32nd Electric Vehicle Symposium (EVS32)
Lyon, France, May 19-22, 2019

Development of a Real-Time Mobility Control and Visualization System with Predictive Vehicle Speed Control for Connected and Automated Vehicles (CAVs)

Tim J. LaClair,[1] Zhiming Gao, Chieh (Ross) Wang, Jaqueline Rios-Torres, Jibonananda Sanyal, Rajasekar Karthik, Phil Nugent, Srinath Ravulaparthy and Anne Beres

Oak Ridge National Laboratory, Oak Ridge, Tennessee, U.S.A.

[1] corresponding author, email: laclairtj@ornl.gov

Summary

The Oak Ridge National Laboratory is developing a cloud-based platform for the deployment and evaluation of mobility- and efficiency-oriented connected and automated vehicles (CAVs) applications that will include fundamental data management, communications and visualization elements. This system will serve as a data portal, providing indirect communications that are relevant to efficiency applications, and can be used for both real world CAVs deployments and virtual systems that allow evaluation of new CAVs applications and algorithms. Predictive vehicle speed control algorithms have been developed and will be integrated with the real-time queryable transportation database and geospatial platform to achieve optimized energy efficiency and improved traffic flow in off-freeway driving. This paper describes the technology and summarizes evaluations being conducted for a corridor in Chattanooga, Tennessee, where the technology is expected to be initially deployed. To evaluate the fuel savings resulting from the vehicle speed control and the RTMCS data framework, vehicle powertrain simulations will be completed using results from a calibrated microscopic traffic model.

Keywords: eco-driving, energy efficiency, mobility system, connected and automated vehicles (CAVs)

1 Introduction

Recent studies have demonstrated very significant energy savings potential from the operation of connected and automated vehicles (CAVs) [1-6]. The term CAVs is used in the transportation community to include a broad category of vehicles with advanced information technology functionality [2]. CAVs rely upon advanced sensors and communications to obtain and share data of the surrounding transportation environment, which can be acted upon for improved vehicle and system-wide performance. CAVs technologies offer very significant potential for improved safety, reduced congestion, and improved energy efficiency, among others benefits. Sharing data corresponding to future times and at distances exceeding what a driver can observe enables very high levels of optimization: predicted energy/fuel savings for individual CAVs technologies often exceeds 20% [3-6], and the benefits from combinations of technologies can be even greater [2]. Speed control algorithms in CAVs research, however, have often used assumptions that are either not appropriate for real-time control or consider idealized conditions, and it has been shown that interactions among vehicles can have a negative impact on CAVs efficiency benefits [7]. With limited penetration of CAVs technology, it is also unclear how the current transportation system will transition to implement these technologies effectively in a manner providing maximum benefits.

While analytical studies and controlled experiments have shown the potential benefits of CAVs technologies, there is growing sentiment that the time has come to move towards larger scale deployments of CAVs technologies in the real world. The Oak Ridge National Laboratory (ORNL) is developing a software framework that will allow web-based data queries and feeds between vehicles and to traffic-based infrastructure that are necessary for many CAVs functions. As an initial demonstration, the system will be integrated with an application for predictive vehicle speed control (PVSC) that is expected to improve traffic flow and reduce fuel consumption. This paper describes the CAVs system under development, which is oriented primarily towards efficiency-based applications, as well as describing the methodology for the PVSC.

2 CAVs System Development, Demonstration and Deployment Plans

ORNL is developing the Real-Time Mobility Control System (RTMCS), which will serve as a flexible and scalable foundation on which further efficiency-based CAVs developments can be built. The RTMCS provides visualization capabilities to show vehicle locations and speeds, traffic signal states and other data that fully characterize the current traffic conditions. The visualization functions can be used by Traffic Management Centers to augment their monitoring and control of the traffic network in real time as well as by researchers to interpret and better understand both real-time and historical data and to visualize simulation results for CAVs. The system will be *flexible*, with the capability to add other data structures as additional CAVs functions requiring different information are implemented within the RTMCS, and it will be *scalable* so that larger geographic areas can be managed using the RTMCS as the initial region represented by the system expands. ORNL's initial development is aimed at a traffic network within the city of Chattanooga, which has an extensive roadside communication network and other CAVs infrastructure that facilitate the deployment of the RTMCS. The project team is developing the software framework for the RTMCS and is working closely with Chattanooga so that the RTMCS is fully ready for integration with the city's transportation and data systems when all necessary partnerships and hardware systems are finalized.

Since the new system is focused on mobility- and efficiency-oriented CAVs applications, not safety, cellular data communications are preferred over dedicated short-range communications (DSRC) due to the longer range capabilities of cellular signals [2,4]. Based on this underlying structure, all communications with the RTMCS take place through a queryable web interface. It is noted that longer communication times are acceptable for the efficiency applications, with expected time intervals for communications on the order of a few seconds.

The RTMCS will fundamentally be a geospatially-enabled high-performance database that will serve to meet the data exchange, data querying, route planning, V2V, and V2I communication requirements. The capability developed uses the Software-as-a-Service in the cloud model to provide data access features through RESTful interfaces to meet the mobility simulation needs while retaining the important elements that a real-world future deployment could build upon. The completed system will have four major components:

- Data exchange mechanisms – the web-based data exchange protocol will function to provide indirect V2V and V2I communications, accepting 'pull' requests as well as delivering 'push' notifications.
- Real-time queries – The system will support the capability for a dynamic number of connected vehicles to request information about their surroundings. A PostGIS database is employed to meet these requirements.
- Visualization subsystem – designed to provide high-level, synoptic assessments of the current state and anticipated future states. Built with OpenLayers over the PostGIS database, and additional widgets to interact with simulation or real-time data feeds.
- Simulation, data analysis, and other component models can be integrated into the system architecture. The use of RESTful interfaces over a PostGIS database will create the cloud-enabled RTMCS sub-system for the simulation sub-system and visualization system to interact with.

A schematic showing data flows and interactions between the RTMCS and the PVSC application is shown in Fig. 1.

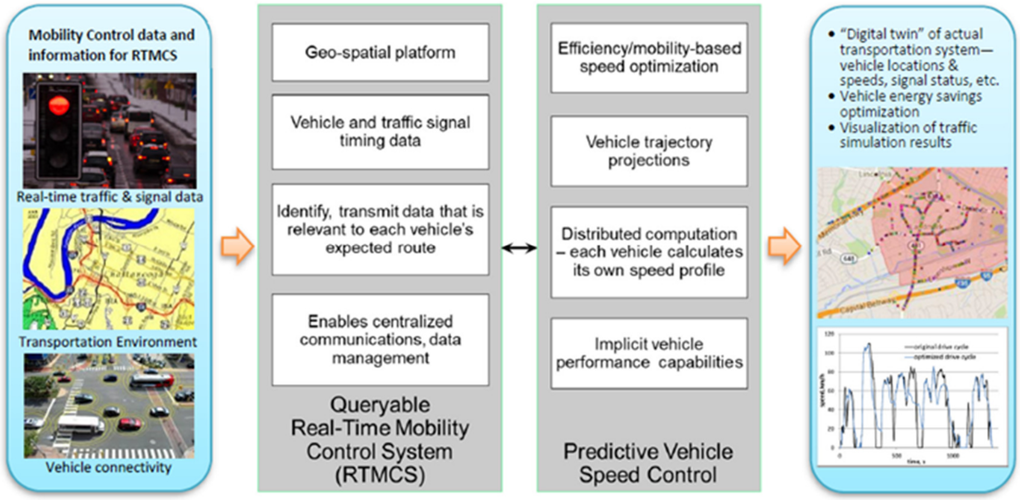


Figure1: Primary features of the RTMCS and interactions with the PVSC application

3 Big Data Framework

The following describes the Big Data framework within the RTMCS system. Our Big Data framework utilizes scalable and high-performance computing (HPC) techniques harnessed by a combination of well-proven and emerging technologies to deliver a state-of-the-art framework for CAVs applications such as the RTMCS. We discuss various technologies and our approach in leveraging them for the RTMCS below.

Popularized by cloud providers like Amazon and Google, Microservices has become a new norm for developing large-scale software frameworks by separating them into small and independent services. Some of the notable benefits of this approach are (a) Agility: develop new features in alignment with business activities, (b) Scalability: scale-out/in (horizontal) and scale-up/down (vertical), and (c) Resiliency: fault-tolerant and easier to identify, isolate, debug, and fix errors making the system less susceptible to catastrophic full-system failures [8]. Our framework has adopted Microservices as a way of developing various components needed to power the RTMCS, taking advantage of these benefits.

Containerization technology like Docker facilitates deploying and scaling Microservices on the cloud infrastructure as per demand needs [9]. We are using Docker to containerize various Microservices for this purpose. New instances of Microservices can be quickly deployed to meet the increased demand during heavy computation phases of predictive vehicle speed control algorithms. Then, the new instances can be un-deployed when computational demand decreases to save resources.

Jupyter Notebook has become very popular with data scientists from exploration to development of machine learning (ML) or artificial intelligence (AI) tools. Our team uses the Notebook for data exploration and analysis of the simulation output from Chattanooga. For integration with the rest of the components in our framework, we have created REST endpoints using Kernel Gateway module. This will allow simultaneous usage of Notebook as a means of development and scaling in a production environment. We have used the module to create REST endpoints on top of each logical cell(s). For example, within a single notebook, we have (a) data ingestion, (b) parsing as per their respective format or ingestion method (database, files, real-time streams), (c) transformation, and (d) storage of the transformed data. So, an ingestion REST endpoint has been created for ingestion cell, parser endpoint for parser cell activities, and so on. To achieve comprehensive ML lifecycle - experimentation, deployment, and reproducibility, we are evaluating Kubeflow and MLFlow and may incorporate one of the selections depending on its feasibility within our project [10].

To store the transformed data from the Jupyter notebook after data analysis, we utilize Apache Hadoop, an open source distributed file system framework for Big Data systems. The Hadoop Distributed File System (HDFS) is a prominent data management layer for Hadoop with linear scalability and fault-tolerant features. Hadoop can in-turn use Parquet to read and write data out of the file system. So, what is Parquet? It is a column storage data format that can be used by any technologies within the Apache Hadoop ecosystem. The key benefit is it provides a common data format regardless of the big data processing framework or model, or even various programming languages. For example, we are currently planning to use Apache Kafka, a stream processing library to serve part of our data pipelines. As we further explore, if the need arises to use say, Apache Spark, we can re-use the same data without any data transformation or massaging. This provides two main values: (a) it optimizes the data storage layer for both performance and immediate use by various technologies, and (b) saves time and cost during the development activity phase.

The architecture and data system for the RTMCS is aimed for an initial deployment in the city of Chattanooga, Tennessee. It is expected to be able to handle all of the requirements of the traffic system in real time under peak operating conditions with approximately 100k vehicles, including management of visualization functions, capabilities for vehicle tracking and communications required for the optimized vehicle speed control application. The system can be scaled on demand, so larger sized traffic networks can be managed by increasing computing power and/or more resources. Future CAVs applications of similar complexity can also be easily implemented in the RTMCS through hardware expansions.

4 Predictive Vehicle Speed Control Algorithm

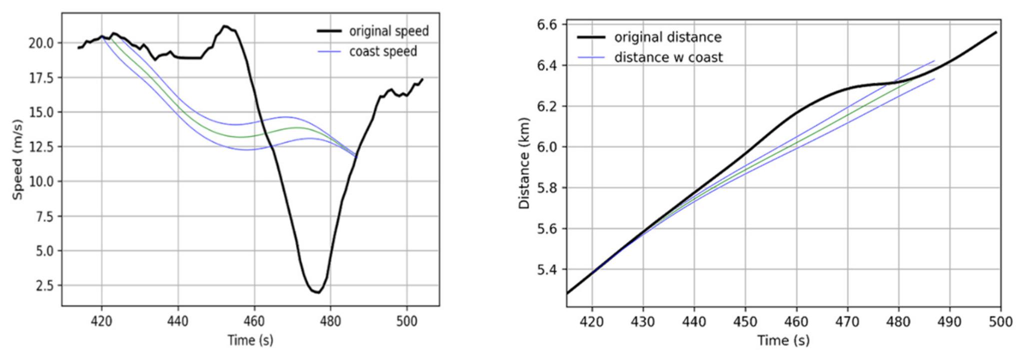
The PVSC control algorithm aims to minimize braking by coasting, in anticipation of required speed changes due to traffic signals. The algorithm can also be used to account for other vehicle traffic if speed profile data is shared among vehicles. By avoiding braking, subsequent accelerations which would be needed to return to the original speed are eliminated, or if extended coasting is used for decelerations it avoids the dissipative energy losses associated with braking. This approach implicitly minimizes tractive energy requirements by avoiding unnecessary accelerations, so it is expected to yield near-minimal fuel use without the need to perform an explicit optimization of the complete velocity profile, which is computationally challenging. Smoother traffic flow and avoidance of full stops at traffic signals are also expected, yielding benefits for reduced congestion.

The optimized speed profile based on the PVSC algorithm is determined iteratively, starting with an expected speed profile such as following the speed limit along a planned route, and considering normal accelerations consistent with vehicle powertrain limitations. The speed profile must be integrated in time to determine the vehicle position vs. time, $X(t)$. The speed profile calculation will normally be performed while considering a fixed distance or time ahead of the current location (the "evaluation window"), for example considering one mile further down the road, although if the complete drive cycle is known in advance the algorithm can be applied to optimize the entire drive cycle, as in [1]. Any known constraints associated with a lead vehicle's trajectory or traffic signals along the route are used to modify the initial speed profile so that $X(t)$ cannot result in an overtake of the lead vehicle or passing through a red light. Signal timing data can be accounted for in a manner consistent with lead vehicle position constraints by treating the light as a dummy lead vehicle with zero speed at its location while it is red [11]. In this way, we only need to consider a single position constraint function, $X_{con}(t)$, which must be updated any time that SPaT data for the timing of upcoming red lights is updated or additional constraint data becomes available.

If $X(t)>X_{con}(t)$ for any time within the evaluation window, then this indicates an overtake condition that must be avoided, and the speed profile must be modified. A coasting profile is sought to avoid the overtake condition, as follows. If the forces opposing vehicle motion are characterized in terms of a,b,c coefficients (as determined from vehicle coastdown testing for vehicle dynamometer testing), the coasting acceleration/deceleration rate is given by

$$a_{coast} = \frac{dV}{dt} = -g\, sin\, \theta(x) - \frac{1}{m}(a + b\, V + c\, V^2), \tag{1}$$

where $sin\ \theta(x) = dh/dx$ is the road grade as a function of position, g is the gravitational constant, m is the vehicle mass, and V is the instantaneous velocity at time t. Solving the time-based equation requires an iterative solution but Eq. (1) can instead be formulated in terms of position, and the speed can be calculated numerically based on an initial or final location and speed without iteration. The result can then be easily mapped back to the time domain. It should be noted that a coasting solution can be calculated from any starting location and speed, but each initial state will result in different end points and times. Figure 2 shows several possible coasting profiles starting at different times/locations for the same initial speed profile. In Fig. 2(a), the speed vs. time is shown, while in Fig. 2(b), the distance vs. time is shown for the same set of coasting solutions. The curve shown in green in each graph corresponds to a case where the end point of the coasting intersects the original speed profile at both the same time and location. Considering the distance travelled in Fig. 2(b), it is obvious that the later starting time results in overtaking of the initial trajectory, while the earlier start time/position results in a larger following distance. The case with equal time and distance is our desired speed profile for the PVSC since the distance at the end of the coasting matches that from the initial speed trajectory.



(a) Speed vs. time (b) Distance vs. time

Figure 2: Speed (a) and distance (b) vs. time for several coasting solutions starting at different locations/times for the drive cycle.

A methodology to determine coasting speed profiles that allow arrival at a downstream location at the same time as occurred in the original cycle can therefore be automated. This allows us to calculate optimized coasting solutions to eliminate braking, which yields a highly efficient speed profile. However, depending on how much advance notice is available to determine the coast profile, it may not be possible to fully avoid braking. Furthermore, if negative grades (down hill segments) are present, in some situations it may not be possible to obtain an optimized solution with pure coasting. The PVSC method must therefore request braking in addition to coasting, depending on traffic conditions and road grade.

For implementation with the RTMCS, each vehicle calculate its own speed trajectory in order to reduce the computational load on the RTMCS itself. The algorithm employs a numerical determination of coasting speed potential functions that satisfy the given constraints for distance traveled, and the road grade and vehicle characteristics are included in this calculation.

5 Evaluations Using Microscopic Traffic Simulation

To critically and realistically evaluate the proposed PVSC algorithm and its potential impact, a traffic simulation model of the Shallowford Road corridor in Chattanooga, Tennessee (also including Gunbarrel Road, Fig. 3) was developed in VISSIM. This is a popular business district in the city, and many of the intersections are equipped with advanced signal controllers from which SPaT and traffic count data can be obtained in real time. VISSIM is a microscopic traffic simulation software that allows transportation engineers and planners to realistically emulate traffic (e.g., vehicles, pedestrians, cyclists, and transit) and evaluate the performance of the underlying transportation network [12]. The basic network of this study was developed and calibrated in VISSIM based on historical traffic data, such as volume, speed, and turn movements (Fig. 4). Traffic signals at all intersections along this corridor were simulated in VISSIM using Ring Barrier Controllers with vehicle actuated programming according to the real-world SPaT plans.

Coordination among multiple signal controllers along the Shallowford Road corridor was also implemented in the VISSIM model.

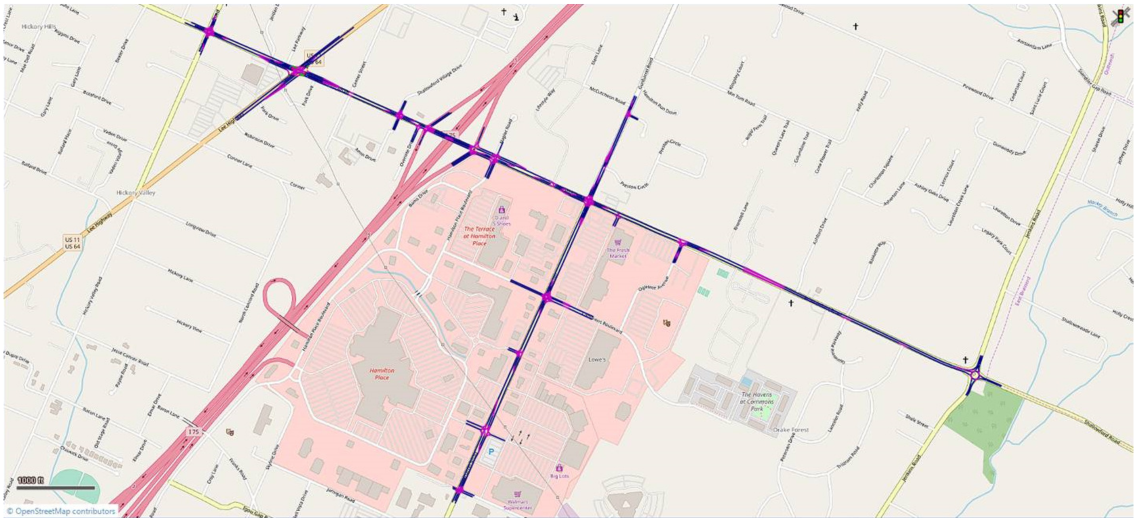


Figure 3: Shallowford Road corridor modelled in VISSIM

Figure 4: Traffic network developed and calibrated in VISSIM

CAVs equipped with the proposed PVSC algorithm are defined as a separate type of vehicle in the model, and the speed of each of these vehicles can be dynamically controlled to minimize braking. In this simulation, different penetration levels (from 0% to 100%) of CAVs equipped with the PVSC algorithm are simulated under different traffic conditions (e.g., low, medium, high traffic volumes) to evaluate the impact of the speed control algorithm on traffic flow. To quantify the fuel savings achieved as a result of the optimized speed control, high fidelity vehicle powertrain models using the speed data calculated from the VISSIM simulations will also be run.

A novel approach will be employed to evaluate the real-time performance capabilities of the RTMCS as part of the development and demonstration phase. By using an application programming interface (API) in VISSIM, it is possible to exchange data between VISSIM and the RTMCS at each time step of the traffic simulation. All data inputs required for the PVSC algorithm will be passed from the RTMCS server to the VISSIM model using the same fundamental web-based communications that would take place during the actual deployment of the RTMCS. Similarly, vehicle positions and speeds calculated in the VISSIM model, in addition to speed profile data calculated by the PVSC algorithm and SPaT data, will be output to a database as the simulation progresses, and this data can be transferred to the RTMCS to emulate the communications from actual vehicles and traffic signals that would occur during normal operations in an on-road deployment. The VISSIM simulation, which runs faster-than-real-time, provides all of the data corresponding to the traffic network while the RTMCS performs all of its intended functions, including visualization of the virtual traffic network. In this manner, we will complete a software-in-the-loop evaluation of the RTMCS that exercises all of its normal functions. A larger-scale evaluation can be performed simply by duplicating the VISSIM data at each time step as many times as needed so that the RTMCS performs the required data processing at a level consistent with a full-scale deployment. This methodology allows the project team to perform an initial assessment of the RTMCS and PVSC system performance before the system is deployed in a real-world traffic system, which will serve to provide a validation of the real-time capabilities of both system as well as to evaluate their performance in realistic traffic conditions.

Acknowledgments

Notice: this manuscript has been authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan.

References

[1] Z. Gao, T.J. LaClair, S. Ou, et al., “Evaluation of electric vehicle component performance over eco-driving cycles,” *Energy*, vol. 172, pp. 823-839, 2019.

[2] U.S. Department of Energy, *Quadrennial Technology Review 2015*, Chapter 8A: Technology Assessments, Connected and Automated Vehicles, 2015, https://www.energy.gov/sites/prod/files/2015/11/f27/QTR2015-8A-Connected-Automated-Vehicles.pdf, accessed on 2018-10-15

[3] C. Manzie, H. Watson, S. Halgamuge, “Fuel economy improvements for urban driving: hybrid vs. intelligent vehicles,” *Transportation Research Part C*, vol. 15, pp. 1-16, 2007.

[4] T. J. LaClair, R. Verma, S. Norris, R. Cochran, “Fuel economy improvement potential of a heavy duty truck using V2X communication,” Proc. 21st ITS World Congress, 2014.

[5] D. Shen, D. Karbowski, A. Rousseau, “Fuel Efficient Speed Optimization for Real-World Highway Cruising,” SAE Technical Paper 2018-01-0589, 2018.

[6] C. Sun, J. Guanetti, F. Borrelli, S. Moura, "Robust Eco-Driving Control of Autonomous Vehicles Connected to Traffic Lights," *IEEE Trans*, arXiv preprint arXiv:1802.05815, 2018

[7] H.A. Rakha, H. Chen, M. Almannaa, et al., “Field Testing of Eco-Speed Control Using V2I Communication,” Virginia Tech Transportation Institute, Final Report submitted to Research and Innovative Technology Administration, U.S. Department of Transportation, 2016.

[8] Oak Ridge National Laboratory Microservices Design for Clinical Decision Support and Analytics, 2018.

[9] C. Pahl, A. Brogi, J. Soldani, P. Jamshidi, "Cloud container technologies: A state-of-the-art review", *IEEE Trans. Cloud Comput.*, 2017.

[10] S. Iqbal, “Scaling Jupyter notebooks with Kubernetes and Tensorflow,” 2019, https://learnk8s.io/blog/scaling-machine-learning-with-kubeflow-tensorflow, accessed on 3/8/2019-03-08.

[11] Q. Xin, R. Fu, W. Yuan, et al., "Predictive intelligent driver model for eco-driving using upcoming traffic signal information," *Physica A*, vol. 508, pp. 806–823, 2018.

[12] PTV Vissim, http://vision-traffic.ptvgroup.com/en-us/products/ptv-vissim/, accessed on 2019-03-08.

Authors

Dr. Tim J. LaClair is a Research Staff Member at the Center for Transportation Analysis at ORNL. His work combines experimental measurement with computer modeling and analysis of data. Recent research activities at ORNL have focused largely on vehicle fuel efficiency, evaluating the effects of different driving conditions and driver style, , and evaluating the benefits of advanced vehicle technologies, including efficiency-oriented CAV systems. Dr. LaClair received a Bachelor's degree in Mathematics from the University of Michigan, a Master's degree in Mechanical Engineering from Florida Institute of Technology and a Ph.D. in Mechanical Engineering from Purdue University.

Dr. Zhiming Gao is a R&D staff member at the National Transportation Research Center (NTRC) at Oak Ridge National Laboratory (ORNL). He has over 15 years of experience in fuel economy and emissions control related to advanced vehicle technologies, particularly in the field of vehicle component and powertrain simulation and modeling. His recent work focuses on vehicle electrification and connected and automated vehicle technologies for light-duty and heavy-duty vehicles. Dr. Gao has published more than 100 journal/conference papers, reports and presentations on combustion engines, aftertreatment, vehicle powertrains and CAV technologies.

Dr. Chieh (Ross) Wang is an R&D Staff Member at the Oak Ridge National Laboratory (ORNL). His research revolves around utilizing emerging sensing and computational technologies for transportation data collection, analysis, visualization, modelling, and application. His recent research has focused on developing computer vision algorithms and microscopic traffic simulation models for connected and autonomous vehicles. Dr. Wang received his Bachelor's and Master's degrees in Civil Engineering from the National Taiwan University and his Ph.D. degree in Civil and Environmental Engineering from the Georgia Institute of Technology.

Jackeline Rios-Torres is an Eugene P. Wigner Fellow with the Energy and Transportation Science Division, Oak Ridge National Laboratory,Oak Ridge, TN, USA. Her research is focused on connected and automated vehicles, intelligent transportation systems, and energy management control of advanced powertrains. She received the B.S. degree in electronic engineering from the Universidad del Valle, Cali, Colombia, and the Ph.D. degree in automotive engineering from Clemson University, Clemson, SC, USA. She is also a GATE fellow with the Center for Research and Education in Sustainable Vehicle Systems, CU-ICAR.

Jibonananda (Jibo) Sanyal is the Acting Group leader for the Computational Urban Sciences group at ORNL. His research at ORNL is at the intersection of simulation and modeling, high performance computing, data sciences, visualization, big-data, as well as sensors and model-predictive control with applications in geographic information sciences, building energy modeling, CHP, transactive controls, climate change, and advanced manufacturing. He has a strong interest in devising sustainable solutions that bridge disciplines, that require synergistic connections across energy, water, geophysical, and anthropomorphic processes in our environment at local, regional, and global scales. He holds a PhD from Mississippi State University.

Rajasekar Karthik is a Research Staff Member at the National Security Emerging Technologies Division at the Oak Ridge National Laboratory. He earned his Master's degree in Computer Science from Purdue Univeristy. His research expertise include specializing in building cyber-infrastructure and web software for scientific research, education, and collaboration; proficiency with a vast array of programming languages; full life-cycle software development process; and search engine technologies. His current research interest areas are (1) Scalable and High-Performance architectures; (2) Geographic Information Systems (GIS); (3) Emerging technologies such as Microservices, Big Data frameworks, and NoSQL; and (4) Cloud computing.

Phil Nugent is a Systems Architect in the Geoinformatics Engineering Group at Oak Ridge National Laboratory. During his 10 years at ORNL, he has worked on developing distributed, geospatial data systems and informatics platforms focused across many domains, including transportion, energy assurance and situational awareness, and energy-water systems. Mr. Nugent received a Bachelor's degree in Geography and a Master's degree in Civil Engineering from the University of Tennessee.

Srinath Ravulaparthy is a Resarch Scientist at Oak Ridge National Laboratory. His research focuses on transportation modeling & simulation, travel behavior analysis, regional science and computational statistics. He is a Ph.D. graduate from the Department of Geography, University of California, Santa Barbara in Transportation Modeling and Simulation. He also received his M.S. and B.S. in degrees in Civil Engineering from Arizona State University and Jawaharlal Nehru Technical University, India, respectively.

Anne Berres is a Postdoctoral Research Associate in the Computational Sciences and Engineering Division at Oak Ridge National Laboratory. Her research is focused on large-scale data science, high-performance computing, modelling and simulation in urban and climate sciences. Dr. Berres received her Ph.D. in the International Research Training Group for Visualization of Large and Unstructured Data Sets, Applications in Geospatial Planning, Modeling and Engineering at the University of Kaiserslautern, Germany, where she also received a B.S. and M.S. in Computer Science. Prior to joining ORNL, Dr. Berres was a Postdoctoral Researcher in the Data Science at Scale Group at Los Alamos National Laboratory.