Available online at www.sciencedirect.com
ScienceDirect
Procedia CIRP 52 (2016) 24 – 29
www.elsevier.com/locate/procedia

Changeable, Agile, Reconfigurable & Virtual Production

Standardized Classification and Interfaces of complex Behaviour Models in Virtual Commissioning

Sebastian Süß[a,*], Dominik Hauf[a], Anton Strahilov[b], Christian Diedrich[c]

[a]*Daimler AG, 89081 Ulm resp. 71059 Sindelfingen, Germany*
[b]*EKS intec GmbH, 88250 Weingarten, Germany*
[c]*Otto-von-Guericke-Universität Magdeburg, 39106 Magdeburg, Germany*

[*] Corresponding author. Tel.: +49-731-505-2459. *E-mail address:* sebastian.suess@daimler.com

---

Abstract

Today's increasing use of Virtual Commissioning during the development process of automated manufacturing plants paired with the increasing request towards better control quality leads to the need of improved virtual plants with more effortless set ups. The common techniques of simulating the plant within Virtual Commissioning do no longer fulfil these needs, new approaches have to be developed. This paper examines ways to standardize Functional Mock-Up Unit based behaviour models of mechatronic components of such automated manufacturing plants. It is argued how such components can be classified to reach a distinction between different types to be able to develop standardized interfaces for every type. Therefore a standardized framework of how these interfaces can look like is proposed. Based on this framework as well as the classification of the components two examples, a pneumatic valve cylinder combination and an industrial robot are exemplarily implemented. Besides the standard interfaces to the control program and the visualisation of the simulation a special effort to implement energetically considerations were made. Therefore the presented work shows a way of how to standardize the interfaces of behaviour models of different classes of mechatronic components while increasing the quality of these behaviour models for more complex and accurate behaviour simulation of production plants for Virtual Commissioning as well as related applications.

Peer-review under responsibility of the scientific committee of the Changeable, Agile, Reconfigurable & Virtual Production Conference 2016

Keywords: Virtual Commissioning; behaviour model; behaviour simulation; functional mock-up interface; co-simulation; digital factory; simulation

---

1. Introduction

Today's evermore trend towards individual products, higher assortments and shorter life-cycles is not only changing the product development process but also the development process of automated production. It has to be more flexible to handle the rising product portfolio which ideally has to be produced on one production plant. One result of this flexibilization is that the usage of mechatronic components combined with control software is continuously rising and ensuring a high flexibility of a plant. Thus mainly software and barely hardware has to be adjusted when new products are going to be produced using the same plant. This trend leads to the fact that the control software developer spends more time for developing, optimizing and testing the control programs.

Virtual **C**ommissioning (*VC*) provides the control software developer with an approach how to develop and test his software based on a virtual model of the production plant [1]. This way of developing control software involves several benefits including more robust, higher quality and better level of maturity of the control programs, earlier and faster ramp-up of plants due to improved programs, higher optimization capabilities since it is easier to elaborate the virtual plant and several more. Even though VC is at the moment about to become standard in the development process of automated production plants [2], the behaviour simulation for VC (extensive described in [3] and [4]) is still a topic of research. The next step in the vision of being able to obtain the behaviour of a component not only as specifications in hard copy but also as a digital behaviour model, first presented in [3] and comparable to the evolution from 2D paper drawings to the deployment of 3D CAD data years ago is presented in this work. Therefore the approach of **co-s**imulation (*CS*) based on the FMI-Standard [5] was taken into account. As described in [3], this method enables the deployment of behaviour models from component manufacturers for the behaviour simulation. The single models are then co-simulated to simulate the behaviour of the plant as shown in figure 6.

2. Former work and motivation for this Paper

In [3] and [4], the importance of behaviour models of components for VC is extensively discussed. Moreover, the contribution [3] gives explicit both views of modelling and using of behaviour models by component manufacturers and users of

Peer-review under responsibility of the scientific committee of the Changeable, Agile, Reconfigurable & Virtual Production Conference 2016
doi:10.1016/j.procir.2016.07.049

VC. Thereby, the challenge that a component manufacturer is facing in modelling and providing behaviour models is debated as well as a way of how he is able to share its behaviour models with its users.Another challenges for the manufacturers is to define interfaces (*in- & output variables*) of their behaviour models user independently.

Regarding users of VC, the contribution [3] distinguishes between common users, plant manufacturers and service providers of VC. Common users (*e.g. OEMs*) are the users that contract out the development of production plants by plant manufacturers. A plant manufacturer, for his part, uses VC to validate the control programs of production plants. Both may also assign a service provider to conduct VC for them. The behaviour models however strongly depend on common user standards that regulate how components have to look like and how to use them in a production plant.

In [3], a distinction between behaviour models created by component manufacturers, referred to as **M**anufacturer **B**ehaviour **M**odels (*MBM*), and common users, referred to as **U**ser **B**ehavior **M**odels (*UBM*) is made. Thereby, *MBMs* are based on know-how of the manufacturer, and are modelled user independently. Moreover, *UBMs* can contain one or more *MBMs* and additional functionalities. The reason of this interleaving of *MBMs* into *UBMs* is to create the possibility to adapt *MBMs* to the requirements and standards of the users of VC.

Currently, *UBMs* are only created by the user of VC without *MBMs*, as these do not exist. Initial thoughts about the definition of interfaces of MBMs are one of the main topics of this contribution. Furthermore, the aim of this paper is to develop and present the needed interfaces to use *MBMs* for various simulations. To demonstrate this, the usage of the **F**unctional **M**ock-up **I**nterface (*FMI*) standard and its corresponding models called **F**unctional **M**ock-up **U**nits (*FMU*) are considered.

As an extension of the preliminary work, various classification systems have to be analysed regarding the possibility to classified each *MBM*. With this classification, the interfaces (in- & outputs) of each class of component (e.g. cylinder) can be defined. Thereby, the component manufacturer can provide his components as *MBMs* with standardized in- and outputs independently to customers respectively users. From a users view, an exchange of a *MBM* into a *UBM* can be done quite automatically with the help of standardized interface definition. Consequently, the modelling time could be reduced. Based on this, the analysis of various classification systems is briefly described in this work. Furthermore, a necessary interfaces framework (in- & outputs) of *MBMs* is presented and two components from different classes are taken as an example.

3. Taxonomy of mechatronic components

Prior to be able to define standardized interfaces for behaviour models, it has to be spotted which kinds of behaviour models are conceivable within VC. To do so, two approaches have to be considered. On the one hand, the common used components within automotive production plants have to be identified (including the classification of these components) to ensure that all currently needed components are considered. On the other hand, classification methods available across different disciplines have to be observed to enable a standardized classification of the used components.

3.1. Existing internal structures and components in companies

Since VC is used for the validation of the common engineering process, the application of behaviour models is state of the art, like described before. At Daimler for example, VC is one part within the *integra* automation standardization framework. Therefore the automation specialists have done some standardization work and classified the particular models within a logical, applicable system. The main groups are divided up as follows: Conveyor technique, subsystems, process engineering, general functions and special functions.

3.2. Appropriate methods of classification

There are a lot of different methods and approaches available to reach a classification of different objects. These methods are basically independent from single structures already existing in companies. Common industrial standards as well as different academic proceedings across different disciplines are taken into account and are described in this section.

3.2.1. Product specifications standards

With the ongoing digitalisation of the industrial world, a various amount of product specifications standards, have been introduced and developed during the last 20 years. The main goal of such standards during their development was the usage of faster and easier handling within sales, marketing and administration departments. Nevertheless, this standards can also be used for technical purposes and their usage in this field rose during the last years. The most common systems are [6], [7], [8]: eCl@ss, ETIM, GPC, proficl@ss and UNSPSC.

After the identification of the most common product specifications standards on the market, they have to be assessed. Therefore the five categories internationality (how many languages are supported, how many national consortia are available, etc.), dissemination on the market (how common is the usage), appropriate scope (are all needed elements available), appropriate structure (is the structure good to use for the focused use case) and appropriate structural depth (is the structure deep enough for the focused use case) were taken into account.

The result of this exploration was that out of the considered product specification standards eCl@ss is the most appropriate for the needs to classify mechatronic components for standardized interfaces of behaviour models in VC.

3.2.2. Multivariate analysis

Multivariate analysis is a method of multivariate statistics which analyses objects considering not only one but multiple variables of the objects. Within the multivariate analysis, there are two main methods, structure-identifying as well as structure-verifying methods. Obviously only structure-identifying methods have to be taken into account to create a taxonomy for mechatronic components. However, many of such methods are existing, e.g. factor analysis, cluster analysis, neural networks, multidimensional scaling and correspondence analysis [9], [10].

The most appropriate methods to classify mechatronic components seem to be the so called clustering or cluster analysis and neural networks. The other mentioned methods are more common to visualize and classify complex variables and not objects.

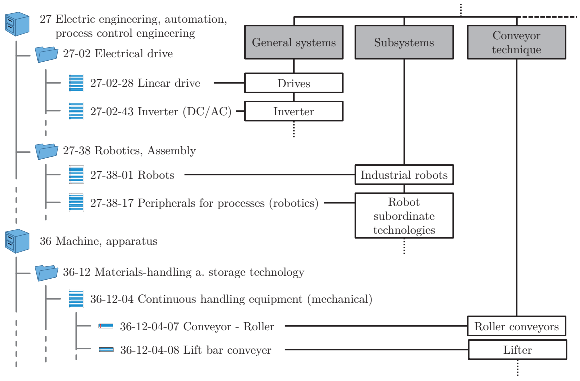

Fig. 1: Combination of *integra* and eCl@ss structures for mechatronical components in automated automotive production plants

3.2.3. Methods of biological taxonomy

Classification, systematics and taxonomy are not only very common within biology, this discipline even invented these methods. Therefore it is more than logical to take a closer look at existing methods of biological taxonomy. The taxonomy of the systematic in biology can be divided into different groups, for example linnaean taxonomy, evolutionary taxonomy, numerical taxonomy or cladistics [11], [12], [13]. Since the classical systems like linnaean or evolutionary taxonomy are rather hardly applicable to non-biology disciplines, they are not taken into account here. Numerical taxonomy and cladistics however seem to have the right requirements to be used to classify mechatronic components. Both take the features of an object to calculate the classification (numerical taxonomy) or evolutionary history and relationships (cladistics) through an algorithm.

3.3. Derived classification for mechatronic components

Concluding the chapters above, clustering and cladistics are two methods which can be use to classify mechatronic components. However, both of them need some effort to deliver suitable solutions which is intense discussed in [14]. Since the eCl@ss system is the best fitting product specifications standard available ready to use, and the focus of this work is not the classification itself but the usage of a classification for further considerations, eCl@ss is used within the rest of the work. Therefore it has to be matched to corresponding standards which might be available within different companies. Using the introduced *integra* standard as an example, the main group structure of that does not directly match to the segments or main groups of the eCl@ss system. Nevertheless the groups of the *integra* standard can be linked to main groups of eCl@ss. This means that the main groups of the *integra* standard constitutes a level in between the segments and main groups of eCl@ss. To use the standardized and common eCl@ss standard, the internal main groups and groups of OEMs and other users of VC can be linked to the (different) main groups, groups and classes of eCl@ss. An exemplary implementation of this approach using the *integra* standard is given in figure 1.

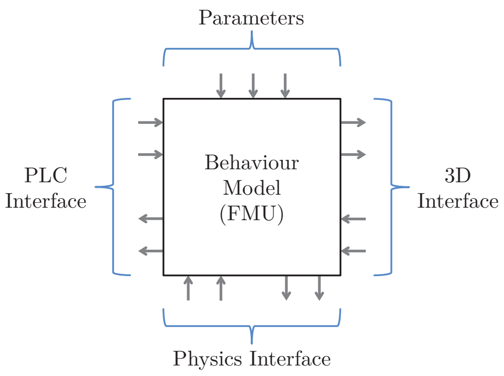

Fig. 2: Standardized interface-sections of any behaviour model

4. Standard interface of a FMU and its exemplary implementation

After being able to classify different components of automotive production plants, interfaces for the corresponding behaviour models have to be defined. Prior to defining these interfaces for every class of component, a general framework has to be developed where all needed interfaces have to be considered and which describes the general requirements of these interfaces. As mentioned in [3], [15], [16] and [2] VC is about to become quite usual in plant life-cycles and therefore the behaviour simulation of its components needs to get more accurate. Not only logical and temporal elements but also differential equations and a connection to external calculated physical values are required more often. At the same time the common interfaces to the PLC as the element controlling the whole plant and the 3D model visualising the plant also have to be considered. This leads to the division of the interfaces and its in- and output signals to the four sections Parameters, PLC interface, 3D interface and Physics interface. While parameters are only inputs to the model and a change during runtime of the simulation is not considered, all other interfaces can have in- or output signals. The division of the interfaces into the four sections is also shown in figure 2.

4.1. Example one: Cylinder

Looking on a pneumatic cylinder, it should be determined that each cylinder is controlled by a pneumatic valve. This connection between valve and cylinder is not visible for the controlling PLC. The PLC only knows both end positions (*retracted & extended*) of the cylinder piston with the help of sensors. In practice, one valve is often used to control several pneumatic cylinders simultaneously. This raises the question of how to model the behaviour of such component constellations. From a manufacturers point of view, each component should ideally be modelled separate. This ensures that the manufacturers' component models are independent from potential users. As a result of this distribution, each valve and cylinder should be used as a separate behaviour model for the co-simulation. To this end, the variable connection between behaviour models has to be configured for the simulation. Some of this variables are only needed between valves and cylinders, e.g. the air

Table 1: In- & output variables of a valve cylinder combination behaviour model

variable	direction	type	class	unit
left throttle	input	float	param.	[*%*]
right throttle	input	float	param.	[*%*]
system air pressure	input	float	param.	[*bar*]
piston load	input	float	physic	[*N*]
valve control extend	input	bool	PLC	[*0/1*]
valve control retract	input	bool	PLC	[*0/1*]
piston position	output	float	3D	[*m*]
piston force	output	float	physic	[*N*]
piston velocity	output	float	physic	[*m/s*]
piston extended	output	bool	PLC	[*0/1*]
piston retracted	output	bool	PLC	[*0/1*]
air flow	output	float	physic	[*l/s*]

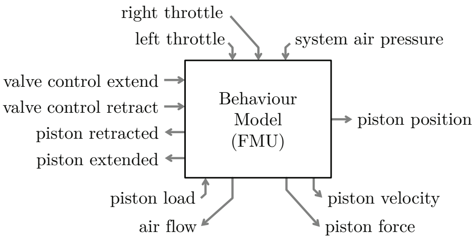

Fig. 3: Standardized Interface of a valve cylinder combination

flow from a valve to a cylinder respectively cylinders by a pneumatic tube. The problem with this is not the increased count of variables but the impure variables as well as the number of behaviour models whose connections have to be configured. Proceeding from this, an useful constellation of the behaviour models of an pneumatic cylinder is to summarize the valve behaviour and cylinder behaviour into one behaviour model. Additional, the behaviour of the air flow into the connecting tube should be added into the same model.

Current experience shows that for a pneumatic cylinder behaviour model, determined input respectively output variables are needed to use the behaviour model for various simulations. In this context we distinguish between variables like the piston position and parameters like system air pressure, right and left throttle, etc. [17]. There are also variables which are needed to calculate the energy consumption of the cylinder, like air consumption and air flow. Table 1 shows all in- and output variables of the behaviour model of a pneumatic cylinder.

Additionally, it is also important to distinguish between variables and settings of the pneumatic cylinder. A setting of a pneumatic cylinder has to be made when creating the behaviour model and can not be changed after that (e.g. piston area, piston stroke, piston mass, etc.).

The main objective of the behaviour model of a pneumatic cylinder is to use it for various simulations. The challenge here is to establish the possibility to use the behaviour model even if not all variables are given. The variable *piston load* for example depends on the mass and inertia force of the moved part, assuming that a physic based simulation is needed to simulate the *piston load* during the simulation. In VC, as one simulation application, a physic based simulation is used in specific cases only. Hence, a pneumatic cylinder behaviour model should be able to run without a value for the variable *piston load*. The same applies to the input parameters *left throttle*, *right throttle* and *system air pressure*. To achieve this, default values have been defined (see table 1, *default*).

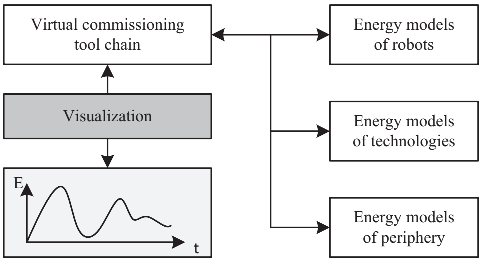

Fig. 4: Integration of energy models into the virtual commissioning tool chain [18]

4.2. Example two: Industrial robot

From an energy simulation and modelling point of view [18] already described that a VC model exists of three main groups (figure 4). Standardized energy models, also *MBMs*, can be divided as follows: robots, robot subordinated technologies and periphery. [18] wrote that first the single electric components of an automated production system must be described energetically. In the area of body-in-white manufacturing of an a automotive production there are industrial robots, robot subordinate technologies like welding, punch riveting, adhesive bonding and handling and all peripheries, like lifter cross conveyors, roller conveyors and so on. For many of the required energy models the state of the art can be used as basis. Thus many approaches for modelling the electric behaviour of industrial robot exist, these models have to be further developed for the use in **e**nergy **V**irtual **Com**missioning (*eVCom*)[19]. Simulation models for subordinate manufacturing technologies only exist in terms of process modelling. The energy consumption modelling in future has to be pushed by the technology manufacturers. It would be worthwhile that in future for every technology also standardized energetic simulation models are provided which can be connected to the eVCom model above predefined interfaces (like described above for behaviour models). For the area of periphery already existing models, like energy models of servo drives, can be used. They must be qualified for the eVCom. In the area of periphery there are often single-unit productions, so it is a challenge to get the individual properties of all periphery into simulation models [18].

The second example shows how a standardized *MBM* can look like for energy consumption modelling of an industrial robot (with n axes) within the VC tool chain. As described in [18] energy models should be integrated in VC to visualize and optimize the whole energy consumption of a production cell. This approach requires good and reliable energy models of all electric consumers, for example robots, technologies and periphery (figure 4). Not only behaviour models but also energy models can and should be provided by the manufacturers. So there has to be defined an interface for the *FMU* black box energy model and the executing co-simulation (figure 5).

In case of an industrial robot wiht n axes some in- & output variables have to be interchanged. For the calculation of energy consumption the position of every axis ($A_1 - n$), the moments of

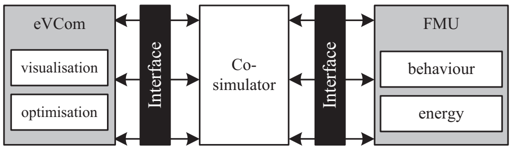

Fig. 5: Connection between robot *FMU* and eVCom

Table 2: In- & output variables of industrial robot energy model

variable	in or out	type	unit
time stamp	input	float	[ms]
position axis 1	input	float	[rad]
⋮	⋮	⋮	⋮
position axis n	input	float	[rad]
moments of inertia (tool)	input	float	[$kg \cdot m^2$]
moments of Inertia (part)	input	float	[$kg \cdot m^2$]
external forces	input	float	[N]
power consumption axis 1	output	float	[J]
⋮	⋮	⋮	⋮
power consumption axis n	output	float	[J]
power consumption of cabinet	output	float	[J]
sum of power consumption	output	float	[J]

Fig. 7: industrial robot with tool

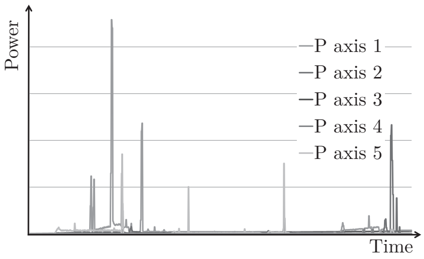

Fig. 8: Power of the six robot axis simulated via CS in [Watt].

inertia of tool and component part and external forces resulting for example from welding process are needed every time step Δt (in this case $\Delta t = 12ms$, because of the robot cycle time). The output variables should be the electrical power values of all axes ($A_1 - n$)

In case of industrial robot energy modelling there are ambitious efforts both from manufacturers and customers to extend an existing standard for *Realistic Robot Simulation* (RRS [20]) to energy consumption modelling. A first draft of this new extended version of the standard and first software releases of energy calculating robot simulation tools are provided until now [19]. The objective of this research is to integrate already existing standards into this new concept of standardized behaviour models for VC.

4.3. Simulated example

Based on the architecture of virtual commissioning, figure 6 shows an example of a virtual commissioning of an industrial robot with an attached handling tool. The tool is build-up with six pneumatic cylinders and a lot of mechanical parts (figure 7). The six pneumatic cylinders (*MBM Cylinder 1-6*) and the robot (*MBM Robot*) are modeled as FMUs and simulated via a co-simulator (*CS*) in real-time. This CS is connected with the software tool *RF::HMI* via a shared memory connection. With the help of this tool, in- and output signals can be manipulated respectively visualized (e. g. controlling of the valve of a pneumatic cylinder, setting of the system air pressure, visualizing both end-positions of all cylinders, showing air or rather energy consumption of the components, etc.).

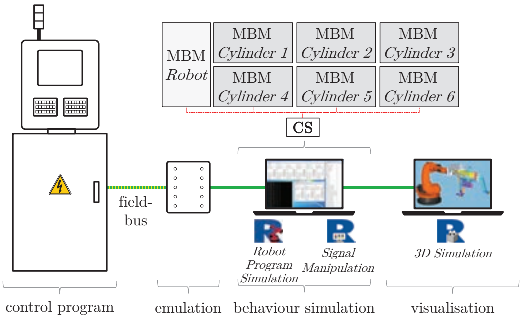

Fig. 6: Structure of the simulated example

As previous well explained, the *MBM-FMU* Robot expects position values of the six axes during the simulation which depends on the robot program. To provide these axis positions, the robot program is simulated with the tool *RF::RobSim*. Thereby, *RF::RobSim* calculates the current axes values regarding to the robot program for each time step. These calculated values are transmit to the *CS* via a shared memory connection.

Independently from the *CS*, the axes values calculated by *RF::RobSim* are transmitted to the 3d geometry model of the exemplary plant to visualize movements of the robot. At the same time, piston positions of each pneumatic cylinder are calculated by a *MBM-FMU* of each cylinder and transmitted to the 3d geometry model. The task to visualize the 3d geometry model and simulate the kinematic of the plant, in real-time is realized by the tool *RF::SGView*.

The following part of this section presents some simulation results of the *CS*. The first diagram (figure 8) shows the calculated power of each robot axis via *CS* over time. Here the used behaviour model is a *MBM*.

Furthermore, the second diagram (figure 9) shows the via *CS* calculated air flow of one of the six pneumatic cylinders over time. In this current example, the pneumatic cylinder was extended with an air pressure of 6 [bar] and maximal open throttles in both directions.

5. Conclusion and Outlook

This paper shows an approach of how standardized FMI-based behaviour models can be classified into a mechatronic

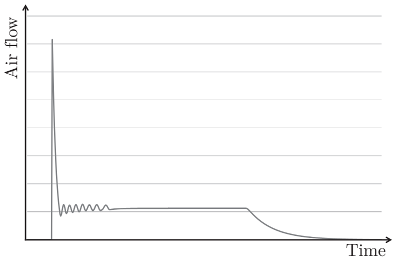

Fig. 9: Air flow of one pneumatic cylinder simulated via CS in [standard liter].

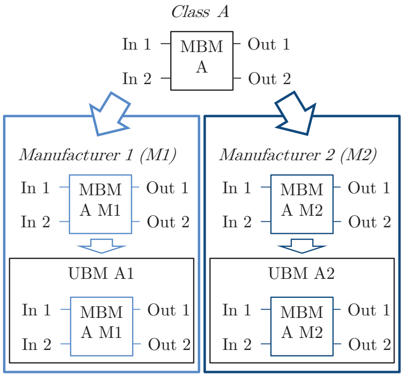

Fig. 10: Concept of supply of a standardized and classified simulation models

scheme. Common industrial standards as well as different academic proceedings across different disciplines have been investigated and evaluated for subdividing such mechatronic simulation models. Different advantages and disadvantages of the systems have been discussed. The very best system, however, has not jet been detected, but the basic need for such a classification has been presented. Additionally an approach for standardized interfaces for simulation models of classified mechatronic components has been suggested.

In combination with the results of [3], figure 10 shows the concept of a standardized and classified simulation models. Each component manufacturer (in this case *M1* and *M2*) uses the appropriate class (in this case *A*) for his product to create and provide his FMU simulation models with corresponding interfaces as *MBMs*. Thus different manufacturers can provide *MBMs* of the same class (in figure 10 *MBM A M1* resp. *M2*). Simulation models coming from different vendors but having the same function within the plant and are translated into *UBMs* are here named *UBM A1* resp. *UBM A2*.

Furthermore, in this work two examples of standardized FMI-based behaviour models are described in detail: a pneumatic cylinder and an industrial robot. Here each needed in- and output parameter is exemplary specified. In addition to that, the opportunity to integrate already existing standards to the presented new one is shown.

The objective of the future work will be further research in the area of classification. The presented systems an approaches have to be compared to each other in more detail also using some examples to identify and evaluate the most appropriate one. Moreover the considered standardization of each class (as soon as they all are identified) has to be further developed and transferred to an approved and valid standard.

ACKNOWLEDGMENT

We gratefully acknowledge financial support from the ITEA Programme under grant 12035, from the German Federal Ministry of Education and Research under grant 01IS13018G [21] and from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant 609391 [22].

References

[1] Drath, R.; Weber, P. & Mauser, N.: Virtuelle Inbetriebnahme - ein evolutionäres Konzept für die praktische Einführung. AUTOMATION, VDI Wissensforum GmbH, 2008
[2] Oppelt, M.; Barth, M. & Urbas, L.: The Role of Simulation within the Life-Cycle of a Process Plant. 2015
[3] Süß, S.; Strahilov, D. & Diedrich, C.: Behaviour Simulation for Virtual Commissioning using Co-Simulation. 20th IEEE Conference on Emerging Technologies and Factory Automation (ETFA), Luxembourg, 2015.
[4] Kiefer, J.; Ollinger, L. & Bergert, M.: Virtuelle Inbetriebnahme Standardisierte Verhaltensmodellierung mechatronischer Betriebsmittel im automobilen Karosserierohbau, AUTOMATION, VDI Wissensforum GmbH, 2009
[5] [Onile]. Available: www.fmi-standard.org
[6] Institut der deutschen Wirtschaft Köln Consult (Hrsg.): Klassifikationsstandards auswählen und einsetzen: Handlungsempfehlung zum Einsatz von eBusiness-Standards, Köln, 2011
[7] Brenner W. & Galuschka, T.: Klassifizierung in der Praxis. A. Bernecker Verlag, Melsungen 2008
[8] Hepp, M.; Leukel, J. & Schmitz, V. A quantitative analysis of product categorization standards: content, coverage, and maintenance of eCl@ss, UNSPSC, eOTD, and the RosettaNet Technical Dictionary Knowledge and Information Systems. Springer-Verlag, 2007
[9] Backhaus K.; Erichson B.; Plinke, W. & Weiber, R.: Multivariate Analysemethoden. Springer Gabler, 2016
[10] Hair, J.; Black, W.; Babin, B. & Anderson, R.: Multivariate Data Analysis. Englewood Cliffs, 2010
[11] Weberling, F. & Stützel, T.:Biologische Systematik. Wissenschaftliche Buchgesellschaft Darmstadt, 1993
[12] Leuschner, D.: Einführung in die numerische Taxonomie. VEB Gustav Fischer Verlag, Jena, 1983
[13] ElMaraghy, H. & AlGeddawy, T.; Laperrire, L. & Reinhart, G. (Eds.): CIRP Encyclopedia of Production Engineering, Chapter Cladistics for Products and Manufacturing. Springer Berlin Heidelberg, 2014
[14] Süß, S. & Diedrich, C.: Classification of mechatronic components for efficient plant behaviour simulation. 12th IEEE International Conference on Automation Science and Engineering (CASE), Forth Worth, 2016
[15] Oppelt, M.; Wolf, G. & Urbas, L.: Towards an integrated use of simulation within the life-cycle of a process plant. 20th IEEE Conference on Emerging Technologies and Factory Automation (ETFA), Luxembourg, 2015
[16] Bär, T.; Riedl, M.; Kärcher, B.; Hämmerle, H.; Beesten, H.; Pfeil, M. & Damrath, F.: Physikbasierte Simulation zur Unterstützung der Virtuellen Inbetriebnahme. AUTOMATION, VDI Wissensforum GmbH, 2014
[17] Strahilov, A.; Effmert, D.; Ovtcharova, J. & Bär, T.: Simulation of the pneumatic behavior in the virtual commissioning of automated assembly systems. 23rd International Conference Flexible Automation and Intelligent Manufacturing (FAIM), Porto, 2013
[18] Hauf, D.; Meike, D.; Paryanto & Franke, J.: Energy Consumption Modeling within the Virtual Commissioning Tool Chain. 11th IEEE International Conference on Automation Science and Engineering (CASE), Gothenburg, 2015.
[19] Meike, D.: Increasing energy efficiency of robotized production systems in automobile manufacturing. Ph.D. dissertation, Riga Technical University, Riga, 2013
[20] Realistic Robot Simulation, RRS-Interface Maintenance Group Std. Rev. 1.3. Sep 1997
[21] [Online]. Available: www.avanti-project.de
[22] [Online]. Available: www.areus-project.eu