2018 21st International Conference on Intelligent Transportation Systems (ITSC)
Maui, Hawaii, USA, November 4-7, 2018

Simulation of Heavy-Duty Vehicles in Platooning Scenarios

Namdoo Kim, Dominik Karbowski, Jongryeol Jeong, and Aymeric Rousseau
Energy Systems Division, Argonne National Laboratory, Lemont, IL, USA

***Abstract*—Connectivity and automation provide the potential to use information about the environment and future driving to minimize energy consumption. In order to achieve this goal, the designers of control strategies need to simulate a wide range of driving situations that can interact with other vehicles and the infrastructure to account for the specific powertrain characteristics of each automated vehicle.**

We present here a simulation framework called RoadRunner, which aims to facilitate the design of powertrain-aware eco-driving algorithms and a better understanding of the interactions between automation and powertrain technology. RoadRunner allows users to simulate both powertrain and longitudinal dynamics within a simulated environment. The user defines the scenario to be simulated by providing a route profile, intersection control types, number of vehicles, vehicle class, powertrain technology, connectivity, and automation level. RoadRunner then builds a Simulink diagram of the scenario, including the information flows between vehicles, and between vehicles and the road. After the simulation, the user can analyze the driving, component operations, and detailed energy consumption rates for each simulated vehicle.

We present a case study on heavy-duty vehicles platooning. Using RoadRunner and detailed road data from digital maps, we quantify the impact of gap setting on fuel consumption, for a real-world route.

I. Introduction

Connectivity and automation can address some of the issues associated with the current transportation system. Connected automotive vehicles (CAVs) can access and share information wirelessly with each other and with infrastructure in real time through vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication protocols. The data-rich environment developed through V2V and V2I communication can be used to adjust vehicle movements in coordination with other vehicles and traffic control systems, and to enhance safety, energy efficiency, and mobility.

In recent years, research on cruise control (CC), with or without connectivity, aimed to develop algorithms that use wireless communication to reduce energy consumption and emissions. CC has been in use for several decades to assist drivers with speed regulation without distance control. The driver remains responsible for maintaining a safe distance from any other vehicles. Adaptive cruise control (ACC) uses radar, sometimes with the addition of a video camera, to add relative distance and relative speed control. Some passenger cars and heavy-duty trucks are currently equipped with this capability. The main problem with ACC is that if three or more ACC vehicles drive in a row, the system as a whole is string unstable [1]. A string of ACC vehicles on highways is less stable than a string of manually driven vehicles because the forward-ranging sensors cannot perceive the actions of vehicles ahead of the immediately preceding vehicle. Adding vehicles to the string causes larger and larger cumulative delays, which increases the unstable behavior. Cooperative ACC (CACC) with V2V is a possible solution to this problem [2].

Simultaneously, research efforts focused on vehicle modeling, simulation, and analysis moving rapidly from a single vehicle to multiple vehicles on the road, where the behavior of one vehicle affects the others. In order to verify the impacts connectivity and automation have on vehicle energy efficiency for various powertrain technologies, it is necessary to look at both powertrain models and driving environment models that include drivers, vehicles, and infrastructure. Creating completely different workflows to respond to new and emerging MBSE (model-based system engineering) needs at the microscopic or macroscopic level requires a substantial amount of time and effort. However, to date, most studies on CC have been limited mainly to a specific configuration; in some cases, they use a backward-looking vehicle simulator, so the dynamic performance of the vehicle cannot be considered explicitly [3].

In this paper, control algorithms for ACC and CACC inspired by literature are implemented in a new MBSE platform, RoadRunner [4], which models vehicle longitudinal movements with non-connected or connected and automated driving in the same environment, taking into account real route information. This modeling framework can collectively provide necessary tools to predict energy consumption for various driving decisions and other characteristics, such as car-following, free-flow, or eco-approach driving, and thereby can help in developing control algorithm for CAVs. RoadRunner is an extension to Autonomie [5], which is a forward-looking vehicle simulator based on MATLAB/Simulink that supports plug-and-play and is widely used for a large number of powertrain technologies.

This paper is organized as follows. First, Section II provides an overview of how the framework is organized, and what process is used. The route attributes (speed limit, grade, etc.) can be automatically extracted as long as the user provides origin and destination from a digital map, and then the user defines which vehicle models to simulate and in which order. Second, Section III.A presents a sample platooning scenario for class 8 vehicles. Section III.B describes how we implemented the literature-inspired control algorithm in the framework. Finally, Section III.C describes how we evaluated the fuel consumption for each truck in the ACC/CACC string, demonstrating the effects of inter-vehicle gaps.

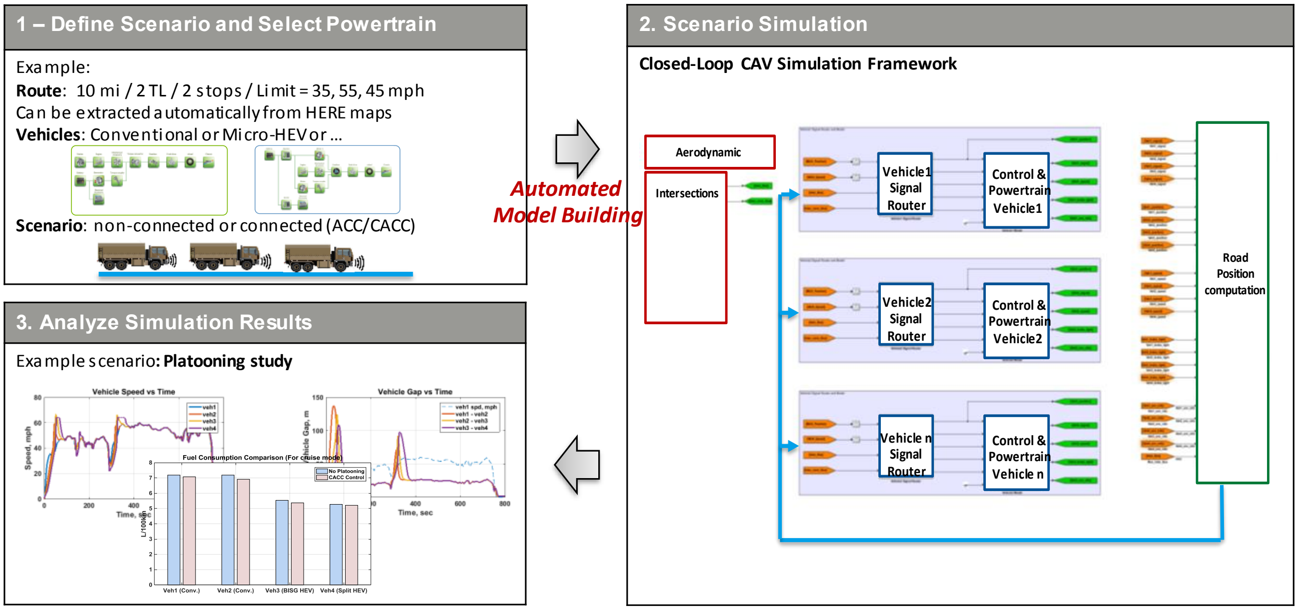


Figure 1. RoadRunner: multi-vehicle and road environment simulation workflow.

II. Simulation Framework for Platooning Vehicles

Based on our previous experience with Autonomie, an established vehicle energy consumption and performance tool, we started to develop a new framework to implement a closed-loop CAV simulation approach to workflows. This new framework is RoadRunner as shown in Fig. 1. RoadRunner can simulate longitudinal movements of one or more user-defined vehicles along a user-defined route. The real route attributes, such as the positions of traffic lights, grade, and speed limits can be automatically initialized from a digital map. The automated building then creates the Simulink model for the driving environment.

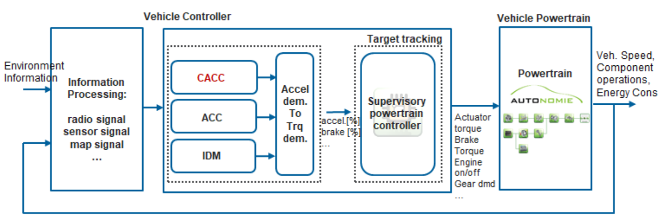


Figure 2. Vehicle model in RoadRunner.

A. Vehicle Longitudinal Model

Fig. 2 shows the schematic of the vehicle model architecture where the road side units share information, such as vehicle state trajectories, with each other via V2V connectivity. Within vehicle model, the user can select the non-connected or connected controller to evaluate the energy benefits of connected driving algorithms (see Fig. 2). The designers of the control strategy can evaluate their own control algorithm using a broad range of driving situations that include interactions with other vehicles, while being able to interact with the powertrain in a closed-loop fashion.

If the user chooses the non-connected driver model, the simulation assumes that drivers know the distance to preceding vehicle, the speed limits, and the current state of traffic lights. For the trailing car, the intelligent driver model [6] serves as reference. If the user chooses the connected controller, the simulation adjusts the driving with more information received via V2V connectivity. The driver model generates the acceleration and brake commands to achieve the target speed provided by the target-generation model; alternatively, it adjusts the vehicle speed to maintain a safe distance from the vehicles ahead using ACC or CACC. Section III presents the control strategy applied in this paper in detail.

The user also defines the vehicle models from the Autonomie vehicle list to simulate, and in which order they drive. The automated building then creates the vehicle models and the previously specified driving environment. It allows users to reuse full or partially complied Autonomie powertrain models for high fidelity. This supports the comparison of different powertrains so the user can quantify the global impacts connectivity has on energy use.

B. Aerodynamics

In close-driving platoons, the vehicles experience a reduced drag coefficient as a function of both inter-vehicle spacing and the number of vehicles in the platoon. In literature [7] by Lawrence Livermore National Laboratory, the researchers measured the average drag coefficient for two to three heavy-duty vehicles in the platoon. We integrated these drag reduction data into RoadRunner by extrapolating the experimental results to longer platoons. Fig. 3 shows the resulting two-dimensional combined extrapolation, as function of gap, number of vehicles, and position in the platoon.

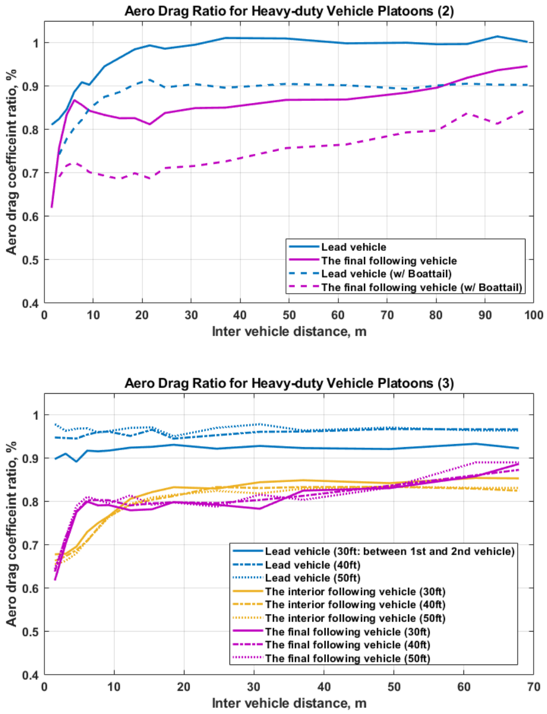


Figure 3. Air drag reduction as a function of inter-vehicle distance [7].

III. Application Example: Platooning Scenarios

To demonstrate RoadRunner, we implemented and evaluated ACC/CACC on two or three heavy-duty line-haul trucks. We evaluated fuel consumption for each truck while the vehicle travelled a simplified route and a real-world route, demonstrating the effects of inter-vehicle gaps.

A. Intelligent Driver Model

The intelligent driver model (IDM) has been used as a reference for modeling ACC/CACC car-following behavior. It defines the target acceleration for vehicles using a free-road strategy for acceleration and a combined strategy of the desired time gap and comfortable deceleration for braking. The IDM essentially defined an acceleration target command $a(t)$ that is generated by the following function [6]:

$$a(t) = a_0 \left[1 - \left(\frac{v(t)}{v_{IDM}}\right)^{\delta} - \left(\frac{\varepsilon^*(v(t),\Delta v(t))}{\varepsilon_i(t)}\right)^2\right] \qquad (1)$$

with $\varepsilon^*(v(t), \Delta v(t)) = s_0 + \tau v(t) + \frac{v(t)\Delta v(t)}{2\sqrt{a_0 b_0}}$,

where v(t) represents the current vehicle speed; v_{IDM} is the desired speed in free-flow traffic conditions; s_o is a minimum desired net distance, τ is the minimum possible time to the vehicle in front; a_o is the maximum vehicle acceleration, b_o is comfortable braking deceleration; and the exponent δ is set to 4. In this study, we assume that IDM controls the vehicle acceleration and deceleration when the vehicle speed is less than 30 mph or ACC/CACC is inactivated.

B. Cooperative Adaptive Cruise Control System

ACC enhances standard CC, even in the presence of traffic. A vehicle equipped with an ACC system has a sensor that measures the distance to the preceding vehicle and the relative velocity.

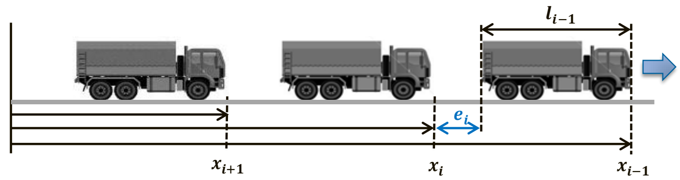


Figure 4. Longitudinal vehicle dynamic control.

The position of the i^{th} vehicle is noted x_i, as shown in Fig. 4. Then the measured inter-vehicle e_i is introduced:

$$e_i = x_i - x_{i-1} + l_{i-1}, \qquad (2)$$

with the preceding vehicle length l_{i-1}. The spacing error δ_i can be introduced:

$$\delta_i = x_i - x_{i-1} + L_{des}, \qquad (3)$$

where L_{des} is the desired spacing and includes l_{i-1}. The ACC's goal is to satisfy this condition: $\ddot{x}_i \rightarrow 0$, $\delta_i \rightarrow 0$.

In this study, we implement a constant time-gap strategy with $L_{des} = l_{i-1} + h \cdot \dot{x}_i$ as spacing strategy [8–10]. The control law is obtained by nonlinearities feedback linearization and given by:

$$\ddot{x}_{des}(t) = -\frac{1}{h}(\dot{e}_i + \lambda_P \delta_i + \int \lambda_I \delta_i), \qquad (4)$$

where h, λ_P, and λ_I are parameters to tune.

The CACC is an extension of the ACC where V2V data are available to improve the ACC system. CACC system build upon the current ACC system by adding V2V communication to supplement forward-looking sensors [11]. As shown in Fig. 5, adding communication reduces sensor processing delays, enabling shorter following gaps while reducing string instability. A CACC controller also aims to automatize the longitudinal control of the vehicle speed with a constant time gap.

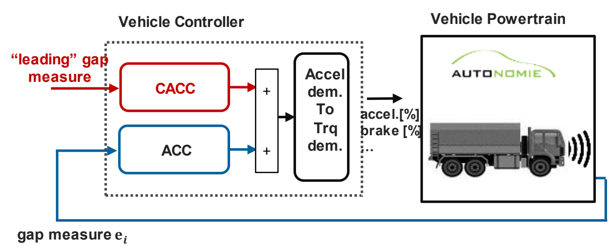


Figure 5. Simplfifed CACC system.

C. Simulation Results

In this section, we present the simulation results of the platooning control algorithm explained in Section III. On RoadRunner, each vehicle implements the control algorithm to assist the driver with speed regulation and distance control. For the simulation, we examine scenarios with two- or three-truck platoons in a single line road with route attributes.

Autonomie conducts detailed modeling of a conventional class-8 line-haul vehicle that runs on an internal combustion engine. The class 8 truck gross vehicle weight rating (GVWR) is a vehicle with a GVWR exceeding 33,000 lb (14,969 kg). Fig. 6 shows the major components and their layout in a conventional vehicle. The baseline configuration is as follows:

• Vehicle total mass: 24,000 kg (50% load)
• Engine: 336 kW (451 HP), diesel, 14.6L
• Drag coefficient = 0.55
• Frontal area = 10 m^2
• Tires: P295/75R22.5 (radius = 0.5105 m)

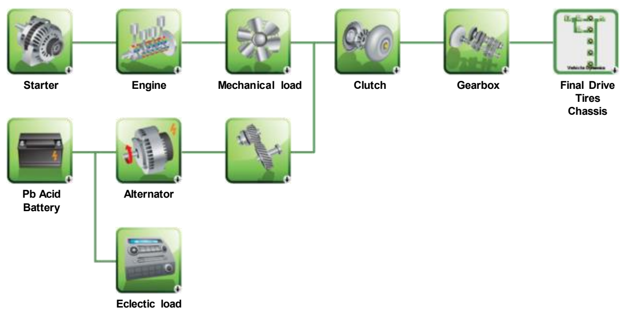


Figure 6. Conventional vehicle architecture for a class 8 truck.

First, we consider a simple scenario of constant speed testing and its corresponding speed limit-distance diagram as shown Fig. 7. We evaluated the IDM, ACC, and CACC algorithms using the same speed profile as that of the leading vehicle, which is driven by a human driver model [4], to continue driving at maximum speed allowed.

Fig. 8 presents the results for the three controllers of trailing vehicle for the same speed changes of lead vehicle on route 1. The top panel shows the speed of the leading car and the behavior of the following car when the vehicle is controlled by the IDM, ACC, or CACC. The bottom panel shows the inter-vehicle clearance distances for each of the three controllers. The IDM experiences a significant delay in its response to the leading car's changes in speed. For the ACC, the delay in response causes a smooth overshoot on each acceleration phase. For the CACC, the delay in the speed response is drastically less, which makes the driver feel that the vehicle responds promptly to the leading car's changes in speed, and the overshoots are eliminated.

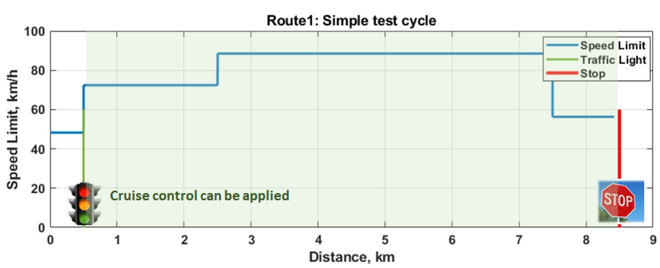


Figure 7. Route attributes for test cycle (route 1).

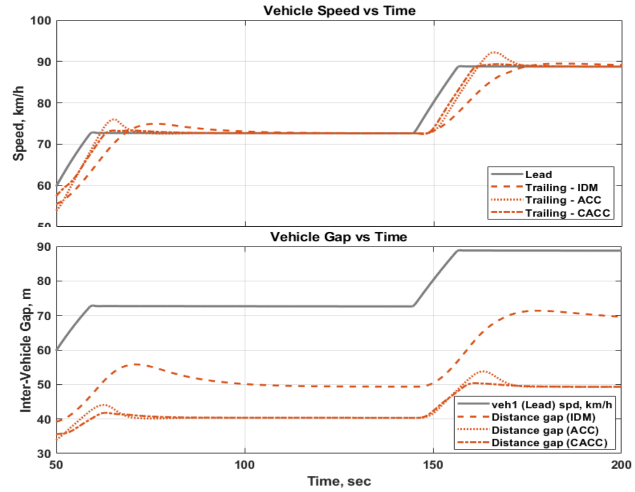
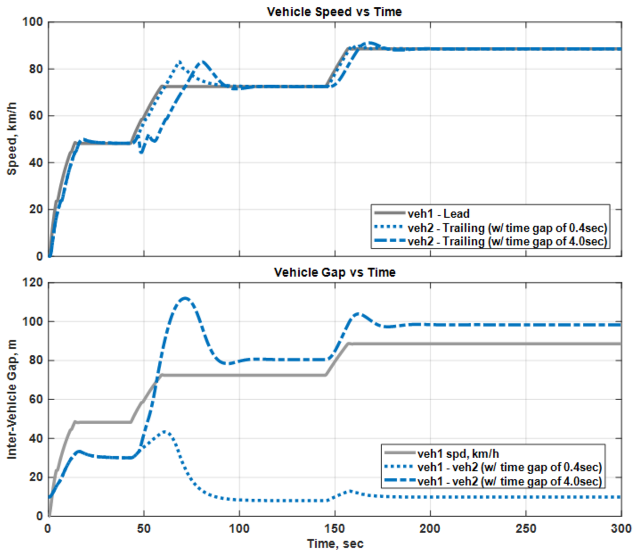


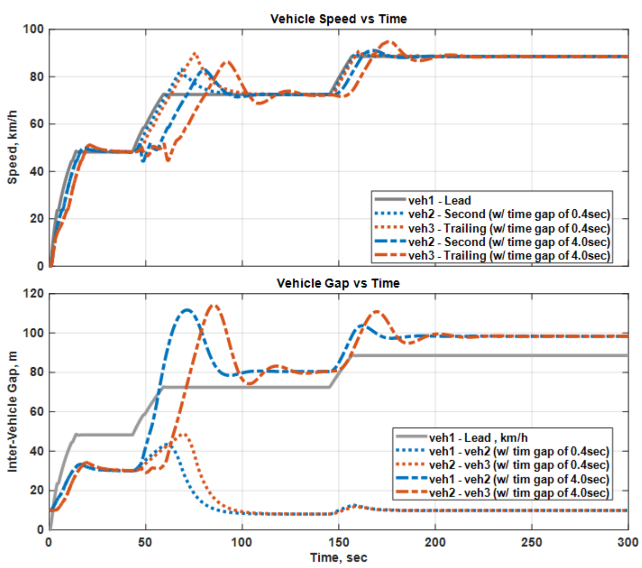
Figure 8. Car-following policies using CACC, ACC, and IDM controllers on route 1.

We then compared the variability in vehicle configurations and simulation conditions. We found that there is significant uncertainty in the potential benefits of a two-truck platoon compared to those of a three-truck platoon. To understand these potential differences, we simulated seven different separation time gaps: 0.4, 0.6, 0.8, 1.0, 2.0, 3.0, and 4.0 sec.

Fig. 9 shows the vehicle speed and distance gap of each CACC equipped vehicle in a two- or three-truck platoon. The trucks were in CACC mode when vehicle speed was above 30 mph; they used wireless V2V communication to augment their radar sensor data, which enabled safe and accurate vehicle following at the specified time gaps. Reducing the time gap from 4.0 to 0.4 seconds causes the inter-vehicle clearance to vary from 98.4 to 9.8 meters at a speed of 88.5 km/h.



(a) two vehicle platoon



(b) three vehicle platoon

Figure 9. Vehicle speed and distance gap for (a) two- and (b) three-vehicle platoons (with time gaps of 0.4 and 4.0 sec).

Fig. 10 shows the fuel consumption results for the lead, middle, and trailing vehicles. The results for the lead vehicles all agree: no savings are realized beyond about a 1.0-sec time gap. However, at closer than 0.5 sec the fuel savings rapidly ramp up to about 2.5%. We evaluated the minimum fuel consumption for the trailing vehicle at the 0.8- and 2.0-s time gaps. Below the 0.8-sec time gap, the fuel consumption of the trailing vehicle increased as the time gap decreased. However, the results show that the fuel consumption of middle vehicle decreases as the time gap decreased for three-vehicle platoon case.

Table 1 compares the two- and three-vehicle platoon simulations with respect to fuel consumption for route 1. These results demonstrate that a three-truck platoon will provide a greater net fuel savings for a multi-vehicle system.

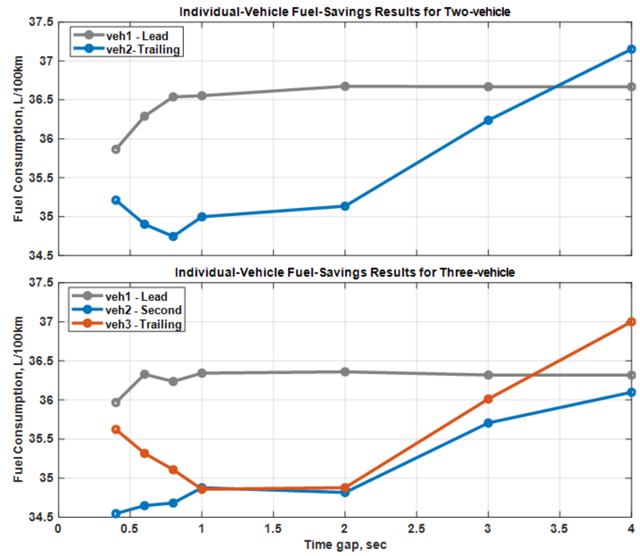


Figure 10. Individual-vehicle fuel consumption results for two (top) and three (bottom) vehicles.

TABLE I. FUEL CONSUMPTION COMPARISON BETWEEN TWO- AND THREE-VEHICLE PLATOONS

Time Gap (s)	Route 1 (L/100 km)[a]				
	Two vehicles		*Three vehicles*		
	Lead	***Trailing***	***Lead***	***Middle***	***Trailing***
0.4	35.9	35.2	36.0 (0.3%)	34.5 (-1.9%)	35.6
0.6	36.3	34.9	36.3 (0.1%)	34.6 (-0.7%)	35.3
0.8	36.5	34.7	36.2 (-0.8%)	34.7 (-0.2%)	35.1
1.0	36.6	35.0	36.3 (-0.6%)	34.9 (-0.3%)	34.9
2.0	36.7	35.1	36.4 (-0.9%)	34.8 (-0.9%)	34.9
3.0	36.7	36.2	36.3 (-1.0%)	35.7 (-1.5%)	36.0
4.0	36.7	37.2	36.3 (-1.0%)	36.1 (-2.8%)	37.0

a. Unadjusted fuel consumption.

To emulate real road conditions for real-world highway cruising, we extracted route attributes in the Chicago area from HERE's digital map by defining an origin and a destination in a geographical interface, as shown in Fig. 11. The scenario consists of a journey from Plainfield to Chicago. The journey for the most part takes place a highway; about 55km (34mlies) distance. The scenario is mostly on a highway and only one intersection at the end.

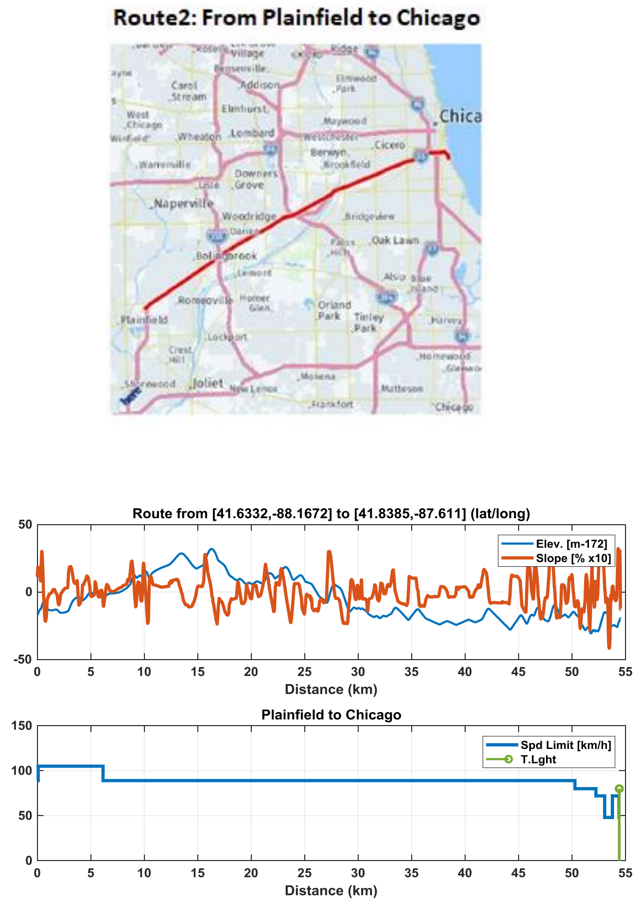


Figure 11. Route attributes in Chiago area, extracted from HERE's digital map (route 2).

We also ran the simulation for route 2. The distance gap of each CACC equipped vehicle in the three-truck platoon is shown in Fig. 12. Even when the time gap is set to 0.4 sec for CACC, the vehicle gap becomes more than 80 meters; when the time gap is set to 4 sec, the vehicle interval exceeds 180 meters. Fig. 13 shows the fuel consumption results for the lead, middle, and trailing vehicles on route 2. Overall, there is significant impact on the fuel consumption saving from platoon, due to the reduced drag coefficient. As the time gap is set below than 1 sec, the fuel consumption for lead and middle vehicle decreases. However, the improvement of fuel consumption for the trailing vehicle decrease in less than 1sec of time gap set.

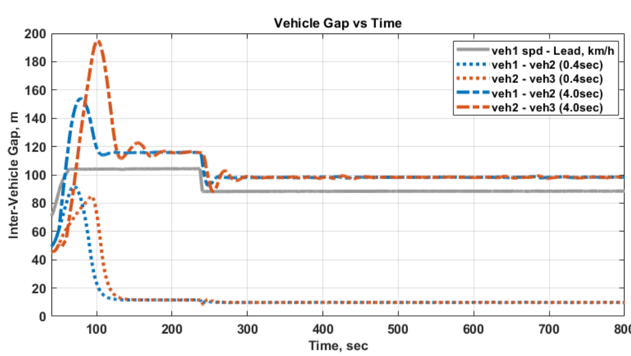


Figure 12. Distance gap for three vehicles (time gaps of 0.4 and 4.0 sec).

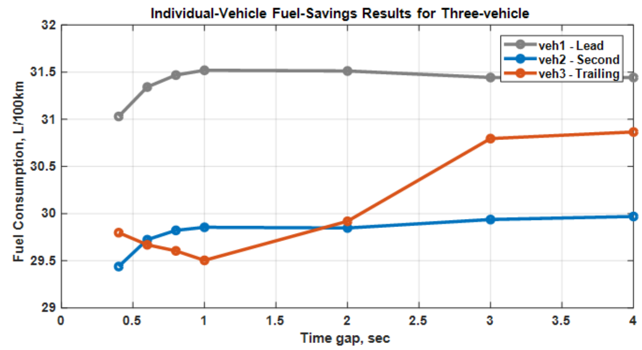


Figure 13. Individual-vehicle fuel consumption results for a three-vehicle platoon.

IV. CONCLUSION

In this study, we implemented literature-inspired control algorithms for ACC/CACC in RoadRunner, a newly developed microtraffic simulation tool. RoadRunner can simulate vehicles driving on the road, including their encounters with stop signs, intersections with signals, speed limits, and grades. RoadRunner was based on Autonomie, a forward-looking simulation tool that can estimate vehicle performance, including fuel economy, based on vehicle dynamics. The features of RoadRunner (or Autonomie) make the simulation close to the real-world driving.

The case study demonstrated how RoadRunner can be used to evaluate a CC. We also applied the IDM, ACC, and CACC control algorithm to a class 8 truck with V2V communication to assist the driver with speed regulation and distance control. Comparing the different controllers shows that the IDM produces smooth car-following behavior, but with very slow responses and large clearance gap variations. This comparison also demonstrates that string stability is not achievable for vehicles using ACC when the leading vehicle speed varies. This instability can be solved by adding V2V communications, leading to the CACC controller. In addition, we evaluated fuel consumption for each CACC equipped truck in the platoon for real-world highway cruising, demonstrating the effects of inter-vehicle gaps. The results show the maximum fuel savings for the trailing vehicle: about 4.4% at 1.0 sec of time gaps.

We will continue further development of RoadRunner before its public release, focusing on model complexity and scenarios. In particular, we will focus on validation of platoon cases using test data from collaborating researchers.

ABBREVIATIONS

ACC—adaptive cruise control
CACC—cooperative adaptive cruise control
CAV—connected automated vehicle
CC—cruise control
IDM—intelligent driver model
MBSE—model-based system engineering
V2I—vehicle-to-infrastructure

V2V—vehicle-to-vehicle

Acknowledgment

This report and the work described were sponsored by the U.S. Department of Energy (DOE) Vehicle Technologies Office (VTO) under the Systems and Modeling for Accelerated Research in Transportation (SMART) Mobility Laboratory Consortium, an initiative of the Energy Efficient Mobility Systems (EEMS) Program. DOE Office of Energy Efficiency and Renewable Energy (EERE) manager David Anderson played important roles in establishing the project concept, advancing implementation, and providing ongoing guidance. We would also like to acknowledge HERE for providing access to its digital mapping platform.

Disclaimer

The submitted manuscript has been created by UChicago Argonne, LLC, Operator of Argonne National Laboratory ("Argonne"). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.

The views and opinions of the authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

References

[1] X. Y. Lu and J. K. Hedrick, "Practical String Stability for Longitudinal Control of Automated Vehicles," *Int. J. Vehicle Systems Dynamics Supplement*, vol. 41, pp. 577–586, 2004.

[2] B. McAuliffe, M. Lammert, X. Y. Lu, S. Shladover, M. Surcel, and A. Kailas, "Influences on Energy Savings of Heavy Trucks Using Cooperative Adaptive Cruise Control," SAE Technical Paper 2018-01-1181, 2018, doi:10.4271/2018-01-1181.

[3] V. Milanés, and S. Shladover, "Modeling Cooperative and Autonomous Adaptive Cruise Control Dynamic Responses Using Experimental Data," *Transportation Research Part C: Emerging Technologies*, vol. 48, pp. 285–300, Nov. 2014, doi:10.1016/j.trc.2014.09.001.

[4] N. Kim, D. Karbowski, and A. Rousseau, "A Modeling Framework for Connectivity and Automation Co-simulation," SAE Technical Paper 2018-01-0607, 2018, doi:10.4271/2018-01-0607.

[5] Argonne National Laboratory, Argonne, IL, Autonomie (computer software), www.autonomie.net, 2018.

[6] M. Treiber, A. Hennecke, and D. Helbing, "Congested Traffic States in Empirical Observations and Microscopic Simulations," *Physical Review E*, vol. 62, pp.1805–1824, 2000.

[7] K. Salari and J. Ortega, "Experimental Investigation of the Aerodynamic Benefits of Truck Platooning," SAE Technical Paper 2018-01-0732, 2018, https://doi.org/10.4271/2018-01-0732.

[8] R. Rajamani, *Vehicle Dynamics and Control*. Boston, MA: Springer US, 2012.

[9] A. Khodayari, A. Ghaffari, S. Ameli, and J. Flahatgar, "A Historical Review on Lateral and Longitudinal Control of Autonomous Vehicle Motions," in *Proc. 2010 2nd International Conference on Mechanical and Electrical Technology (ICMET)*, 2010, pp. 421–429.

[10] P. A. Ioannou and C. C. Chien, "Autonomous Intelligent Cruise Control," *IEEE Transactions on Vehicular Technology*, vol. 42, no. 4, pp. 657–672, Nov. 1993.

[11] C. Nowakowski, S. E. Shladover, X.-Y. Lu, D. Thompson, and A. Kailas, "Cooperative Adaptive Cruise Control (CACC) for Truck Platooning: Operational Concept Alternatives," UC Berkeley: California Partners for Advanced Transportation Technology, 2015.