Conceptual Design of Advanced Condition Monitoring for a Self-Optimizing System based on its Principle Solution

Christoph Sondermann-Woelke[a*], Tobias Meyer[a], Rafal Dorociak[b], Jürgen Gausemeier[b], Walter Sextro[a]

[a]University of Paderborn, Mechatronics and Dynamics, Paderborn, Germany
[b]University of Paderborn, Heinz Nixdorf Institute, Product Engineering, Paderborn, Germany

Abstract: The rapid development of communication and information technology opens up fascinating perspectives, which go far beyond the state of the art in mechatronics: mechatronic systems with inherent partial intelligence. These so called self-optimizing systems adapt their objectives and behavior autonomously and flexibly to changing operating conditions. On the one hand, securing the dependability of such systems is challenging due to their complexity and non-deterministic behavior. On the other hand, self-optimization can be used to increase the dependability of the system during its operation. However, it has to be ensured, that the self-optimization works dependable itself. To cope with these challenges, the multi-level dependability concept was developed. It enables predictive condition monitoring, influences the objectives of the system and determines suitable means to improve the system's dependability during its operation.
In this contribution we introduce a procedure for the conceptual design of an advanced condition monitoring based on the system's principle solution. The principle solution describes the principal operation mode of the system and its desired behavior. It is modeled using the specification technique for the domain-spanning description of the principle solution of a self-optimizing system and consists of a coherent system of eight partial models (e.g. requirements, active structure, system of objectives, behavior, etc.). The partial models are analyzed separately in order to derive the components of the multi-level dependability concept. In particular, the reliability analysis of the partial model active structure is performed to identify the system elements to be monitored and parameters to be measured. The principle solution is extended accordingly: e.g. with system elements required for the realization of the dependability concept. The advantages of the method are shown on the self-optimizing guidance module of a railroad vehicle.

Keywords: Mechatronic Systems, Principle Solution, Condition Monitoring, Conceptual Design

1. INTRODUCTION

The conceivable development of information and communication technology will enable mechatronic systems with inherent partial intelligence. We refer to this by using the term "self-optimization". Self-optimizing systems react autonomously and flexibly on changing environmental conditions. They are able to learn and optimize their behavior during operation [1]. Self-optimization is a process that consists of the three steps: 1) analysis of the current situation, 2) determination of system's objectives and 3) adaptation of system behavior. A condition monitoring is essential for the support of the self-optimization process. It is especially important for its first step – the analysis of the current situation.
To avoid design changes in late development phases, which are generally costly and time-intensive, it is necessary to develop the condition monitoring concept during an early design phase. This early design of the condition monitoring concept goes beyond the current standard ISO 17359 [2]: in this standard, the conception of the condition monitoring starts when the system has already been taken into operation.
In this contribution a procedure for the conceptual design of an advanced condition monitoring for self-optimizing systems, called multi-level dependability concept, is presented. The starting point is the specification of the principle solution; it is modeled using the specification technique for the domain-spanning description of advanced mechatronic systems, which has been developed in the Collaborative Research Centre (CRC) 614 "Self-optimizing concepts and structures in mechanical engineering". The description of the principle solution contains all required information for the design of the multi-level dependability concept during the early engineering phase of conceptual design. This contribution shows how this information is used to define the components of the multi-level dependability concept.

Preprint from: Sondermann-Woelke, C.; Meyer, T.; Dorociak, R.; Gausemeier, Jü. & Sextro, W.: Conceptual Design of Advanced Condition Monitoring for a Self-Optimizing System based on its Principle Solution. Proceedings of the 11th International Probabilistic Safety Assessment and Management Conference (PSAM11) and The Annual European Safety and Reliability Conference (ESREL2012), 2012.

This paper begins with a section giving a brief overview of the design of self-optimizing systems. In the following two sections, the multi-level dependability concept is presented and the method for the conceptual design of the multi-level dependability concept is introduced. After this, the application of the method to the case example of the self-optimizing active guidance module of the innovative railroad vehicle RailCab follows. Before concluding the paper, a short survey of related work is given.

2. DESIGN OF SELF-OPTIMIZING SYSTEMS

The key aspects and mode of operation of a self-optimizing system are illustrated in Figure 1 a) [1][3]. The self-optimizing system determines its currently active objectives on the basis of the encountered influences on the technical system. New objectives can be added, existing objectives can be rejected or the priority of objectives can be modified during operation. Therefore the system of objectives and its autonomous changing is the core of self-optimization. Adapting the objectives in this way leads to a continuous adjustment of the system`s behavior to the environmental state. This is achieved by adapting parameters or reconfiguring the structure (e.g. switching between different controller types). For the description of the architecture of the information processing of self-optimizing systems the hierarchical architecture concept of the Operator-Controller-Module is used (Figure 1 b)).

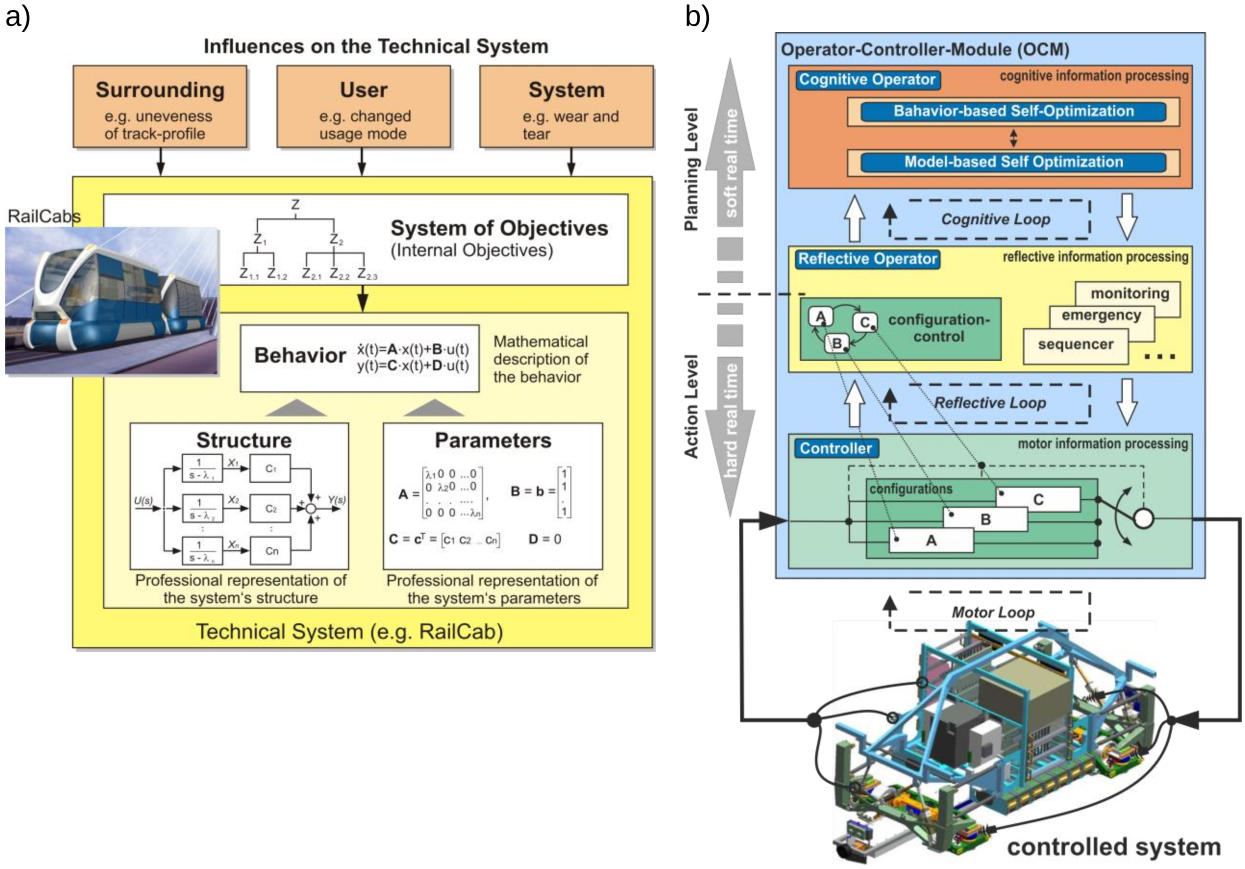

Figure 1. Aspects of self-optimizing systems (a) and the Operator-Controller-Module (b)

The basic structure of the Operator-Controller-Module (OCM) [4] is composed of the three levels controller, reflective operator and cognitive operator. The controller level stands for the control loop with direct access to the technical system. As a matter of course the software at this level operates continuously under hard real-time conditions. The controller itself can be made up of a number of controller configurations with the possibility of switching between them.

The reflective operator supervises and regulates the controller. It does not directly access the actuators of the system but it modifies the controller by initiating parameter changes or changes of the structure. If changes of the structure do appear (e.g. as in reconfigurations), not just the controllers will be replaced but also corresponding control and signal flows will be switched within the controller itself. Combinations that consist of controllers, circuit elements and corresponding control or signal flows are described as controller configurations. The controlling of the configurations, realized by a state machine, defines which state of the

system uses which kind of configuration. It also determines under which circumstances it is necessary to switch between the configurations. The reflective operator offers an interface (working as a conjunctional element to the cognitive level of the OCM) between the elements operating not in real-time or soft real-time and the controller.
The cognitive operator is the highest level of the OCM-architecture. Here the system uses knowledge about itself as well as its environment in order to improve its own behavior by using varied methods (such as learning methods and model-based optimization). The main emphasis is on the cognitive abilities for carrying out of the self-optimizing process. Model-based processes allow a predictable optimization that is, to a large extent, decoupled from the underlying levels while the system is in operation.
The design process of self-optimizing systems is subdivided into two major phases: conceptual design and concretization. The result of the conceptual design phase is the description of the principle solution, which serves as basis for the communication and cooperation of domain experts and the further concretization of the system. For the description of the principle solution the specification technique CONSENS (CONceptual design Specification technique for the ENgineering of complex Systems) is used [3]. The description of the principle solution with CONSENS consists of the aspects Environment, Application Scenarios, Requirements, Functions, Active Structure, Shape, Behavior and System of Objectives (Figure 2). The Behavior consists of a whole group of different kinds of behavior, e.g. the logic behavior, the dynamic behavior of multi-body systems, the cooperative behavior of system components, etc. The aspects are specified and computer-internally represented as partial models. Furthermore, the cross-references between the partial models are modeled. For example, it is modeled, which system elements (partial model Active Structure) concretize which functions (partial model Functions).
Altogether it is necessary to work alternately on the aspects of the system and the corresponding partial models. A corresponding procedure model has also been developed; see [3] for details. To support an engineer during the modeling process, the software tool Mechatronic Modeller has been developed [5].

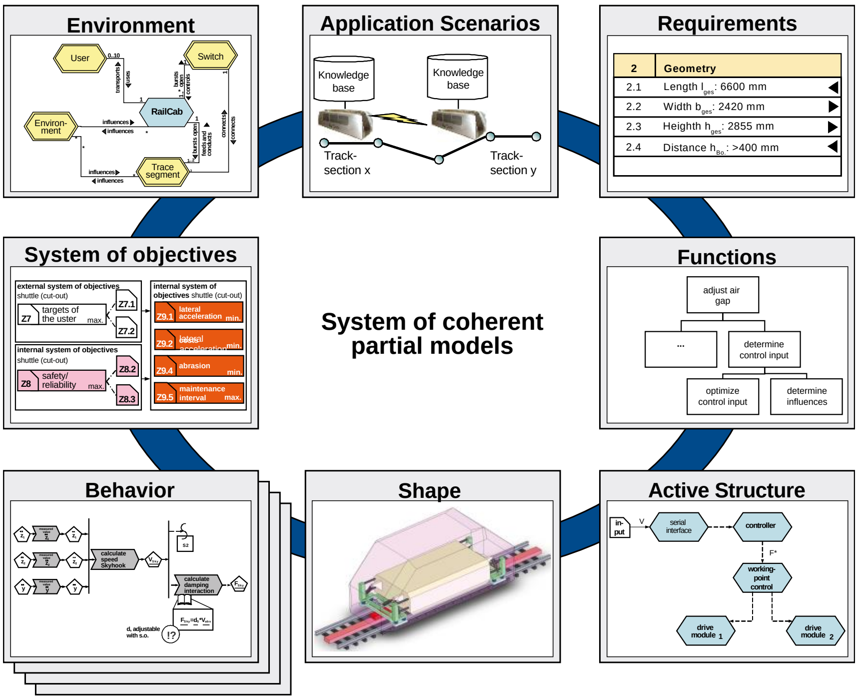

Figure 2. Partial models for the description of the principle solution of self-optimizing systems

3. ADVANCED CONDITION MONITORING OF SELF-OPTIMIZING SYSTEMS

For the condition monitoring of self-optimizing systems the so called multi-level dependability concept was developed [6]. It contains four hierarchically ordered levels for the characterization of different system states and is part of the reflective operator (Figure 3). Furthermore, the system's objectives are affected differently in each level. Thus, a relation between the current system state, which in turn is influenced by the current dependability of the system, to the system's objectives is constructed. The levels of the dependability concept and the impact, which each level has on the System of Objectives, are as follows:

Level I: The system operates in a dependable way. Dependability is one objective among others.

Level II: An error occurred. Self-optimization is used to ensure dependable operation. To this end, the priority of the objective of dependability affected by the error is increased.

Level III: A severe error occurred, but the system can still be controlled. First emergency mechanisms are triggered to reach a safer state. In the system of objectives, safety is the primary objective to avoid the failure of the whole system and the consequences involved. The other attributes of dependability (e.g., reliability, availability) may occur as sub-objectives of safety.

Level IV: Control over the system is lost. Mechanisms like emergency routines are executed to reach a pre-defined fail-safe state.

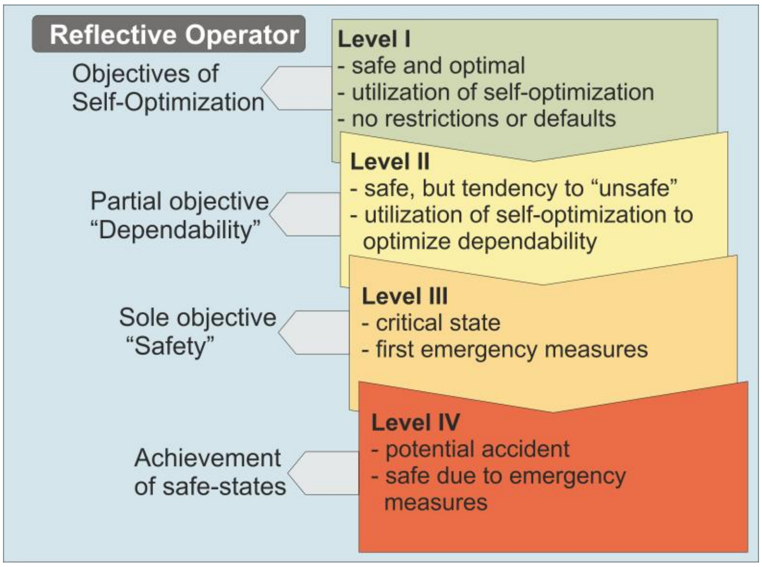

Figure 3. Structure of the multi-level dependability concept

This concept is implemented in the reflective operator. This way, it is both able to get sensor information from the controller layer of the Operator-Controller-Module and to communicate directly with the cognitive operator to influence the optimization of the system. Situated in the reflective operator, the concept is also able to initiate switching operations between different control strategies via the configuration control. The proposed design method for the multi-level dependability concept is explained in detail in the following section.

4. DESIGN OF THE MULTI-LEVEL DEPENDABILITY CONCEPT BASED ON THE PRINCIPLE SOLUTION

This section explains our proposed procedure to design the multi-level dependability concept based on the principle solution. The proposed procedure to design the multi-level dependability concept is a modification and an enhancement of the procedure described in ISO 17359 [2][6]. The basic approach is outlined in Figure 4. As can be seen, there are five major steps which are now explained in detail. These steps are not to be seen as a sequence; the procedure is characterized by a number of iterations, which are not depicted.

4.1. Step 1: System Analysis

The system analysis is conducted in three steps, which correspond to the self-optimization process: Analysis of the current situation, determination of system's objectives, adaptation of system behavior. The required information and their relations are already modeled in the principle solution, so it comes naturally to use the partial models Active Structure, System of Objectives and Behavior for this analysis.

For the first step of the system analysis, the partial model Active Structure is used. It describes the system elements chosen to fulfill the required functions of the system. To get an overview of the abilities to monitor the system state, a list of all sensors is generated by the program Mechatronic Modeller.

In the second step of the system analysis, the objectives that are relevant with regard to the dependability of the system and suitable sensors for them are identified. The objectives regarding dependability like "Maximize reliability", "Minimize down-time", "Reduce wear", which are influenced later on by the multi-level dependability concept, are extracted from the partial model System of Objectives. In most cases, these objectives are not as obvious as stated above, thus the relevance of each objective concerning reliability needs to be evaluated. Afterwards, the list of sensors from the first step is related to the dependability-oriented objectives to determine which sensors are to be used for the multi-level dependability concept.

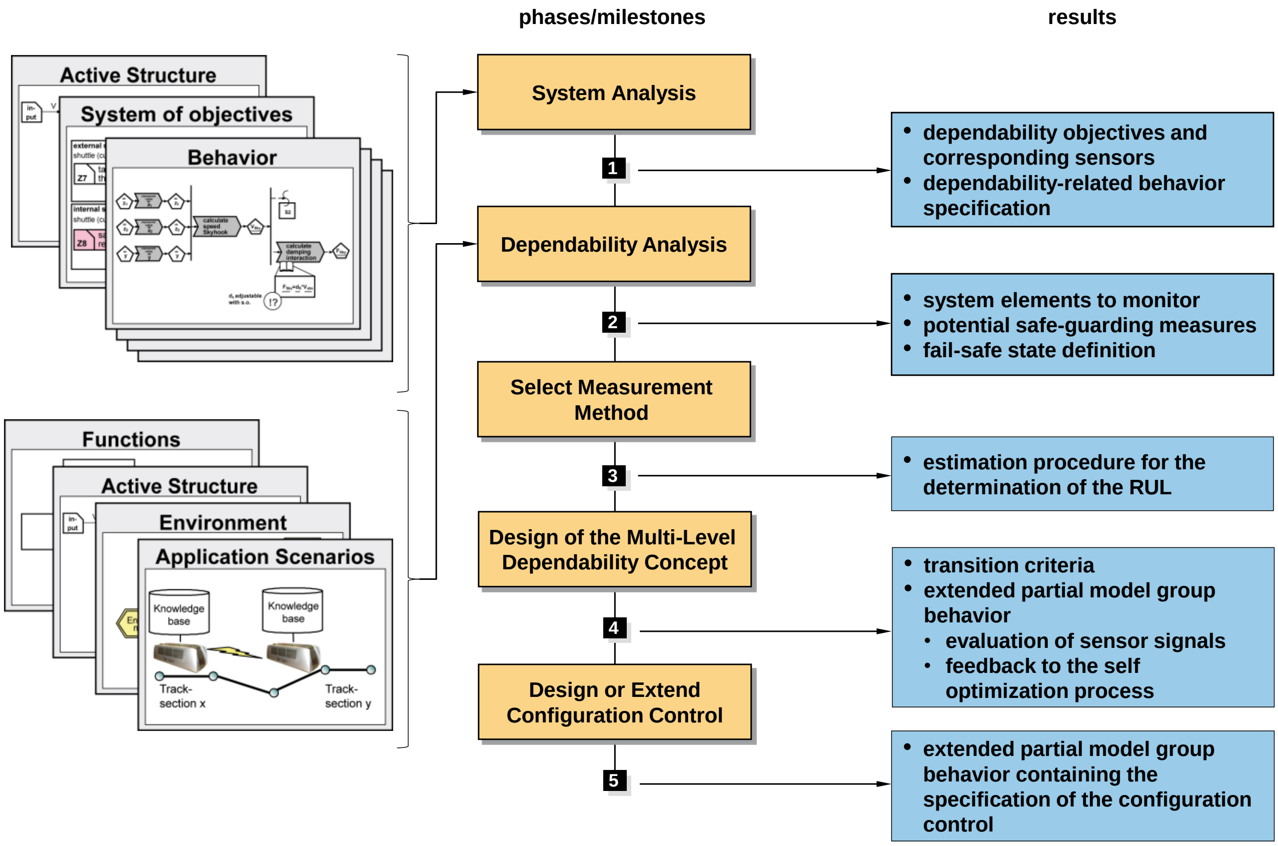

Figure 4. The procedure for the design of the multi-level dependability concept; information contained in the partial models of the principle solution serves as input

The third step is to identify possibilities to influence the system behavior. The relevant partial model group Behavior illustrates the different system states and reconfiguration options. These states and reconfiguration options will be part of the configuration control which is designed to switch to the desired control strategy. Those behavior specifications that support dependability-oriented actions are marked for future reference.

4.2. Step 2: Dependability Analysis

The second step – the dependability analysis – is conducted primarily on the partial models Functions and Active Structure. Mainly the established reliability engineering method of Failure Mode and Effects Analysis (FMEA [7]) is used. In order to conduct the FMEA, the system elements of the Active Structure are exported to an FMEA tool [8]. This procedure is also supported by the program Mechatronic Modeller, which is capable of exporting the Active Structure model. Based on this data, the failure modes and countermeasures are determined together with the engineers developing the system. Not only system inherent effects but also the surrounding environment could be a reason for failures. The partial models Application Scenarios and Environment support the process of identifying these influences.
For the design of the multi-level dependability concept, several results of the FMEA are important. Firstly, the failure modes point out which system elements are subject to wear and fatigue failures. These system elements are primary candidates for a condition monitoring in combination with a life time observer. The risk priority number obtained from the FMEA is crucial for the decision which system elements to monitor. If a critical system element's failure mode is not related to an objective of the system, a corresponding objective has to be added. Furthermore, countermeasures also indicate which system elements are of special interest because a failure would lead to a long down-time or even severe damage. The countermeasure list

also shows failures which lead to the establishment of a fail-safe state, which has to be defined for any self-optimizing system. Finally, failures of those system elements which have a negative influence on the dependability-oriented objectives are identified. For these, safeguarding against failures, e.g. redundancy, might be required [9]. If safeguards are used, the principle solution has to be updated accordingly.

4.3. Step 3: Select Measurement Method

In the system and dependability analysis, sensors for monitoring the dependability-related objectives as well as critical system elements were identified. Based on this information, the dependability-oriented objectives are related to quantifiable general measures like the remaining useful life (RUL) or the current failure probability of the system. These general quantities simplify the comparison between different system elements and subordinated system elements. The estimation of the remaining useful life can be based on model-based approaches in which the actual stress of the system elements is compared with their tolerable load. In combination with damage accumulation hypotheses, e.g. Palmgren-Miner [10], the RUL is estimated. If desired, the failure probability of the system elements is calculated using the corresponding distribution functions.

4.4. Step 4: Design of the Multi-Level Dependability Concept

The multi-level dependability concept influences the system behavior. For this, an evaluation of the current system state and a feedback loop is needed. The system is influenced in two ways: Either by adapting the prioritization of the objectives of the system or by switching between different control strategies.
To determine whether the objectives need to be adapted, thresholds between the four levels based on the remaining useful life or other general criteria, as selected in step 3, need to be defined. For this, the safety requirements of the module have to be taken into account. As explained in section 3, the dependability-oriented objectives, which support different attributes of the dependability like reliability, availability, and safety (cf. [11]), are adapted if the second level is reached. If even the third level is reached, the objectives for safety will have absolute priority. In both cases, the system is later on influenced by an increase of the priority of the dependability-related objective, which will lead to more dependable operation. In order to increase the priority, a suitable fixed value for the priority or a strategy to increase it has to be implemented in the cognitive operator. Both evaluation of the sensor signals and feedback to the optimization process are integrated into the partial model group Behavior.
If a failure requires a switching action, this is conducted by the configuration control. How to implement this is explained in section 4.5. The fourth level corresponds to the fail-safe state determined in the dependability analysis. If it is reached, emergency routines are engaged.

4.5. Step 5: Design or Extend Configuration Control

Certain failures (identified by the FMEA) could lead to a switching action of the control strategy, e.g. if a required sensor fails and redundancy requires to switch to another sensor signal. This reconfiguration is conducted by the configuration control, which is embedded into the reflective operator. For the reconfiguration, different control strategies are designed. If switching actions are necessary, they have to be included in the configuration control. Since switching actions have to be initiated very quickly, the required failure detection methods are implemented in the configuration control as well. If there is already a preliminary configuration control included in the partial model group Behavior, it is enhanced by the dependability aspects. Otherwise a new configuration control is set up.

With all necessary system elements included in the principle solution, the design of the multi-level dependability concept and all its interfaces is concluded. In the following section, the proposed procedure is applied to an example system.

5. APPLICATION TO THE ACTIVE GUIDANCE MODULE

One of the main demonstrators of the CRC at the University of Paderborn is an innovative railroad system. Independent vehicles, called RailCabs, transport passengers or goods non-stop from departure to destination. These RailCabs form convoys to take advantage of slipstream and thus reduce the energy consumption of following RailCabs. Since at high velocities (intended maximum velocity is about 180 km/h), normal switches would not be able to change the direction sufficiently quick to dissolve convoys, passive switches are used. When going over these, each RailCab individually steers in its desired direction. The system module for this steering action is called active guidance module. It is shown in Figure 5. Besides steering in passive switches, the guidance module actively controls the wheel guidance in normal tracks. This leads to reduced wear on wheels and rails because both flange contacts on straight tracks as well as in curves are avoided and wheel slip is minimized. While doing so, disturbances like track irregularities and side wind are compensated. There are also other influences on the active guidance module that have their origin in the RailCab itself. These are, for example, the necessity to restrict energy consumption or a variation of the velocity.

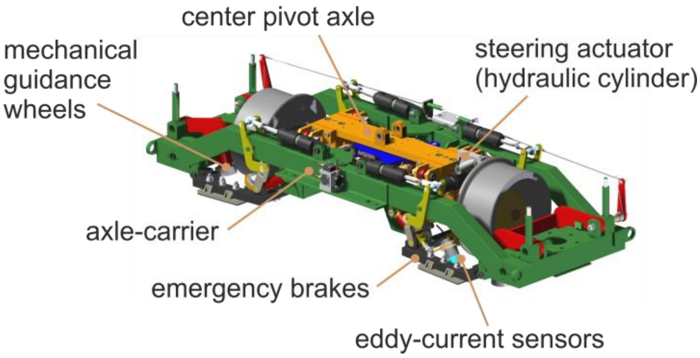

Figure 5. Active guidance module of the RailCab

The active guidance module is a key element of a RailCab and as such needs to function dependably. To ensure this, the multi-level dependability concept has been implemented. The realization of the five main steps to design the multi-level dependability concept, as described in section 4, is explained in the following.

5.1. Step 1: System Analysis

At first, an overview of the abilities to monitor the system state is needed. To analyze the current situation, the active guidance module is equipped with several sensors. One incremental sensor at each wheel determines the longitudinal position of the RailCab. This information is required for the feedforward control, which takes a map of the track and the track clearance into account, as well as for the selection of the desired direction at a passive switch. Since a drift of the incremental sensor’s signal cannot be avoided, the longitudinal position is corrected regularly by a proximity switch which passes over a reference plate. Furthermore, eddy-current sensors on each side of each wheel are used to measure the current lateral position which is the deviation from the center line within the track, and the current clearance which could be used for optimization of the trajectory within the track limits. Two acceleration sensors and a yaw rate sensor are provided for further information about the RailCab movements. A displacement sensor is integrated into the hydraulic steering actuator.

The active guidance module uses multi-objective optimization for the steering control strategy. The optimization objectives are given in the partial model System of Objectives; an extract is depicted in Figure 6. The main goal is to steer within the track clearance while neither having flange contacts nor wasting energy on unnecessary steering actuator movements. At the same time, the lateral accelerations have to be kept low to ensure comfort for passengers and a certain safety margin has to be ensured [12]. The lowest risk for flange contacts is to move along the center line of the track. For this, even small track deviations would have to be corrected by the steering actuator, which would not be energy efficient. Opposed to this is ‘cutting corners’, where the vehicle stays on a very smooth path which requires little movements of the steering actuator. This forces to use the clearance up to its full capacity and increases the probability of flange contacts if the controller cannot compensate disturbances as desired. The optimization enables the system to work in between these two extremes. A trajectory within the track is calculated which is optimal with regard to the current objectives. A feedforward controller is used to steer according to this trajectory, while a feedback controller keeps deviations due to disturbances like side wind low.

The main dependability issue is the minimization of flange contacts to increase the reliability and thus the availability of the RailCab. Another objective is to minimize the wear of the hydraulic actuator. This is similar to the objectives ”Maximize comfort” and ”Minimize energy consumption”, since all three lead to minimal actuator movements.

The configuration control comprises several control strategies. The most advanced strategy uses the optimization described above and both a feedforward controller as well as a feedback controller. If no optimization is available, the trajectory generated for the feedforward controller is oriented towards the center line within the track. If the determination of the lateral position fails, the feedforward control can still be used. If all systems fail, the steering will become stuck, which leads to fast wearing of the flanges. In this case, the mechanical guidance wheels are activated and the vehicle is slowed down.

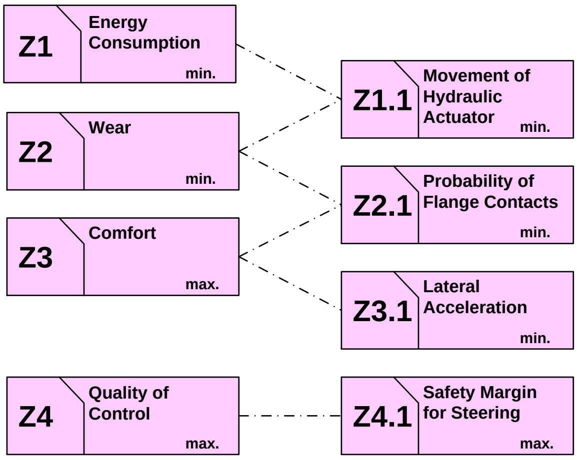

Figure 6. System of objectives of the active guidance module (cut-out)

5.2. Step 2: Dependability Analysis

The matrix for the FMEA is directly generated from the information in the principle solution. The failure modes are determined and suitable countermeasures are derived together with the developers of the system. The FMEA reveals which system elements are subject to wear and fatigue failures. For the active guidance module, the system elements to be monitored are the wheels since the rolling contact leads to wear. Besides the continuous wear due to the unavoidable motion between wheel and rail, flange contacts increase wear considerably. This is already represented in the objective “Minimize probability of flange contacts”, so no additional corresponding objective is necessary. Other parts like the oversized axle-carrier are of lower importance for dependability considerations and therefore are neglected.

The second important fact revealed by the dependability analysis is the fail-safe state of the system. By taking the Application Scenarios into account, it becomes obvious that severe accidents could happen while going over passive switches. If information about the current position on the track is lost, the vehicle will not be able to steer according to upcoming passive switches. In addition, incorrect information provided by the eddy-current sensors could also lead to undesired system behavior. Therefore two different countermeasures are integrated into the partial model Behavior. The first countermeasure is the fail-safe state. In this state the steering axle is fixed, if possible in center position, the velocity of the RailCab is reduced and mechanical guidance wheels for going over passive switches are engaged. These mechanically drag both axles of the RailCab to one pre-defined direction. The second countermeasure is to safeguard the eddy-current sensors. At minimum two of these, one on each side, are required to determine the lateral displacement and the clearance within the track. If one of these failed, these values could not be measured sufficiently precise which would result in a failure. In order to avoid this, on each side a pair of redundant sensors is used, which makes a total of four. The principle solution and the list of sensors of the system analysis are extended accordingly.

5.3. Step 3: Select Measurement Method

To determine the wear of the wheels, flange contacts are counted via the eddy-current sensors and the distance travelled is monitored over the incremental sensors. The maximum running length of the flanges in contact is compared to this value to obtain the remaining useful life. The hydraulic actuator is subject to wear as well. The wear of the actuator is assumed to be proportional to its total travelled displacement, which is calculated using the displacement sensor integrated into the actuator.

5.4. Step 4: Design of the Multi-Level Dependability Concept

The multi-level dependability concept is defined as follows. In the first level of the multi-level dependability concept, the self-optimization process is able to choose from all objectives without any constraints. The second level is reached if the monitored parameter, in this case the remaining useful life of the wheel due to wear, falls below a certain threshold which has been defined by an expert. In our example, the second level is reached if only 50% of the predefined life time is remaining. If one of the eddy-current sensors fails, redundancy is lost. This failure is also classified as level 2 since an error occurred, but self-optimization can be used to ensure dependable operation. In order to increase the reliability, the objective “Minimize probability of flange contacts” gets higher priority. The third level is reached if only 25% of the predefined lifetime of the wheels is remaining or the loss of the lateral position is detected. The dominant objective is now “Minimization of the probability of a flange contacts”, which leads to a feedforward trajectory following the center line of the track. The fail-safe state ”axle fixed and mechanical guidance activated” is activated if the loss of the longitudinal position is ascertained.
The partial model group Behavior is extended by the evaluation of the sensor signals, as described in section 5.3, and all required switching actions or adaptations of the objectives.

5.5. Step 5: Design or Extend Configuration Control

If failures require switching actions, the corresponding reactive measures are implemented in the configuration control. For the active guidance module, switching is required if one of the redundant eddy-current sensors fails. If this is the case, the failed sensor’s signal has to be neglected; switching to a different control strategy is needed. Both the detection of the failure as well as the switching process are embedded within the configuration control.
The final configuration control is based on a preliminary configuration control, which had been set up for general steering purposes already. The additional actions extend the partial model group Behavior.
With all required components included in the principle solution, the design of the multi-level dependability concept is concluded.

6. RELATED WORK

The idea to monitor the system degradation state has its origin in the field of maintenance strategies. Condition-based maintenance is used to determine the remaining useful life of a system element during operation. Contributions dealing with condition monitoring methods can be found, for example, in Davies [13]. How to set up a condition monitoring is described in the standard ISO 17539 [2]. In [6], this procedure has been modified for self-optimizing systems. The modification was required since a concept for the condition monitoring should be developed during the conception of the system design and not, as the standard states, when the system is already running. This modification is also consistent with the further development of maintenance strategies. Lee and Wang [14] point out that the future of maintenance lies in self-maintaining systems. There are already contributions on this topic like Shimomura et al. [15]. The main idea is to use functional redundancy to keep the system working. The focus is mainly on already existing system failures and therefore a reactive approach whereas self-optimization offers the possibility of implementing proactive strategies.

7. CONCLUSION

This contribution proposes a procedure for the early design of the multi-level dependability concept, which combines condition monitoring with the possibility to influence the behavior of a self-optimizing mechatronic system. The multi-level dependability concept evaluates the current system degradation state and initiates countermeasures if the system is in a risky state. It does so by either adapting the system’s objectives towards a more dependable behavior or by switching between different control strategies.
The starting point of the procedure is the specification of the principle solution, which is the result of the conceptual design phase. It consists of a coherent system of partial models, which contains all information required to conduct the proposed design procedure. In order to derive the multi-level dependability concept from the principle solution, five major steps are performed. In the first three steps, both static measures to increase the dependability, like possibly necessary safeguarding measures and fail-safe states, and possibilities to dynamically adapt the system’s behavior during its operation are identified. For this

adaptation, these are the input to the multi-level dependability concept, which can be life time observers that estimate the remaining useful life, and the objectives of the self-optimizing system that need to be prioritized if dependable operation is at risk. During the steps four and five, the multi-level dependability concept itself is designed. Transition conditions between the different levels of the multi-level dependability concept and appropriate countermeasures are defined.

During all five steps, the principle solution does not only provide the necessary information but it is also enhanced by the results of the distinct steps. The end result is the multi-level dependability concept fully embedded into the principle solution, which then serves as basis for further development steps. The proposed procedure was exemplified by the active guidance module of the innovative railroad vehicle RailCab.

Further steps will be the generation of test cases for the following development phases to validate the multi-level dependability concept and the integration of the multi-level dependability concept in a maintenance framework to optimize the maintenance strategy within the operating phase.

Acknowledgements

This contribution was developed in the course of the Collaborative Research Center 614 “Self-Optimizing Concepts and Structures in Mechanical Engineering” funded by the German Research Foundation (DFG).

References

[1] Gausemeier, J.; Kahl, S., Pook, S.: From mechatronics to self-optimizing systems. Self-optimizing Mechatronic Systems: Design the Future, 7th International Heinz Nixdorf Symposium, Feb. 20-21, 2008, Paderborn, Germany, 3–32, 2008

[2] International Organization for Standardization: ISO 17359: Condition monitoring and diagnostics of machines – General guidelines, 2011

[3] Gausemeier, J.; Frank, U.; Donoth, J.; Kahl, S.: Specification technique for the description of self-optimizing mechatronic systems. Research in Engineering Design, 20 (4), 201–223, 2009

[4] Hestermeyer, T.; Oberschelp, O.; Giese, H.: Structured information processing for self-optimizing mechatronic systems. Proc. of 1st International Conference on Informatics in Control, Automation and Robotics (ICINCO 2004), 230–237, 2004

[5] Gausemeier, J.; Dorociak, R.; Nyßen, A.: The Mechatronic Modeller: A Software Tool for Computer-Aided Modeling of the Principle Solution of an Advanced Mechatronic System. 11th International Workshop on Research and Education in Mechatronics, 2010

[6] Sondermann-Woelke, C.; Sextro, W.: Integration of Condition Monitoring in Self-optimizing function Modules Applied to the Active Railway Guidance Module. Int. Journal on Advances in Intelligent Systems, 3 (1&2), 65-74, 2010

[7] International Electrotechnical Commission: IEC 60812: Analysis techniques for system reliability – Procedure for failure mode and effects analysis (FMEA), Januar 2006

[8] Dorociak, R.; Gausemeier, J.: Absicherung der Zuverlässigkeit komplexer mechatronischer Systeme auf Basis der domänenübergreifenden Prinziplösung. 25. Fachtagung: Technische Zuverlässigkeit (TTZ), Leonberg bei Stuttgart, 2011

[9] Isermann, R.: Fault-Diagnosis Systems - An Introduction from Fault Detection to Fault Tolerance, Springer Berlin, 2005

[10] Miner, M. A.: Cumulative Damage in Fatigue, ASME Journal of Applied Mechanics, Vol 12, A159-A164, 1945

[11] Avizienis, A.; Laprie, J.; Randell, B.; Landwehr, C.: Basic concepts and taxonomy of dependable and secure computing, IEEE Transactions on Dependable and Secure Computing 1 (1), 2004

[12] Geisler, J.; Witting, K.; Traechtler, A.; Dellnitz, M.: Multiobjective optimization of control trajectories for the guidance of a rail-bound vehicle. Proceedings of the 17th World congress of The International Federation of Automatic Control, 4380–4386. 2008

[13] Davies, A.: Handbook of Condition Monitoring – Techniques and Methodology. Chapman and Hall, 1998

[14] Lee, J; Wang, H.: New Technologies for Maintenance. In: Complex System Maintenance Handbook, Springer Series in Reliability Engineering, Part B, 49-78, 2008

[15] Shimomura, Y., Tanigawa, S.; Umeda, Y.; Tomiyama,T.: Development of self-maintenance photocopiers. AI Magazine, 16(4), 41–53, 1995