Scenario Test Analyst Homework

Aarati Noronha
Carnegie Mellon University
anoronha@andrew.cmu.edu

1. Task1

Define test scenarios for the given operational design domain (ODD)

1.1. Assumptions

Assumptions for the self driving vehicle (SV)

- The SV is only intended to make right-turns to do the loop.
- There are no passengers on-board, only a safety driver.
- The SV is not intended to operate in inclement weather.
- We assume that the speed limit is 30 kmh per hour the entire loop.

Assumptions about the given operational design domain

- We assume the trajectory is a closed loop around the office starting and terminating at Ungererstr. 69 in Munich as shown in figure 1. Upon closer inspection, baricades at Fritz Hommel Weg reveal that SVs cannot enter. The route is therefore through the roads as follows - Ungerstrasse – Soxhletstrasse – Theodor-Dombart-Strasse – Schenkendorfstrasse
- We assume that there are no traffic lights and construction zones present.
- We assume that there are only Pedestrians, Bicyclists and normal mid-sized cars present.

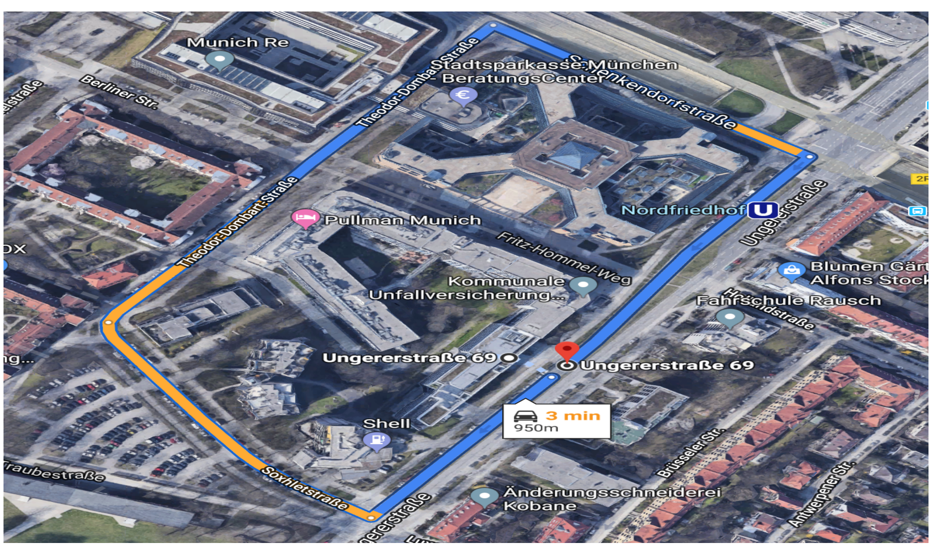

Figure 1: Given operational design domain

1.2. Define Test Scenarios

Around 12 test scenarios have been defined. The driving functionality that is tested in each case has been delineated in italics. Some of the scenarios are accompanied by images from the street view of the operational design domain. The images serve as examples of particular scenarios and are not to be construed as general representations of the scenario at large. These scenarios have been further categorized on the basis of what the self driving vehicle is responding to.

1.2.1 Scenarios involving other vehicles

1. Adjacent vehicle cutting in
   SV travelling on a 2 lane one way road like the one shown in figure 2. An adjacent vehicle attempts to cut in. The SV intends to travel straight ahead.
   The SV must be able to yield, decelerate, stop, shift within a lane or change lanes.

2. Parked vehicle cutting in ahead
   SV is travelling in the right lane of a 2 lane one way road. A vehicle parked on the right is attempting to cut into the lane ahead of the SV. Both the SV and the previously parked vehicle intend to travel straight ahead.
   The SV must be able to yield, decelerate, stop, shift within a lane or change lanes.

3. Lead vehicle cutting out or performing lane change
   SV is travelling on a straight road and intends to keep moving straight. Lead vehicle intends to change lanes or cut out to enter the petrol pump as shown in the figure 3.
   The SV must be able to accelerate, decelerate and stop

4. Lead vehicle making a turn
   SV is moving along a straight road. The lead vehicle in front intends to make a turn up ahead. The SV can either make the same turn or travel straight ahead.
   The SV must be be able to decelerate, stop, change lanes and safely pass the vehicle if it intends on following a straight path

5. Lead vehicle approaching unsignalized intersection
   SV is moving along a straight road. The lead vehicle in front approaches an unsignalized intersection. *The SV must be be able to decelerate, stop, accelerate and perform lane change if necessary*

6. SV approaching unsignalized intersection
   SV approaches an intersection without signals as assumed. A vehicle approaches the same intersection from the right. The SV intends to travel straight ahead or make a turn as shown in figure 4.
   The SV should yield to all vehicles going right, decelerate, stop and accelerate

7. SV changing lanes or merging in
   The SV intends travelling straight intends to change lanes on a 2 lane or three lane road in order to travel at a different speed or perform lane change at a later period. This is depicted in figure 5
   The SV must be able to yield, stop, decelerate, accelerate in order perform lane change. It should also be able to detect the white arrows on the ground

1.2.2 Scenarios involving pedestrians and cyclists

1. SV approaches unsignalized intersection controlled by yield signs and pathways for cyclists and pedestrians
   SV intends to move straight ahead or make a right turn. Cyclist or pedestrian has right of way and is crossing the road the SV traverses as shown in figure 7
   The SV must yield to all vehicles coming on the right, must yield, decelerate, stop before all cyclists and pedestrians in path, should accelerate to resume path.

2. SV attempts a right turn in the vicinity of cyclists
   SV is travelling on a straight path and intends to make a right turn ahead. A cyclist exists in the adjacent right lane to the SV and intends to make the same right turn.
   The SV must be capable of safely passing the cyclist by accelerating and making the turn without cutting him off too

Figure 2: Adjacent or parked vehicle cutting in

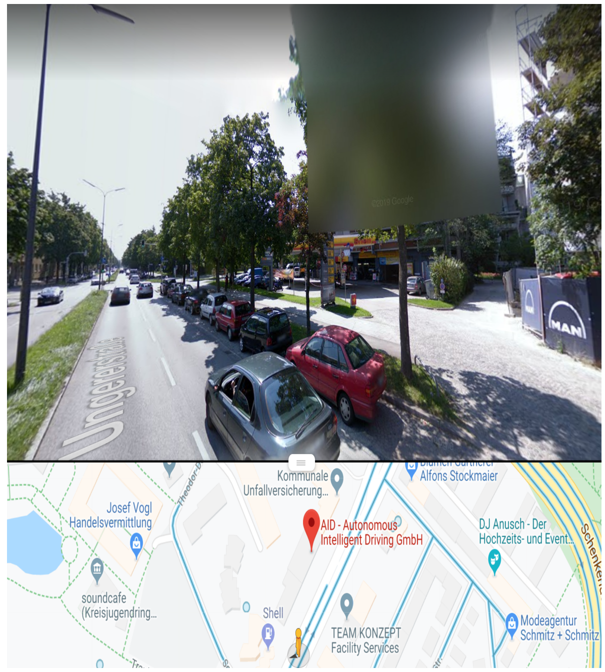

Figure 3: Lead vehicle cutting out: Entering petrol pump

closely. If this is not possible, the SV must be able to yield, decelerate, stop, accelerate and make the turn after the cyclists makes it.

1.2.3 Scenarios involving the city infrastructure

1. SV travelling under overhead bridge
   SV travelling on a straight road performing normal car following. An overhead bridge dims the light around the SV temporarily a shown in figure 6.
   The SV must be able to decelerate, stop and perform car following based on reasonable predictions. Its sensors should have partial visibility to make proper estimations.

1.2.4 Scenarios involving emergency vehicles

1. Emergent vehicle in the vicinity of SV
   SV travelling on a straight road. Emergent vehicle (ambulance, police or school bus) is trying to bypass trafic flow in the same direction, opposite direction or even in the wrong lane
   The SV must be able to decelerate, stop, perform quick lane change and yield to the emergent vehicle

1.2.5 SV approaching traffic signs

1. In the given operational design domain, several signs were observed such as stop, yield, crosswalk, no stopping and waiting, bicycle lane.
   The SV should be able to stop, yield, accelerate, decelerate and give right of way accordingly

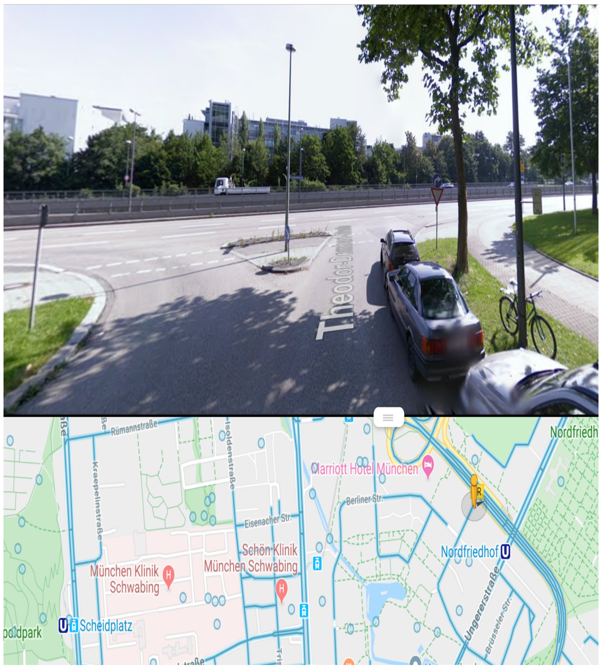

Figure 4: SV approaching unsignalized intersection

Figure 5: SV changing lanes or performing cut in

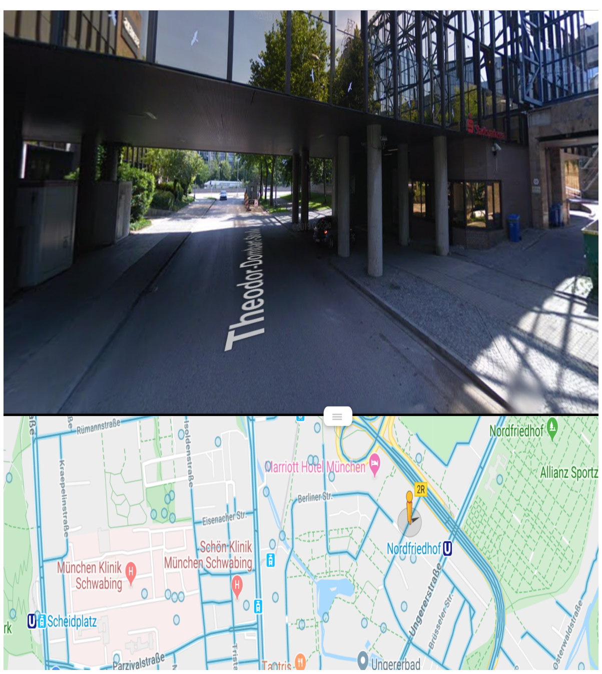

Figure 6: SV travelling under overhead bridge

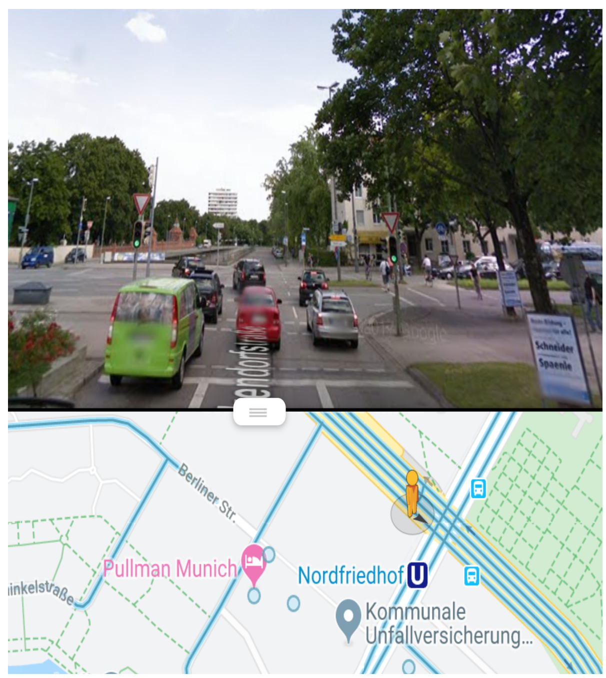

Figure 7: SV at intersection with cyclists, pedestrians

2. Task2

Develop a verification and validation plan for one of 3 approaches Basically, testing, validation and verification of an L4 autonomous vehicles can be broadly divided into 2 categories: hardware testing and software testing. Less emphasis is given to my elaboration of hardware testing.

2.1. Base level Hardware Testing

Testing of hardware components must be done at every level of integration on every part, every subsystem and every system as a whole individually before the vehicle is subjected to on-road tests. Each component be it a cable, connector or door must be subjected to a stress test and performance test under inclement conditions to ensure optimal functionality. Failure mode effect analysis (FMEA) is performed for each component.

2.2. Software Testing

Software testing can be approached in different ways: simulation, testing on a specific course closed to the public and on road testing.

2.2.1 Simulation

Modelling and simulating the self driving vehicle and the environment it traverses is one of the most feasible options for autonomous software testing. It allows you to extract scenarios from the given real world data and turn them into virtual scenarios for the autonomous vehicle to practise on. It also allows for testing of system and subsystems and different control inputs for each test scenario.
Advantages

1. Inexpensive
2. Saves time as the temporal relation can be skewed so that autonomous vehicles are subjected to more test scenarios in a much shorter time than the real world
3. Systems, subsystems and hardware components can also be tested for failure and the SV's reaction to it through efficient modelling
4. Different control inputs and test variables can also be tested
5. A million test scenarios can be synthetically generated based on a few scenarios from extracted data by changing the conditions during operation
6. Can skew the statistics of conditions that are anomalies to facilitate accelerated testing

Disadvantages

1. If the system is not modelled correctly or a slight change exists in the design, the simulation will no longer stand effective and will not tackle real world scenarios as efficiently as predicted

2.2.2 Testing on a course closed to the public

Extremely skilled drivers who have undergone high level performance training take the L4 autonomous vehicles to a track or course closed to the public. This facilitates a more realistic approach to testing actual features of the vehicle as opposed to simulation. Different releases of software can be tested in the same operational design domain using this approach.
Advantages

1. Offers a significant amount of control over the ODD, test conditions and control inputs
2. Can repeat the same test with the same conditions with different software versions to test highest efficiency
3. Allows actual real life testing of features of the vehicle

Disadvantages

1. Very expensive in the long run
2. Time consuming
3. Impossible to generate a very wide variety of road scenariosnfor testing
4. Not very safe for the driver
5. Does not account for traffic scenarios that are anomalies

2.2.3 On road testing

On road testing should be done after the two afore mentioned approaches. It is the final step and there is little control over it. The greater the distance covered by the vehicle safely during on-road testing, the higher the level of confidence in the software version. *Advantages*

1. Is an actual test with little to no control over the SV in any ODD
2. Allows actual real life testing of features of the vehicle

Disadvantages

1. Expensive
2. Time consuming
3. Little to no control can be exerted over the ODD, test conditions or the control input
4. Not very safe for the driver or the commuters on the road
5. Software version may not be able to handle a certain anomaly

2.3. Emphasis on modelling and simulation as an approach for our ODD

In an ideal situation, all 3 of the testing approaches mentioned above would be deployed in the greatest detail for the testing and verification of our level 4 self driving vehicle. However for the sake of convenience and as per the instructions, I have chosen simulation and modelling as the go to approach for my system verification plan for the given ODD.

2.3.1 Test Requirements

1. A modelling and simulation software that supports hardware component and software integration is required. In recent time, the simulation software CARLA has gained popularity over conventional software like MATLAB. As quoted by the developers [1], CARLA has been developed from the ground up to support development, training, and validation of autonomous driving systems. In addition to open-source code and protocols, CARLA provides open digital assets (urban layouts, buildings, vehicles) that were created for this purpose and can be used freely. The simulation platform supports flexible specification of sensor suites, environmental conditions, full control of all static and dynamic actors, maps generation and much more.
2. In addition, any real world data collected from similar operational design domains would be of great help to re-enact scenarios that have actually occurred

2.3.2 Description of ODD

The ODD consists of the loop around Ungerestrasse 69. It includes several unsignalized intersections, a bridge, several traffic sign boards. The weather is not inclement at any given time. All the scenarios that have been described in section 1 can occur in the ODD.

2.3.3 Modelling the system

The SV is modelled as an exact replica of its real world counterpart down to the subsystems and dynamics. The SV is equipped with **sensors** (lidars, radars, cameras in addition to conventional sensors for steering and throttle). They facilitate **perception, mapping and localization**. It basically helps describe and **model the state of the system** at large and the **other agents** in the environment along with city infrastructure. The software has pre-defined **decision and control algorithms** to deal with different scenarios. This is also loaded into the simulation system. Lastly **control input and feedback** are integrated in the simulation through modelling the necessary vehicle dynamics so that the SV can perform the necessary manoeuvres safely.

2.3.4 Running the generic tests

1. Extract the traffic scenarios defined in section 1 from data-sets with similar ODDs. Subject the autonomous vehicle to these tests.
2. Define a thousand similar tests based on observation and research and include possible anomalies and thresholds that may not have been defined. Skew control inputs and operating conditions to generate synthetic scenarios with different permutations and combinations of the original scenarios.
3. Subject the SV to millions of miles of driving daily around the loop in simulation by skewing the temporal relation that exists in traffic scenarios.

2.3.5 Running advanced tests

The challenges of AV evaluation [2] stem from two facts. i) Crashes are exceedingly rare events. In the U.S., one needs to drive on average 530 thousand miles to experience a police-reported crash and nearly 100 million miles for a fatal crash. ii) AVs can cheat to pass predefined tests. Traditionally as described above, vehicle test protocols and test conditions are pre-defined and fixed. This is not a problem when the vehicle is dumb, but becomes a problem when the vehicle is intelligent and can be customized to excel in the predefined tests, and performance in other test conditions receives less attention.

1. Extract rare events, near crashes and anomalies from other datasets. Skew the temporal relation between them, perform Monte Carlo simulations with the skewed probability density functions and skew this back to attain testing performance under extreme conditions.
2. Extract safety critical events from other datasets - events that are not generic, are considered very rare but still have a solution by the controller. They can be extracted by using multiplayer differential game theory. Unsafe backward reachable sets are defined from the collision set and if the SV has entered the backward reachable set at any time during its trajectory through a given scenario, the scenario is deemed safety critical.

2.3.6 Performance Quality Metrics

An L4 autonomous vehicle is evaluated using the following metrics in the given ODD.

1. Activation of manual mode
   If the ADAS system is not equipped to handle a scenario, it fails with a warning, is deactivated and the car shifts into manual mode. It is then considered a failure for the software that has been pushed to the system.
2. Separation distance between SV and other vehicles, pedestrians and cyclists
   Another important metric is the separation distance that exists between the SV and any vehicles, cyclists or people. It should not be below a pre-defined threshold. Also if the car crashes, it is considered an instant failure.
3. Aggressiveness or passiveness of a approach
   Another evaluation metric is the ratio of the relative velocity to relative distance between the SV and other bodies. It describes the comfort with which a manoeuvre is executed.
4. Failure status of subsystems
   Another metric of performance is the failure status of the hardware or the individual subsystem and component status in the simulation after the SV is subjected to a host of different scenarios. This is however subject to how accuractely the vehicle was modelled

2.3.7 Definition of done

The L4 autonomous vehicle is said to have passed each test successfully if it does not crash, obeys all the traffic rules, sticks to operating under the assumptions made for the given operational design domain and passes all the afore mentioned performance metrics.

3. References

1. Dosovitskiy, Alexey, et al. "CARLA: An open urban driving simulator." arXiv preprint arXiv:1711.03938 (2017).

2. Zhao, Ding, et al. "Accelerated evaluation of automated vehicles in car-following maneuvers." IEEE Transactions on Intelligent Transportation Systems 19.3 (2017): 733-744.