A Holistic Methodology for Model-based Design of Mechatronic Systems in Digitized and Connected System Environments

Xiaobo Liu-Henke, Sven Jacobitz, Sören Scherler, Marian Göllner, Or Aviv Yarom and Jie Zhang
Institute for Mechatronics, Ostfalia University of Applied Sciences, Salzdahlumer Str. 46/48, Wolfenbuettel, Germany

Keywords: Model-based Design, Holistic Design Methodology, Mechatronic Systems, Digitized and Connected Systems, Cyber-physical Systems, MiL, SiL, HiL, DSiL, Driving Simulation.

Abstract: This paper presents a holistic methodology for model-based design of mechatronic systems in digitized and connected system environments. On the one hand, this includes system structuring for the controllability of complex system relations. On the other hand, it comprises the extension of the design and safeguarding process by requirement, data and evaluation management as well as the integration of novel technologies (e.g. Driving-Simulator-in-the-Loop (DSiL) simulations) for the execution of closed-loop simulations under realistic and at the same time reproducible operation conditions. Furthermore, a low-cost rapid control prototyping development platform (LoRra) is presented, with which the presented methodology can be applied. The new holistic methodology is verified by case studies.

1 INTRODUCTION

Society and economy are undergoing disruptive changes due to increasing digitization and networking. One example is the automotive industry, which has to cope with rapidly increasing demands on vehicle development and software due to autonomous driving and the challenge of flexible, application-specific vehicle use (Zhou et al., 2020). But also the manufacturing industry is imperatively dependent on Industry 4.0 solutions to increase productivity and flexibility (Matt et al., 2020). This is to maintain its competitiveness while at the same time producing more varied products. Therefore, new technologies such as artificial intelligence (AI) and the Internet of Things (IoT) are being used, more and more also by small and medium-sized enterprises (SMEs).

Complex algorithms, e.g. from the field of AI or complex feedback controllers, as well as the constantly increasing degree of networking significantly increase the effort required for the design and testing of the resulting cyber-physical systems (CPS) (Alshareef and Sarjoughian, 2018). A holistic approach is necessary to develop and test such intelligent systems in an increasingly fast-moving, complex, digitized and interlinked environment (Maldonado et al., 2019). However, current design methods do not take this into account sufficiently. In this paper, we extend the proven mechatronic design method to a holistic methodology for the model-based development of mechatronic systems in digitized and networked system environments. Main research goals are the integration of several methodology parts into a holistic methodology as well as a seamless tool support. For this purpose, both, the system structuring as well as the design and validation process, are critically questioned and adapted to the new requirements by integrating new concepts.

2 STATE OF THE ART

2.1 Definition of Mechatronic Systems

According to VDI guideline 2206 (VDI, 2004), mechatronic systems (Figure 1) consist of a basic system, sensors, actuators and information processing. It interacts with its environment through the flow of information, energy and materials.

The basic system is a physical system, which usually consists of mechanical, electromechanical, hydraulic and pneumatic components. It constitutes the core of the mechatronic system. With the help of sensors, selected state variables of the basic system are determined by measurement or observation. An information processing system calculates the necessary actions to achieve the desired system behavior, which are implemented by the actuators on the basic system. A mechatronic system is always embedded in a (sys-

Liu-Henke, X., Jacobitz, S., Scherler, S., Göllner, M., Yarom, O. and Zhang, J.
A Holistic Methodology for Model-based Design of Mechatronic Systems in Digitized and Connected System Environments.
DOI: 10.5220/0010566702150223
In *Proceedings of the 16th International Conference on Software Technologies (ICSOFT 2021)*, pages 215-223
ISBN: 978-989-758-523-4
Copyright © 2021 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

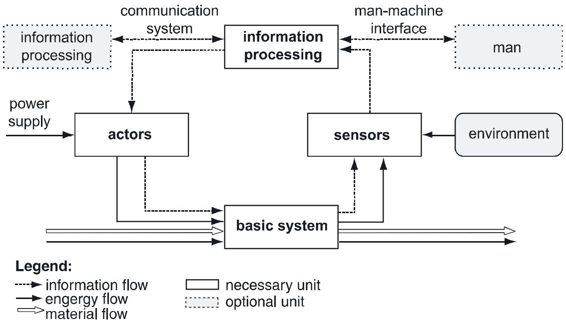

Figure 1: Basic structure of a mechatronic system according to VDI 2206 (VDI, 2004).

tem) environment and can optionally have a human-machine interface (HMI) and/or a communication interface to other systems.

2.2 System Structuring

(Liu-Henke et al., 2002) describes an approach for system structuring, that has been proven in numerous research and development projects. It is based on the generalized cascade principle, according to (Lückel et al., 2001). A hierarchical structuring of the entire system into encapsulated function modules with defined interfaces is carried out in a top-down process. It considers subordinate systems with higher dynamics as part of the controlled system during design. The following hierarchical levels are used for structuring:

- **Mechatronic Function Module (MFM):** The MFMs are the basic elements of the system and represent the lowest level of the hierarchy. They consist of sensors, actuators, information processing and basic system related mechanical structure. This functionally encapsulated modules are the most vital element of the system. It owns defined physical and signal interfaces to the superordinate parts.
- **Mechatronic Function Group (MFG):** The coupling of several MFMs results in a MFG with its own information processing and sensors. They use the subordinate MFMs with their actuators and mechanical structure. MFMs are mainly used for structuring the information processing.
- **Autonomous Mechatronic System (AMS):** Several MFGs, which are coupled by physical and signal interfaces form an AMS in their entirety. An AMS is completely independent of its environment and has its own sensors and information processing. It includes the top level of the mechanical structure.
- **Cross-linked Mechatronic System (CMS):** The CMS is an signal based coupling of several AMS and is the top hierarchical level. It coordinates and optimizes operations by regulating the flow of information and passing on decisions that affect all the AMSs in the network.

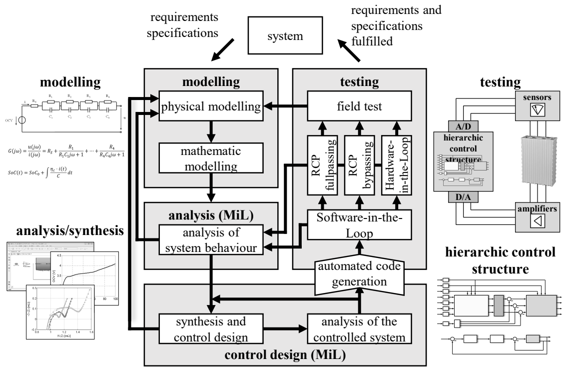

Figure 2: Mechatronic development cycle for design and testing according to (Liu-Henke et al., 2002).

2.3 Design and Testing

After structuring the system, design and testing are carried out seamlessly model-based, using the mechatronic development cycle by (Liu-Henke et al., 2002) (Figure 2) in the bottom-up procedure. This is characterized by iteration possibilities at any time as well as tests in early phases of the design, so that failures can be eliminated at an early stage. This reduces the development time and thus ultimately the development costs.

The development cycle begins, always taking into account the requirements and specifications, with the description of the basic system in a physical and mathematical model. The parameters are determined from technical documents or by measurements on the real (sub-)system in frequency and time domain. Hardware-in-the-Loop (HiL) test benches, for example, can be used for this purpose. Subsequently, the model is validated and verified by comparing the simulation results with measurement data.

This modelling process is followed by an analysis of the system behavior, using model-in-the-loop (MiL) simulations. If the requirements, e.g. for modeling depth and accuracy, are not met, a feedback loop to the modeling takes place. The analysis results are then used for function design and synthesis. Further MiL simulations are used to analyze and optimize the controlled system with respect to the specifications. Once a sufficient functional state has been reached, the function is transformed to program code (e.g. C code) by automated code-generation without error-prone manual programming. For testing, this generated code is examined again in software-in-the-loop

(SiL) simulations before further verification and optimization is carried out under real-time conditions in rapid control prototyping (RCP) full- or bypassing or in HiL simulations. Suitable development tools and test benches are used for these tasks. Finally, field tests are carried out, which end the mechatronic development cycle when all requirements and specifications are met.

2.4 Development Platforms

The presented development and testing process is seamlessly supported by a highly automated computer-aided engineering (CAE) development platform, consisting of software and hardware (Liu-Henke et al., 2014). The resulting minimization of manual work avoids random errors and significantly reduces development time. The high reproducibility of the results also simplifies validation and certification processes.

In the industry, the expensive combination of Matlab / Simulink and a real-time system from dSPACE (Hanselmann, 1996) is widely used (Liu-Henke et al., 2014). The model is built using existing libraries such as the dSPACE Automotive Simulation Model (ASM). For analysis and synthesis, extensive Matlab / Simulink functions such as pole/zero calculation or frequency response analyzers are available. Using the Simulink Coder, the developed function model can be automatically transformed into C code and then be implemented on a target hardware using dSPACE Real-Time Interface. Depending on the requirements, a selection of different real-time platforms such as the Scalexio system is available. Online experiments for measurement and calibration tasks can be performed via the HMI ControlDesk by using widespread protocols like the Universal Measurement and Calibration Protocol (XCP) (Lemon, 2003).

3 CONCEPTION

3.1 Challenges

Customers and users are constantly demanding innovations and more functionality at ever shorter intervals. This challenge can only be met by high end mechatronic products. These growing demands are accompanied by increased system complexity, which on the one hand makes the system design and testing increasingly complicated and on the other hand requires a higher level of safety. Shorter product life-cycles and increasing competition in the course of globalization are exerting ever greater cost pressure, especially on SMEs. Furthermore, the increasing individualization of systems or products to meet customer demands requires that the user and his usage behavior must be considered during development. Current methods of MiL, SiL and HiL simulation are performed ”open-loop”, i.e. without integration of the user into the control loop. This results in user needs being disregarded.

Due to the increasing computing power, more and more AI-based functions are being used. However, their behavior in unknown situations is not predictable due to their complex structure (Montavon et al., 2018). Therefore, especially high demands are placed on the verification of such methods (Aeberhard et al., 2015). Also, in contrast to conventional functions, the design is no longer possible analytically but is done via machine learning. For these processes, a lot of preprocessed data is needed, which covers most operating situations.

Another challenge is the increasing networking of mechatronic systems in CPS and the IoT. This makes new types and sources of information available, which can lead to innovative, cross-system functionalities. The use of novel algorithms and procedures changes the system structure and sometimes results in deviations from the generalized cascade principle (cf. section 2.2), since not only reference values but also further information may be communicated between different hierarchical levels. In addition, tests for validation in the real system environment are often safety-critical. Tests for validation in a virtual simulation environment, on the other hand, usually do not offer any possibility for the system to interact with the user.

It is obvious, that existing methods have to be extended, based on the new requirements. Also, completely new methods need to be developed. Furthermore, there is only one approach known that tries to deal with these challenges. Further research as presented in this paper is necessary.

3.2 Requirements for the Holistic Methodology

The requirements for the holistic methodology are derived from the challenges by using a structured analysis, intensive literature review and experience from various projects. In the following, the main requirements for a holistic methodology, which result from the challenges elaborated above, are derived.

R1 A procedure for system structuring is required to reduce the complexity of the strongly interconnected system by defining encapsulated functional modules with clearly specified interfaces to each other. In addition to existing approaches (cf. section 2), this procedure must offer opportunities to deviate from the generalized cascade principle.

R2 The development process has to be supplemented by a system for requirements management in order to take these fully into account on the one hand and to be able to check their fulfillment on the other. This is particularly necessary from the point of view of interdisciplinary cooperation.

R3 The functional design and testing shall be supplemented by a system for data management of learning and test data. The data management system must enable the data to be reused and thus enable reproducible test scenarios.

R4 To generate the exorbitantly large amount of test data, it must be possible to supplement real data with synthetic data from simulation-based development tools.

R5 Functional safeguarding shall evaluate the behavior of complex functions (like AI) with respect to safety of the intended functionality (SOTIF, cf. (ISO, 2019)) and functional safety (cf. (ISO, 2018)). It shall be possible to reliably detect faulty learning results such as under- and overfitting.

R6 HiL technology must be used and extended to minimize safety risks and to perform reproducible learning and test scenarios for interconnected systems under real operating conditions in real-time.

R7 Methods to ensure the real-time capability of novel functions based on AI and Big Data must be integrated into the design process.

R8 There shall be possibilities to consider the user in the testing process by closed-loop simulations.

R9 The development tools for applying the methodology should also be affordable for SMEs to enable them to maintain and increase their competitiveness.

3.3 Conception of the Holistic Methodology

In order to meet the requirements derived in section 3.2, the following elements are added to the methodology presented in section 2:

- **Extended System Structuring:** Two new hierarchical levels in mechatronic system structuring are introduced to meet the requirements of CPS and deviations from the generalized cascade principle.
- **Advanced Design and Testing Process:** The design and testing process is extended by a systematic requirements management, a continuous data management and an evaluation management. In addition, measures to ensure real-time capability throughout the process are considered.
- **Closed-loop Driving Simulator:** The process gap between open-loop simulation methods (MiL, SiL, HiL) and prototype testing is closed for automotive applications by integrating a closed-loop driving simulator.
- **Development Platform:** A low-cost RCP development platform is introduced, which enables usage of the development methodology also by SMEs.

4 HOLISTIC METHODOLOGY

4.1 Extended System Structuring

Since distributed systems with decentralized intelligence in a cyber-physical environment have a highly complex and heterogeneous structure, the method of mechatronic structuring presented in section 2.2 is extended by two additional function levels to master the system complexity:

- **Autonomous Function Group (AFG):** If several AMS are networked with each other so that they can exchange information, a swarm is formed, which is called AFG. The autonomy of each individual AMS is still given, only the sum of available information has grown. The AFG has additional sensors that provide data for all subordinate AMS. The difference to the original definition of the CMS is that no decisions are made for subordinated systems, but information is exchanged and cooperative operation is possible.
- **Cross-linked Function Group (CFG):** Several CMSs can be grouped across domain boundaries as CFG, so that data can be exchanged in structured clusters. A CFG establishes an exchange of information between the CMS in the sense of a complete networking and digitization.

Furthermore, the interfaces of AMS and CMS from section 2.2 are redefined. The application of this extended approach to system structuring is explained using the restructuring of FREDY as an example in Figure 3. The advantage of this new structure is that it allows differentiated treatment of systems outside the vehicle at the levels above the AMS. In this way,

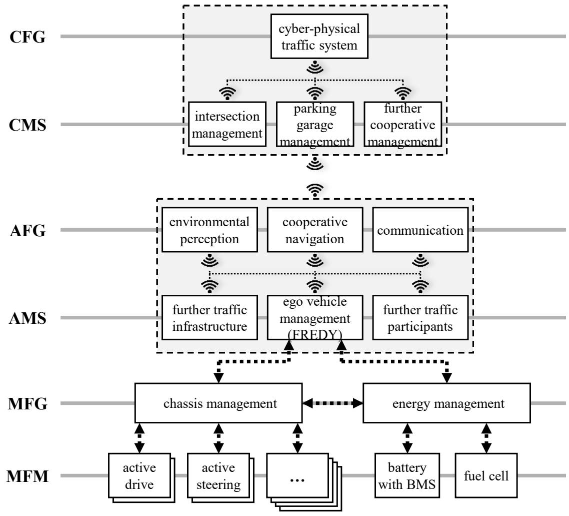

Figure 3: Hierarchical system structure of the mechatronic system FREDY as an example (Liu-Henke et al., 2016).

the cooperative functionalities at AFG level can be designed in a more targeted manner with a view to exploiting synergies by all participants. The spatially or domain-specifically separated functionalities at CMS level can be coupled to form a cyber-physical overall system that enables the transfer of information and data between different CMS by CFG. This will enable new functionalities using prediction and AI.

4.2 Advanced Design and Testing Process

In the following, the extensions to the design and testing process (cf. section 2.3) are described. The resulting new process is shown in Figure 4. Requirements management, data management and evaluation management as well as the test level Driving Simulator-in-the-Loop (DSiL) were added.

In order to meet the new challenges, despite the increasing complexity of products or systems, the mechatronic development cycle is expanded to include the requirements management process. This ensures systematic elicitation, documentation and management of requirements as well as their linking with the system structure or its components. Requirements management offers the possibility of managing dynamic requirements throughout the entire product development process or life-cycle. It also clearly communicates and tracks changes of requirements and the impact on the entire system (Inkermann et al., 2019).

In particular, AI algorithms or even functions based on (V2X) communication often rely on enormous amounts of data for design and validation. Also, they generate a lot of data (Kumar et al., 2017). Therefore, the tasks of data management are, among others, the acquisition, preprocessing (like filtering or annotation), storage and management of all required and generated data, e.g., for training an artificial neural network. The origin of the data is flexible. Depending on the use case, they can originate, for example, from measurements, the requirements, communication or simulations (Siddiqa et al., 2016).

The objective of design is a function that fulfills the initially established requirements. The subsequent verification of whether this goal has been achieved takes place during function validation. However, this usually generates a lot of information and signals with a physical context, which, in the case of complex systems, can hardly be reconstructed, despite a well-founded understanding (Conrad et al., 2005). Therefore, it is the task of evaluation management to transform this data into a form that can be interpreted and evaluated with reference to the requirements (Garousi and Elberzhager, 2017).

The existing design and testing process (section 2.3) contains only open-loop simulations without user influence. For the design of vehicle mechatronic systems, not only the driver assistance systems but also the user's (cognitive) behavior plays an important role. To take this influence into account, a multi functional driving simulator can be used in the domain of vehicle mechatronics. Therefore, the model-based, verification-oriented development and validation process was extended by the test level DSiL simulation in addition to the already established MiL, SiL and HiL

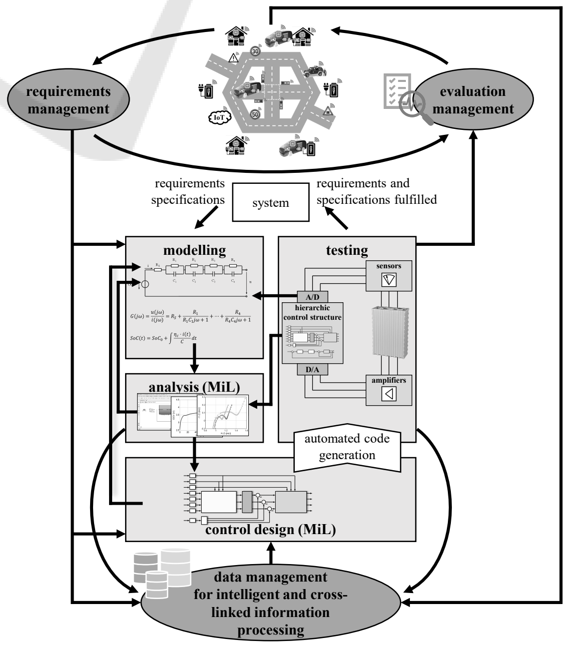

Figure 4: Extended Mechatronic development cycle for design and testing.

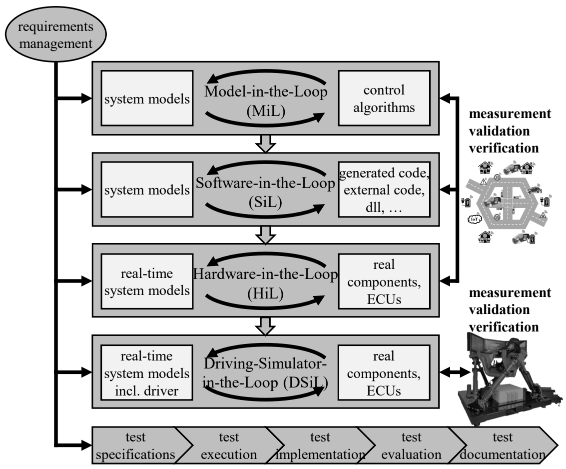

Figure 5: Holistic model-based function development and testing process extended by DSiL simulation.

simulations (cf. Figure 5).

5 SEAMLESS TOOL SUPPORT

5.1 Closed-loop Driving Simulator

Test scenarios can be carried out realistically and reproducible including human behavior using a closed-loop simulator. Key elements of the simulation system are actuators for the stimulation of vestibular, auditory and visual stimuli as well as sensors to measure the test person's behavior. By using a hexapodic motion platform, the test person's sense of balance is stimulated. The simulation is coordinated by a higher-level simulator control computer. Sensors and actuators are initially controlled locally by subordinate control systems in mechatronic function groups. Further communication with the higher-level simulator components takes place via a central router using a UDP Ethernet protocol. Due to the real-time information processing, the simulator can be interfaced with additional HiL test benches or real ECUs (Liu-Henke et al., 2020a).

Using this simulator, the closed loop system including the driver's behavior can be investigated before tests are carried out in real prototypes (Göllner and Tao, 2018). In addition, driver assistance functions - e.g. assistance systems for pedestrian protection - can be tested without having to accept safety risks in real driving tests. It is thus possible to investigate at an early stage of development how a user reacts physiologically and psychologically in interaction with the functions and how the functions are influenced by the human driver. Concrete examples are the brake stuttering of an anti-lock braking system (ABS), which can lead to an anxiety reaction or the vehicle's distance control system (ACC), which may react unpredictably due to the intervention of the driver. But higher-level autonomous driving functions can also be tested.

5.2 Low-cost Development Platform

The development platform presented in section 2.4 supports the original development process. However, in order to implement the enhancements presented in this paper, various modifications and extensions are necessary, especially in the area of requirements, data and evaluation management. In addition, the combination presented is very cost-intensive and is therefore only conditionally suitable for use in SMEs. The low-cost RCP development platform LoRra (cf. (Jacobitz and Liu-Henke, 2020)) developed at Ostfalia addresses exactly these gaps. Based on the open-source CAE environment Scilab/Xcos, a seamless platform is provided to support the presented holistic development and validation methodology, according to Figure 4. By using a low-cost microcontroller as real-time hardware, SMEs will also be able to use the new methodology.

The starting point for system design with the development platform LoRra are the LoRra model libraries. By means of version and configuration management, existing models from previous projects can be optimally used and further developed. The synthesis and analysis of the controlled system is performed by the dynamic system simulator Xcos, which is part of the open source CAE environment Scilab. With the LoRra code generator, Xcos models can be automatically transformed into effective C code. Online simulations are enabled by the LoRra Real-Time Interface (RTI). This offers the possibility to control interfaces of the real-time hardware from the model as well as to implement the generated program automatically on the real-time hardware (Jacobitz and Liu-Henke, 2019). Due to the open interfaces of the LoRra RTI, the real-time hardware can be chosen flexibly, from low-cost microcontrollers to powerful multi-core systems. As a human-machine interface for online experiments, the integrated Graphical Experimental Software (iGES) provides an intuitive graphical user interface. Flexibly configurable instruments can be used to measure functional states and manipulate parameters.

6 VERIFICATION BY CASE STUDIES

In various currently ongoing and future research and development projects, the presented development methodology is applied, further verified and optimized using the requirements from section 3. Figure 6 shows an example usage of data and evaluation management for the model-based design of a function for automated lateral guidance using Artificial Neural Networks (ANN) and Genetic Algorithms (GA). ANN consist of several networked layers of computing units (neurons), similar to the human brain, which accumulate input signals and calculate an associated output. GA are nature-analogous optimization methods, which are suitable for training ANN.

These AI algorithms are very data- and computation-intensive procedures, both in the functional design and in their validation. Therefore, data management was used to store and manage all data needed for the training and test processes, such as requirements, training routes, simulation parameters and results. Evaluation management was used to analyze the designed functions in detail and to evaluate them with regard to the requirements. The lower part of Figure 6 shows the result of an ANN that meets the requirements for lane center keeping without strong oscillations. The results of the evaluation are fed back to the data management. More information about this application can be found in (Yarom et al., 2020). Further verifications were performed by setting up a cyber-physical laboratory test field for smart mobility applications (Liu-Henke et al., 2020b) and a predictive energy management system for fuel cell electric-hybrid vehicles in a connected traffic system (Scherler et al., 2020).

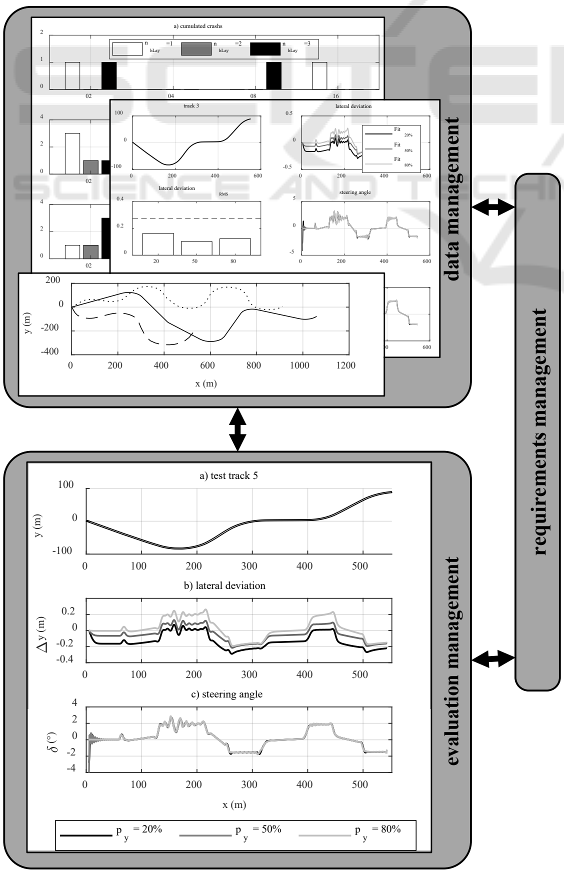

Figure 6: Example of the requirements, data and evaluation management for an AI application.

7 CONCLUSION AND OUTLOOK

In this paper, first an overview of the current development process of mechatronic systems is given. Afterwards, the presentation of current challenges due to the changing technological framework (e.g. digitization and networking in CPS, increasing system intelligence through AI, increasing influence of user behavior on system development) is presented. Requirements are derived which are used to design measures for the optimization of the development methodology. These measures include on the one hand the extension of the system structuring approach for the controllability of the increasingly complex system interconnection. On the other hand, they include the extension of the design and testing process by a requirements, data and evaluation management as well as the integration of new technologies, e.g. for DSiL simulation, to perform closed-loop simulations under realistic and at the same time reproducible operating conditions.

In summary, a holistic methodology for model-based development of connected digitized mechatronic systems with seamless tool support was presented. Furthermore, a low-cost RCP development platform (LoRra) is presented. With this platform, the presented methodology can be applied even by SMEs. The new holistic methodology was basically verified by case studies from different application areas. Further work is dealing with more extensive verification and optimization of the seamless low-cost tool support. The focus here is in particular on the setup of a cyber-physical laboratory test field for the real-time simulation of complex networked mechatronic systems.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

ACKNOWLEDGEMENTS

Funded by the Lower Saxony Ministry of Science and Culture under grant number ZN3495 within the Lower Saxony ”Vorab” of the Volkswagen Foundation and supported by the Center for Digital Innovations (ZDIN).

REFERENCES

Aeberhard, M., Rauch, S., Bahram, M., Tanzmeister, G., Thomas, J., Pilat, Y., Homm, F., Huber, W., and Kaempchen, N. (2015). Experience, Results and Lessons Learned from Automated Driving on Germany's Highways. *IEEE Intelligent Transportation Systems Magazine*, 7(1):42–57.

Alshareef, A. and Sarjoughian, H. S. (2018). Model-driven time-accurate DEVS-based approaches for CPS design. In *Model-driven Approaches for Simulation Engineering*. Society for Modeling and Simulation International.

Conrad, M., Sadeghipour, S., and Wiesbrock, H.-W. (2005). Automatic Evaluation of ECU Software Tests. In *SAE Technical Paper Series*, volume 2005-01-1659, Warrendale, PA, United States. SAE International.

Garousi, V. and Elberzhager, F. (2017). Test Automation: Not Just for Test Execution. *IEEE Software*, 34(2):90–96.

Göllner, M. and Tao, H. (2018). Simulator development. *Automotive Testing Technology International*, September 2018:102–106.

Hanselmann, H. (1996). Automotive control: from concept to experiment to product. In *Proceedings of the Joint Conference on Control Applications Intelligent Control and Computer Aided Control System Design*, pages 129–134.

Inkermann, D., Huth, T., Vietor, T., Grewe, A., Knieke, C., and Rausch, A. (2019). Model-Based Requirement Engineering to Support Development of Complex Systems. *Procedia CIRP*, 84:239–244.

ISO (2018). *Road vehicles - Functional safety*. Number ISO 26262:2018. Beuth Verlag GmbH, Berlin, Germany.

ISO (2019). *Road vehicles: Safety of the intended functionality*. Number ISO/PAS 21448:2019. Beuth Verlag GmbH, Berlin, Germany.

Jacobitz, S. and Liu-Henke, X. (2019). A Real-Time Interface for Xcos – Demonstrated on a Battery-management System. In *Proceedings of the 2nd Scilab Conference*.

Jacobitz, S. and Liu-Henke, X. (2020). The Seamless Low-cost Development Platform LoRra for Model based Systems Engineering. In *Proceedings of the 8th International Conference on Model-Driven Engineering and Software Development*, pages 57–64. SCITEPRESS.

Kumar, A., Boehm, M., and Yang, J. (2017). Data Management in Machine Learning. In *Proceedings of the 2017 ACM International Conference on Management of Data*, pages 1717–1722, New York, NY, USA. ACM.

Lemon, K. (2003). Introduction to the Universal Measurement and Calibration Protocol XCP. In *SAE Technical Paper Series*, Warrendale, PA, United States. SAE International.

Liu-Henke, X., Feind, R., Roch, M., and Quantmeyer, F. (2014). Investigation of low-cost open-source platforms for developing of mechatronic functions with rapid control prototyping. In *10th International Conference on Mechatronic Systems and Materials*, Opole, Polen.

Liu-Henke, X., Gollner, M., Jacobitz, S., Scherler, S., Zhang, J., and Yarom, O. A. (2020a). Driving-Simulator-in-the-Loop — Virtual Function Design with Consideration of Human Behaviour. In *IEEE International Symposium on Systems Engineering*.

Liu-Henke, X., Jacobitz, S., Gollner, M., Zhang, J., Scherler, S., and Yarom, O. A. (2020b). Cyber-physical Industry 4.0 laboratory test field to simulate self-optimizing intralogistics. In *19th International Conference on Mechatronics*.

Liu-Henke, X., Lückel, J., and Jäker, K.-P. (2002). An active suspension/tilt system for a mechatronic railway carriage. *Journal of IFAC - Control Engineering Practice*, 10(9):991–998.

Liu-Henke, X., Scherler, S., Fritsch, M., and Quantmeyer, F. (2016). Holistic development of a full-active electric vehicle by means of a model-based systems engineering. In *IEEE International Symposium on Systems Engineering*.

Lückel, J., Hestermeyer, T., and Liu-Henke, X. (2001). Generalization of the cascade principle in view of a structured form of mechatronic systems. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 123–128.

Maldonado, L., Chang, W., Roy, D., Annaswamy, A., Goswami, D., and Chakraborty, S. (2019). Exploiting System Dynamics for Resource-Efficient Automotive CPS Design. In *Design, Automation & Test in Europe Conference & Exhibition*, pages 234–239.

Matt, D., Modrak, V., and Zsifkovits, H. E. (2020). *Industry 4.0 for SMEs: Challenges, opportunities and requirements*. Springer International Publishing, Cham.

Montavon, G., Samek, W., and Müller, K.-R. (2018). Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1–15.

Scherler, S., Liu-Henke, X., and Henke, M. (2020). Predictive Energy Management for an Electric Vehicle with Fuel Cell Range Extender in Connected Traffic Systems. In *19th International Conference on Mechatronics - Mechatronika*. IEEE.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

Siddiqa, A., Hashem, I. A. T., Yaqoob, I., Marjani, M., Shamshirband, S., Gani, A., and Nasaruddin, F. (2016). A survey of big data management: Taxonomy and state-of-the-art. *Journal of Network and Computer Applications*, 71:151–166.

VDI, editor (2004). *Design methodology formechatronic systems*. Number VDI 2206:2004-06. Beuth Verlag GmbH, Berlin, Germany.

Yarom, O., Scherler, S., Goellner, M., and Liu-Henke, X. (2020). Artificial Neural Networks and Reinforcement Learning for Model-based Design of an Automated Vehicle Guidance System. In *12th International Conference on Agents and Artificial Intelligence*, pages 725–733. SCITEPRESS.

Zhou, H., Xu, W., Chen, J., and Wang, W. (2020). Evolutionary V2X Technologies Toward the Internet of Vehicles: Challenges and Opportunities. *Proceedings of the IEEE*, 108(2):308–323.

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS