

# Representing the Unknown – Impact of Uncertainty on the Interaction between Decision Making and Trajectory Generation

Marcus Nolte, Susanne Ernst, Jan Richelmann and Markus Maurer

Institute of Control Engineering  
Technische Universität Braunschweig  
Braunschweig, Germany

Email: {nolte, ernst, richelmann, maurer}@ifr.ing.tu-bs.de

**Abstract**—Even though motion planning for automated vehicles has been extensively discussed for more than two decades, it is still a highly active field of research with a variety of different approaches having been published in the recent years. When considering the market introduction of SAE Level 3+ vehicles, the topic of motion planning will most likely be subject to even more detailed discussions between safety and user acceptance. This paper shall discuss parameters of the motion planning problem and requirements to an environment model. The focus is put on the representation of different types of uncertainty at the example of sensor occlusion, arguing the importance of a well-defined interface between decision making and trajectory generation.

## I. INTRODUCTION

Motion planning for automated vehicles is a topic which is currently receiving much attention in the scientific community. A variety of frameworks has been presented in the past. Those approaches range from discretized maneuver- or motion-primitive-based (e.g. [1], [2]) algorithms, to optimization-based approaches, such as optimal (receding horizon) control, often based on Model Predictive Control (MPC) [3], [4]. Efforts are also spent on guaranteeing safety of motion planners in uncertain environments [5], [6]. Regarding the capabilities of the presented approaches, a majority of these planners is tailored to a specific application domain, such as parking or highway scenarios, while only some planners consider inner city scenarios.

One reason for these application specific frameworks is that motion planning is an optimization problem (cf. [7]). However, the complete parameterization of this problem with respect to its cost-function and constraints is at least difficult, if not impossible without making simplifications. This leads to a variety of assumptions being made to simplify the optimization problem or to decompose it into solvable sub-problems which can be treated at different levels of the automation system. Different assumption here lead to different motion planners. The particular challenge for the parametrization of the optimization problem is the presence of uncertainty in the environment model [7] (e.g. measurement uncertainty and sensor occlusion), as well as in the motion dynamic objects (cf. Section IV for a detailed classification).

A central question with regard to the market introduction of automated vehicles and their user-acceptance must deal with how a motion planner treats this uncertainty. A main challenge in this context will be the implementation of motion planners which are safe and not annoying for the vehicle's passengers at the same time. While it is questionable that such a motion planner is even implementable, it becomes important to discuss the parameters of motion planning as well as the risks human drivers take and accept when entering public traffic. An exemplary situation comprises driving at velocities which make it impossible to brake in front of movable objects emerging from occlusions. While taking such risks when driving manually, people expect automated vehicles to be safe and to outperform human driving performance [8]. Thus a very conservative (and thus probably safer) motion planner which calculates trajectories at slow speeds and stops the vehicle often will likely receive less acceptance than a planner which plans trajectories at higher velocities, risking collisions.

This is a twofold problem: In order to resolve those ambiguities in public perception, technical solutions are only partly sufficient. In addition, expectation management for automated vehicle systems must be addressed in public discussions. On the other hand, when automated vehicles are involved in (possibly deadly) accidents, the vehicle's driving strategy and behavior must be explained to the public and the authorities. Those discussions will be difficult to lead at a mathematical level, such that semantic explanations of the executed decisions will be required, which can only be given at a tactical level of the system architecture.

In this paper we discuss the challenges for motion planning arising from an uncertain and incomplete environment representation, extending the argumentation of [7] and [9]. Possibilities to exploit an explicit representation of uncertainty are presented at different levels of a system architecture at the example of sensor occlusion. In this context, we highlight the importance of a clear separation between decision making and trajectory generation in order to explain the generated behavior of an automated vehicle.

For this purpose, Section II discusses related work with

respect to the formulation of the motion planning problem. Section III presents the challenges when dealing with uncertainty and specifically sensor occlusion in more detail. Based on this, Section IV discusses these challenges from a system's perspective at different architectural levels, before Section V presents a possibility to incorporate the architectural considerations in an MPC framework. Section VI concludes the paper with a summary and future work.

## II. RELATED WORK

Benenson *et al.* formally define motion planning as a constraint optimization problem, while having a strong focus on motion safety [7]. In this context, formal safety criteria are given. Additionally, they give a definition of an *incomplete* and *uncertain world model*, which is the consequence of occlusions, a limited sensor range and measurement uncertainties, respectively. For a *conservative world model*, probability distributions representing the uncertain world model are thresholded. The authors take the approach of formulating a cost term for all safety-related assumptions.

Thornton *et al.* present a motion planning framework based on Model Predictive Control [10]. In contrast to [7], constraints are explicitly formulated as hard and soft constraints. This comprises the need for collision-free trajectories (hard constraints) as well as ethical considerations and traffic rules (soft constraints).

This argumentation is picked up in [5]. Based on the conservative world model, a notion of provably safe motion planning for robots with limited sensor range is developed. The paper shows how the planning space in which safety guarantees can be given shrinks in a dynamic environment when calculating the reachable sets of moving obstacles with conservative models.

A technique to deal with this issue in the context of automated driving is to apply maneuver-based motion primitives and assume that other vehicles adhere the given traffic rules (e.g. as proposed in [6], [7]). However, it can be argued that an automated vehicle must also be able to react to non rule-conforming behavior of other vehicles, which then requires the fully conservative prediction strategy [9].

Shalev-Shwartz *et al.* propose a formal framework for formulating common sense and responsibilities in case of an accident [9]. For this purpose, they formulate safety metrics in terms of distances and time-to-collision for longitudinal and lateral maneuvers. While the core of their argumentation for formal verification is not reflecting realistic operating conditions, such as assuming that all sensors work correctly while trying to formalize safety and comfort aspects, they contribute some interesting metrics for increasing behavioral safety. In addition to those metrics, the authors also deal with the problem of occluded space. For this, they derive a maximal velocity at which occluded areas may be approached, considering that pedestrians suddenly emerge from an occlusion.

Occlusions are also explicitly addressed in [11] in the context of path planning for an indoor robot. Occluded areas are modeled in an occupancy grid. A safety distance is calculated,

assuming a maximum velocity for hidden moving obstacles. The resulting area is avoided by introducing a cost term in a cost function for shortest-path-navigation.

Andersen *et al.* also highlight the importance of modeling occlusions, while presenting a Model Predictive Control framework minimizing the occluded area by performing lane changes [3].

While all of the previous approaches have contributed to the motion planning problem in some ways, the practical implication of the formulated issues from a system's perspective have rarely been discussed, if not completely neglected.

## III. CHALLENGES IN MOTION PLANNING

As discussed in [7], from a mathematical point of view, motion planning is an optimization problem. Solving this optimization problem yields a nominal trajectory for the automated vehicle. By controlling the vehicle to this desired trajectory, the system generates *external behavior* [12]. *Unsafe external behavior*, or *unsafe motion* can cause harm (as defined in [7]). For the description of safe motion, it is thus sufficient to only consider the system's trajectory  $\pi(t)$  as a series of system states  $\mathbf{x}(t)$ .

For a more general formulation of the motion planning problem as an optimal control problem, we consider a non-linear control system with inputs  $\mathbf{u}(t)$  and outputs  $\mathbf{y}(t)$  given by

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \quad (1)$$

$$\mathbf{y}(t) = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)) \quad (2)$$

In the tuple notation of [7], the vehicle's trajectory is then formulated as

$$\pi(t) = \{(\mathbf{g}(\mathbf{x}(t_i), \mathbf{u}(t_i)), t_i) \mid i \in \mathbb{N}_0\}. \quad (3)$$

Using the cost function  $c(\pi)$  and extending the notation of [7], we write the motion planning problem in a general form with time-varying hard equality and inequality or box constraints  $\mathbf{k}_{\text{eq}}$  and  $\mathbf{k}_{\text{ineq}}$

$$\begin{aligned} \min_{\mathbf{u}(t), \mathbf{x}(t)} \quad & c(\pi) \\ \text{s.t.} \quad & \mathbf{k}_{\text{eq}}(t) = \mathbf{k}_{\text{constr}}(t), \\ & \mathbf{k}_{\text{ineq}}^{\min}(t) \leq \mathbf{k}_{\text{ineq}}(t) \leq \mathbf{k}_{\text{ineq}}^{\max}(t). \end{aligned} \quad (4)$$

This formulation is extended by slack variables  $\varepsilon(t)$  for including soft constraints, as e.g. proposed in [10]. In this case the cost function is modified, weighting the slack variables for each soft constraint with a factor  $s$ , to

$$c'(\pi) = c(\pi) + c_{\text{slack}}(\varepsilon(t), \mathbf{s}), \quad (5)$$

while adding constraints of the form

$$\begin{aligned} k_{\text{ineq},i}^{\min}(t) - \varepsilon_i(t) &\leq k_{\text{ineq},i} \leq k_{\text{ineq},i}^{\max}(t) + \varepsilon_i(t), \\ \varepsilon_i &\geq 0 \end{aligned} \quad (6)$$

for the  $i$ -th added soft constraint.

The cost function can include a variety of requirements such as weights for passenger comfort (e.g. by weighting jerk),

mission goals (e.g. shortest path to a target pose) and safety requirements (distance from other vehicles). In cooperative systems, the cost function for the ego vehicle is also influenced by the goals and requirements of other traffic participants.

The challenge for the vehicle automation system now lies in a suitable parametrization of the constraints as well as of the weights in the cost function. While this is obvious for motion planners which are based on continuous optimization methods, all other types of planners solve a simplified formulation of this problem, e.g. by using a-priori knowledge, such as a discretized state space (state-lattice and related approaches).

Regardless of the actual implementation, all motion planning algorithms must deal with uncertainty [7]. Considering the explainability of vehicle behavior, it is beneficial to differentiate two different types of uncertainty:

*Epistemic* uncertainty [13] describes missing information, which would be required to build a perfect world- and self-model. For the world model, this uncertainty is based on non-observable parts of the environment caused by limited sensor range or occlusions, as well as missing knowledge about future actions of other traffic participants. It is referred to as the *incomplete world model* in [7]. Epistemic uncertainty also applies to the ego vehicle, e.g. in the form of unobservable system states or due to incomplete models arising from simplifications or errors when designing the vehicle model. When planning motion, it is thus always important to ensure that a generated plan is executable by the automated vehicle.

*Aleatoric* uncertainty, in contrast, is of statistical origin [13]. It thus e.g. includes measurement and process noise when modeling objects or the ego vehicle. Regarding the environment model, it is referred to as the *uncertain world model* in [7].

Epistemic uncertainty is often modeled in an aleatoric manner by introducing probability densities for unknown aspects of the environment (such as existence probabilities for perceived objects). A clear separation between these types of uncertainty is thus debatable in some cases [13]. However, both categories can help making modeling decisions more explicit and thus making system behavior more explainable, as will be discussed below.

Both types of uncertainty make the parameterization of the motion planning problem difficult. Epistemic uncertainty is the reason for an inherently incomplete configuration of constraints, as not all possibly relevant objects are included in the world model. At the same time aleatoric uncertainty allows only conservative assumptions about the evolution of the perceived environment. In this context, a *conservative world model* is proposed in [7] by thresholding probability distributions for the environment. However, the application of conservative predictions only enables safety guarantees for minimal parts of the planning space, as can be seen in [5].

This already becomes obvious, when considering the following simple inner city scenario, as shown in Figure 1: Given a two lane road with oncoming traffic, a van is parked at the right side of the road, occluding parts of the sidewalk. A conservative system must account for objects such as

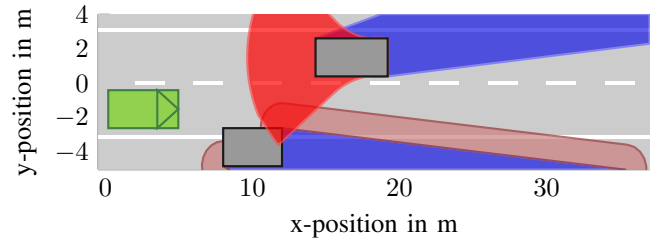


Fig. 1. Scenario illustrating the consequences of epistemic and aleatoric uncertainty in the world model. The ego vehicle is depicted in green. Occlusions are marked blue, possible space occupied by the oncoming vehicle under aleatoric motion uncertainty in red, and possible occupation of unknown objects, such as pedestrians in light red. The resulting occupancies result from explicitly modeling epistemic relations in an aleatoric way. Prediction horizon: 1 s.

pedestrians emerging from the occluded area in any direction at a wide range of possible velocities.

The occluded areas introduces epistemic uncertainty to the scene. When introducing the occupancy of possible emerging objects, epistemic relations are modeled in an aleatoric fashion by making assumptions about possible accelerations and velocities of objects. This in turn can then help when arguing, e.g. why an automated vehicle decelerates or performs a lane change for no immediately apparent reason.

Figure 1 depicts the scenario with the predicted occupancies after one second. A worst-case maximal acceleration  $a_{p,wc}$  of  $0.3 \text{ m/s}^2$  and an initial velocity of  $v_{p,0}$  of  $1.3 \text{ m/s}$  are assumed for potential objects in the occupied area. For the vehicle we assume a maximal acceleration of  $0.1 \text{ m/s}^2$  and an initial velocity of  $4.7 \text{ m/s}$ , which is already a very conservative motion model. The occupied area ends 5 m in front of the ego vehicle. For this illustration, no (traffic)-rule-based constraints are imposed on the vehicle's motion. If we additionally assume a maximal deceleration of  $9 \text{ m/s}^2$  for the ego vehicle, given dry asphalt as the road surface, at  $10 \text{ m/s}$ , a safety guarantee cannot be given for the planned trajectory, as the stopping distance for the ego vehicle is  $5.56 \text{ m}$ . When requiring safety guarantees for a trajectory, the optimization problem is over-constrained in this scenario with no feasible solution available. Even when reducing the maximal acceleration of the oncoming vehicle or increasing the possible deceleration for the ego vehicle, the only safe action in this scenario is stopping.

This example shows that even a very simple scenario, which is encountered regularly in public traffic, causes a dilemma for the development of motion planning algorithms between safety and mobility. Even for conservative planning, safety cannot be fully guaranteed in every possible scenario. As motion planning is an iterative problem, one could argue that a conservative approach will already react before coming into a situation where safety cannot be guaranteed. However, such approaches will implement systems which drive slowly and stop often, which will impede user acceptance. In addition, absolute safety can never be proven, as unexpected situations e.g. loss of cargo, trees falling onto the road (described as the *falling crane scenario* in [7]) can always occur. Approaches, which apply traffic rules (e.g. as argued by [7],[14]) for the

prediction of other traffic participants can only give weakened safety guarantees, but allow for higher velocities of the ego vehicle.

Vehicle-to-x communication could be seen as a means to limit aleatoric uncertainty (e.g. by including additional measurements in the world model) and epistemic uncertainty (by exchanging planned trajectories with other traffic participants). However, one must keep in mind that information received from other entities can always be flawed and that high market penetration is required to fully exploit the benefits of V2X. Thus it must never serve as the single input for safety-critical decisions and is thus also unsuitable when safety guarantees are required.

Looking at human drivers and the traffic system as a whole, it becomes clear, that human drivers always perform a personal trade-off between mobility and a need for safety. If human drivers showed a behavior as described above for a conservative planning approach, inner cities would be blocked by stopped vehicles. Because of this fact, basic regulations such as the "principle of trust" or the requirement of "driving defensively" [15] can be found in almost all international traffic laws in some form. This relieves human drivers from expecting unreasonable behavior of other traffic participants and is thus a major contribution to a functioning traffic system.

While it can thus be argued that driving a vehicle seems to be an accepted risk, two key issues arise from these facts: On the one hand, the question is what personal risk passengers are willing to accept when riding in an automated vehicle and if a principle of trust is applicable to a computer system. On the other hand, it must be determined how a trade-off between a conservative and a more human-like motion planner can look like and which system-wide requirements arise for environment perception, scene representation and decision making in order to parameterize the motion planning problem accordingly.

One feasible approach can be the differentiation between a conservatively planned fall-back trajectory with (weakened) safety guarantees (cf. [6]) in addition to a less conservatively planned trajectory.

Even with trajectory alternatives, a key requirement for planning both trajectories is the explicit representation of uncertainty. While it is common to represent aleatoric uncertainty, e.g. as measurement noise in inverse sensor models for occupancy grids or model-based filtering algorithms for vehicle tracking, approaches which explicitly represent epistemic uncertainty (e.g. [11], [16], [3]) are not as frequently applied in the field of automated driving. For this paper, we assume a grid-based environment representation as presented in [16], combining multiple grid layers such that drivable but occluded spaces can be identified as well as lane markings, curbs and higher elevated objects in the surroundings. The fine grained semantic differentiation between elements in the grid representation is already an approach to represent and decrease epistemic uncertainty.

The following section will give an overview, how the suggested environment representation can be used for decision

making and trajectory generation from an architectural point of view. It will also argue for the importance of well-defined interfaces between decision making and trajectory generation.

#### IV. ARCHITECTURAL CONSIDERATIONS

As described in Section III, motion planning is an optimization problem under the influence of uncertainty. Design decisions about handling uncertainty in decision making and motion planning should be traceable. Thus roles of modules should be explicitly defined in a functional system architecture, as the semantics of available information vary between upper and lower system layers (cf. [17]). However, this separation often remains unconsidered in recent publications (e.g. [3], [10]).

Figure 2 shows logical components of the tactical and the stabilization level. This follows the argumentation of Matthaei and Maurer [18] and Donges [19] to organize the system according to how a human performs the driving task. A detailed functional description of the tactical and the stabilization level can be found in [17].

The guidance block at the tactical level receives information about the current scene, which contains the environment model (including semantic relations) and information about the vehicle's current abilities [20], while the stabilization level only receives feature-level information. This means that information, e.g. about lane markings, the relative position of relevant static and dynamic obstacles, and the relevant traffic rules for a given situation are only fully available at the tactical level. In the following, we use an exemplary scenario in order to argue why this differentiation is required for traceable decision making. We also highlight which consequences arise for the executed maneuvers and thus the external behavior of an automated vehicle.

For the examples at hand, we assume German traffic rules to apply. Thus e.g. solid white lines may not be crossed without good reason and slower vehicles may not be passed on the right hand side.

Consider Figure 3 as an example with the ego vehicle approaching a slower vehicle on a three lane road. In this example, the given situation can be resolved in two different ways: One option would be to stay behind the slower vehicle and perform a *following* maneuver. A different choice could be to perform a *lane change* maneuver in order to overtake.

Both maneuvers can be achieved by at least two different strategies. The system could choose a target pose at the end of the system's sensor range, assuming here that e.g. a sensor (e.g. a roof mounted LiDAR scanner) can perceive the space in front of the slower vehicle. However, without careful parametrization of the algorithm, the executed maneuver becomes a result of the trajectory planner's cost-function, depending on which maneuver causes fewer optimization cost. In the worst case, even the side on which the slower vehicle is passed, would remain undefined. Considering differing traffic regulations in different countries, this is problematic, as passing on the right is e.g. prohibited in Germany, while being allowed in the U.S.

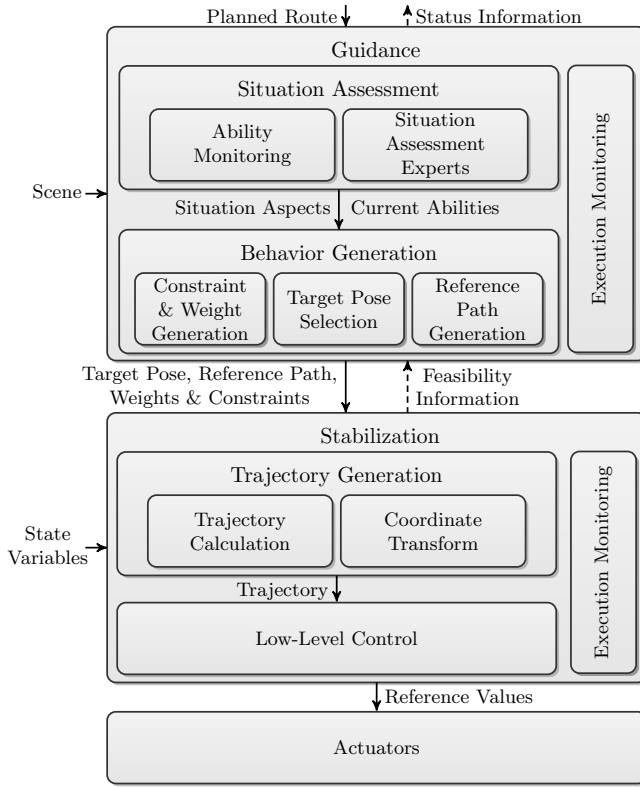


Fig. 2. View on the tactical (guidance block) and stabilization level in a functional architecture with blocks contributing to decision making and motion planning (cf. [17]).

On the other hand, the system could explicitly choose a target pose behind the leading vehicle. This choice represents an explicit decision for *following* the lead vehicle. For performing an overtaking maneuver under European traffic rules, the target pose is placed in the left lane in front of the leading vehicle. As soon as the left lane is reached, the target pose can be returned to the right lane, such that the ego vehicle can merge back.

The latter way of executing the maneuver leaves the tactical layer direct control of the system's state. When considering the first option, the choice of maneuvers is left to the stabilization level, which makes the system's behavior difficult to trace.

This differentiation becomes even more important if the decisions to make implies violating traffic rules. Consider a lane bounded by a solid lane center line, which is blocked by a parked vehicle (cf. [10]). A trade-off must be made between crossing that line and stopping. While in this specific example, legislations allow crossing solid lines by necessity, e.g. if a road would be impassible otherwise, the decision to violate a traffic rule must be an explicit decision, not a mere output of an optimization algorithm. In turn, when resolving such situations only at the stabilization level, a non-carefully parametrized trajectory planner could plan a path which violates traffic rules without the tactical system modules being aware of this fact. As a result, the system's tactical layer should have full control of the trajectory planner's cost-function and the relevant constraints, which derive from the chosen target pose

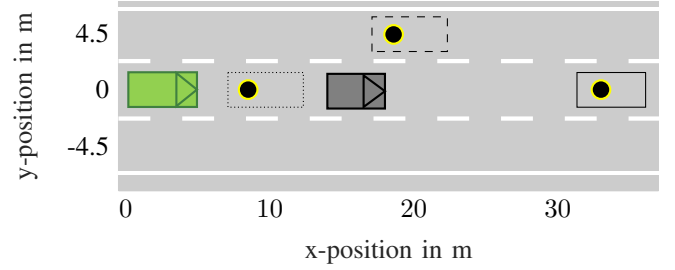


Fig. 3. Illustration of different target poses, which result in different maneuvers. The dotted target pose results in following the slower vehicle. The dashed target pose implies a lane change maneuver, as part of an overtaking maneuver. Skipping the dashed target pose results in an evasion maneuver.

and the semantics of the perceived environment.

If uncertainties are to be respected when making decisions, the described separation between tactical decisions and the bare execution of driving maneuvers becomes even more important. In this case, frequent decisions must be made by weighting a conservative driving style against a more risk prone style. Regrading the trajectory planner's cost function, the tactical modules must then be able to modify the cost terms to enable balancing between comfort and safety parameters (e.g. weighting of yaw-rates and side slip angle). The control of hard constraints explicitly models rules which must be adhered to, such as collision avoidance requirements. Soft constraints can be used to model rules which can be violated under circumstances by prioritizing constraints, e.g. when breaking a rule by necessity.

By explicitly parameterizing the optimization problem from the tactical level, the system can make active decisions which then reflect in the optimization problem. If the system then decides to place the target pose at the end of the sensor range, objects in the ego lane can, of course, still be avoided. However, the semantics of the executed maneuver changes: The result becomes an evasion rather than an overtaking maneuver. While this example is arguably a minor distinction, making explicit decisions results in increasingly traceable system behavior. Being able to trace and argue the resulting system behavior on both, a technical and a semantic level in case of errors can contribute to public discussions, when it becomes necessary to argue why an automated vehicle system behaved in a particular way.

## V. SIMULATIVE EVALUATION OF AN EXEMPLARY MPC PLANNER

Section III and IV described the basic relation between decision making under uncertainty and architectural decisions for trajectory generation. To further illustrate this example, we present an exemplary implementation of an MPC-based optimization framework, based on our previous implementation [17]. The approach is extended in order to support the inclusion of uncertainty emerging from occluded parts of the environment as well as the definition of soft constraints to support different driving strategies. In contrast to [17], the planner is implemented in the ACADO Optimization Framework [21], using the QPOASES solver [22] and the

provided MATLAB/Simulink interface. A linear discrete time single track model similar to [17] is used with rear steering fixed at 0 rad.

The model is extended by a slack term  $\varepsilon$ , modeled as an additional system input with linear constraints

$$\mathbf{u}_{\text{soft-constraint}}(k) = \begin{bmatrix} u_{\text{vehicle}} \\ \varepsilon \end{bmatrix} (k), \quad \varepsilon \geq 0. \quad (7)$$

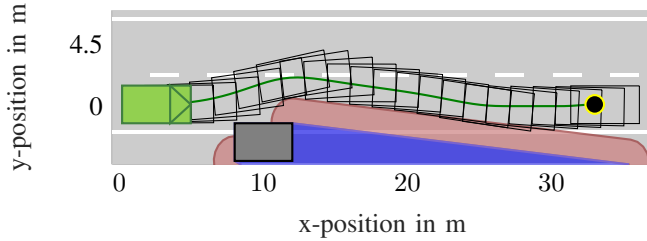
In this way, the slack term can be weighted in the cost-function and can be subject to additional constraints itself. Constraining the slack term permits controlling how far soft constraints may be violated.

The relevant constraints are put on the lateral deviation  $e(k)$  from a reference path, resulting in lower and upper soft constraints for the lateral path deviation:

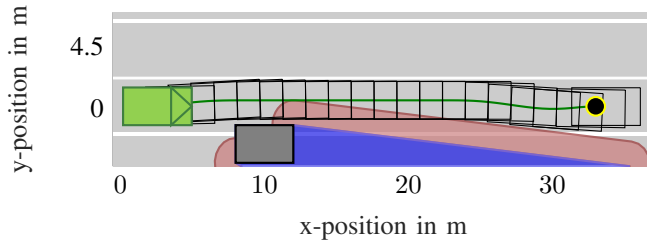
$$e(k) \geq e_{\text{slack,min}}(k) - \varepsilon, \quad (8)$$

$$e(k) \leq e_{\text{slack,max}}(k) + \varepsilon. \quad (9)$$

The application of these slack constraints shall be illustrated in the following example: A static object is parked at the edge of the ego lane, creating an occluded area which extends over parts of the ego lane. Solid lines are modeled as hard constraints, the dashed center line in Figure 4a is not assumed to be relevant in this case, as no object on the opposite lane is detected.



(a) Exemplary parameterization with no constraint on the lane center line and soft constraints on the areas adjacent to occluded space. Vehicle avoids area where objects can emerge from occlusion. Consideration of uncertainty is prioritized compared to staying inside the ego lane.



(b) Lane center line modeled as hard constraint, vehicle stays at a leftmost position inside the ego lane. Lane keeping is prioritized over a safety distance to the occluded areas.

Fig. 4. Figures a) and b) show a comparison of a different set of constraints with similar optimization parameters. Green: ego vehicle, red: possible occupancy generated by an object emerging from the occluded space in blue.

Possible objects emerging from the occluded area (blue) are modeled by an occupancy (red) according to the assumptions made in Section III. The possible occupancy created by those assumptions is modeled as soft constraints on the lower bound

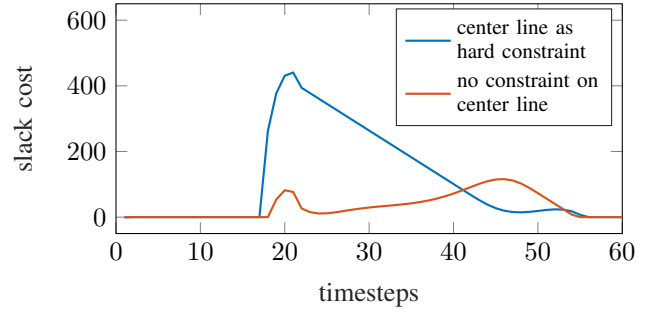


Fig. 5. Illustration of cost caused by the slack term

of the lateral deviation. In the target pose (depicted as a black dot), orientation (0 rad) and x-position (0 m) are imposed as hard constraints.

The semantics in the given scenario differ from the overtaking example in Section IV: As the obstacle is only blocking parts of the ego lane, the situation can be resolved by performing a passing maneuver which does not require a lane change. This passing maneuver can be executed inside the ego lane, if required. Figure 4a and 4b show two configurations of the MPC-Problem, including a dashed and a solid lane center line.

The explicit decisions modeled in both situations are the following: Safety margins are prioritized over staying in the ego lane by not modeling the dashed center line as a constraint in Figure 4a, as dashed centerlines may be passed and no oncoming traffic is visible. Uncertainty about emerging objects is taken into account by avoiding the modeled occupancies.

By imposing a hard upper constraint on the lateral path deviation at the position of the solid lane center line in Figure 4b, traffic rules gain explicit precedence over accounting for uncertainty. The vehicle is still able to perform an evasion maneuver, while staying in its lane and keeping to the left most lateral position, without violating the hard constraint. The latter strategy could be chosen e.g. if available semantic information (time of day, location, etc.) makes emerging pedestrians unlikely.

The slack value in the example is chosen to be greater than the highest weight on inputs and states in the cost function by a factor 3. The resulting slack values are depicted in Figure 5. It is visible how the second scenario incurs higher cost for the slack variable. It is also visible how the vehicle still violates the soft constraints slightly in the first scenario. This is due to two facts: On the one hand, a factor 3 compared to the weights e.g. on the yaw rate means that the slack term does not dominate the cost function entirely, which is desired. On the other hand, the optimization algorithm must still respect the yaw angle constraint in the target pose, which causes the slight curve near the end of the green trajectory.

The MPC framework (code generated from Simulink and compiled with MSVC 10.0) for trajectory planning runs with an average execution time of ca. 95 ms on an Intel i7-4800MQ CPU (2.7 GHz).



## VI. CONCLUSION & FUTURE WORK

In this paper we have discussed the effects of different types of uncertainty on system design for the interaction between decision making and motion planning. We discussed preceding work, discussing motion planning as an optimization problem while interpreting the role of decision making modules for the parametrization of the optimization algorithms. In this context we argue for a clear separation between explicit tactic decisions and implicit decisions made at the stabilization level in order to perform a trade-off between conservative and optimistic (or more risk-prone) system implementations.

We also presented an MPC framework which provides the possibility to reflect such explicit decisions in the selection of hard and soft constraints. In the given example, we modeled epistemic uncertainty caused by occluded space in an aleatoric distribution and show how formulating soft constraints can be used to differentiate between a conservative and optimistic driving styles. For future work, the proposed system is currently being integrated in our research vehicles *MOBILE* [23] and *Leonie* and will be validated using the environment perception framework described in [16].

## ACKNOWLEDGEMENT

This work has partially been funded by the Daimler and Benz Foundation. The authors would like to thank their colleagues Gerrit Bagschik, Jens Rieken and Cordula Reisch for valuable discussions.

## REFERENCES

- [1] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a Frenet Frame," in *2010 IEEE Intern. Conference on Robotics and Automation (ICRA)*, 2010, pp. 987–993.
- [2] F. von Hundelshausen, M. Himmelsbach, F. Hecker, A. Mueller, and H.-J. Wuensche, "Driving with tentacles: Integral structures for sensing and motion," *Journal of Field Robotics*, vol. 25, no. 9, pp. 640–673, 2008.
- [3] H. Andersen, W. Schwarting, F. Naser, Y. H. Eng, M. H. J. Ang, D. Rus, and J. Alonso-Mora, "Trajectory Optimization for Autonomous Overtaking with Visibility Maximization," in *2017 IEEE 20th Intern. Conference on Intelligent Transportation Systems (ITSC)*, Yokohama, Japan, 2017, pp. 1906–1913.
- [4] B. Gütjahr, L. Gröll, and M. Werling, "Lateral Vehicle Trajectory Optimization Using Constrained Linear Time-Varying MPC," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, no. 99, pp. 1–10, 2016.
- [5] S. Bouraine, T. Fraichard, and H. Salhi, "Provably safe navigation for mobile robots with limited field-of-views in dynamic environments," *Autonomous Robots*, vol. 32, no. 3, pp. 267–283, 2012.
- [6] S. Magdici and M. Althoff, "Fail-safe motion planning of autonomous vehicles," in *2016 IEEE 19th Intern. Conference on Intelligent Transportation Systems (ITSC)*, 2016, pp. 452–458.
- [7] R. Benenson, T. Fraichard, and M. Parent, "Achievable safety of driverless ground vehicles," in *2008 IEEE 10th Conf. on Robotics and Vision 2008 10th International Conference on Control, Automation, Robotics and Vision (ICRAV)*, 2008, pp. 515–521.
- [8] B. Schoettle and M. Sivak, "A survey of public opinion about autonomous and self-driving vehicles in the US, the UK, and Australia," 2014.
- [9] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "On a Formal Model of Safe and Scalable Self-driving Cars," 2017. arXiv: 1708.06374 [cs, stat].
- [10] S. M. Thornton, S. Pan, S. M. Erlien, and J. C. Gerdes, "Incorporating Ethical Considerations Into Automated Vehicle Control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 6, pp. 1429–1439, 2017.
- [11] W. Chung, S. Kim, M. Choi, J. Choi, H. Kim, C. b Moon, and J. B. Song, "Safe Navigation of a Mobile Robot Considering Visibility of Environment," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, pp. 3941–3950, 2009.
- [12] M. Nolte, G. Bagschik, I. Jatzkowski, T. Stolte, A. Reschka, and M. Maurer, "Towards a Skill- And Ability-Based Development Process for Self-Aware Automated Road Vehicles," in *2017 IEEE 20th Intern. Conference on Intelligent Transportation Systems (ITSC)*, 2017, pp. 739–744.
- [13] A. D. Kiureghian and O. Ditlevsen, "Aleatory or epistemic? Does it matter?" *Structural Safety*, vol. 31, no. 2, pp. 105–112, 2009.
- [14] M. Althoff and S. Magdici, "Set-Based Prediction of Traffic Participants on Arbitrary Road Networks," *IEEE Transactions on Intelligent Vehicles (ITSC)*, vol. 1, no. 2, pp. 187–202, 2016.
- [15] T. M. Gasser, "Fundamental and Special Legal Questions for Autonomous Vehicles," in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 523–551.
- [16] J. Rieken, R. Matthaei, and M. Maurer, "Toward Perception-Driven Urban Environment Modeling for Automated Road Vehicles," in *2015 IEEE 18th Intern. Conference on Intelligent Transportation Systems (ITSC)*, 2015, pp. 731–738.
- [17] M. Nolte, M. Rose, T. Stolte, and M. Maurer, "Model Predictive Control Based Trajectory Generation for Autonomous Vehicles – An Architectural Approach," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, Los Angeles, USA, 2017, pp. 798–805.
- [18] R. Matthaei and M. Maurer, "Autonomous Driving - A Top-Down-Approach," *at - Automatisierungstechnik*, vol. 63, no. 3, 2015.
- [19] E. Donges, "A Conceptual Framework for Active Safety in Road Traffic," *Vehicle System Dynamics*, vol. 32, pp. 113–128, 2-3 1999.
- [20] S. Ulbrich, A. Reschka, T. Menzel, F. Schuldt, and M. Maurer, "Defining and Substantiating the Terms Scene, Situation and Scenario for Automated Driving," in *2015 IEEE 18th Intern. Conference on Intelligent Transportation Systems (ITSC)*, 2015, pp. 982–988.
- [21] B. Houska, H. J. Ferreau, and M. Diehl, "ACADO toolkit-An open-source framework for automatic control and dynamic optimization," *Optimal Control Applications and Methods*, vol. 32, no. 3, pp. 298–312, 2011.
- [22] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, "qpOASES: A parametric active-set algorithm for quadratic programming," *Mathematical Programming Computation*, vol. 6, no. 4, pp. 327–363, 2014.
- [23] P. Bergmiller, "Towards Functional Safety in Drive-by-Wire Vehicles," PhD Thesis, Technische Universität Braunschweig, 2014.