

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Hybrid Planning for Self-Optimization in Railbound Mechatronic Systems

Philipp Adelt, Natalia Esau, Christian Hölscher,
Bernd Kleinjohann, Lisa Kleinjohann, Martin Krüger, Detmar Zimmer
*University of Paderborn
Germany*

1. Introduction

Self-optimizing mechatronic systems with inherent partial intelligence are the research objective of the Collaborative Research Centre - Self-optimizing concepts and structures in mechanical engineering (CRC 614, 2010). A mechatronic system is called a self-optimizing system in this context, if it is not only able to adapt the system behavior to reach a set of given objectives or goals but also can adapt the objectives themselves (or their weighting) on the basis of an analysis of the actual situation.

Hence, the self-optimization approach promises to leave degrees of freedom in choosing objectives for the system open until runtime. This means that the system can decide upon internal objectives based on external user input and current environmental conditions while the system is running. This is in contrast to a system where all internal objectives are set before the system is started. Having such fixed parameters leads to complicated and overly pessimistic approximations of the parameters that are needed to be set over the course of action that the system will take.

Using self-optimization, leaving that decision open is the key idea of our approach. The external objectives like quality of control, comfort or total energy consumption along with constraints (e.g. maximal peak powers or average power consumption) are still embedded or entered into the system. But settings like distribution of energy usage among subsystems can be determined during runtime based on the actual system conditions.

One application of this approach to mechatronic system design is a novel transportation system that is developed in close collaboration with CRC 614. The core of the system consists of railbound vehicles called RailCabs that enable groups of up to 12 passengers to travel directly without intermediate stops. A test track in a scale of 1:2.5 and two RailCabs in the same scale have been built at the University of Paderborn under the name Neue Bahntechnik Paderborn (NBP; see Figure 1). The RailCab is equipped with a doubly-fed linear motor as well as several innovative subsystems that feature inherent intelligence. Among these subsystems are an Air Gap Adjustment System (AGAS) and an active suspension system described more closely below (see Section 3). Operating parameters of these (sub)systems need to be adapted depending on environmental conditions and properties of the track sections that are being travelled.



Fig. 1. Autonomous shuttles traveling on the NBP test track at the University of Paderborn

This paper will present a hybrid planning approach for realizing self-optimization concepts in complex mechatronic systems using the RailCab as application example. The following scenario will explain the tasks fulfilled by the hybrid planner and the focus of this paper. As the presented mechatronic system is used in a railbound environment, a RailCab has the task to travel from one point on a given track network along several track sections to a destination point. Finding a path along this track network is a discrete planning problem of the hybrid planner that we are not going into more detail here. Our starting point for this paper is a consecutive set of track sections to travel. Along with this comes a power budget that assigns a maximum peak and mean power to each section that must not be exceeded. The task of the hybrid planner is to choose a parametrization or configuration of the control systems that does not violate these constraints and follows a global goal like maximizing passenger comfort. Candidate controller configurations for different mechatronic subsystems like e.g. the active suspension system are determined before system operation using hierarchical optimization (see Section 5). The planner then constructs an initial discrete plan by selecting controller parametrizations for different mechatronic subsystems for each track section to be travelled. The discrete planning part is described elsewhere (Adelt & Klöpper (2009)). Since the controller configurations have been calculated offline, their optimality cannot be assured in any cases due to environmental changes or changes to subsystems happening. Hence, it is necessary to change the configuration and controller respectively, see (Geisler & Trächtler (2009)) for an exemplary model-based approach. However, these runtime changes, that have to be continuously monitored, often result in deviations from the initial plan that will lead to a violation of the overall system objectives and constraints (e.g. minimum comfort or available energy budget in our case). In such situations a replanning is necessary that may also adapt the objectives of the system or its parts. For this purpose the hybrid planner will use simulations of the system and environment model to predict the continuous effects of different controller parametrizations (see Section 6.2). In contrast to traditional approaches the hybrid planning approach presented in this paper, also takes into account the actual system and environment status and is able to adapt system objectives or goals accordingly during runtime.

The paper is structured as follows. After introducing the mechatronic subsystems, air gap adjustment system and active suspension system, used as application example in Section 3, the architecture of the hybrid planner and its use within the application example is explained in Section 4. Subsequently in Sections 5 and 6 the system models needed for optimization, monitoring, control or prediction of system behavior are described. Section 5 introduces the optimization approach by example of the active suspension system while Section 6 concentrates on the measurement, control and prediction of continuous effects using the

AGAS as example. An evaluation of our approach and some concluding remarks follow in Sections 7 and 8.

2. Related work

The term *hybrid planning* has been used in different contexts that describe a combination of general purpose planning with domain specific reasoning (Kambhampati et al. (1993)), combining hierarchical domain properties with other planning concepts (Kambhampati et al. (1998)) or matching planning techniques to business processes (Nutt (1982)). We use the term hybrid planning for the integration of discrete and continuous domains in planning. Techniques used in classical planning like hierarchical task networks and the refinement of plans are usable as part of a hybrid planning system too (Ghallab & Traverso (2004); Russell & Norvig (2003)). Planning of continuous actions, e.g. the planning of trajectories, is not in the focus of this approach to hybrid planning as we deliberately use discrete actions and choose to work on the granularity of the actions. Continuous planning as a constant interleaving of planning and execution is similar to our approach of simulating planned actions with current system state as the basis. Dearden (2005) uses a probabilistic hybrid automaton for diagnosis after initial planning. Thus a discrete plan is checked and evaluated on basis of a hybrid model and afterwards changed or even discarded. This relates to the simulation part in our hybrid planning approach.

The active suspension system of the RailCab is used for hybrid planning in this work. Such active suspension systems for automobiles and also for trains have been extensively studied in recent years. A survey of different strategies could be found in (Li & Goodall (1999)), for example. Active suspension systems are often used to enhance the contradicting objectives comfort and safety simultaneously, which is not possible by means of passive systems. Actual research mostly deals with novel control strategies and methods. Methods related to artificial intelligence as Fuzzy-based controller (Chiou & Huang (2007), Kou & Fang (2007)) or neural networks (Jin et al. (2007)) are used as well as model-based methods as H^2 and H^∞ controller, that can be found in (Yousefi et al. (2006)) for example. Approaches, that are also dealing with energy aspects as we do in our application example, can be found in (Zhang et al. (2007)) and (Stribrsky et al. (2007)).

A description of an air gap adjustment system developed at the CRC 614 is described in (Adelt et al. (2008); Schmidt, Adelt, Esau, Kleinjohann, Kleinjohann & Rose (2008)) and (Esau et al. (2008)). The adjustment of the air gap will be achieved by the usage of additional actuators, working against return springs. An alternative concept of an air gap adjustment system and the related measurement of the air gap are presented in (Gabel (2009)). This concept realizes an air gap control by the usage of the system-inherent normal force.

3. Mechatronic system

At the University of Paderborn a new, innovative traffic and transport concept has been developed (Henke et al. (2008), RailCab (2010)). The basic idea of this concept is to use the existing railway infrastructure with small autonomously driven vehicles, called RailCabs, which are adjusted to the needs of the passengers. These RailCabs accelerate by means of a doubly fed asynchronous linear motor laying in the middle of the existing railway and are equipped with an active guidance as well as an active suspension system.

Implementation of this kind of propulsion requires some kind of information processing at the track and fast communication hardware between the RailCab and the track hardware.

Hence, the track is separated into small sections each one equipped with its own information processing, the so called *Track Sectioning Controls (TSC)*. This TSC is necessary to realize the propulsion but it is also beneficial for self-optimization of the RailCab. It offers the possibility to store track section specific data that can be accessed by an arriving RailCab, see Figure 2. In this paper we concentrate on two kinds of information that are directly related to our hybrid planning approach. On the one hand the TSC provides an actual excitation profile that has been learned previously from former RailCabs. So we are in the situation, that a RailCab knows the future disturbances due to the excitation. On the other hand the measured air gap as well as the calculated air gap, that are both required for air gap modeling and prediction, are delivered by the TSC, too. More details concerning usage and relation to hybrid planning are described in the following sections. All information is updated by the measurements of the leaving RailCab.

In order to continuously enhance the RailCab system, a number of test benches for different components are used to evaluate new methods separately. Therefore we illustrate the hybrid planning approach on the basis of two test benches: the air gap adjustment system and the active suspension system. Their physical functionality is described in detail in the following two subsections at first. Afterwards we present the theoretical methods that we used for the hybrid planning approach.

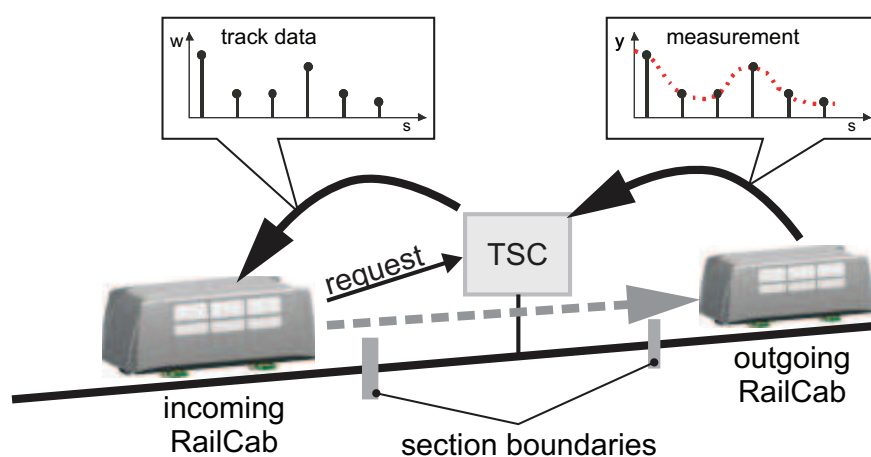


Fig. 2. Data transmission between Track Sectioning Control (TSC) and passing RailCabs

3.1 Active suspension system

The active suspension system performs the task of compensating bumps and other excitations of the railway in order to increase comfort for the passengers. The controller doing this task consists of two parts: a so-called Sky-Hook controller (Li & Goodall (1999)) tries to minimize the absolute coach body acceleration and a relative controller introduces additional virtual spring-damper forces to the coach body. More Details of the controller realization can be found in (Vöcking & Trächtler (2008)). To set up the controller there exists a Hardware-in-the-Loop (HiL) test rig, which emulates the active suspension system of a RailCab, see Figure 3(a).

The structure of the test bench can be partitioned into nine mechatronic modules arranged at three different levels as shown in Figure 3(b). On the top level there is the coach body, which can move in vertical, horizontal and rotational (body roll) degrees of freedom. Beneath the coach body there are two symmetrically allocated actuator groups, each one consisting of a nonlinear guide kinematic, which is connected to a glasfiber reinforced polymers (GRP) spring and three separately controlled hydraulic cylinders that form the third level of the

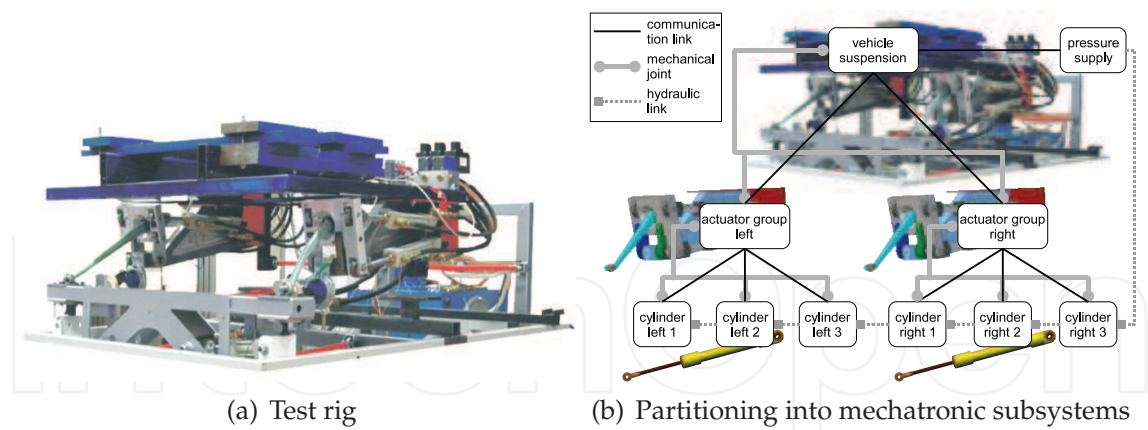


Fig. 3. HiL Test rig for the active suspension system

structure. The main function of the actuator groups is to exert forces on the coach body by a deflection of the GRP-springs. A chassis framework that can again be displaced by three hydraulic cylinders is used to simulate the excitation profile. The pressure in these cylinders is controlled separately by a hydraulic pump and therefore we can disregard the dynamics of the pressure supply and work with constant pressure values for the excitation. The acceleration of the coach body can be measured by two sensors placed at both sides of the coach body. Furthermore, the relative displacement between the coach body and the chassis framework can be quantified in all three dimensions by two contactless sensors.

3.2 Air Gap Adjustment System (AGAS)

The propulsion system of the RailCab is realized by a doubly-fed asynchronous linear drive. In addition to the force transmission this drive enables a contactless energy transfer into the RailCab. A detailed description of direct drives is located in (Zimmer & Schmidt (2005)) and (Zimmer et al. (2005)). The energy transfer of a doubly-fed linear drive is presented in (Schneider et al. (2009)). The immovable part of the drive in the following named stator is installed in the track bed. The rotor is the movable part of the drive and is connected to the RailCab. The distance between rotor and stator is the so-called air gap δ and the optimization variable of the AGAS. Different influences such as incorrectly laid tracks and setting processes lead to an intense fluctuating air gap (Figure 4).

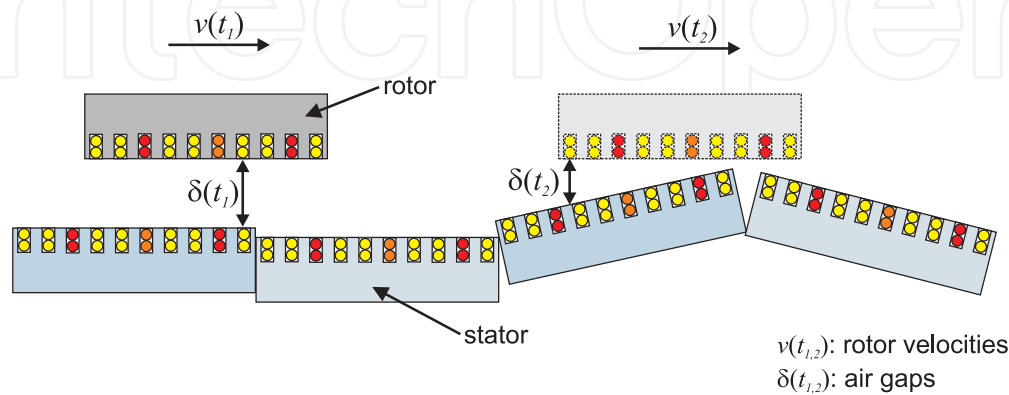


Fig. 4. Air gap over time between rotor and stator

To avoid a collision between rotor and stator a large nominal air gap of 10mm was chosen. This large air gap generates high electrical losses and reduces the efficiency η of the linear drive. The relationship between efficiency and air gap is reciprocal.

$$\eta \sim \frac{1}{\delta} \quad (1)$$

This implies an improvement of the efficiency by minimization of the air gap. The fluctuating air gap at the track motivates a dynamic air gap adjustment. In order to validate a self-optimizing air gap adjustment a HiL¹ test rig (Figure 5) was developed. In contrast to the RailCab the rotor of the linear drive is mounted at the frame of the test rig and propels the stator elements, which are mounted on the rotating track. The stator elements arranged in a circle allow a continuous track simulation.

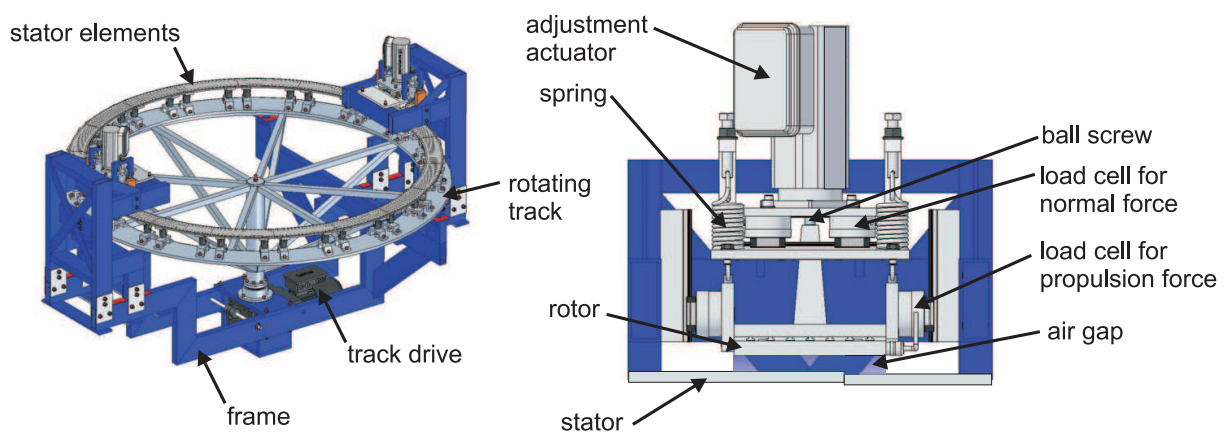


Fig. 5. HiL test rig of the AGAS

The components of the test rig include the adjustment actuators with ball screws. They position the rotor vertically to realize the required air gap. The adjustment actuators work against return springs to minimize the air gap. In case of an actuator failure the linear drive will be raised by the return springs to a safe distance from the stator elements. Hence, a collision between stator and rotor can be avoided. This concept provides the fail safe principle. Load cells measure the normal and propulsion force. They provide the detection of characteristic diagrams for the normal and propulsion force depending on air gap and rotor current. They are used to analyse the system behaviour. The above-mentioned linear drive propels the rotating track, simulating the stator elements laid in the track bed. The stator elements can be adjusted manually in their vertical position to simulate an air gap gradient. The track drive allows to drive or brake the rotating track.

4. Hybrid planning

In this section we will present the hybrid planning architecture. The needed components and information flows are described. We then present the application scenario in the domain of the RailCab in detail and give a step by step description of how the hybrid planning approach is applied.

¹ HiL: Hardware in the Loop

4.1 Overview of the hybrid planning architecture

Planning problems are an integral part of the design and operation of mechatronic systems when it comes to real world problems. Generating a plan that achieves a given objective under a number of constraints and following some global objectives (e.g. save energy, transport in minimal time) appears in most mechatronic systems. Operation needs to be planned along a timeline incorporating several dimensions of continuous physical effects. Under resource constraints like limited energy supply, depending on environmental as well as plan details (e.g. path to travel), consideration of more than the local control problem is needed. A planner needs to take into account these details to generate a plan that is actually executable. This is where hybrid planning is used.

It does not only generate a plan that is feasible under discrete constraints like reaching the target position by traveling along a set of adjacent track sections but it also approximates the continuous behavior of the system while the plan is being built and checks for constraint violations along the way. Doing this with system state updated online with actual data allows for the updating of a plan to reflect the best known approximation of future action’s effects. The term hybrid planning tackles an approach to integrate the continuous effects of physical systems without giving up the possibility to build a plan. Discrete planners generally expect the outcome of an action to be deterministic. An action considered to be executed after another can expect a certain state. We extend this scheme by actively approximating the system state that the execution of an action will result in. It bases on discrete planning in that a discrete problem is expected as a basis. Along with the discrete problem, different discrete parameterizations of the planned system can be considered as alternative ways to execute an action.

The way to get the system state including the continuous effects of an action is to run a simulation of the system or the relevant system part. Figure 6 shows the components used to implement such an approach.

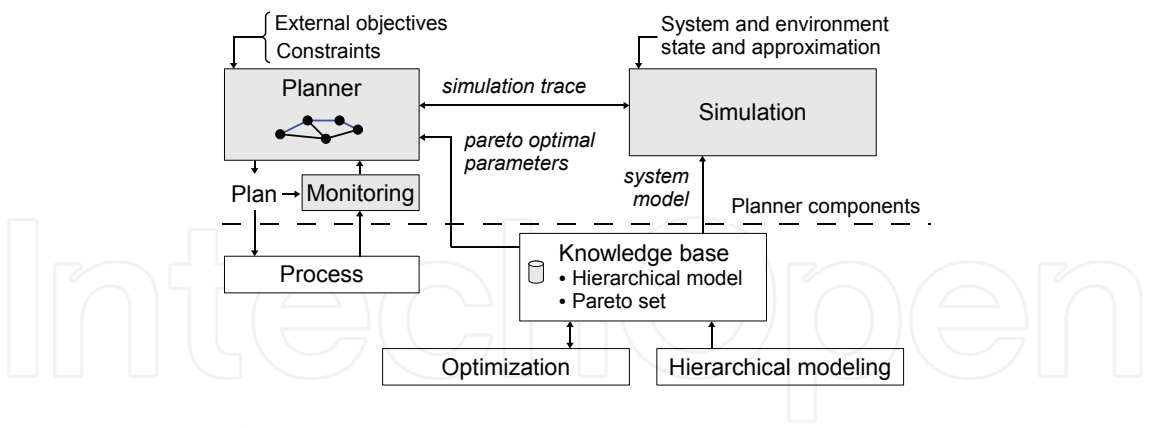


Fig. 6. Components of the hybrid planning architecture

The *planner* is equipped with overall *external objectives* and *constraints* that need to be fulfilled at any time. While creating a plan, the planner initializes a *simulation* with the best known system and environment state for the considered action, parametrizes the simulation and runs it for the projected duration of the planned step. The result of the simulation is a number of continuous value traces that are evaluated according to the constraints and objectives. The constraints are used to rule out an action, e.g. if maximum peak power is too high or comfort value used by the system is too poor during the simulation. Likewise, the global objectives, for

instance the mean comfort or energy consumption of the overall section, are used to evaluate the planned action and to decide upon possible alternatives.

The *environment state* and *approximation* input delivers the necessary information needed for simulation under the expected environmental conditions (e.g. wind, temperature, rainfall). Those parts of the system that no exact physical models are available for (e.g. "black box" modules) need to be approximated, too (system approximation). For this purpose we use fuzzy approximation which will be described in more detail and illustrated by an example in Section 6.2.

For system parts where more exact mathematical models exist, the *optimization* and *modeling* components deliver a set of pre-calculated pareto optimal controller configurations and an abstracted model of the system part. The different parameter sets are a discrete dimension of choice to the planner. Due to the applied abstraction the delivered model simulates several times faster than realtime and thus enables a predictive planning for the upcoming track sections. During runtime, delivery of a plan is guaranteed that can be updated to reflect more recent data known about the system state.

During the execution of the plan a *monitoring* component constantly compares the actual system behavior like energy use and damping performance with the predicted behavior. If deviations are detected, replanning may be necessary to choose, for instance, a safer and less energy efficient or less comfortable parametrization. Failing to do this could leave the system running out of energy while in transit, as the models used to simulate system behavior obviously do not reflect the actual system performance.

4.2 Application scenario

As already mentioned in Section 3 we present here a scenario based on some of the subsystems that are being developed around the RailCab: The air gap adjustment system and the active suspension system. They are both parts of the concept of the RailCab and have to work together onboard the vehicle. A central aspect of such mechatronic systems is their energy supply in the form of electrical power that stems from the same system bus. Since we have a mobile system with no overhead contact wire, energy is a scarce, finite resource that needs to be distributed both over time and over the consuming subsystems.

The need to plan ahead of time both distribution of energy is amplified by the fact that settings of one subsystem influence the available amount of energy. We will go on to explain how this works to find out that the scenario actually profits from the use of hybrid planning for the setting of systems.

The RailCab is equipped with a energy storage subsystem that uses capacitors and batteries to bridge times where more energy is consumed onboard that is delivered to it. Obviously this resource is finite. The energy is delivered to the system using the linear motor. In certain operational modes (acceleration and braking) electrical energy can be transferred from the stator elements in the track bed through the magnetic field into the rotor element and can be stored or consumed there. The amount of energy that can be transferred is also dependent on the air gap size between the rotor and stator elements. A larger air gap leads to less energy transfer into the vehicle while a smaller air gap allows the vehicle to extract more energy.

The air gap adjustment system is able to change the height between the rotor element of the linear motor and the nominal axis level. The part that cannot be influenced is the height of the stator element. That has a nominal height relative to the upper rail edge. That level has proven to have a rather large tolerance even on the test track that is hardly used (see the AMV results in (Schmidt, Esau, Adelt & Stern (2008))). An additional factor is the actual

diameter of the wheels that influences the distance between the axis and the rail and so influences the resulting actual air gap between stator and rotor elements. To predict the non-AGAS-originating distance components, the air gap prediction presented in Section 6 was developed. Historical data stored in decentralized track section controls provides best-effort known stator heights measured during earlier runs over the same track section (see Section 3.).

While small air gap sizes are desirable to allow more energy to be transferred, a larger air gap leaves a larger safety area for unexpected stator height deviations from the nominal or last known level. These conflicting goals need to be considered while making a decision upon set points for the AGAS that can then operate to keep the desired level. To make matters worse, every action of the AGAS also needs energy.

The active suspension system is the other subsystem considered here. It also draws power from the electrical supply and therefore needs to be conservative with energy usage while providing a maximum comfort for the passengers by damping the coach body from the disturbances introduced by the wheel-rail-contact. This leads to the desire to moderate the available energy usages and objectives between the subsystems.

The solution to these problems presented here as hybrid planning is based on choosing operational modes for the subsystems per section (called configurations) and using simulations of system models to evaluate a plan choice. The scenario focuses on the two subsystems but we will eventually need to take into account all subsystems. Every subsystem is operable with a finite number of configurations, each of them is pareto optimal regarding a set of objective functions. The common objective function here is energy use. Opposing objective functions are for example suspension quality (active suspension system) or safety (AGAS).

Since we don't have a model of the energy storage ready, we will focus on the case of a finite amount of energy onboard the vehicle that must not be depleted by the end of a travel along a network of track sections. Also we will set an upper limit on the peak energy usage that must not be exceeded at any time to keep the electrical system bus from dropping in supply voltage. Under these constraints, we will present the planning of active suspension configurations for a given set of track sections that are consecutively travelled and that exhibit different disturbances to the system. All this under the consideration of maximum peak and mean power usage, validated using system model simulations. At the same time, the air gap prediction emits the air gap that is expected if not influenced by AGAS. The presented scenario emits a plan with optimal suspension performance under the given constraints. An extension to this will be the choice of alternative track sections in the case of constraint violations.

5. Modelling and optimization of the suspension system

For design and analyzing tasks of such complex mechatronic systems as the active suspension system it is convenient to modularize and introduce a hierarchical structure as it is already done in Section 3.1. This allows to focus design activities on an entity or a limited neighborhood of entities that is less complex than the complete system. In the following we give a brief overview of our structuring concept and show how to build a hierarchical model out of such a complex mechatronic system. This model is particularly modified to be used for hierarchical optimization, that is also described in this section. The aim of the hierarchical optimization is to calculate feasible controller configurations that are used for decision making of the hybrid planning as explained in Section 4.2.

5.1 Hierarchical modelling

Often mechatronic systems can be divided into mechatronic subsystems or modules, which can be ordered hierarchically according to their function within the whole system (Lückel et al. (2001)). Such a decoupling is shown for the active suspension system in Figure 3. The information processing of each module can be described by a so called Operator-Controller-Module (OCM, see Figure 7(b)), which is explained in detail in (Hestermeyer & Oberschelp (2004)).

In this work we only need to look at the cognitive operator, which is one part of the OCM. It contains various methods to implement cognitive abilities and usually works under soft real-time constraints. Inside, there is among other things a so called knowledge base, which provides different kinds of knowledge for self-optimizing tasks. A priori there exists only information about the associated module. No information is available about overlying and underlying modules. This encapsulates a single OCM and reduces the dependencies on the remaining OCM. But to fulfill self-optimizing tasks in a hierarchically structured mechatronic system, an OCM has to have information about the behavior of the surrounding modules due to physical couplings. So knowledge has to be exchanged at runtime between the OCMs in the hierarchy.

For hierarchical optimization the knowledge mainly consists of different mathematical models. They are arranged in a hierarchical way, so that they form a hierarchical model. This hierarchical model preserves the hierarchical structure coming from the modularization and achieves the following aims. On the one hand it provides detailed mathematical models for every subsystem in the hierarchy and on the other hand it considers the behavior of the underlying subsystems only in a simplified form. Therefore the complexity and the simulation time is reduced on higher levels.

The modelling task can be divided into two main parts. Firstly a general hierarchical model called base model has to be built. In order to optimize the modelled system, the base model has to be extended by some additional parts, which results in the so-called optimization model. Both models are explained in the following.

In the base model the motion behavior of the corresponding module regarding all relevant dynamical effects is described. The structure of such a base model is shown in Figure 7(a) and consists of detailed, perhaps nonlinear mathematical models of the plant and the corresponding controller. During runtime models from the underlying subsystems are requested, so as to consider their behavior as well. This is necessary because having base models on each level of the hierarchy, which do not know anything about the underlying modules, are inapplicable for hierarchical optimization and hybrid planning. They would produce inaccurate results, as they disregard the physical couplings between the different modules. Conflating all base models to one overall system on higher levels of the hierarchy solves this problem principally, but the resulting model would be very complex and additionally has the disadvantage of increasing simulation time. Hence, only simplified mathematical models, which are generated from the base models, are exchanged between the OCMs. The orders of these simplified models are reduced as much as possible under consideration of retaining all dynamical effects that are relevant for the neighboring subsystems. In our approach we use reduced linear models to transfer knowledge between different OCMs. These reduced models are generated as follows.

The first step comprises of linearizing the base model of an OCM at the current operating point. The resulting linear system is transformed to a minimal realization and afterwards it is simplified by applying a combination of several model reduction techniques. At first the

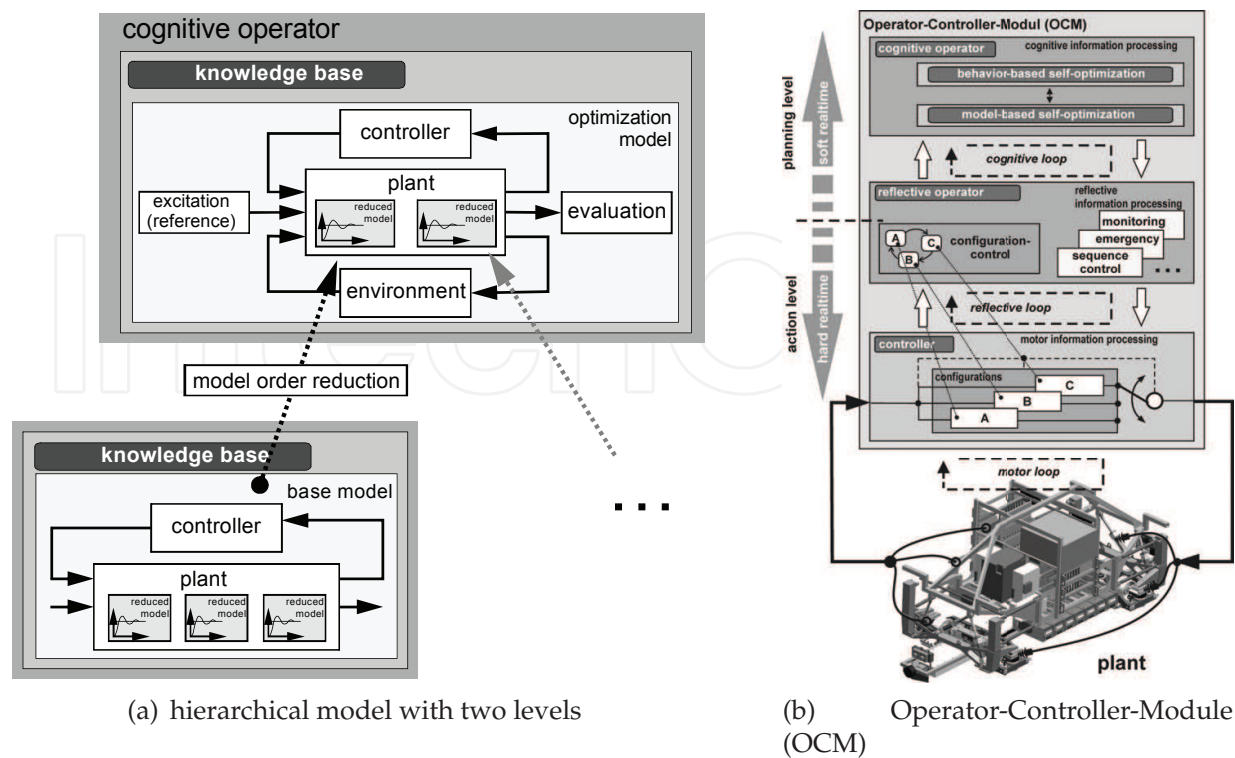


Fig. 7. Information processing for hierarchical modelling

order of the linear system is reduced by *balanced truncation* (Moore (1981)). Afterwards, a *modal model reduction* is used to remove further high eigenvalues systematically. The static error arising from both methods is reduced as much as possible by applying *singular perturbation* (both methods could be found in (Obinata & Anderson (2001))). Currently also methods for *parametric model reduction* are investigated to further improve the hierarchical modelling techniques. These methods go back to a multi-moment-matching firstly suggested in (Daniel et al. (2004)). For more details we refer to (Krüger et al. (2010)).

In order to perform model-based optimization a specific optimization model is needed. Therefore, the base model is extended by an *environment model* as well as an *excitation and evaluation model*. The structure and the interfaces of the resulting model are also shown in Figure 7(a).

The *environment model* emulates the behavior of the surrounding modules in a very simplified manner. It is generated by the overlying OCM and transferred downwards. In contrast to the environment concerning parts in the fuzzy model, presented in Section 6.2, it does not aim to approximate the real environment. It is only used to ensure a correct and reasonable optimization. In fact, this kind of model is necessary, whenever a stand-alone simulation of the base model is not reasonable, e.g. if the base model describes the operating forces of an actuator. This should become clear by looking at the optimization models of the suspension system described subsequently.

The *excitation and evaluation models* emulate excitation or reference signals and calculate the current objective functionals respectively.

A nonlinear base model as well as an optimization model emulating the respective relevant dynamical effects has been built for each module of the HiL test rig of the active suspension

system. Particularly they describe the motion behavior and the hydraulic power. In the following the main elements of each base model in the hierarchy are briefly presented.

hydraulic cylinder: The hydraulic cylinder is modeled as an actuator. The model describes particularly the cylinder and the servo valve including all effects of the control edges. The pressure supply is assumed to act as a constant pressure source and the oil is modeled as compressible. The position of the cylinder is controlled by a PD controller. A single mass, connected with the environment over a spring, works as the environment model, as the base model would be unstable without. A step signal on the input of the controller is used as reference model. Two objectives for *performance* z_{perf} and for *energy consumption* z_{energy} are defined:

$$z_{perf} = \int_{t=0}^T |x_{cyl} - x_{ref}| \cdot t \, dt \quad (2)$$

$$z_{energy} = \int_{t=0}^T (P_{cyl})^2 \, dt \quad (3)$$

Here P_{cyl} is the hydraulic power, x_{cyl} denotes the cylinder position and x_{ref} the reference position.

actuator group: Each model of an actuator group includes an MBS-model of the guide kinematic and a linear approximation of the GFK-spring. The controller uses an inverse model of the guide kinematic as well as inverse spring characteristics to control the spring forces. There is no feedback controller inside this model. A band limited white noise signal is used as reference force for the actuator group. The objectives are the deviation of the spring forces F_{act} and the reference forces F_{ref} , as well as the energy consumed by the system.

$$z_{perf} = \int_{t=0}^T |F_{act} - F_{ref}|^2 \, dt \quad (4)$$

$$z_{energy} = \int_{t=0}^T \sum_{j=1}^3 P_{cyl,j} \, dt \quad (5)$$

vehicle suspension: The base model contains the coach body and the reduced models of the actuator groups both comprehended as actuators. The controller emulates virtual spring-damper-elements between the coach body and the chassis framework and also the forces of a sky-hook damping. It can be adjusted by nine controller parameters. The calculated forces are referred to the two actuator groups. A band limited white noise signal is used as a track excitation on the chassis. Two objectives for the *comfort* and the *energy consumption* are defined for the entire system.

$$z_{comfort} = \frac{1}{T} \int_{t=0}^T |f(a_{coach})| \, dt \quad (6)$$

$$z_{energy} = \frac{1}{T} \int_{t=0}^T \sum_{j=1}^6 P_{cyl,j} \, dt \quad (7)$$

The objective $z_{comfort}$ is calculated from the frequency weighted coach body acceleration a_{coach} . The filter $f(a)$ rates the frequencies of the acceleration signal with respect to the human sensibilities (VDI 2057 Part 1: "Human exposure to mechanical vibrations – Whole-body vibration" (2004)).

Due to the model reduction the simulation step size of the base model on the topmost level can be increased by a factor of 5 compared with the nonlinear model; the reduced model on

this level, which is used for hybrid planning, can be simulated more than 56 times faster. More details can be found in (Münch et al. (2008)).

Figure 8 shows a comparison between the hierarchical model and the nonlinear model of the test rig. A controller configuration that leads to moderate power consumption has been chosen and the system has been excited with a vertical step of the railway. At the left the resulting coach body position in vertical direction is depicted. Differences are hardly identifiable in the chosen view, which accounts for the quality of the reduced model. Even the hydraulic power, that is itself a nonlinear function of the system states, can be well approximated as the right part of Figure 8 shows.

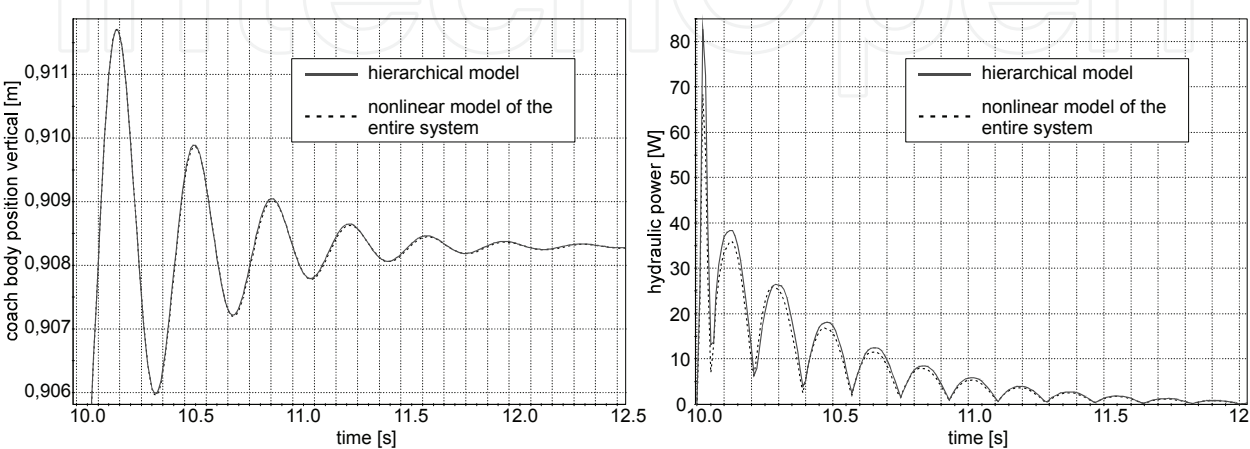


Fig. 8. Hierarchical model of the vehicle suspension compared with a nonlinear model of the entire system

5.2 Hierarchical optimization

A multitude of requirements has to be considered in a mechatronic system. Most of them can be expressed by means of feasible objective functions that can be summarized by the objective vector z . Optimization techniques are used to identify the best suited controller configurations due to the objective functions.

Mathematically, the continuous optimization problem can be formulated as follows:

$$\min_p z(p) \tag{8}$$

$$h(p) < 0 \tag{9}$$

Here, $p \in \mathbb{R}^{n_p}$ describes the design parameter space, e.g. the controller parameters and the function $h(p) : \mathbb{R}^{n_p} \mapsto \mathbb{R}^{n_h}$ describes the optimization constraints.

In general, the components of $z \in \mathbb{R}^{n_z}$ do not have a common minimum. Thus there is no unique minimum, but a set of optimal points, the so-called pareto set. Points of this set can be improved in some objectives only by deteriorating others.

There exists a multitude of methods that calculate the pareto set. There are Methods that approximate the whole pareto set, e.g. Dellnitz et al. (2005), as well as methods which compute single pareto points by summarizing the whole objective vector into a single scalar objective function, e.g. Hillermeier (2001). This objective function is then minimized by a conventional optimization method. In this paper we use the *goal attainment method* (Gembicki (1974)) to calculate single points. With this method, a pareto point with a specific objective

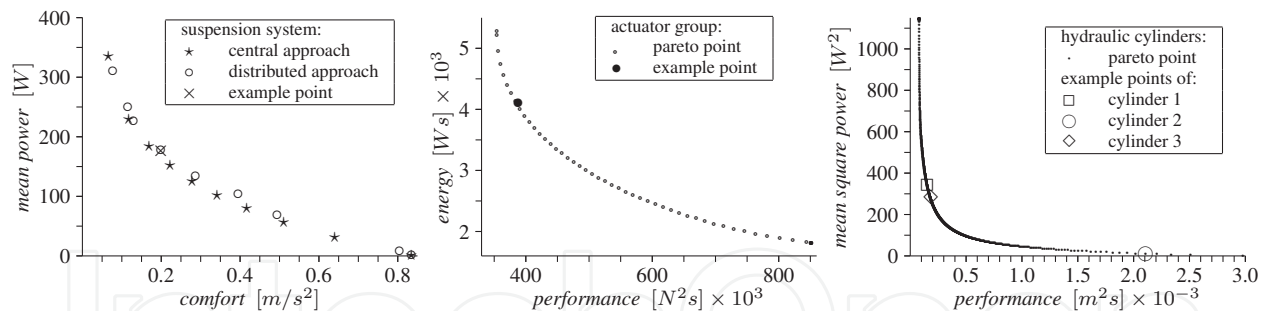


Fig. 9. Pareto sets of the different modules

ratio is selected by a weighting vector. Multiple points of the pareto set are then calculated by repeating the optimization with different weightings.

To optimize the entire hierarchical model, two different approaches are considered: a *central optimization* and a *distributed optimization*.

The central optimization is the simplest approach of optimizing the hierarchical model. Only the objective functionals of the topmost level are considered. Objectives of the lower levels could be regarded as additional optimization constraints if necessary. The optimization, i.e. the controller parameters, are directly transferred from the topmost level of the hierarchy to the underlying modules. Therefore the structure of the whole system has to be accessible in the optimization process.

This contradicts the idea of encapsulated modules that exchange as few information as possible, which has been one aim of the hierarchical modelling. Our second optimization approach, the distributed optimization, overcomes this shortcoming. The main idea of the distributed approach is to decompose the whole optimization problem into several smaller, less complex optimizations. Each OCM performs its own optimization of the optimization model considering only its “local” objectives. It influences the remaining system by setting its own “local” design parameters and also by referring aims to underlying OCMs. Due to the latter the optimizer gets the ability to affect the behavior of the underlying systems. On the basis of its pareto set the underlying system determines the actual controller parameter, adjusts the base model and transfers the new reduced base model back to the overlying OCM. Since the pareto sets of the underlying systems have to be known by the optimizer, the distributed optimization is executed sequentially from the lowest to the topmost level.

As mentioned, the exchange of information between different OCMs is restricted to transferring aims s_i and reduced base models m_i . A direct access to the design parameters or the values of the objective functionals of the underlying systems does not take place. The determination of the actual design parameters is realized in the more abstract way of specifying aims. Thus the exchanged information is highly generic and this leads to a strong encapsulation of the OCM.

The pareto sets of the different modules are plotted in in Figure 9. Due to the symmetry of the two actuator groups their pareto sets are identical and equal aims are used during the distributed optimization process. The pareto sets of the six different hydraulic cylinders are also identical. In the left graph one can see a comparison of both optimization approaches. The central approach performs slightly better than the distributed optimization. This can be put down to the fact that the design parameters of the vehicle suspension are limited to the pareto sets of the underlying system by using the distributed approach.

In order to point out the interrelation between the coach body, the actuator groups and the hydraulic cylinders, an example point of the pareto set of the coach body has been chosen and

the corresponding pareto points of the underlying actuator groups and hydraulic cylinders are marked. Although the three hydraulic cylinders are described by identical mathematical models, the distributed optimization results in different pareto points for each cylinder. This is caused by the different mounting positions of the cylinders. Cylinder 2 has less influence to the comfort, since the optimizer determined a poor *performance* for the benefit of a very low energy consumption.

The pareto set of the topmost level (suspension system) is used for the hybrid planning.

6. Modelling for air gap adjustment and prediction

While the active suspension system in the previous section is a good example for the use of the hierarchical OCM (Operator-Controller-Module) structure, the AGAS subsystem is modelled as a single OCM. It illustrates different kinds of models and their role during the design and operation of self-optimizing mechatronic systems. First Section 6.1 describes a physical model of the AGAS, which is used for system control and for determination of the actual air gap during operation based on measurements of physical parameters, i.e. proper adjustment of the air gap. Such a physical model can hardly consider all parameters influencing a system's behavior or the variety of operating conditions that may arise for instance due to environmental conditions (wear and tear, weather, etc.). These conditions are however important for the hybrid planner to predict the dynamic course of the air gap for future track sections via its simulation component. In such cases a fuzzy model, as described in Section 6.2 for the air gap, is used. It provides a promising alternative to predict the system behavior considering all relevant aspects of such an eventually non-linear system. This fuzzy model is initially based on expert knowledge and may be incrementally improved by measurements gained during operation based on the physical models.

6.1 Physically motivated modelling of the AGAS

To control and optimize the AGAS a physically motivated approach is recommended. Based on (Adelt et al. (2008)) an air gap adjustment system is developed to reach optimal system behavior. It consists of a linear drive and an adjustment actuator which allows a vertical adjustment of the rotor. The electric drives and the mechanical structure can be described clearly by applying physical methods and techniques.

The optimization of the propulsion system requires the optimal compromise according to the objective functions of the linear drive and the adjustment actuator. The losses of the linear drive P_{ls} and the adjustment power P_{aa} will be minimized. The electrical power P_b fed into the RailCab should be a maximum value. The losses will be reduced by minimization of the air gap and the fed-in power. A small air gap counteracts to the minimization of the adjustment power. This contrast is founded in the return springs located at the rotor. They are working against the adjustment actuator. The optimal compromise between the diverging objective functions will be strived by a multi-objective-optimization. It calculates the optimized vectors for the air gap δ_{opt} , the rotor frequency $f_{r,opt}$, the RailCab speed $v_{m,opt}$ and the propulsion Force $F_{m,opt}$ of the drives. From these vectors the air gap control and the operating point control of the propulsion system computes the setpoint-values for the rotor and stator current i_{rq}^* and i_{sd}^* , the rotor frequency f_r^* of the linear drive, the RailCab speed v_m^* and stator current i_{sq}^* of the adjustment actuator. The rotor speed ω_r and rotation angle e_r of the adjustment actuator will be measured by an integrated resolver. The stator current i_{sq} of the adjustment actuator, the stator and rotor current i_{sd} and i_{rd} of the linear drive will be measured by power electronics. The linear drive temperature ϑ_{ld} will be computed by the drive currents. This

temperature also influences to the multi-objective-optimization because of increasing losses at increasing drive temperature. A direct link exists between the linear drive and the adjustment actuator by the system-inherent normal force F_n (Figure 10). The spring forces F_s are working

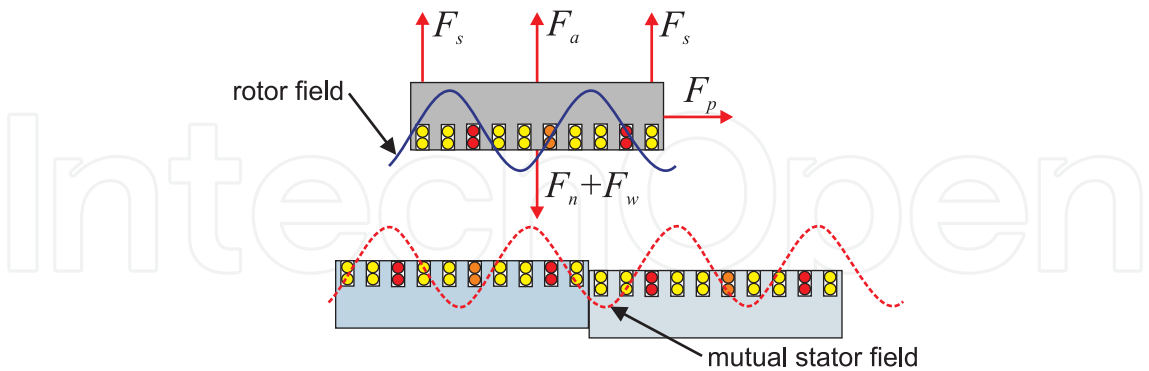


Fig. 10. Forces at the linear drive of the AGAS

against the normal and weight force F_w . Therefore the adjustment force F_a allocated by the adjustment actuator can be reduced. Hall sensors measure the magnetic flux B_s of the stator field (Figure 11). Among other things the air gap δ , rotor frequency f_r and RailCab speed v_m can be determined. A detailed description of air gap and stator frequency measurement will be presented below. The constraints ct of the optimization will be preset by the requested driving profile. Besides these constraints a collision of stator and rotor has to be avoided. The driving profile is composed of the position, the speed and the acceleration which has to be achieved by the RailCab. To avoid a collision of the stator and the rotor the RailCab speed and the maximum of the reachable adjustment acceleration have to be considered. In addition the self-optimizing air gap adjustment system fetches up data from the track section control (TSC) described in Section 3. This information includes inter alia the stator positions of the track section and also have influence on the multi-objective-optimization. The value of the

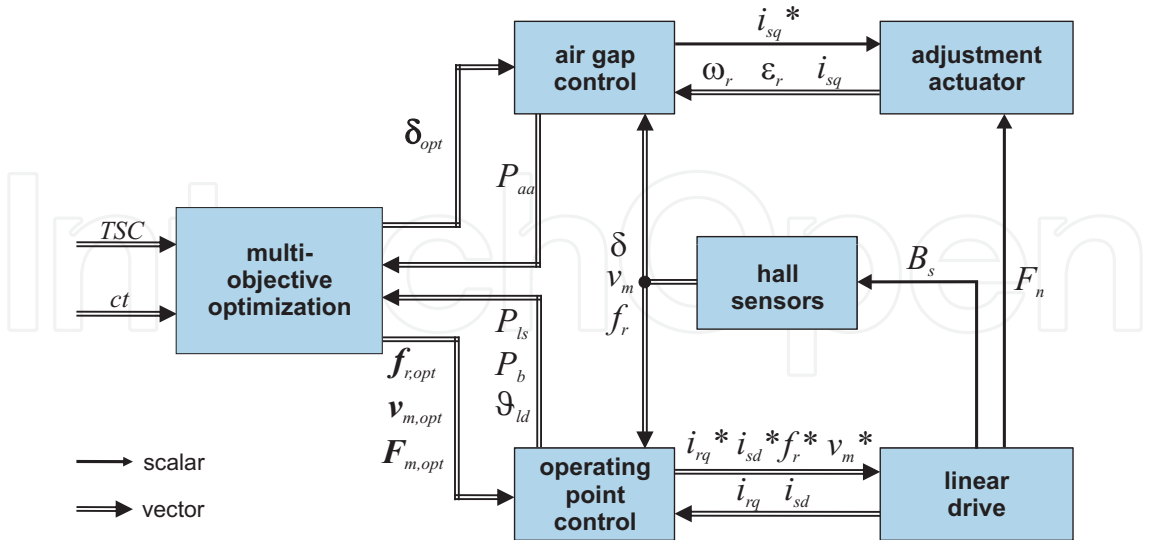


Fig. 11. Block diagram of the AGAS

air gap is also essential for the air gap control. It is possible to model the air gap gradient by a fuzzy-model taking into account enviromental influences as described in Section 6.2. But also a continuous measurement of the air gap is achievable. To measure the air gap of

a linear drive different sensors are conceivable. In (Gabel (2009)) the air gap is measured by an inductive distance sensor. Based on (Böcker et al. (2006)) a hall sensor was chosen for the self-optimizing air gap adjustment system. This sensor detects the magnetic flux of the stator field. The arrangement of two sensors with a distance of half a pole pitch allows the detection of the phase angle ϵ_r of the stator field. Therefore the position and the speed of the rotor can be determined in parallel to the air gap. The hall sensor arrangement features a high level of function integration. For measuring the air gap the sinusoidal output signals will be rectified. The value of the signal will be associated to the air gap by using an appropriate calibration (Figure 12). Air gap gradients will be detected by comparing actual measured data with data of the past.

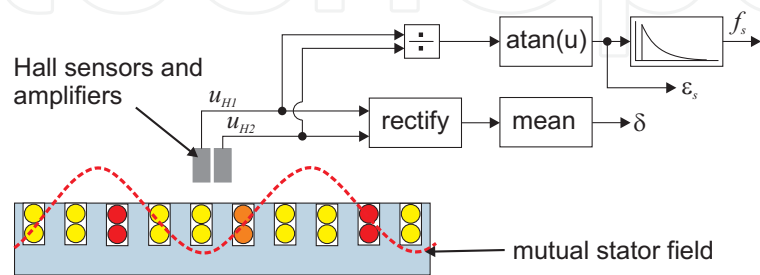


Fig. 12. Hall sensor arrangement

6.2 Fuzzy model for air gap prediction

A fuzzy model of the air gap is used by the simulation component to predict the expected air gap δ_{ag} during system operation given the system and environment state. Two fuzzy models were implemented that reflect the physical system: One model describes the influence of the track and the integrated stator elements on the air gap and a second one describes the influence of the vehicle including the rotor element. Together these models comprise the system's and the environmental conditions that influence the air gap's change.

6.2.1 Fuzzy modelling

The process of modelling was also separated into two steps. During *structure identification* the influencing factors for the shuttle and the track fuzzy model were determined and structured in a so called *influencing factor tree* (IFT, Figure 13). These input variables and the model outputs were fuzzified, and two fuzzy rule systems describing the input-output dependencies for the shuttle, respectively the track fuzzy model, were constructed. The subsequent *parameter adaptation* step is responsible for adapting the parameters of the fuzzy membership functions of the system's input and output variables, for adapting the structure of the IFT and the set of rules and their weights according to new measurement data. Structure identification and parameter adaptation for the AGAS are described below in more detail.

Structure Identification

At first the influencing factors for the two AGAS models shuttle and track are classified and represented as an influencing factor tree. The root node of an IFT represents the model output, i.e. the change of the air gap resulting from the shuttle $\Delta\delta_{shuttle}$ or the track $\Delta\delta_{track}$ in our case. The total change of the air gap $\Delta\delta_{ag}$ can then be calculated as

$$\Delta\delta_{ag} = \Delta\delta_{shuttle} + \Delta\delta_{track} \tag{10}$$

The leaf nodes of the IFT represent the influencing factors, i.e. the models’ inputs, and the internal nodes of the tree mirror the classification hierarchy. Figure 13 shows part of the IFT for the shuttle fuzzy model.

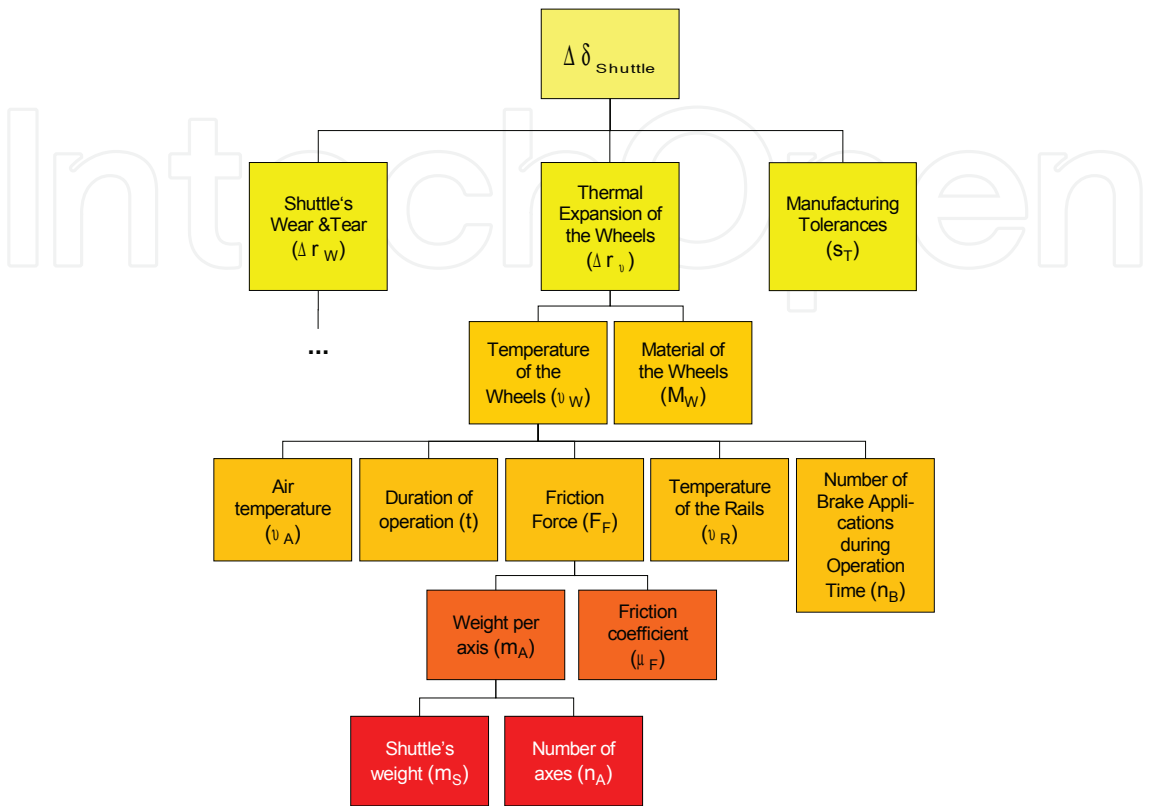


Fig. 13. Influencing factor tree (IFT) of the shuttle fuzzy model

In the shuttle fuzzy model the change of the air gap depends on three classes of influencing factors: *shuttle’s wear and tear*, *thermal expansion of the wheels* and *manufacturing tolerances* (see Figure 13). While *manufacturing tolerances* is a leaf node of the tree corresponding to an influencing factor, *shuttle’s wear and tear* and *thermal expansion of the wheels* in turn depend on several classes of factors. As leaf nodes in the sub tree describing *thermal expansion of the wheels* the influencing factors *material of the wheels*, *air temperature*, *duration of operation*, *temperature of the rails*, *number of brake applications during operation time*, *friction coefficient*, *shuttle’s weight*, *number of axes* are distinguished. The classification hierarchy for *shuttle’s wear and tear* is omitted due to spatial reasons.

After their classification the influencing factors (leaf nodes of the IFT) first have to be fuzzified. As an initial choice, symmetrical triangular membership functions separating the influencing factors’ domains in equal parts are reasonable. The domains and the number of membership functions (ranging between three and seven) were determined by domain experts. The *shuttle’s weight*, for instance, may range from 6 tons to 32 tons. It is split into five parts by the membership functions *small*, *smallmedium*, *medium*, *mediumlarge*, *large*. The *number of axes* for instance, lies between 2 and 8 and is represented by three membership functions *small*, *medium*, *large*.

Based on the fuzzification of all influencing factors and factor classes (internal IFT nodes) the fuzzy rule bases for the shuttle and track fuzzy models were developed, relying on the knowledge of domain experts. Each rule base is described by a so called knowledge matrix.

$m_a \setminus \mu_f$	small	medium	large
small	small	small	small
medium	small	medium	medium
large	medium	large	large

Table 1. Knowledge matrix related to friction force

For example, the rule base relating to the *friction force* contains 9 rules, because the influencing factors *weight per axis* (m_a), *friction coefficient* (μ_f) and the *friction force* (F_f) are represented by three membership functions each. The corresponding knowledge matrix is shown in Table 1. The upper row relates to μ_f , the leftmost column to m_a , and the body of the matrix specifies the resulting fuzzy value of F_f .

The knowledge matrix of the *friction force* is completely occupied. But some knowledge matrices were only sparsely occupied. Hence, the initial rule base contained only a few significant rules. For example the rule base of the *temperature of the wheels* has only 17 rules (the complete rule base of the *temperature of the wheels* would contain 2205 rules).

Similar matrices were developed for all leaf and internal nodes of the shuttle and the track IFTs and combined to two knowledge matrices for the complete shuttle and track fuzzy models. The complete rule bases hence reflected the IFT structure in their rule dependencies. The initial knowledge matrix of the shuttle fuzzy model contained 102 rows and the one for the track fuzzy model contained 40 rows. These initial rule bases were tested and adapted during the parameter adaptation phase described below.

Parameter adaptation

The correctness of the initial rule bases was checked by assigning values to the input parameters (influencing factors corresponding to IFT leaf nodes) for which the output (change of the air gap) was known. As expected, the outputs of the two rule bases deviated significantly from the expected ones. In order to find out how parameters e.g. of membership functions should be adapted each rule base was decomposed into sub rule systems corresponding to sub trees in the IFT, that span only two levels of hierarchy and had one root node. Hence, for these sub rule bases the input parameters (corresponding to the leaf nodes of the sub tree) could be directly assigned and their influence on the output (corresponding to the sub tree’s root) could be observed, facilitating an adaptation of rules and membership functions. In some cases also new nodes were inserted into the IFT hierarchy. For instance in the sub tree describing the *thermal expansion of the wheels* a new internal node *temperature* was inserted to combine the influencing factors *temperature of the rails* and *air temperature*. After decomposing the models and extending the IFT, the rule bases were extended. For the shuttle fuzzy model 534 rules were created and the track fuzzy model contained 81 rules.

6.2.2 Air gap prediction by the TSC

Starting point for the air gap prediction is the path along track sections selected by the hybrid planner. The driving actions planned for these track sections have to consider several parameters for the AGAS in order to find a tradeoff between safe, energy efficient or energy maximizing operation, that fulfills the given constraints. The track section control (TSC) is responsible for predicting the air gap for a track section that is currently planned to be traveled next. For this purpose it relies on historic data and on the two fuzzy models for the shuttle and the track. This allows arriving RailCabs to request a forecast of the air gap in order to adjust an optimal air gap according to their internal system of objectives. An important aim of the TSC

is also to deliver an exact prediction of the air gap’s change that depends on many influencing factors like environment conditions, RailCab and track properties. The actual values of these influencing factors are needed as inputs for the *Air Gap Fuzzy Model* and are determined by the *RailCab System*, the *Track System* and the *Environment System*. These calculations have to consider several mutual interdependencies. This structure is depicted in Figure 14.

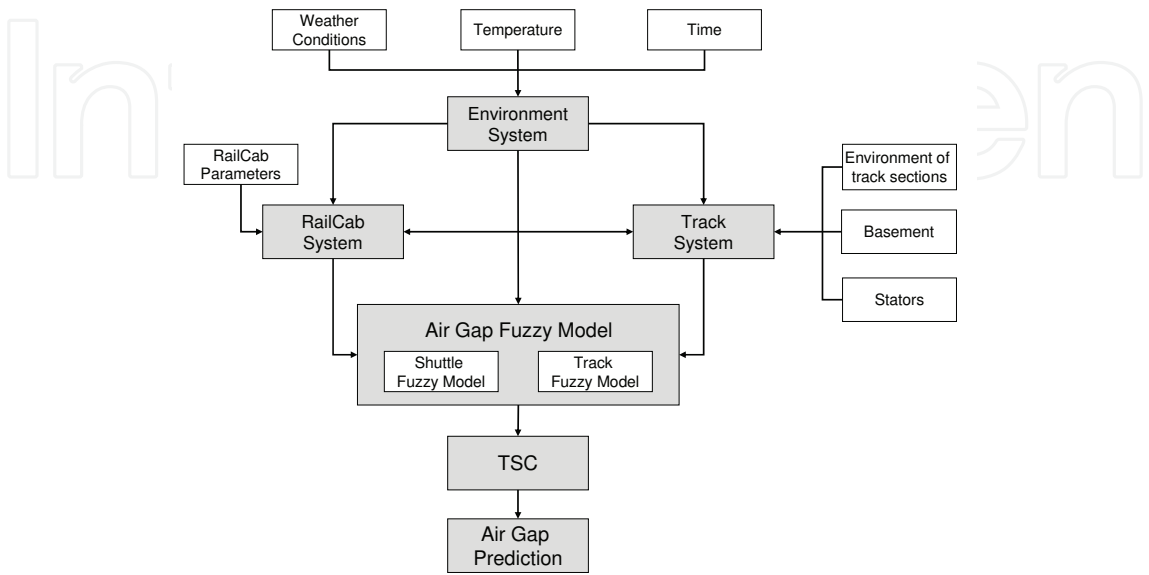


Fig. 14. Principle concept of the air gap prediction by the TSC

The *Track System* determines the factors derived from the track itself, as well as their change, e.g., the wear of the rails. As already mentioned, the *RailCab System* and the *Track System* have also effects on the properties of each other. A track is made up of different section elements like *track switch*, *station*, *bridge* or *normal section*. The track sections are surrounded by one specific environment like *forest*, *ocean*, *shadow*, *normal* and *tunnel*. The basement can be specified to be *gravel* or *concrete*. These parameters influence parts of the fuzzy approximation model for the air gap. Additionally, each section element is characterized by a slope value. A track section consists of a number of stators that have a fixed length and are layed out with a regular spacing. Each stator has *sleeper distance*, *profiling of the rails*, *assembling tolerance* and *manufacturing tolerance* as parameters which are used in the fuzzy approximation model of the air gap.

The *RailCab System* handles all factors which are influenced and determined by a RailCab itself, as for example its weight and the number of its axes. The RailCab parameters for the *Air Gap Fuzzy Model* can be configured as well: *average weight per wheel*, *total distance per wheel*, *total mass* and *number of brake activations*.

The *Environment System* includes weather, temperature and time data which influence on the one hand the *RailCab System* and the *Track System*. For example, rain lowers the friction coefficient between rail and wheel. On the other hand, the *Environment System* provides also the input variables, like e.g. temperature, for both fuzzy models. A forest element, for example, has leaves on the rail in autumn and is generally cooler than open terrain. Through the systems of the *RailCab System*, the *Track System* and the *Environment System* the TSC can adapt the input parameters of the *Air Gap Fuzzy Model* dynamically to the respective circumstances.

7. Evaluation

We are going to evaluate the hybrid planning approach in two parts. The first part will cover the pure simulation based planning results, followed by the results of the air gap modelling based on an example scenario.

7.1 Simulation based planning

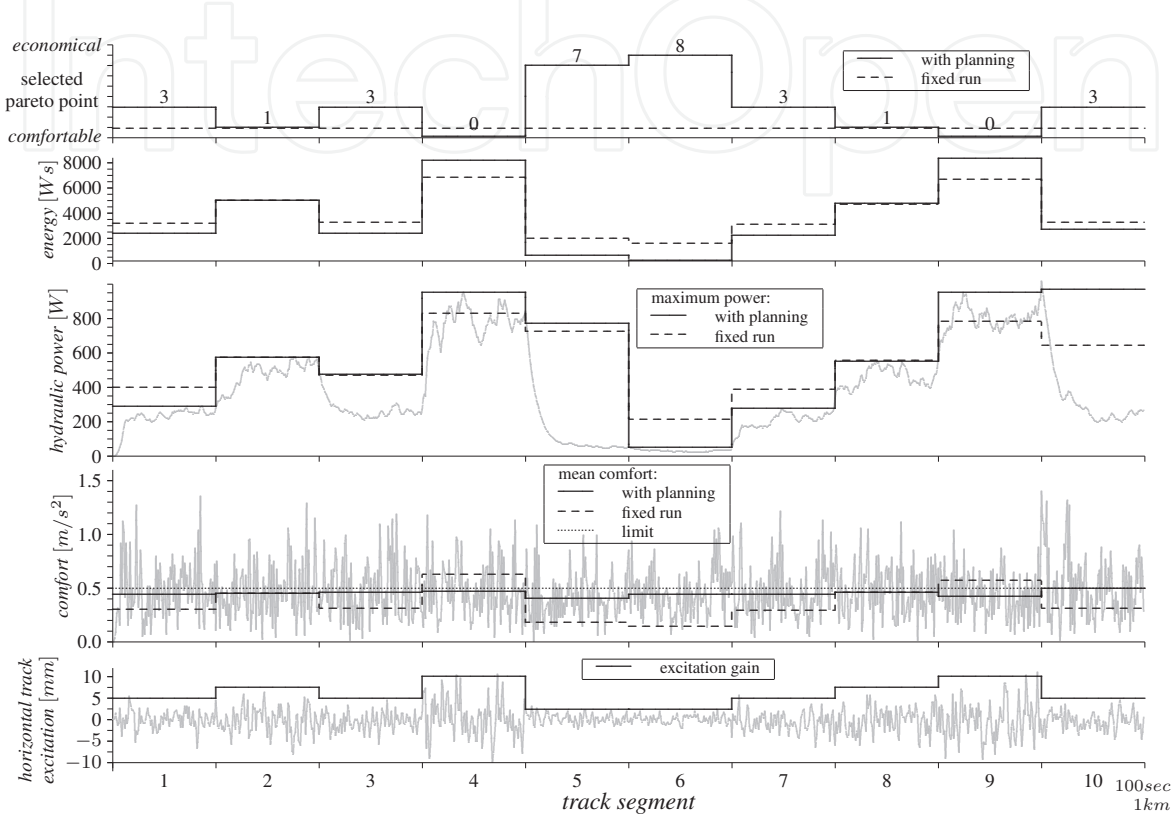


Fig. 15. Energy use (third chart from top) and comfort levels (fourth chart) on planned and static (fixed) configuration. Lower values are better.

Figure 15 shows an example run of a RailCab. It drives with a constant speed over a course with a random but fixed excitation profile. The excitation acts as a disturbance that the active suspension controller needs to damp in order to reach a lower comfort value. The passenger experiences less vibration in a setting with a low comfort value. The system has the choice among ten pareto optimal configurations for the active suspension module resulting from the previous described hierarchical optimization (Figure 9). Each pareto point, numbered from 0 to 9, is a tradeoff between energy use (maximized at *economical*, number 9) and comfort (maximized at *comfortable*, number 0). The topmost chart in Figure 15 illustrates the chosen configuration in the scenario.

The course is divided into ten sections. Each of these sections is configured with a factor that multiplies the otherwise equally distributed excitation profile, resulting in the excitations plotted in the lowermost chart in Figure 15. This is used to realize different types of rail quality. Larger factors correspond to larger disturbances.

We compare two types of operation. In a fixed run, we choose one configuration for all the sections. The configuration is chosen by maximizing the comfort while not exceeding the energy budget limit. The chosen configuration is pareto point 1.

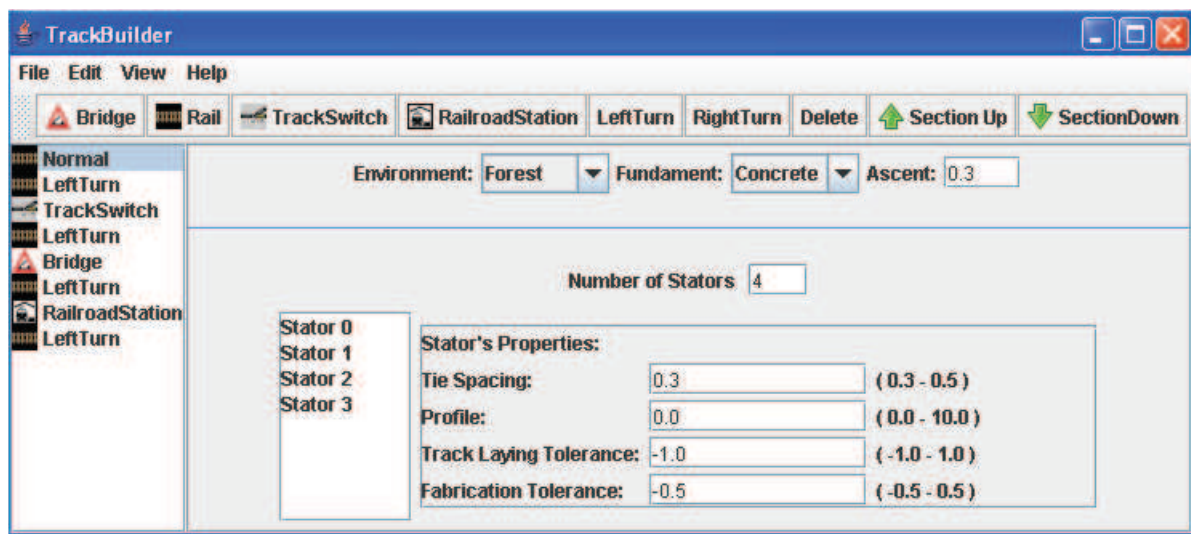


Fig. 16. TrackBuilder interface for building simulated track sections

The other run is the result of the hybrid planner trying to maximize the passenger comfort while staying in all the limits that the fixed configuration had to stay in, too. The figure shows that the upper limit for the comfort could only be met using the planned version. Using the static configuration, sections 4 and 9 violate the limit. The planned version uses sections with little excitations (1-3 and 5-8) to sacrifice comfort to save some energy. This is then used to improve comfort in the critical sections without violating the energy budget limit. This way, sections 4 and 9 can be travelled with a comfort level well in the set limit. In the planned run the energy use was 6,8% lower. At the same time, the peak energy use was higher.

7.2 Air gap prediction sample scenario

For executing the simulation steps with various parameter values the Track Sectioning Control Simulation System (TSCSS) is realized. The TSCSS is used to test and analyze the air gap system using simulated tracks and simulated environment conditions. The system is split up into two components. The first one called *TrackBuilder* is used to construct a model of track sections (Figure 16). The second component of the TSCSS is the simulation system with the track visualisation of a simulated track network.

The example scenario consists of a course, a RailCab and a sample weather profile. The course consists of 8 track sections with 5 stators each. The sections are a train station, a forest, a bridge, a tunnel, a sea-side section, another train station, a track switch and another forest. These sections have properties, which are influenced by the environment such as a rail temperature. A train station, for example, is roofed, so there is no influence of rain or snow and therefor the friction coefficient between rail and wheel does not lower in this section. In a tunnel there's no influence of solar radiation. As a result, the tunnel conditions do not lead to a rail temperature increase and therefor there is no rail expansion.

We simulate a RailCab travelling over this course and calculate once per stator, what air gap value is predicted. To do this we use properties of the stator, the section, the RailCab and the current air temperature along with the other weather properties. In the simulation the RailCab had to move for 60,000 kilometers. To obtain realistic simulation results we also consider specific properties of the RailCab like wheel wear. The configuration of the RailCab is that the wheels were profiled each 15,000 kilometers. This leads to a strong increase of a wheel wear.

For each of the 40 stators there is a temperature and an air gap change. In Figure 17 the predicted air gap deviation from the nominal value of 10 mm is shown. The resulting value of the air gap change is between -16 mm and -17.4 mm. So there is a strong deviation from the nominal value because of the high wheel wear.

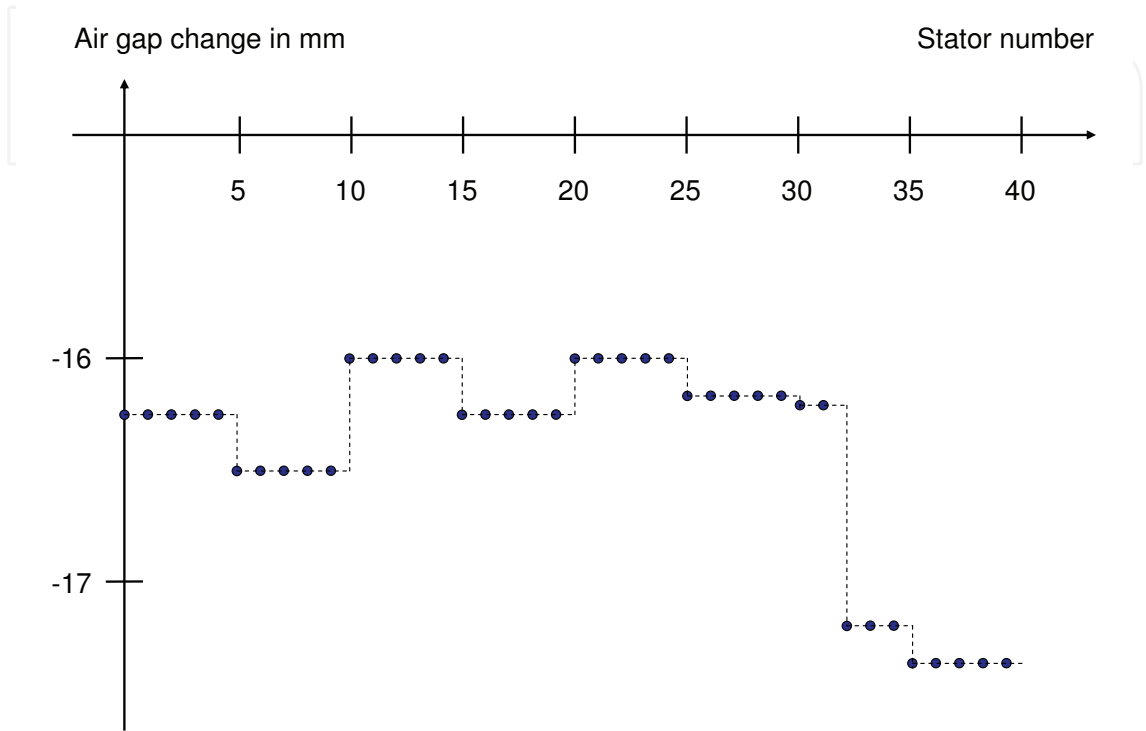


Fig. 17. Predicted air gap change over the simulated course of 40 stators

One noteworthy point is the section with the track switch between stator number 30 and 35. Before and after the track switch, different properties of the stators result in a rather large change in the air gap.

In our presented example, the planner chooses a path along track sections. While considering an action, it considers parameters for the AGAS selecting a preference to maximize the safety. Along with historic data of the TSC, the fuzzy modeled air gap can be simulated for the track section and the actual modeled air gap can be evaluated against safety margins. The prediction we see here could now be used to have the AGAS keep the desired nominal air gap in order to avoid a collision between the stators and vehicle’s rotors.

8. Conclusion and outlook

This paper presented a new approach for self optimization based on hybrid (re)planning and its application in a railbound mechatronic system. In contrast to many existing approaches not only discrete actions are planned but also continuous system aspects are taken into account. The planner predicts future system behavior based on mathematical or physical models of system (parts) when possible. However, these are often not available as for instance for environmental conditions. For such cases fuzzy models to predict the behavior of respective system parts - in our example for the AGAS were incorporated. Furthermore, the planner

is able to online monitor the execution of a plan and initiate replanning if necessary. Such a replanning can not only change the course of action but also the planning goals which is essential for self-optimization. Evaluations of our approach revealed that hybrid planning with online adjustments to actual system and environment conditions can considerably improve the system performance compared to an a priori fixed plans and fixed parameters from a set of parameter settings determined e. g. by Pareto optimization. Furthermore, it is able to detect impending future violations of system constraints like peak power or comfort level and avoid them by corresponding online parameter and plan adaptations. Ongoing work will try to distribute the planning process along the hierarchical structure of mechatronic systems using the tree-like properties to distribute work load and improve planning results.

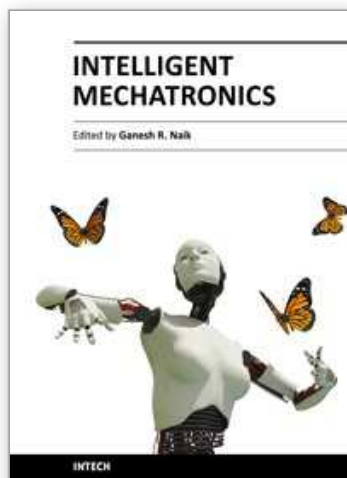
9. References

- Adelt, P., Donoth, J., Gausemeier, J., Geisler, J., Henkler, S., Kahl, S., Klöpper, B., Krupp, A., Münch, E., Oberthür, S., Paiz, C., Podlogar, H., Pormann, M., Radkowski, R., Romaus, C., Schmidt, A., Schulz, B., Vöcking, H., Witkowski, U., Witting, K. & Znamenshchykov, O. (2008). *Selbstoptimierende Systeme des Maschinenbaus – Definitionen, Anwendungen, Konzepte.*, Vol. 234, HNI-Verlagsschriftenreihe, Paderborn.
- Adelt, P. & Klöpper, B. (2009). Buildings blocks and prototypical implementation of a hybrid planning architecture, in B. Klöpper & W. Dangelmaier (eds), *Self-x in Engineering*, MV Verlag, pp. 55–67.
- Böcker, J., Henke, C., Schneider, T. & Schulz, B. (2006). Redundante Positionserfassung für ein spurgeführtes linearmotorgetriebenes Bahnfahrzeug, 4. Workshop Entwurf mechatronischer Systeme, Heinz Nixdorf Institute, Paderborn, Germany.
- Chiou, K.-C. & Huang, S.-J. (2007). An adaptive fuzzy controller for 1/2 vehicle active suspension systems, *IEEE International Conference on Systems, Man and Cybernetics (SMC 06), October 08-11, 2006, Taipei*, IEEE Service Center, Piscataway, NJ, pp. 1010–1015.
- CRC 614 (2010). University of Paderborn, <http://www.sfb614.de/en/sfb614/>.
- Daniel, L., Siong, O. C., Chay, L. S., Lee, K. H. & White, J. (2004). A multiparameter moment-matching model-reduction approach for generating geometrically parameterized interconnected performance models, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 23(5): 678–693.
- Dearden, R. (2005). Planning and learning in hybride discrete domains - continuous models, *20th International Joint Conference on Artificial Intelligence*.
- Dellnitz, M., Schütze, O. & Hestermeyer, T. (2005). Covering pareto sets by multilevel subdivision techniques, *Journal of Optimization, Theory and Applications* 124(1): 113–136.
- Esau, N., Adelt, P., Schmidt, A. & Rose, M. (2008). Interval optimization in a hybrid planner context, *Asia International Symposium on Mechatronics*.
- Gabel, R. (2009). *Aktive Luftspaltregelung von Antrieben in linearen Transportsystemen*, Cuvillier Verlag Göttingen.
- Geisler, J. & Trächtler, A. (2009). Control of the pareto optimality of systems with unknown disturbances, *7th IEEE International Conference on Control & Automation (ICCA 2009)*, Christchurch, New Zealand.
- Gembicki, F. W. (1974). *Vector Optimization for Control with Performance and Parameter Sensitivity Indices*, PhD thesis, Case Western Reserve Univ., Cleveland, Ohio.
- Ghallab, M. & Traverso, P. (2004). *Automated Planning: Theory and Practice*, Morgan Kaufmann.

- Henke, C., Tichy, M., Schneider, T., Böcker, J. & Schäfer, W. (2008). System architecture and risk management for autonomous railway convoys, *IEEE International Systems Conference*, Montreal, Canada.
- Hestermeyer, T. & Oberschelp, O. (2004). Structured information processing for self-optimizing mechatronic systems, *1st International Conference on Informatics in Control, Automation and Robotics*, Setubal.
- Hillermeier, C. (2001). *Nonlinear multiobjective optimization: a generalized homotopy approach*, Birkhäuser Verlag, Basel.
- Jin, Y., Yu, D. & Song, X. (2007). An integrated-error-based adaptive neuron control and its application to vehicle suspension systems, *IEEE International Conference on Control and Automation (ICCA 2007)*, Guangzhou, China, pp. 564–569.
- Kambhampati, S., Cutkosky, M. R., Tenenbaum, J. M. & Lee, S. H. (1993). Integrating general purpose planners and specialized reasoners: Case study of a hybrid planning architecture, *IEEE Transactions on Systems, Man and Cybernetics* 23(6): pp.1503–1518.
- Kambhampati, S., Mali, A. D. & Srivastava, B. (1998). Hybrid Planning for Partially Hierarchical Domains, *AAAI/IAAI*, pp. 882–888.
- Kou, F. & Fang, Z. (2007). An experimental investigation into the design of vehicle fuzzy active suspension, *IEEE International Conference on Automation and Logistics*, Jinan, China, pp. 959–963.
- Krüger, M., Scharfenbaum, I. & Trächtler, A. (2010). Parametrische Modellreduktion in hierarchisch modellierten selbstoptimierenden Systemen, in J. Gausemeier, F. Rammig, W. Schäfer & A. Trächtler (eds), *7. Paderborner Workshop Entwurf mechatronischer Systeme*, Vol. 272, HNI-Verlagsschriftenreihe, pp. 203–218.
- Li, H. & Goodall, R. M. (1999). Linear and non-linear skyhook damping control laws for active railway suspensions, *Control Engineering Practice* 7: pp. 843–850.
- Lückel, J., Hestermeyer, T. & Liu-Henke, X. (2001). Generalization of the cascade principle in view of a structured form of mechatronic systems, *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Villa Olmo, Como, Italy.
- Moore, B. C. (1981). Principal component analysis in linear systems – controllability, observability, and model reduction, *IEEE Transactions on Automatic Control* 26(1): pp. 17–32. IEEE Control System Society, Boston.
- Münch, E., Adelt, P., Krüger, M., Kleinjohann, B. & Trächtler, A. (2008). Hybrid planning and hierarchical optimization of mechatronic systems, *International Conference on Control, Automation and Systems*, Seoul, Korea.
- Nutt, P. C. (1982). Hybrid Planning Methods, *The Academy of Management Review* 7(3): pp. 442–454.
- Obinata, G. & Anderson, B. D. O. (2001). *Model Reduction for Control System Design*, Springer-Verlag, London. ISBN 1-85233-371-5.
- RailCab (2010). <http://www.railcab.de/>;
- Russell, S. & Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*, 2nd edn, Prentice-Hall, Englewood Cliffs, NJ.
- Schmidt, A., Adelt, P., Esau, N., Kleinjohann, L., Kleinjohann, B. & Rose, M. (2008). Approximation of environment models for an air gap adjustment system in a hybrid planning context, *IEEE Multi Conference on Systems and Control*, San Antonio, Texas, USA.

- Schmidt, A., Esau, N., Adelt, P. & Stern, C. (2008). Dynamic optimization of track sectioning in mechatronic systems for a hybrid planner, *Proceedings of IEEE International Conference on Mechatronics and Automation (ICMA)*, Takamatsu, Kagawa, Japan, pp. 653–658.
- Schneider, T., Schulz, B., Henke, C., Witting, K., Steenken, D. & Böcker, J. (2009). Energy transfer via linear doubly-fed motor in different operating modes, *Electric Machines and Drives Conference (IEMDC2009)*, Miami, Florida, USA.
- Stribrsky, A., Hyniova, K., Honcu, J. & Kruczek, A. (2007). Energy recuperation in automotive active suspension systems with linear electric motor, *Mediterranean Conference on Control & Automation (MED 07)*, Athens, Greece.
- VDI 2057 Part 1: “Human exposure to mechanical vibrations – Whole-body vibration” (2004). Beuth Verlag, Berlin.
- Vöcking, H. & Trächtler, A. (2008). Self-optimization of an active suspension system regarding energy requirements, *International Conference on Control, Automation and Systems*, Seoul, Korea.
- Yousefi, A., Akbari, A. & Lohmann, B. (2006). Low order robust controllers for active vehicle suspensions, *IEEE International Conference on Control Applications (CCA 06)*, Oktober 04-06, 2006, Munich, IEEE Operations Center, Piscataway, NJ, pp. 693–698.
- Zhang, Y., Huang, K., Yu, F., Gu, Y. & Li, D. (2007). Experimental verification of energy-regenerative feasibility for an automotive electrical suspension system, *IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, Beijing, China.
- Zimmer, D., Böcker, J., Schmidt, A. & Schulz, B. (2005). Direktantriebe passend ausgewählt, *Antriebstechnik* 02: 2–6.
- Zimmer, D. & Schmidt, A. (2005). Linear läuft es schneller, *Antriebstechnik* 02: pp. 2–5.

IntechOpen



Intelligent Mechatronics

Edited by Prof. Ganesh Naik

ISBN 978-953-307-300-2

Hard cover, 248 pages

Publisher InTech

Published online 28, February, 2011

Published in print edition February, 2011

This book is intended for both mechanical and electronics engineers (researchers and graduate students) who wish to get some training in smart electronics devices embedded in mechanical systems. The book is partly a textbook and partly a monograph. It is a textbook as it provides a focused interdisciplinary experience for undergraduates that encompass important elements from traditional courses as well as contemporary developments in Mechatronics. It is simultaneously a monograph because it presents several new results and ideas and further developments and explanation of existing algorithms which are brought together and published in the book for the first time.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Philipp Adelt, Natalia Esau, Christian Hölscher, Bernd Kleinjohann, Lisa Kleinjohann, Martin Krüger and Detmar Zimmer (2011). Hybrid Planning for Self-Optimization in Railbound Mechatronic Systems, Intelligent Mechatronics, Prof. Ganesh Naik (Ed.), ISBN: 978-953-307-300-2, InTech, Available from: <http://www.intechopen.com/books/intelligent-mechatronics/hybrid-planning-for-self-optimization-in-railbound-mechatronic-systems>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen