

SMART CITY REAL-TIME DATA-DRIVEN TRANSPORTATION SIMULATION

Abhilasha Saroj
Somdut Roy
Angshuman Guin
Michael Hunter

Richard Fujimoto

Civil and Environmental Engineering
Georgia Institute of Technology
Atlanta, GA 30332, USA

Computational Science and Engineering
Georgia Institute of Technology
Atlanta, GA 30332, USA

ABSTRACT

This study assesses feasibility aspects of using a real-time data-driven transportation simulation model to evaluate and visualize network performance indices to provide dynamic operational feedback in a real world environment, in a big data context. A hybrid traffic simulation model, consisting of a mix of preset and real-time data-driven intersections, is developed. The hybrid model represents a traffic corridor partially equipped with smart devices generating high velocity, high volume datasets with limited shelf-life. The model used in this study emulates seventeen consecutive intersections on a corridor. Signal controls and vehicle volumes at two of the intersections are driven by real-time data while the remaining intersections are driven by preset data. An optimized architecture is developed to enable control of the signals and the vehicle volumes using real-time data from in-field detectors, and real-time processing of the vehicle trajectories from the simulation output to generate travel-time, energy, and emissions performance indices.

1 INTRODUCTION

With leadership from the United States Department of Transportation, Smart City programs are fast becoming a reality (Rouse 2017; U.S. DOT n.d.). A Smart City uses information and communication technology (ICT) facilities to develop insights to improve the quality of life of residents, workers, and visitors (U.S. DOT 2014). Within the transportation context Smart City initiatives emphasize technology deployment, with roadways and vehicles being equipped with connected vehicle (CV) technologies and numerous new embedded sensors. Such a Smart City is expected to consume real-time information and leverage communication technologies to address urban challenges, including transportation related issues such as traffic congestion and environmental pollution. Addressing these challenges entails the consumption of high volumes of high velocity data to influence real-time decisions in an effort to optimize the use of transportation systems and services. The early major incentive for the deployment of many CV and embedded sensor technologies was the potential for reductions in vehicle crash related injuries and fatalities (U.S. DOT 2017; U.S. DOT n.d.). However, with high-speed data transactions in a connected environment, travelers and traffic management centers are expected to receive relevant information in real-time, which will not only enable safer travel, but also enhance environmental friendly route/mode choices and optimized traffic operations.

An example Smart City transportation implementation is currently being championed by the Renew Atlanta program, Atlanta, Georgia, USA, as part of the North Avenue Smart Corridor project. The project aims to make the North Avenue NW Corridor a smart connected corridor by integrating data, communications, and analytics in the intelligent and informed management of the transportation system (RenewAtlantaBond 2018). This paper presents a hybrid real-time data-driven transportation microscopic

simulation model of 2.6 miles of the North Avenue Smart corridor, that allows real-time visualization of network performance metrics.

Smart connected vehicle corridors are envisioned to use wireless communications such as Dedicated Short-Range Communications (DSRC) or 5G, on-board computer processing, advanced vehicle sensors, GPS navigation, and smart infrastructure to build a connected network. However, as CV is still an emerging technology, with no requirement from either the Department of Transportation or automobile manufacturers to deploy infrastructure or on-board units, research is limited by the availability of testbeds. In addition, there is currently a frequent arrival of new embedded sensors being developed for smart city corridors. It is expected that as these technologies are tested, and with the incentives provided by smart cities initiatives, there will be incremental adoption of the technologies in the field. During this transition phase, as the technologies are deployed, it is expected that there will be a mixed or hybrid environment in which the equipment, vehicles, and drivers will have to operate until there is a full saturation of technology adoption. For investigating the effective application of additional data available through CV and embedded sensor technologies and to prepare for future deployments or modification of operations, it is critical to develop the capability to simulate this hybrid infrastructure environment so that different scenarios can be developed and tested. Such a hybrid model that simulates an incremental deployment scenario where connectivity to the edges is partial, is presented here.

The model contains seventeen field calibrated signalized intersections with two of the signals driven by real-time signal status and vehicle arrival data streams. This study develops and implements the architecture required to integrate multiple physically separated systems, to enable communications, drive the simulation, and generate performance metrics in real-time. To evaluate the robustness of the approach, simulation results and performance metrics from the real-time data injection scenario are compared to the control case where all inputs are predetermined.

2 BACKGROUND

The integration of smart technologies such as sensors, networked communications, and hardware and software computing with the physical infrastructure is central in creating a Smart City (U.S. DOT 2014; Pop and Proştean 2018; U.S. DOT n.d.). One area of significant focus is the integration of the new technologies enabling an improved estimation of the traffic state and real-time traveler and traffic information (Allström et al. 2017; Chen and Du 2017). Given the inherent challenges in field experimentation, the use of traffic simulation models driven by real-time input data to emulate the real-world environment is utilized in numerous studies (Xiaowen et al. 2003; Brüggmann et al. 2013). Various traffic simulation tools have been used by researchers depending on the modeling requirements. A University of Leeds report in 2000 compared the capability of several macroscopic and microscopic simulation models in developing real-time traffic management solutions ("SMARTEST" 2000). More recently, a similar study looked into 17 simulation software tools and noted “a lack of online traffic simulation software applications specially designed for heterogeneous road transportation networks” (Pell et al. 2017).

The concept of using real-time data to drive traffic simulation model is not new. Previous studies such as Henclewood et al. (2010) injected real-time vehicle detection data into microscopic simulation models to simulate the current traffic state, which was then used to generate predictions of future traffic states (Henclewood et al. 2010). An effort, tested in the Dutch City of Assen, built a real-time traffic model that used traffic flow and travel-time data from different sensor technologies such as cameras, highway loop detectors and Bluetooth® sensors to predict the short-term traffic state. The model architecture connected the real-time traffic measurements with the macroscopic dynamic traffic assignment model “StreamLine”. Traffic counts were used for model calibration and forecasting (Wismans et al. 2014). A study by Sturari, Catani et al. presented the use of in-field mobile and fixed sensor data to drive a real-time microscopic traffic simulation model built using the Simulation of Urban Mobility (SUMO) simulation package. Model input included the real-time traffic count and vehicle location data obtained from different sources such as induction loop, camera counter, radar counter, automatic vehicles location (AVL) systems, etc. (Sturari et

al. 2016). In these studies, the real-time data comprised of vehicle detection or vehicle position data; however, infrastructure information such as the traffic signal state, state of ramp meters, information from variable speed limit signs, etc. were assumed to be pre-encoded in the simulation model based on known logic or field calibration. Today, with the richness of information in a Smart City CV environment, where the vehicle sensor data may result in real-time changes to the signal control, ramp meter rates etc., it is imperative that the state of the infrastructure is updated in the simulation to ensure the accuracy of the simulation results.

The availability of these various data streams also provide interesting challenges. The volume, velocity, and wide variety of these data streams naturally suggest the use of big data technologies for extracting useful information. Previous studies such as Amini et al. (2017) have used tools such as Kafka to address the issues of volume and variety of big data (Amini et al. 2017). Lv et al. (2015) used deep learning techniques for predicting traffic flow (Lv et al. 2015). The current study takes a hybrid approach, where the architecture allows for the use of big data concepts in the extract-transform-load (ETL) stages preceding the injection of the data into a simulation model.

3 MODEL ARCHITECTURE

To capture the intricacies of the incremental deployment of CV technologies and infrastructure connectivity, this study proposes the use of a hybrid simulation that is partially modeled using preset data and partially modeled using real-time data. This requires the integration of multiple disparate modules. There are three high-level tasks 1) real-time data injection into the simulation, 2) simulation model execution, and 3) generation of performance metrics. Dynamic links are established between these three components to ensure faster than real-time computations, ensuring that the simulation does not fall behind the real-world operations. Figure 1 below presents the complete model architecture.

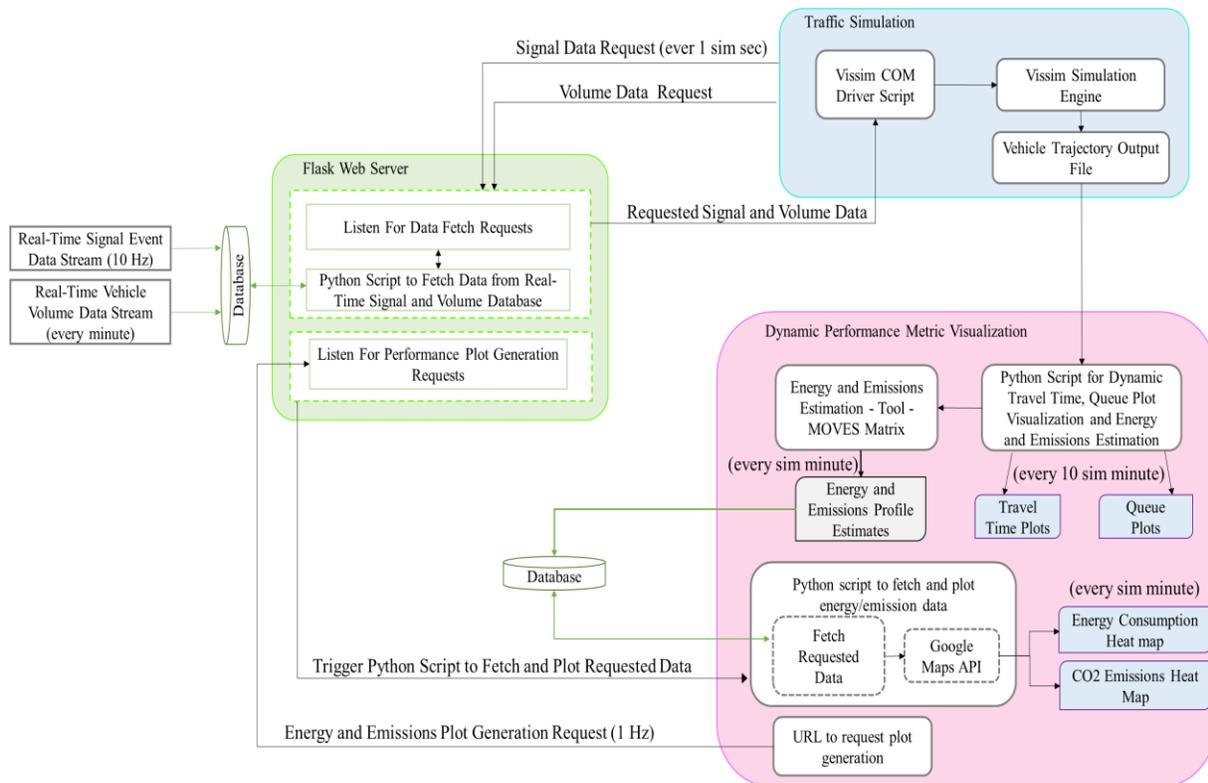


Figure 1: Model architecture.

Central to the architecture is the Flask micro web framework that acts as the communication facilitator and data broker. Several PERL and Python scripts perform the task of continuously fetching the data from the sensors on the field and populating a database. The microscopic simulation requests the data from this database via requests through the Flask engine and receives data as and when it becomes available. The simulation writes the vehicle trajectories to an output file which is scanned continuously by another set of Python scripts that generate the performance metrics. The results are available for retrieval via requests through the Flask interface. Details of the major components and the links between them are discussed in the following subsections.

3.1 Traffic Simulation Model

The traffic simulation model is built using an off-the-shelf simulation package, PTV’s Vissim version 9.00-08. Primary inputs for the Vissim transportation model are signal timing plans, vehicle volume inputs and turning movement percentage splits at the intersections. The signal timing plans of the intersections on the corridor are obtained from the City of Atlanta, the agency responsible for operation of the signals. The vehicle volume inputs and turning ratios are obtained by manual vehicle count data collection at the intersections. At two consecutive intersections, Glen Iris Drive at North Avenue and Ponce City Market at North Avenue, signal controls and vehicle volumes are driven using real-time emulation of signal control events and vehicle volume data from the field sensors. The remaining 15 intersections use the preset values in the calibrated model. Figure 2 shows a snapshot of the network corridor with the two real-time data-driven intersections highlighted.

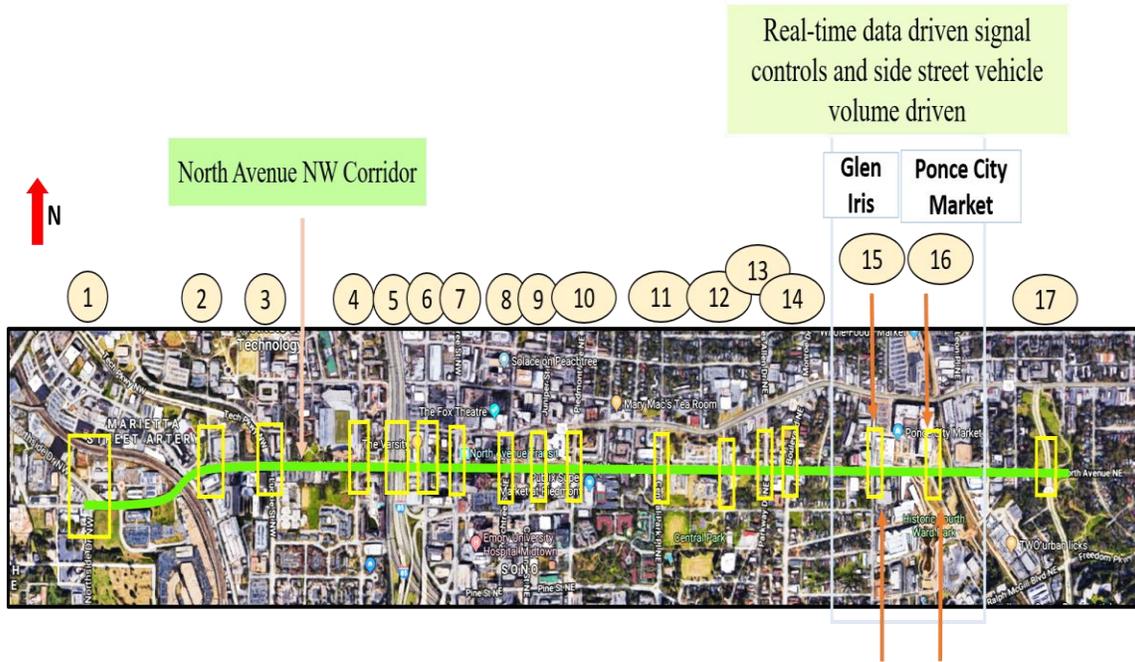


Figure 2: Study corridor. (Courtesy Google Maps®).

3.1.1 Volume Update Logic

The vehicle volume simulation inputs for the northbound and southbound approaches at the two real-time data driven intersections (Glen Iris Dr. and Ponce City Market) are updated every 5 minutes to match the fluctuations in real-time vehicle volume. This is implemented based on the previously field collected initial flow data used in the model calibration, adjusted to reflect the 5 minute interval real-time proportional

fluctuations in the traffic volume. This approach allows the vehicle volume inputs at the two real-time data driven intersections to remain consistent with the calibration field-collected corridor volumes.

3.2 Injecting Real-Time Intersection Signal Control and Vehicle Volume State Data

The volume and signal state input data are obtained from sensors in the field. The data are polled out of the field sensors by a set of PERL scripts and stored in a database. The data are injected into the simulation during runtime using Vissim's COM interface to emulate the real-time state of signal control and vehicle volume for the two intersections in the simulation model. The Flask server provides the interface for Vissim COM to fetch the data from the database.

3.2.1 Signal Controller Data Parsing

The signal events data that is polled out of the field signal controllers contains a 10Hz series of timestamped eventIDs and the corresponding parameters. The events of interest required to drive the signal controllers in the simulation model are:

- 1: Start of Green Indication
- 8: Start of Amber Indication
- 10: Start of Red Indication

For these Event Type IDs, the parameter column provides corresponding signal phase number.

3.2.2 Vissim COM-Interface

The Vissim COM interface is used to change the simulation network objects during the simulation runtime based on the signal phase and sensor inputs. Python 2.7 is used to drive the Vissim simulation model using its COM interface, by running the simulation model in single step mode, and changing the signal controls and vehicle volumes during run time at respective time steps. The simulation resolution is set to 10, that is 10 simulation time steps are executed per simulation second. For every time step of the simulation period, series of COM commands are executed along with the command to run the next simulation step.

At every 10th time step that marks start of a simulation second, the COM driver script requests real-time signal event data for both intersections for the corresponding timestamp. The driver script then assesses the fetched data to determine whether the event type ID for a signal phase indicates a change of state such as start of green, start of amber or start of red indication, and updates the state of the corresponding signal head object in the simulation.

To update the vehicle volume-input for the two real-time intersections, the COM driver script sends a request at every 3000th time step (5th simulation minute) to fetch the vehicle volume count for the last 5 minutes. The input volumes are determined using the logic described earlier. The COM driver script's flowchart to execute each simulation time step is shown in Figure 3, with the requests for real-time signal and volume data marked as callouts 1 and 2.

3.3 Dynamic Performance Evaluation Visualizations

The simulation's output is captured in a vehicle trajectory output file which updates during runtime. The performance evaluator module fetches data from the Vissim trajectory output file, with non-locking reads, and performs the computations for queue lengths, energy, and emissions. The visualization of these parameters in the form of time series plots, are available in real-time, as the simulation progresses.

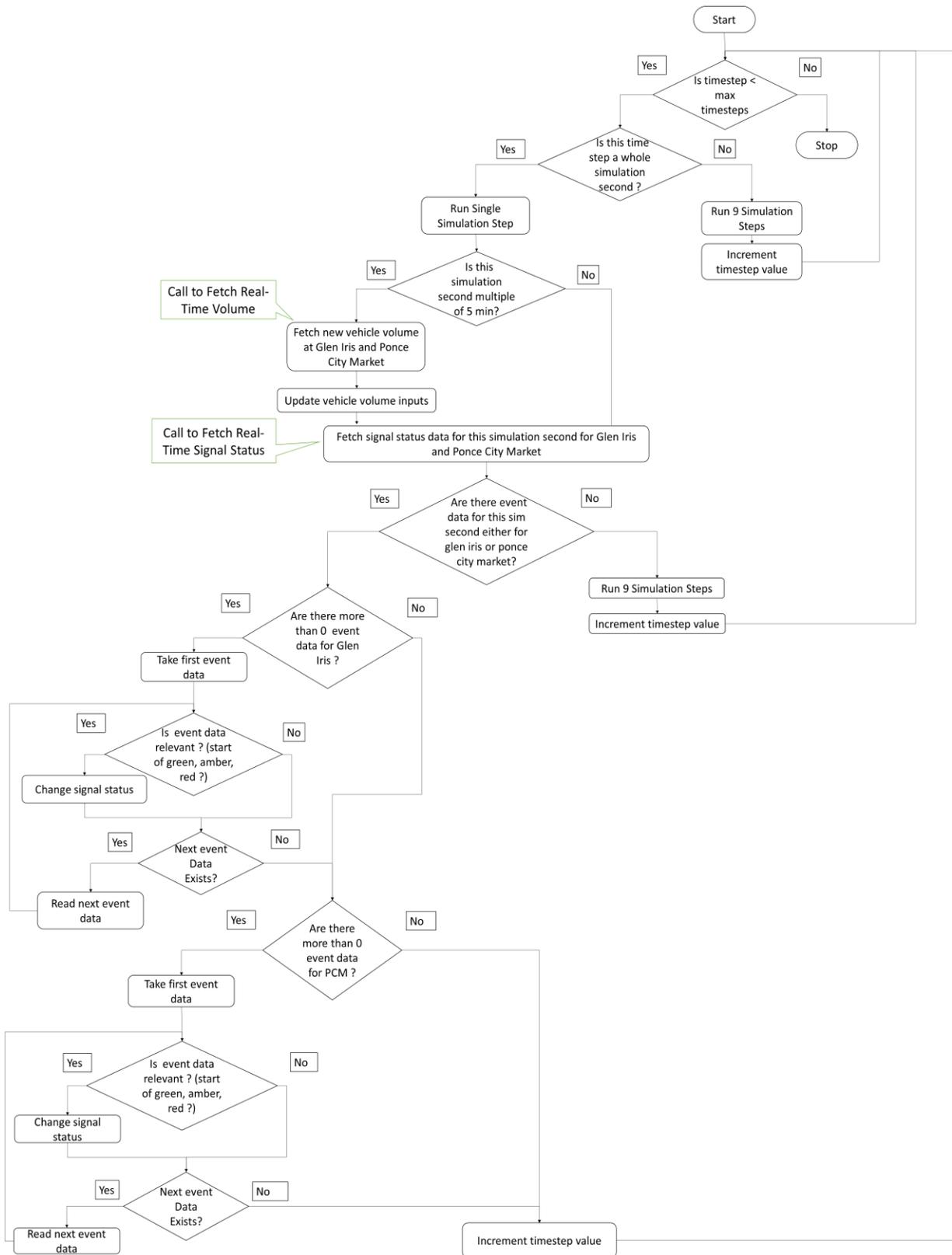


Figure 3: Flowchart for vissim com driver script.

3.3.1 Queue Length

The trajectory output file is used to measure the queues at the different approaches at the intersections. For faster computation, one record every ten records is chosen for each vehicle from the trajectory file. Then, the vehicle is mapped to the appropriate intersection approach's queue-counter based on its coordinate, where each intersection approach has a queue-counter zone mapped from the stop bar of the intersection approach to the upstream intersection. A vehicle is considered part of the queue if it is either at a standstill or is moving slowly with a possible intention to stop within the queue-counter zone. Thus, vehicles are chosen for queue measurement if and only if the speed is less than a threshold of 6.2 miles per hour, the acceleration is non-positive, and the vehicle headway to the next downstream vehicle is less than 65.6 feet (unless the vehicle is in the leading position on the intersection approach, in which case the headway criteria is ignored). These criteria are based on the Vissim default queuing threshold values. If a vehicle satisfies all these criteria, the distance of the vehicle from the beginning of the queue-counter is calculated. The maximum distance for all qualifying vehicles for a specific approach at a specific instance is read as the queue length for that instance. The described architecture is shown in Figure 4.

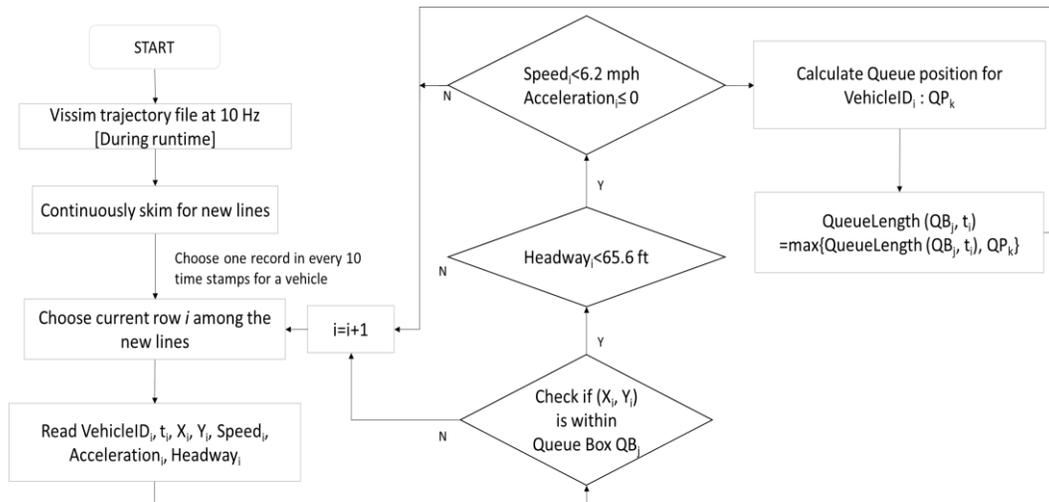


Figure 4: Working architecture for queue length computation.

3.3.2 Energy-Emission

The trajectory output file is also used to estimate energy and CO₂ emissions profile in real-time at a predetermined regular interval using the Motor Vehicle Emission Simulator (MOVES) matrix. MOVES matrix is a computationally improved version of the energy estimation tool MOVES, developed and mandated by the US Environmental Protection Agency (USEPA) (Guensler et al. 2017). It estimates energy and emissions for off-road and on-road vehicles based on the vehicle type, weather conditions and vehicle model, make, and year. For energy and emissions estimates, the 10 Hz. data in the trajectory output file is too noisy for direct use. The 10 Hz. data is condensed to 1 Hz. by taking the median of every 10 records for every vehicle.

The vehicles' coordinates are used to assign the vehicles to short 200-ft segments on the network. A visualization of the cumulative energy consumption per segment for fixed intervals, such as 60 seconds, is created using the Google Maps API. Energy and CO₂ emission heat maps refresh automatically at fixed intervals (say every 60 seconds) synchronous with the simulation run. Figure 5 depicts the described architecture used for dynamic visualization of the energy performance metrics.

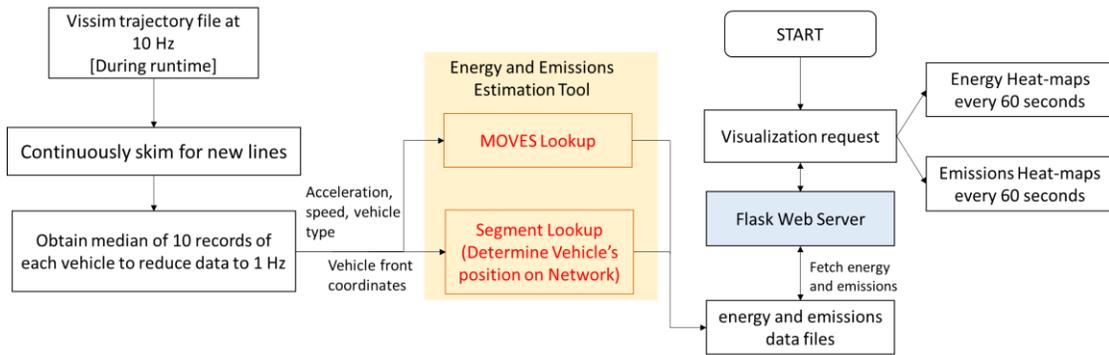


Figure 5: Working architecture for dynamic visualization of energy performance indicators.

4 RESULT AND DISCUSSION

The results presented here are two fold. The first set of results refer to the real-time outputs from the model during runtime. The second set presents the results of the test of robustness of the simulation model in the presence of real-time inputs and the test of sensitivity of the model to these real-time inputs. Due to the need for multiple iterations of the model run (with exact same input conditions) for the comparative analysis, the experiments were conducted with pre-fetched input data.

4.1 Real-time Performance Measures Computation

Real-time traffic information visualized for performance measures such as queue lengths, energy consumption, and CO₂ emissions can be used by in real time traffic operations to interpret and update eco-driving routes. With communications network available in the CV environment in smart corridors, these real-time updates can be fed back to the travelers, helping them make environment friendly route choices and encourage environmentally friendly eco driving patterns. The study for this effort was undertaken for a 3-hour PM-peak model run, during which real-time data is used to drive the simulation model. The plots for queue lengths, energy, and CO₂ emissions were created dynamically during the run as described in section 3.3. The presented model architecture simulated traffic at 1.3x rate compared to wall clock and the generated dynamic performance evaluation plot at 0.98x rate, for an hour of experiment on a dual processor machine with 192 GB RAM and 2.5 GHz frequency. Due to space limitations only an example energy heat map generated at the end of the 47th simulation minute is shown in Figure 6, presenting the cumulative energy consumption during the 46-47 simulation minute interval throughout the corridor. The points in the heat map symbolize the midpoint of 200 feet segments in which the corridor is segregated into. The heat image around each point represents the total energy consumed in the area in the last minute.



Figure 6: Energy plots are generated dynamically during simulation runtime.

4.2 Model Sensitivity to Real-Time Input

The model’s sensitivity to real-time input is tested by comparing the performance measures for simulation runs with and without real-time input at the intersections of Glen Iris Drive with North Ave and Ponce City Market with North Ave, with all other intersections utilizing the calibrated data in both scenarios. For the two scenarios vehicle travel times, energy consumption, and CO₂ emissions performance measures were measured for eight selected routes that traverse the two intersections.

4.2.1 Experiment Design

The simulation period for both real-time model and preset-input model is 3 hours (10800 sec) with simulation resolution of 10 time steps per simulation second. To reflect randomness 10 replicate runs were undertaken for each scenario, where the Vissim Random Seed parameter is changed in each replicate. This allowed stochastic variation of vehicle arrivals into the network. Eight routes that go across the real-time driven intersections – Glen Iris and Ponce City Market are selected to specifically investigate the built model’s sensitivity to real-time input. Figure 7 shows the selected eight routes, of which the four westbound routes start at the Freedom Pkwy intersection and four eastbound routes start near the I75/85 Connector. Additionally, for the eight selected routes, energy consumption and CO₂ emissions with simulation time are compared.

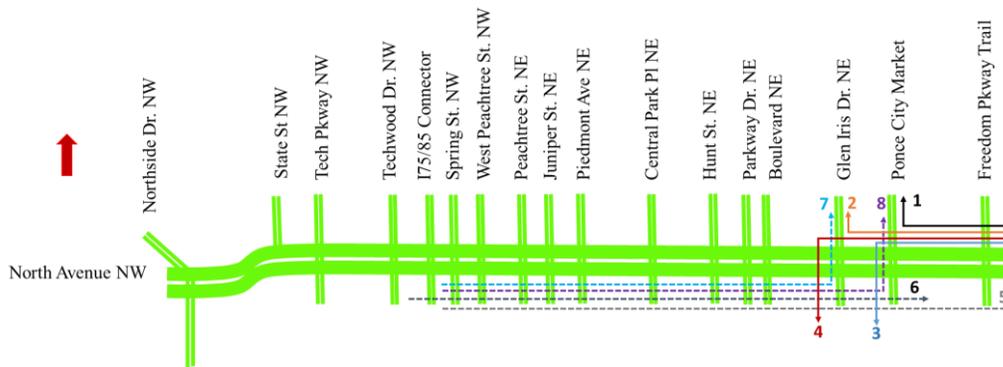


Figure 7: Routes selected for studying model’s sensitivity to real-time input.

4.2.2 Vehicle Travel Time

For brevity, results are given for two routes. Figure 8(a) below shows boxplots of vehicle travel time versus simulation time interval (15 minute intervals) for the 10 random runs with real-time input and with calibrated data on westbound route 4. Figure 8(b) shows the same for eastbound route 6. These box plots show the average vehicle travel times of all vehicles that end their trip on the respective routes, for the corresponding simulation interval on x-axis.

For route 4, it is observed that the vehicle travel times for the real-time model and the preset-input model are comparable. The two sets of travel times fluctuate similarly over the simulation period. For route 6, the difference in travel times between real-time and preset-input simulation is larger compared to that observed in route 4; however, the values are still comparable. The maximum difference in average travel times is of approx. 60 seconds (1 minute) for the simulation interval of 135-150 min. Among the 8 selected routes, route 4 and route 6 shown in the figures here, recorded the maximum differences in westbound and east bound travel times. The closeness in vehicle travel time values for 10 random simulation runs across all routes for the two compared models demonstrates that the results obtained from the real-time input model are within plausible bounds, with the potential that the real-time data is allowing for improved estimations from the simulation. However, field collected travel times are needed to confirm this supposition, which is the next step in the overall project.

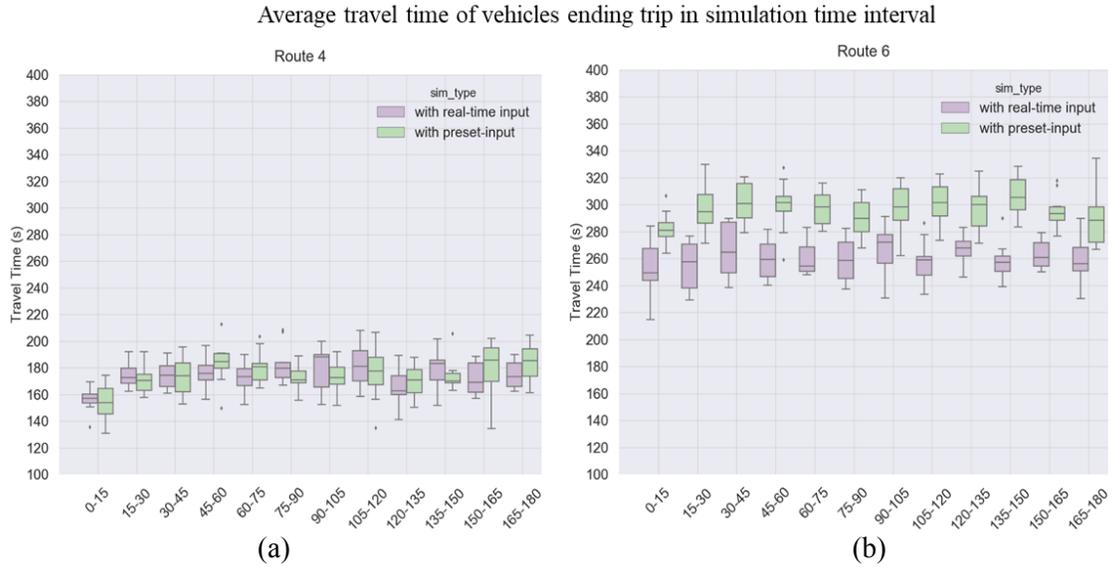


Figure 8: Average vehicle travel time versus simulation time intervals plots for (a) westbound route 4 and (b) eastbound route 6.

4.2.3 Energy Consumption and CO₂ Emission

The comparison of the energy and emissions performance metrics from one pair of runs is shown in Figure 10. The energy consumption follows a fairly linear pattern with increases in travel time (Figure 9(a)). The longer a vehicle takes to traverse a given path, the more energy it is expected to consume, under most conditions. As reflected in the cumulative density function of energy (Figure 9(b)), the slightly higher travel times in the real-time simulation for route 3 (whereas in route 4 and route 6 above real-time data driven simulation travel times tended to be lower) results in consistently higher energy estimates than the simulation without real-time data. The emissions values followed very similar trends as the energy plots. These results are to be field validated. However, closeness of results for the two scenarios shows the potential of the real-time data driven simulation to provide improved performance metric estimates.

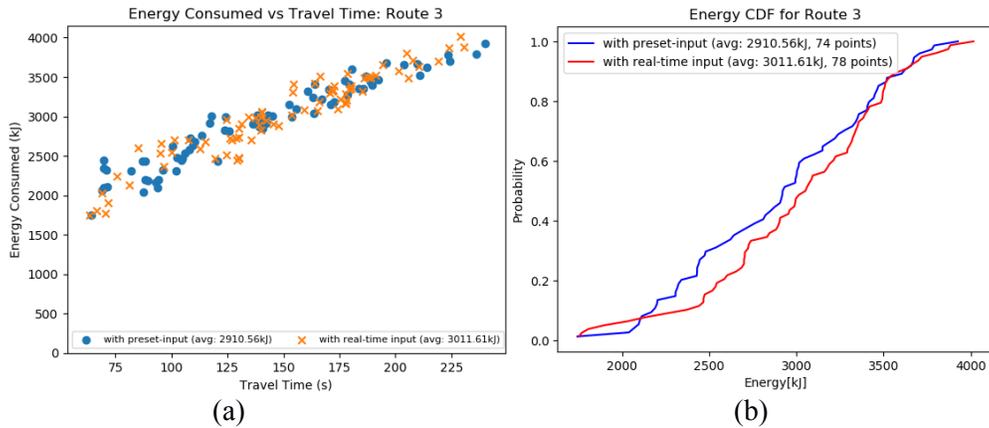


Figure 9: (a) Scatterplot for energy consumption with travel time for route 3 and (b) cumulative density function for energy consumption for vehicles route 3.

5 CONCLUSION AND FUTURE WORK

This study assesses feasibility aspects of using a real-time data-driven transportation simulation model to evaluate and visualize network performance indices that can be used to provide dynamic feedback to the real world operations environment. A hybrid traffic simulation model, consisting of a mix of pre-programmed and real-time data-driven intersections, is developed. The principal challenge identified in building the hybrid transportation simulation model is in integrating the multiple components underlying the complete model and ensuring they work in synchrony with real-time. However, an architecture was developed to enable such hybrid modeling. The simulation results from the hybrid model were compared with results from the model using preset values. The performance results indicated that with the presented architecture it is plausible to work with the high velocity data while ensuring sufficient responsiveness of the model to input changes.

However, this architecture is not without limitations. The architecture developed so far uses a single instance on a simulation model. In general, traffic microscopic simulation models do not inherently use a parallel architecture; hence, they do not scale well with increases in the size of the network. As the size of the network increases, the model will run slower and slower until a point is reached where the simulation clock will be slower than the wall clock. If wide area simulation is required, then the architecture will need to integrate a distributed simulation structure.

Future work includes expanding the corridor simulation model to incorporate all available real-time data and optimizing the dynamic performance metric visualization component of the model architecture to facilitate real-time plot generation. Further, the simulation results will be validated by field data obtained from DSRC equipped vehicles to be driven on the corridor.

ACKNOWLEDGMENTS

The research work presented here is sponsored in part by the Renew Atlanta Program, City of Atlanta, Georgia, USA and the Air Force Office of Scientific Research Award FA9550-17-1-022. Opinions expressed here are of the authors and not necessarily of the sponsors.

REFERENCES

- Allström, A., J. Barcelo, J. Ekström, E. Grumert, D. Gundlegård and C. Rydergren 2017. "Traffic Management for Smart Cities". In *Designing, Developing, and Facilitating Smart Cities* (1st ed.), edited by V. Angelakis et al., 211-240. Springer International Publishing.
- Amini, S., I. Gerostathopoulos and C. Prehofer. 2017. "Big Data Analytics Architecture for Real-time Traffic Control". In *Proceedings of the 2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, June 26th - 28th, Napoli, Italy, 710-715.
- Brügmann, J., M. Schreckenberg and W. Luther (2013). "Real-Time Traffic Information System Using Microscopic Traffic Simulation". In *Proceedings of the 2013 8th EUROSIM Congress on Modelling and Simulation*, September 10th-13th, Cardiff, UK, 448-453.
- Chen, S. and L. Du. 2017. "Simulation Study of the Impact of Local Real-time Traffic Information Provision Strategy in Connected Vehicle Systems". *International Journal of Transportation Science and Technology* 6(4): 229-239.
- RenewAtlantaBond. 2018. RENEWATLANTATSPLOST. <http://renewatlantabond.com/>, accessed 24.05.2018.
- Guensler, R., H. Liu, Y. Xu, A. Akanser, D. Kim, M. P. Hunter and M. O. Rodgers. 2017. "Energy Consumption and Emissions Modeling of Individual Vehicles". *Transportation Research Record: Journal of the Transportation Research Board* 2627: 93-102.
- Henclewood, D., A. Guin, R. Guensler, M. Hunter and R. Fujimoto. 2010. "Real-time Data Driven Arterial Simulation for Performance Measures Estimation". In *Proceedings of the 2010 Winter Simulation Conference*, edited by B. Johansson et al., 2057-2069. Piscataway, New Jersey: IEEE.

- Lv, Y., Y. Duan, W. Kang, Z. Li and F. Y. Wang. 2015. "Traffic Flow Prediction With Big Data: A Deep Learning Approach". *IEEE Transactions on Intelligent Transportation Systems* 16(2): 865-873.
- Pell, A., A. Meingast and O. Schauer. 2017. "Trends in Real-time Traffic Simulation". *Transportation Research Procedia* 25: 1477-1484.
- Pop, M.D. and O. Proştean. 2018. "A Comparison Between Smart City Approaches in Road Traffic Management". *Procedia - Social and Behavioral Sciences* 238: 29-36.
- IOT Agenda. 2017. Smart City. <https://internetofthingsagenda.techtarget.com/definition/smart-city>, accessed 16.05.2018.
- SMARTTEST. 2000. "Final Report for Publication of the SMARTTEST Project". ITS, University of Leeds (GB), England, United Kingdom.
- Sturari, M., L. Catani, A. Mancini and E. Frontoni. 2016. "An Integrated Mobility System Using Real-time Data for Traffic Simulation". In *Proceedings of the 2016 12th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, August 29th-31st, Auckland, New Zealand, 1-6.
- U.S. DOT, I. T. S. J. P. O. 2014. "The Smart/Connected City and Its Implications for Connected Transportation". John A. Volpe National Transportation Systems Center, Cambridge, Massachusetts.
- U.S. DOT, I. T. S. J. P. O. 2017. "Connected Vehicle - Safety, ITS Benefits, Costs, and Lessons Learned: 2017 Update Report". U.S. Department of Transportation, Intelligent Transportation Systems Joint Program Office, Washington, DC.
- U.S. DOT, I. T. S. J. P. O. (n.d.). Beyond Traffic: The Smart City Challenge. <https://www.its.dot.gov/factsheets/smartcity.htm>, accessed 16.05.2018.
- U.S. DOT, I. T. S. J. P. O. (n.d.). Connected Vehicle Basics. https://www.its.dot.gov/cv_basics/cv_basics_what.htm, accessed 16.05.2018.
- Wismans, L., E. de Romph, K. Friso and K. Zantema. 2014. "Real Time Traffic Models, Decision Support for Traffic Management". *Procedia Environmental Sciences* 22: 220-235.
- Xiaowen, D., M. A. Ferman and R. P. Roesser. 2003. "A Simulation Evaluation of a Real-time Traffic Information System Using Probe Vehicles". In *Proceedings of the 2003 IEEE International Conference on Intelligent Transportation Systems*, October 12th-15th, Shanghai, China, 1:475-480.

AUTHOR BIOGRAPHIES

ABHILASHA SAROJ is a Ph.D. student in Civil and Environmental Engineering at the Georgia Institute of Technology. Her e-mail address is asaroj3@gatech.edu.

SOMDUT ROY is a Ph.D. student in Civil and Environmental Engineering at the Georgia Institute of Technology. His e-mail address is somdut.roy@gatech.edu.

RICHARD FUJIMOTO is a Regents' Professor in Computational Science and Engineering at the Georgia Institute of Technology. He received a Ph.D. in Computer Science & Electrical Engineering from the University of California-Berkeley in 1983. His e-mail address is fujimoto@cc.gatech.edu.

ANGSHUMAN GUIN is a Senior Research Engineer in Civil and Environmental Engineering at the Georgia Institute of Technology. He received his Ph.D. in Civil and Environmental Engineering from the Georgia Institute of Technology. His email address is angshuman.guin@ce.gatech.edu.

MICHAEL HUNTER is a Professor in Civil and Environmental Engineering at the Georgia Institute of Technology. He received his Ph.D. in Civil Engineering from The University of Texas at Austin. His email address is michael.hunter@ce.gatech.edu.