Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 00 (2016) 000–000

www.elsevier.com/locate/procedia

49th CIRP Conference on Manufacturing Systems (CIRP-CMS 2016)

# Consistency check of the functional solution model in special purpose machinery

Tobias Helbig[a,b,]*, Johannes Hoos[a], Engelbert Westkämper[b]

*Festo AG & Co.KG, Ruiter Straße 82, 73734 Esslingen, Germany*
*Graduate School of Excellence advanced Manufacturing Engineering (GSaME), Nobelstraße 12, 70569 Stuttgart, Germany*

* Corresponding author. Tel.: +49-711-347-50725, *E-mail address:* tobias.helbig@gsame.uni-stuttgart.de

**Abstract**

Individual customer demands and increasing technical complexity are placing an even greater importance on the engineering process for special purpose machine manufacturers. To support the engineering process the Manufacturing System Dependency Model (MaSDeM) was developed. The basic idea of the MaSDeM concept is to install a cross-domain solution model at the beginning of the engineering process representing the principle solution. The building blocks for this model are functionally categorized automation components. However, since the resulting system is more than the sum of its components, the links between the elements need to be examined. In this paper a consistency check for the MaSDeM cross-domain solution model is proposed. This involves the identification of the different types of links between the components. The features of the links and the component categorization are used to build up a knowledge base for the consistency check. Thereby the static and procedural structure and the process functionality of the solution model can be verified. Moreover, the links can provide further engineering information. A profound principle solution is hugely important for the engineering process, and thus the consistency check is a vital contributor to increasing the efficiency of the engineering process in special purpose machinery.



*Keywords:* Engineering; Consistency; Model; Special purpose machinery

## 1. Introduction

Manufacturing in Europe is under a great pressure from structural changes in the global economy [1]. Providers of manufacturing systems must face two major trends. The first trend is brought about by the consumer market which demands individualized products and shorter product life cycles [2]. This creates the need to produce a large number of varieties on a manufacturing system and the need to quickly adapt the system to new products [3,4]. The second trend is the technical progress and the integration of information and communication technologies into manufacturing systems [5]. Whereas in the past manufacturing systems used to be characterized by the mechanical basic structure that was supplemented by several electrical components, nowadays innovations are mainly based on the cooperation between the domains mechanical design, electrical design and software [6].

In view of these trends, the challenge of engineering is to manage the complexity of individualized special purpose machines with a cross-domain engineering team. Hence engineering is becoming more and more important and its efficiency is a critical factor for success [7]. Three performance parameters are used to rate the efficiency of the engineering process in special purpose machinery: time, cost and quality [8].

In order to improve the cross-domain cooperation in special purpose machine manufacturing and thus to increase the efficiency of the engineering process, the Manufacturing System Dependency Model (MaSDeM) was developed [9]. The basic idea of the MaSDeM concept is to introduce a cross-domain solution model at the beginning of the engineering process. This is the stage during which the principle solution is created and thus most of the features are defined and fixed. As the cross-domain solution model forms the basis for the engineering process, it must be easy to understand for all participating domains and the consistency of the model must be guaranteed.

This paper presents an approach to verify the consistency of the MaSDeM cross-domain solution model by identifying contradictions in the interlinking of components and using the model to gather information. Therefore an overview of the MaSDeM concept is given in the next section which can be seen as the framework for the model checking approach that is presented afterwards.

## 2. Basics of MaSDeM

Special purpose machine engineering is a cross-domain challenge and entails a need for cooperation to reach the optimal mechatronic solution. Each domain brings its specific expertise to the process and contributes to making the system operational [10]. The customer has the specific knowledge of the product to be manufactured and the manufacturing process. In order to realize the process in a special purpose machine the engineering is assigned as appropriate to the domains mechanics, electrics and software. The mechanical design determines the geometrical structure of the system as well as the selection of the automation components. The wiring and communication infrastructure is within the responsibility of the electrical design and the software determines the logical sequences and implements the controller code. [11]

However, several problems can occur during the cross-domain cooperation. Each domain has specific engineering tools and models which hamper the exchange of information between the domains. That is why the domains mainly execute their tasks autonomously, leading to misunderstandings, errors and suboptimal overall solutions. [12]

Breaking down the walls between the domains by introducing a cross-domain solution model at the beginning of the engineering process is the basic idea of the MaSDeM concept. This cross-domain solution model is used as the common platform to discuss, harmonize and optimize the solution with all participants in the engineering process. [9] That is why the cross-domain solution model is very important as the basis for the engineering process and has a crucial influence on the engineering efficiency.

In addition to defining the cross-domain solution model, the MaSDeM concept also includes the integration of the solution model in the overall engineering process, but this part is beyond the scope of this paper.

### *2.1. Cross-domain solution model*

The cross-domain solution model consists of three tightly interconnected levels (Fig. 1). The process model contains the description of the manufacturing process for the product and all the customer requirements. In the layout model the realization of the manufacturing process is divided into single units, called stations, and the material flow between the stations is defined. The third level is the detail model. For each station the detail model represents the principle solution of how the process step can be realized within this station.

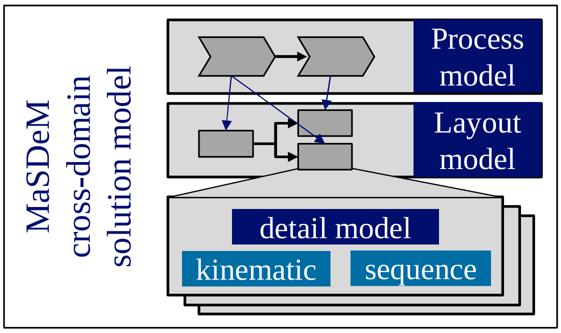


Fig. 1. Structure of the MaSDeM cross-domain solution model

The detail model consists of a kinematic model and an associated sequence model (Fig. 2). The kinematic model determines the geometrical structure of the solution. It is composed of automation components that are linked, creating a kinematic system and representing the active structure of the solution. Each automation component brings certain skills into the system. These skills are the basis for the associated sequence model which defines the succession in which the skills are executed. Thus the detail model represents the component-based solution that performs the station’s defined process step. [9] In the example of Fig. 2 a handling system with two linear axis and a gripper is shown. The simplified sequence for this handling process involves opening the gripper, going down, gripping the workpiece and going up again.

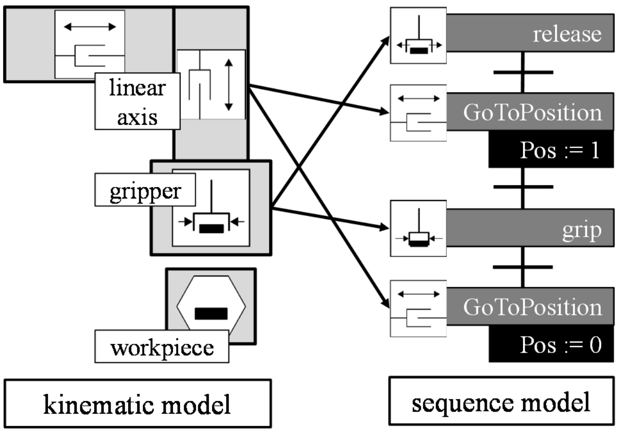


Fig. 2. Detail model representing the principle solution

The automation components and their associated skills are the building blocks to model the system’s principle solution. That is why a well-defined description language needs to be installed.

### *2.2. Functional categorization of automation components*

The description of the automation components in the cross-domain solution model has to be understood beyond the borders of the participating domains mechanical design, electrical design and software. That is why the function of a component is used as abstraction layer for the representation of automation components. [13] The function of a pneumatic cylinder, for example, can be described as a linear movement. This abstracts from the geometrical view of the mechanical design and the variable-based view of the software and can be used as a language for a common discussion and optimization of the system. In order to achieve a definite functional description the components need to be categorized [14].

Helbig et al. [13] propose a taxonomic hierarchy of the categories (Fig. 3). The basic idea of this categorization is to create generic elements at the top level of the hierarchy to make sure that any component can be categorized. The description becomes more precise with each sublevel so that a distinct description is provided at the lower hierarchical levels.

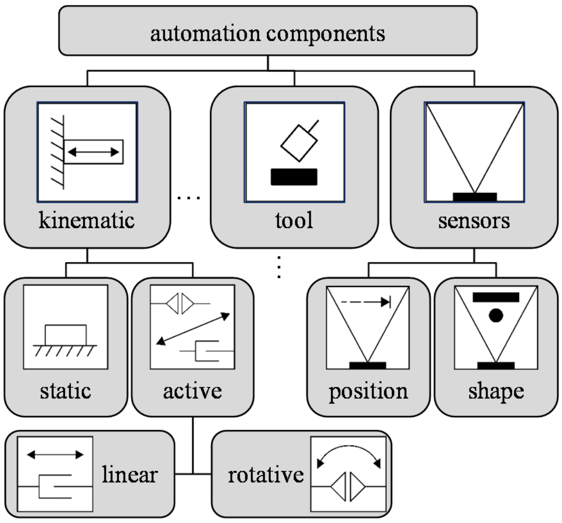


Fig. 3. Taxonomic categorization of automation components (extract)

Summarizing the MaSDeM concept it can be stated:

- All domains cooperate in order to create, discuss and optimize the cross-domain solution model.
- The solution model is based on automation components as building blocks.
- These components are functionally categorized to provide a well-defined description language.

The basic concepts of MaSDeM and the structure of the solution model are the framework for the consistency check approach presented in this paper.

## 3. Challenge and objective

The MaSDeM concept defines the cross-domain solution model and provides a set of categorized automation components as the building blocks to create the model. But providing well-defined building blocks is not sufficient to ensure the consistency of the model. A system is more than the sum of its elements, as only the links between the elements define the special character of the system [15]. The configuration of the components must be verified to ensure the consistency of the system and thereby to support the engineer when creating the model.

In this paper a model checking approach for the MaSDeM cross-domain solution model is presented. This includes an analysis of the types of links in the solution model. For each type of link a set of rules has to be defined in order to evaluate the proposed links in the model. The set of rules is used as the knowledge base to verify the consistency of the model. Verifying the quality of the principal solution, represented in the solution model, supports the engineering process and contributes to an increase in engineering efficiency.

## 4. State of the art

Model checking approaches are mainly applied in software engineering. To manage system complexity different views and partial models are used, entailing a risk that models are incomplete and inconsistent [16].

PROMELA (Process Meta Language) is a powerful specification language that is available to specify non-deterministic, communicating distributed systems [17]. In addition, PROMELA can also represent system behavior.

The SPIN approach [18] is based on the PROMELA specification and verifies the software design. SPIN can also be used to verify software architectures [16]. Nevertheless, the SPIN approach was designed to verify software design and not to verify mechatronic systems and it is therefore not directly applicable to mechatronics. However, the basic idea of using an abstracted meta model in order to check the model can be transferred to other settings.

In the context of special purpose machinery, Rauscher and Göhner propose abstracting the mechatronic model for verification [6]. Hardware and software components are connected with links that can either represent a dependency or a flow (of material, energy or information). The rule-based consistency check verifies the rules for the formal structure and the content of the model by comparing the features and parameters of the objects. However, apart from this basic idea and the meta-model, there are no specifications for the kind of information that can be compared and this approach can therefore only be used if the rules can be concretized in a certain branch. Kaiser et al [19] propose to model the components using their energy in- and output. The model check is based on a comparison of the energy flows to obtain information on the consistency of the system.

In summary, existing approaches for model checking mainly focus on the software engineering. In mechatronics, the ideas are limited to generic meta-models. There is a lack of concrete rules to verify the consistency of the model. That is why this paper develops a consistency check based on the model structure of the MaSDeM cross-domain solution model and the component categorization.

## 5. Consistency check in the cross-domain solution model

Only by consistent interlinking of single components one common system arises, fulfilling one overall function. Consistency can be defined as freedom of contradictions. Inconsistencies occur if consistency rules are violated. [20]

Checking consistency depends on the system aspect under consideration. The different views to the systems require the examination of specific types of links in order to execute a consistency check.

### 5.1. Types of links between components

Three characteristic views of a system can be distinguished [21,15]. The *functional view* focuses on the in- and output of the system. The *structural view* can be subdivided into the static structure and the procedural structure of the system. Finally the *hierarchical view* examines the hierarchy levels and their composition. Each view requires a certain set of links between components to build up this view. That's why these links must be identified to take them into account for the view-specific consistency check (Fig. 4).

The functional view requires the analysis of physical interactions with the workpiece. *Physical interactions* describe a defined, temporary interaction between a component and the workpiece with the aim of exerting an influence on the workpiece. There is no permanent physical connection between component and workpiece, they only get in contact for exerting the influence. A typical example is a tool that interacts temporarily to execute the defined process step.

For the static structure of the system the physical connections between the components must be examined. A *physical connection* indicates that two components are physically attached to each other permanently. Movements and forces are conveyed between the components.

For the procedural system structure the behavior of the components must be considered. Behavioral interactions stand for dependencies that components have on each other when executing their specific skills. An example for a behavioral interaction is the prevention of collisions between two kinematic elements having a shared workspace.

Finally the hierarchical view of the system can't be represented by links within the detail model. The hierarchy is represented in the connection of the detail model to the other layers in the MaSDeM solution model (Fig. 1).

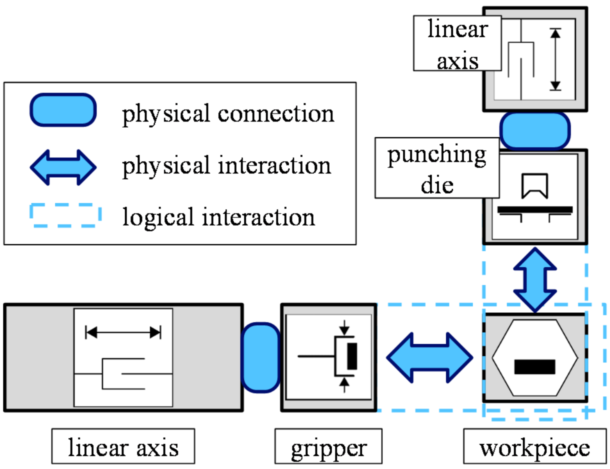


Fig. 4. Types of links between components

Each of the three types of link, physical connection, physical interaction and behavioral interaction, creates a certain view of the system and can be used for a certain consistency check in the specific view. The challenge is to create a set of rules for the links between components that serve as the knowledge base for the consistency check.

### 5.2. Verification of the static model structure

The physical connections build up the basic structure of the machine. Just focusing on this type of link analyzes the system configuration without considering the process that is executed on it. In order to check the physical connections the categorization of the automation components, presented in section 2.2, can be taken as the basis. Each single component that is used for modeling the structure in the kinematic model can be assigned to a well-defined category. The knowledge of the categorized components can be evaluated in order to perform the model check.

At the beginning the connectors of each component category have to be identified. Two types of connectors can be distinguished: basis connectors and transmission connectors. The basis connector describes the connector where the component under consideration is attached itself. Consequently each component has a basis connector as it has to be attached somewhere to build up the structure. The transmission connector models that further components can be attached to the component under consideration. Thereby it serves as the basis for the attached component. In contrast to the basis connector, not any component provides a transmission connector.

In Fig. 5 the connectors of the component category “linear axis” are shown as an example. A linear axis has two connectors. The static part of the axis is the basis connector where the axis is attached. The transmission connector is the movable part of the axis to which those components are attached that are moved by the linear axis.

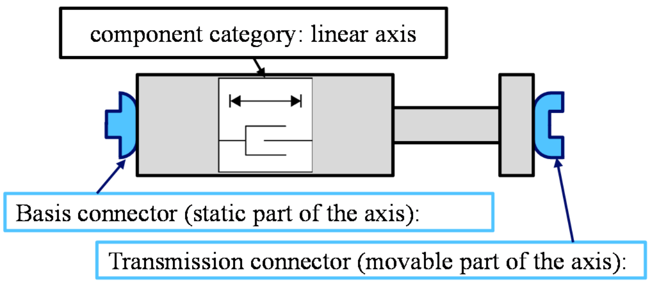


Fig. 5. Definition of the connectors of a component category

In the solution model the components are inserted and linked to each other. The challenge of the consistency check is to verify the links between the connectors of the components.

The first check verifies the correct modeling of the system. Each basis connector of each component in the system needs to be connected respectively to exactly one transmission connector.

The second check is about the connectivity between the components, verifying if the component types can principally be matched together. The rule base for the consistency check is derived from the description of the component categories and their connectors (Fig. 6). It defines which connections between components are allowed and excluded.

The linear axis is used once more to give an example for the rule-based consistency check. The basic connector, the static part, can be attached to a static element or to the movable part of another axis when being used in a gantry system. The transmission connector describes the components that can be attached to the moving part of the axis. In the rule base one can see, that another axis can be attached as well as grippers and sensors. But, it is not consistent to attach a bore tool as the bore tool needs a rotatory axis (Fig. 6).

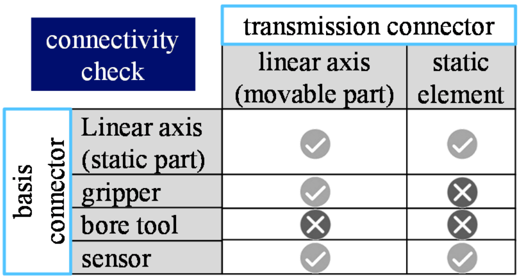


Fig. 6. Rule base for the consistency check (extract)

These consistency rules for each connector of each component category are the knowledge base to verify the structural consistency of the model.

## *5.3. Verification of the process functionality*

The objective of verifying process functionality is to find out whether the modeled structure is able to execute the specified manufacturing step on the product. This provides a consistency check for the functional view of the system.

As in the previous section, the basic idea is to define the necessary connectors and to specify the features for performing a rule-based model check. The physical interactions that model the processing of the product need to be taken into account in order to verify the process functionality.

Process functionality specifies the steps that are necessary to process the workpiece. Each product has its own individual description of the necessary steps, in contrast to the standardized connectors of the component categories that were used in the previous section.

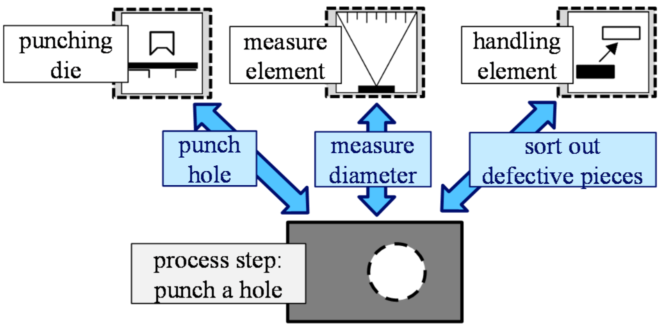


Fig. 7. Representation of a process step

Fig. 7 shows an example of the structure of a process step. The process step to be executed in this system is to punch a hole in the workpiece. To describe the process more precisely it was specified that after punching the hole the diameter of the hole must be measured and defective pieces must be sorted out. Each of the partial steps requires an interaction with some kind of component which is why the partial steps are added to the model as physical interactions. Furthermore, it is possible to specify which type of component is able to execute a particular partial step using the standardized categorization of the components. In the example, the partial step “punch a hole” needs a punching die to execute this step and “sorting out” needs some kind of handling element.

The process functionality of the modelled system can be verified with this description of the process step. First of all, each partial step must be executed which means that each partial step needs to have a physical interaction with a component. The second verification is to check the category of the linked component to make sure that it is able to execute the specified physical interaction.

## *5.4. Verification of the sequences*

The behavioral interactions are used to verify the consistency of the sequences and provide the procedural view to the system. Sequences define the succession of executing the skills. The skills are the actions a component can take, for example “go to a position” for a linear axis. For each component category there is a set of well-defined skills representing the capabilities of this component.

Analyzing behavioral interaction the skills of the two components that might come in conflict must be identified and rules for their interaction must be deduced. In terms of the example of two axes that might collide (Fig. 4) it can be stated that the second axis can move if the first one is in the initial position and vice versa.

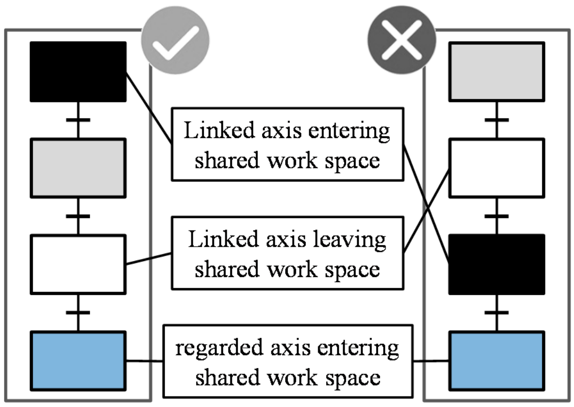


Fig. 8. Consistent (left) and inconsistent (right) process sequences

Before the axis under consideration enters the shared work space, the linked axis must have left this work space (Fig. 8). There must thus be at first a skill call for the linked axis to leave the shared work space in a consistent sequence. As the description of the sequences is based on well-defined component skills, the relevant skills for evaluating the shared work space can be identified. This knowledge allows the sequence to be verified and ensure that the behavioral interactions have been respected.

## 6. Deriving information from links

In addition to using the links between the different components to verify the consistency of the model, further information can also be derived from the system. Therefore dependencies can be analyzed that go beyond a direct link.

Fig. 9 shows an example for deriving information. A gripper is connected to the vertical axis of a gantry system. The requirement for the gripper is that the accuracy of its position must be within a certain tolerance in order to ensure the part is correctly gripped. Although the only direct, physical connection of the gripper is to the vertical axis, the horizontal axis also contributes to the positioning of the gripper. That is why the information about the accuracy of the gripper must be transmitted to all components that are relevant for its positioning. These components can be found by following the kinematic chain. Each active kinematic element is attached to another component with its static part. This component can be further analyzed until finally reaching a static kinematic element, for example a bearing. This means that the whole kinematic chain can be identified and the gripper requirement for a certain accuracy can be replicated to all components in the chain. The same procedures apply to torque and force generated by a tool that must be absorbed by the kinematic components downstream.

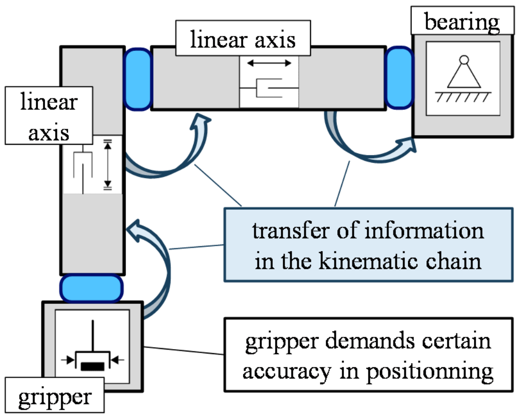


Fig. 9. Model-based system analysis

The linked components and their categorized types and features are not limited to being used for a model check, but they can also be used for a system analysis supporting the quality of information in the engineering process.

## 7. Summary and outlook

The MaSDeM concept uses the cross-domain solution model as the principal platform to discuss and optimize the solution with all participating domains. The model structure and the functionally categorized automation components as building blocks are the basis for the solution model, but it is the links between the components that create the configuration and specificity of the system. Verifying the links between the components, as presented in this paper, means to verify the consistency of the solution model. As the quality and consistency of this model is essential for the engineering process, the consistency check makes an important contribution to engineering efficiency.

## References

[1] Jovane, F., Westkämper, E., Williams, D.J., 2009. The ManuFuture road: towards competitive and sustainable high-adding-value manufacturing. Springer.
[2] Piller, F.T., 2007. Mass Customization, in: Albers, S., Herrmann, A. (Eds.), Handbuch Produktmanagement. Gabler Verlag, Wiesbaden, pp. 941-968.
[3] Aßmann, S., 1996. Methoden und Hilfsmittel zur abteilungsübergreifenden Projektierung komplexer Maschinen und Anlagen. Shaker, Aachen.
[4] Wiendahl, H.-P., ElMaraghy, H.A., Nyhuis, P., Zäh, M.F., Wiendahl, H.-H., Duffie, N., Brieke, M., 2007. Changeable manufacturing-classification, design and operation. CIRP Annals-Manufacturing Technology 56 (2), 783–809.
[5] Westkämper, E., 2013. Zukunftsperspektiven der digitalen Produktion, in: Westkämper, E., Spath, D., Constantinescu, C., Lentes, J. (Eds.), Digitale Produktion. Springer Berlin Heidelberg, pp. 309-327.
[6] Rauscher, M., Göhner, P., 2013. Agent-based consistency check in early mechatronic design phase.
[7] Westkämper, E., 2012. Engineering Apps: Eine Plattform für das Engineering in der Produktionstechnik. wt Werkstatttechnik online 102 (10), 718–722.
[8] Ehrlenspiel, K., Kiewert, A., Lindemann, U., 2013. Kostengünstig Entwickeln und Konstruieren: Kostenmanagement bei der integrierten Produktentwicklung. Springer, Berlin.
[9] Helbig, T., Erler, S., Westkämper, E., Hoos, J., 2015. Modelling dependencies to improve the cross-domain collaboration in the engineering process of special purpose machinery. CIRP Conference on Manufacturing Systems, 2015, Ischia.
[10] VDI-Richtlinie 2206. Entwicklungsmethodik für mechatronische Systeme. Beuth Verlag GmbH, Berlin 03.100.40. Accessed 8 January 2015, 118 pp.
[11] Frank, G., 2015. Durchgängiges mechatronisches Engineering für Sondermaschinen. Fraunhofer Verlag, Stuttgart.
[12] Schilke, M., 2010. Einsatz von Produktdatenmanagement-Systemen im Sondermaschinenbau für die Automobilindustrie.
[13] Helbig, T., Henning, S., Hoos, J., 2015. Efficient engineering in special purpose machinery through automated control code synthesis based on a functional categorisation. Machine Learning for cyber physical systems, 2015, Lemgo.
[14] Gehrke, M., 2005. Entwurf mechatronischer Systeme auf Basis von Funktionshierarchien und Systemstrukturen.
[15] Ropohl, G., 2012. Allgemeine Systemtheorie. Ed. Sigma, Berlin.
[16] Inverardi, P., Muccini, H., Pelliccione, P., 2001. Automated check of architectural models consistency using SPIN. 16th Annual International Conference on Automated Software Engineering, 2001.
[17] Kleuker, S., 2009. Formale Modelle der Softwareentwicklung. Vieweg + Teubner, Wiesbaden.
[18] Holzmann, G.J., 2004. The spin model checker. Addison-Wesley, Boston, Munich [u.a.].
[19] Kaiser, L., Dumitrescu, R., Holtmann, J., Meyer, M., 2013. Automatic Verification of Modeling Rules in Systems Engineering for Mechatronic Systems. Proceedings of the ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, 2013.
[20] Marr, S. Modellkonsistenz. Hasso-Plattner-Institut, Universität Potsdam. http://stefan-marr.de/pages/modellkonsistenz/. Accessed 18 November 2015.
[21] Landherr, M.H., 2014. Integrierte Produkt- und Montagekonfiguration für die variantenreiche Serienfertigung. Fraunhofer-Verl, Stuttgart.