New Object-Oriented Approach of Modelling Mechatronics System in Conceptual Stage

Raivo Sell, Mart Tamre

TALLINN TECHNICAL UNIVERSITY DEPARTMENT OF MECHATRONICS
Ehitajate tee 5
Tallinn, Estonia
raivo@staff.ttu.ee, mart@staff.ttu.ee

Abstract

The paper is dealing with new object-oriented concept of the mechatronics modelling methodology, focusing to the early design phase. The idea is based on the artificial intelligence methods, particularly genetics algorithms. Genetic algorithms are used to calculate different solution for given problem. The engineering problem is defined by Unified Modelling Language. The aim of the developed methodology is to create engineering environment for fast and effective design in early design phase. The result of this phase is product specification and optimized block diagram which is compatible with well-known mechatronics modelling systems and can be imported for future development.

Key words: mechatronics product design, conceptual stage, artificial intelligence

I. Introduction

In this paper, we focus on the pre-conceptual and conceptual stage of product development process. Techniques and methods used today are covered as well as new trends and research in this field. The automation of the conceptual design phase development becomes more important because the complexity of mechatronic systems and arising of new techniques together with the increase of computational power as well as time and cost saving demands in today’s world. Many efforts are still needed to develop industry proved semi-automated design tools for conceptual design phase. To achieve this goal we need to use a unified problem definition method, together with a pre-defined component database and a rule set for the automated design process. The method for customer requirements definition must be understandable for product designers and other engineering staff as well as for customer. The aim of this work is to develop a semi-automated algorithm, based on new computing techniques. The algorithm has to process the design specification to propose a design candidate as a block diagram of the desired product. An interactive process between engineer and algorithm can lead for an optimal design concept with effective consumption of time and cost. The semi-automated interactive software tool makes it easier go carry through the conceptual design process and reach to the optimal conceptual solution.

There are two principally different architectures of mechatronic systems: conventional hierarchical architecture and new network architecture. Hierarchical control has dominated in our lives including engineering systems for a long time. The network structure which consists of intelligent units suits much better for today’s dynamic environment [9]. Designing such kind on network of autonomous intelligent units, the new approach, based on artificial intelligent technologies is necessary for effective and relevant result. Developing this kind of methodology in the objective of our work and the current situation is covered in this paper.

The paper discusses a design process in the conceptual phase and techniques for this phase from the point of the objective. The paper covers
development of the initial modelling concept, the component structure and some future steps of the algorithm development. This work focuses on main concepts and alternatives for the component-based mechatronics modelling methodology.

II. Product Design in Conceptual design Stage

Product design process in general can be described as a set of certain phases, need to be passed despite the diversity of the products even in a single domain. The phases mentioned are specification definition, conceptual design, product design and product support phase.

On the specification phase the main goal is to understand the design problem thoroughly respectively to the customer requirements. The biggest problem here is that there is very little information about the design problem available in this phase. The result of this stage is a design specification and a clearly formulated set of desired measurable behaviours of the future product, which introduces the quality measure into the design process.

The next stage, conceptual design, is very important because decision made here affects most the overall design process. On this stage uncertainty and limited knowledge about the product and big design freedom as well, are the major keywords. The danger here is that designers tend to take the first idea and start to refine it towards the desired product. This is a poor methodology because there is great possibility that this idea is not optimal or the best one at all. Therefore it is very important to generate at least more than one or two design candidates [1].

One goal of this stage is to choose best alternatives with the least expenditure of time and other resources. Techniques generating concepts and making decisions are used interactively as knowledge is increasing with new ideas [1]. The iteration is less expensive during this phase than on the following phases. The result of this stage is usually system architecture, functional and behavioural specification of the future product.

In the conceptual design process, designer takes the problem statement and generates a broad solution for it in the form of schemes, block diagrams, etc. On this stage engineering science, practical knowledge, commercial aspects, production methods and other relevant aspects need to be brought together, and the most important decisions are taken [2].

Although the design process varies from product to product and industry to industry, it is possible to draw out generic conceptual design diagram describing the activities have to be completed for most projects (Fig. 1).

![Fig. 1. Process of conceptual stage design](image)

Last two stages are well described in ref. [1].

A good scheme for describing mechatronic design process is the V-shape lifecycle model (Fig 2.). The model is based on the traditional waterfall process model but has a set of useful properties in description in the sense that relations between different design stages can be easily illustrated. This paper focuses mostly on the first and second phase of the V-shape model (specification and conceptual phase) as emphasized on above.

![Fig. 2. V-shape model](image)
In mechatronics system the general structure can be divided as following: basic system, sensors, actuators and control unit. Important aspect is also operative environment. The basic system comprises in general a mechanical, electrical and/or pneumatic, hydraulic combination. Sensors are elements determining selected state variables of the basic system and environment. Sensors are the input suppliers of the control units. The control unit determines the effects necessary to influence the state variables of the basic system in the desired way. The implementation of the effects takes place by actuators directly on the basic system. The interrelationship between the basic system, sensors, actuators and control unit can be effectively described by means of flows. In principle, a distinction has been made between three types of flow: material flow, energy flow and information flow [7].

![Diagram](image)

**Fig. 3. General structure of a mechatronics system in terms of flows.**

Material flows: Examples of materials which flow between units of mechatronics system are solid bodies, objects under test, objects being treated, gases or liquids.

Energy flows: Energy is to be understood in this connection as meaning any form of energy, such as for example mechanical, thermal or electrical energy, but also variables such as force or current.

Information flows: Information exchanged between units of mechatronic system are for example measured variables, control pulses or data [5].

The basic system of a mechatronic system comprises units which are linked by all three types of flow. At the forefront here are generally energy and material flows. The flows which connect the basic system and the environment to the sensors and actuators have both the character of energy flows and information flows, since energy flows both for the measuring and for the acting, but on the other hand the information is also transmitted. The control unit uses the information flows of the sensors and provides itself information flows for the actuators. The control units are often interconnected by communication system and can share resources or functions. The communication with system users may take place via man-machine interface. In both cases communications are represented as information flows [5].

Flow based ideology described above has been selected for object-oriented modelling approach, which is dealt in this paper.

### III. Software Tools for Modelling

Computer aided design software has been developed in very wide range of different domains and different levels within many years already. There are proved software tools for electrical, mechanical, control and other engineering domains with very wide functional scope. There is covered engineering calculus and analysis like strength, reliability calculus and several other topics with different methods like finite element method (FEM), numerical analysis and others. On recent years several mechatronics oriented software packages are entered on the marked. These software tools are dealing more complex problems compared with task oriented software tools. Some software packages worth to point out are 20-SIM, Dymola, Dynast, MatLab/Simulink toolboxes, etc. Although there are many software tools oriented to the different stages of the product design process, there is significant lack of the software tools for the conceptual design stage. This is quite natural, because this stage has a very big uncertainty and most of the conceptual work is done in the brains of designer. Modelling human creative thinking is quite impossible task for machines even nowadays and the fact sets limitations for the conceptual stage oriented software too. Due to increase of computational power there is nowadays possible to exploit some biological analogies in modelling artificial intelligence design tools. Techniques under discussions are artificial neural networks (NN), genetic algorithms (GA), multi-agent systems (MAS).
These techniques enable to develop semi-automated systems for the conceptual design stage where big uncertainty and variety of conditions are common factors. Develop of the methodology based on these techniques is one of the future targets of current work.

IV. Object-Oriented Approach for Conceptual Design

There are several modelling methodologies in the field of mechatronics under the development today. Most of them are based on a particular technique and have the limitation of this technique as well.

Modelling complex mechatronic systems like most of the real-life products are there is unreasonable to concentrate only on the one particular method. Mechatronical problem needs a more complex approach and one of the possibilities is to combine different suitable techniques and methods.

Our modelling approach exploits UML Use Case, UML Class Diagram and Bond Graphs analogy for defining the system and Genetic Algorithms combined with Neural Networks for optimization of the system.

To represent the system architecture we are using Bond Graph analogy. Bond-graphs are a domain-independent graphical description of dynamic behaviour of physical systems. This means that systems from different domains, like electrical, mechanical, hydraulic, acoustic, thermodynamic, material, etc. are described in the same way. The basis is that bond graphs approach is based on information and energy exchange. Therefore bond-graph modelling is a tool for modelling engineering systems, especially for mechatronics systems as interdisciplinary domain. Bond-graph modelling is a form of object-oriented physical systems modelling and therefore quite convenient to proceed. Furthermore, because the bond-graph submodels can be re-used as they are non-causal the approach suits for describing mechatronic systems submodels. The submodels can be seen as objects in its own, reused, stored in databases and combined. Analogies between domains are more than just equations being analogous: the used physical concepts are analogous [4].

Based on the Bond Graphs Bond elements hierarchal structure a class based metamodel is created. The top classes are analogous to Bond Graph elements in addition of some classes like class “Info”. The main differences between these two approaches is that a bond element represents some physical phenomena or detail but by our approach the class (analogous to the bond element) can represent a detail, phenomena as well as a complete device or module of the mechatronic system. In the conceptual stage it is very important to leave some design solutions open and do not to describe the future system on very detail level.

In very beginning of the design process the design problem is defined by the customer needs, which are usually not very clear or engineering oriented. Therefore these needs have to be formalized without losing important information in this process. Our approach offers a unique problem formalization method based on the UML Use Case analogy for fast and easy problem definition. The main objectives for creating this method were easy to use, both by engineers and customers, well known industry standard analogy and an open format of data description.

The desired product and/or functional description provided by the customer are practically all information we have in very beginning. Now this needs to be formalized and the engineer has to generate some conceptual solutions, preferably more than one alternative solution candidate. A metadiagram of the mechatronic system in conceptual level is in Fig. 4. [8]

![Fig. 4. Conceptual Metamodel Diagram (UML Use Case analogy)](image)
A formalized problem definition, shown on Fig. 4, exploits UML Use Case analogy with some exceptions and some extensions. On the formalized description the desired system is bounded with box containing ovals (use cases), which represent the system components. This is the principal difference from the UML Use Case as there ovals represent system processes. The ovals consist of the name of the component and under the line the name of the metatype, described in section VII. The symbol of the stick figure is called an actor. An actor portrays any entity (or entities) that perform certain roles in a given system. An actor in a use case diagram interacts with a use case (oval). An actor in a use case diagram depicted outside of the system boundary if it is not the part of the system [6].

Modelling steps

- Main components of the new system will be determined (should be robust and suitable for managing by a beginner engineer or even for customer itself).
- Metatype class will be assigned for every component where possible. Metatypes are defined by the system and are analogous for Bond Graphs element with some additional types.
- All components will be connected with each other guided by energy, information or material transfer ideology. The direction of connections will be determined as well.

The result of these steps is Conceptual Diagram accordingly metamodel in Fig. 4.

Now the aim will be to generate a more detailed system model by a semi-automatic process.
- The conceptual model will be transferred to BG type model
- Model will be validated
- Missing and confliction issues will be repaired by the user or by the algorithm.
- Back to step 2 until the designer and the validator will be satisfied.
- Final model with detailed component structure will be composed.

The result of these steps is validated and Block Diagram as system architecture which can be developed and analyzed further with engineering software described in section III.

Component Metamodel Database

A component database and a metamodel are needed by the algorithm, performing the automatic combination task for the mechatronic system structure. The database used, consist of different mechatronic system components and their relations.

In this research some versions of component structures have been developed in current stage though the work is continuing. These versions have some common features like the hierarchical structure, generalization, I/O ports and metainfo. Differences are basically in the level of abstraction, transfer function and variable definition.

Hierarchical structure enables us to classify the components and to create the component metamodel. Classification may be done in several ways, for example domain oriented, physical phenomenon oriented or even behaviour oriented. One important feature in our hierarchical model is relations between different classes. The idea is analogous to the UML ideology. Relation between classes can be: association, aggregation and generalization/inheritance.

The Base classes are:
- Source
- Resistor
- Capacitor
- Inertial element
- Transformer
- Gyrator
- Info
- Universal (represents all others)

All the base classes have many subclasses representing more precisely some technical solutions or existing devices. Sub classes inherit all parameters and behaviour from the parent.

Component Structure

The component represents an abstraction of the real system detail or assembly. On conceptual level components are, in most cases, unified system elements with some generalized functionality. Every component has one or more input and output port, which enables to communicate with other components. Ports carry through also the information and/or energy from one component to another. Transferred information is usually represented by one or more variables. In our approach the component ideology is analogous to Bond Graphs.
Fig. 5. A graphical representation of a component

Input/output parameters are effort and flow of energy, which enable energy transfer through the component. Types of the components used are analogous with the metamodel classes, described in previous section.

V. Conclusions

An important question of design process is: The actual needs of the customer are most important question has to be solved before start of the design process. The problem, how to put together customer needs and design result where uncertainty and lack of requirements specification is present, have to be solved quickly and effectively in actual design process. Here the new systems, based on artificial intelligence methods can play a significant role.

An overview of the new approach for advanced mechatronics modelling in conceptual design stage is discussed and described. Development of the method, new algorithm and design rules have to be further evaluated and tested in real design. Genetic algorithm together with neural network schema based algorithm have to be developed for automated and optimized mechatronics design methodology, which is the objective of our continuing work.

Acknowledgments

The work is supported by Estonian Science Foundation Grant 5908.

References

[5] VDI 2206 Design methodology for mechatronic systems, ICS 03.100.40; 31.220, June 2004