7 Modeling ego motion uncertainty

In Chapters 4 to 6, the full planning stack was introduced and evaluated. In this chapter, a post-processing method is presented which compensates for ego motion uncertainty. So far, it has been assumed that the optimized trajectory would be executed by the vehicle’s control and actuator systems with no or only minor deviations. The trajectory actually driven can differ far more than that, however. Even though the planning results are still completely safe, these circumstances would cause frequent re-initializations. But any re-init strategy (presented in Section 5.1) is only a countermeasure and no solution to this problem. The idea of handling ego motion uncertainty within the planner was published in [9] for IEEE Intelligent Robots and Systems (IROS) in 2015. This chapter goes beyond the initial proposal.

7.1 Why modeling uncertainty matters

A trajectory actually driven may differ from the planned trajectory. The planner presented reacts with a re-init at the beginning of planning cycles. This process resets the initial pose from an earlier planning result to the current localization pose. Position and orientation experience a sudden reset with the consequence of a jerky input on steering wheel or throttle. In addition to a localization problem as shown in Fig. 7.1b, ego motion uncertainty addresses future vehicle poses and their dependencies by means of a vehicle dynamics model.

In addition, a receding horizon approach has the problem of not having global knowledge, which is present in all local planning approaches. For example, it is straightforward to think of a trajectory that allows an automated car to merge behind another vehicle. Let us assume vehicle ahead was predicted along its lane on the basis of a high acceleration measurement. For one, acceleration of other vehicles is hard to measure and it is even harder to predict is its future behavior. So for an prediction in the ideal world, the time gap would be a fit for a merge behind the car. The optimizer chooses the extreme maneuver on the assumption that every option is known in the decision process. As time passes, the system could find out that there is not as much space as initially assumed. Thus, all alternatives plans are then invalid. Ideas on how to cope with this issue in the context of automated driving have already been presented (e.g. hysteresis) [42]. Likewise, the measurements of the ego motion can be erroneous as well. Terrain, slope, weather or inertia can pull planned and real-world execution apart.

Modeling vehicle state uncertainty works as well for all traffic participants and their motion predictions. The linear-quadratic Gaussian (LQG) method that is the basis for the approach used in this thesis was also used in [108] to model the vehicle’s
7 Modeling ego motion uncertainty

Figure 7.1: All measurements are noisy and have errors. Perception sensor measurements describe object properties, such as size and motion. Localization methods and motion sensors measure the current state of the ego vehicle. Errors depend on various parameters: Ego vehicle speed, orientation change rate, sensor characteristics and quality, dirt and others.

Environment. The LQG approach supports the planning system by canceling out potential trajectory candidates that are likely to be invalidated after a small number of control cycles. Fewer re-init calls add value to the robustness, which is addressed in research question three.

Similar to probabilistic sampling-based planners, the post-processing method has the property of being probabilistically complete [13], as the algorithm’s probability of solving the given (solvable) problem converges to 1 when the run time approaches infinity. Thus, it is guaranteed that the post-processing method will find a solution to the trajectory generation problem, provided that it has a sufficient amount of time.

7.2 Ego motion uncertainty

Every real-world control problem has to deal with uncertainty. This also applies to vehicle driving modes that do not touch the limits of the circle of friction. Terrain properties, slope, wind or latencies in the vehicle’s communication and drivetrain systems can cause increasing numbers of measurement errors in the control loop, as shown in Fig. 7.2a. As it is not possible to eliminate the cause, it can be compensated for by modeling the existence of motion uncertainty into the system. The state transition is described by a function which contains the vehicle’s dynamics and a