Scenario Factory: Creating Traffic Scenarios For Automated Vehicles

Advanced seminar: Simulation-based Testing of Autonomous Cars

Shine Jose 90767
Chair of Software Engineering II
University of Passau, Passau

Overview

- Introduction
- Background
- Scenario Factory
- Architecture & System Flow
- Test Results
- BeamNG Implementation
- Results
- Evaluation
- Conclusion

Introduction

- Virtual testing is an important tool for validating the safety of automated vehicles.
- Even though the number of publicly-available datasets has increased, they usually feature only a small number of maps and require much effort to record.
- The framework generates a set of database, safety-critical scenarios for testing of motion planning algorithms for automated vehicles (AV).

Introduction

- The safety validity of motion planning algorithm for AV´s need large amount of data for Virtual Testing(VT), so they have collected data from real test drivers which is **costly and in-efficient**, also they considered the minority of traffic scenarios posing challenges to motion planners.

- To overcome all this critical issues, In the paper they have introduced 3 major studies as follows:

1) Open Street Map (OSM).
2) Traffic SUMO Simulator.
3) Non-Linear Optimization.

Architecture

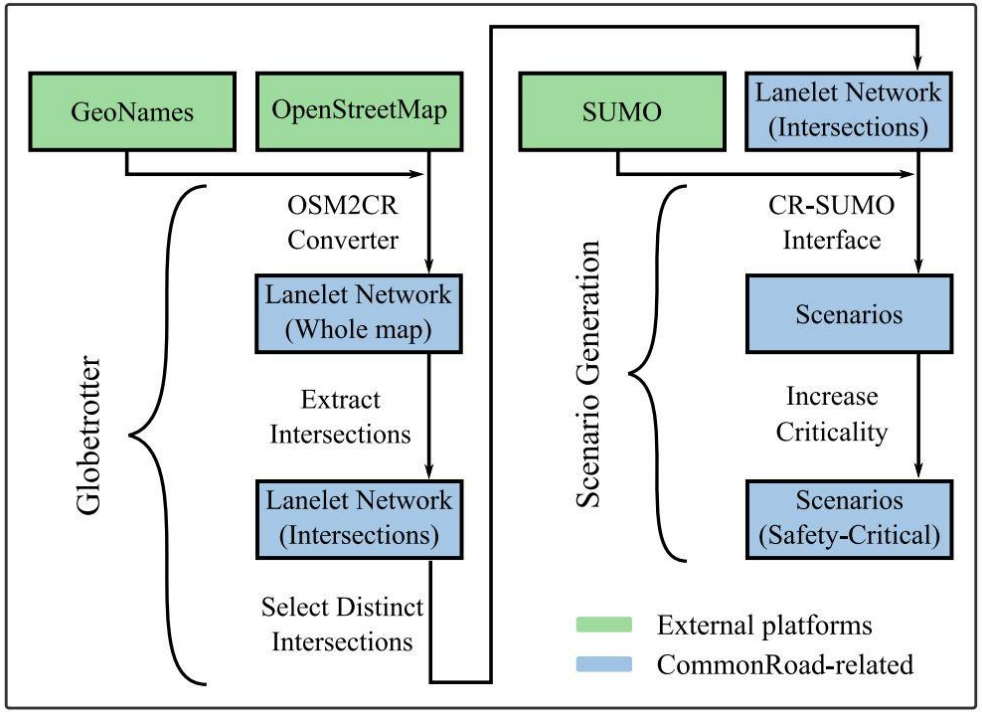


Figure 1. Architecture Process [1].

Inputs

```
{
   "base-path": "./scenarios/interactive/NGSIM/US101",
   "scenario-name":"USA/US101-27-4-1-1-1",
   "base-path":"./scenarios/interactive/SUMO/",
   “scenario-name":"USA/US101-27-47I-1-1",
   "output-path“ :"./outputs",
   "gui":true,
     "ego-vehs":{
      "veh0":{
         "wid": 1.6,
         "length": 4.3,
         "departSpeed":0,
         "pos":[0, 0, -0.68]
      }
  }
  }
```

Inputs

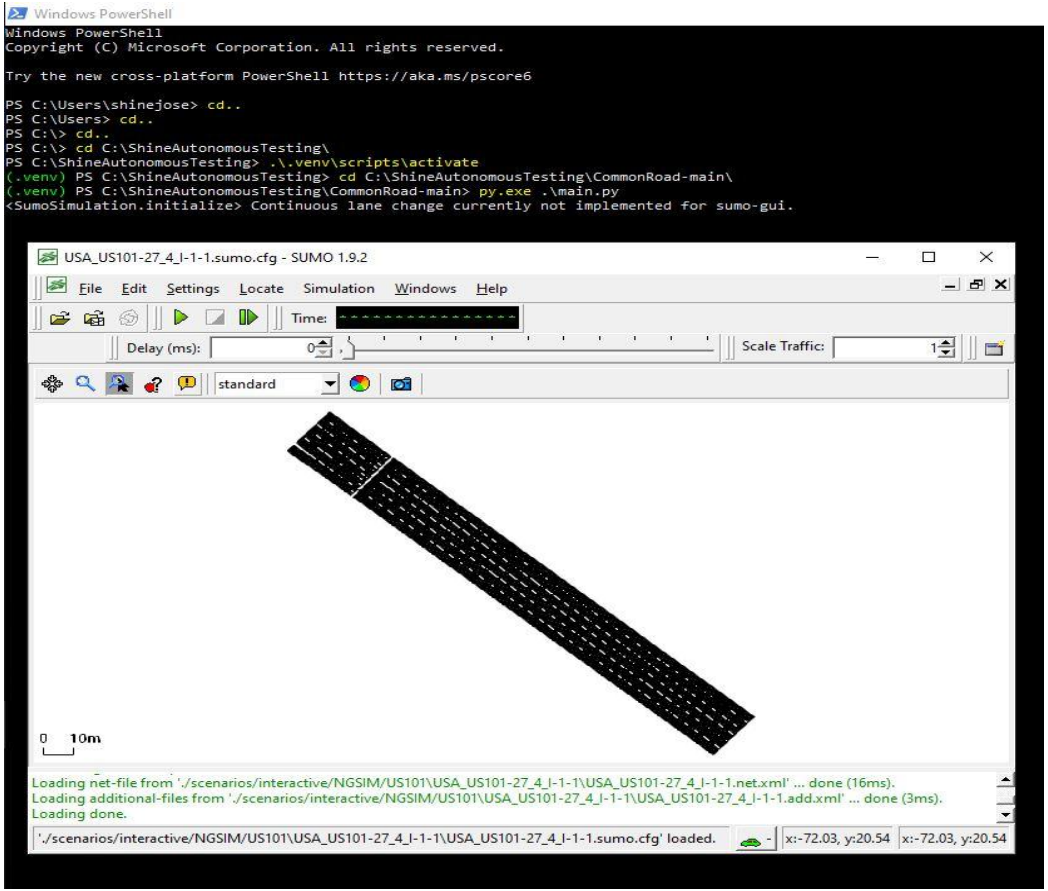


Figure 2. SUMO Inputing [2].

Output Generated

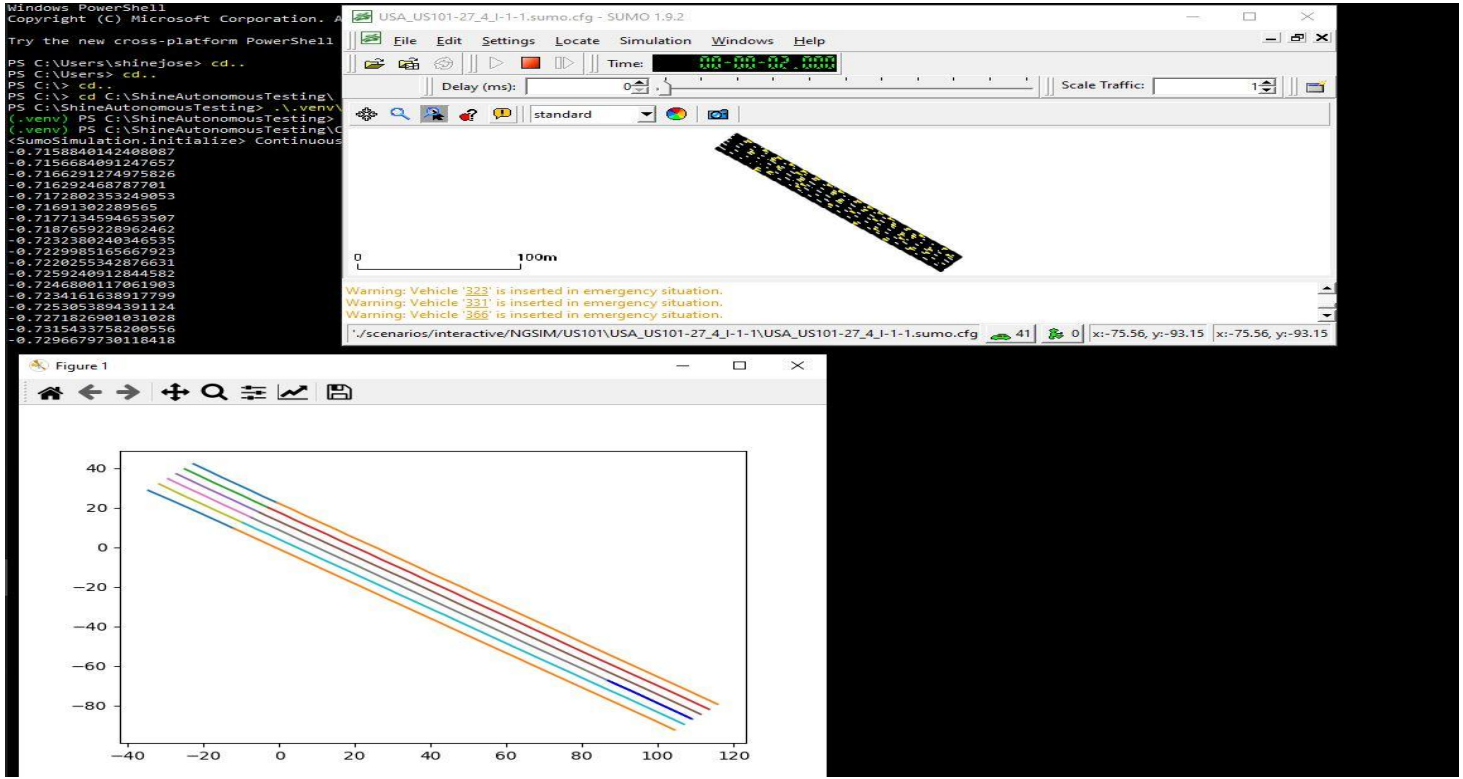


Figure 3. SUMO Simulation Of Given Map Inserted [3].

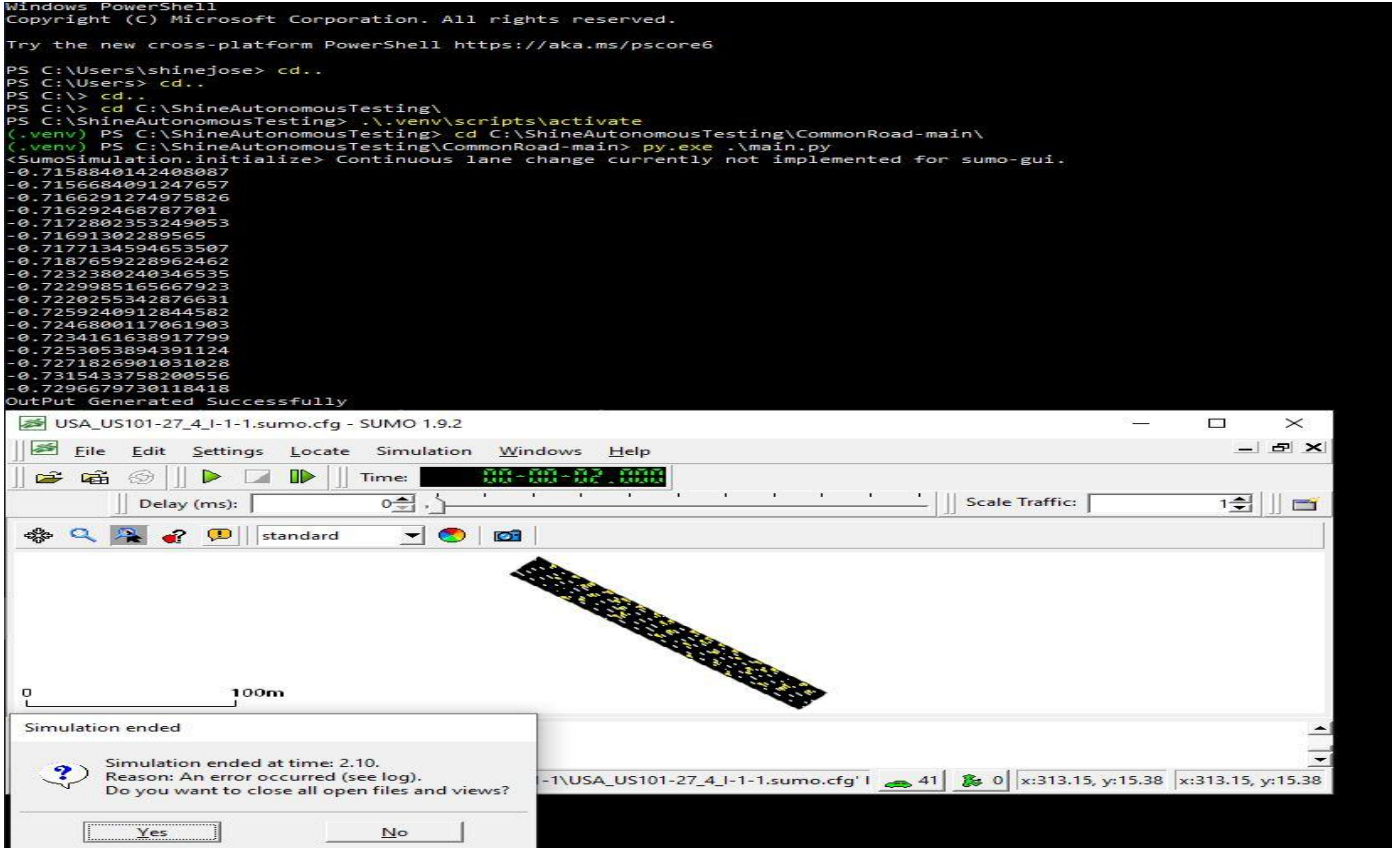


Figure 4. To add a new map [4].

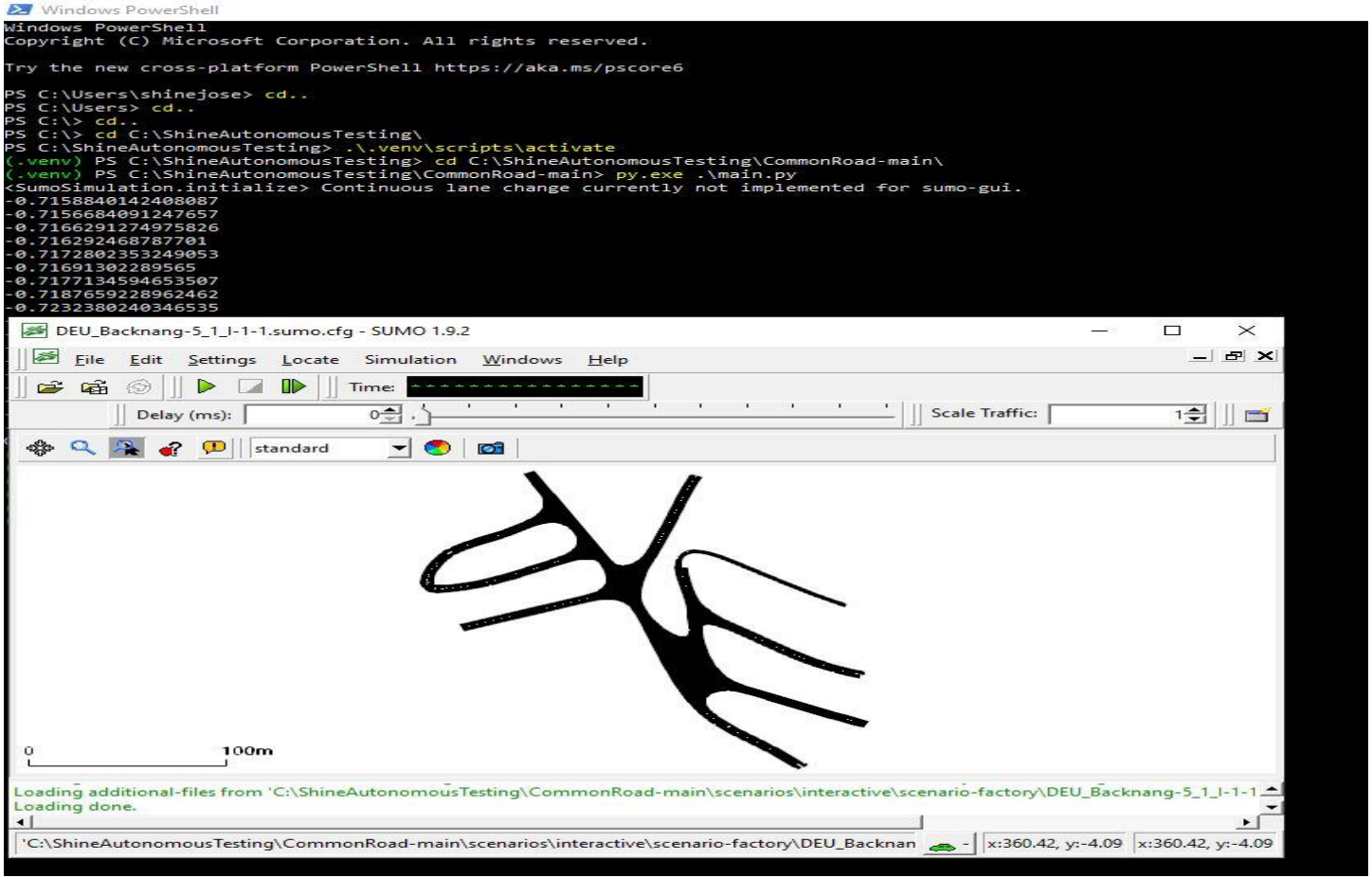


Figure 5. New Map & Features Added [5].

Output Generated

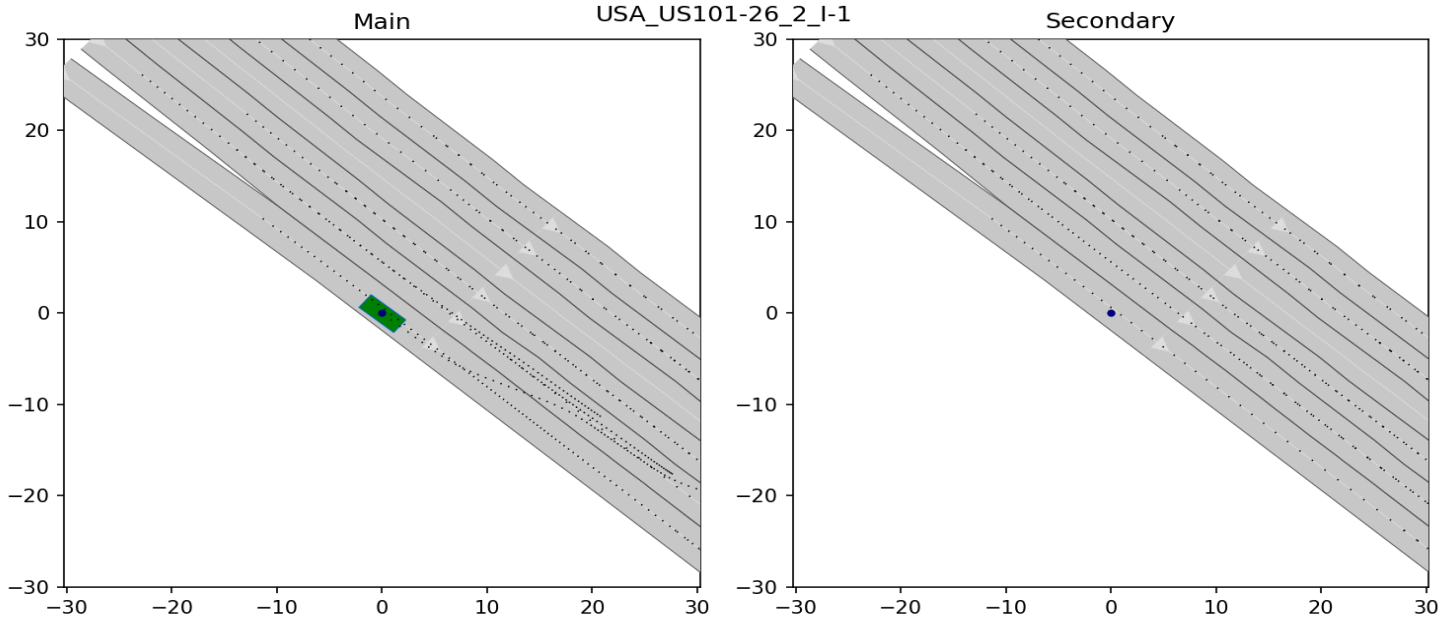


Figure 6. SUMO Simulation Of Given Map Inserted [6].

Results

- This approach can significantly decrease the drivable area, and thus it increases the criticality.

- And the second image shows the reduction of critical area.

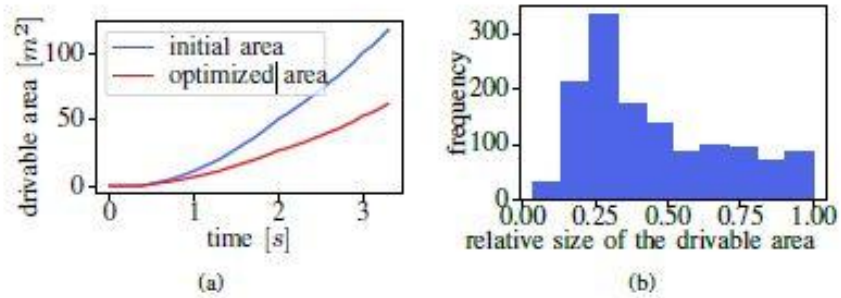


Figure 7. SUMO Simulation Of Given Map Inserted [7].

BeamNG.Tech

- The same Scenarios what we tested before are going to test in BeamNG tech simulator.
- Roads, Vehicles are created and placed using the same old files from Scenario Factory.
- We must extract the XML file format to JSON file format and to test the simulation successfully.
- Let's get into the simulator testing.

Inputs

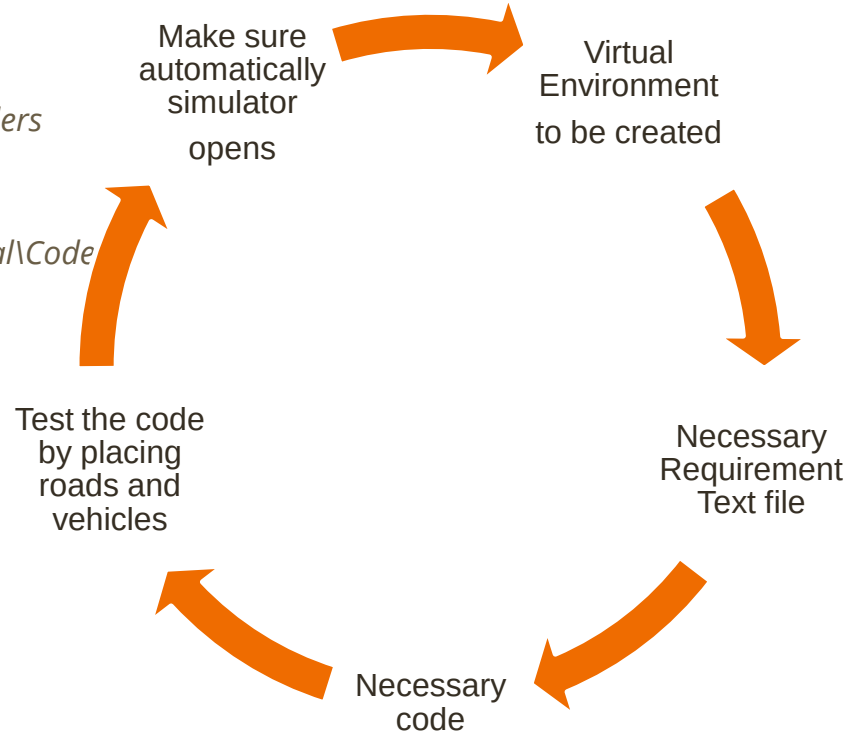
```
from beamngpy import BeamNGpy, Scenario, Road

# Specify location of BeamNG home and BeamNG user folders

BNG-HOME = "C:/ShineAutonomousTesting/BeamNG.tech"
BNG-USER = "C:/ShineAutonomousTesting/sbst-2021-tutorial\Code

beamng = BeamNGpy('localhost',
64256, home=BNG/HOME, user=BNG/USER)

# Start BeamNG by setting launch to True
bng = beamng.open(launch=True)
try:
input('Press enter when done...')
finally:
bng.close()
```



Evaluation

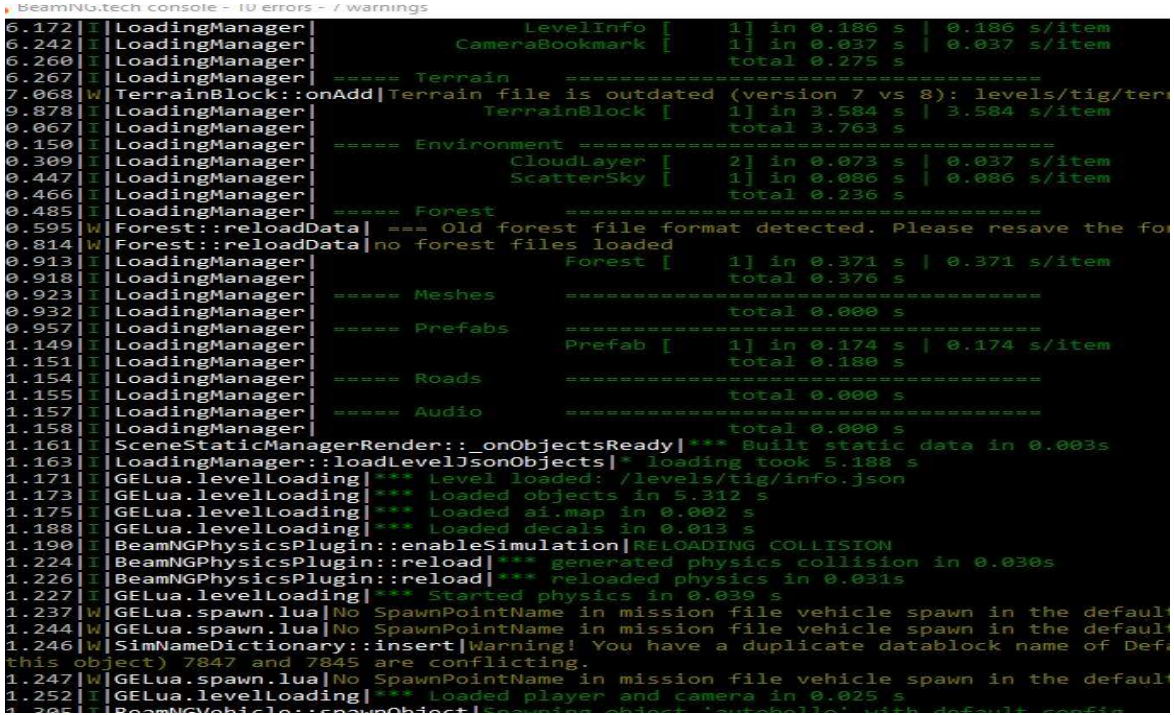


Figure 8. Given Prefabs, Meshes & Roads are Loading [8].

```
(.venv) PS C:\ShineAutonomousTesting\sbst-2021-tutorial\Code> py.exe .\xmlvisualization.py
<Element 'commonRoad' at 0x000001B1C15844A0>
commonRoad {'timeStepSize': '0.1', 'commonRoadVersion': '2020a', 'author': 'Edmond Irani Liu, Fabian Höltke, Moritz Klischat', 'affiliation': 'Technical University of Muni
ch, Germany', 'source': 'Scenario Factory (OpenStreetMaps, SUMO Traffic Simulator)', 'benchmarkID': 'ARG_Carcarana-4_1_T-1', 'date': '2020-08-23'}
location {}
scenarioTags {}
lanelet {'id': '7223'}
lanelet {'id': '6255'}
lanelet {'id': '7238'}
lanelet {'id': '6259'}
lanelet {'id': '7890'}
lanelet {'id': '7945'}
lanelet {'id': '5847'}
lanelet {'id': '7237'}
lanelet {'id': '5843'}
lanelet {'id': '7222'}
lanelet {'id': '7891'}
lanelet {'id': '7946'}
lanelet {'id': '7239'}
lanelet {'id': '7224'}
lanelet {'id': '7892'}
lanelet {'id': '7944'}
lanelet {'id': '7240'}
lanelet {'id': '7241'}
lanelet {'id': '7242'}
lanelet {'id': '5846'}
lanelet {'id': '6258'}
lanelet {'id': '7225'}
lanelet {'id': '7226'}
lanelet {'id': '7227'}
lanelet {'id': '7888'}
lanelet {'id': '7889'}
lanelet {'id': '6256'}
```

Figure 9. Features Extracted From XML File Format [9].

Figure 10. Features Extracted From XML File Format To JSON [10].

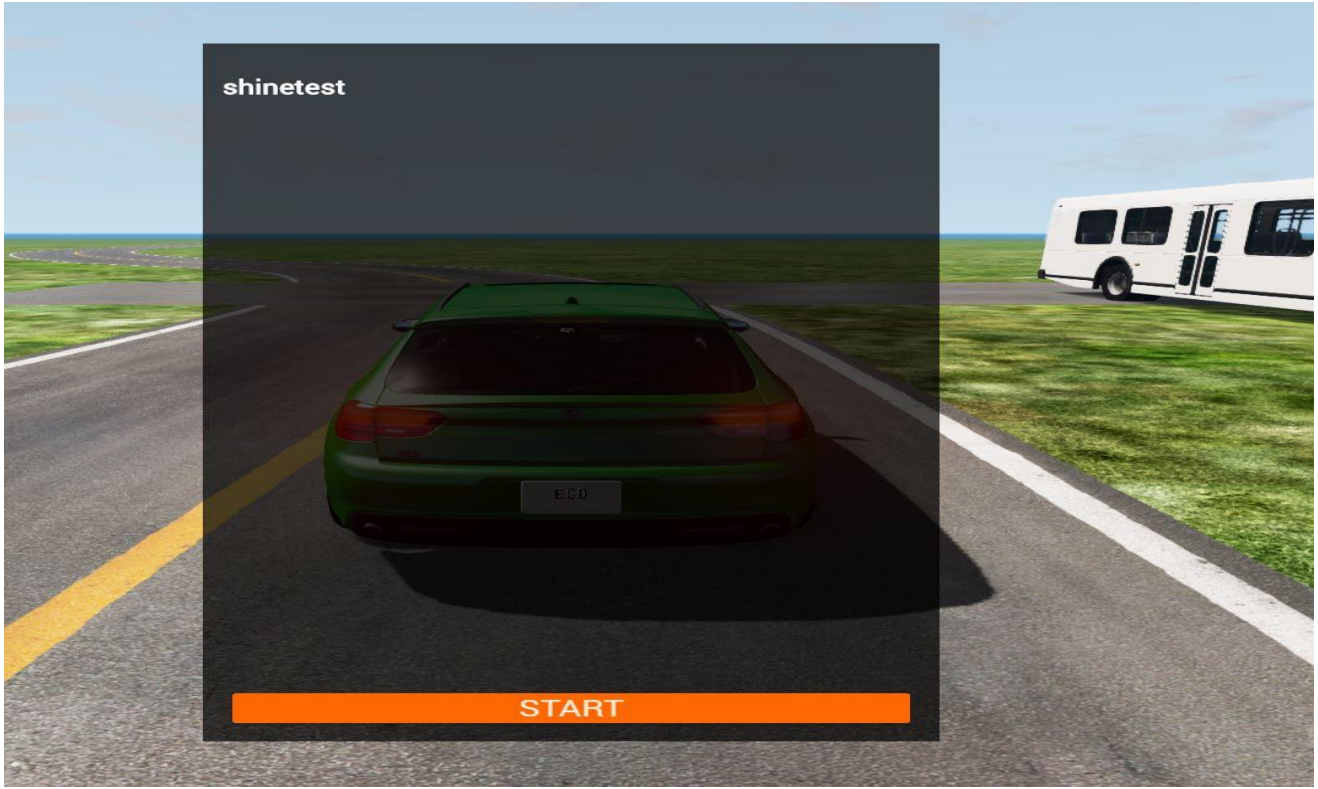


Figure 11. Simulator Loaded With ShineTest [11].

Figure 12. Simulator Loaded With ShineTest Result [12].

Conclusion

- By shinetest generator, I tried to implement the scenario factory by creating roads and placing Vehicle.
- Tracks are created using the given end points and visualized it.
- Obstacles are created as a parking bus.
- Successfully tested them in the virtual environment by the discussed paper in the meeting.

References

1. M. Klischat, E. I. Liu, F. Holtke and M. Althoff, "Scenario Factory: Creating Safety-Critical Traffic Scenarios for Automated Vehicles," *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, 2020, pp. 1-7, doi: 10.1109/ITSC45102.2020.9294629.
2. https://commonroad.in.tum.de/
3. https://www.geonames.org/
4. A.Gambi, M. Mueller, and G. Fraser, “Automatically testing self driving cars with search-based procedural content generation,” in Proc. of the 28th ACM SIGSOFT Int. Symposium on Software Testing and Analysis, 2019, pp. 318–328.
5. N. Kalra and S. M. Paddock, “Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?” Transp. Res. Part A: Policy Pract., vol. 94, pp. 182–193, 2016.
6. https://beamng.tech/
7. https://sumo.dlr.de/docs/index.html
8. https://github.com/shinejose0007/AutonomousCarTesting
9. https://gitlab.lrz.de/tum-cps/commonroad-io
10. https://commonroad-io.readthedocs.io/en/latest/
11. https://github.com/se2p/sbst-2021-tutorial/

Thank You...