Master's Thesis

Supervisors: Prof. Dr. Alexander Pretschner, Florian Hauer
Email: {alexander.pretschner, florian.hauer}@tum.de
Phone: +49 (89) 289 - 17885
Starting date: immediately

Context
Driver assistance systems exist for over three decades now with increasing functionality and the overall goal of highly autonomous driving seems to be not out of reach anymore [1]. However, the systems are getting increasingly complex as they are not only passive, but active systems interfering with the driver. Thus, for advanced driver assistance systems (ADAS) extensive testing needs to be performed, before they can be deployed for series production [2][3].

For an autonomous highway pilot, it is estimated that approximately 6.62 billion kilometers of test driving on highways are necessary [4]. Considering this and other complexity and feasibility issues, simulation is arguably the most practical and effective way of testing software systems used for autonomous driving [5].

A lot of such simulation tools exist, e.g. CarMaker by IPG Automotive [6]. However, within these tools, test scenarios are created and evaluated in a manual and very ad hoc manner. Test results are evaluated manually or the test evaluation script is written manually.

To improve this process, specification-based test oracle generation could be done. The specifications consist of preconditions, which have to be fulfilled for this specific test oracle, and the desired system behavior, which is tested for. Both is automatically transferred to evaluation scripts analyzing the simulation values.

Figure 1: Screen shots from CarMaker: Automated emergency braking system in action on the left and the simulation values on the right.

Goal
The main goal of this thesis is the development of a framework for automated test case evaluation. From a user-provided specifications, necessary preconditions and the desired system behavior is transformed to test evaluations scripts. This includes several partial goals:

1. The preconditions and the system behavior has be specified by the user. For this, a suitable way of description has to be chosen.
2. Both are formulated for simulated values, provided as values over time. The transformation from specification to specific interactions of such values has to be automated, possibly by using transformation templates.
3. Once the user finished specification, the process should be automated. For this and the first steps, a framework has to be developed.
Working Plan
1. Get familiar with CarMaker and understand how simulation based testing is done
2. Develop a description technique for preconditions and system behavior
3. Implement a toolchain for test case evaluation
4. Identify templates like described above
5. Round out your framework for automation
6. Evaluate your framework by applying it to a lane-keeping system

Deliverables
- The framework’s source code and modules
- A demo of the framework, including instructions on how to run the demo
- Technical report with comprehensive documentation of the implementation, i.e. design decision, architecture description, API description and usage instructions
- Final thesis report written in conformance with TUM guidelines

References