A Classification Method for the Systematic Identification of Models and Workflows in MBSE

Gregor Hoepfner[1,*], Julia Kowalski[2], Clemens Faustmann[3], Thilo Zerwas[1], Philipp Kranabitl[3], Seyedmohammad Vafaei[1], Georg Jacobs[1], Hannes Hick[3]

[1] Institute for Machine Elements and Systems Engineering, RWTH Aachen University
[2] Chair of Methods for Model-based Development in Computational Engineering, RWTH Aachen University
[3] Institute of Machine Components and Methods of Development, TU Graz

** Corresponding Author:*
Gregor Hoepfner
Institute for Machine Elements and Systems Engineering, RWTH Aachen
Eilfschornsteinstraße 18
52062 Aachen, Germany
☎ *+49 241 / 80 95221*
✉ *gregor.hoepfner@imse.rwth-aachen.de*

Abstract

Modern engineering uses models for virtual verification of systems. Such models are usually combined in workflows, where the results of models are linked to verify system requirements. Model-Based Systems Engineering (MBSE) has evolved as an approach to ease the usage of models and workflows. One goal in MBSE is to reuse models and workflows from libraries. However, the step of identifying and classifying both models and workflows for such a library is not yet systematized. We propose a method on how to identify models and workflows for an MBSE model library. Possible purposes of models are identified and afterwards models satisfying that purpose are retrieved. The identified models are systematically combined to workflows. Thereby a systematic approach to create a model library is given.

Keywords

MBSE, Model Libraries, Model Classification, Workflows

 | DOI: https://doi.org/10.35199/dfx2022.17

1. Introduction

Nowadays new products are mainly cyber physical systems (CPS). CPS are characterized by interacting subsystems of the mechanical, electronic and software domain. Especially the interactions of the different subsystems and the varying development processes of the domains lead to an immanent complexity in the development of CPS [1, 2]. Furthermore, the global competition demands companies to develop CPS in a shorter time, with lower costs and in a higher quality in order to remain successful in the market [3]. The resulting techno-economic tension makes engineering CPS challenging and requires methods in product development that handle complexity while providing a shorter time-to-market.

A key strategy in mastering the complexity of CPS within the techno-economic tension is the virtual verification in product development using virtual behavior models. However, to manage the interactions in CPS, virtual behavior models of the single domains and subsystems need to be connected to enable virtual verification of the full system. An established method for seamless virtual verification of CPS including their interacting subsystems and models is Model-based Systems Engineering (MBSE) [4, 5].

In MBSE, system models provide a central architecture of the CPS under development. Within the system models, all kind of virtual models related to the system are interconnected and linked with their parameters. Within the system models, a functional architecture is derived from modeled requirements, which structures all functions hierarchically and represents physical dependencies between them through functional flows of material, energy, or signal [5]. As a link between function and product, solutions describe how and in which domain a function is realized. Here, a principle solution describes which physical effects, active surfaces and material properties can be used to convert the incoming into the outgoing functional flows [6]. Common ways for modelling system models may are the Systems Modeling Language (SysML) [7] or language profiles based on it.

While the described function-oriented system architecture can provide a structure for development of a system, for designing a solution's behavior, certain virtual behavior models are used. In mechanics, CAE models are used for this purpose, which represent a certain scope with respect to certain modelled purposes with a certain fidelity and can be executed in specialized software environments, such as Finite-Element models for fatigue testing of structural components. Such models need to be systematically linked to the principle solution providing solution elements or system solutions in the system model [4]. For specific, standardized solution elements and system solutions, such as the common machine elements bearings or gears, research has built up a large, heterogeneous, often unstructured landscape of CAE models over the past decades, which has mainly been used for component-oriented development. To use these models in the function-oriented development of mechanical extents of CPS, the existing models need to be assigned to the solutions in MBSE system models and thereby structured within the function-oriented system architecture. In addition, CAE models are usually combined with each other to generate answers to specific questions in product development. Such workflows on how to combine CAE models also need to be part of MBSE system models [8]. However, since the existing virtual behavior models are not structured or documented in a uniform and machine-processable way, the identification of relevant virtual behavior models and their integration into the system model can only be done manually, laboriously and error-prone by specific model experts and the physical content is not assessable. As there are typically a high number of already existing models, a structured method is required. Therefore, a central challenge for the reuse of models for the development of CPS is the systematic identification and structured integration, i.e. classification of existing virtual behavior models and the integration of workflows using these models into function-oriented, model-based system development.

In this paper we propose a novel approach to identify and classify virtual behavior models belonging to a solution element in MBSE and identify possible workflows for using such models. The paper is structured as follows: After providing the state of research in section 2, in section 3 the research questions are given and two methods are proposed: one for the identification and classification of virtual behavior models (section 3.1) and one for the combination of such classified models within workflows (section 3.2). Both methods are applied to the example of the solution element rolling contact. In section 4, we discuss on the two methods and section 5 concludes.

2. State of Research

Model-based development approaches reflect the megatrend of digitalization in product development. In general, the main objectives of applying model-based approaches are [9]:

- Higher system maturity in early phases through modeling activities that lead to better system understanding also in the beginning of development and therefore higher maturity, and
- Higher quality of the system under development by applying early verification and validation methods already in early stages of design.

One approach in this context is model-based systems engineering (MBSE). As the definition from INCOSE states [10] MBSE is the formalized application of modeling and models to support system development regarding different aspects throughout the lifecycle.

In many MBSE approaches, central system models are used [9, 11]. System models describe aspects relevant on system level and can be of descriptive or quantitative form on high system level. System models can be differentiated from domain-specific or discipline-specific models, but there are links between system models and domain-specific models. [12]

There are various MBSE approaches covering the development of interdisciplinary CPS [13-15]. One approach capable of also describing the mechanical domain within a CPS is given by Jacobs et al. [4]. Herein, solution elements provide a formal structure of reusable elements that represent the technical solution of one function. The solution elements include a functional description via principle solutions (incl. physical effect and active surfaces) following a formal SysML profile [5], corresponding models [16], and workflows. The defined structure of a solution element (Figure 1) is considered as the foundation for the presented research. The principle solution of a rolling contact is described by a principle effect, in this case by the Hertzian contact with its simplifications, boundary conditions and basic formulas. Principle effects are based on Koller's publications [17, 18].

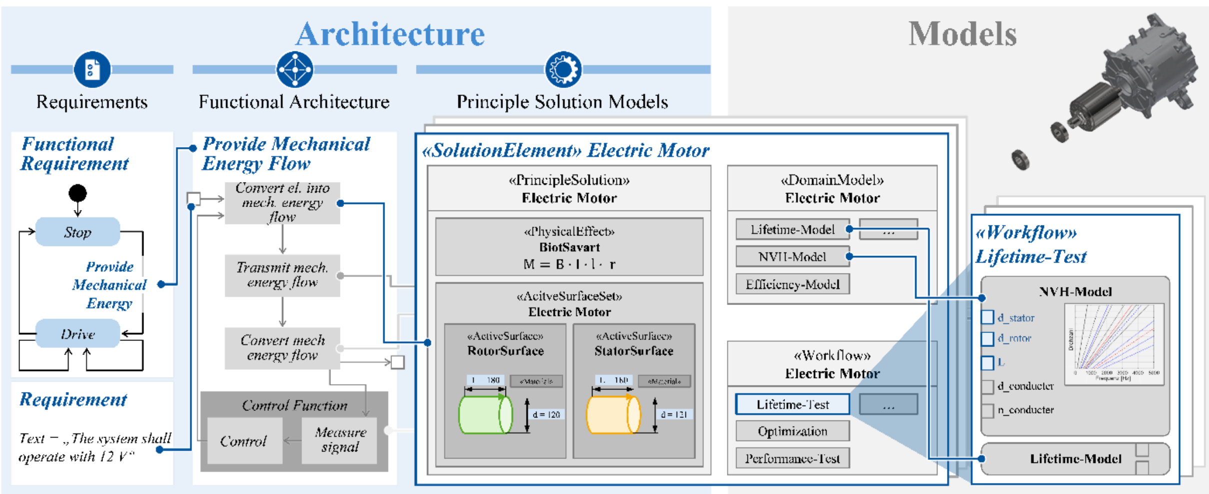

Figure 1: Architecture in a system model including solution elements [4]

The models within the solution element need to be structured in a way, that they can be reused easily and to ensure traceability and consistency [4]. To provide such a structure, there are various approaches for structuring different kinds of models in literature. Within these approaches, the term model itself is defined in different ways [19]. The definitions of Stachowiak [20] and Kossakiof et al. [21], can be considered as most commonly used definitions. According to Kossiakoff et al., a model “can be thought of as a simplified representation or abstraction of reality used to mimic the appearance or behavior of a system or system element”. [21]

For model classification and structuring, Stachowiak describes the three main characteristics of models (mapping property, reduction property and pragmatism). In the context of MBSE, the focus is on digital models [22]. Jockisch and Rosendahl describe models by their characteristics on a high level by the four groups graphical, technical, semantic and semantic-scientific [23]. Faustmann et al. provide a link between the system model and domain specific models using a three-dimensional cube consisting of discipline, technical domain and level [24]. Jacobs et al. [4] adapt the three main characteristics of Stachowiak for domain models in MBSE context by providing the three axes of model classification: scope (mapping property), purpose (pragmatism) and fidelity (reduction property). Jacobs et al. [4] recommend to use these axes to structure the models within a solution element.

The structure proposed by Jacobs et al. [4] is defined to classify models on a very deep level of the system. Many models on this level describe physical effects and are therefore of quantitative nature. For more general model structure on higher system level and also for project relevant models such as requirements models, other classification schemes for models are more applicable [12]. The purpose of a model from a user perspective can be considered differently. A model does not exist for representation purposes only, it also serves a purpose in the development process, e.g. the geometry modelled in a 3D CAD model supports to in providing manufacturable components and assemblies. Therefore, a model has a certain breadth and depth, by considering the views of different disciplines and by covering aspects on different levels of the system hierarchy [9 ,12].

To perform specific development tasks, models of different purpose are usually executed in specific orders. Bajzek et al. refer to such procedures of activities or tasks of model execution as methods and define them as one of the four interlocking pillars of systems engineering [19]. Here, a method describes a transformation of an input to an output with a certain purpose, for example the generation of a verification model. It requires a specified input, which can also be a set of models. One specific form of methods are simulation methods, which are defined as virtual manipulation and execution of one or arbitrary many models in a specified environment including boundary conditions. For example, the method *strength analysis using FEM* uses a CAD model and a load model as input in order to generate a model of stress and deformation. [22]

Especially in the context of virtual development, the term workflow is used synonymously for methods. In literature, a workflow is defined as a procedure how a specific activity is being done [25]**.**

Höpfner et al. [8] define workflows, which link single model execution activities within a system model to execute verification or design tasks during development [8]. Herein, workflows are semi-automated, executable procedures which hand over parameters between model execution activities. In the solution element, Jacobs et al. propose workflows as part of reusable elements, as workflows provide important knowledge on how to execute different models. [4]

In this publication, the focus is on models that are used to describe behavior aspects of a system element combined within a workflow.

3. Research Problem

The state of the art shows various methods on how to classify models within MBSE and how to link them in workflows. The solution element according to Jacobs et al. [4] provides a structure for the various simulation models. However, no method is provided to initially identify such models and workflows when starting to set up a solution element in MBSE. To identify models of different purposes for a solution element and combine suitable workflows from these models, a systematical identification and combination method is required. Such a method is not yet described in literature. Thus, we derive the following research questions:

1. How can the models within a solution element be identified based on physical effects?
2. How can workflows be combined and selected from the identified models in the solution element?

4. Proposed Method

For answering the two research questions, we propose a research method on how to initially integrate models and workflows into a solution element, cf. Figure 2. As a use case for applying the method, we use the solution element of a rolling contact. In a rolling contact, force is conducted on two interacting surfaces, which deform in normal direction of the contact. Initially we identify the different models of a solution element systematically from the physical description within the solution element and classify them (4.1). Afterwards we combine workflows from the identified models by systematically evaluating the model classification (4.2).

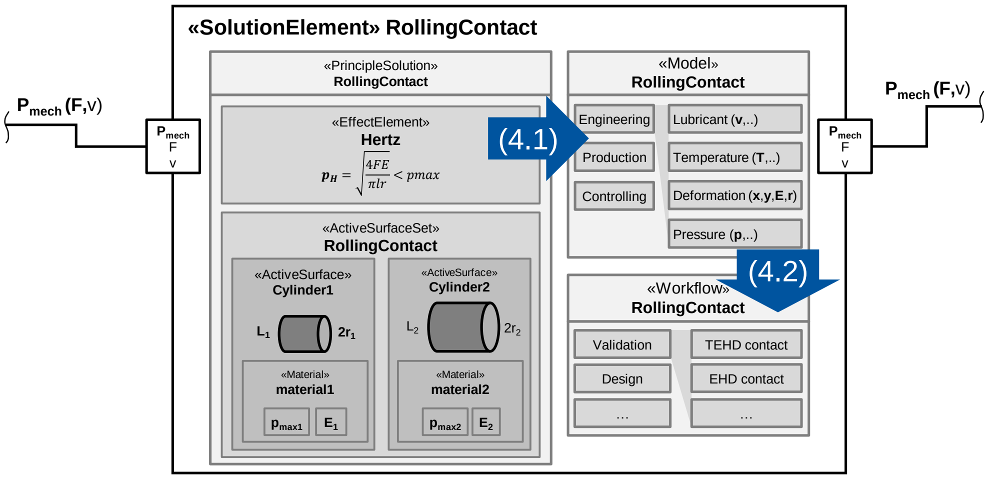

Figure 2: Methodological way on identifying models from physical effects (1) and combining workflows from the identified models (2)

4.1. Identification of Models from Physical Effects

A solution element as described by Jacobs et al. consists the principle solution, which describes how a functional transformation is performed from a physical effect, a collection of models and a collection of workflows. The models are classified by the aspects scope, purpose and fidelity:

- Model scope: Each model has a defined scope, which defines the solution elements and elements of a system it can be applied to. In this case, the model scope includes just one solution element, which is the rolling contact.

- Model purpose: Although a model contains various parameters and therefore includes different aspects, it has a certain purpose it is used for. The different aspects of the behavior of a solution element are the different model purposes in this case such as deformation, temperature, or pressure.
- Model fidelity: In order to describe the same purpose of a defined system, different models exist. The level of detail and the influencing factors included in the models determine a higher or lower fidelity.

As soon as the scope of a solution element is predefined – in the use-case it is the rolling contact – possible model purposes for the solution element need to be identified. The purpose of a model describes the physical behavior that is modelled within it. For identifying possible purposes, it is therefore required to find possible physical behaviors, i.e. physical effect that might be modelled. These are not only the physical effect in the principle solution but all effects that might occur in the solution element, e.g. fluid friction, elastic and plastic deformation, or thermal conduction. To identify further relevant physical aspects, a list of existing physical effects is required. Such a list is given in Koller's catalogue [18]. From this list, the relevant physical effects in a specific solution element are selected with the help of domain experts. While for example fluid friction is a significant influence in a rolling contact, the effect of the magnetic air gap can be neglected here. Koller's catalogue covers more than 200 physical effects in total. For the given use-case 56 physical effects are considered relevant for the rolling contact. From these 56 physical effects, model purposes can be derived. Table 1 shows the process of selecting the purposes from Koller's catalogue exemplarily for a reduced list.

Table 1: Selection of physical effects and derivation of purposes from Koller's catalogue

All physical effects	**Selected physical effects**	**Derived purposes**
Absorption		
Adhesion	Adhesion	Adhesion
Barnett Effect		
Centrifugal Force		
Dry Friction	Dry Friction	Dry Friction
Elastic / Plastic Deformation	Elastic / Plastic Deformation	Deformation
Electrohydraulic Effect		
Fluid Friction	Fluid Friction	Fluid Friction / Lubrication
Gravity	Gravity	Gravity
Impulse Theorem	Impulse Theorem	Impulse Theorem
Ionisation		
Joule Heating		
Magnetic Air Gap		
Pressure	Pressure	Pressure
Pyroelectricity		
Temperature Dependency	Temperature Dependency	Temperature
Thermal Conduction	Thermal Conduction	Thermal Conduction
Transformator		
Vaporization		

For the identified solution element and purposes, as a next step, models can be identified by literature survey or other investigation methods. For the given purposes from Table 1 multiple models can be identified. Figure 3 exemplarily shows four selected purposes from Table 1 and adds identified models to each purpose. As a last step, the identified models are sorted according to their modelling fidelity. This is done by evaluating the assumptions and restrictions of the models. Applied to the purpose pressure, simplifications can be identified as visible in Table 2. Following this, the Reynolds equation is the most simplifying model, while the Navier-Stokes equations cover the purpose pressure with the highest fidelity. After evaluating the assumptions for the different models, the relative fidelity of the models can be identified and a solution element with structured models is given. To identify fidelities quantitatively, test scenarios and a comparison to real data are required for each model and an absolute scale for the fidelity might be derived the. This is effortful, as there are values which are not directly measurable, such as pressure directly in a rolling contact. Quantitative scope estimation is out of scope of this paper.

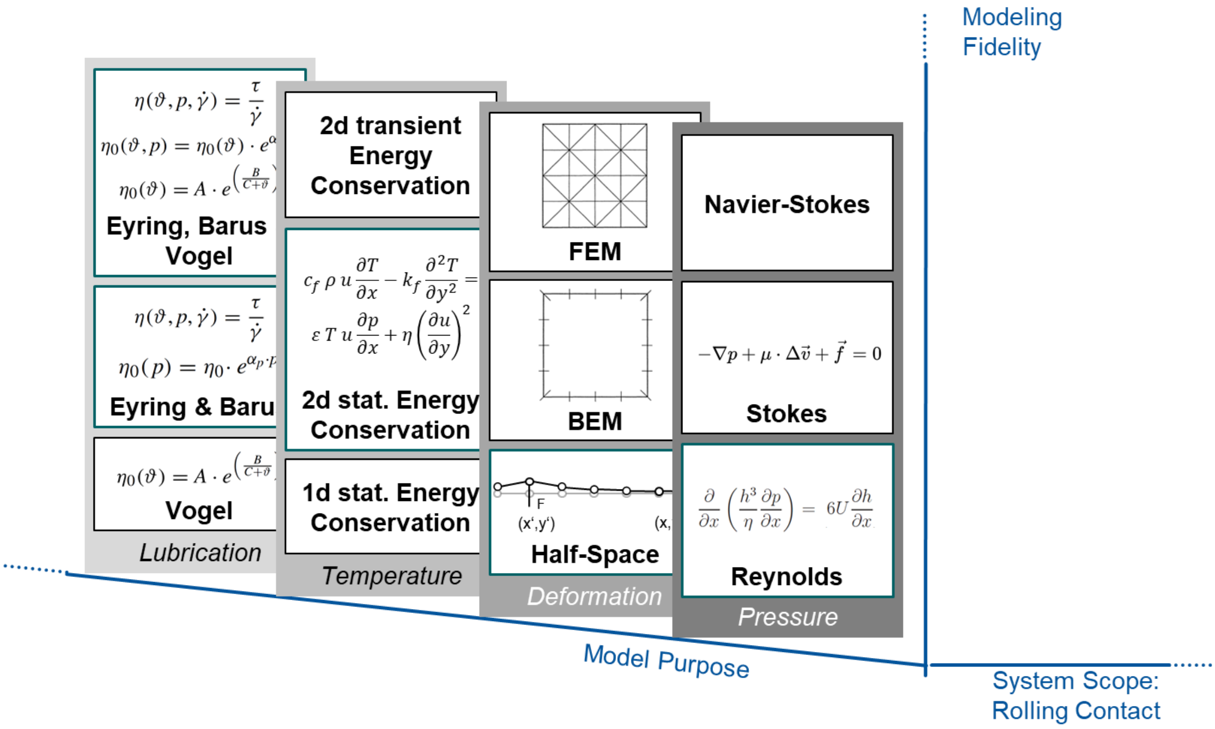

Figure 3: Structured behavior models for specific purposes

Table 2: Assumptions for different kinds of pressure models

Model	Large lubricating film height	Compressible Fluids	Fluid inertia considered	Pressure gradient considered	Time dependency	Convection considered
Reynolds	No	No	No	Yes	Yes	Yes
Stokes	Yes	Yes	No	No	No	No
Navier-Stokes	Yes	Yes	Yes	No	Yes	Yes

4.2. Combining Workflows from Models

To apply the identified models in product development, workflows are used, which put the execution of models in an order and connect the single models. A workflow combines models of different purposes one after another. For example, workflows for calculating elasto-hydrodynamic (EHD) behavior in a rolling contact lines up the purposes pressure, deformation and lubrication in a row (Figure 5, green), while the thermal elasto-hydrodynamics (TEHD) workflows also consider the purpose temperature (Figure 5, yellow). In the previous section, we have identified the models of specific purposes and thereby filled the model section of the solution element. From this model library, we can as a next step derive possible workflows for a given chain of purposes like EHD and TEHD. As workflows combine different purposes, all possible workflows for a given task are the cross combination of all models with each other. I.e. when searching for a combination of pressure and deformation purpose, possible workflows are the combinations of Reynolds – Half-space, Stokes – Half-space, Navier-Stokes – Halfspace, Reynolds – Boundary Element Method (BEM), and so on. Using the cross-combination method, a total of 27 workflows for EHD and 81 workflows for TEHD can be combined from the exemplary models. The cross-combination logic can be used to determine all possible workflows in a set of purposes and models.

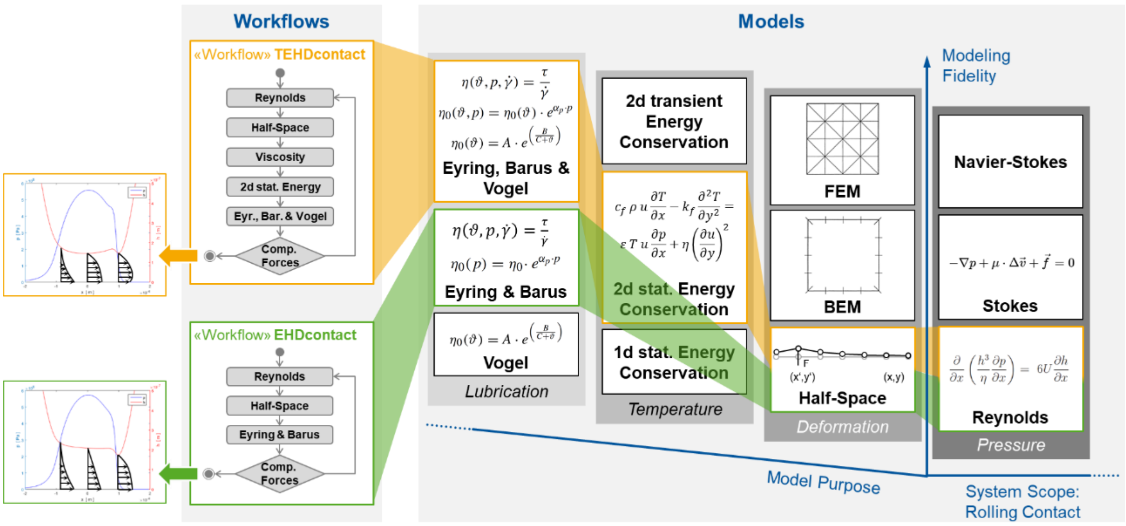

Figure 4: Illustration of workflows and connected models for the example of a rolling contact

The resulting workflows are of different fidelity depending on the models that are used. A workflow consisting of low-fidelity models has a lower resulting fidelity than one combined from only high-fidelity models. A qualitative filtering of workflows regarding fidelity can be done by evaluating the assumptions of the models, that have been documented in section 4.1. After combining all possible workflows, it is then possible to filter for workflows that only use models fulfilling specific assumptions. This provides a relative comparability regarding the fidelity of workflows. A quantitative comparability is still ongoing research.

5. Discussion

The presented methods provide a way to initially fill a model and a workflow library as they are often wanted for MBSE applications. To do so, the methods identify models and derive their possible usage in workflows. The methods are evaluated as described using the example of a rolling contact.

The methods describe a systematic approach on how to identify required purposes and the corresponding models in a solution element. Koller's catalogue as a list of physical effects is successfully applied to identify purposes by physical effects and a total of 56 possible purposes was identified from it. Exemplarily, twelve models for four different purposes are selected and classified. With the described cross-combination method, 27 possible workflows for EHD calculation and 81 possible workflows for TEHD calculation are combined. The wide array of possible workflows can be reduced when filtering for only specific models, e.g. only models resolving in three dimensions, or a minimum fidelity of a workflow. In total, the method can support in creating an initial model and workflow library for single solution elements in MBSE.

However, the method is mainly investigated on the level of deep physical effects. It has not been investigated yet how to apply such a method for higher system levels, e.g. for the bearing system in which the rolling contact is integrated. Here, solution elements are combined and other physical effects may occur due to emergence effects, such as complex heat transfer mechanisms. Hence, a differentiation by single acting effects may be hard. Further on, during the investigation, simplifications have been made. The evaluation has happened based on the described purposes and one solution element.

A quite difficult topic is the classification of both models and workflows regarding their fidelity. In the given approach, a relative comparison is chosen, which is based on evaluating the physical assumptions that have been made during modelling. For the given models, the approach was easy-to-use and sufficient. However, for other cases, especially when models or workflows are similar in modelling assumptions, it may reach limits. A quantitative comparison of both models and workflows regarding their fidelity might be required. However, such an approach might require the context and parametrization of models. A well parametrized model of low fidelity might provide more realistic results than a high-fidelity model with high deviation in parameter accuracy. Also, the workflow fidelity might depend on more aspects than just the model fidelity.

6. Summary and Outlook

In the present research, a method on how to identify and classify models for a MBSE model library using solution elements and how to combine workflows from such models is presented. For identifying the models of a solution element, the method uses Koller's catalogue to select relevant physical effects which are to be investigated within the solution element. From these effects, model purposes are derived. For each purpose, models are identified from literature. To define model fidelity, the physical simplifications and assumptions in the models are described and thereby the relative model fidelity is determined. Thereby, the identified models are classified and can be integrated into a solution element. For using the models, workflows are required, which describe the order in which models of different purpose are executed. It is demonstrated how possible workflows can be combined from the derived model library and how from all possible workflows, the relevant ones can then be selected.

The proposed approach provides a structured method for classifying models and workflows. However, there are still open questions. The method is only evaluated on the example of a rolling contact. Application to other solution elements and larger system solutions, such as bearings or full drives has not been investigated yet. For a more general validation, further solution elements, purposes and workflows should be considered. The fidelity of both models and workflows is only determined on a relative scale. Further research is required to derive a fully comparable and quantitative scale. In addition, the given classification approach structures the physical content of a model. For usage of the identified models, additional information is required, which might be given e.g. in a model signature. The given research may provide a starting point for investigating such questions.

References

[1] Broy, Manfred: Challenges in Automotive Software Engineering. In: Proceeding of the 28th international conference on Software engineering - ICSE ’06 (2006).

[2] France, Robert; Rumpe, Bernhard: Model-Driven Development of Complex Software: A Research Roadmap. In: Future of Software Engineering 2007 at ICSE (2007).

[3] Pahl, G. et al.: Engineering design. A systematic approach, 3rd ed. London: Springer, 2007.

[4] Jacobs, Georg et al.: Function-Oriented Model-Based Product Development. In: Krause, D.; Heyden, E. (Hrsg.): Design Methodology for Future Products Data Driven, Agile and Flexible. Springer, Cham, 2022, S. 243–263.

[5] Drave, I. et al.: Modeling mechanical functional architectures in SysML. In Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems (2020), pp 79–89.

[6] Zerwas, T. et al.: Mechanical concept development using principle solution models. In: IOP Conf. Ser.: Mater. Sci. Eng. (2020) 1097 (1).

[7] Object Management Group. OMG System Modeling Language Specification: Version 1.5. https://www.omg.org/spec/SysML/1.5 (accessed July 14, 2022).

[8] Höpfner, G. et al.: Model-Based Design Workflows for Cyber-Physical Systems Applied to an Electric-Mechanical Coolant Pump. In: IOP Conf. Ser.: Mater. Sci. Eng. (2021), 1097 (1).

[9] Friedenthal, Sandford; Moore, Alan; Steiner, Rick: A Practical Guide to SysML: The Systems Modeling Language. Bd. 38: OMG Press, Elsevier, 2009.

[10] International Council on Systems Engineering (INCOSE): Systems Engineering Vision 2020, 2007.

[11] Weilkiens, Tim: Systems Engineering with SysML/UML. Heidelberg, Germany: dpunkt. verlag, 2006.

[12] Hick, Hannes; Bajzek, Matthias; Faustmann, Clemens: Definition of a system model for model-based development. In: SN Applied Sciences Bd. 1, Springer International Publishing (2019), Nr. 9.

[13] Moeser G.; Albers A.; Kumpel S.: Usage of free sketches in MBSE raising the applicability of Model-Based Systems Engineering for mechanical engineers. In: IEEE International Symposium on Systems Engineering (ISSE) (2015), IEEE, pp. 50–55.

[14] Gausemeier J. et al.: Computer-aided cross-domain modeling of mechatronic systems In: DS 60: Proceedings of DESIGN 2010, the 11th International Design Conference, Dubrovnik, Croatia, 2010, pp. 723–732.

[15] Burmester, Sven; Giese, Holger; Tichy, Matthias: Model-driven development of reconfigurable mechatronic systems with mechatronic UML. In Model Driven Architecture. Springer, 2006, 47–61.

[16] Spütz, Kathrin et al.: Classification of Simulation Models for the Model-based Design of Plastic-Metal Hybrid Joints. In Procedia CIRP 109, 2022; pp 37-42.

[17] Koller, Rudolf; Kastrup, Norbert: Prinziplösungen zur Konstruktion technischer Produkte. Berlin, Heidelberg: Springer, 1994.

[18] Koller R.: Konstruktionslehre für den Maschinenbau: Grundlagen zur Neu- und Weiterentwicklung technischer Produkte mit Beispielen. Berlin, Heidelberg: Springer, 1998.

[19] Bajzek, Matthias; Fritz, Johannes; Hick, Hannes: Systems Engineering Principles. In: Hick, H.; Küpper, K.; Sorger, H. (Hrsg.): Systems Engineering for Automotive Powertrain Development. Cham, Switzerland: Springer Nature, 2021.

[20] Stachowiak, Herbert: Allgemeine Modelltheorie: Springer, 1973.

[21] Kossiakoff, Alexander et al.: Systems Engineering: Principles and Practice: Wiley, New Jersey, 2011.

[22] Walden, David; Roedler, Garry; Forsberg, Kevin: INCOSE Systems Engineering Handbook Version 4: Updating the Reference for Practitioners. In: INCOSE International Symposium (2015), Vol. 25. No. 1.

[23] Jockisch, M.; Rosendahl, J.: Klassifikation von Modellen. In: Bandow, G.; Holzmüller, H. H. (Hrsg.): Das ist gar kein Modell! Unterschiedliche Modelle und Modellierungen in Betriebswirtschaftslehre und Ingenieurwissenschaften. Wiesbaden: Gabler, 2010, pp. 23–52.

[24] Faustmann, C. et al.: System models and model classification in tribological system development. In: Systems Engineering 23 (2020), No. 6, pp. 783-794.

[25] Cambridge Dictionary. URL https://dictionary.cambridge.org/de/worterbuch/englisch/workflow. - retrieved on 04.07.2022.