the permitted voltage, the proposed motor could be used for the given drive task.

6. Conclusions

Domain allocation is an important step in the conceptual design of mechatronic systems. The development of a superior product requires a deliberate design of the product structure. In mechatronic design, the domain structure is an important aspect of the product concept, since it defines the interrelations between the engineering disciplines and the functions of a product.

For the creation of the domain structure, appropriate methods and tools are required. In this context, especially the means used for the representation of the product concept can be considered as a crucial aspect. Since today there exists no common language for the description of solution concepts from different domains, a mix of different means of representation seems reasonable. The presented methodical approach tries to meet these requirements by providing a modelling approach which supports formal and informal descriptions of solutions as well as a coexistence of functions, working principles and components in the same structure. On the one hand, the semiformal approach can support creativity, since the designer is not forced to formalize his thoughts and ideas. On the other hand, a stepwise formalization is possible which enables the designer to evaluate a concept by simulations or interrelation analysis.

In order to guide the design team during the conceptual design of mechatronic systems, a set of general rules or suggestions for the creation of the domain structure should be provided. For example, the mechanical domain cannot be completely substituted by electronics and software, if the considered function requires to influence the movement of material. Provided that they are used in the right context, general allocation rules can increase efficiency and creativity in the design process.

The design of the interfaces between the different domains has been identified as a major issue for domain allocation. Therefore, a systematic approach which can support the design and the selection of actuation concepts has been developed. The approach is based on an analysis of the required movements within a specific product concept. Scalable models are used to evaluate the applicability of the different actuation principles. For the purpose of an efficient preselection of feasible transformation principles, principle-specific performance limits have been identified. For the detailed investigation of the feasibility of a particular actuation concept, the required value of the electrical input variable is calculated and compared to the permissible value. An optimization of properties is performed by genetic algorithms. The applicability of the method has been tested in a mechatronic design process.

The results of the validation indicate that the proposed methodical approach can enhance the efficiency and transparency of the design process. The practical usability of the method is strongly related to its implementation in a software tool. Therefore, the method will be extended and detailed in future work, accompanied by an optimization and completion of the presented computer program.

Acknowledgement

Many thanks to the German Research Foundation (DFG) for financing the project "Entwicklung einer Partitionierungsmethodik für mechatronische Systeme" within which this contribution had been worked out.

References

[1] Salminen, V., Verho, A., "Systematic and innovative design of a mechatronic product", Mechatronics, Vol 2, No. 3, 1992, pp. 257-275.

[2] Buur, J., "A theoretical Approach to Mechatronics Design", PhD Thesis, Technical University of Denmark, Institute for Engineering Design, 1990.

[3] Gausemeier, J., Möhringer, S., „New Guideline VDI 2206 – A flexible procedure model for the design of mechatronic systems", Proceedings of the 14th International Conference on Engineering Design (ICED), Stockholm, 2003.

[4] French, M.J., "The partition of functions in mechatronic design", Proceedings of the International Conference on Mechatronics: Designing Intelligent Machines, Cambridge, September 1990, pp. 209-211.

[5] Isermann, R., "Modeling and Design Methodology for Mechatronic Systems", IEEE/ASME Transactions on Mechatronics, Vol. 1, No. 1, March 1996, pp. 16-28.

[6] Isermann, R., "Mechatronische Systeme - Grundlagen", Springer Verlag, Berlin, 2002.

[7] VDI 2206: „Entwicklungsmethodik für mechatronische Systeme (Design methodology for mechatronic systems)", Beuth Verlag, Berlin, 2004.

[8] Lippold, C., "Eine domänenübergreifende Konzeptionsumgebung für die Entwicklung mechatronischer Systeme", PhD Thesis, Institute of Engineering Design, Ruhr-University Bochum, Bochum, 2001.

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

[10] Buur, J., Andreasen, M.M., "Design models in mechatronic product development", Journal of design studies", Vol. 10, No. 3, 1989.

[11] Riepe, B., "Integrierte Produktstrukturmodellierung in den frühen Phasen der Produktentstehung", PhD Thesis, University of Paderborn, 2003.

[12] Van de Straeten H.J., Degezelle, P., De Schutter, J., Belmans, R.J.M., „Servo Motor Selection Criterion for Mechatronic Applications", IEEE/ASME Transactions on Mechatronics, Vol 3, No. 1, March 1998, pp. 43-50.

[13] Huber, J.E., Fleck, N.A., Ashby, M.F., "The selection of mechanical actuators based on performance indices", Proceedings of The Royal Society of London: Mathematical, Physical and Engineering Sciences, Vol. 453, No. 1965, October 1997, pp. 2185-2205.

[14] Zupan, M., Ashby, M.F., Fleck, N.A., "Actuator Classification and Selection - The Development of a Database", Advanced Engineering Materials, Volume 4, No. 12, December 2002, pp. 933-940.

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

[16] Ulrich, K.T., Eppinger, S.D., „Product design and development", 3rd ed., McGraw-Hill, New York, 2004.

[17] Pimmler, T.U., Eppinger, S.D.: „Integration Analysis of Product Decompositions", Proceedings of the ASME Sixth International Conference on Design Theory and Methodology, Minneapolis, MN, September 1994, pp. 343-351.

[18] Lossack, R.S., „Konstruktionsarbeitsräume als Grundlage zur Integrierten Produktmodellierung", PhD Thesis, University of Karlsruhe, 1997.

Sebastian Jansen Ruhr-University Bochum Institute of Engineering Design 44780 Bochum Germany
Tel: Int +49 234 / 32-26314 Fax: Int +49 234 / 32-14159 E-mail: jansen@lmk.rub.de
URL: http://www.lmk.ruhr-uni-bochum.de/