Maneuver planning for highly automated vehicles

Cristina Menéndez-Romero[1], Franz Winkler[1], Christian Dornhege[2] and Wolfram Burgard[2]

***Abstract*— One important aspect of autonomous driving lies in the selection of maneuver sequences. Here the challenge is to optimize the driving comfort and travel-duration, while always keeping within the safety limits. Human drivers analyze and try to anticipate the traffic situation choosing their actions not only based on current information but also based on experience. The decision making process can be treated as a planning problem. Classical planning systems consider the autonomous driving task as a global numeric optimization problem, which in populated dynamic environments can become computationally intractable. In addition, purely numeric computations hamper the understanding of the decision making for the human user. We propose a planning system that presents a multi-level architecture, similar to the human reasoning process, which combines continuous planning with semantic information. This allows the planning system to deal with the complexity of the problem in a computationally efficient way and also provides an intuitive interface to communicate the decisions to the driver. We validate our approach in simulation and through a set of experiments carried out with a real vehicle and an integrated traffic simulation also known as vehicle in the loop (VIL).**

I. INTRODUCTION

Since the DARPA Grand Challenge, the interest in automated driving systems has increased not only within the research community but also by the general public. Many automotive manufacturers have brought driver assistance systems with different automation degrees to market, including the lane departure warning, the adaptive cruise control (ACC) or the lane change assistant. Compared to assistance systems, where the last decision and the responsibility still fall back to the driver, in a highly automated driving vehicle, the driver does not need be continuously in control. The great challenge of autonomous systems is to guarantee the safety of the selected maneuvers during the autonomous driving phase. The ability to react to unexpected situations should be ensured under defined constraints. In addition, if the system wants the human taking over control, safety should be guaranteed while the driver is warned and at least until the driver gets back into control.

This paper presents a framework to safely accomplish the driving task. The approach is described using the example of a highway entrance ramp scenario, is valid also for interchanges and further highway scenarios. In entrance ramp situations (Fig. 1) the ego vehicle has to achieve a mandatory lane change on a limited space and to merge itself into the traffic flow. The challenge is to perform the driving activity based on the partially available knowledge of the situation.

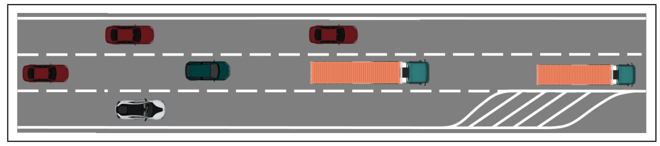

Fig. 1. Entrance Ramp Scenario

Even if the observed data can be complemented by backend information, the sensor range is still limited. Besides, the behavior of the other road members is predictable only partially and for a short time horizons. Therefore, the planning system is forced to deal with uncertainties and partial knowledge.

Classical approaches consider the autonomous driving task as a global optimization problem, resulting in intractable computational effort. Other approaches present a top-down architecture where some critical information is lost within the layers. The trade-off between safety and computational effort results in solutions that provide reactive lane changes, that is they perform conservative lane changes only when the freespace near the vehicle is big enough. This conservative behavior overlooks valid and safe maneuvers that a human driver would actually select.

In this paper we consider the overall problem as a planning problem and present a novel framework based on the integration of semantic and continuous planning. The same planning domain is defined on different abstraction levels for all the planning layers. The approach deals with the partial information in an efficient and traceable way. The planner clusters the different options, assesses them and selects the best policy based on the expected future reward. The integration of different abstraction levels allows to deal with the increasing time horizon as well as with the increasing uncertainties. It does not only take into account the information provided by the environment but also the observed and learned values from past situations. This results in a proactive driving strategy that plans and executes the best policy to reach a desired lane and the desired velocity.

This paper is organized as follows: the next section presents a review of the related work. In Section III-B the domain used by the planning task is formalized and the framework is presented. Section III-A describes the different abstraction levels of the planning task. Simulation and experimental results are presented in Section IV.

II. RELATED WORK

The concept of legal safety as basis for the interaction between human and automated drivers was proposed by Vanholme *et al.* [1]. This concept was expanded to a lane safety

[1]C.Menéndez-Romero and F.Winker are with BMW Group, Munich, Germany cristina.menendez@bmw.de

[2]C.Dornhege and W.Burgard are with Autonomous and Intelligent Systems, University of Freiburg, D-79110 Freiburg, Germany

assessment based on the ICS (Inevitable Collision States) and its stochastic equivalent PDS (Probabilistic Collision States) [2]. One limit of this safety assessment is the infinite branching factor because the number of possible scenarios tends to infinity when the horizon time increases. Althoff *et al.* [3] simplified this problem considering the extreme possible behavior (maximal acceleration and deceleration) for the other road users. Their online formal verification can guarantee safety for all times but the method can lead to extremely conservative behaviors because they only accept a solution for the ego vehicle if no overlap with other road users happens during the whole horizon. Nevertheless, an evaluation of the risk is necessary for the whole maneuver and the safety needs to be guaranteed under defined constraints for the immediate time horizon.

Many proposed architectures rely on a top-down architecture where the lower level optimizes a safe trajectory for a selected longitudinal or lateral goal [4], [5], [6], [7], [8]. The main problem of the top-down architecture is its level dependency. If the high level fails to make its decision, the lower module can only return that no solution was found forcing the high level to propose another solution, consequently loosing critical time in dangerous situations. Therefore it is important to guarantee that the subgoals selected by the higher layers of the architecture are still feasible on the lower layers. A combination of hybrid automata and decision tree is the architecture proposed by Ardelt *et al.* [9], integrating a discrete decision architecture within the continuous data processing. Ardelt *et al.* [10] evaluate the utility of each lane considering measurement uncertainties. Some works concentrate on the cooperation between road users. Schwarting *et al.* [11] present a cooperative decision making algorithm that anticipates and solves predictable conflicts. Based on the game-theory, a combined prediction and planning framework is presented by Bahram *et al.* [12]. All these methods optimize the immediate utility for the lane change or lane keep decision, also considering the cooperation between vehicles but they do not take into consideration its further effects, in other words, this methods work within a short horizon time but do not provide a medium term strategy.

The Planning Domain Definition Language (PDDL) [13], was introduced as an attempt to unify the planning languages. Since then, several planning formalisms have been developed and extended to integrate temporal dependencies [14], [15] or to model continuous domains [16]. Planning formalisms provide the tool to deal with the complexity, improving the knowledge representation and the reasoning process. Dornhege [17] presents a semantic planning where the high and low level work more tightly integrated for a robotic planning task. Zhao *et al.* integrate ontology-based knowledge into the decision making system [18].

In order to get over the computational disadvantages of a down-top approach and the information-loss of a top-down approach and to achieve integration and coherence between different abstraction levels, in this contribution the system combines symbolic and continuous planning. Besides, it handles with the lack of information and the uncertainties derived from the behavior of other traffic participants.

III. APPROACH

The objective of this work is to provide an adequate framework to allow the autonomous vehicle a safe drive maintaining the requested goals and providing robustness over other traffic users behavior for a further horizon time.

When considering the planning horizon, there is a trade-off between the accuracy of the information and the time horizon, that means the further the planning horizon is considered the more imprecise is the prediction of the scene evolution. This characteristic is not a problem for a human driver, because they are used to combine an anticipatory behavior evaluating the current situation and their evolution with a more reactive behavior dependent on the immediate actions, for example a driver wants to change to the left lane to drive faster, selects a gap between two vehicles, drives towards this gap and is able to adapt his velocity if another unexpected vehicle merges in front of him. In a similar way, this paper presents a multilevel architecture, where the different levels interact with each others in different abstractions degrees. A higher level works on the generic lane and velocity selection according to the current traffic flow, an intermediate level provides a longitudinal and lateral strategy given the desired lane and velocity and the lower levels implement on a short horizon the requested sub-goals. The general framework and its domain is presented as well as the intermediate level, responsible for planning and executing the driving activity on medium time horizon, the tactical planning.

A. System Overview

Many classical planning algorithms are based on a discretized model of the continuous solution space, where a guided search like A* or D* generates a plan near to the optimal one. The complexity of the solution increases with the resolution of the discretization leading to the branching factor problem, where the set of different possibilities to be selected tends to infinity.

In order to deal with the branching factor, the proposed planner presents a multilevel architecture where the first levels work with higher abstraction of the domain and longer time horizon and the lower levels operate with more precision for a shorter time horizon. Firstly the vehicle receives information about the environment through the different sensors and back-end communication, this information is processed in the Environmental Model and given to the Maneuver Planner. At the highest level the Maneuver Planner selects the desired lane and maximal velocity (Global Planner). The intermediate level (Tactical Planner) considers the behavior of the surrounding road users to identify and cluster the different possibilities to keep driving on the lane or changing into the neighbor lane, and selects the best policy for the current situation. Then, the Trajectory Planner provides a drivable and collision-free trajectory towards the goal that the trajectory tracking and the vehicle controller follow by

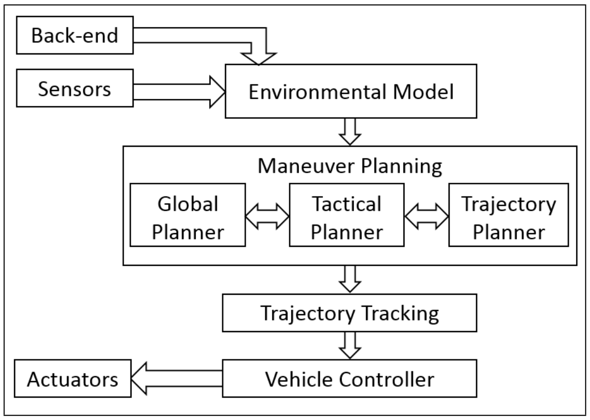

Fig. 2. System Overview

controlling the vehicle's actuators and thus, closing the circle. Fig. 2 shows the process.

The main difference of this system with a top-down architecture is its parallelism. The planning combines an abstract decision making process with numerical mechanisms. This leads to a complete and structured exploration, identification and assessment of the different options during the planning task. The abstraction of maneuver clusters allows the planner to reduce the complexity overriding the infinite branching factor of the planning task. Besides, the abstraction level allows a better traceability of the planner and can be directly used as interface to communicate with the driver. To achieve a complete understanding of the world and the correct interaction within the levels, the problem needs to be defined.

B. Problem and Task Description

The decision making process takes place on the above presented Maneuver Planner. This paper focuses on the lane change and lane keep strategies but the framework is defined for the whole planning process.

According to Dornhege [17], a planning instance is defined to be a pair $I = (Dom, Prob)$ where the domain Dom describes the model of the world and the generic actions and the problem $Prob$ instantiates the current world. The planning domain is a tuple of a finite set of predicate symbols, each one with an associated arity, a finite set of function symbols that represent numerical values and a finite set of schematic operators. A schematic operator is a tuple formed by a precondition, an effect and the function that maps the operator cost. The main operators for the driving task are *keep the lane* $\{\mathrm{KL}\}$ and *change the lane* $\{\mathrm{CL}\}$.

As presented by Kuipers *et al.* [19], the ontology of the topological level can be defined considering the world as a 2D abstraction. A region is a two dimensional subset of the environment that can be defined by one or more boundaries. A path describes part of the environment as a one dimensional subspace and can be directed. A place describes a part of the environment as a zero dimensional subspace. The same element can be defined through different dimensional subspaces depending on the abstraction needed

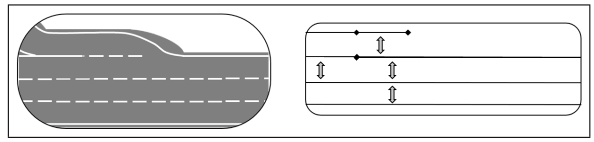

Fig. 3. Exit-ramp represented as 2D abstraction and 1D abstraction

by the planner. For example, a lane is characterized as a region with at least one drivable boundary defined through one reference path, two lateral boundaries and two longitudinal boundaries. A local planner provides the lateral and longitudinal trajectories driving within the defined boundaries and considering the 2D description of the lane (it can be desired to drive towards a centerline and deviations are accepted within the lane boundaries but punished). The higher level planner is concerned with the lane selection over time, therefore, it reasons over the 1D dimensional abstraction of the lanes and the connectivity. Fig. 3 presents an example of a highway. The two-dimensional abstraction of the world does not consider the street profile explicitly. Nevertheless, this information is relevant in those situations where the available power of the vehicle is reduced. For example, a vehicle driving with a trailer could have less power available for an overtake maneuver. Also if the weather conditions change due to rain or snow, the available transmission force to the road will determine the vehicle behavior and has to be considered during the planning step. Through the context specification at the problem definition, the available acceleration can be limited so that the inclination, load and friction are implicitly included.

The problem $Prob = (T, Obs, Ags, C, Goal)$ is described by a 5-tuple consisting of the current topology, the objects and agents interacting on it and the context and goal state specifications. Objects are located on the road. Some of them are not drivable obstacles as for example barrier boards on roadworks, which can reduce the available section for the autonomous vehicle. Objects can also signal a modification of the road attributes (maximal velocity, connectivity between lanes...) or represent a new boundary. Different from objects, agents are proactive, temporary continuous, autonomous and reactive [20]. They are also not drivable, and usually dynamic. As autonomous entities they follow their own goals. Different models like physic-based models, maneuver-based models or interaction-aware based models [21] can be used for their prediction. The prediction is an useful input for the planning algorithm but the algorithm also needs to be able to deal with the lack of information. Therefore an assessment parameter derived from uncertainties is integrated into the selection policy. In Section III-C, the integration of the prediction information within the planning process is explained in more detail.

The world and its entities are fully defined and instantiated with the domain and the problem definition. The driving task takes place on a defined location (based on street topology), where different objects can be placed, some of them changing the attributes of the location. There are different agents moving and iterating on the location: traffic users which follow the road topology and are conscious of the traffic rules and no road users (like animals or pedestrians). The vehicle follows two different goals: maintenance goals (always drive safe) and achievement goals (drive on the selected lane with a selected velocity).

C. Tactical Planning

This work focuses on the medium horizon planning, the description of the Global Planner is not focus of this paper. The main functionality of the Tactical Planner is to identify the different maneuver sequences to reach a selected lane and velocity (which can be a function or an input from the driver), quantify the cost of each of them, select the best policy and forward the strategy to the trajectory planning. This planner level combines a semantic abstraction of the configuration space with a continuous estimation of the scene evolution. The configuration space is abstracted into gap spaces allowing the planner to reason over the cluster of all possible motion behaviors related to each gap. This abstraction provides the planner a simplified interface to reason over time and restrict the search space for the numerical optimizer. So a lane keeping strategy includes the possibilities of continuing on the current lane as keep on driving on current front gap or keep on driving considering the gap generated for a predicted merging vehicle or brake until stillstand on the current lane. A lane change strategy includes all the possibilities of changing whether to the left or right lane into the different defined gaps.

The tactical planner is composed by five different submodules, as shown in Fig. 4. *Gap Generation* processes the agent list to provide a list of current and potential gaps. A gap is defined by a front and a rear limitation. Those restrictions of the available space could be generated by agents or static restrictions as an ending lane. For each relevant lane, the relevant agents are sorted with respect to their longitudinal distance, generating the gap information. Phantom vehicles are included at the sensor range limits. For the vehicles with a lane change prediction, two different gap sets are generated to consider both, the scenario in which it does not change the lane and also the scenario in which it changes the lane. Both scenarios are forwards simulated, the cost are computed and weighted proportionally to the belief of a lane change to be successful. Once the gaps are generated, a feasibility and assessment evaluation is performed. The current lane is evaluated for the available distance in front considering the front vehicle, potential merging vehicles and other obstacles at *Lane Keep assessment*. On a similar way at *Lane Change Assessment* for the neighbor lanes and based on the gap list, the reachable gaps for a lane change are identified and an associated utility cost is calculated. The feasibility and assessment functions are also presented. The block *Policy Selector* has two different tasks, firstly it selects the most suitable maneuver from the available set and gives it to the optimization module *Maneuver Optimization*. And secondly it selects the most adequate maneuver once they are optimized. The *Maneuver Optimization* is based on classic discrete graph search where starting from the current position and velocity, each node is expanded using a set of predefined accelerations. For practical reasons, the maneuvers are restricted to maximum of one acceleration, one deceleration and three constant velocity phases. The block *Maneuver Optimization* could use other optimizer instead the graph search if it is required. To assure a safe behavior, on the first place the most conservative maneuver is selected and optimized and then, depending on the available computing time, further options are explored. Thus, a fall-back policy is considered during each step.

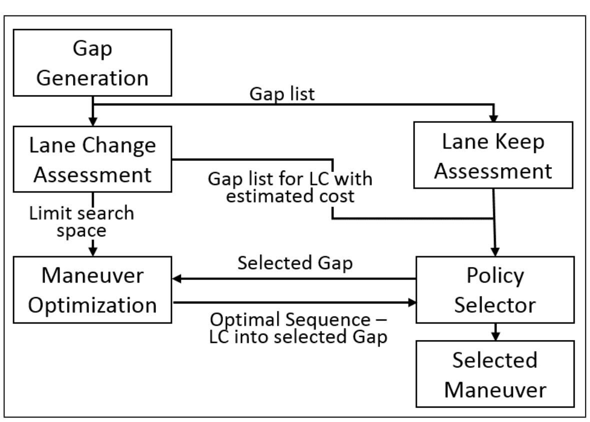

Fig. 4. Tactical Planner Work-flow

D. Feasibility and Assessment

The operators for lane change and lane keep policies are plausibilized (*Lane Change Assessment*, *Lane Keep assessment*) based on a reachability analysis for available gaps.

For the current lane, the front vehicle and potential front merging vehicles are analyzed and the lane keeping policies for the comfort and also for the safety limits are plausibilised and included with their associate cost and success assessment in the policy list. Then the lane change policies are computed for the comfort limits and included in the policy list. The total cost computation is defined as $c_{MF} = w_1 \cdot c_{\Delta v} + w_2 \cdot c_{LCD} + w_3 \cdot c_{gap} + w_4 \cdot c_{SA}$.

This term considers the physical values obtained by the environmental model and the success rate on past situations. The value is a weighted combination of the variations on velocity, the estimated time span to do the lane change, the temporal gap size and the probability of the successful lane change.

- Cost for Velocity Deviation ($c_{\Delta v}$)
  The ego vehicle should try to drive as close to the desired velocity (v_{des}) as possible, respecting the safety distance to the other road users. Equation (1) evaluates the maximal velocity deviation from the desired one during the maneuver.

$$c_{\Delta v} = \frac{max(|\Delta v(t)|)}{v_{des}} \tag{1}$$

- Cost for Estimated Lane Change Duration (c_{eLCD})

Usually the minimal estimated time for the ego lane changes t_{eRG} provides the relevant information to compare between strategies. Lane changes that finished later are penalized because they imply a risk of the situation to change. For spatial restrained situations (as ending lanes or highway nodes) the available distance is converted into available time (t_{max}). For not restrained situations, a generic available time is considered. As shown in (2) the cost is calculated with a quadratic function so that the cost increases faster when the estimated time tends towards the limit of the available time.

$$c_{eLCD} = \begin{cases} \infty, & t_{eRG} > t_{max} \\ (\frac{t_{eRG}}{t_{max}})^2, & t_{eRG} \leq t_{max} \end{cases} \quad (2)$$

- Cost for Temporal Gap Size (c_{gap})
  A minimum criteria for the gap acceptance is defined as gap_{min}. This value represents the minimal intervehicular gap size. Smaller gap sizes get an infinite cost (are not accepted) while bigger values get a normalized cost assessment. The gap size ($gap_{size} = t_{follow} + \Delta t_{adaption}$) is defined as the sum of the required adaption time ($t_{adaption}$) to reach the front vehicle velocity (if it is driving faster) and the resulting intervehicular-time ($t_{follow} = \frac{\Delta s}{v_{adapted}}$) between the front vehicle and the rear vehicle.

$$c_{gap} = \begin{cases} \infty, & gap_{size} < gap_{min} \\ \frac{gap_{min}}{gap_{size}}, & gap_{size} \geq gap_{min} \end{cases} \quad (3)$$

- Cost for Learned Success Assessment (c_{SA})
  The success assessment evaluates the consequences of a scene evolution different from the predicted one. The most likely evolution is evaluated and forwards simulated within the cost function term. The success assessment gives, based on learned values from past situations, an associated cost for the probability that the situation evolves worse than predicted and a selected gap becomes not reachable. So the uncertainties in the behavior of other traffic participants are considered through the cost value c_{SA} calculated as the probability of an unsuccessful lane change given a selected maneuver: $c_{SA} = 1 - p(Success|selectedManeuver)$

For the computation of this cost parameter, a set of simulations with different start configurations was run and a neural network with two layers and 30 nodes was trained. For each simulation, once lane change started, the information of the selected gap and the result of the maneuver (the lane change was successfully completed on the selected gap or not) are saved and given as target value for the network. As input vector the feature vector is defined through the intervehicular time, the intervehicular distance and the time to collision between the ego vehicle and the main vehicles involved in the maneuver. Those vehicles are the front and rear vehicles defining the goal gap of the desired lane and the vehicle in front of the ego vehicle on the current lane.

Once a selected maneuver is optimized, its cost value is updated: $c_{MS} = c_c + c_{oLCD} + c_{gap} + c_{SA}$

- Cost for Comfort (c_c)

$$c_c = \frac{max(\Delta a)}{a_{comf} - dec_{comf}} \quad (4)$$

  being a_{comf} and dec_{comf} the maximal values for acceleration and deceleration on comfort mode.
- Cost for optimized Lane Change Duration (c_{oLCD})
  Similar to (2), using the resulting optimized time t_{oRG} instead of the estimated time to reach gap t_{eRG}

With the updated cost, the best policy is selected and given to the Trajectory Planner to provide a collision-free and jerk minimal trajectory for the short horizon.

The system is able to deal with dynamic environments because the foresight planning structure considers the most likely evolution of the situation but also integrates the past experience values and includes the vehicle dynamic limitations. The next section presents the simulation and experimental results.

IV. Experiments

The system was validated within two different frameworks. The first one was simulative and the second one was carried out on a prototype vehicle and a simulated traffic environment (VIL). The experiments setup consists of an entrance ramp scenario as defined in Fig. 1. The vehicle has to achieve the lane change before the current lane ends to be successful. In case it cannot achieve the lane change, the vehicle has to break into standstill at the end of the lane, then the maneuver is not successful but still safe. For practical reasons, the Global Planner is replaced with an external input of desired velocity and desired lane. The module Trajectory Planner is the defined by Rathgeber *et al.* [7]. The planner has an intern model of the motion of other agents, independent from the planning structure. For those experiments the model was defined as follows:

Agents Motion Model

The velocity of a lead vehicle v_{lead}, is defined as the combination of a constant acceleration (CA) and constant velocity (CV) as shown in (5). The lead vehicle keeps its current acceleration (a_{lead}) during a defined t_a and then drives with a constant velocity. The defined velocity profile will also be considered for the lap vehicle on a first step. From the moment when the ego vehicle makes a lane change

Fig. 5. Test Vehicle with dGPS

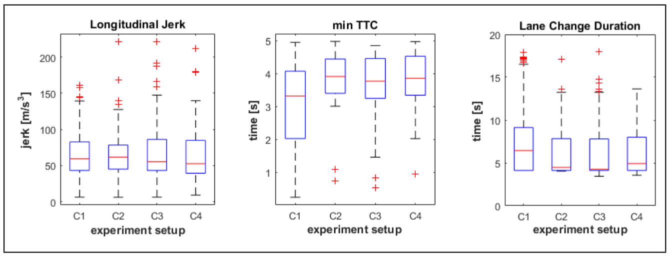

Fig. 6. Longitudinal Jerk, minimal TTC and Lane Change Duration

(t_{LC}), a reactive behavior enhances the longitudinal behavior of the lap vehicle v_{lap} as described in (6). The deceleration during the reactive part (dec_{LC}) is defined as a relaxation of the following behavior of the Intelligent Driver Model [22]. As we do not control the deceleration parameters from other road users a maximal conservative deceleration (b_{max}) based on experience is assumed.

$$v_{lead}(t) = v_{lead}(t_0) + a_{lead}(t_0) \cdot min(t_a, t), t \geq 0. \tag{5}$$

$$v_{lap}(t) = \begin{cases} v_{lap}(t_0) + a_{lap}(t_0) \cdot min(t_a, t), & t < t_{LC} \\ v_{lap}(LC) + dec_{LC} \cdot \Delta t_{LC}, & t \geq t_{LC}. \end{cases} \tag{6}$$

$$dec_{LC} = \begin{cases} b_{max}, & v_{lap} > v_{ego} \ \& \ s_{lap} \geq s_{safety} \\ 0, & else \end{cases} \tag{7}$$

A. Simulated experiments

The system was firstly evaluated on several entrance ramp scenarios. As shown in Fig. 1, the ego vehicle is forced to merge into the neighbor lane before the entrance ramp lane ends, several vehicles are driving on the contiguous lane and the ego-vehicle has to select the most adequate strategy. The scenario runs in a Co-Simulation of MATLAB/Simulink® and the traffic simulator Pelops [23]. Pelops offers different driver-types based on the work of Wiedemann [24] that are used for the simulation of the surrounding vehicles.

The experiment consist of 140 scenarios. Each one of the simulated vehicles drives with a different driver profile in order to evaluate the limitations of the planner and the dependency of the parameters to successfully achieve the mission.

The experiment was tested on four different systems, a baseline of a merely reactive application and three different configurations of the proposed approach:

- C1 is the baseline, where the lane change is only allowed when the current gap is free.
- C2 is the proposed approach with the first three parameters of the cost estimation and with a heuristic velocity selection instead of maneuver optimization.
- C3 is the proposed approach without the learned model, with a maneuver optimization through graph search.
- C4 is the proposed approach with the learned success probability and a maneuver optimization through graph search.

Fig. 6 presents the longitudinal jerk of the complete lane change maneuver and the Lane Change Time for the successful lane change. The minimal Time to Collision (TTC) for all the experiments is also presented. Literature considers TTC values as relevant under 5 seconds and critical under 2 seconds.

TABLE I
SIMULATION RESULTS

Approach	success rate	computational time	critical situations
C1	90.9%	0.0022 ms	8.7%
C2	94.9%	0.0037 ms	1.4%
C3	95.6 %	0.2839 ms	4.3%
C4	96.4 %	0.2848 ms	0.7 %

The longitudinal jerk gives an idea of the comfort of the maneuver. The TTC is an indicator of the vehicle's safety during the drive. The time until Lane Change indicates the time span between the lane change desire and the successfully completed lane change. Table I summarizes the rate of successfully accomplished lane changes, the computational time and the rate of critical situations ($TTC \leq 2 seconds$) for each strategy.

Lower values of longitudinal jerk point out a more comfortable behavior of the strategies C2 and C4 over C1 and C3. The TTC improves substantially with the use of a proactive strategy (C2, C3, C4) compared with a reactive strategy. Regarding the warn rate based on TTC, the values of a proactive strategy get firstly worse with the introduction of a maneuver optimization but then improve with the inclusion of the learned success probability. The proactive strategy improves generally the success rate of a lane change when the space is limited. The consideration of the prior experiences when selecting the strategy increases the success rate further. The computational time increases due to the maneuver optimization (C3, C4), but it is still computable in real time on the vehicle platform.

B. Real experiments

Simulations provide a useful tool to realize sensitivity analysis. To validate the correct integration of the planning approach with the vehicle, the system was also evaluated on a vehicle-in-the-loop configuration. The real vehicle (Fig. 5) drives on a test track and the road users run on an integrated traffic simulation. The planner runs on a real time platform Autobox®.

Fig. 7 shows a situation with a proactive strategy, where the vehicle reduces its velocity during a lane change to keep the safety distance with a slower front vehicle, when it is safe accelerates and finishes the lane change. The blue line represents the selected longitudinal and lateral trajectories and the red-ones are the measured state of the vehicle. The Figure shows how our planning framework is able to provide maneuvers that are smooth and feasible on the real system.

C. Discussion

The experiments show that the use of a proactive strategy increases the success rate for a lane change compared with a merely reactive strategy. The success rate of the proactive strategy can be increased improving the maneuver

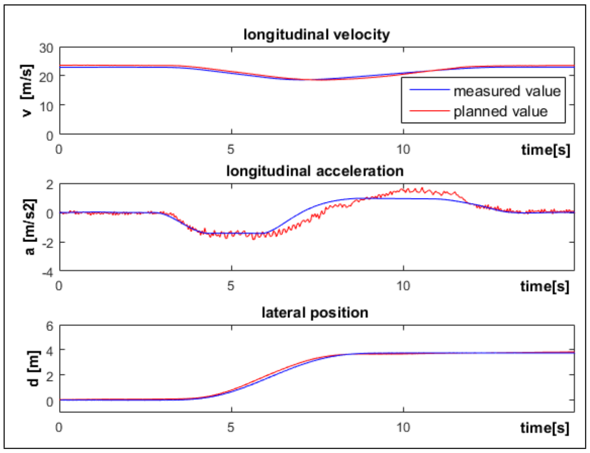

Fig. 7. Vehicle dynamics during the lane change maneuver

optimization for a selected gap, computing the maneuver over the time horizon with more detail. Nevertheless, the longitudinal jerk and the TTC statistics become at the first time lightly worse. This effect probably occurs because of the detailed optimization is more sensitive to changes of the selected gap over the time. The results improve with the introduction of a learned success probability. The planning including the learned success probability performs better without increasing the risk of the maneuver.

The model presents still some limitations. The assumption of other vehicles braking behind us within conservative limits allows us to change into gaps that are closing, that is, we assume a cooperative reaction of the other vehicles to our lane change. Nevertheless, on situations with high traffic density the original predicted gap size could be still too small and the proactive algorithm would not accomplish the lane change. This situations could be mastered with an cooperative strategy for gap opening, which is out of the scope of this paper.

Real experiments show a successful smooth integration of the planning on the real vehicle. It allows a safe validation of critical situations and enables the future step of driving with sensors on real traffic.

V. CONCLUSIONS

This paper presented a system that provides a robust framework for the autonomous driving task through the integration of semantic and numeric reasoning between different planning levels. It also introduced an uncertainty assessment method based on learned situations. The system has a highly flexible structure that allows to include different implementation levels depending on the available information and computational power. We presented experimental results obtained in simulation and on a real vehicle with virtual surrounding traffic. Future work will include the evaluation in real traffic situations.

REFERENCES

[1] B. Vanholme, D. Gruyer, S. Glaser, and S. Mammar, "A legal safety concept for highly automated driving on highways," in *2011 IEEE Intelligent Vehicles Symposium (IV)*, June 2011, pp. 563–570.

[2] D. Althoff, M. Werling, N. Kaempchen, D. Wollherr, and M. Buss, "Lane-based safety assessment of road scenes using inevitable collision states," in *2012 IEEE Intelligent Vehicles Symposium*, June 2012, pp. 31–36.

[3] M. Althoff and J. M. Dolan, "Online verification of automated road vehicles using reachability analysis," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 903–918, Aug 2014.

[4] M. Aeberhard, S. Rauch, M. Bahram, G. Tanzmeister, J. Thomas, Y. Pilat, F. Homm, W. Huber, and N. Kaempchen, "Experience, results and lessons learned from automated driving on germany's highways," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 1, pp. 42–57, Spring 2015.

[5] D. Ferguson, T. M. Howard, and M. Likhachev, "Motion planning in urban environments: Part i," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2008, pp. 1063–1069.

[6] T. Gu, J. Snider, J. M. Dolan, and J. w. Lee, "Focused trajectory planning for autonomous on-road driving," in *2013 IEEE Intelligent Vehicles Symposium (IV)*, June 2013, pp. 547–552.

[7] C. Rathgeber, F. Winkler, and S. Müller, "Kollisionsfreie längs- und quertrajektorienplanung unter berücksichtigung fahrzeugspezifischer potenziale." *Automatisierungstechnik*, vol. 64, no. 1, pp. 61–76, 2016.

[8] M. Werling and D. Liccardo, "Automatic collision avoidance using model-predictive online optimization," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, Dec 2012, pp. 6309–6314.

[9] M. Ardelt, P. Waldmann, N. Kaempchen, and F. Homm, "Strategic decision-making process in advanced driver assistance systems," in *Advances in Automotive Control*, 2010, pp. 566–571.

[10] M. Ardelt, C. Coester, and N. Kaempchen, "Highly automated driving on freeways in real traffic using a probabilistic framework," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1576–1585, Dec 2012.

[11] W. Schwarting and P. Pascheka, "Recursive conflict resolution for cooperative motion planning in dynamic highway traffic," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Oct 2014, pp. 1039–1044.

[12] M. Bahram, A. Lawitzky, J. Friedrichs, M. Aeberhard, and D. Wollherr, "A game-theoretic approach to replanning-aware interactive scene prediction and planning," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3981–3992, June 2016.

[13] D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, and D. Wilkins, "Pddl-the planning domain definition language," 1998.

[14] J. Frank and A. Jónsson, "Constraint-based attribute and interval planning," *Constraints*, vol. 8, no. 4, pp. 339–364, 2003.

[15] J. Barreiro, M. Boyce, M. Do, J. Frank, M. Iatauro, T. Kichkaylo, P. Morris, J. Ong, E. Remolina, T. Smith, *et al.*, "Europa: a platform for ai planning, scheduling, constraint programming, and optimization," *4th International Competition on Knowledge Engineering for Planning and Scheduling (ICKEPS)*, 2012.

[16] M. Fox and D. Long, "Modelling mixed discrete-continuous domains for planning." *J. Artif. Intell. Res.(JAIR)*, vol. 27, pp. 235–297, 2006.

[17] C. Dornhege, "Task planning for high-level robot control," Ph.D. dissertation, University of Freiburg, 2015.

[18] L. Zhao, R. Ichise, Y. Sasaki, Z. Liu, and T. Yoshikawa, "Fast decision making using ontology-based knowledge base," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, June 2016, pp. 173–178.

[19] B. Kuipers, "The spatial semantic hierarchy," *Artificial intelligence*, vol. 119, no. 1-2, pp. 191–233, 2000.

[20] S. Franklin and A. Graesser, "Is it an agent, or just a program?: A taxonomy for autonomous agents," in *Proceedings of the Workshop on Intelligent Agents III, Agent Theories, Architectures, and Languages*, ser. ECAI '96. London, UK, UK: Springer-Verlag, 1997, pp. 21–35.

[21] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH Journal*, vol. 1, no. 1, pp. 1–14, 2014.

[22] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Phys. Rev. E*, vol. 62, pp. 1805–1824, Aug 2000.

[23] F. Christen and Q. Huang, *Das Fahrermodell im Verkehrsflusssimulationsprogramm PELOPS : Modellierung und Applikationsmoglichkeiten*, ser. Fortschritt-Berichte VDI : Reihe 22, Mensch-Maschine-Systeme. Dusseldorf: VDI -Verl., 2008, vol. 28.

[24] R. Wiedemann, "Simulation des strassenverkehrsflusses," Ph.D. dissertation, Schriftenreihe des Instituts fuer Verkehrswesen der Universitaet Karlsruhe, Band 8., Karlsruhe., 1974.