



Final Report

# Enabling GLOSA through Domain Knowledge Aware SPAT Prediction and Queue Length Aware Trajectory Optimization

**Amr Shafik**
Virginia Tech Transportation Institute
Phone: (540) 231-1500; Email: ashafik@vt.edu

**Seifeldeen Eteifa**
Virginia Tech Transportation Institute
Phone: (540) 231-1500; Email: seteifa@vt.edu

**Hesham A. Rakha**
Virginia Polytechnic Institute and State University
Charles E. Via, Jr. Department of Civil and Environmental Engineering
Virginia Tech Transportation Institute
Phone: (540) 231-1505; Fax: (540) 231-1555; Email: HRakha@vt.edu

Date
August 2024

Prepared for the Sustainable Mobility and Accessibility Regional Transportation Equity Research Center, Morgan State University, CBEIS 327, 1700 E. Coldspring Lane, Baltimore, MD 21251

# ACKNOWLEDGMENT

*This research was supported by the Sustainable Mobility and Accessibility Regional Transportation Equity Research Center at Morgan State University and the University Transportation Center(s) Program of the U.S. Department of Transportation.*

# Disclaimer

*The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.*

| 1. Report No. SM | 2. Government Accession No. | 3. Recipient's Catalog No. |
|---|---|---|
| **4. Title and Subtitle**<br>Enabling GLOSA through Domain Knowledge Aware SPAT Prediction and Queue Length Aware Trajectory Optimization | | **5. Report Date**<br>August 29, 2024<br>**6. Performing Organization Code** |
| **7. Author(s)** *Include ORCID #*<br>Amr Shafik (https://orcid.org/0000-0002-8689-9955)<br>Seifeldeen Eteifa (https://orcid.org/0000-0001-8650-1452)<br>Hesham Rakha (https://orcid.org/0000-0002-5845-2929) | | **8. Performing Organization Report No.** |
| **9. Performing Organization Name and Address**<br>Virginia Tech Transportation Institute<br>3500 Transportation Research Plaza, Blacksburg, VA 24061 | | **10. Work Unit No.**<br>**11. Contract or Grant No.**<br>69A3552348303 |
| **12. Sponsoring Agency Name and Address**<br>US Department of Transportation<br>Office of the Secretary-Research<br>UTC Program, RDT-30<br>1200 New Jersey Ave., SE<br>Washington, DC 20590 | | **13. Type of Report and Period Covered**<br>Final, September 2023 - August 2024<br>**14. Sponsoring Agency Code** |

**15. Supplementary Notes**


**16. Abstract**

This research presents a novel approach to enhancing Green Light Optimal Speed Advisory (GLOSA) systems by predicting signal phase and timing (SPaT) switching times and assessing the confidence in these predictions. The proposed architecture leverages transformer encoders, combined with various deep learning methods such as Multilayer Perceptrons (MLP), Long-Short Term Memory (LSTM) networks, and Convolutional LSTM (CNN-LSTM) networks, to form an ensemble of predictors. This ensemble is utilized to predict SPaT information for six intersections along the Gallows Road corridor in Virginia, addressing three primary tasks: predicting phase changes within 20 seconds, determining the exact switching time, and assigning a confidence level to these predictions. The experiments demonstrate that the transformer-based architecture outperforms traditional deep learning methods, achieving 96% accuracy in phase change prediction and a mean absolute error (MAE) of 1.49 seconds in exact time prediction. The ensemble predictions, particularly those with high consensus, are highly accurate, being within one second of the true value 90.2% of the time.

In parallel, the study explores the implementation of GLOSA in proximity to actuated traffic signals, focusing on optimizing vehicle trajectories by incorporating real-time queue estimation from loop-detector and probe vehicle data. Simulation experiments evaluate the fuel savings potential and system performance both at the individual vehicle level and across the network. The results reveal substantial fuel savings of up to 35.7% for individual vehicles when considering queueing, with a saving of 10% for the case of an isolated intersection. However, the integration of queue estimation, while beneficial for individual vehicles, does not lead to significant network-wide performance improvements. This research underscores both the advantages and limitations of GLOSA systems, highlighting the importance of integrating real-time traffic data to optimize vehicle trajectories and improve system performance in real-world traffic scenarios.


| 17. Key Words: actuated traffic signals, eco-driving, optimizing vehicle trajectories, queue length estimation, stochastic optimization, uncertain switching times, Eco-CAC, Signal Phasing and Timing Data; traffic signal timing prediction; machine learning; ensemble machine learning techniques. | 18. Distribution Statement | | |
|---|---|---|---|
| **19. Security Classif. (of this report) :** Unclassified | **20. Security Classif. (of this page)** Unclassified | **21. No. of Pages** 51 | **22. Price** |

# Table of Contents

# Abstract

This research presents a novel approach to enhancing Green Light Optimal Speed Advisory (GLOSA) systems by predicting signal phase and timing (SPaT) switching times and assessing the confidence in these predictions. The proposed architecture leverages transformer encoders, combined with various deep learning methods such as Multilayer Perceptrons (MLP), Long-Short Term Memory (LSTM) networks, and Convolutional LSTM (CNN-LSTM) networks, to form an ensemble of predictors. This ensemble is utilized to predict SPaT information for six intersections along the Gallows Road corridor in Virginia, addressing three primary tasks: predicting phase changes within 20 seconds, determining the exact switching time, and assigning a confidence level to these predictions. The experiments demonstrate that the transformer-based architecture outperforms traditional deep learning methods, achieving 96% accuracy in phase change prediction and a mean absolute error (MAE) of 1.49 seconds in exact time prediction. The ensemble predictions, particularly those with high consensus, are highly accurate, being within one second of the true value 90.2% of the time.

In parallel, the study explores the implementation of GLOSA in proximity to actuated traffic signals, focusing on optimizing vehicle trajectories by incorporating real-time queue estimation from loop-detector and probe vehicle data. Simulation experiments evaluate the fuel savings potential and system performance both at the individual vehicle level and across the network. The results reveal substantial fuel savings of up to 35.7% for individual vehicles when considering queueing, with a saving of 10% for the case of an isolated intersection. However, the integration of queue estimation, while beneficial for individual vehicles, does not lead to significant network-wide performance improvements. This research underscores both the advantages and limitations of GLOSA systems, highlighting the importance of integrating real-time traffic data to optimize vehicle trajectories and improve system performance in real-world traffic scenarios.

# Introduction

Fossil fuels continue to dominate as the primary source of energy for both household and industrial uses, playing a significant role in the global economy (1). However, rising concerns over the environmental impacts of fossil fuel consumption, which is growing at an annual rate of 2%, are driving efforts to mitigate these effects. The combustion of fossil fuels, particularly in the transportation sector—which accounts for 28% of total energy consumption in the U.S. (2)—contributes significantly to greenhouse gas emissions, leading to global warming that has already exceeded 1°C (3). In response, strategies such as eco-driving have been promoted, with studies showing that improved driving behavior can reduce fuel consumption by 10% on average (4).

Furthermore, the advent of Connected and Automated Vehicles (CAVs) presents a transformative opportunity to enhance the efficiency and safety of the transportation system (5). CAVs, through Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communications, enable the exchange of critical information that can be used to optimize vehicle trajectories, reduce traffic incidents, and improve fuel efficiency (6). Leveraging these capabilities, this study focuses on enhancing vehicle trajectory planning through the development of an eco-cooperative Adaptive Cruise Control system at signalized intersections (ECO-CACC-I). This system incorporates queue estimation using shockwave theory and connected vehicle data to optimize trajectories in the vicinity of traffic signals with uncertain timings.

Central to this optimization is the use of Phasing and Timing (SPaT) messages, a key form of infrastructure-to-vehicle (I2V) communication. SPaT messages provide detailed information about traffic signal phases, including phase status and timing, which are crucial for applications that aim to improve both traffic safety and efficiency. These applications range from enhancing intersection safety through turn assist and red light warnings to improving traffic operations by reducing delays and optimizing fuel consumption through Green Light Optimal Speed Advisory (GLOSA) systems (7). By integrating real-time SPaT predictions and considering the effects of vehicle queues, this study evaluates the network-wide impacts of the ECO-CACC-I system, demonstrating its potential to significantly improve traffic flow and reduce environmental impacts in mixed traffic scenarios (8–12).

# Literature Review

## Vehicle Trajectory Optimization

The optimization of vehicle trajectories has been a well-explored topic in the literature. In 1977, Schwarzkopf and Leipnik (13) introduced a feedback algorithm that calculates the optimal vehicle speed to minimize fuel consumption on varying road grades, applying Pontryagin's maximum principle to determine the optimal speed under a trip time constraint set by the driver. Recent studies have focused on the impact of optimizing speed trajectories near signalized intersections, often considering fixed-time traffic signals (14–18). Kamalanathsharma and Rakha (10) developed an Eco-Cooperative Adaptive Cruise Control (ECACC) system using infrastructure-to-vehicle (I2V) communication to optimize trajectories near fixed signals. This system was later refined by Yang et al. (19) and Chen et al. (20) to account for queue effects in single-lane approaches. Their studies demonstrated that incorporating queue effects via shockwave theory in trajectory planning could result in fuel savings of up to 16% without queue consideration and up to 18% when queues are considered. However, these studies only addressed fixed signal timings and occasionally recommended crawling speeds under certain conditions, depending on the switching time, initial speed, and activation distance. Field tests by Almannaa and Chen et al. (21,22) showed that ECACC systems could achieve average fuel savings of 31% and travel time reductions of 9%. Additionally, research by Chen and Rakha (23) indicated that battery electric vehicles (BEVs) could achieve energy savings of approximately 9.3% using similar optimization strategies.

Fewer studies have examined the uncertainties in actuated traffic signal timings. Mintsis et al. (24) evaluated the effects of eco-driving on transportation network performance through a microscopic simulation model, which tested an enhanced velocity planning algorithm (EVPA). The study found that while EVPAs could significantly reduce emissions and improve comfort and safety, their effectiveness at the network level could decrease by up to 80% compared to individual vehicle-level improvements depending on factors such as intersection characteristics, signal plans, activation distance, and traffic conditions. Chen et al. (20) developed an eco-driving approach for vehicular platoons that considered queues and uncertain signal timings, showing notable fuel economy improvements near actuated signals. Zhai et al. (25) introduced an eco-driving algorithm for CAVs at signalized intersections, factoring in queue effects and employing shockwave theory for queue length estimation to optimize fuel economy. However, this study was limited to low traffic volumes and did not consider the uncertainties of actuated signal timings. Shafik et al. (8,11,12,26) presented an ECO-CACC system that accounts for uncertain signal timings, with results showing potential fuel savings of up to 51%, with an average of 30% when considering various roadway grades and actual switching times. However, this study was limited to a single vehicle approaching a signalized intersection without accounting for surrounding traffic.

## SPaT Prediction

Researchers have explored various methods for predicting Signal Phase and Timing (SPaT) information, particularly the most likely switching times, using probabilistic, statistical, and machine learning-based approaches.

*Probabilistic Approaches:* Bodenheimer et al. employed a Bayesian graph-based method that converts signal states into graphs and calculates the probabilities and most likely timings of phase transitions, focusing on a single intersection (27). Similarly, Ibrahim et al. used conditional expectation and confidence-based predictions for SPaT, also concentrating on a single intersection (28).

*Statistical Approaches:* Van de Vyvere et al. utilized historical frequency distributions of SPaT data, using means and medians as predictors, and determined that data from the past 20 minutes was most relevant (29). Moghimi et al. developed a sparse multivariate time series model to predict cycle lengths across a corridor (30).

*Machine Learning Approaches:* These methods are the most prevalent in SPaT prediction. Weisheit and Hoyer applied support vector machines (SVM) to determine switching times in traffic signals, selecting from three potential values (31). Genser et al. compared linear regression, random forests, and long short-term memory neural networks (LSTM), finding that LSTM provided superior performance (32). Eteifa et al. implemented an LSTM-based regression approach with a novel loss function to balance short-term and long-term predictions (33). Islam et al. introduced a convolutional long-short term memory neural network (CNN-LSTM) architecture, comparing it with multiple machine learning models, including multilayer perceptrons (MLP), extreme gradient boosting (XGB), gated recurrent units (GRU), and LSTM across two different corridors. The study found that CNN-LSTM performed best, though only slightly better than the vanilla LSTM (34).

## Queue Length Estimation

Queue length estimation in the literature is commonly addressed using two primary methods: shockwave theory and the input-output method (35). Shockwave theory estimates the space occupied by the queue, while the input-output method calculates the number of vehicles in the queue. Several studies have aimed to enhance queue estimation using probe vehicle data. Comert (36) developed analytical models to improve cycle-based queue estimates at traffic signals using probe vehicles. Similarly, Tan et al. (37) utilized a maximum likelihood method with historical probe data for queue length estimation. Liu et al. (38) focused on identifying stop states from probe vehicle data to estimate queue length over time. Tavafoghi et al. (39) proposed an algorithm for cycle-based queue length estimation effective at low connected vehicle (CV) penetration rates (1.5%), relying on historical queue trends for accurate real-time predictions. Badillo et al. (40) combined loop detectors with probe vehicle data to enhance shockwave-based queue length estimation, using CV sensors to determine both the front and back of the queue. Zhang et al. (41)

applied Bayesian deduction and CV-enhanced traffic volume estimates at low penetration rates to accurately estimate queue length at signalized intersections.

## Problem Statement

This research expands on the algorithm proposed by Shafik et al. (8), enhancing it to address the specific requirements of Green Light Optimal Speed Advisory (GLOSA) and eco-driving applications. The proposed algorithm incorporates real-time queue estimation using upstream loop detector data in combination with probe vehicle data, improving its robustness under varying traffic conditions. Given the critical importance of accurate Signal Phase and Timing (SPaT) predictions for eco-driving, the enhanced system is designed to perform reliably even with missing data, which can occur as the distance from the transmitter increases or as traffic density rises. The prediction accuracy is optimized for vehicles within 300 meters of the transmitter, the recommended range for DSRC communications and 4G LTE, corresponding to about 19.2 seconds before a signal change at 56 km/h (35 mi/h). Additionally, the system increases prediction accuracy as the time to the green phase decreases and includes a confidence measure for each prediction. This allows vehicles to make informed decisions on whether to accelerate or decelerate based on the uncertainty level. The developed ECO-CACC-I system is tested for its effectiveness in improving fuel economy, focusing on a single vehicle approaching an actuated signal with uncertain timings and queues. Comprehensive simulations are conducted for isolated traffic signals, considering market penetration levels (MPLs) from 0% to 100%. The following sections outline the Spat prediction, queue estimation process and the vehicle trajectory optimization algorithm.

## Research Significance and Contribution

This research makes significant advancements in SPaT prediction, particularly for Green Light Optimal Speed Advisory (GLOSA) and eco-driving applications. The study contributes to the body of knowledge in the following ways:

1. Novel Transformer-Based Architecture: This research introduces a new model architecture for SPaT predictions using transformer encoder blocks. To the best of the authors' knowledge, this is the first time such an architecture has been applied in this context, and the results demonstrate superior predictive power compared to previously used deep learning models.

2. Innovative Ensemble Approach for Prediction Certainty: The study presents a novel data-driven method for quantifying the certainty of model predictions by leveraging an ensemble approach. The consensus or variance among the models serves as a proxy for confidence in the predictions, offering a robust mechanism to mitigate the issue of model hallucinations, a common challenge in deep learning applications.

3. Comprehensive Data Utilization: The study is grounded in over a year's worth of data from Virginia, marking the first time such an extensive dataset has been used for SPaT prediction in this region. This long-term data collection captures a wide range of temporal variability and rare traffic events, enhancing the model's reliability.
4. Development of ECO-CACC-I System: The research extends the ECO-CACC-I system to address uncertainties in traffic signal switching times and surrounding traffic. By integrating real-time queue length estimation, the system maximizes potential fuel savings in actual traffic conditions.
5. CV-Enhanced Queue Estimation Algorithm: The study develops a real-time queue tail estimation algorithm that combines shockwave theory with connected vehicle (CV) trajectories. This method addresses the limitations of traditional shockwave analysis, such as instantaneous vehicle deceleration and acceleration, and operates without relying on historical queue data.
6. Demonstration of Fuel Savings: The integration of the CV-enhanced queue tail estimation within the ECO-CACC-I algorithm shows significant fuel savings from a single vehicle's perspective, surpassing the results achieved using standalone shockwave theory.
7. Evaluation of Queue Considerations in Optimization: The research quantifies the benefits and limitations of incorporating queue considerations into the stochastic optimization logic, providing valuable insights into potential fuel savings compared to scenarios where queues are not accounted for.

Overall, this research offers a comprehensive approach to improving SPaT prediction and vehicle trajectory optimization, with significant implications for eco-driving and GLOSA applications.

# Methodology

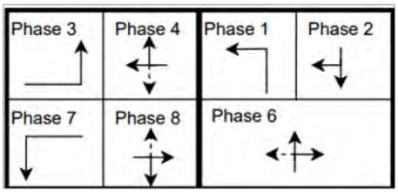
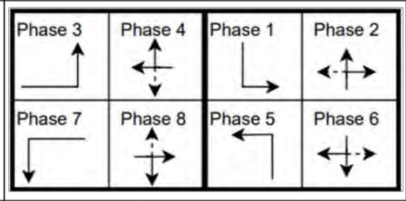
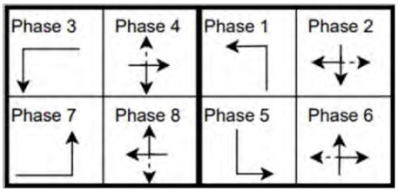
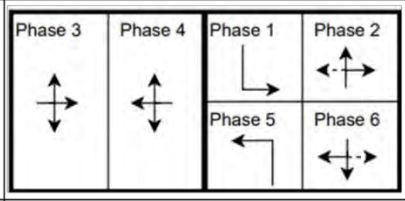
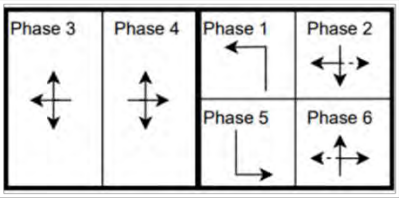
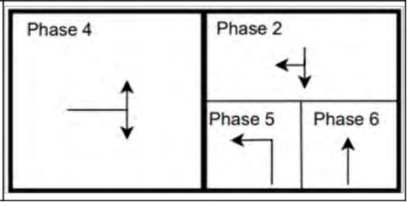
### Data Description for SPaT Prediction

Data were collected from six intersections along Gallows Road in Northern Virginia, as illustrated in Figure 1. This arterial road connects various land use areas, including residential, commercial, and office spaces. The studied intersections experience heavy traffic throughout the day. These six signalized intersections feature diverse phasing schemes, ranging from four to eight phases, as detailed in Table 1. Intersection 650075, a T-intersection, operates with four phases. Intersections 650063 and 650065 have six-phase operations with split phasing on the side roads, where each road discharges traffic in all directions during its own phase. Intersection 650058 operates with seven phases, notably prohibiting the southbound left turn. The other two intersections, 650060 and 650064, have eight-phase operations. The phasing plans, as depicted in the figure, do not strictly follow the NEMA phasing scheme and are represented as dictated by the data. Notably, the two sides of the barrier phases are reversed because phases 2 and 6 conclude each cycle. This setup primarily facilitates coordination with other traffic signals along the arterial and ensures that these phases are served for the entire cycle when no actuations are detected.


**Figure 1: The 6 intersections with Gallows Road in Northern Virginia from which data was collected**

The data used in this study were sourced from the Virginia SmarterRoads web portal. The data were broadcast via a web API every second, and a custom script was created to capture the data in JSON format, building a database of traffic signal timings. The data, generated by traffic signal controllers, provide detailed insights into the controller's state. However, due to communication losses between the controller and the web API, and then from the API to our servers, up to 30 percent of data might be missing on certain days. These missing values are treated as such in the model without imputation, reflecting the data losses that occur in infrastructure-to-vehicle communication. Figure 2 outlines the unique data elements in the VDOT SmarterRoads dataset.

**Table 1: Phasing schemes for all 6 intersections studied**

| Intersection | Ring Barrier Diagram | Intersection | Ring Barrier Diagram |
|---|---|---|---|
| 650058 |  | 650064 |  |
| 650060 |  | 650065 |  |
| 650063 |  | 650075 |  |

| Data Elements | |
|---|---|
| Time Variables | Includes temporal embedding for time of day, time spent within the current phase and time spent within the current cycle |
| Controller State | Includes detailed data of controller status like for the current phase whether it is in minimum green or passage time. It also include the termination method for each phase like gapping out when there are no longer any actuations or maxing out when max green is reached |
| Detector State | Includes the actuations received as well as speed, volume and occupancy for each lane |
| Pedestrian Data | Includes pedestrian (walk/ don't walk) indications and the status of pedestrian push buttons |

**Figure 2: Different variables in SmarterRoads data**

It's important to note that the complexity of SPaT prediction varies significantly across the intersections studied. This variability can be attributed to factors such as differences in traffic signal controller logic, driver behavior, intersection geometry, and phasing schemes. The intersections in question used D4 traffic signal controllers, and the data presented specific challenges for prediction. These challenges include:

1. Data sequences with up to 30 percent missing information, impacting prediction accuracy but providing a realistic representation of infrastructure-to-vehicle communication.
2. Controller logic that frequently skips left-turn and side-road phases, leading to highly stochastic predictions  where a single vehicle arriving on a secondary road can drastically alter the time to green on the primary road.
3. A controller logic feature across all six intersections where, during some cycles, if no actuations occur on any approach, the entire cycle serves phases 2 and 6, which cover primary road through, right, and permissive left movements. This extends the time to green by an entire cycle, complicating long-term prediction efforts.

**SPaT Prediction Models Development**

Four machine learning models were fitted and compared for tasks 1 and 2, as introduced in the methodology section. These include three models previously utilized in literature: MLP, LSTM, and CNN-LSTM. Additionally, a new architecture based on Transformer Encoder Blocks is proposed, which will be referred to as the Transformer in this paper for brevity, even though it lacks the decoder component of a full transformer model.

*Elements Common to All Models*

All four models were implemented using the PyTorch deep learning library (42) and were formulated as regression models with outputs comprising PP features, where PP represents the number of phases being predicted at the intersections. These output features correspond to the residual time to green (TTG) or time to red (TTR) for each of the PP traffic signal phases. The models utilized the Adam optimizer, consistent with previous deep learning models for SPaT prediction. Two general hyperparameters—batch size and learning rate—were tested across all models. The input for all models was standardized as an F×SF×S input sequence, where FF denotes the number of features and SS represents the length of the time window selected to generate the input sequence. Various time window values, including 3, 10, 30, 60, and 120 seconds, were tested, with the best results obtained using a 10-second time window from the past. The Mean Absolute Percentage Error (MAPE) was employed as the loss function for all models.

$$MAPE = \frac{|y_{pred} - y_{true}|}{y_{true}} \times 100 \quad (1)$$

MAPE was chosen as the loss function for all models because it heavily penalizes errors when predictions are close to zero, enhancing prediction accuracy for near-future events. This is particularly important for traffic signal timing predictions, where reliable short-term forecasts are crucial for applications like GLOSA. A prior study on intersection 650065 demonstrated that MAPE performs best for short-term predictions, aligning with the focus of GLOSA applications (33).

To optimize model performance, hyperparameters were tuned using the Raytune distributed model selection library (43). The tuning process employed the Asynchronous Successive Halving (ASHA) algorithm, which aggressively stops less promising model configurations early, favoring those with better performance to maximize computational efficiency (44). Each model type was allocated 8 hours of computational time, using 2 Nvidia Tesla A100 GPUs to select the optimal hyperparameters. The model variants were trained for a maximum of 10 epochs, with up to 10 model variants trained in parallel. This approach allowed hundreds of configurations to be tested for each model type, ensuring the selection of the best possible configurations.

*MLP*

MLPs represent the simplest neural network architecture used as a baseline in this study. The multilayer perceptron was designed to process the same 10-second data sequence as the other models. In this setup, each of the 10 time steps in the sequence was treated as an independent feature, with separate weights assigned to each step. This approach means the model receives data from the last 10 seconds without considering the temporal order or giving more weight to more recent information. Consequently, the model must determine the importance of each time step's data on its own, allowing it to focus on relevant data from earlier time steps if needed.

The MLP model has two primary hyperparameters: the number of nodes per layer (N) and the number of dense layers with sigmoid activation (n_dense). Although alternative activation functions like ReLU were tested, sigmoid activation proved to be the most effective for this application.

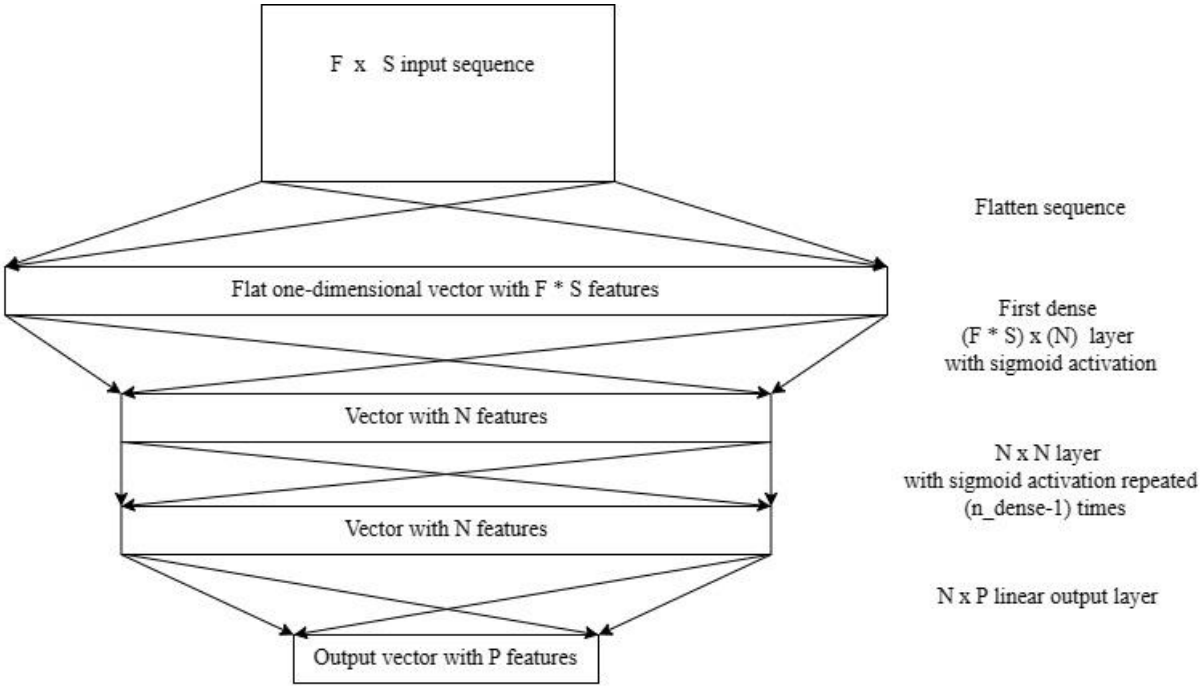



**Figure 3. MLP Neural network architecture**

*LSTM*

LSTM neural networks have been effectively applied in this context and have demonstrated strong performance in similar applications (32,33,45). LSTM networks are particularly adept at capturing temporal dependencies between data elements across different time steps. This capability allows LSTM networks to recognize patterns and relationships between various factors such as signal states, traffic volumes, speeds, vehicle arrivals, and pedestrian movements. By considering both short-term dependencies and long-term trends, LSTM networks enhance prediction accuracy.

For this study, we utilized the LSTM architecture proposed by Graves in 2013 (32). The LSTM cell is defined by the following five equations:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (2)$$

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right) \quad (3)$$

$$c_t = f_t c_{t-1} + i_t \, tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (4)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \quad (5)$$

$$h_t = o_t \, tanh(c_t) \quad (6)$$

In this model, the variable $x_t$ represents the input at time *t*, while the input, output, and forget gates are denoted as *i*, *o*, and *f*, respectively. The internal LSTM cell, where information is stored, is referred to as *c*, and the hidden state is represented by *h*. Weights are denoted by *W* and biases by *b*, with the Tanh activation function applied within the LSTM layers.

The model's hyperparameters include:

- **N**: The number of features in the LSTM sequence, the output vector of the LSTM, and the number of neurons in the dense layer.
- **n_lstm**: The number of LSTM layers.
- **n_dense**: The number of dense layers using sigmoid activation that aggregate the LSTM output.

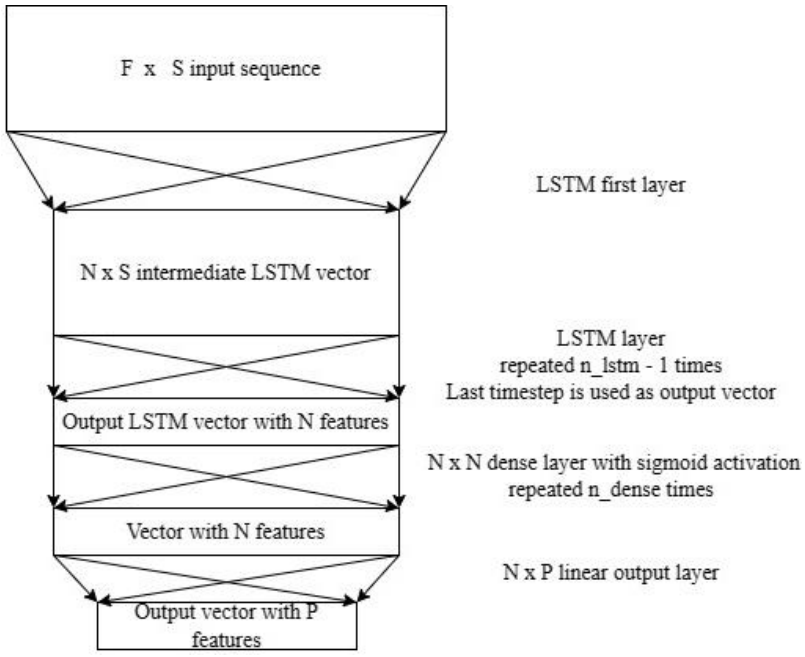



**Figure 4. LSTM Neural Network Architecture**

*CNN-LSTM*

The CNN-LSTM model, although similar in prediction performance to a simple LSTM, was previously suggested as a robust method for SPaT prediction (34). Two-dimensional convolutions, however, were found to be less effective for aggregating features in this context. Unlike in computer vision, where features are spatially interrelated, the features from traffic signal controllers do not exhibit spatial relationships. Consequently, the output of a two-dimensional CNN-LSTM model would be overly dependent on the specific order of the features. Therefore, a one-dimensional CNN-LSTM is more suitable for this problem, as it effectively captures trends across each feature and integrates them into the LSTM to address temporal dependencies.

The CNN-LSTM model involves five key hyperparameters:

- **C**: The filter size for the convolution.
- **n_conv**: The number of convolution layers.
- **n_lstm**: The number of LSTM layers.
- **n_dense**: The number of dense layers.
- **N**: The number of neurons in each layer, including both the LSTM cells and the dense layers.

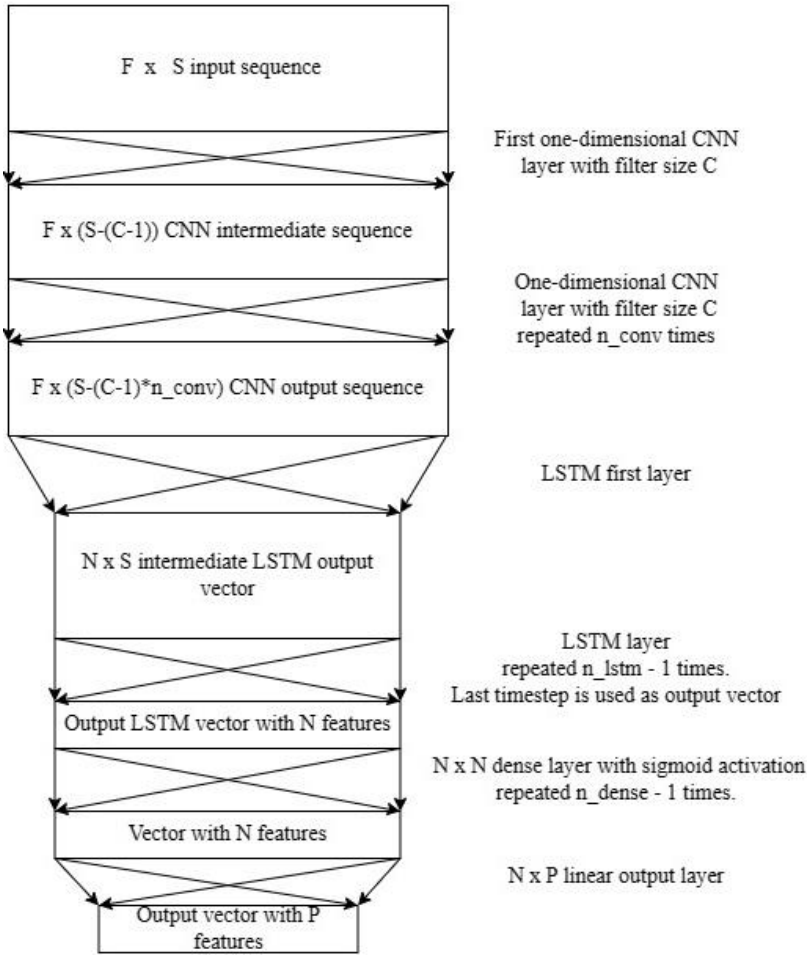



**Figure 5. CNNLSTM Neural Network Architecture**

*Transformer*

A typical transformer architecture includes both an encoder and a decoder block. The encoder processes the input sequence with self-attention mechanisms to capture relationships between time steps, while the decoder performs a similar function for the output sequence to establish a mapping between the input and output. However, the architecture proposed here, as illustrated in Figure 6, uses only the encoder block of the transformer. It replaces the decoder block with dense layers that use sigmoid activations and a final linear layer for regression. In this setup, the encoder block focuses on self-attention within the input sequence, as the output is a single vector representing the residual time to green or time to red for each phase rather than a sequence.

The encoder block utilized in this study follows the structure outlined by Vaswani et al. (46). Key hyperparameters for this model include:

- **N**: The size of the input embedding, the transformer output, and the dense layers before the final output layer.
- **n_heads**: The number of attention heads in each multi-head attention mechanism within the encoder.
- **n_encoder**: The number of stacked encoder layers.
- **n_dense**: The number of dense layers with sigmoid activations that aggregate the transformer encoder output before the final linear layer.

Figure 7 shows the transformer encoder block architecture, adapted from Vaswani et al. (46).

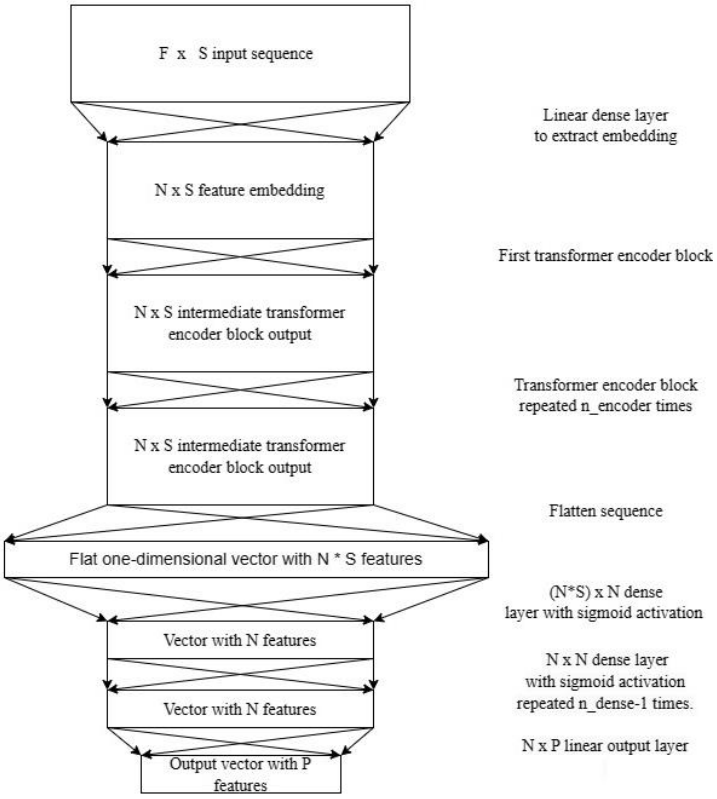



**Figure 6. Transformer encoder-based architecture**

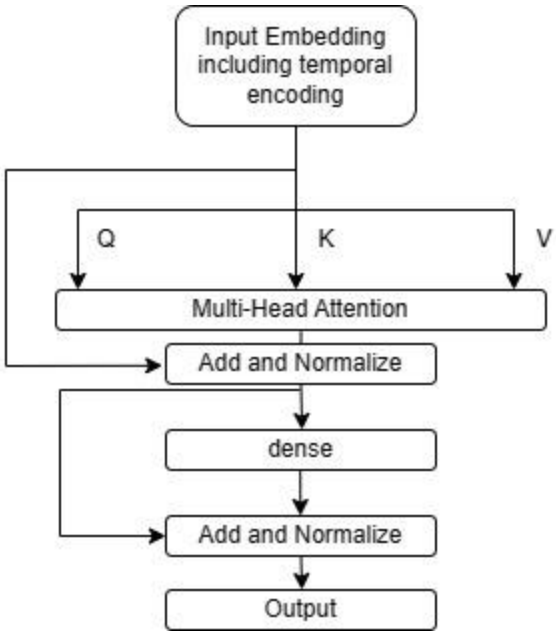



**Figure 7. Encoder block architecture**

In this context, the variables $Q$, $K$, and $V$ represent the query, key, and value, respectively. These are linear projections of the input embedding, which includes temporal encoding for each attention head. For each attention head, the dot product attention can be computed using the following formula (46):

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{7}$$

In this mechanism, the matrix multiplication between $Q$ (query) and $K$ (key) quantifies the similarity between these matrices, with the result being scaled by a scaling factor equal to the square root of the dimension of the key matrix $d_k$ for normalization. This scaling ensures that the dot product does not grow too large and maintains numerical stability. The purpose of this dot product attention is to compute self-attention, which helps the model identify which time steps are more relevant for predicting the time to change and which are less relevant.

The dot product of $Q$ and $K$ serves as a similarity measure, allowing the model to assess the importance of each time step in relation to the prediction task. This similarity is then passed through the softmax function, which enhances the differences between the self-attention weights, making it easier for the model to focus on the most relevant time steps.

The result is then multiplied by the value matrix $V$ to produce the final output of the dot product attention. This output highlights the features of the input sequence, giving more emphasis to the time steps that are deemed most important for the prediction.

*Ensemble Approach*

The four models—MLP, LSTM, CNN-LSTM, and Transformer—differ greatly in their data handling and computational approaches. This diversity in model logic allows us to leverage ensemble-based prediction methods. To ensure effective ensemble performance, diversity among the constituent models is crucial (47). This diversity is achieved in two ways:

1. Architectural Variation: By employing four distinct architectures, the data undergoes various processing methods and logic, contributing to the overall diversity of the ensemble. Each model architecture processes data differently, adding to the ensemble's logical diversity.
2. Hyperparameter Diversity: For each model, multiple variants with varying numbers of parameters are explored during hyperparameter tuning. This variability in model size and parameter count introduces additional complexity and diversity among the models within the ensemble.

The ensemble approach for SPaT information is based on two key hypotheses:

Hypothesis 1: Diverse predictions from multiple models can enhance prediction performance by reducing the impact of outlier predictions or "hallucinations." Since each model is unlikely to produce the same incorrect prediction due to differing biases or variances, an ensemble can mitigate errors that are specific to individual models.

Hypothesis 2: Consensus among ensemble models can serve as a strong confidence indicator for predictions. Predictions that are agreed upon by a majority of models are expected to be more accurate. This can be tested by analyzing error distributions relative to the level of consensus; as consensus increases, the mean error should decrease, and the proportion of distant outlier predictions should decline.

**Real-Time Back of the Queue Estimation**

*Overview*

In this study, accurate estimation of queue length is critical for optimizing a vehicle's speed profile as it approaches a traffic signal. Two primary methods are commonly employed for queue length estimation: the input-output method and the shockwave method. The input-output method estimates the vertical queue size by counting the number of vehicles, while the shockwave method assesses the space occupied by the queuing process (35).

For this research, the shockwave method is utilized, relying on upstream loop detector data to estimate the queuing process. Additionally, probe vehicle data is incorporated via V2V communication to refine the queue length estimates.

The fundamental diagram in Figure 8 illustrates the three states in the queuing process:

- State A: Represents the approaching flow with initial density.
- State B: Occurs when the signal turns red, and vehicles stop at the stop bar, creating a queue with jam density $K_j$. The queue continues to grow until the signal turns green.
- State C: Represents the queue dissipation phase, where the queue dissipates at the road's saturation flow rate.

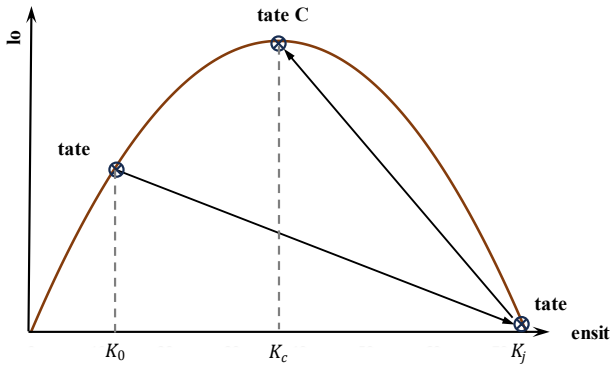



**Figure 8: Fundamental Diagram**

The shockwave theory calculates the growth speed of the queue, $v_{AB}$, based on the arrival flow $q_0$, arrival density $K_0$, and jam density $K_j$. This speed indicates how rapidly the queue forms. Additionally, the queue dissipation speed, $v_{BC}$, is determined using the capacity flow rate $q_c$, critical density $K_c$, and jam density $K_j$. Figure 9 illustrates the backward-forming shockwave and the backward queue dissipation wave at a signalized intersection.

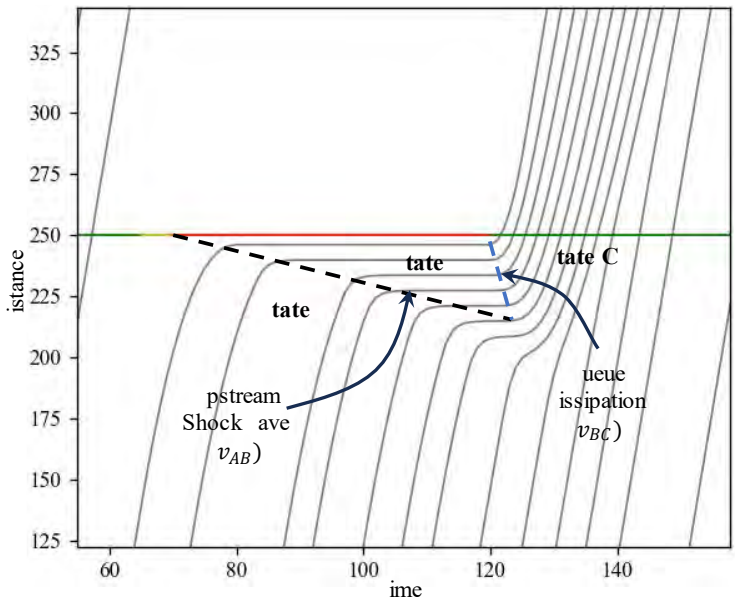



**Figure 9 Queuing Process at Signalized Intersections**

*Queue Estimation Algorithm*

In a connected vehicle (CV) environment, controlled vehicles may modify their speed profiles, leading to discrepancies between measured and actual arrival flow rates at the upstream loop detectors. This discrepancy can result in significant errors in queue length estimation, affecting the effectiveness of vehicle control strategies. To address this issue, complementary probe vehicle data are utilized to improve queue length estimates. The system shares the following information with controlled vehicles to refine the estimates:

1. **Arrival Time Estimation:** When a CV enters the network, the GLOSA system uses the current queue estimate from the shockwave method to predict the vehicle's arrival time at the end of the queue. This estimated arrival time is then used to adjust the arrival flow rate, which in turn refines the queue length estimate.
2. **Queue Length Update:** When the CV stops at the end of the queue, the queue length estimate is updated based on the CV's current location.

Algorithm 1 presents the pseudocode for the queue estimation algorithm, which integrates shockwave theory with complementary CV data.

**Algorithm 1: Queue Estimation Algorithm**

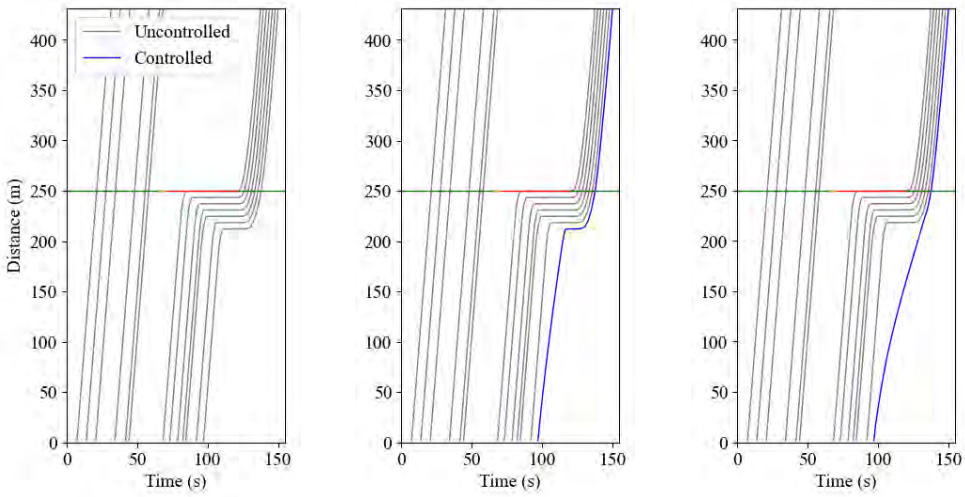
calculate the arrival flow rate and density $q_0, K_0$ using loop detector at time $t$;
**if** a CV enters the upstream link **then**
  **if** the CV will alter its speed profile according to GLOSA **then**
    calculate the estimated arrival time of this CV at the back of the queue according to GLOSA;
    calculate the modified arrival flow rate and density $q'_0, K'_0$ at time $t$;
  **else**
    continue with the detector-provided arrival flow rate and density $q_0, K_0$ at time $t$;
  **end if**
**else**
  continue with the detector-provided arrival flow rate and density $q_0, K_0$ at time $t$;
**end if**
calculate the shockwave speed $v_{AB}$ at time $t$;
calculate the queue length at time $t$;

## Optimizing Vehicle Trajectories Considering Queue Effects

*System Overview*

As the literature indicates, optimizing vehicle trajectories can significantly reduce fuel consumption and greenhouse gas emissions. However, the impact of trajectory optimization for vehicles approaching an actuated signal controller is often overlooked, especially when considering surrounding traffic. This paper addresses this gap by formulating the optimization problem within the context of queue impacts, expanding on the work by Shafik, Eteifa, and Rakha (8), which reported fuel savings of up to 51% with an average of 30%.

Figure 10a displays the trajectory of an uninformed driver, while Figure 10b and Figure 10c show optimized trajectories for a vehicle approaching an intersection with a queue. Figure 10b depicts an optimized trajectory that does not account for the queue, resulting in the vehicle idling at the end of the queue. In contrast, Figure 10c illustrates an optimized trajectory where the control algorithm incorporates queue length and clearing time, enabling the vehicle to avoid stopping at the back of the queue and idling until the queue clears.




**(a)** **(b)** **(c)**

**Figure 10: Comparison between Trajectory Planning Alternatives a) Uncontrolled Vehicle, b) Controlled without Consideration of Queue, c) Controlled with Consideration of Queue**

The following sections will detail the stochastic optimization problem formulation followed by an overview of the underlying models used in this study, including the car-following model, vehicle dynamics model, and fuel consumption models.

*The A* Algorithm*

The A* algorithm, a widely used pathfinding method, estimates the least-cost acceleration/deceleration policy for vehicle trajectory planning by assuming that the chosen policy remains constant over time. For each permissible acceleration policy, the A* algorithm assumes that the same level of acceleration or deceleration will be maintained until the vehicle reaches its destination, whether that is the stop bar for the upstream section or the end of the downstream link.

The heuristic cost is computed for both upstream and downstream sections by iterating through feasible acceleration policies and selecting the one that results in the minimum total fuel consumption. Policies that violate constraints are deemed infeasible. This process ensures that the trajectory optimization achieves the lowest possible fuel consumption from the starting point to the destination. For additional details on the application of the A* algorithm in this research, please refer to the literature.

*The Vehicle Control Algorithm*

In this application, the A* algorithm is applied to both the upstream and downstream sections. For the upstream section, it determines the optimal acceleration at each time step to minimize fuel consumption, while ensuring that constraints related to red lights and vehicle dynamics are respected. Downstream of the traffic signal, the algorithm calculates the optimal acceleration policy based on the vehicle's speed at the end of the upstream section and includes a speed jerk

constraint to maintain a comfortable acceleration. This iterative process is performed at each time step to derive the fuel-optimal acceleration profile.

Algorithm 2 outlines the outer loop of the ECO-CACC-I system implementation. This system continuously monitors the traffic signal status to estimate the vehicle's next state variable. It is constrained by the maximum deceleration limits from the car-following model to prevent collisions with other vehicles. Additionally, to address the uncertainty in signal switching times, a safety strategy is employed to prevent running red lights or making abrupt stops. Specifically, the critical stopping distance $d_{cr}$ is calculated and compared with the current distance to the intersection $D_i$ at each time step. When $D_i$ is less than $d_{cr}$, the system initiates deceleration at a maximum comfortable rate α\alphaα, set at 1.3 m/s² (48).

**Algorithm 2: The GLOSA Algorithm**

**Input:** $S_0$: controlled vehicle's initial state;
$D_i$: distance to intersection;
$S_l$: leader vehicle state;
**Output:** the update controlled vehicle state;
**for** each time step $t_i$ **do**
  check remaining distance to intersection ($D_i$);
  compute the minimum stopping distance ($d_{cr}$);
  **if** signal light is red **then**
    **if** $D_i < d_{cr}$ **then**
      **while** signal light is red **then**
        **do** decelerate with $\alpha$;
      **end while;**
    **else**
      **if** signal light is red **then**
        check the expected time to green $t_s$;
        generate next upstream state variable using A* algorithm ($S_{i+1}$);
        check and validate the generated state $S_{i+1}$ with car-following model;
      **else**
        generate next downstream state variable using A* algorithm ($S_{i+1}$);
        check and validate the generated state $S_{i+1}$ with car-following model;
      **end if**
    **end if**
  **end if**

*Stochastic Problem Formulation*

The optimization problem is framed as a stochastic optimal control problem to handle the uncertainty in traffic signal switching times at actuated signal controllers. The objective function, detailed in Equation (8) aims to minimize the total expected fuel consumption from the initial time point $t_0$ to the final time point $t_f$, with $a_t$ denoting the instantaneous vehicle acceleration at time t.

Equation (9) describes the relationship between speed, time, and total traveled distance, where $X_t$ represents the distance traveled from the start until time t. Equation (10) specifies the total traveled distance from the beginning until the expected signal time $\mathbb{E}(t_s)$.

Equation (11) introduces the upstream safety constraint, with $\alpha$ indicating the maximum deceleration level (-0.6 m/s²). This constraint ensures the vehicle does not run a red light by preventing acceleration when the distance to the stop bar is less than or equal to $d_{cr,t}$, which is the minimum distance required to stop comfortably. $X_M$ represents the location of the stop bar.

Equation 12 defines the downstream distance constraint, where "downstream" refers to the distance from the vehicle's position when the signal turns green ($X_s$) at time $t_s$ to the destination location ($X_N$).

Equation (13) establishes the kinematic relationship between the vehicle's speed at time $t$ and $t + \Delta t$. Equation (14) imposes a speed limit constraint. Finally, Equation (15) represents the acceleration jerk constraint.

$$\text{Min} \sum_{t=t_0}^{t_f} \mathbb{E}\big(FC_t(a_t)\big) \cdot \Delta t \quad (8)$$

Subject to:

$$X_t = \sum_{t_0}^{t} x_t = \sum_{t_0}^{t} v_t \cdot \Delta t \quad (9)$$

$$X_s = \sum_{t=t_0}^{\mathbb{E}(t_s)} v_t \cdot \Delta t \le X_M \quad (10)$$

$$v_{t+\Delta t} = v_t + \alpha \cdot \Delta t \quad \forall\, t \le \mathbb{E}(t_s), \forall\, X_t \; where \; X_M - X_t \le d_{cr,t} \quad (11)$$

$$\sum_{t=t_s}^{t_f} v_t \cdot \Delta t = X_N - X_s \quad (12)$$

$$v_{t+\Delta t} = v_t + a_t \cdot \Delta t \quad \forall t \in \{t_0, \dots, t_f\} \quad (13)$$

$$v_t \le v_{lim} \quad \forall t \in \{t_0, \dots, t_f\} \quad (14)$$

$$a_{t+\Delta t} \le a_t + 1.3 \cdot \Delta t \quad \forall t \in [t_0, \dots, t_f], \; \forall\, f_{b,t} > 0 \quad (15)$$

## Underlying Systems

### *Vehicle Dynamics Model*

The vehicle dynamics model used for light-duty vehicles (49) is employed to simulate the vehicle's acceleration and deceleration behavior. Equation 16 calculates acceleration based on the tractive force $F_t$ and the resisting force $R_t$. Equation 17 describes tractive force calculation. It considers factors such as the throttle input $f_{b,t}$ driveline efficiency $\eta_d$, a constant β for gear shift effects, the vehicle mass $M_{ta}$ on the tractive axle, and the road friction coefficient μ.

Equation 18 determines the instantaneous vehicle power $P_t$. Equation 19 outlines the resisting forces on the vehicle, which include rolling resistance, aerodynamic drag, and grade resistance. Factors such as air density at sea level and 25°C ρ), the drag coefficient ($C_d$), the altitude correction factor ($C_h$), vehicle frontal area ($A_f$), and rolling resistance constants ($c_{r0}$, $c_{r1}$, $c_{r2}$) are included in this calculation.

$$a_t = \frac{F_t - R_t}{m} \quad (16)$$

$$F_t = \min\left[3600 f_{b,t} \eta_d \beta \frac{P_t}{v_t}, M_{ta} g \mu\right] \quad (17)$$

$$P_t = \left(\frac{R_t + 1.04\, m a_t}{3600 \eta_d}\right) v_t \quad (18)$$

$$R_t = \frac{\rho}{25.91} C_d C_h A_f {v_t}^2 + mg \frac{c_{r0}}{1000}(c_{r1} v_t + c_{r2}) + mg G_t \quad (19)$$

*Car-Following Model*

The Fadhloun-Rakha (FR) car-following model (50) is utilized in this research to represent scenarios involving uninformed drivers. This model leverages the vehicle dynamics model for light-duty vehicles and incorporates a collision avoidance strategy to ensure safe following distances. The mathematical expression for the FR model is provided in Equation 20. It applies a reduction multiplier $F$ to the vehicle's acceleration, with $F$ depending on $X_n$, a factor derived from the vehicle's steady-state speed, steady-state spacing, and its current speed and spacing (Equations 21 and 22). Additionally, the model includes a collision avoidance term CA, which accounts for the desired deceleration rate and the kinematic deceleration required to align the following vehicle's speed it h that of the leading vehicle Equations 23 and 24).

$$a_n = F(X_n)\, a_n^{DYN} + CA(v_n, s_n, \Delta u_n) \quad (20)$$

$$F(X_n) = e^{-aX_n}\left(1 - X_n^b e^{b(1-X_n)}\right)^d \quad (21)$$

$$X_n = \frac{s_n^{VA}}{s_n} * \frac{v_n}{v_n^{VA}} \quad (22)$$

$$CA(v_n, s_n, \Delta u_n) = \frac{d^2}{(d_{desired} - gG)} \quad (23)$$

$$d = \frac{\left[v_n^2 - v_{n-1}^2 + \sqrt{(v_n^2 - v_{n-1}^2)^2}\right]}{4(s_n - s_j)} \quad (24)$$

*Fuel Consumption Model*

The Virginia Tech Comprehensive Power-Based Fuel Consumption Model (VT-CPFM) type 1 is employed to estimate instantaneous fuel consumption (51). VT-CPFM-1 is recognized for its accuracy in fuel consumption estimates, aligning well with field measurements. It effectively mitigates the issue of bang-bang control behavior and offers ease of calibration with publicly available data. The fuel consumption formula, detailed in Equation (25), incorporates calibrated model constants $\alpha_0$, $\alpha_1$, and $\alpha_2$ specific to the vehicle in use.

$$FC_t = \begin{cases} \alpha_0 + \alpha_1 P_t + \alpha_2 {P_t}^2 & \forall P_t > 0 \\ \alpha_0 & \forall P_t \leq 0 \end{cases} \quad (25)$$

# Results and Analysis

## SPaT Prediction

*Task 1: Identifying Change Times Less than 20 Seconds in the Future*

The task focuses on distinguishing between predictions made for time intervals less than 20 seconds and those extending further into the future. For this purpose, each of the 12 models produces a regression prediction, which is subsequently converted into a binary classification: a value of 1 is assigned if the predicted time is under 20 seconds and 0 if it is over 20 seconds. Regression models are chosen due to their superior performance in classification tasks, owing to their more nuanced loss function, which better captures the subtleties of the prediction problem.

Three ensemble techniques are tested to improve classification accuracy: taking the mean of the predictions, taking the median, and using a majority vote among the 12 models. Table 2 illustrates the classification accuracy achieved by each individual model at various intersections, as well as the performance of the different ensemble methods.

**Table 2: Classification accuracy of different models distinguishing whether a signal phase will change 20 seconds in the future**

| Intersection | Model Rank | LSTM | MLP | CNNLSTM | Transformer | mean | median | votes |
|---|---|---|---|---|---|---|---|---|
| 650058 | 1 | 94.84% | 94.97% | 94.37% | **95.31%** | **95.24%** | 95.12% | 95.08% |
| | 2 | 94.59% | 94.71% | 94.63% | 95.25% | | | |
| | 3 | 94.68% | 94.82% | 94.64% | 95.14% | | | |
| 650060 | 1 | 95.20% | 95.23% | 93.47% | 95.04% | **95.53%** | 95.03% | 95.17% |
| | 2 | 94.40% | 95.11% | 93.22% | 95.09% | | | |
| | 3 | 94.20% | 95.05% | 93.58% | **95.32%** | | | |
| 650063 | 1 | 95.95% | 95.90% | 95.88% | **96.07%** | **96.17%** | 96.10% | 96.08% |
| | 2 | 95.70% | 96.01% | 95.54% | 95.96% | | | |
| | 3 | 95.87% | 95.84% | 95.91% | 96.04% | | | |
| 650064 | 1 | 94.98% | 95.84% | 94.20% | **96.59%** | **96.51%** | 96.04% | 96.17% |
| | 2 | 95.06% | 95.81% | 93.77% | 96.24% | | | |
| | 3 | 95.67% | 95.97% | 94.36% | 96.45% | | | |
| 650065 | 1 | 95.63% | 96.36% | 94.33% | **96.39%** | **96.43%** | 96.08% | 96.07% |
| | 2 | 95.88% | 95.75% | 94.18% | 96.35% | | | |
| | 3 | 95.76% | 95.92% | 94.39% | 96.08% | | | |
| 650075 | 1 | 96.01% | **96.63%** | 96.01% | 96.07% | **96.73%** | 96.07% | 95.86% |
| | 2 | 96.08% | 96.19% | 95.54% | 96.18% | | | |
| | 3 | 95.77% | 96.05% | 95.59% | 96.04% | | | |

*Task 2: Predicting the exact time to change when less than or equal to 20 seconds*

After determining whether the time to change for each phase is within 20 seconds, the subsequent task is to predict the exact time to change for each phase. Table 3 and Table 4 present the Mean Absolute Percentage Error (MAPE) and the Mean Absolute Error (MAE) of each model for the subset of data where the time to change is less than 20 seconds in the future, across all six intersections. The prediction performance of the ensemble methods, specifically the mean and median of the models, is also included in both tables

**Table 3: Mean Absolute Percentage Error of Each Model for Values under 20 seconds**

| Intersection | Model Rank | LSTM | MLP | CNNLSTM | Transformer | mean | median |
|---|---|---|---|---|---|---|---|
| 650058 | 1 | 20.45 | 16.65 | 19.40 | **15.39** | 15.58 | **15.03** |
| | 2 | 19.70 | 16.33 | 20.43 | 15.95 | | |
| | 3 | 21.15 | 16.68 | 20.90 | 15.29 | | |
| 650060 | 1 | 22.64 | 19.49 | **17.80** | 18.07 | 16.72 | **15.89** |
| | 2 | 20.90 | 19.12 | 17.84 | 19.01 | | |
| | 3 | 21.79 | 19.29 | 20.27 | 19.03 | | |
| 650063 | 1 | 22.47 | 19.65 | 20.62 | **16.29** | 16.45 | **15.61** |
| | 2 | 23.14 | 19.20 | 18.67 | 16.90 | | |
| | 3 | 22.47 | 19.19 | 21.23 | 17.56 | | |
| 650064 | 1 | 22.29 | 21.28 | 21.92 | 19.88 | 18.31 | **17.60** |
| | 2 | 23.08 | 21.18 | 22.43 | **18.75** | | |
| | 3 | 25.07 | 22.58 | 23.09 | 19.38 | | |
| 650065 | 1 | 22.74 | 19.97 | 19.65 | **17.56** | 16.70 | **16.32** |
| | 2 | 22.21 | 18.97 | 19.82 | 17.90 | | |
| | 3 | 22.54 | 19.19 | 20.33 | 17.34 | | |
| 650075 | 1 | 16.38 | 15.01 | 12.95 | **11.37** | 11.94 | **10.99** |
| | 2 | 17.64 | 13.68 | 14.96 | 13.25 | | |
| | 3 | 17.62 | 13.64 | 16.25 | 12.94 | | |

**Table 4: Mean Absolute Error of Each Model for Values under 20 seconds**

| Intersection | Model Rank | LSTM | MLP | CNNLSTM | Transformer | mean | median |
|---|---|---|---|---|---|---|---|
| 650058 | 1 | 1.585 | 1.456 | 1.572 | **1.345** | 1.330 | **1.309** |
| | 2 | 1.509 | 1.450 | 1.707 | 1.428 | | |
| | 3 | 1.636 | 1.480 | 1.614 | 1.351 | | |
| 650060 | 1 | 1.839 | 1.720 | 1.516 | 1.684 | 1.483 | **1.419** |
| | 2 | 1.619 | 1.728 | **1.501** | 1.760 | | |
| | 3 | 1.677 | 1.773 | 1.621 | 1.718 | | |
| 650063 | 1 | 1.785 | 1.647 | 1.711 | **1.473** | 1.446 | **1.428** |
| | 2 | 1.786 | 1.695 | 1.632 | 1.511 | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | 3 | 1.776 | 1.662 | 1.702 | 1.573 | | |
| 650064 | 1 | 1.843 | 1.909 | 1.864 | 1.801 | 1.644 | **1.615** |
| | 2 | 1.945 | 1.898 | 1.917 | **1.685** | | |
| | 3 | 2.051 | 2.052 | 1.954 | 1.782 | | |
| 650065 | 1 | 1.836 | 1.659 | 1.776 | 1.605 | 1.502 | **1.496** |
| | 2 | 1.757 | 1.688 | 1.811 | 1.595 | | |
| | 3 | 1.821 | 1.671 | 1.762 | **1.556** | | |
| 650075 | 1 | 1.177 | 1.194 | 1.033 | **0.962** | 0.933 | **0.888** |
| | 2 | 1.264 | 1.086 | 1.247 | 1.056 | | |
| | 3 | 1.213 | 1.074 | 1.168 | 1.005 | | |

Tables 3 and 4 demonstrate that the proposed transformer-encoder-based architecture outperforms all other architectures across all intersections except for intersection 650060, where the CNN-LSTM model slightly surpasses the transformer. In the other intersections, the CNN-LSTM is slightly outperformed by the MLP model, while the vanilla LSTM is the worst performer, though only by a small margin. All architectures show reasonably low MAPE values, ranging from 11% to 23%, indicating that the absolute error is between 11% and 23% of the predicted value. Consequently, the best models have an average deviation of about 0.96 seconds from the true value, while the worst models are off by approximately 2.05 seconds, as shown in Table 4.

The ensemble approach reveals that both the mean and median values of all models enhance prediction accuracy. However, the median value is the most effective estimator, as it is robust against outliers, unlike the mean. The median consistently provides a non-trivial improvement over the best single model prediction across all 12 models.

This finding supports Hypothesis 1 outlined at the end of the methodology section: by taking the median of predictions from 12 diverse models, errors or "hallucinations" from some models can be mitigated, resulting in a more accurate overall prediction that surpasses the performance of any individual model.

*Task 3: Assigning a level of certainty to the prediction*

In this third task, the focus is on determining the certainty level for each prediction, which is crucial yet often overlooked in existing literature. The idea is to leverage the variation in predictions made by the 12 best models to gauge certainty. Since each model has inherent errors due to bias and variance, it's rare for all models to converge on an incorrect prediction unless the error is data related.

To quantify this certainty, a metric is established based on the consensus among the models. Given a set of predictions $P$ from different models for the same input, the median prediction m is identified. A tolerance level t is set, usually around 20%, within which predictions are considered to be in agreement. The certainty level is then determined by counting the number of predictions

within this tolerance range. Instead of expressing certainty as a percentage of agreeing models, which could be misleading, the number of models in agreement is used to indicate the level of consensus and, consequently, the certainty of the prediction.

$$certainty = count\ (p) \qquad p \in P \qquad (1-t)\ m < p < (1+t)\ m \qquad (26)$$

To test this approach, a tolerance level of 5% is selected, and the number of models falling within this tolerance is evaluated. This produces 12 distinct confidence levels, ranging from 1 to 12, where 1 represents the lowest confidence (only one model agrees with the median prediction within the set tolerance) and 12 represents the highest confidence (all 12 models agree within the tolerance). The Mean Absolute Percentage Error (MAPE) corresponding to each consensus level is illustrated in Figure 11. It's important to note that the error displayed for each consensus level includes the data from that level or higher. Therefore, the MAPE for a consensus level of 1 represents the overall error across the entire dataset, as all predictions have at least one model in agreement within the tolerance.

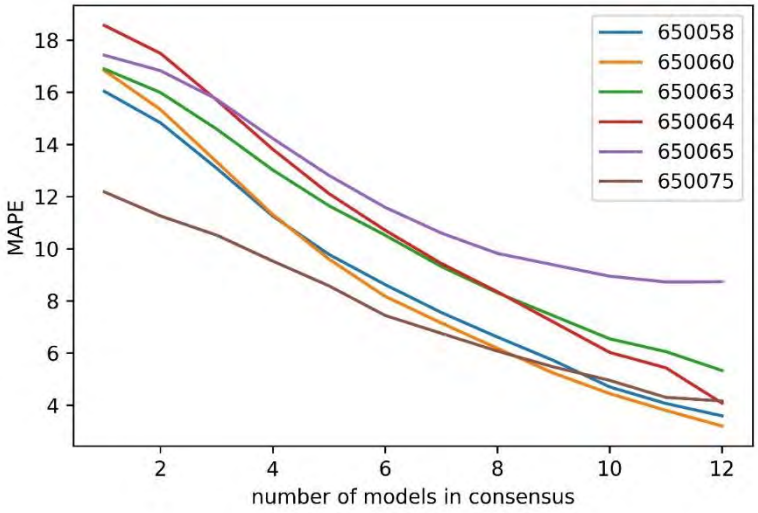



**Figure 11. MAPE as a function of the level of consensus with a 5% tolerance level for all 6 intersections**

The figure highlights the effectiveness of the consensus metric in distinguishing between predictions with varying levels of confidence. For instance, Intersection 650058 shows an overall MAPE of 16.04%, but for predictions where all 12 models reach consensus, the MAPE significantly drops to 3.59%. Similarly, Intersection 650060 exhibits the most substantial variation in MAPE, where it decreases by 80% from 16.83% to just 3.2% when the highest consensus score is achieved. For Intersection 650063, the MAPE declines from an overall 16.89% to 5.33% at a consensus score of 12. This validates Hypothesis 2 outlined at the end of the methodology section, demonstrating that the consensus score serves as a reliable metric for quantifying the uncertainty

of each prediction before the actual value is known. This consensus-based ensemble method is versatile, applicable across various model types, including both statistical and machine learning models, and can be easily integrated with other methods for determining prediction certainty, such as probabilistic predictions. Figures 12 to 15 depict the probability distribution of model-based errors across each level of consensus, from 1 to 12. The error is calculated using the following equation:

$$Error = y_{pred} - y_{true} \tag{27}$$

Where $ypred$ represents the median of the model-predicted values, and $ytrue$ is the actual ground truth. A positive error indicates that the prediction overestimates the traffic signal time to change, while a negative error indicates an underestimation. The error distribution displayed in Figures 9 to 12 shows errors ranging between -10 seconds and 10 seconds within the 20-second prediction window. Errors exceeding 10 seconds are considered distant outliers, and their corresponding percentages for each level of consensus are provided in Figure 16 for thoroughness. The errors are aggregated across all six intersections. Since predictions are rounded to the nearest second and the ground truth is measured in seconds, the values shown in the histograms in Figures 12 to 15 reflect the exact error value. For example, the central bar represents the percentage of instances where the error is zero, indicating that the rounded prediction is precisely equal to the ground truth.

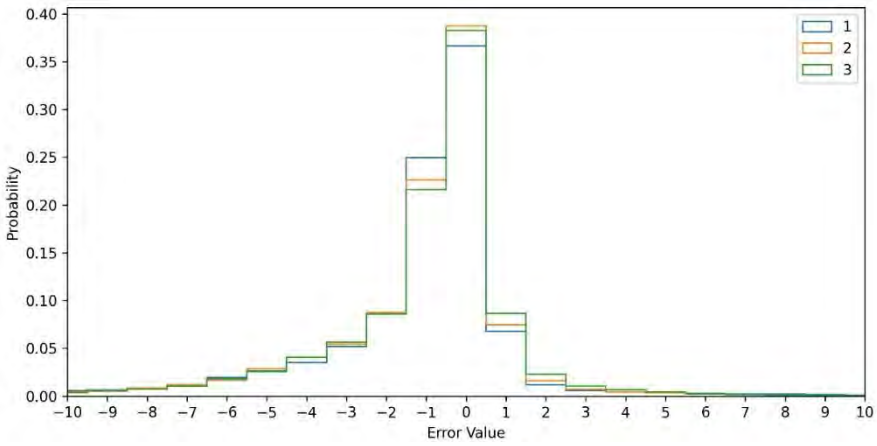



**Figure 12. The error distributions for consensus values 1 to 3**

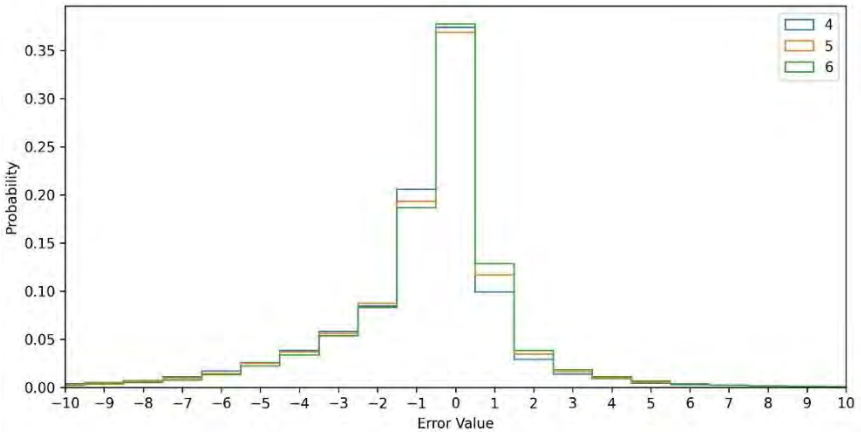



**Figure 13. The error distributions for consensus values 4 to 6**

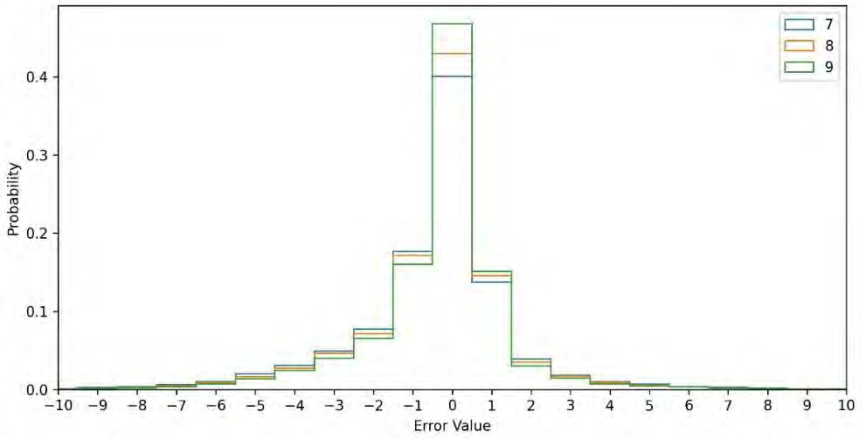



**Figure 14. The error distributions for consensus values 7 to 9**

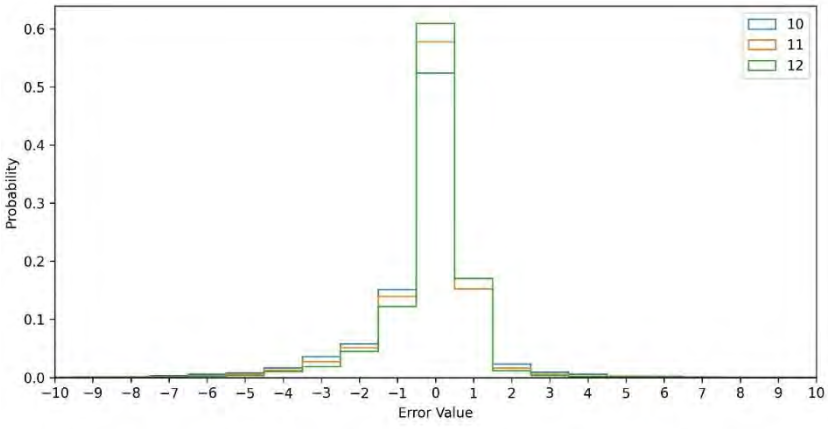



**Figure 15. The error distributions for consensus values 10 to 12**

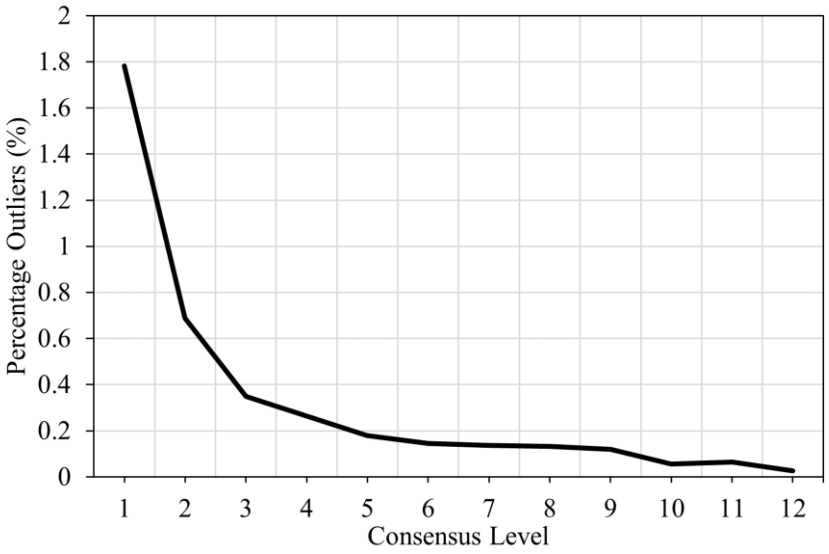



**Figure 16. Percentage of outlier predictions with error greater than 10 seconds**

*Fuel Consumption for ECO-CACC System*

To further quantify the impact of the consensus level on the overall performance of the GLOSA system, a simulation is conducted using the stochastic control algorithm developed by Shafik et al. (8) to observe its effect on a single vehicle passing through an intersection. The relationship between the consensus level and fuel consumption is depicted in Figure 17.

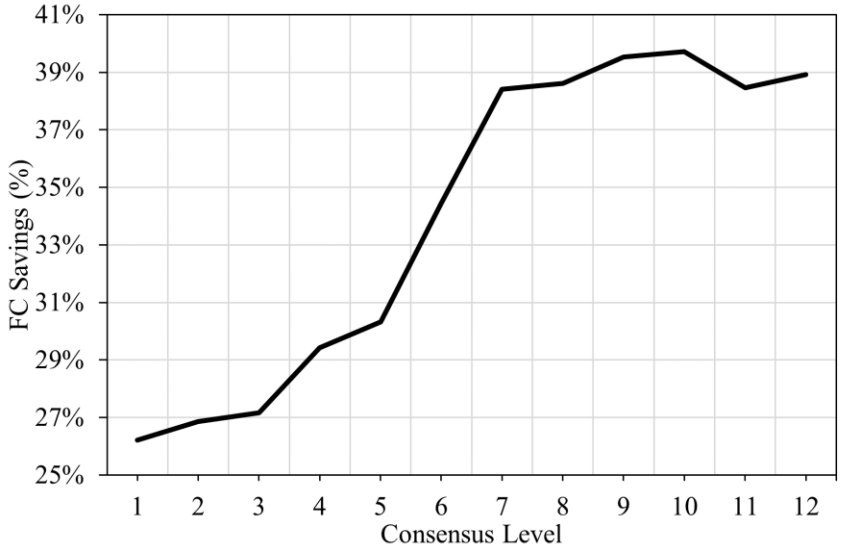



**Figure 17. The error distributions for consensus values 10 to 12**

## Queue Estimation Results

The proposed method for estimating the queue tail was tested at an isolated signalized intersection across various CV market penetration levels, ranging from 0% to 100%. Table 5 presents the accuracy of the queue tail estimation, measured by the mean absolute error (MAE) in meters and the coefficient of determination ($R^2$). The results demonstrate that the CV-enhanced estimation method significantly outperforms the shockwave theory-based approach, with estimation accuracy improving markedly at each market penetration level. These improved queue tail estimates enable the ECO-CACC-I system to more efficiently plan speed profiles in the presence of queues, as will be further detailed in the following sections.

**Table 5: Queue Estimation Accuracy Statistics per CV MP Level**

| MP Level | MAE (m) | $R^2$ |
|---|---|---|
| **0% (Only SW Theory)** | 7.02 | 0.600 |
| **10%** | 5.42 | 0.733 |
| **20%** | 4.62 | 0.758 |
| **30%** | 3.87 | 0.828 |
| **40%** | 3.69 | 0.829 |
| **50%** | 3.54 | 0.836 |
| **60%** | 2.64 | 0.884 |
| **70%** | 2.09 | 0.900 |
| **80%** | 1.27 | 0.955 |
| **90%** | 1.13 | 0.959 |
| **100%** | 0.83 | 0.970 |

## Analysis of ECO-CACC-I system

*Experimental Design*

In this section, we assess the performance of the proposed trajectory planning algorithm through a series of scenarios. Initially, we investigate how the algorithm benefits a single vehicle by examining potential fuel savings under varying entry times into the intersection. Following this, we simulate a single-lane signalized intersection with a traffic flow rate of 300 vehicles per hour and a volume-to-capacity ratio of 0.3. The upstream and downstream links measure 250 meters and 180 meters, respectively, with the speed limit set at 56 km/h (35 mph), a lane saturation flow rate of 1800 vehicles per hour, and a jam density of 160 vehicles per kilometer. Signal timings include red for 50 seconds, amber for 5 seconds, and green for 65 seconds. Loop detectors placed at the beginning of the upstream link measure the traffic inflow, and a flat grade is assumed. The algorithm is designed to replicate the uncertain switching times of actuated signals, with the signal controller providing real-time, noisy predictions of switching times within the DSRC range. These predictions are generated using an LSTM neural network based on data from Gallows Road and

Prosperity Avenue/Park Tower Drive in Northern Virginia, simulating the real-world SPaT messages vehicles would receive.

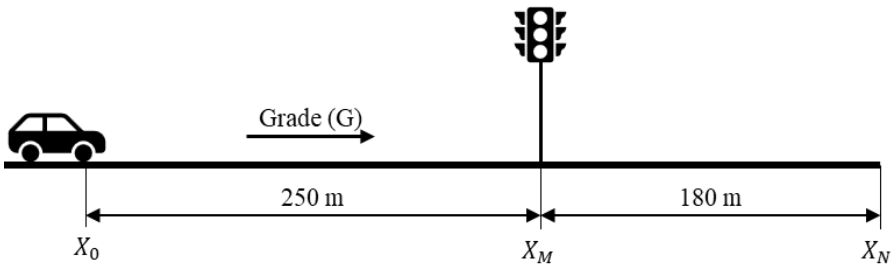



**Figure 18: Problem Setup**

*Performance of a Single Vehicle*

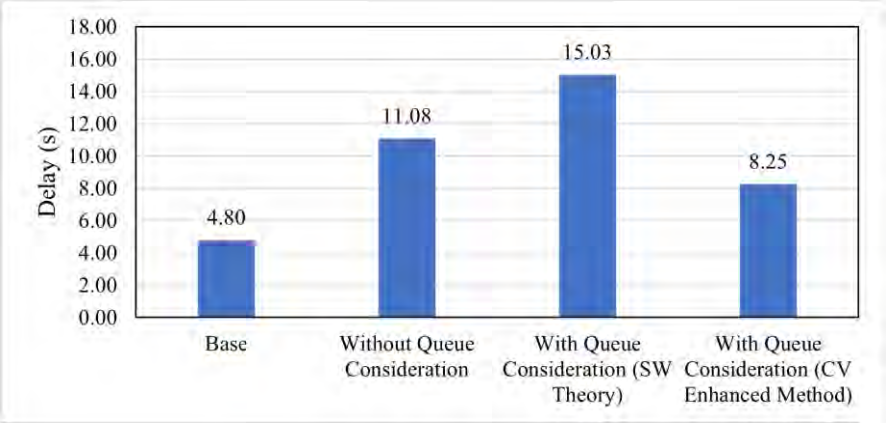
We evaluated the performance of the ECO-CACC-I system by testing it with and without considering queue effects from a single-vehicle perspective. The results, detailed in Table 6 reveal that average fuel savings improved from 28.7% to 35.7% when queue considerations were included in the optimization algorithm. Additionally, the maximum fuel savings increased from 41.8% to 46.2% with queue consideration.

The table also compares the effectiveness of queue tail estimation using shockwave theory alone versus using Connected Vehicle (CV) data. Shockwave theory alone achieved an average savings of 29.3%, while CV-enhanced queue estimation yielded 35.7% savings. Similarly, maximum savings with shockwave theory were 33.4%, compared to 46.2% with CV-enhanced estimation, highlighting the significant impact of accurate queue tail estimates on trajectory planning.

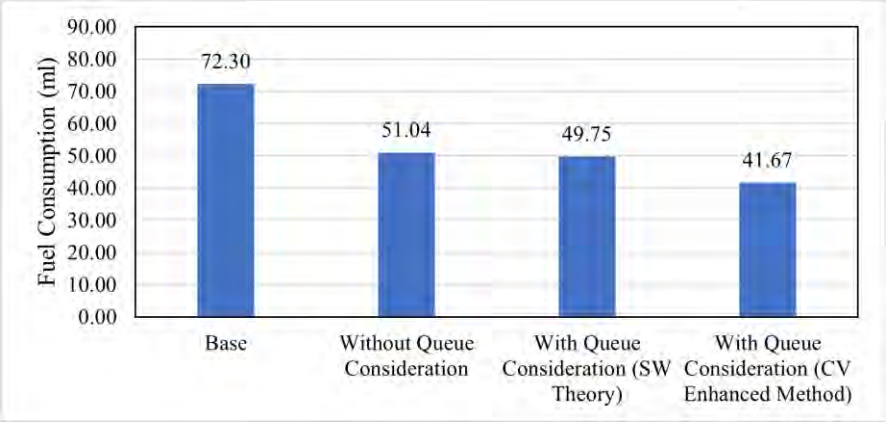
Figure 19 illustrates the delay, number of stops, and fuel consumption for an example trajectory across different scenarios. It shows that incorporating queue considerations in the optimization led to fewer stops and reduced fuel consumption compared to the base case, though it did result in increased vehicle delay. The figure also highlights the improved queue tail estimation achieved with CV data compared to shockwave theory alone.

**Table 6: Fuel Saving Statistics For A Single Controlled Vehicle**

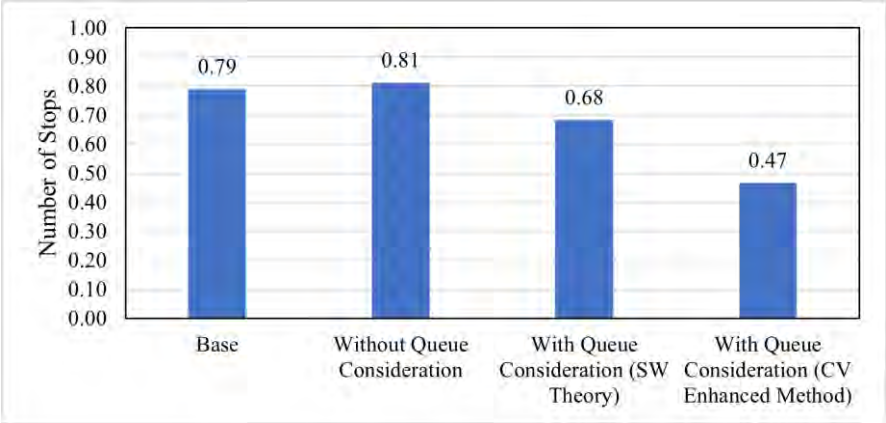
| | Average Fuel Savings | Maximum Fuel Savings |
|---|---|---|
| Without Queue Consideration | 28.7% | 41.8% |
| With Queue Consideration (SW Theory) | 29.3% | 33.4% |
| With Queue Consideration (CV Enhanced Method) | 35.7% | 46.2% |




(a)




(b)




(c)

**Figure 19: Single vehicle results: (a) vehicle delay; (b) number of stops; (c) fuel consumption**

*Performance at an Isolated Intersection*

Realistic simulations of traffic approaching an isolated intersection were conducted to evaluate the algorithm's performance across different Market Penetration Levels (MPLs) ranging from 0% to 100%. Figure 20 depicts the base case with uninformed drivers using the FR car-following model, where vehicles lack SPaT information and no trajectory optimization is applied. Figure 21 illustrates the trajectories for MPL = 50%, with (a) showing the results without queue consideration and (b) showing the results with queue consideration. Similarly, Figure 22 presents the simulated trajectories for MPL = 100%, comparing scenarios with and without queue consideration. These figures highlight that incorporating queue information in the control algorithm reduces idling time at the back of the queue, as evidenced by comparing the trajectories of vehicles 16, 17, 18, 19, 33, 39, and 40 in Figures 21 and 22.

Table 7 details fuel consumption levels for different MPLs, including the baseline (MPL = 0%), and MPLs of 10%, 50%, and 100%. The results indicate that fuel savings were achieved both with and without queue consideration. Specifically, at MPL = 100%, 13% of vehicles experienced an average fuel saving of 9%, with a maximum saving of 19% when queue consideration was included.

As detailed in Table 7, the control algorithm results in increased delays compared to the baseline across all MPL levels. This delay manifests in two regions: upstream and downstream of the stop bar. When the algorithm accounts for queues, vehicles arrive at the queue clearing time with non-zero speeds, leading to larger spacings according to the car-following model as opposed to the stationary vehicles at jam spacing. This larger spacing results in additional delay. Additionally, the optimization logic produces less aggressive acceleration, which lowers the speed profiles compared to the baseline, contributing to the increased vehicle delay. This behavior accounts for the majority of the observed delay.

**Table 7: Performance Statistics for the Isolated Intersection Simulation**

| **MOP** | **Case** | **Market Penetration Levels** | | | |
|---|---|---|---|---|---|
| | | **0%** | **10%** | **50%** | **100%** |
| **Fuel Consumption Savings (\%)** | No Queue | - | 0.5% | 2.0% | 11.2% |
| | With Queue | | 0.5% | 2.3% | 12.1% |
| **Average Delay (s)** | No Queue | 18.3 | 19.3 | 22.8 | 23.9 |
| | With Queue | | 19.2 | 22.9 | 24.3 |
| **Average Number of Stops** | No Queue | 0.563 | 0.567 | 0.607 | 0.601 |
| | With Queue | | 0.565 | 0.602 | 0.599 |

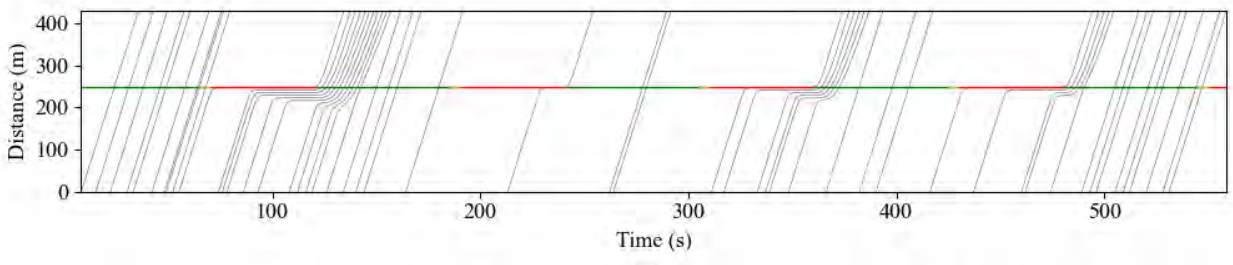
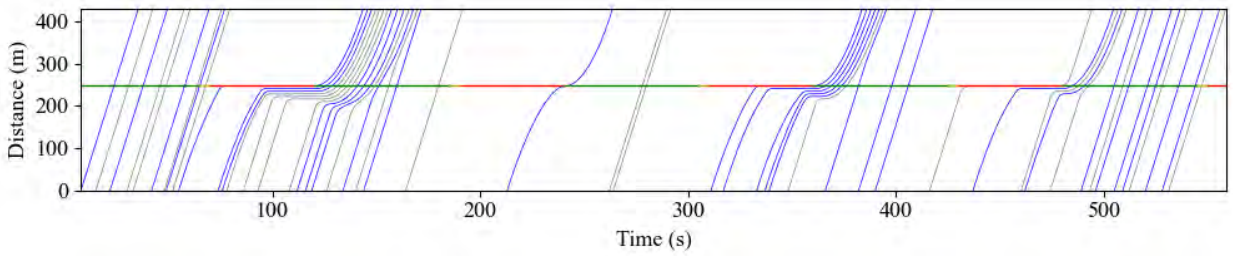
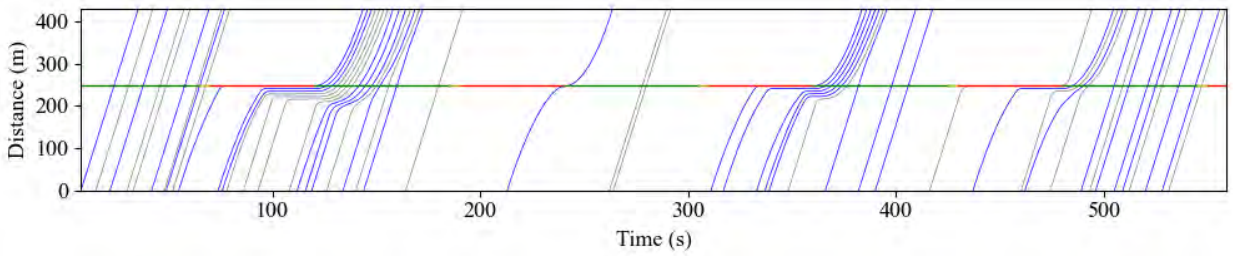



**Figure 20: Sample space-time diagram at the isolated intersection simulation with 0\% CV MPL (Base Case)**

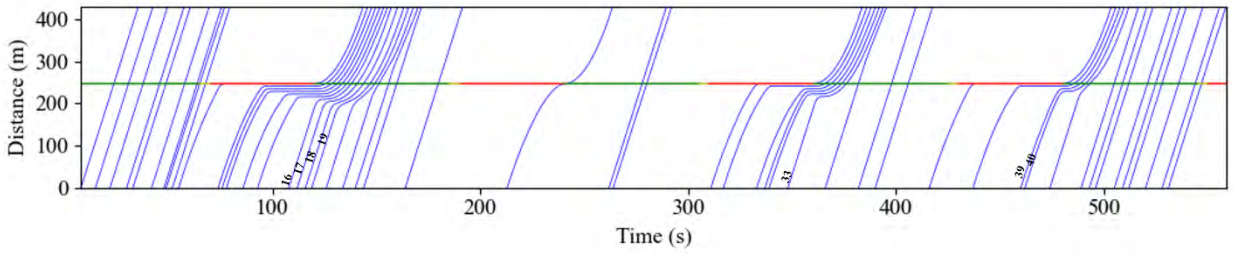



(a)

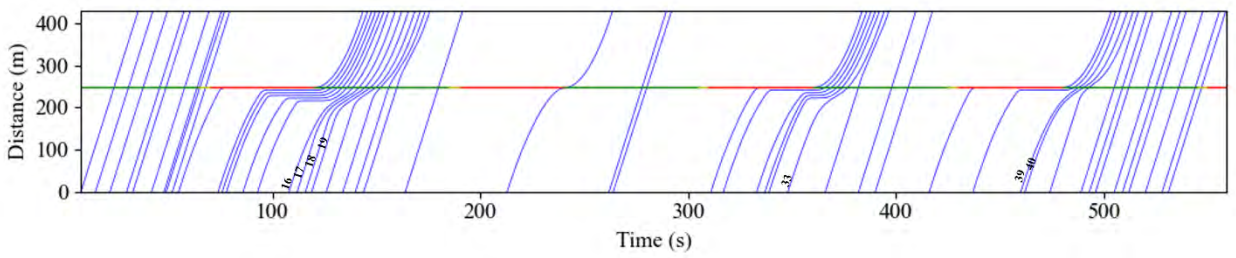



(b)

**Figure 21: Sample space-time diagram at the isolated intersection simulation with 50\% MPL: (a) without queue consideration; (b) with queue consideration.**




(a)




(b)

**Figure 22: Sample space-time diagram at the isolated intersection simulation with 100\% MPL: (a) without queue consideration; (b) with queue consideration.**

*Performance under Varying Traffic Demand Levels*

Various traffic demand levels were simulated to assess the algorithm's performance under different conditions, specifically low, medium, and high demands corresponding to volume-to-capacity (v/c) ratios of 0.3, 0.6, and 0.8, respectively. The algorithm's benefits were evaluated by comparing scenarios involving uninformed vehicles and controlled vehicles with queue consideration at a 100% Market Penetration (MP) level.

The simulation results demonstrate a decreasing trend in fuel savings as traffic demand increases. Figure 23 shows the fuel savings achieved at low, medium, and high demand levels. It is evident that higher traffic demand diminishes the potential fuel savings from upstream vehicle speed control. This reduction is due to increased vehicle interactions, which reduce the effectiveness of the algorithm and adversely affect overall traffic flow.

At low demand levels, the algorithm can optimize vehicle speeds effectively, resulting in significant fuel savings. However, as demand rises to medium and high levels, increased traffic density limits the potential for improved trajectories. These conditions hinder the algorithm's ability to maintain optimal speeds and reduce fuel consumption.

Additionally, higher demand introduces greater variability in vehicle interactions, complicating the achievement of coordinated movements that the algorithm depends on. In high-demand scenarios, controlled vehicles experience delays and disruptions due to uncoordinated actions from other vehicles, further decreasing the algorithm's effectiveness. These findings underscore the need to enhance the benefits of the algorithm through improved coordination and communication among controlled vehicles.

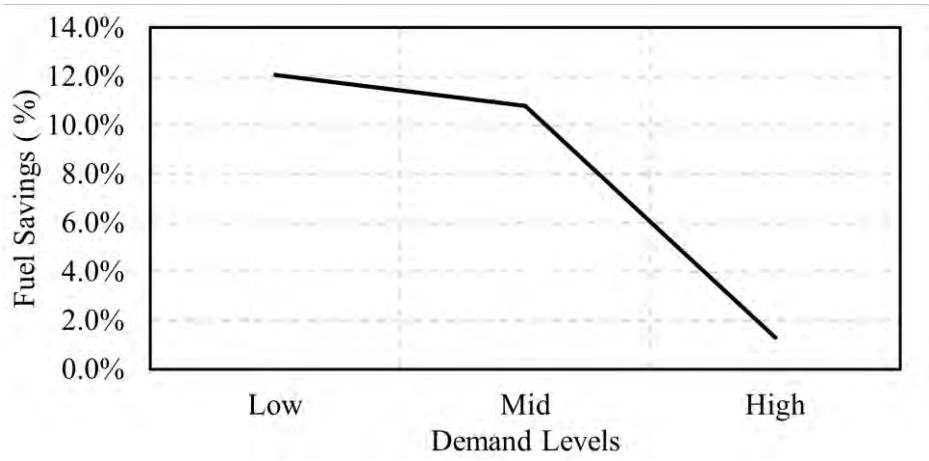



**Figure 23: Fuel savings for each Traffic Demand**

# Discussion

The proposed algorithm exhibits remarkable performance in terms of fuel savings, achieving average reductions of 28.7% and 35.7% for cases without and with queue consideration, respectively, from a single-vehicle perspective. Incorporating queue dynamics into the optimization process enhances average fuel savings by 7% compared to scenarios without queue consideration.

Figure 24 shows the fuel savings results from network simulations, comparing cases with and without queue consideration for optimal trajectory planning. The figure indicates a modest improvement of about 1% between the two approaches. However, at a 100% Market Penetration Level (MPL), 13% of vehicles experienced average and maximum fuel savings of 9% and 19%, respectively, when queue dynamics were considered. This benefit arises from the ability to optimize fuel savings within a specific spatiotemporal window relative to the signal's switching time, though the potential is minimal outside this window. In our simulations, 13% of vehicles approached the intersection during this optimal window, realizing significant savings by factoring in queue dynamics.

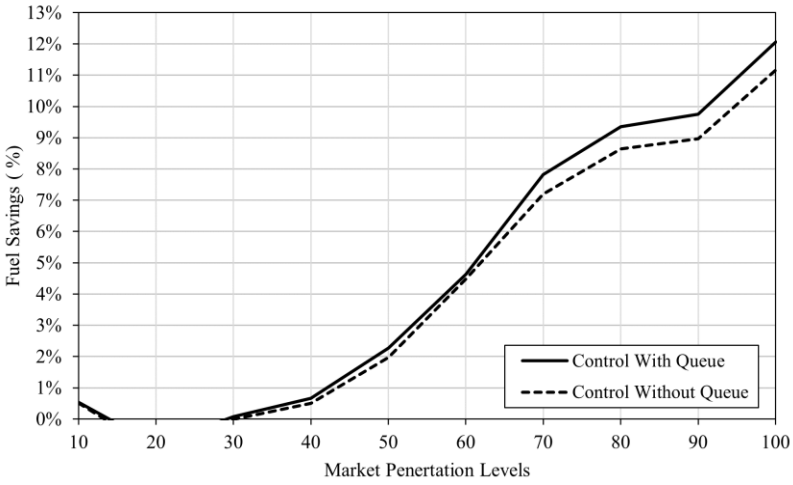
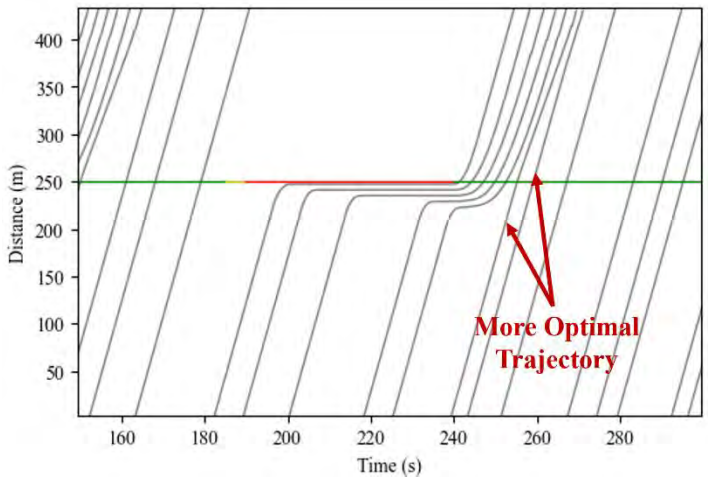


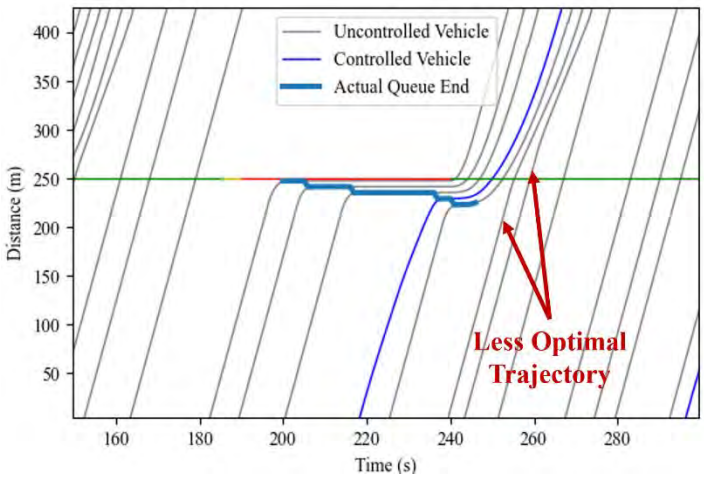

**Figure 24: Fuel savings for each MPL**

Additionally, incorporating queue considerations for a leading vehicle in a platoon can sometimes limit the solution space for following vehicles. The optimal deceleration strategy for the leading vehicle may result in less efficient trajectories for those behind it. As illustrated in Figure 25, this can force some following vehicles to decelerate, leading to less optimal overall system-wide benefits. Our simulations show an overall marginal benefit of approximately 1%, with single vehicles achieving up to 19% savings on average. Using only shockwave theory provided an 8% marginal benefit.

These findings align with previous literature (19) which reported a 2% improvement in fuel economy for a single-lane approach to a fixed-plan traffic signal when queue considerations were included in the control algorithm. This underscores the potential benefits and limitations of queue estimation on overall fuel savings. Future research could explore cooperative optimization of vehicle trajectories to enhance network-wide performance and address these limitations.




(a)




(b)

**Figure 25: Impact of vehicle control on surrounding traffic at low MPLs: (a) No Control, (b) Control with low MPL.**

# Conclusion and Recommendations

The study presents a comprehensive evaluation of an ECO-CACC-I system designed to address uncertainties in traffic signal switch times at actuated intersections in mixed traffic conditions. The system integrates real-time queue length estimations, which are enhanced through shockwave analysis and probe vehicle data. The accuracy of queue estimation was notably high, with R2R^2R2 values of 0.733 and 0.836 for Market Penetration Levels (MPLs) of 10% and 50%, respectively. When the queue dynamics were incorporated into the trajectory optimization process, the average fuel savings improved significantly, reaching 28.7% without queue consideration and 35.7% with queue consideration. The maximum savings for a single vehicle simulation achieved up to 46.2%. Furthermore, the proposed CV-enhanced queue estimation method proved to be 12.8% more effective in improving fuel economy compared to the shockwave theory alone.

The system was tested across various scenarios, including different MPLs and traffic demands. Network simulations revealed that the inclusion of queue considerations in the optimization process resulted in average fuel savings of up to 11.2% and 12.1% for scenarios without and with queue consideration, respectively. However, the marginal improvement in fuel savings was minimal when considering the queue, and the benefits decreased with increasing traffic demand. These findings illustrate the effectiveness and limitations of the ECO-CACC-I system from a network-wide perspective. The results emphasize the need for future research to explore cooperative optimization strategies that balance delay and fuel consumption, particularly in the context of consecutive actuated traffic signals.

In addition, the study evaluated a transformer encoder-based architecture for enhancing Signal Phasing and Timing (SPaT) data. This architecture was compared with other models, including MLP, LSTM, and CNN-LSTM, and was found to outperform them in two critical GLOSA tasks: predicting phase changes within the next 20 seconds and determining the exact change time within that period. The ensemble framework, which integrated three variants of each model type, further enhanced prediction performance by reducing model hallucinations and providing a more accurate measure of prediction certainty. The consensus of models indicated a significant reduction in Mean Absolute Percentage Error (MAPE), with the highest-confidence predictions achieving up to an 80% improvement over the lowest-confidence predictions. Future research directions should focus on field-testing these ensemble predictions, addressing challenges such as communication latencies and vehicle dynamics, and exploring error-aware eco-driving strategies that incorporate prediction certainty to optimize fuel savings and overall network performance.

# References


1. Shafiee S, Topal E. An econometrics view of worldwide fossil fuel consumption and the role of US. Energy Policy. 2008 Feb;36(2):775–86.

2. EIA. Use of energy for transportation - U.S. Energy Information Administration (EIA) [Internet]. 2021 [cited 2021 Jul 19]. Available from: https://www.eia.gov/energyexplained/use-of-energy/transportation.php

3. McKay DIA, Staal A, Abrams JF, Winkelmann R, Sakschewski B, Loriani S, et al. Exceeding 1.5°C global warming could trigger multiple climate tipping points. Science. 2022 Sep;377(6611).

4. Barkenbus JN. Eco-driving: An overlooked climate change initiative. Energy Policy. 2010 Feb;38(2):762–9.

5. Elliott D, Keen W, Miao L. Recent advances in connected and automated vehicles. J Traffic Transp Eng Engl Ed. 2019 Apr;6(2):109–31.

6. Zohdy IH, Kamalanathsharma RK, Rakha H. Intersection management for autonomous vehicles using iCACC. IEEE Conf Intell Transp Syst Proc ITSC. 2012;1109–14.

7. Abernethy B, Andrews S, Pruitt G. Signal phase and timing (SPaT) applications, communications requirements, communications technology potential solutions, issues and recommendations. United States. Federal Highway Administration; 2012.

8. Shafik AK, Eteifa S, Rakha HA. Optimization of Vehicle Trajectories Considering Uncertainty in Actuated Traffic Signal Timings. IEEE Trans Intell Transp Syst. 2023 Jul;24(7):7259–69.

9. Rakha H, Ahn K, Kamalanathsharma RK. Eco-vehicle speed control at signalized intersections using i2v communication. US Dep Transp Tech Rep. 2012;

10. Kamalanathsharma RK, Rakha HA. Leveraging connected vehicle technology and telematics to enhance vehicle fuel efficiency in the vicinity of signalized intersections. J Intell Transp Syst. 2014;20(1):33-44,.

11. Shafik A, Eteifa S, Rakha HA, Virginia Tech Transportation Institute, Virginia Polytechnic Institute and State University, Urban Mobility & Equity Center. Optimal Trajectory Planning Algorithm for Connected and Autonomous Vehicles towards Uncertainty of Actuated Traffic Signals [Internet]. 2023 Apr [cited 2024 May 13]. Report No.: UMEC-050. Available from: https://rosap.ntl.bts.gov/view/dot/67184

12. Shafik A, Rakha HA. Queue Length Estimation and Optimal Vehicle Trajectory Planning Considering Queue Effects at Actuated Traffic Signal Controlled Intersections. 2024 Jan 8 [cited 2024 May 13]; Available from: https://hdl.handle.net/10919/117840

13. Schwarzkopf AB, Leipnik RB. Control of highway vehicles for minimum fuel consumption over varying terrain. Transp Res. 1977 Aug;11(4):279–86.

14. Wei Y, Avcı C, Liu J, Belezamo B, Aydın N, Li P, et al. ynami c programming-based multi-vehicle longitudinal trajectory optimization with simplified car following models. Transp Res Part B Methodol. 2017 Dec;106:102–29.

15. Jiang H, Hu J, An S, Wang M, Park BB. Eco approaching at an isolated signalized intersection under partially connected and automated vehicles environment. Transp Res Part C Emerg Technol. 2017 Jun;79:290–307.

16. Wu X, Zhao X, Xin Q, Yu S, Sun K. Dynamic Speed Optimization in the Vicinity of Signalized Intersections during Green Phase. CICTP 2018 Intell Connect Mobil - Proc 18th COTA Int Conf Transp Prof. 2018;2318–29.

17. Omidvar A, Pourmehrab M, Emami P, Kiriazes R, Esposito JC, Letter C, et al. Deployment and Testing of Optimized Autonomous and Connected Vehicle Trajectories at a Closed-Course Signalized Intersection. Transp Res Rec J Transp Res Board. 2018 Dec;2672(19):45–54.

18. Wang Z, Wu G, Barth MJ. Cooperative Eco-Driving at Signalized Intersections in a Partially Connected and Automated Vehicle Environment. IEEE Trans Intell Transp Syst. 2020 May;21(5):2029–38.

19. Yang H, Almutairi F, Rakha HA. Eco-driving at signalized intersections: A multiple signal optimization approach. IEEE Trans Intell Transp Syst. 2022 Mar 9;22:2943–55.

20. Chen X, Qian L, Wang Q. Eco-driving at signalized intersections under uncertain traffic conditions. Proc Inst Mech Eng Part J Automob Eng. 2022 Oct;095440702211281.

21. Almannaa MH, Chen H, Rakha HA, Loulizi A, El-Shawarby I. Field implementation and testing of an automated eco-cooperative adaptive cruise control system in the vicinity of signalized intersections. Transp Res Part Transp Environ. 2019 Feb 1;67:244–62.

22. Chen H, Rakha HA, Loulizi A, El-Shawarby I, Almannaa MH. Development and Preliminary Field Testing of an In-Vehicle Eco-Speed Control System in the Vicinity of Signalized Intersections. IFAC-Pap. 2016 Jan 1;49(3):249–54.

23. Chen H, Rakha HA. Battery Electric Vehicle Eco-Cooperative Adaptive Cruise Control in the Vicinity of Signalized Intersections. Energ 2020 Vol 13 Page 2433. 2020 May;13(10):2433.

24. Mintsis E, Vlahogianni EI, Mitsakis E. Dynamic eco-driving near signalized intersections: Systematic review and future research directions. J Transp Eng Part Syst. 2020;146(4).

25. Zhai C, Chen C, Yang X, Liu G, Yan C, Luo F, et al. Ecological Driving for Connected and Automated Vehicles at Unsaturated Intersections Considering Queue Effects. IEEE Trans Veh Technol. 2022 Dec;71(12):12552–63.

26. Shafik AK, Rakha HA. Integrated Back of Queue Estimation and Vehicle Trajectory Optimization Considering Uncertainty in Traffic Signal Timings. IEEE Trans Intell Transp Syst. 2024;1–0.

27. Bodenheimer R, Brauer A, Eckhoff D, German R. Enabling GLOSA for adaptive traffic lights. In IEEE; 2014. p. 167–74.

28. Ibrahim S, Kalathil D, Sanchez RO, Varaiya P. Estimating phase duration for SPaT messages. IEEE Trans Intell Transp Syst. 2018;20(7):2668–76.

29. Van de Vyvere B, ’h aene K, ’ haene K, Colpaert P, Verborgh R. Predicting phase durations of traffic lights using live Open Traffic Lights data. In Technical University of Aachen; 2019. p. 1–7.

30. Moghimi B, Safikhani A, Kamga C, Hao W, Ma J. Short-term prediction of signal cycle on an arterial with actuated-uncoordinated control using sparse time series models. IEEE Trans Intell Transp Syst. 2018;20(8):2976–85.

31. Weisheit T, Hoyer R. Prediction of Switching Times of Traffic Actuated Signal Controls Using Support Vector Machines. In: Advanced Microsystems for Automotive Applications 2014. Springer; 2014. p. 121–9.

32. Genser A, Ambühl L, Yang K, Menendez M, Kouvelas A. Enhancement of SPaT-messages with machine learning based time-to-green predictions. In European Association for Research in Transportation; 2020.

33. Eteifa S, Rakha HA, Eldardiry H. Predicting coordinated actuated traffic signal change times using long short-term memory neural networks. Transp Res Rec J Transp Res Board. 2021;2675(9):127-138,.

34. Islam Z, Abdel-Aty M, Mahmoud N. Using CNN-LSTM to predict signal phasing and timing aided by High-Resolution detector data. Transp Res Part C Emerg Technol. 2022;141:103742.

35. Cao J, Hu D, Hadiuzzaman M, Wang X, Qiu TZ. Comparison of queue estimation accuracy by shockwave-based and input-output-based models. 2014 17th IEEE Int Conf Intell Transp Syst ITSC 2014. 2014 Nov;2687–92.

36. Comert G. Simple analytical models for estimating the queue lengths from probe vehicles at traffic signals. Transp Res Part B Methodol. 2013 Sep;55:59–74.

37. Tan C, Yao J, Tang K, Sun J. Cycle-Based Queue Length Estimation for Signalized Intersections Using Sparse Vehicle Trajectory Data. IEEE Trans Intell Transp Syst. 2021 Jan;22(1):91–106.

38. Liu H, Liang W, Rai L, Teng K, Wang S. A Real-Time Queue Length Estimation Method Based on Probe Vehicles in CV Environment. IEEE Access. 2019;7:20825–39.

39. Tavafoghi H, Porter J, Flores C, Poolla K, Varaiya P. Queue Length Estimation from Connected Vehicles with Low and Unknown Penetration Level. In: 2021 IEEE International Intelligent Transportation Systems Conference (ITSC) [Internet]. Indianapolis, IN, USA: IEEE; 2021 [cited 2024 Jul 9]. p. 1217–24. Available from: https://ieeexplore.ieee.org/document/9564477/

40. Badillo BE, Rakha H, Rioux TW, Abrams M. Queue length estimation using conventional vehicle detector and probe vehicle data. In: 2012 15th International IEEE Conference on Intelligent Transportation Systems [Internet]. 2012 [cited 2024 Jul 9]. p. 1674–81. Available from: https://ieeexplore.ieee.org/document/6338891/footnotes#footnotes

41. Zhang Z, Zhang S, Mo L, Guo M, Liu F, Qi X. Traffic Volume Estimate Based on Low Penetration Connected Vehicle Data at Signalized Intersections: A Bayesian Deduction Approach. IEEE Trans Intell Transp Syst. 2022 Aug;23(8):10596–609.

42. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In: Advances in Neural Information Processing Systems [Internet]. Curran Associates, Inc.; 2019 [cited 2024 Aug 26]. Available from: https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

43. Liaw R, Liang E, Nishihara R, Moritz P, Gonzalez JE, Stoica I. Tune: A research platform for distributed model selection and training. ArXiv Prepr ArXiv180705118. 2018;

44. Li L, Jamieson K, Rostamizadeh A, Gonina E, Ben-Tzur J, Hardt M, et al. A system for massively parallel hyperparameter tuning. Proc Mach Learn Syst. 2020;2:230–46.

45. Genser A, Ambühl L, Yang K, Menendez M, Kouvelas A. Time-to-Green predictions: A framework to enhance SPaT messages using machine learning. In IEEE; 2020. p. 1–6.

46. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. Adv Neural Inf Process Syst. 2017;30.

47. Nam G, Yoon J, Lee Y, Lee J. Diversity matters when learning from ensembles. Adv Neural Inf Process Syst. 2021;34:8367–77.

48. Wang C, Zhao X, Fu R, Li Z. Research on the comfort of vehicle passengers considering the vehicle motion state and passenger physiological characteristics: Improving the passenger comfort of autonomous vehicles. Int J Env Res Public Health. 2020 Sep 18;17(18).

49. Rakha H, Snare M, Dion F. Vehicle Dynamics Model for Estimating Maximum Light-Duty Vehicle Acceleration Levels. Transp Res Rec J Transp Res Board. 2004 Jan;1883(1):40–9.

50. Fadhloun K, Rakha H. A novel vehicle dynamics and human behavior car-following model: Model development and preliminary testing. Int J Transp Sci Technol. 2020 Mar;9(1):14–28.

51. Rakha HA, Ahn K, Moran K, Saerens B, Bulck EV. Virginia Tech Comprehensive Power-Based Fuel Consumption Model: Model development and testing. Transp Res Part Transp Environ. 2011;16(7):492-503,.