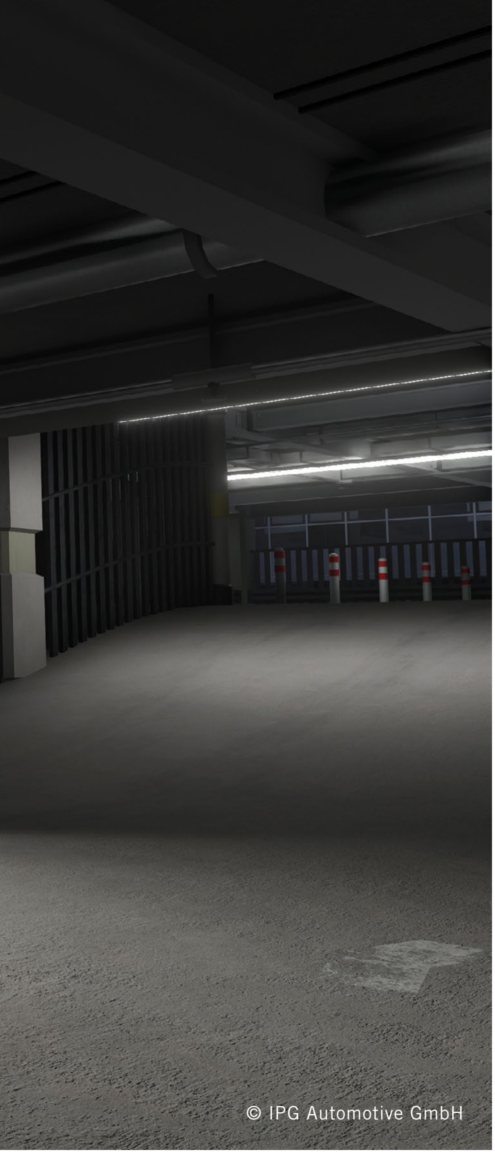
Automated Valet Parking – Simulation-based Driving Corridor Validation of a Trajectory Planner

Low-speed functions for passenger cars such as automated valet parking in parking garages have been prevalent in vehicle development settings for several years. Here, the Technical University of Darmstadt and IPG Automotive describe the driving corridor validation test of a trajectory planner. Based on a global reference trajectory, this assistance system calculates local sub-trajectories and subsequently travels them. Testing of this concept is based on simulation using a requirement and scenario catalog.

In discussions of autonomous driving (SAE level 5), the initial associations are often of robot taxis in urban areas which enable automated driving without human driving tasks. However, it is highly probable in reality that low-speed functions of SAE level 2+, 3 and 4 will be the first to be ready for series production.

Slow speed driving range includes automated valet parking that this article is going to address in greater detail. With automated valet parking [1], the driver leaves the vehicle after having entered a parking garage. Subsequently, the vehicle automatically drives to the parking space that it has been assigned and parks there. For that purpose, a communication function with the parking garage is established providing the vehicle with information about the position at which a free parking space is available in the parking garage. At the driver's request (at the end of the parking period), the vehicle independently pulls out of the parking space again and automatically drives back to the entrance of the parking garage.

The planned route guidance of the vehicle is referred to as global trajectory, whereas the space through which the vehicle travels following this trajectory is referred to as driving corridor. The simulation-based validation of this



WRITTEN BY

Bo Sun
is Research Assistant at the Institute of Automotive Engineering (FZD) of Technical University of Darmstadt (Germany).

Moritz Berghöfer
is Research Associate at the Institute of Automotive Engineering (FZD) of Technical University of Darmstadt (Germany).

Dr.-Ing. Sami Bilgic Istoc
is Consultant at IPG Automotive GmbH in Karlsruhe (Germany).

Henning Kemper
is Specialist Editor at IPG Automotive GmbH in Karlsruhe (Germany).

driving corridor is the focus of the present article. The global trajectory, for instance from the parking garage entrance to the targeted parking space, is assumed to be a given. The vehicle perceives the environment and, based on that, executes local trajectory planning [2] and vehicle control considering dynamic objects such as pedestrians or other vehicles. The so-called low-level trajectory planner plans the local trajectory.

Due to testing in simulation, it is possible to reproducibly test a very large number of scenarios and requirements in a short time – that is crucial to ensuring the safety of all traffic participants in all conceivable situations and achieves great testing depth and width. Reproducibility is a key element in that regard. The scenarios can be randomly varied, and every randomly generated scenario can be simulated under identical overall conditions as often as desired, enabling the continuous further development of automated driving functions.

REQUIREMENTS AND TEST SCENARIOS

In the context of the project on driving corridor validation carried out by the Technical University of Darmstadt and IPG Automotive and described in this article, examples of requirements were derived and defined. They can be divided into two categories: functional requirements and safety requirements.

Functional requirements encompass aspects such as trajectory updates. When the ego vehicle has completely traveled the trajectory segment or when a critical object has been detected and threatens to collide with the current trajectory, the trajectory is adjusted accordingly. Because the ego vehicle is always supposed to respond to the most critical object – in terms of risk of collision –, a differentiation must additionally be made between critical and non-critical objects.

The safety requirements, for example, concern the emergency braking function. As soon as the distance between the ego vehicle and an object falls below a predefined value the deceleration is raised to -4 m/s². Another requirement is that the ego vehicle must be able to follow a vehicle traveling in front of it at the same speed. If due to a critical object a deceleration is necessary, such deceleration additionally should impair ride quality as little as possible.

In the course of the project presented here, the driving function was tested by way of example in the scenarios 1 to 5, **FIGURE 1**. Each testing scenario served to prove that a respective requirement was met. It was tested whether the ego vehicle reacts according to the requirements in the respective scenario.

In the scenario 1, the necessity of a new plan of the trajectory arises because the originally planned global trajectory can no longer be traveled as planned due to previously unknown static objects. In this specific case, a cardboard box (blue square) on the planned trajectory prevents the ego vehicle from continuing to travel. In the scenario 2, several critical objects are present in the environment of the ego vehicle which must be included in the driving corridor validation accordingly. The identification of the most critical object is a prerequisite for the corresponding local trajectory adjustment. In the scenario 3, the necessity of an emergency braking event arises because an object is present in the vicinity of the ego vehicle and moving toward it, which causes a critical distance. The scenario 4 represents a succession situation in which an object precedes the ego vehicle in the same lane at constant speed. The final scenario 5 requires the ego vehicle to decelerate slightly due to an oncoming object.

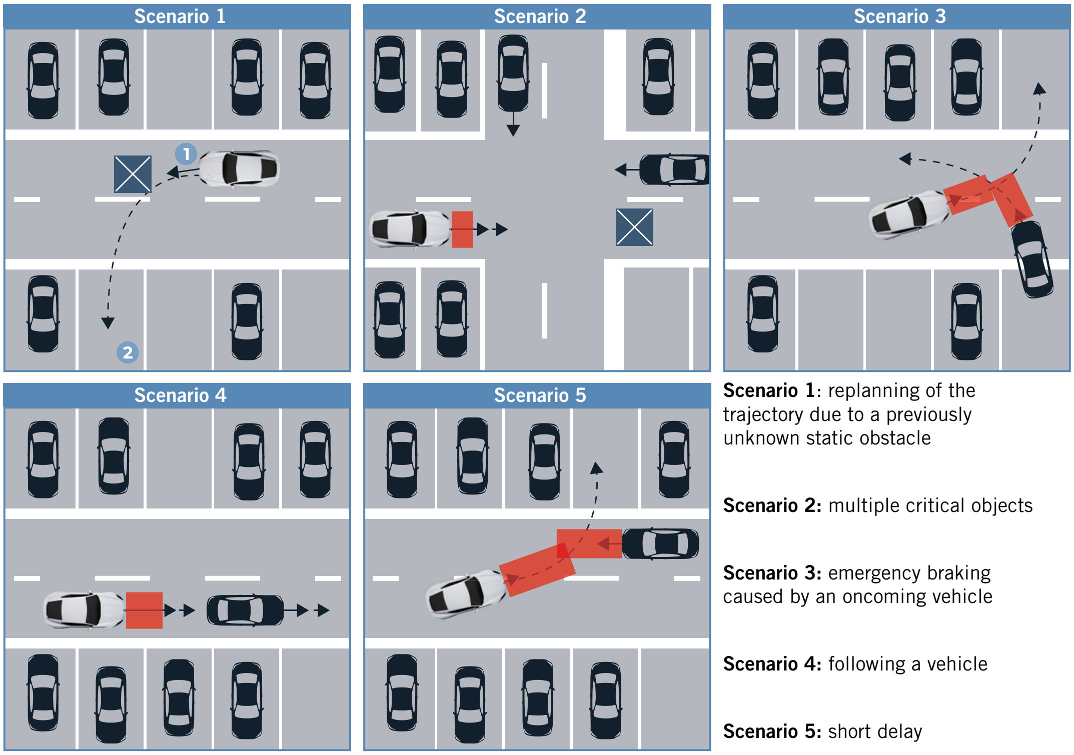


FIGURE 1 Test of the driving function in five different scenarios (© Technical University of Darmstadt, IPG Automotive GmbH)

Scenario 1: replanning of the trajectory due to a previously unknown static obstacle

Scenario 2: multiple critical objects

Scenario 3: emergency braking caused by an oncoming vehicle

Scenario 4: following a vehicle

Scenario 5: short delay

IMPLEMENTATION IN THE SIMULATION

The validation and control functions were integrated into the open integration and test platform CarMaker from IPG Automotive via a multi-level structure. For the actual adjustment of the required vehicle trajectory, the CarMaker's Trajectory Driver-Add-on was used. This is an integrated trajectory controller enabling adjustment of the ego vehicle to a required trajectory defined in terms of space and time. In addition, the controller allows dynamic updates of the trajectories during the simulation. As inputs, the spatial trajectory in longitudinal and lateral direction and the course of the required speed are expected. Consequently, this controller provides the interface between the

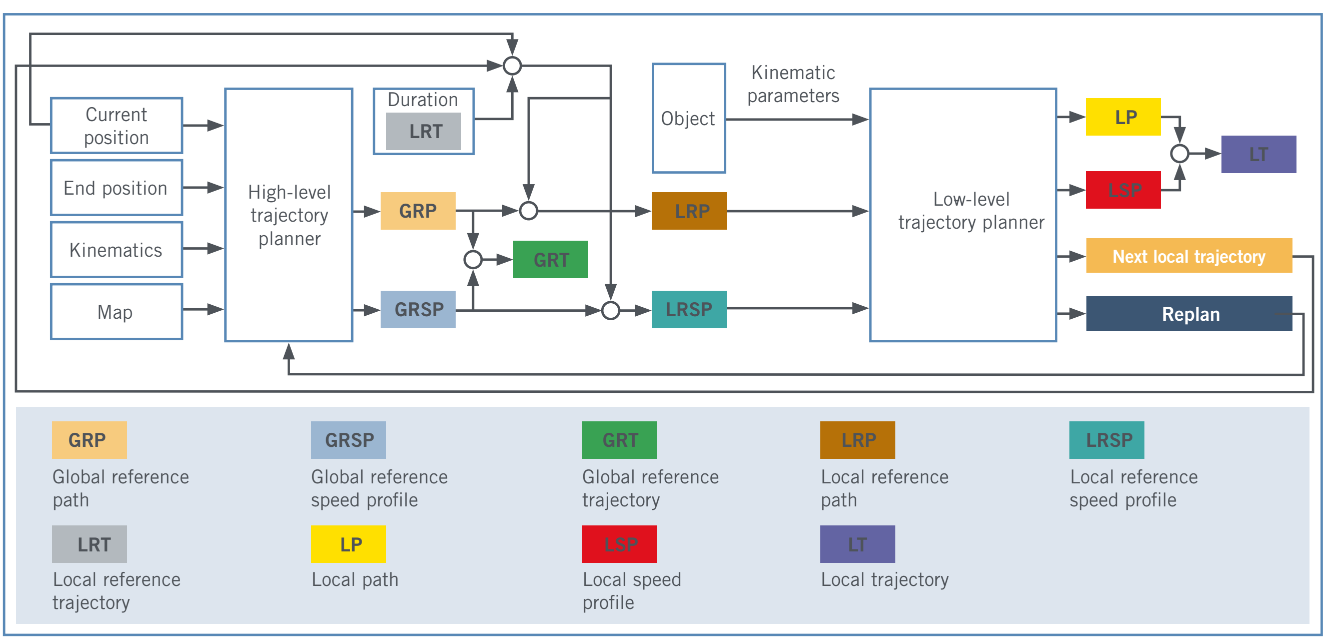


FIGURE 2 Total structure of the trajectory planner (© Technical University of Darmstadt, IPG Automotive GmbH)

FIGURE 3 Scenario in the parking garage: predefined trajectory (blue) and the trajectory actually traveled by the ego vehicle (orange) (© Technical University of Darmstadt, IPG Automotive GmbH)

actual acceleration-pedal-braking-steering events in the virtual vehicle and the low-level trajectory planner in the form of the validation and control functions for the dynamic trajectory.

The low-level trajectory planner itself was developed in the programming language C++ and directly integrated in CarMaker via the respective interface. The actual trajectory planning process is subdivided into various modes. They include emergency braking, replanning of the local trajectory caused by a newly detected obstacle or following a vehicle driving in front for example. **FIGURE 2** shows the total structure of the trajectory planner.

The high-level trajectory planner is responsible for the entire global trajectory – considering information from a highly precise map known beforehand. Therefore, only static and dynamic objects present in the map are considered. In contrast, the low-level trajectory planner generates a dynamic respectively local trajectory considering the previously unknown static and dynamic objects based on the simulated environment data. The local trajectory with the kinematic parameters is preplanned for the next 5 s and replanned either after 5 s or immediately in case of changes in environmental conditions – so-called triggers.

When one of the trigger conditions of the modes has been met while traveling the local trajectory, the corresponding replanning mode is activated and an updated local trajectory generated. If, for instance, a collision between the ego vehicle and unknown static or dynamic objects along the local trajectory is calculated, the emergency braking or deceleration mode must be initiated depending on the criticality. Of fundamental importance in this regard is the correct definition of the priorities of all modes.

By means of the Trajectory Driver-Add-on the predefined trajectory and the trajectory actually traveled by the ego vehicle can be visualized in various colors in the 3-D visualization environment IPGMovie, **FIGURE 3**.

Following the implementation of the trajectory controlling and planning process, additional test scenarios are necessary for running the simulation. They are modeled in the Scenario Editor of CarMaker as usual. For that purpose, the ego vehicle and other traffic participants such as other vehicles or pedestrians if applicable are positioned in a parking garage environment and provided with defined motion paths, **FIGURE 4**. In this case, the ego vehicle is a generic example vehicle because the vehicle itself is not the object of the investigations. The static trajectory is read in via the C++ interface of CarMaker upon starting the simulation. The other objects follow the routes that have been predefined in the Scenario Editor.

FIGURE 4 Test scenario in a parking garage including the trajectory traveled by the ego vehicle (white) (© Technical University of Darmstadt, IPG Automotive GmbH)

RESULTS

Due to the total system architecture, the vehicle control proves to be challenging even for supposedly simple tasks. Unknown factors such as acceleration and braking capabilities, steering performance, and potential starting delays affect the ability to follow the required trajectory. This leads to differences between the vehicle's end point after 5 s and the end point of the desired local trajectory that has been preplanned for 5 s. When the starting position of the vehicle for the subsequent intervals does not match the corresponding section of the reference trajectory, significant differences and controlling instabilities may occur. In the worst case, this will end in steering overshoot and potentially resulting collisions.

The discrete modes for global trajectory planning prove to pose a challenge to stable vehicle control, too. For instance, when the vehicle decelerates to follow a preceding vehicle. Here, overshooting behavior may occur as well due to reaction times caused by the discrete modes – in that case, in longitudinal direction when the controller of the ego vehicle in CarMaker brakes more forcefully than planned. In the worst case, the vehicle will constantly alternate between the modes, which may result in a zig-zag-shaped speed curve.

A prediction of the driving corridors of objects is very difficult when the available basic information merely consists of the current angle and speed of these objects. The reason is that objects do not maintain their speed or angle speed at constant levels. Particularly pedestrians have a major impact on the trajectory of the vehicle because their movements can be predicted only with great difficulty. The substantial differences result in wrong responses by the ego vehicle.

EVALUATION AND CONCLUSION

Based on a given reference trajectory, IPG Automotive implemented a full local trajectory control and dynamic driving corridor validation and tested it for the use case of valet parking on the simulation platform CarMaker. The project revealed that the selection of the control strategy has a significant impact on the results in terms of safety and comfort – among other things with reference to the prediction of the movement of other traffic participants. Particularly the movements of pedestrians proved to be hard to predict and often resulted in critical situations.

With simulation it is possible to identify critical situations and conceptual problems that would be detected only in later project stages with conventional test methods early in the development process. To enhance these results even further in the future, one of the focal points of future research will be the improved prediction of motion in the environment and resulting robust and safe control.

REFERENCES

[1] Banzhaf, H.; Nienhüser, D.; Knoop, S.; Zöllner, J. M.: The future of parking: A survey on automated valet parking with an outlook on high density parking. In: 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, USA, pp. 1827-1834, doi: 10.1109/IVS.2017.7995971

[2] Guo, Y.; Guo, Z.; Wang, Y.; Yao, D.; Li, B.; Li, L.: A Survey of Trajectory Planning Methods for Autonomous Driving – Part I: Unstructured Scenarios. In: IEEE Transactions on Intelligent Vehicles, doi: 10.1109/TIV.2023.3337318