A Micro-simulation Framework for Studying CAVs Behavior and Control Utilizing a Traffic Simulator, Chassis Simulation, and a Shared Roadway Friction Database

Liming Gao, Satya Prasad Maddipatla, Craig Beal, Kshitij Jerath, Cindy Chen, Lorina Sinanaj, Hossein Haeri, Sean Brennan

Abstract—**The ability of connected and autonomous vehicles (CAVs) to share information such as road friction and geometry has the potential to improve the safety, capacity, and efficiency of roadway systems, and the study of these systems often necessitates synergistic investigation of the vehicle, traffic behavior, and road conditions. This paper presents a micro-simulation framework for studying CAVs behavior and control utilizing a traffic simulator, chassis simulation, and a shared roadway friction database. The simulation utilizes three levels of data representations: 1) a traffic representation that explains how vehicles interact with each other and follow location-specific rules of the road, 2) a vehicle dynamic representation of the Newtonian response of the vehicle to driver inputs interacting with the vehicle which in turn interacts with the pavement, and finally 3) a road surface representation that represents how friction of roadway changes with space and time. The interactions between these representations are mediated by a spatiotemporal database. The framework is demonstrated through a CAVs application example showing how the mapping of road friction enables advanced vehicle control by allowing the database-mediated preview of road friction. This framework extends readily to real-time implementation on actual CAVs systems, providing great potential for improving CAVs control performance and stability via database-mediated feedback systems, not only in simulation, but also in practice.**

I. Introduction

Research into the performance of connected and autonomous vehicles (CAVs) and intelligent traffic systems has been increasing in recent years due to the possibilities of improved performance, efficiency, and safety afforded by the sharing of information between vehicles and each other and/or infrastructure, and the control of vehicles from this information[1]. This paper is motivated by the study of using CAVs to share information such as road friction and geometry to improve the safety, capacity, and efficiency of roadway systems; this study necessitates synergistic investigation of the vehicle, traffic behavior, and road conditions with software tool integration and data frameworks that are not readily available within the research community.

The complex, large scale and highly dynamic nature of CAVs systems introduce significant challenges before efficient and reliable solutions and applications can be deployed [2]. Investigation of those problems through real-world experimentation with such systems is often infeasible due to the complexity and cost. In this context, simulation tools can be used, and it is desirable that these tools are accurate, easy to use, integrate well with real vehicle testing, and can perform computations quickly and verifiably. These tools range from traffic simulations, chassis dynamic simulations, map and traffic network interfaces, and data storage and retrieval systems.

For large-scale traffic systems, there are many techniques, theories, and platforms to simulate large numbers of vehicles such as continuum flow modeling, cellular models, and microscopic simulations where instances of each vehicle are simulated. And particularly when simulating sensor interaction with specific localized road conditions, microsimulations can provide near-real insight both on algorithm behavior and data management. One of the most widely used commercial traffic simulators is AIMSUN (Advanced Interactive Microscopic Simulator for Urban and non-urban Networks) modeling package [3]. It is used to model traffic on road networks, which gives the ability to generate vehicle activity following traffic management strategies with alternative traffic scenarios, such as lane changing, car-following model, and so on. Similar to AIMSUN, there are several other traffic simulators, such as VISSIM [4], SUMO[5], TraffSim[6], and MovSim[7] with very similar interfaces and behavior prediction capabilities.

However, the study of CAVs systems requires modeling the interaction between vehicular traffic, vehicle dynamics, and the road surface. Unfortunately, the aforementioned traffic simulations do not include vehicle dynamics such as tire forces, road-tire-contact, friction values of the road, driving/braking torques, and desired Newtonian dynamics for the vehicle

Liming Gao is a graduate student in Mechanical Engineering at The Pennsylvania State University, University Park, PA 16803, USA (e-mail: lug358@ psu.edu).

Satya Prasad Maddipatla is a graduate student in Mechanical Engineering at The Pennsylvania State University, University Park, PA 16803, USA (e-mail: szm888@psu.edu).

Craig Beal is with Department of Mechanical Engineering, Bucknell University, Lewisburg, PA 17837, USA (e-mail: cbeal@bucknell.edu).

Kshitij Jerath is with Department of Mechanical Engineering, University of Massachusetts Lowell, Lowell, MA 01854, USA (e-mail: Kshitij_Jerath@uml.edu).

Cindy Chen is with Department of Computer Science, University of Massachusetts Lowell, Lowell, MA 01854, USA (e-mail: cindy_chen@uml.edu).

Sean Brennan is with the Department of Mechanical Engineering, The Pennsylvania State University, University Park, PA 16803, USA (e-mail: snb10@psu.edu). IEEE Member.

maneuvers. Vehicle chassis simulation tools, such as Simulink, CarSim, CarMarker, veDYNA can have the exceptional capability for vehicle dynamic simulation, interaction with road texture, and control strategy design and validation. However, those tools can be limited in their ability to scale when using these to study traffic, chassis behavior, pavement interaction, and individual sensor interactions across hundreds or more vehicles interacting simultaneously, as in traffic simulations.

An example of a real-world problem requiring such a tool would be the estimation of localized friction changes in a traffic network from data streaming from CAVs wheel-force sensors. A brute-force method would be to copy individual vehicle chassis simulations into one "traffic" simulation tool many hundreds or thousands of times – one simulation instance per vehicle. This imposes significant challenges to real-time implementation and memory management of vehicle states and deep algorithmic challenges in representing the time-varying interaction of vehicles with each other based on spatial position. In addition, managing vehicle behavioral rules as a function of location within a road network to correctly represent rules-of-the-road is challenging. Location-specific behavior changes would include traffic light start/stop behaviors and rules, merging traffic in interstates, changing speed conditions and zone-setting, congestion management devices, etc. In copying a vehicle chassis code repeatedly, one realizes the challenges in essentially programming both the traffic simulation and road-network geometry relationships into individual instances of vehicle chassis simulations. Nonetheless, an example of small-scale multi-vehicle simulations exist incorporating chassis behavior; for example, MOBATSim simulates CAVs using Simulink by duplicating vehicle models in Simulink, but only small scales are supported in this method and the traffic interaction is not flexible to diverse road networks [8].

A solution to this challenge is to separate the traffic and chassis simulation components by using appropriate software for each. This paper presents an example of such integration within a CAVs simulation process that utilizes three levels of data representations: 1) a traffic representation that explains how vehicles interact with each other and how each follows location-specific rules of the road, 2) a vehicle dynamic representation of the Newtonian response of the vehicle to driver inputs interacting with the vehicle which in turn is interacting with the pavement, and finally 3) a road surface texture representation that represents how friction of roadway changes with space and time or even with vehicle traversals atop the road. The interactions between these representations are mediated by spatiotemporal databases merge and archive data between traffic simulations, vehicle dynamic simulations, and roadway friction grids.

The use of the proposed framework in this paper is demonstrated through a CAVs application example showing how the mapping of road friction enables advanced vehicle control by allowing the database-mediated preview of road friction. The remainder of this paper is organized as follows: Section II discusses the architecture of the proposed micro-simulation framework. Section III shows the application example. Finally, a conclusion section summarizes the main results of the work.

II. ARCHITECTURE OF THE MICRO-SIMULATION FRAMEWORK

A general overview of the architecture of the simulation framework is shown in Fig. 1. In the simulation framework, a shared roadway spatiotemporal database is employed to merge and archive data between the components. These include lane-level road network maps, friction grids of lane surface texture, a traffic simulator generating vehicle activities include vehicle trajectory information, and vehicle dynamic simulations generating the motion states and road friction coefficient estimation. Each of the components is explained in the remaining subsections.

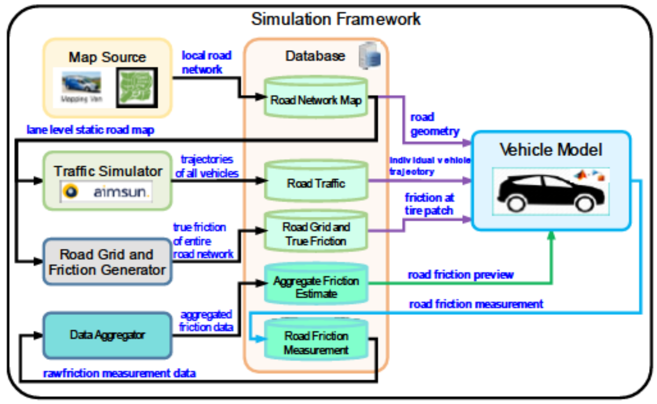


Figure 1. Data flow of the simlation framework showing the connectivity relationships between database, road maps, traffic simulator, and vehicle chassis simulations.

A. *Road Network Description*

Road maps are essential information in the simulation. This paper proposes a road map representation schema based on Lanelets [9] and OpenDRIVE[10]. Since the difference of map demands for different traffic applications, the road maps schema comprise three-level granularities: sections, lanes, and physical objects. Fig. 2 shows a map example with this data schema.

- Section-level: The basic geometry of each road section is defined by a reference line, and other road shapes such as superelevation, grade, and curvature are associated with the reference line. The topology relationship of sections is represented as a graph, where a road section is an edge and an intersection is regarded as a vertex. This level can be used as the road level map query and navigation.
- Lane-level: The lane-level representations are necessary for traffic scenarios such as lane-keeping, changing, and merging. Also, within the intersection, the connectivity paths are described by lane elements topology. Each lane within a road section is represented as a separate element, whose geometry is characterized by a lane centerline (or a rough guideline) and its left and right bound. The lane level topology relations are defined as the same as Lanelets [9].
- Physical object level: There are two categories of physical objects: regulatory objects and non-regulatory objects. The former consists of traffic signs, traffic lights, and traffic rules, and the latter includes bridge, tunnel, tree, barrier, and so on. Each object is associated with a section or lane element. In this research, we are particularly interested in

the road objects, for example, bridges, where surface friction feature is different from the surrounding region.

The lane level road map information was mapped using an Instrumented Vehicle (IV). The map data are stored in the road network map relations of the shared database.

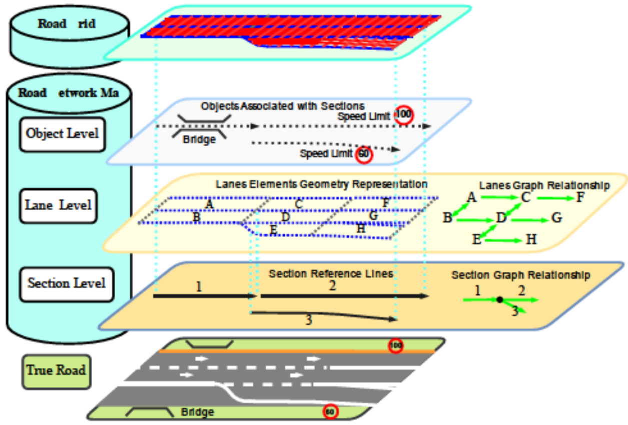


Figure 2. Map example for a highway road with exit and the resulting four levels map data representations.

B. Road Surface Friction Grid

To investigate the interaction between vehicle tires and the pavement, it is necessary to numerically describe the road surface feature. Thus, the road grids (mesh) are utilized to represent the distributed scalar properties of the road surface on a centimeter-level spatial scale. The grids are generated for each lane element based on the lane centerline, whose idea is similar to OpenCRG [11]. Each grid cell contains information about the road surface, such as elevation, friction, temperature, color, and so on. In this way, a microscopic view of the road surface can be provided.

This research focuses on the friction attributes, which has a significant influence on the transmission of longitudinal and lateral force between vehicle and road and thereby on the driving stability and safety[12]. True friction characteristics distributions are generated synthetically to represent deliberate operating conditions and disturbances. For example, we use friction coefficient values 0.7 to 1.0 to describe the dry road condition, 0.3 to 0.6 for wet roads, and 0.1 to 0.3 for snow-covered roads.

All the data are managed by the road grid and true friction relations in the shared database. In the database, the schema of the grid is built referring to the road lane element. Thus, the road grid can also be taken as the fourth-level granularity of the road map. Fig. 2 shows a map example with this data schema. Moreover, a time attribute is associated with each grid in the schema, so that the variation of road surface condition with time can be described. With this data model, a user can test any desired road feature distribution or real road measurement data.

Generally, to test the tire-based estimation of road varying friction, the grid resolution is required to be finer than a tire's contact patch. Consequently, the number of grids is quite large for a road and there is a need to represent the friction distribution compactly and effectively. More compact representations of friction can be realized by clustering the friction grid into regions defining the friction transition boundaries for similar friction clusters within the roadway database. Moreover, from the numerical aspects, low precision data is sufficient for friction value.

C. Traffic Simulator

In the simulation framework, an AIMSUN traffic simulator is employed to generate vehicle trajectories and states following traffic management strategies with desired traffic scenarios. The road network map for the traffic simulation is queried from the shared database and import to AIMSUN through the RoadXML[13] format file. Diverse traffic activities can be simulated by tailoring the parameters in AIMSUN such as traffic volume, vehicle types, traffic speed, car-following model, etc.

The generated vehicle data are streamed into the road traffic relations of the shared database in real-time via a Robot Operating System (ROS) middleware layer. The AIMSUN simulation is connected to ROS through User Datagram Protocol (UDP) socket and built-in API functions. And then the shared database is connected to the ROS layer via the Python Database API. This middleware bridges the data flow gap between traffic simulator and database.

D. Vehicle Dynamic Model and Friction Measurement

Vehicle dynamic simulations are created to predict how a vehicle following a trajectory would change in behavior in response to friction variations and thereby be able to estimate changes in road conditions. A general structure of a vehicle model in this simulation framework is shown in Fig. 3. The vehicle model comprises vehicle dynamics, driver model and controller, road friction estimator, and database query blocks. The model is used to conduct vehicle chassis simulation in response to road friction variation.

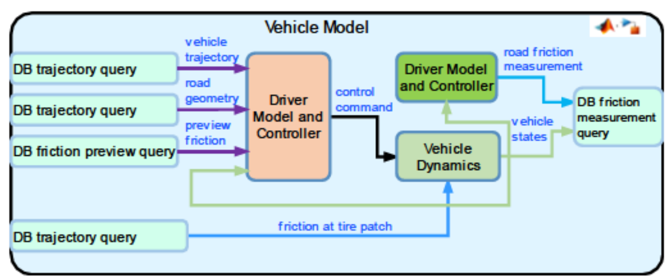


Figure 3. The structure of Vehicle Model in the simulation framework.

In this paper, the vehicle chassis dynamic model is built using Simulink. In the model, a driver model is built based on the assumption that a driver intends to look ahead with a 1 to 2 seconds preview horizon when tracking a desired trajectory. The vehicle states are projected ahead of the vehicle over the horizon to produce a predicted lane tracking error, and the vehicle steering is controlled to minimize this look-ahead error. The throttle/brake is also controlled to accelerate/decelerate the vehicle to follow the desired speed. The produced throttle/brake and steering angle control commands from the driver model are input to a Newtonian bicycle chassis model with a coupled brush tire model to predict the vehicle motion behavior. A driving assist controller was also included to be activated to intervene in the driver's commands to prevent the vehicle from exiting the safe handing constraint, similar to a vehicle stability control system. A rapid road friction estimator is integrated into the model based on the work by Beal [14]; in this, the surface friction is estimated through direct model inversion using the independent measurements of the left- and right-side front steering torques.

When the simulation is running, the vehicle exchanges data with the shared database through query blocks. For each vehicle, the "trajectory query" block queries the database to receive data produced by the traffic simulator for that specific vehicle; this trajectory is often not physically valid and includes, for example, instantaneous lane change events. This traffic simulator trajectory therefore is treated as the desired trajectory for the vehicle's driving algorithm. When the vehicle is tracking the trajectory, the vehicle is subject as well to road curvature and road friction, and this information is retrieved from the database via the "road geometry query" block and "tire friction query" block. These feed the road geometry and friction information to the vehicle dynamic model respectively, so that the vehicle can interact with the road surface in simulated real-time. The "friction measurement query" block collects the simulated vehicle states, true friction values encountered by each vehicle's tires, and sensor-corrupted simulated friction measurements as if each vehicle's sensors were operating. This information is then streamed into the shared database in real-time via a database cursor as if the vehicle is operating in a V2x system.

The aggregated road friction estimate results are organized and extracted from the large-scale raw measurement data via a data aggregator and then populated into the aggregate friction estimate relations of the database. The aggregated data can be shared with vehicles as a road friction preview to enable an advanced vehicle control.

E. Data Management via a Relational Database

In this simulation framework, large quantities of data need to be managed including storage of diverse data types, querying data by location or other attributes, sorting data in space and time, and delivering or receiving data from separate different software. For example, if the simulation runs at 100Hz, then it produces 360 million rows of data for 1000 vehicles just in 1 hour. Conventional file systems are poorly suited to manage this amount of heterogeneous data; thus, we employ a PostgreSQL spatiotemporal Relational DataBase Management Systems (RDBMSs) with a common data schema for the heterogeneous data. The schema includes: road network map, road friction grids, road traffic, road friction measurement, and aggregate friction information.

With the scheme, all types of information are logically organized as relations. Additionally, several common SQL based database middleware routines are developed in ROS, MATLAB, and Python to bridge the data flow gap between different software. Furthermore, the shared database can be accessed by different users simultaneously, allowing multiple parallel simulated vehicles or groups of vehicles to share information. Therefore, with this simulation framework, users can generate various traffic scenarios efficiently to study the CAVs behavior and control with customized data representation, vehicle dynamics, driver models, and controllers, especially the control strategy with a database in the loop, without considering the heterogeneity of information types.

III. Illustrative Case Study

This work presents two experiments to demonstrate the capability and application of the simulation framework: 1) the first experiment consists of road surface friction coefficient estimation for the double-lane 1-km highway segment; 2) the second experiment demonstrates vehicle trajectory-keeping through longitudinal control using the database-mediated preview of the road friction condition.

A. Road Friction Mapping

To assess the friction sensor and road friction mapping capability through CAVs, we simulated a highway traffic scenario for 35 minutes with highway flow volumes calibrated to PennDOT reported 2019 Average Daily Traffic (ADT) values for typical afternoon traffic of interstate I-99[15]. During this time, 816 vehicles passed through one direction, or two lanes, with a specific time-invariant road friction distribution. In this traffic scenario, each vehicle measured the friction under road-tire contact points when tracking their trajectory defined by the traffic simulation. The simulation framework with a simple processor can currently simulate vehicle trajectories at approximately 10 times faster than real-time with centimeter-level vehicle contact patch interactions and 100Hz sampling frequency.

Fig. 4 shows an example from one vehicle, and one tire on that vehicle: the estimated friction and true friction at the contact point of the front left tire. Here, the term "station" refers to the path length measured from the start of the road segment in each lane, indicating spatially where the vehicle has traveled. Intentional friction transitions were added at three bridge locations within this road, causing a severe decrease in friction, a slight increase in friction, and a moderate decrease in friction for bridges 1, 2, and 3 respectively. From the plot, we can see that the estimated results match with the actual true data well, demonstrating that the simulated friction estimator works well in detecting road friction, as expected.

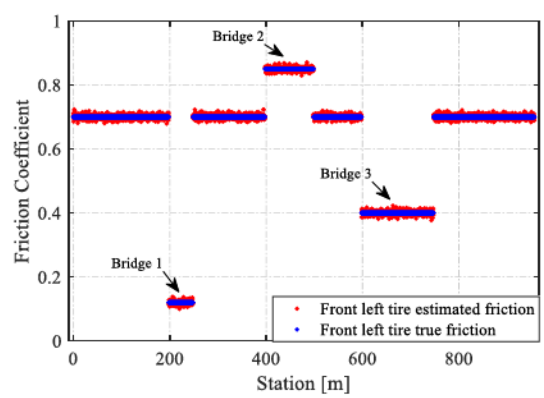


Figure 4. The actual friction and estimated friction at the contact point of the front left tire of a vehicle when tracking a trajectory on the road, versus station coordinate, e.g. the distance the vehicle has traveled.

Fig. 5 shows the friction estimation results of all four tires for a situation where the right side of the road has less friction than the left side. Each tire is assumed to measure the road friction coefficient independently, but a single passenger vehicle is unable to obtain the whole map road friction distribution. The advantage of the shared database is that it can collect and aggregate the measurement data from all the vehicles in the traffic flow, including samples between typical tire paths when vehicles change lanes or depart slightly from the typical path. With the shared database, the friction distribution of the road can be obtained shown as Fig. 6 which indicates that the estimation matches well with the true data, and the areas with friction variation are recognized clearly. By sharing their information to the connected database, the identified road friction distribution can be known by all the other vehicles to control their respective vehicle. The use of this group-shared information is demonstrated in the next experiment.

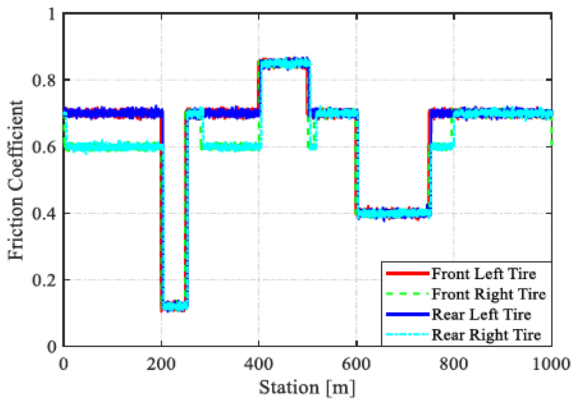


Figure 5. The estimated friction at all four contact points of four tires of a vehicle when tracking a trajectory on a road with spatial friction variation. In this simulation, the vehicle trajectory is such that the vehicle departs the road slightly to the right at moments, causing the right side of the vehicle to occasionally have less friction.

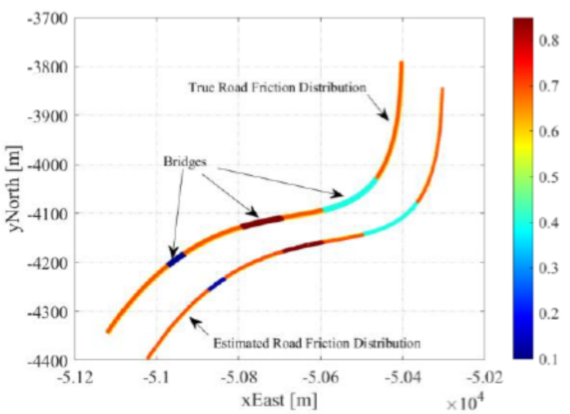


Figure 6. True road friction distribution (top) and estimated friction distribution (bottom) using database aggregated information from traffic simulated for 35 minutes of highway usage, with 816 vehicles traversing this location. (Note: for side-by-side comparison, the true friction data is offset northwest by 100m.)

B. *Trajectories Tracking Assistant Control with the Friction Coefficient Preview*

In this experiment, we consider the problem where a vehicle is tracking a pre-planned space trajectory at a constant speed along a highway where there are severe friction changes during curves. With a steady-state assumption, a vehicle will start sliding when $a_y \geq \mu g$ where μ is the road friction coefficient and g is the gravitational constant. The steady-state lateral acceleration of a vehicle when negotiating a constant-radius curve is:

$$a_y = U^2 / R \tag{1}$$

where the R is the radius of the curve and U is the forward velocity. The maximum allowable speed at a specific location in the curve is approximated as:

$$U_{max} = \sqrt{\mu g R} \tag{2}$$

In open-loop driving with the vehicle on a low-friction curve, drivers are often unable to stabilize the vehicle. Fig. 7 shows the vehicle's friction at each location, as well as the friction preview results for the friction condition 120m ahead. With the preview of friction and road curvature, a speed controller can plan the velocity profile to prevent the vehicle from excess speed in a curve and thereby preventing poor trajectory tracking or even road departure.

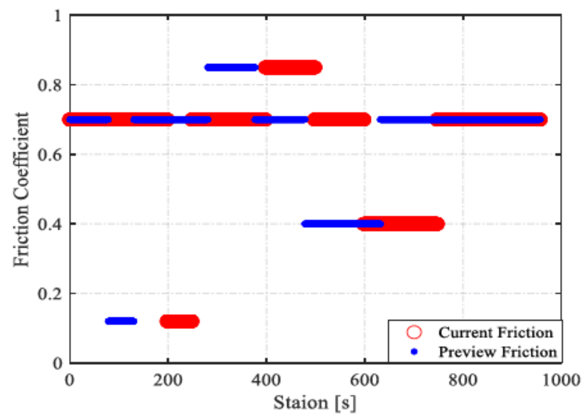


Figure 7. Friction preview ahead of 120m through aggregated friction data in the database, shared ahead of the vehicle tracking a trajectory on the road in Fig. 6.

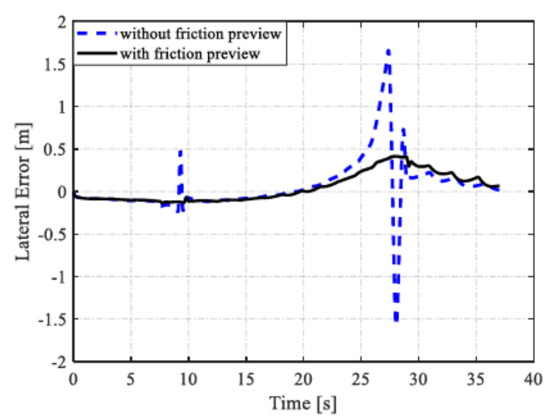
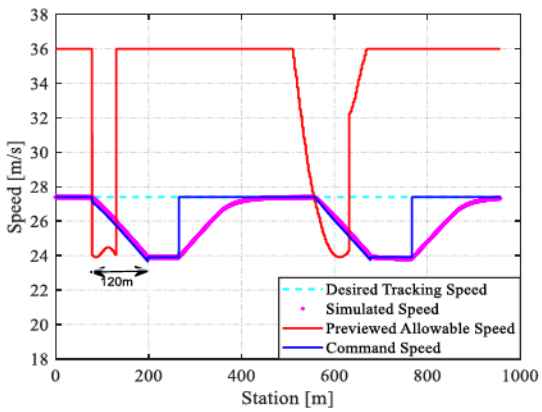


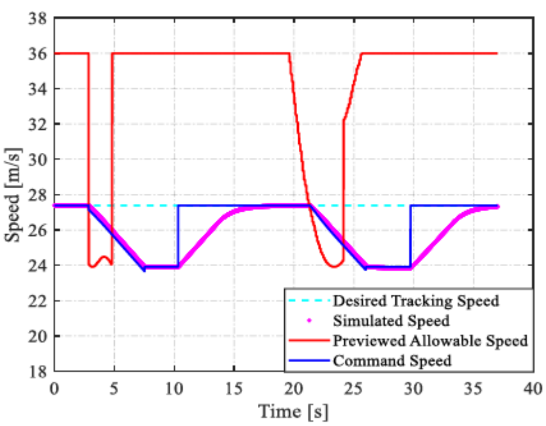
Figure 8. Lateral tracking error for the situation of Fig. 9. The difference between the driving control with and without friction preview is evident, showing that vehicle can track the desired trajectory with much less lateral error with friction preview.

Fig. 9 depicts the vehicle speed profile with the preview of friction to a vehicle coming from a database, causing the vehicle to anticipate necessary changes in allowable curve-keeping speed. Once the desired speed exceeds the anticipated allowable speed, the vehicle starts to decelerate constantly to the allowable speed and passes through the low friction area with reduced speed. The vehicle then accelerates smoothly to the desired speed once it leaves the low friction area.

From Fig. 8, the lateral tracking error for the situation with and without friction preview is evident, showing that the vehicle can track the desired trajectory with much less lateral error with friction preview. In the low friction region, large deviations in lane-keeping occur without preview-based velocity control. Note that the longitudinal and lateral controllers are not integrated in this experiment for the optimal utilization of preview information; this integration could be achieved via model-predictive control which is an active area of study for many researchers.



(a) Effect of preview, shown vs space



(b) Effect of preview, shown vs time

Figure 9. Vehicle speed profile with the preview of friction to a vehicle coming from a database, causing the vehicle to anticipate necessary changes in allowable curve-keeping speed. With the preview of friction and road geometry data, the allowable speed can be foreseen, enabling tracking.

IV. Conclusion and Future Work

The paper presents a micro-simulation framework that combines traffic simulations, chassis dynamic simulations, and databases of road friction and trajectory information. These are integrated in this work to demonstrate fleet-wide estimation and control strategies across large numbers of CAVs. The goal is to present a tool and framework for users to study the CAVs behavior and control with customized data representation, vehicle dynamics, driver models, and controllers. A particular challenge and research area are control strategies with a database in the loop.

This framework was shown to be faster than real-time in execution - roughly 6 times faster than real-time on an i7-7700K-CPU 32GB-RAM PC with native MATLAB and Simulink code, a speed suggesting that the framework could extend to real-time implementation on actual CAVs systems. This provides great potential for studying and improving CAVs control performance and stability via database-mediated feedback systems in simulation and practice. The results also demonstrate the potential of database-mediated CAVs systems not only for the research community but also for road operators.

Acknowledgment

This material is based upon work supported by the National Science Foundation under grant no. CNS-1932509 "CPS: Medium: Collaborative Research: Automated Discovery of Data Validity for Safety-Critical Feedback Control in a Population of Connected Vehicles".

References

[1] S. E. Shladover, "Connected and automated vehicle systems: Introduction and overview," *J. Intell. Transp. Syst. Technol. Planning, Oper.*, vol. 22, no. 3, pp. 190–200, 2018, doi: 10.1080/15472450.2017.1336053.

[2] M. Rondinone *et al.*, "ITETRIS: A modular simulation platform for the large scale evaluation of cooperative ITS applications," *Simul. Model. Pract. Theory*, vol. 34, pp. 99–125, 2013, doi: 10.1016/j.simpat.2013.01.007.

[3] M. Y. Madi, "Investigating and Calibrating the Dynamics of Vehicles in Traffic Micro-simulations Models," *Transp. Res. Procedia*, vol. 14, pp. 1782–1791, 2016, doi: 10.1016/j.trpro.2016.05.144.

[4] M. Fellendorf and P. Vortisch, "Microscopic traffic flow simulator VISSIM," in *Fundamentals of traffic simulation*, New York: Springer, 2010, pp. 63–93.

[5] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent Development and Applications of {SUMO - Simulation of Urban MObility}," *Int. J. Adv. Syst. Meas.*, vol. 5, no. 3, pp. 128–138, 2012, [Online]. Available: http://elib.dlr.de/80483/.

[6] C. Backfrieder, C. F. Mecklenbräuker, and G. Ostermayer, "TraffSim - A traffic simulator for investigating benefits ensuing from intelligent traffic management," in *Proceedings - UKSim-AMSS 7th European Modelling Symposium on Computer Modelling and Simulation, EMS 2013*, 2013, pp. 451–456, doi: 10.1109/EMS.2013.76.

[7] M. Treiber and A. Kesting, "An open-source microscopic traffic simulator," *IEEE Intell. Transp. Syst. Mag.*, vol. 2, no. 3, pp. 6–13, Sep. 2010, doi: 10.1109/MITS.2010.939208.

[8] M. Saraoglu, A. Morozov, and K. Janschek, "MOBATSim: MOdel-Based Autonomous Traffic Simulation Framework for Fault-Error-Failure Chain Analysis," *IFAC-PapersOnLine*, vol. 52, no. 8, pp. 397–402, 2019, doi: 10.1016/j.ifacol.2019.08.077.

[9] P. Bender, J. Ziegler, and C. Stiller, "Lanelets: Efficient map representation for autonomous driving," *IEEE Intell. Veh. Symp. Proc.*, no. Iv, pp. 420–425, 2014, doi: 10.1109/IVS.2014.6856487.

[10] M. Dupuis, M. Strobl, and H. Grezlikowski, "OpenDRIVE 2010 and Beyond – Status and Future of the de facto Standard for the Description of Road Networks," *Driv. Simul. Conf. 2010 Eur.*, pp. 231–242, 2012.

[11] VIRES Simulationstechnologie GmbH, "OpenCRG Homepage," 2020. http://www.opencrg.org/ (accessed Sep. 20, 2020).

[12] T. Weber, P. Driesch, and D. Schramm, "Introducing Road Surface Conditions into a Microscopic Traffic Simulation," in *SUMO User Conference*, 2019.

[13] P. Ducloux and G. Millet, "RoadXML: Format specification RoadXML 2.4.1 [RoadXML 2.4.1] Road Network Description," pp. 1–93, 2009.

[14] C. E. Beal, "Rapid Road Friction Estimation using Independent Left/Right Steering Torque Measurements," *Veh. Syst. Dyn.*, 2019, doi: 10.1080/00423114.2019.1580377.

[15] PennDOT, "Traffic Data Report 2019," 2020. https://www.penndot.gov/ProjectAndPrograms/Planning/TrafficInformation/Pages/2019-Pennsylvania-Traffic-Data.aspx (accessed Sep. 14, 2020).