

Multi-Modal Traffic Simulation Calibration and Integration with Real-Time Hardware in Loop Simulator

Vikhyat Kalra¹[\[https://orcid.org/0000-0002-5340-1719\]](https://orcid.org/0000-0002-5340-1719), Punit J
Tulpule¹ and Jacob A. Isaman¹

¹ The Ohio State University, USA
Kalra.50@buckleyemail.osu.edu, tulpule.3@osu.edu

Abstract

The ongoing research in intelligent transport systems and connected and automated vehicles, enabled by advancements in artificial intelligence, integrating traffic simulations has become an essential part of product/software development for the automotive industry. Nowadays, traffic simulations are used to mimic real-world environment scenarios for virtual testing of advanced transportation technologies. . With the increase in data collection methods for traffic flow, the calibration of the microscopic traffic simulations has emerged as an important research area. The underlying question in traffic modeling is how accurately simulations can mimic the real environment traffic flow conditions? This paper attempts to create a framework for microscopic traffic simulation calibration procedure which can be scaled for large networks. This paper makes the following major contributions. First, a calibration framework is proposed which harnesses the existing data set collected from The Ohio State University (OSU) campus bus service (CABS) busses using Global Positioning System (GPS) sensors to determine the traffic state in the real environment and create a microscopic traffic simulation. The traffic simulation is implemented for a section of the OSU campus ("Woody Hayes Drive") in an open-source traffic simulator – Simulation of Urban MObility (SUMO). The traffic flow generation is probabilistic to introduce variability between scenarios. The second contribution is the development of a communication interface between real-time dSpace ASM Hardware in Loop setup with SUMO to create a complete real-time simulation of urban environments for advanced driver assist systems (ADAS) virtual testing. A demonstration scenario is the Ohio State University campus network with traffic demand generated using the calibrated model from the first part of the work.

1 Introduction

Traffic simulations are widely used by city planners and government agencies to understand the movement of people and goods in a road network environment. With increased interest in ADAS and connected and autonomous vehicles (CAV)

technologies for intelligent transportation, traffic simulations have become an important part of automotive research. The development of ADAS and other connected vehicle fields requires millions of miles of testing to demonstrate improvements. Most often virtual simulation testing is a safer, more efficient and faster alternative to road testing[1]. For a realistic testing environment setup, traffic simulations play a very critical part. Traffic simulations consist of multiple modes of transportation (vehicles, pedestrians, scooters, etc.) interacting with a road network environment and infrastructure (e.g. traffic lights and signs). In a microscopic traffic simulation, each individual traffic participant is modeled as independent agent while ensuring the overall traffic flow matches the desired flow. The task of calibration of traffic simulation entitles the accurate vehicle flow information and vehicle interaction in a closed-loop road environment. Despite the wide research in the field of traffic simulations, the calibration approach used requires on-road traffic data which intern requires sensors infrastructure [2], such as, cameras, induction loop detectors, etc. This data may not be readily available due to cost and time. A university campus is a unique environment as it consists of cars, buses, utility vehicles, and foot traffic. The traffic patterns tend to follow a specific pattern that depends on class schedule and vacations. Besides, the OSU campus offers unique opportunities for further research related to smart cities. Hence, this study focuses on modeling multi-modal traffic on the OSU campus. The key challenge is that vehicle counts data is very limited due to lack of sensors. The calibration approach used in this paper is based on GPS locations available from CABS busses. This approach does not require vehicle counts data from infrastructure sensors. In this approach we first compute the average travel time at intersections using the GPS data and calibrate the model to match this travel time. The underlying assumption is that the travel time is an indicator of traffic flow rate. Calibrated traffic models alone are not sufficient to test ADS and ADAS safety. The vehicle under test (a.k.a. ego vehicle) should be controlled using the algorithm being tested. The fidelity of the ego vehicle models, and traffic simulator models are decided based on the validation level. However, establishing a seamless communication between vehicle simulators and traffic simulators is not always trivial as the models have different accuracies and time steps. In this paper we consider one such application of traffic models to Hardware in the Loop (HIL) testing of ADAS algorithms. A lot of research has been done in scenario-based testing of ADAS and CAVS [3]. The scenario-based testing consists of few vehicles, shorter time horizon and smaller distances. There are lot of ADAS testing software packages like MATLAB® ADT, dSpace ASM, CARLA, Roadrunner, etc. But no commercial or open-source available software has linked the traffic simulation with a vehicle simulator in a HIL setup [1]. In this paper the traffic simulation model calibrated for a section of the OSU campus is linked with the ADAS equipped vehicle model running on a HIL setup to demonstrate a closed loop virtual testing

environment. The challenge is due to difference in the vehicle dynamics model fidelities and differences in time steps between SUMO and dSpace ASM. The SUMO model can run at maximum of 10Hz, but the HIL setup must run at-least at 100Hz. Besides, the yaw rate of SUMO vehicle dynamics is not accurate enough to feed directly to the high Degree of Freedom (DoF) vehicle dynamics model of ASM. Hence a synchronization interface is developed to approximate vehicle positions. This paper is structured as follows. Section 2 of the paper describes the traffic model calibration process and results of calibration for a section of the OSU campus. In Section 3 we explain the process to integrate traffic simulation models with HIL simulator. Finally, in Section 4 we make concluding remarks.

2 Multi-Modal Traffic Calibration using Probe Data

The simulation structure in SUMO is represented by Figure 1. The three primary components in a sumo simulation configuration file are: vehicle demand, additional files (virtual detectors and other road infrastructure) and network (road network with links and junctions etc.). The additional files and the network components of the simulation are not the variables for traffic flow calibration. The calibration parameters in this paper are the vehicle flow(count) in the network. The car following model used is Intelligent Driver Model (IDM). The selection of IDM as the candidate for car following model is based on current literature which depicts its better performance in high density junction traffic scenario to match the real-world driving behavior. [4]

The road network selected is Woody Hayes Drive in the OSU campus. The selected road network map is shown in [Figure 1](#). The network consists of six intersections, out of which, five intersections are controlled by traffic signals. There are total twelve entry points for the vehicles in the road network. The primary reasons for selecting this section of the network are:

- Woody Hayes connect the major routes of CABS on OSU campus
- Woody Hayes Drive connects major department and student classes buildings thus it has the most footfall during office hours 8:00 a.m – 9:00 a.m (time considered in this simulation for traffic data)



Figure 1: Woody Hayes Drive Network

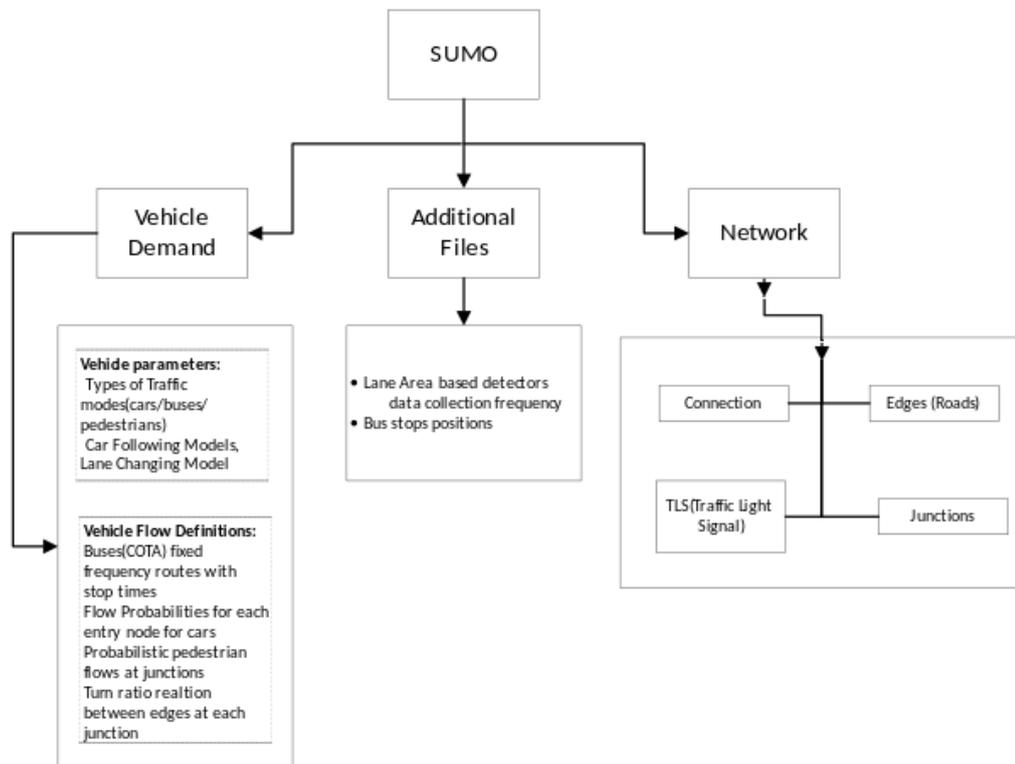


Figure 2: SUMO Traffic Simulation File Structure

The traffic simulation is multimodal with three vehicle classes - pedestrian, passenger cars and buses (CABS). The parameters used for these three modes are given in Table 1.

Table 1 Traffic Agents Parameters

Parameter	Pedestrian	Car	Bus
Length	0.215	4.3	12
Width	0.478	1.8	2.5
Height	1.719	1.5	3.4
Minimum Headway Gap (m)	0.25	2.5	2.5
Maximum Acceleration (m/s ²)	1.5	2.9	1.2
Maximum Deceleration (m/s ²)	2	7.5	4
Emergency Braking Deceleration (m/s ²)	5	9	7
Maximum Speed (km/h)	5.4	180	85

2.1 Traffic Data

Any traffic model calibration process has two key steps – 1) road network calibration, and 2) traffic flow calibration. Road network calibration requires information about road segments and junctions including number of lanes, road length, speed limits, stop signs etc. Open Street Maps (OSM) is an open-source data consisting of all the necessary information regarding roads and junctions. However, the map imported from the OSM may have errors, particularly at junctions the number of lanes may not be correct or turn only lanes may not be correctly assigned. These errors are corrected using the process depicted in [Figure 4](#). The corrections were validated by manually confirming against google maps street view. The other part of road network is traffic signal timings. The traffic signals assignment is done in SUMO using the traffic controller phase timings provided by the OSU Traffic and Transportation Management (TTM) Department and the phase timings are then inspected at the actual location to validate the controller data.

The vehicle flow data for three different modes are decided as per the following: -

- The CABS busses frequency and schedule are used to generate the flow(count) data. All the CABS busses are equipped with GPS logging devices. The data is available from 2011 to 2019. This data consists of GPS coordinates (latitude and longitude) and corresponding GPS timestamp. The data was filtered to extract this data of the busses on routes that pass through the selected region on campus. The data is not sampled consistently, meaning the sampling time varies depending on the number of available satellites and other uncertainties. Hence, another filter was implemented to discard inconsistently logged data. [Figure 3](#) shows the variability in successive data logs from GPS.

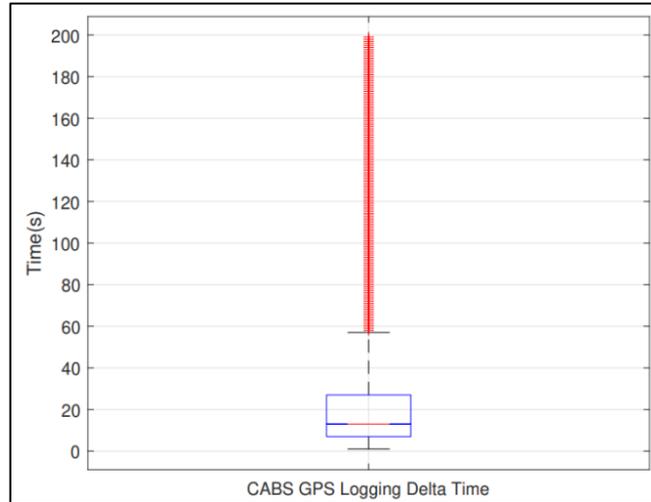


Figure 3: CABS GPS Timestamp delta distribution

- The traffic flow data was separately collected for three days which gives the number of vehicles for the vehicles at all five junctions in the network. The count data is used to generate turning ratios at each junction. *Figure 5* shows an example for turning ratios from the count data. The turning ratios are used as probability of turning in the SUMO simulation. For example, a vehicle coming from west direction in *Figure 5* as 30% probability of a left turn.
- Due to lack of pedestrian count data, the pedestrian flow is assumed to be constant at one pedestrian every 60 seconds at each crossing lane at the junctions. This frequency is used since the pedestrian walk green signal timing is fixed and it is activated upon pedestrians' request. The chosen frequency ensures that the pedestrian walk signal is requested in each cycle.

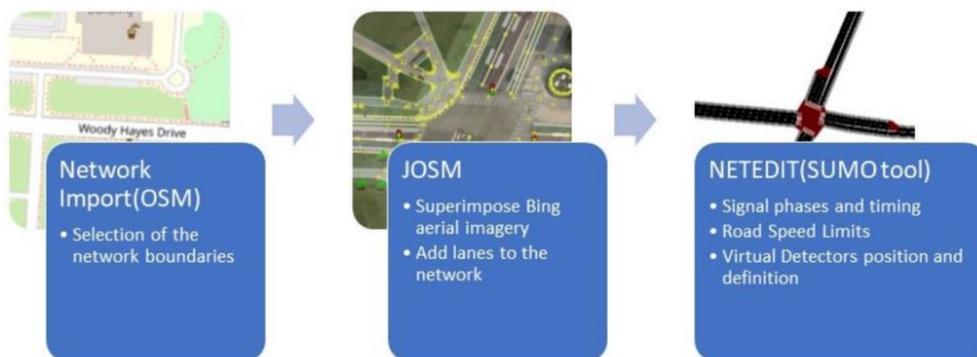


Figure 4: Network Correction Process

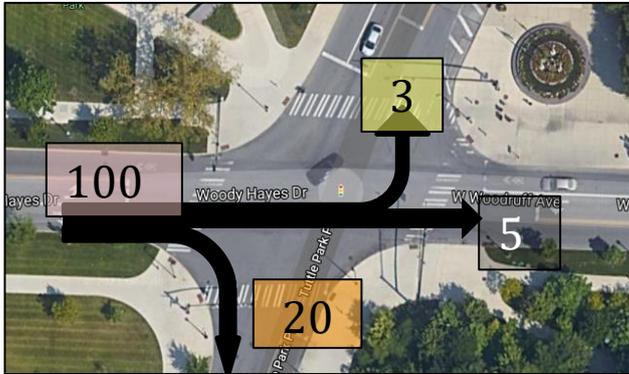


Figure 5: Turn Ratio Example

2.2 Vehicle Generation

SUMO has three packages to build the traffic in a network environment. The three packages are DFROUTER, JTRROUTER and DUAROUTER. In this work JTRROUTER is used to generate the vehicles. JTRROUTER requires turning ratios and vehicle flow probabilities from entry points in the network. The turning ratios are calculated from turn count data and doesn't change during a simulation run. The flow probabilities are used as a calibration parameter by the optimization algorithm.

2.3 Objective function and calibration parameters

The CABS GPS data is used as an indicator for the traffic density in the network. The entire network is divided into twelve sections. Six in West-East direction and six in East-West direction. A simple algorithm is developed which calculated the travel time in the network section as shown in [Figure 6](#).

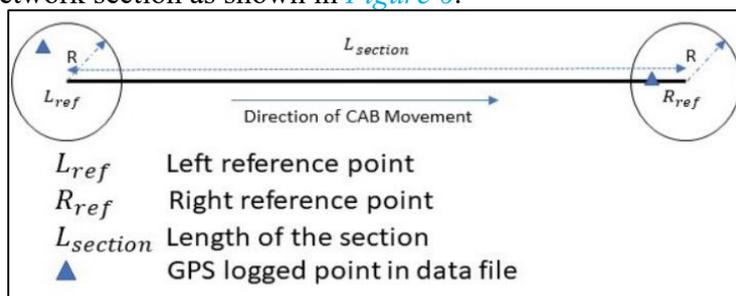


Figure 6: Network Section CABS Travel Time

GPS data has uncertainty in the position log. In this research we did not have access to the sensor information to quantify the uncertainty. Hence, as shown in [Figure 6](#), a

circle with radius R centered at the point of interest on the road is used as a region to find a GPS data point. The R used in the algorithm is 30m. Each section has two reference GPS coordinates, the start reference point (L_{ref}) and end section reference point (R_{ref}). The algorithm searches the GPS data file and if a data point is found within the circle around L_{ref} then the timestamp ($T_{L_{ref}}$) and position coordinates ($X_{L_{ref}}$) are stored in a memory. Then the algorithm finds the next timestamp ($T_{R_{ref}}$) when the GPS coordinates are in the circle centered at R_{ref} . The $T_{R_{ref}}$ and GPS position coordinates ($X_{R_{ref}}$) are stored in a separate memory buffer. To compute average travel time and speed across a junction, L_{ref} and R_{ref} are chosen on the road such that R_{ref} is upstream of the junction and L_{ref} immediately downstream of the junction. Timestamps of the CABS busses passing through these two points are logged using the algorithm described above. The time-stamps and corresponding GPS coordinates are used to compute average speed and average travel time.

$$\bar{V} = \frac{D_{GPS}}{\Delta T}$$

$$\bar{T}_{obs} = \frac{L_{section}}{\bar{V}}$$

Where $\Delta T = T_{R_{ref}} - T_{L_{ref}}$, $D_{GPS} = f(|X_{R_{ref}} - X_{L_{ref}}|)$, \bar{V} is the average velocity within the captured GPS points and \bar{T}_{obs} is average travel time in the predefined section length. f is function to convert geolocations to linear distance. The SUMO simulation environment is also divided into twelve segments and the travel time in simulation for vehicles is compared with the travel time from CABS GPS data analysis.

The objective function used is a point mean relative error between measured travel time $T_{obs,j}$ and simulated travel time $T_{sim,j}$ for a section j . and the objective is to minimize this error. The genetic algorithm in the MATLAB optimization toolbox is used to minimize the function.

$$\text{minimize}_{P_k} \sqrt{\sum_{j=1}^{12} \left(\frac{T_{obs,j} - T_{sim,j}}{T_{obs,j}} \right)^2}$$

Such that $T_{sim_j} = f_{SUMO}(P_k)$

$$P_k \in [0.05, 0.20] \quad \forall k = 1, 2, \dots, 12$$

where, k are the entry points of the network, j is the section number, P_k is the probability of vehicle generation at every one second, and f_{SUMO} is the SUMO traffic simulation model. The results of this simulation model are the of travel times for each section obtained from E3 detectors. The assumption here is that

$T_{sim,j} \propto \frac{1}{P_k}$. The flowchart in [Figure 7](#) shows the process for the optimization using GA.

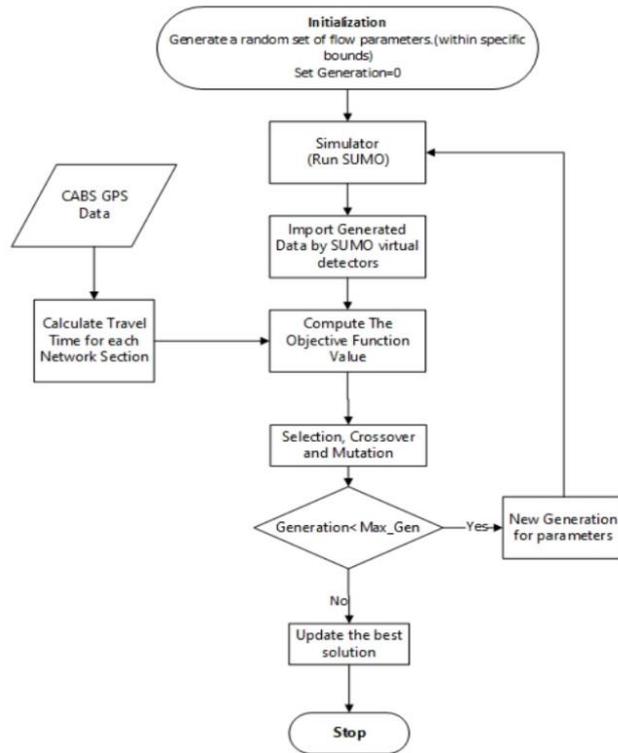


Figure 7: Genetic Algorithm Optimization Process

2.4 Calibration Results and Discussion

The maximum number of generations used for GA are ten. For each generation a population size of 150 is generated. Here each population refers to the calibration parameter P_k which is a vector of 12 entry probabilities in the network. From the [Figure 8](#) it can be observed that the objective function fitness value reaches steady state in the 4th generation and does not change from next generation. The [Figure 9](#) represents the average travel time comparison between travel time obtained from E3 detectors (T_{sim}), and the observed travel time from the CABS GPS data (T_{obs}) for each road section. In some of the sections the error is large and the primary reason for that is the inconsistency in the logged GPS data for CABS. If the delta time between CABS GPS timestamps can be reduced the uncertainty in the average travel time from calibration data set will reduce. The simulation average travel time doesn't have uncertainty because the virtual detectors can track the vehicle trajectory with a

delta timestamp of 1 s which is very accurate compared to actual data with mean time of 15 s.

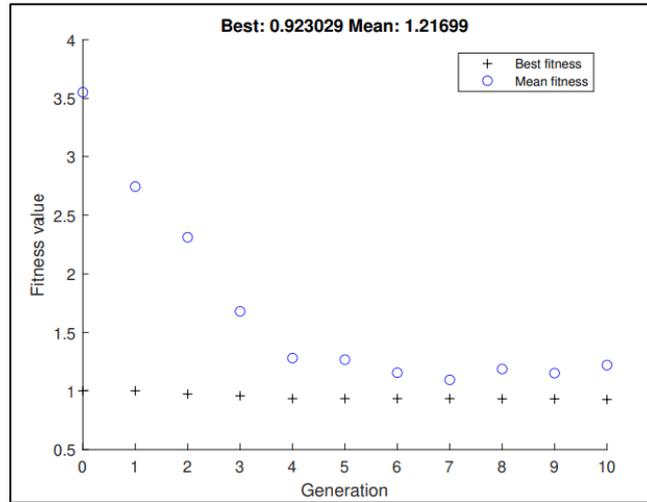


Figure 8: PMRE (Objective Function) Value Evolution

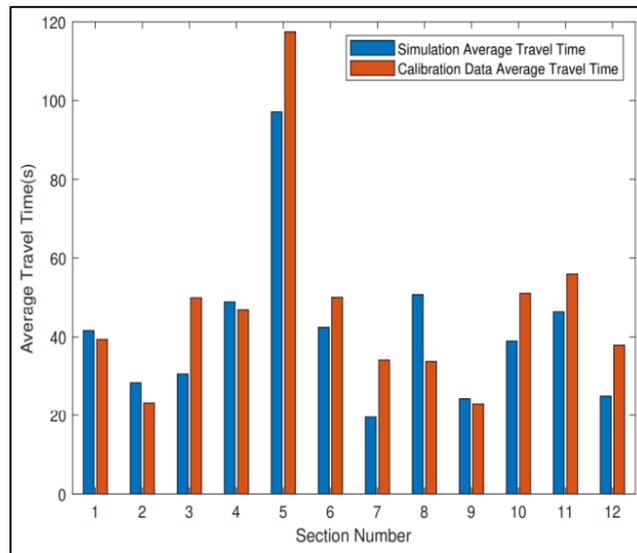


Figure 9: Average Travel Time Comparison

3 Hardware in Loop Simulator

In automotive research and development vehicle testing is done virtually using different x-In the Loop validation procedures. HIL setup is typically used for real time implementation of various electronic control units (ECU) along with

communication channels. Although HIL testing is not new in automotive applications, the implementation of HIL for ADS testing is not trivial due to their extremely complex Operational Design Domains (ODDs). Testing CAVs can be performed either in the real world with all the necessary hardware or parts of the environment could be simulated using virtual representation. The simulation-based approach is a much faster, safer, and efficient alternative compared to actual road testing[5]. The simulation for emulating the real environment involves a critical component of replicating the real traffic conditions in terms of road network and nearby vehicle behavior with respect to ego vehicle. In this paper a framework is created that integrates real time HIL setup(dSPACE) with large scale traffic simulator. The components for this framework are a high-fidelity simulation for ego vehicle dynamics compared to traffic vehicles and a communication interface with extrapolation algorithm to transfer the traffic information between traffic simulator and the HIL setup (dSPACE ASM)

3.1 Ego Vehicle Information Interface (EVI)

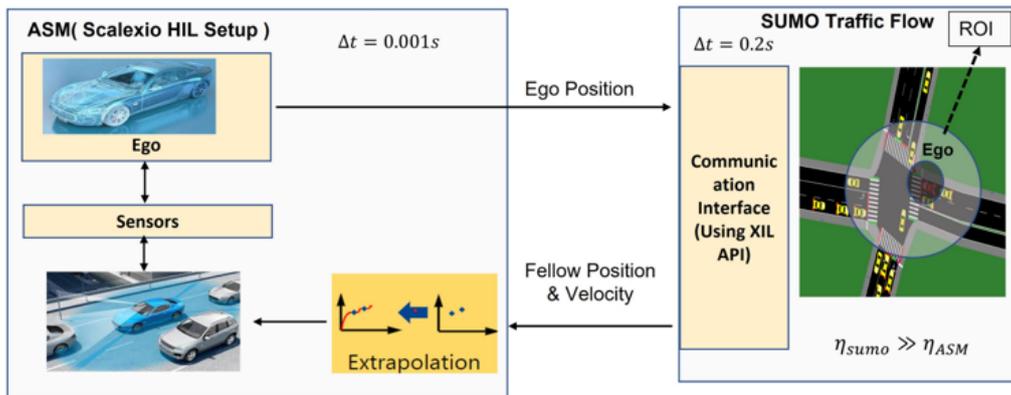


Figure 10: SUMO and dSpace ASM Co-Simulation Framework

The [Figure 10](#) shows how the information from the SUMO is passed to dSPACE ASM regarding the traffic vehicles and the feedback of ego vehicle speed and position from ASM to SUMO. The challenge in real time implementation is proper clock time synchronization between the two simulators. In SUMO TraCI is used with the command mentioned in [Table](#) . The SUMO simulator calculates the new position for traffic vehicles at every 0.2(5 Hz) and the ego vehicle dynamics model and ADS control is running on HIL at 0.001s (1000 Hz). The difference between the time update is handled using an extrapolation algorithm in the HIL setup which updates the position of traffic vehicles every 1ms till the next update from SUMO at every 0.2s is available. Besides, rendering visualization and using high fidelity

models for all the agents on the road is computationally inefficient and un-necessary. From ADS control perspective only fewer other significant traffic agents affect the perception and control of the ego vehicle. On the other hand, traffic simulation models simulate all the agents movements Hence, a region of interest (ROI) is calculated around the ego vehicle and the vehicles in this ROI are considered as nearby traffic vehicles. In this paper ROI with nearest 10 vehicles have been considered. Simultaneous to traffic vehicle information the traffic signal SPaT is also sent to dSPACE ASM. For SPaT only the ego vehicle nearest junction is considered as all the traffic vehicles are controlled by SUMO and do not need control command in the HIL simulator.

Table 2: Traci Commands for the EVI

TraCI Command	Information
traci.start()	Start the SUMO simulation
traci.simulationStep()	Progress simulation by one time step
traci.vehicle.getIDList()	IDs for all vehicle in SUMO scenario
traci.vehicle.getNextTLS('ego')	Get State Information for next traffic light in front of Ego
traci.vehicle.moveToXY()	Move Ego Vehicle to position specified by ASM
traci.vehicle.getPosition()	Get XY position for the fellow traffic in SUMO network
traci.vehicle.getLateralSpeed()	Get lateral speed in m/s t
traci.close()	Close the SUMO simulation
traci.vehicle.getAngle()	Get Yaw angle in degrees for traffic vehicles

3.2 Co-Simulation Scenario & Results Discussion

The scenario used in this paper is the calibrated traffic on Woody Hayes drive as shown in [Figure 12](#). This figure shows the ego vehicle position in ASM network and its corresponding position in SUMO is passed. The ego vehicle travelled from West to East crossing six intersections. The [Figure 11](#) is the representation of the real time implementation of the communication interface. At $t = 0$ simulation is started and

ego position is transferred to SUMO. Traffic vehicles positions are calculated and then transferred to dSpace ASM using an Ethernet interface. The SUMO and EVI execution time is around 150ms for the network used in this paper. The sumo step is paused till the clock time is 200ms and then next time step calculation starts in SUMO. As caching data to HIL takes time the writing of information about the fellow vehicles starts at 20ms before the indented clock time.

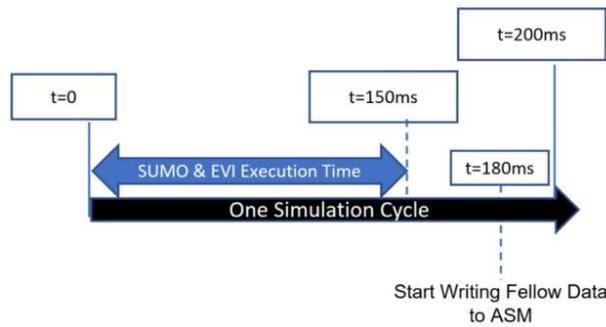


Figure 11: One Cycle of Simulation Synchronization: Repeats every 200ms

Figure 13 shows the extrapolation trajectory vs SUMO output for a fellow vehicle in the region of interest near ego vehicle. It can be observed that the extrapolated drifts from the actual path. The maximum deviation observed was 40 cm which can be reduced by reducing the 200ms SUMO time step to 100ms or lesser.

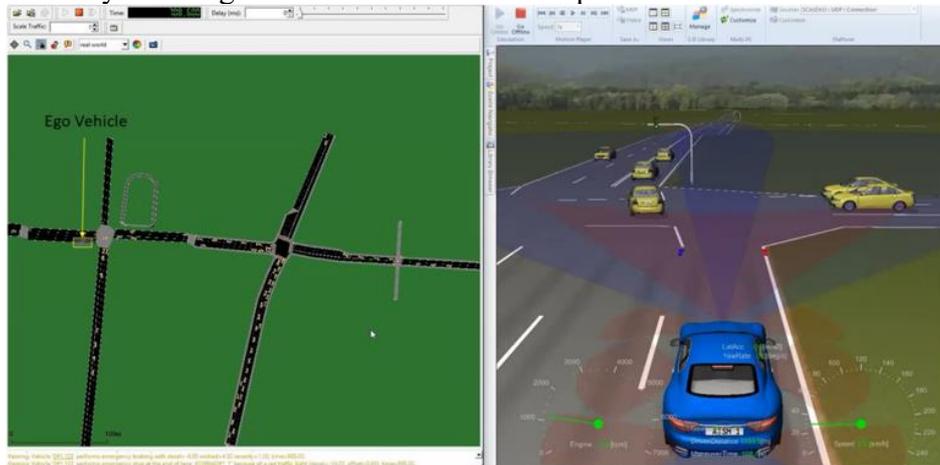


Figure 12: Co-Simulation Snapshot: ASM & SUMO

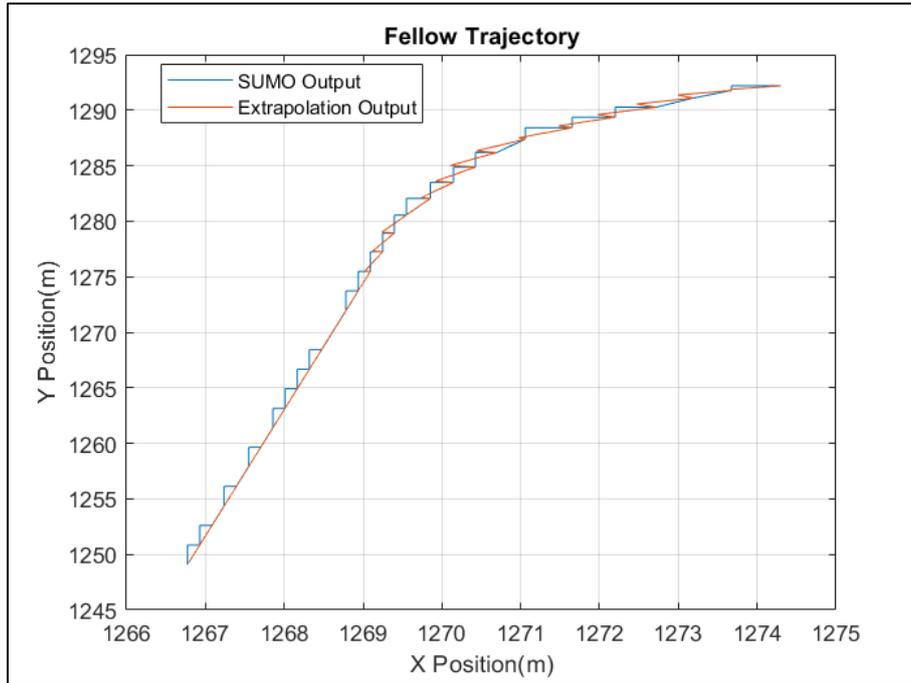


Figure 13: SUMO & Extrapolation Algorithm Fellow Trajectory Example

4 Conclusions

In this paper we presented an approach to calibrate the microscopic traffic simulation using limited dataset with genetic algorithm optimization approach. It is shown that the travel time across all the junctions could be approximately captured in the simulation using this approach. The benefit of this approach is that it does not require traffic count data from infrastructure sensors. To demonstrate a use case of traffic simulation models in testing ADS features, the calibrated traffic simulation model in SUMO is then combined in a framework with the dSpace ASM HIL simulation. The framework uses region of interest concept to transfer only the relevant fellow traffic information to HIL simulator. This gives us an opportunity to create large network of roads in SUMO and link them with high fidelity vehicle powertrain and ADAS simulations. The time synchronization problem is solved by using the extrapolation between the larger timestep of SUMO.

The presented work can be extended to including pedestrian vehicle class in the HIL simulation. Also, the framework can be extended to other xIL simulation testing platforms such as camera in the loop. The traffic calibration will be improved using the vehicle counts from the camera detection installed on OSU campus traffic signals.

5 Definitions/Abbreviations

ASM	Automotive Simulation Models(dSpace)
SUMO	Simulation for Urban Mobility
SPaT	Signal Phase and Timing
GA	Genetic Algorithm
HIL	Hardware in Loop

References

- [1] P. Tulpule, S. Midlam-Mohler, A. Karumanchi, and Y. Jin, "A Simulation Tool for Virtual Validation and Verification of Advanced Driver Assistance Systems," *SAE Tech. Pap.*, no. 2021, pp. 1–9, 2021, doi: 10.4271/2021-01-0865.
- [2] R. Dowling, A. Skabardonis, J. Halkias, G. McHale, and G. Zammit, "Guidelines for Calibration of Microsimulation Models: Framework and Applications:," <https://doi.org/10.3141/1876-01>, no. 1876, pp. 1–9, Jan. 2004, doi: 10.3141/1876-01.
- [3] H. Singh, S. Midlam-Mohler, and P. Tulpule, "Simulation Based Virtual Testing for Safety of ADAS Algorithms - Case Studies," *SAE Tech. Pap.*, no. 2021, pp. 1–15, 2021, doi: 10.4271/2021-01-0114.
- [4] L. Bieker-Walz, M. Behrisch, M. Junghans, and K. Gimm, "Evaluation of car-following-models at controlled intersections," *31st Annu. Eur. Simul. Model. Conf. 2017, ESM 2017*, no. Krauss, pp. 238–243, 2017.
- [5]. Hallerbach, Sven, et al. "Simulation-based identification of critical scenarios for cooperative and automated vehicles." *SAE International Journal of Connected and Automated Vehicles* 1.2018-01-1066 (2018): 93-106.