INTEGRATED MODELING OF BEHAVIOR AND RELIABILITY IN SYSTEM DEVELOPMENT

Hentze, Julian (1); Kaul, Thorben (2); Graessler, Iris (1); Sextro, Walter (2)
1: Paderborn University, Heinz Nixdorf Institute, Germany; 2: Paderborn University, Mechatronics and Dynamics, Germany

Abstract

The integrated modeling of behavior and reliability in system development delivers a model-based approach for reliability investigation by taking into account the dynamic system behavior as well as the system architecture at different phases of the development process. This approach features an automated synthesis of a reliability model out of a behavior model enabling for the closed loop modeling of degradation of the system and its (dynamic) behavior. The approach is integrated into the development process following Systems Engineering. It is based on standard models used in model-based development methodologies i.e. SysML or Matlab/Simulink. In addition to the theoretical description of the necessary steps the procedure is validated by an application example at two stages of the development process.

Keywords: Design for X (DfX), Product modelling / models, Robust design, Systems Engineering (SE), Reliability

Contact:
Julian Hentze
Paderborn University
Heinz Nixdorf Institute
Germany
julian.hentze@hni.upb.de

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 21st International Conference on Engineering Design (ICED17), Vol. 4: Design Methods and Tools, Vancouver, Canada, 21.-25.08.2017.

1 INTRODUCTION

Today's technical systems offer a wide range of functionality which is enabled by system inherent intelligence. Since adding sensors, actuators and information processing mechanical systems have changed into mechatronic systems. The availability of powerful information processing and computation power enabled for networked drive further to intelligent systems. Not only the possibility of implementation of intelligent features leads to expansion or innovative development of existing systems, but also the necessity to fulfil wide-ranging requirements as well as to extend existing functions and features to customer or user. A digital layer was added upon the mechanical and electrical layer of the system, which offers new possibilities, e.g. autonomous adaption to changing environmental or system conditions. The digital layer creates dependencies between subsystem as well as advanced system aspects that strongly rely on software. Although, the system inherent intelligence offers promising perspectives for advanced functionality, it also threatens reliability because of the increasing system complexity. Taking advantage of the intelligence in operation strategy, autonomous behavior adaption of the system is possible, that can be used to adapt system behavior to the current reliability and in turn increase reliability during operation.

However, increasing system complexity makes the design process of such systems more prone to errors, e.g. common mode failures are major threats to reliable systems (Sagan, 2004). Thus ensuring reliability becomes considerably challenging and has to be taken into account in early design phase using a model-based approach. These aspects can be solved by advanced modeling techniques that support handling of large and complex systems as well as complex failure behavior. The design process for mechatronic systems according to (Gausemeier et al., 2003; Pahl et al., 2007) is considered to be an iterative process. To ensure reliability throughout the design process, a common approach is to generate a reliability model by hand, which is a laborious manual task and thus very error prone (see Figure 1) (VDI, 2004).

To overcome these issues, the integrated modeling of system behavior and reliability was introduced (Kaul et al., 2015) to support the design process by offering an automated synthesis of a reliability model. Thus it is possible to integrate reliability-related objectives into the operation strategy by using appropriate objective functions ($\underline{f_R}$) to compute feasible operating points (see Figure 1). Those operating points are a set of optimal solutions for different prioritizations of the contradictory objectives.

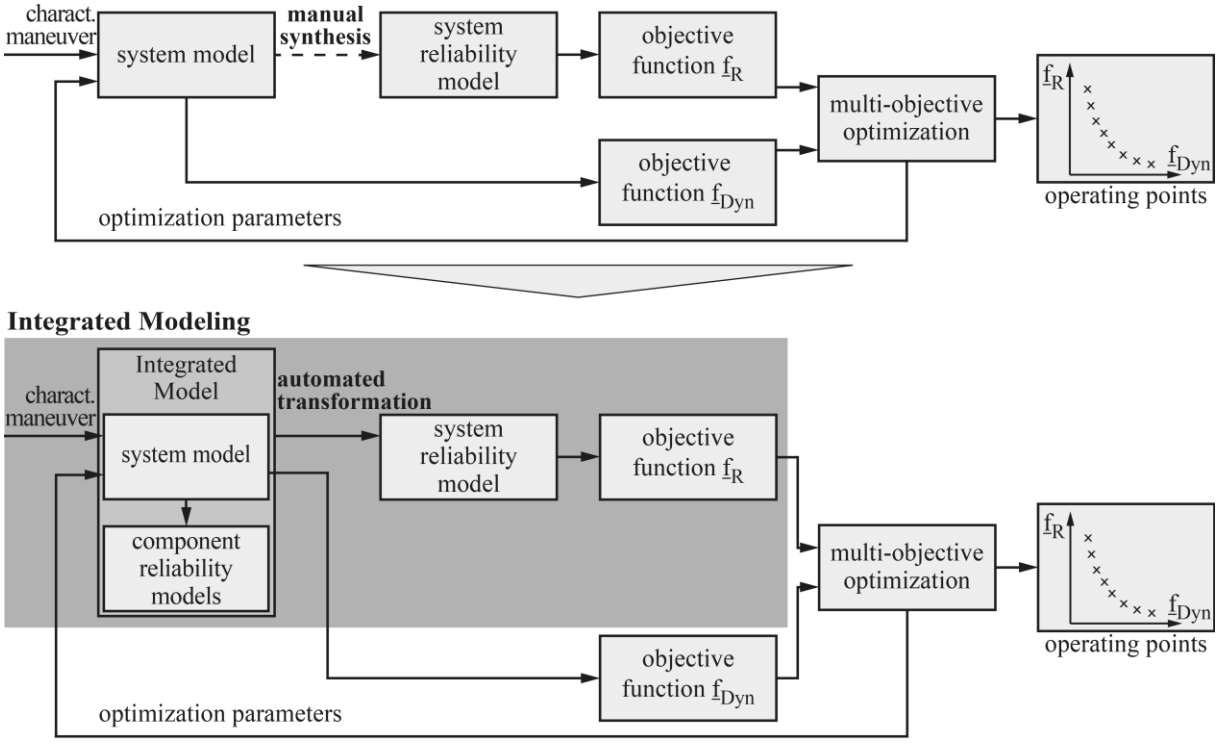

Figure 1. Comparison of modeling approaches for use of behavior- and reliability-related objective functions in multi-objective optimization

In order to efficiently apply the integrated modeling to the design process, this work aims the integration of these modeling techniques into the development and especially design processes.
The remainder of the paper is structured as follows: The following section introduces the current state of research regarding integrated modeling of behavior and reliability. In Section 3, the integrated modeling of behavior and reliability is briefly described. In Section 4 this technique is integrated into the development process using Systems Engineering (SE) and the corresponding process model is introduced. In Section 5, the integrated modeling is applied to the design and structure of a test rig, where the reliability of the test rig is evaluated at different design phases based on a system description model using SysML and a model of dynamic system behavior in Matlab/Simulink. The results are discussed in the last section.

2 MODELING SYSTEM BEHAVIOR AND RELIABILITY

State-of-the-art technical systems demand for advanced modeling methods that closely incorporate modeling of behavior and reliability. In current research, different approaches were introduced to ensure reliability and safety throughout the development process. Those approaches are based on description languages from multiple domains, e.g. SysML, Modelica, VHDL-AMS, LARES or Matlab/Simulink.
In order to exploit existing system models in system design for reliability and safety analysis, SysML-models are semi-automatically transformed into reliability block diagrams (RBDs) (Helle, 2012) and fault trees (FTs) (Xiang et al., 2011) respectively. Cressent introduced a synthesis of SysML-models into AADL-models (Cressent et al., 2010), that give the opportunity to formally analyze real-time and embedded systems. Different paper propose methods to support failure mode and effects analysis (FMEA) (David et al., 2010; Mehnni et al.) extended this method to semi-automatically perform fault tree analysis (Mhenni et al., 2014). Although, the mentioned methods offer a detailed reliability analysis based on SysML-models, they solely rely on SysML and are not suitable to investigate reliability based on the dynamic behavior of a system, e.g. it can be modeled using Matlab/Simulink or Modelica.
Schallert introduced Modelica libraries for simulation of dynamic behavior as well as for reliability and safety analysis of aircraft onboard electric power systems (Schallert, 2011). Reliability and safety analysis is conducted by evaluating automatically generated FTs or RBDs. Bestory describes electronic circuit behavior and component-based degradation models in VHDL-AMS. Statistical reliability analysis using Monte-Carlo-Simulations are exploited (Bestory et al., 2007). These two methods are restricted to electrical systems and are thus not capable of modeling overall mechatronic systems.
Walter introduced the LAnguage for Reconfigurable Systems (LARES) to investigate dynamics and reliability of fault-tolerant systems (Walter et al., 2009). While originally developed to model computer systems, it can also be used to evaluate reliability of mechatronic systems (Meyer et al., 2013b). A major limitation arises from the use of Markov Models, which limit the modeling scope due to their restriction to exponentially distributed state transitions.
Papadopoulos introduced a method to automatically synthesize FTs out of Matlab/Simulink-models. A description of the component failure behavior is obtained from a hazard and operability study (HAZOP) (Papadopoulos and Maruhn, 2001). This approach is restricted by the use of FTs, since they rely on Boolean methods. In contrast, Bayesian Networks as used in this work investigate reliability in state space and thus offer larger modeling capabilities.

3 INTEGRATED MODELING OF BEHAVIOR AND RELIABILITY

The dynamic behavior of a technical system determines the prevailing load on its components. The dynamic behavior of mechatronic systems is influenced by the chosen controller parameters, i.e. small eigenvalues of the system can be compensated in order to establish high system dynamics. This requires more powerful actuators that in turn increase the load on the components of the system. The degradation of technical components is strongly influenced by the loads. In turn, loads on individual system components highly influence system reliability.
The integrated model (Kaul et al., 2015) introduced a methodology for model-based investigation of this dependency by implementing an automated transformation of a behavior model into a reliability model. Dynamic Bayesian Networks are used as reliability model. The transformation algorithm is only briefly introduced in the scope of this work. For further reading please refer to (Kaul et al., 2015).
The integrated model comprises a system model and a reliability model (Figure 2). A topology-oriented modeling approach was chosen for the behavior model of the system. Thus, components are directly

represented, that provides good accessibility to the loads on components. The behavior model is evaluated for a characteristic maneuver, which represents the expected operation and environmental conditions of the systems. To each component, a degradation model (lifetime estimator) is added which estimates component lifetime for prevailing loads.

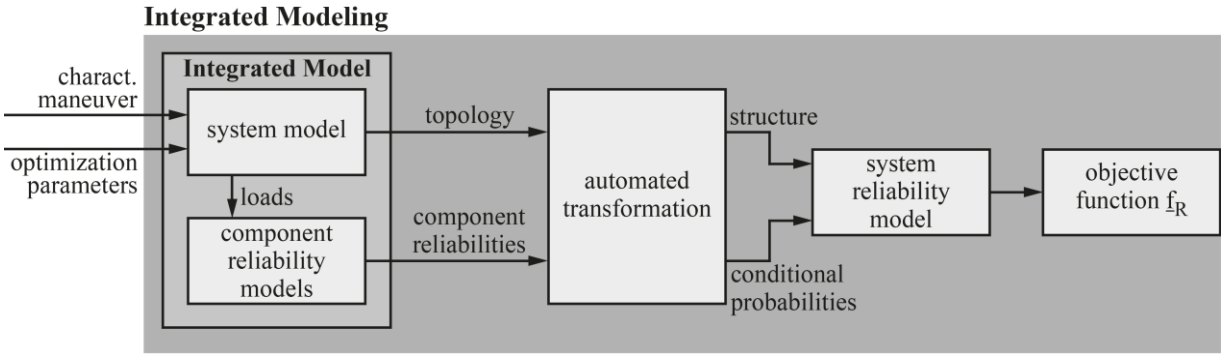

Figure 2. Model structure (according to (Kaul et al., 2015))

A transformation algorithm allows for autonomous generation of a reliability model from the system model which propagates changes within the behavior model to the reliability model. Changes can either be architectural changes in system topology or changes in parameters.
In addition, knowledge taken from a Failure Mode and Effects Analysis (FMEA) can be added to the integrated modeling approach in order to also cover failures that cannot be directly derived since they may not be referenced in the prevailing behavior modeling approach.
However, the generated reliability model requires qualitative and quantitative aspects, topology and component reliabilities. The component reliabilities are directly taken from the lifetime estimators, whereas the topology of the behavior model is analyzed to obtain the topology of the reliability model.
The results of the FMEA can be used to appropriately choose lifetime estimators for the identified failure modes of each component. The topology of the reliability model can be extended using the identified failure effects of the FMEA.
The integrated model is capable of analyzing different system behavior models, i.e. Matlab/Simulink, ADAMS, Dymola and SysML, while maintaining the original model by associating the lifetime estimators to the components in an external file. The model-based design of technical systems already provides these behavior models. Thus, the design process can be strongly supported using the integrated modeling by evaluating reliability in early design phase as well as providing reliability-related objectives for operating strategies in the following phases of the development process.

4 INTEGRATION INTO DEVELOPMENT PROCESS

The all known challenges of complexity, interdisciplinary working, multi-project management and high competitive pressure lead to the usage of development approaches as Systems Engineering (SE) and thereby to an increased focus on frontloading in development process (Walden, 2015; Gräßler et al., 2016). Important SE topics are described in the theoretic base of SE, the INCOSE Handbook (Walden, 2015). Architecture and design processes define the structure of the System of Interest (SoI). The architecture process aims the overall description of the system by subsystems, elements/components and their interfaces. The design process specifies the individual subsystems and elements/components integrating functions and features. To handle technical risks in topic of reliability, it is useful to analyze the system as early as possible to identify critical subsystems and elements/components (Walden, 2015). As early as possible means here, that there have to be first specification models for System of Interest (SoI) architecture. The approach for integrated modeling of behavior and reliability is able to support the decision for candidates of system architecture, which means the identification of the architecture that is as close as possible to the theoretical optimum (Walden, 2015). It provides an automated procedure for verification by reliability prognostics for a SoI or any other concerned systems. INCOSE describes this testing and simulation while architecture and design phases as "in-process validation" (Walden, 2015). The approach for integrated modeling of behavior and reliability thereby combines on the one

hand the system model with its elements and interdependencies with additional information about reliability of the individual elements and on the other hand the analysis of the system behavior.
Apart from the architecture process, the approach can be used in further phases of system development defined in Systems Engineering. As shown in Fig. 3 the behavior-based reliability analysis mainly takes place in domain-specific design at the bottom of the V-model (VDI, 2004; Kaul et al., 2015).

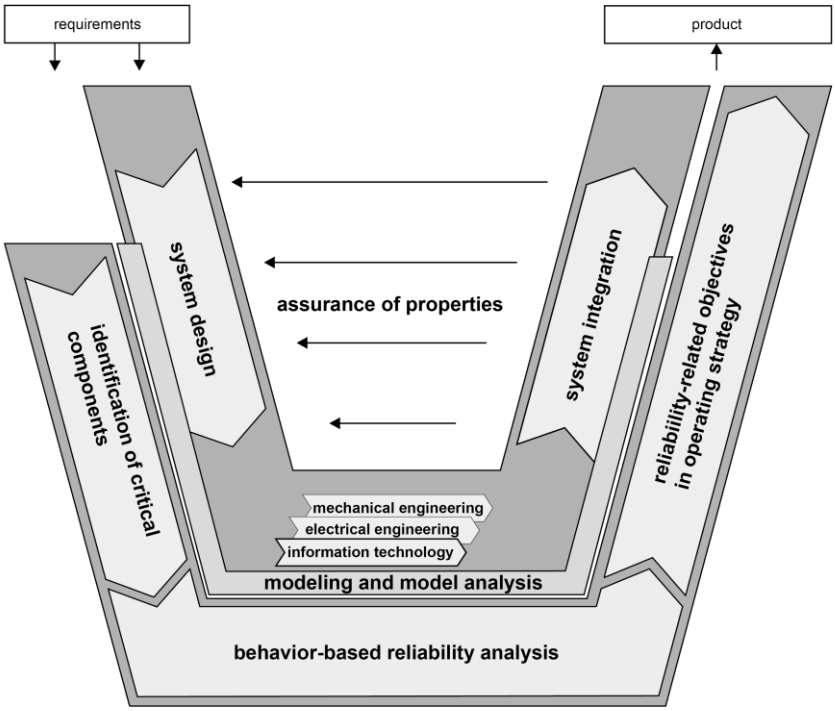

Figure 3. V-Model and actions of Integrated Modeling (based on (VDI, 2004))

The approach delivers support for computation and identification of operating points concerning reliability aspects for integrated systems. This is illustrated at the right wing of the V-model in Figure 3 by incorporating reliability-related objectives into the operating strategy of the system. Despite the focus of integrated modeling on the domain-specific design, its algorithms can be applied in system design in the left wing of the V-model. Although, a behavior-based reliability analysis is not yet possible in this early design stage, it is possible to take advantage of existing system models, i.e. implemented in SysML, to identify critical components within the prevailing architecture.
The methodological approach of integrated modeling of behavior and reliability is shown and described in Figure 4. It should be regarded as a sub-process of the V-model, which occurs several times during the development process. The inputs, the system model and measurements or durability estimations, lead to a tangible value which have to be compared to predefined requirements for reliability. Corrective actions for the improvement of system design deliver the updated system model for the next iteration of the process if necessary. The surrounding boxes give compressed information about the different arrows representing the steps of the integrated modeling process. Because of the automated procedure of the approach a high number of repetitions are possible without much effort. This fact delivers a useful service for in-process testing of the current status during development process, whether the architectural changes lead in the right direction.

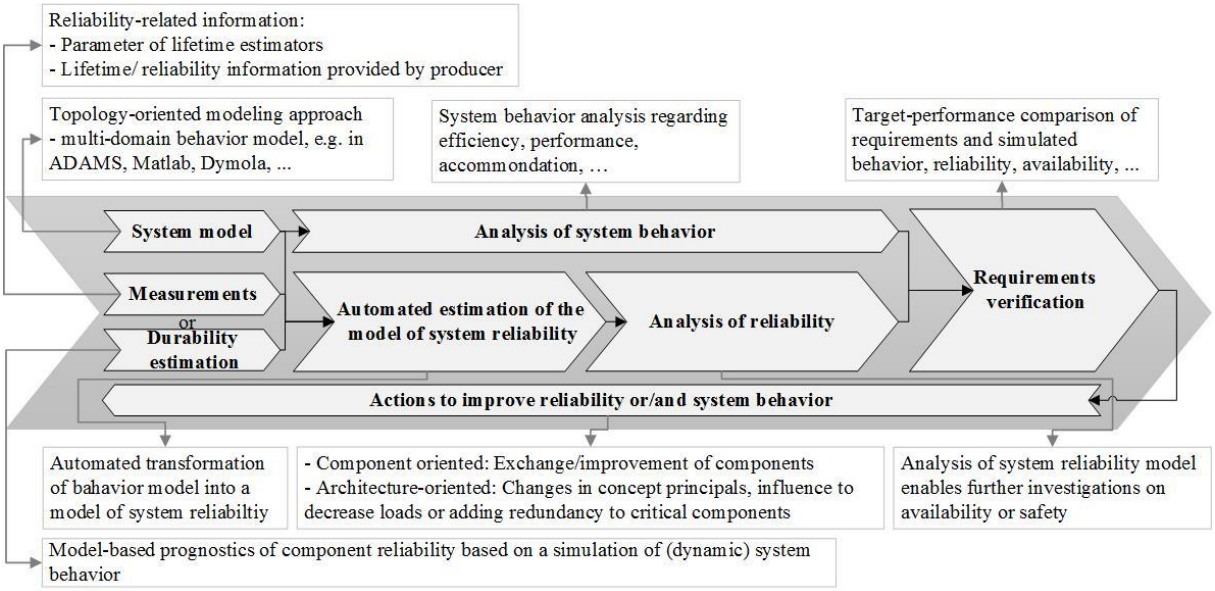

Figure 4. Process model of integrated modeling for system behavior and reliability

5 APPLICATION EXAMPLE

A single plate dry clutch system was chosen as application example (Meyer et al., 2013a; Meyer and Sextro, 2014), which is shown in a basic outline in Fig. 5. The test rig consists of an identically constructed drive and load systems. The drive side system represents the internal combustion engine of a road vehicle, the load side the accelerated mass. Each system is comprised of a brushed DC motor which drives one friction plate. The plates are mounted on a shaft which is connected to the motor via belt drive. Each shaft is held in the belt housing by two ball bearings.

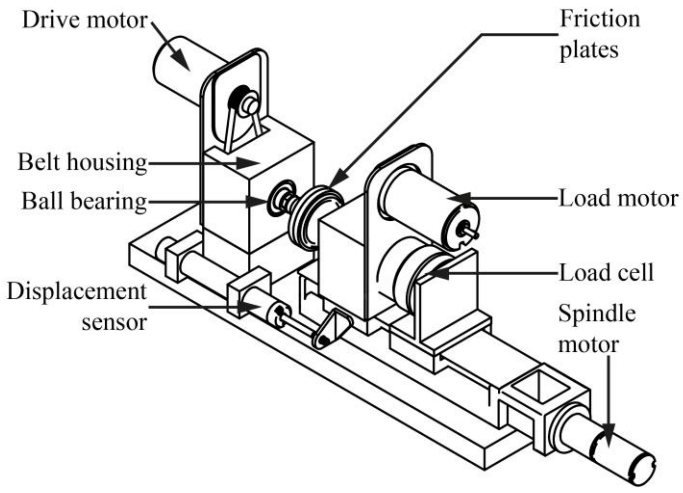

Figure 5. Single plate dry clutch test rig

The typical operation of the clutch system is one actuation cycle: The friction plates are pressed against each other, thus transmitting torque from the drive motor via the friction plates to the load system. Once rotational speed of drive and load side are identical, the clutch is kept engaged for several seconds before it is released and the actuation cycle ends.

This test rig was designed to have one dominating failure mode with contribution of only one component: wearing-out of the friction plates, which results in the inability to transmit torque. A failure of the friction plates is by far more likely to occur than any other component failure, e.g. actuator or sensor defects, broken mechanical parts or failures in control units (Meyer and Sextro, 2014). The clutch actuation strategy directly influences system degradation.

The test rig is exemplarily investigated at two different design phases. At first, a SysML-model is analyzed during system design in order to evaluate structural changes on the system to its reliability. Critical components are identified and solely taken into account at later design phase to reduce system complexity. A Matlab/Simulink-model representing the dynamic system behavior of the system is investigated. The simulation of the model of system behavior is used to analyze the influence of the current operating point and its corresponding parameter set on the overall system reliability using the lifetime estimators of the individual components.

5.1 SysML-Model

The system model in Figure 6, implemented in SysML 1.3 modeling language, represent the elements and its interdependencies of the system. Therefore, a Block Definition Diagram (BDD) is used to model the system and its containing elements. It is applied in early development phases to deliver a model for understanding the whole system in interdisciplinary work and to show different alternative architectures for decision before subsystem and element implementation (Alt, 2012). Figure 6 shows a simplified SysML-model of the clutch test rig. Dashed arrows illustrate information flow between two elements, solid arrows should be interpreted as energy flow. The SysML BDD model is an overview of the static system. It does not include dynamic aspects (Weilkiens, 2014). Every illustrated element contains only its name and a short designation. Further information as values, which are normally shown in SysML-diagrams (Weilkiens, 2014), are left out to simplify the overview.

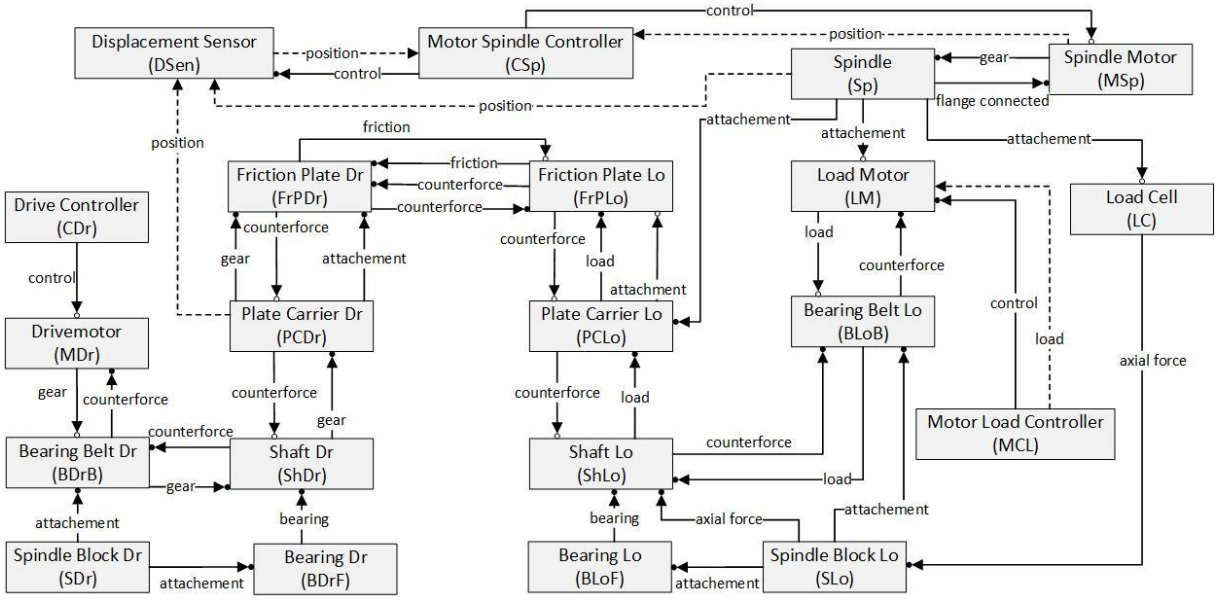

Figure 6. Simplified Block Definition Diagram (BDD) in SysML of clutch test rig

The BDD is analyzed using the integrated modeling methodology in order to investigate reliability for a given system architecture. Since this SysML-model does not provide a model for the dynamic system behavior from which the loads on individual system components can be obtained, the use of lifetime estimators is thus obsolete. Instead, at early design phase it is sufficient to use databases for reliability parameters (Department of Defense, 1995) as a comparative approach. Thus, the obtained component reliabilities are not intended to accurately estimate the real system.

The structure identification algorithm investigates the UML-file in which the main information of the SysML-model is condensed. Since UML is the basis for SysML, the projects can easily be derived from the modeling software in UML-files. In fact, this file comprises also information from the other diagrams of the SysML-model. Depending on the project this could be all kinds of diagrams i.e. sequence, requirements, package and activity diagrams. Thus, the graph of the reliability model in Figure 7 comprises information not only from the BDD in Figure 6 (Alt, 2012; Weilkiens, 2014). The graph of the reliability model shows a generalized entity of identified dependencies between components within the prevailing SysML-model and represents them as arcs between the nodes. If no prior assumptions are

made to reject any component from the reliability analysis, all components within the SysML-model are taken into account for structure identification.
The extracted structure of the system reliability model is shown in Figure 7 at two consecutive time steps t_i and t_{i+1}. This representation in two time slices enables for modeling temporal dependencies among the components that are identified in the model of system behavior. The node 'model' has no representation in the SysML-model, but rather represents the overall system.
The chosen reliability model, Dynamic Bayesian Networks, does not allow for cyclic subgraphs and is thus demanding for algorithms to break those cycles. In Dynamic Bayesian Networks, the investigated system is modeled at different time steps, which allows for modeling of temporal dependencies. If two consecutive time steps are chosen sufficiently close, cyclic subgraphs can be interpreted as a temporal problem which can be modeled using Dynamic Bayesian Networks. Those temporal dependencies appear as arcs connecting components across the two time steps (Kaul et al., 2016). In addition, each component in t_i is connected with its future self in the next time step t_{i+1} in order to model degradation.
To allow for the transformation of the SysML-model (and Simulink-model as well), it is assumed that each system component has a unique function that is required to perform the main function of the system. Hence, the system, represented in Figure 7 as node 'model' and 'model_1', depends on all system components, since there are no redundant components.

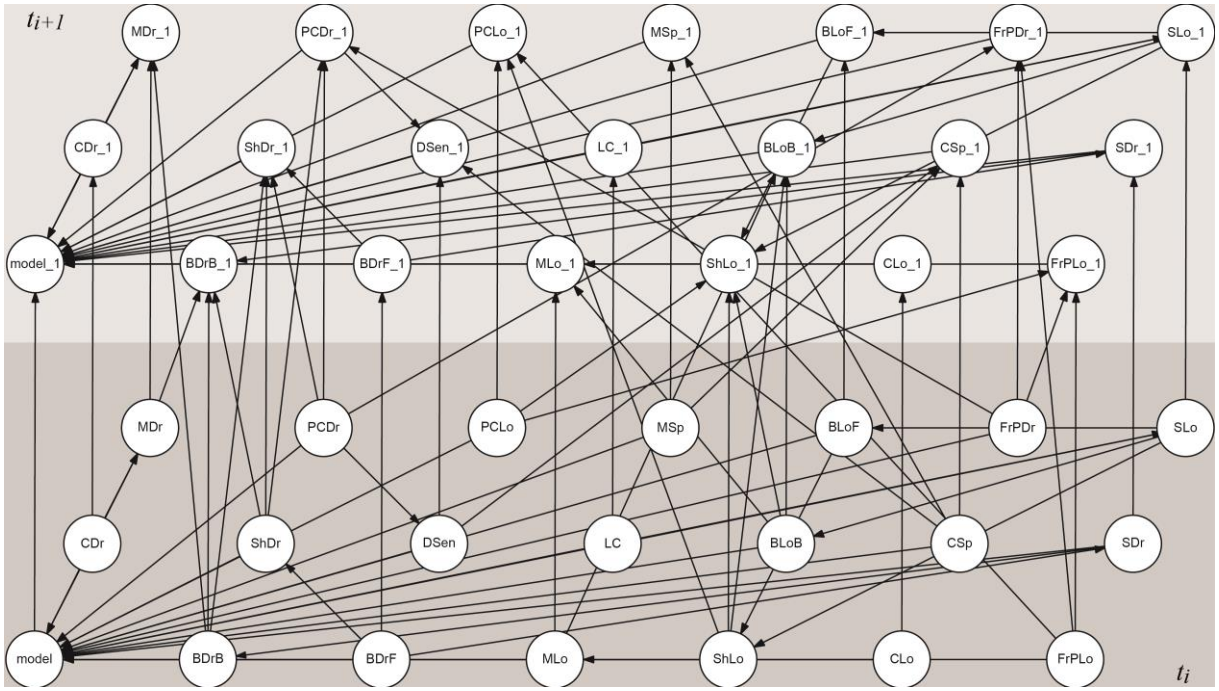

Figure 7. Extracted topology of the reliability model in two time slices t_i and t_{i+1}

The obtained reliability model allows for identification of critical components as well as a comparative evaluation of different system architectures. Taking advantage of diagnostic methods inherent to the chosen reliability model, it is possible to identify the components that are most likely to fail if a system failure occurred. If model complexity increases during design process, the identification of critical elements in early design phases allows for focusing on relevant elements in consecutive design phases based on behavior models. The knowledge about reliability shifts to earlier phases of the development process and thereby gives advanced options in the consideration of the most critical components.

5.2 Matlab/Simulink-Model

In domain-specific design phase as well as in the system integration phase (Figure 3) detailed behavior models, i.e. implemented in multi-domain modeling tools, are used to investigate behavior-related requirements.
The clutch test rig is modeled in Matlab/Simulink in order to implement the controller strategy and for verification of the dynamic behavior.

The identification of critical components based on the SysML-model (Figure 6) is carried out to provide the components, which are most likely to lead to a system failure. These identified critical components are the ball bearings mounted on the shaft on drive and load system (BDrB, BDrF, BLoF, BLoB). Thus, the drive and load motor are also taken into account, because a failure of a bearing is one of the most likely failure modes. The displacement sensor is also identified as critical. Since the clutch is designed to have one dominating failure mode, which is the wear-out of the friction plates, these are also interpreted as critical. The neglected components, such as the micro controller and structural components (shafts, plate carriers and spindle blocks), are found to be not critical in this context.
In contrast to the BDD SysML-model, the Simulink-model can be used to model the dynamic behavior of the clutch. Thus, lifetime estimator are annotated to the previously identified components in order to model the influence of current dynamic behavior on component degradation. A more detailed outline on the implementation of these estimators is shown in (Kaul et al., 2015).
In Figure 8 the reliability functions of the components as well as of the overall system is shown. Bearings BDrB and BLoB are only subjected to a radial force induced by the required belt tension. Bearings BDrF and BLoF are the same type as the previously introduced, but are subjected to an axial force, which is the normal force applied to engage the clutch. The lifetime estimator of the drive and load motor (MDr, MLo) is reduced to model only a failure of the bearings due to simplicity. Since these bearings have a much smaller dimension than BDrB and BLoB, the impact of the belt tension on their reliability is more crucial. The simulation confirms the design objective for the friction plates to be the dominating failure mode of the clutch. Thus, system reliability is close to the reliability of the friction plates, but always smaller. The friction plates appear in Figure 6 as two independent components. The friction plates are assumed to be alike and are represented by one lifetime estimator due to simplicity.
The system reliability function is used to verify reliability requirements on the overall system and additionally be used to formulate reliability-related objectives for the implementation of an operating strategy.

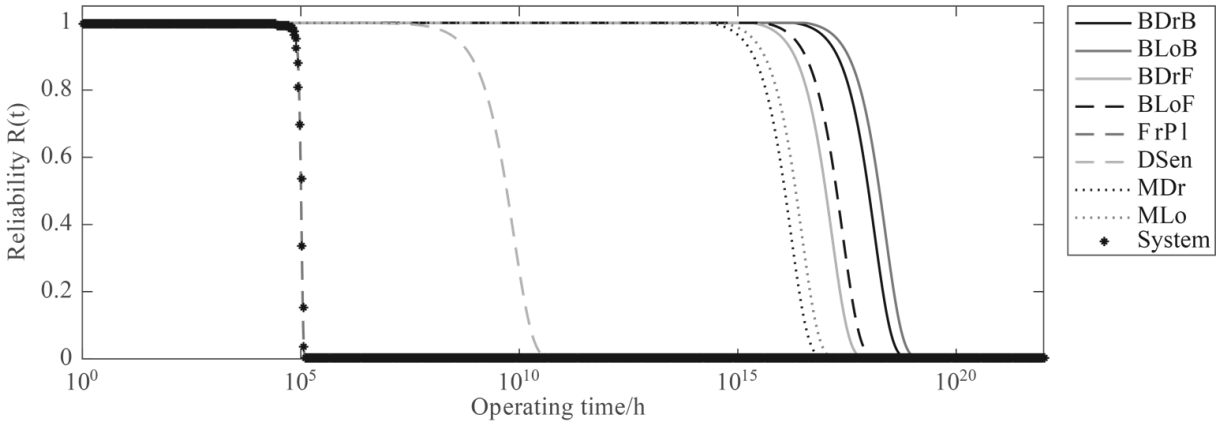

Figure 8. System and component reliability graphs obtained from the analysis of behavior model in Simulink

6 CONCLUSION

The approach for integrated modeling of behavior and reliability is able to support the development process and evaluation of system reliability in early development phases. In addition of the application in domain specific design and in identification of operation points, this paper describes the third possible usage of the approach. The validity for system models as well as models describing systems behavior and the automated use of the approach, lead to an easy integration into standardized development and design processes. The approach thus meets the demand for reliability investigation at the earliest possible phases, which means the time where a first consistent model of the system is available. The existence of system models by approaches like model-based Systems Engineering (MBSE) deliver the necessary bases for the analysis without adding effort. Automated inclusion of standard models e.g. SysML simplifies use of the integrated approach as shown in the application example of the clutch test rig. Not only but especially complex systems and products, whose business model is based on providing operation time and thus relies on durability, can gain high potentials for development by using the

integrated approach. Goals for further development of the approach are validation on a higher number of exemplary products - particularly with higher complexity - and to generate a tool and a detailed description of the procedure for application.

REFERENCES

Alt, O. (2012), *Modellbasierte Systementwicklung mit SysML: In der Praxis*, Hanser, Carl, München.

Bestory, C., Marc, F. and Levi, H. (2007), "Statistical analysis during the reliability simulation", *Microelectronics Reliability*, Vol. 47 No. 9, pp. 1353–1357.

Choley, J.-Y., Rivière, A., Nguyen, N., Kadima, H. and others (2012), "SysML and safety analysis for mechatronic systems".

Cressent, R., David, P., Idasiak, V. and Kratz, F. (2010), "Increasing reliability of embedded systems in a SysML centered MBSE process: Application to LEA project".

David, P., Idasiak, V. and Kratz, F. (2010), "Reliability study of complex physical systems using SysML", *Reliability Engineering & System Safety*, Vol. 95 No. 4, pp. 431–450.

Department of Defense (1995), *Military handbook Reliability Prediction of Electronic Equipment: MIL-HDBK-217F Notice 2*.

Gausemeier, J., Moehringer, S. and others (2003), "NEW GUIDELINE VDI 2206-A FLEXIBLE PROCEDURE MODEL FOR THE DESIGN OF MECHATRONIC SYSTEMS".

Gräßler, I., Hentze, J. and Yang, X. (2016), "Eleven Potentials for Mechatronic V-Model", *6th International Conference Production Engineering and Managment*, 29.09.-30.09., Lemgo, Deutschland.

Helle, P. (2012), "Automatic SysML-based safety analysis".

Kaul, T., Meyer, T. and Sextro, W. (2015), "Integrated Model for Dynamics and Reliability of Intelligent Mechatronic Systems", in Podofillini et al (Ed.), *European Safety and Reliability Conference (ESREL2015)*, Taylor and Francis, London.

Kaul, T., Meyer, T. and Sextro, W. (2016), "Modeling of Complex Redundancy in Technical Systems with Bayesian Networks".

Meyer, T. and Sextro, W. (2014), "Closed-loop Control System for the Reliability of Intelligent Mechatronic Systems", available at: https://www.phmsociety.org/node/1222

Meyer, T., Sondermann-Wölke, C., Kimotho, J.K. and Sextro, W. (2013a), "Controlling the Remaining Useful Lifetime using Self-Optimization", *Chemical Engineering Transactions*, Vol. 33, pp. 625–630.

Meyer, T., Sondermann-Wölke, C., Sextro, W., Riedl, M., Gouberman, A. and Siegle, M. (2013b), "Bewertung der Zuverlässigkeit selbstoptimierender Systeme mit dem LARES-Framework", in Gausemeier, J., Dumitrescu, R., Rammig, F.J., Schäfer, W. and Trächtler, A. (Eds.), *9. Paderborner Workshop Entwurf mechatronischer Systeme*, Heinz Nixdorf Institut, Universität Paderborn, Paderborn, pp. 161–174.

Mhenni, F., Nguyen, N. and Choley, J.-Y. (2014), "Automatic fault tree generation from SysML system models".

Pahl, G., Beitz, W., Feldhusen, J. and Grote, K.H. (2007), *Engineering design: a systematic approach*, Vol. 157, Springer Science & Business Media.

Papadopoulos, Y. and Maruhn, M. (2001), "Model-based synthesis of fault trees from matlab-simulink models".

Sagan, S.D. (2004), "The Problem of Redundancy Problem: Why more Nuclear Security Forces May Produce less Nuclear Security", *Risk Analysis*, Vol. 24 No. 4, pp. 935–946.

Schallert, C. (2011), "Inclusion of reliability and safety analysis methods in modelica".

VDI (2004), *VDI 2206 - Design methodology for mechatronic systems* No. 2206, Verein Deutscher Ingenieure, Düsseldorf.

Walden, D.D. (2015), "INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities".

Walter, M., Gouberman, A., Riedl, M., Schuster, J. and Siegle, M. (2009), "LARES-A Novel Approach for Describing System Reconfigurability in Dependability Models of Fault-Tolerant Systems".

Weilkiens, T. (2014), *Systems Engineering mit SysML/UML: Modellierung, Analyse, Design*, 3., überarb. u. akt. Aufl., rev. Ausg, dpunkt, Heidelberg, Neckar.

Xiang, J., Yanoo, K., Maeno, Y. and Tadano, K. (2011), "Automatic synthesis of static fault trees from system models".