IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 17 April 2024, accepted 9 June 2024, date of publication 17 June 2024, date of current version 24 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3415495

RESEARCH ARTICLE

Eco-Driving Framework for Autonomous Vehicles at Signalized Intersection in Mixed-Traffic Environment

ABDULRAHMAN AHMAD[1], (Student Member, IEEE),
AMEENA S. AL-SUMAITI[2], (Senior Member, IEEE),
YOUNG-JI BYON[3], (Member, IEEE),
AND KHALIFA AL HOSANI[2], (Senior Member, IEEE)

[1]Electrical Engineering and Computer Science Department, Khalifa University, Abu Dhabi, United Arab Emirates
[2]Advanced Power and Energy Center, Electrical Engineering and Computer Science Department, Khalifa University, Abu Dhabi, United Arab Emirates
[3]Department of Civil Infrastructure and Environmental Engineering, Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates

Corresponding authors: Ameena S. Al-Sumaiti (ameena.alsumaiti@ku.ac.ae) and Abdulrahman Ahmad (abdulrahman_hamdy@ieee.org)

This work was supported by Khalifa University under Award kkjrc-2019-trans 2.

ABSTRACT It is interesting to realize that traffic intersections provide an essential mechanism to handle traffic flows in different directions while ironically traffic bottlenecks, gridlocks, and accidents tend to occur in the vicinity of the intersections. Meanwhile, with the recent developments in internet of things (IoT) technologies, there is a great potential for integrating them for improving operational efficiencies of road infrastructure and connected autonomous vehicles (CAVs). This paper aims to leverage the capabilities of both autonomous intersection manager (AIM) and CAVs for more energy-saving and safe-traffic management. A mixed-traffic environment where human-driven vehicles (HDVs) and CAVs sharing the same road is considered. A two-layer framework is adopted to handle signal and vehicle controls effectively. The first layer is a signal control layer where the AIM receives the traffic network states, trains with the data through machine learning (ML), and outputs a set of optimal green times for each intersection phase. The second layer is a decentralized-vehicle control layer where the CAVs receive the signal phase and timing (SpaT) information from the AIM to compute the optimal speed values. The proposed solution helps the CAVs to minimize idling at red signals or to speed up to safely arrive at and pass through a green signal. Our proposed framework is designed to optimize intersection efficiency and minimize vehicle average delay and fuel consumption. All experiments have been conducted in a microscopic traffic simulation environment, the PTV-VISSIM, simulating real-world dynamics of vehicles and drivers' behaviors based on decades of field-data.

INDEX TERMS Intelligent transportation, connected autonomous vehicles, signalized intersections, mixed-traffic, machine learning, adaptive speed control.

I. INTRODUCTION

Managing traffic intersections is one of the most sophisticated problems due to its impacts on the frequency and severity of accidents, energy consumption, traffic delay-time, and general traffic state. Since the early existence of traditional road intersections, scientific and engineering communities have always tried to provide solutions to improve efficiency, safety, and comfortability of commuters and drivers. The attention to reducing accidents by enhancing the road intersections has been rising gradually. As per the European Union's community database on road accidents, CARE, although there is a reduction in the number of accidents by 40% from 2007 to 2016, there were 5000 deaths from 25700 accidents in 2016 because of accidents at intersections [1]. In the US, a study in 2008, showed that from 5,811,00 crashes,

The associate editor coordinating the review of this manuscript and approving it for publication was Wei-Yen Hsu.

almost 40% of crashes were related to intersections [2]. In China, according to [3], from 2007 to 2016, about 30% of road accidents and collisions were intersection related. In addition to the invaluable cost of people's lives, there are monetary field dataciated with such accidents (e.g., the costs of the intersection-related crashes in USA are US$120 billion and US$371 billion in economic and societal aspects, respectively [4]). Furthermore, traffic congestion is globally a leading contributor to CO_2 emissions representing 23% as the highest share of gas emission from transportation [5].

Consequently, there exists a high demand to improve traditional traffic signal control by utilizing the recent progress in artificial intelligence (AI) and internet of things (IoT) technologies [6]. The vehicle-to-everything (V2X) enabling technology, which combines V2V and V2I, can provide communications among vehicles and infrastructures. This merit offers great potential for researchers to improve traffic operations. Although there are numerous research efforts in developing traffic intersections, all technological advances are still not fully utilized.

Traffic signal control methods have extensively been studied by the transportation engineering research communities starting from the fixed cycle-time signal control to the latest advanced methods such as adaptive signal control. Beginning with the early decades, Webster [7] introduced analytical models for traffic signal formulation to produce the optimal settings for reducing the average vehicular delay, which is still regarded as a classic academic basis in transportation engineering. Transitioning from the traditional analytical to modern adaptive signal control, Wu et al. [8] developed an adaptive signal control scheme, which outperformed the analytical formulation, to enhance the intersection operation and optimize the queue spillover condition relying on the vehicle speed rather than the queue length. With the recent advances in autonomous vehicles and the enabling technologies (e.g., IoT and V2X), Lee et al. [9] utilized such advancements and built an autonomous intersection manager to enable receiving information about the vehicles' states and the traffic network states and then generate optimal green phases of signals towards the intersection's maximum throughput and minimum average vehicle-delay.

The eco-driving approach at signalized intersections is aiming at improving the efficiency of the traffic intersection operation by utilizing the connectivity-enabling technologies. The main purpose of Eco-driving is to generate real-time speed advice for connected vehicles to enable them to adjust their velocities or driving behavior and perform particular actions to minimize fuel consumption and reduce their delays. In a multitude of research works, it is indicated that the eco-driving approaches are able to decrease fuel consumption and greenhouse gas (GHG) emissions by almost 10% on average [10].

From the perspective of methodology, optimal control methods have been introduced in the literature. Feng et al. [11] presented an adaptive signal control algorithm, which is a bi-level optimization utilizing V2X to reduce the delay time of vehicles and queue length. The experiments demonstrated that the total delay had decreased significantly under high market penetration rate (MPR) of CAVs compared to the traditional actuated control. A model predictive control has been investigated by many researchers due to its practicality. Kamal et al. [12] proposed a model predictive control approach for generating optimal signal phases in urban traffic providing online modifications for all free traffic parameters, i.e., cycle length, offsets, and split times. Other methods depend on intelligent computation to produce optimal signal control such as Artificial Neural Networks (ANNs), which are commonly used in machine learning (ML) and deep learning, Fuzzy Systems [13] and Reinforcement Learning (RL) [14].

ML techniques have introduced revolutionary solutions for many research problems and changed the way data-driven solutions can be utilized. Supervised learning can be used for improving the road traffic networks and signal controllers by considering the data recorded from the networks considering real-time traffic conditions as can be seen in [8], [15], and [16]. In the literature review article by Nguyen et al. [17], the deep learning method utilized in the intelligent transportation systems domains has been discussed.

Traffic simulation software tools play a significant role in designing effective traffic solutions. A variety of simulation tools have been used to simulate traffic at intersections. In the study of [18], the authors provided detailed modeling of the reservation-based technique for intersection control. They used the VISSIM external driver model EDM, which provides a possibility to replace or modify the internal driving behavior of PTV-VISSIM. The EDM dynamic link library is called by VISSIM to pass the current state data of each vehicle at every simulation run to the dynamic link library. Chen and Sun [19] utilized microscopic simulation tools to improve an algorithm for real-time signal control with dynamic programming (DP). The objective is to optimize the average vehicle delay, intersection throughput, and queue length. Moreover, in [20], VISSIM simulation is conducted with real-time data gathered for the City of Riverside, CA, USA. To recap, many distinguished works in the literature utilized PTV-VISSIM, which is also used in our paper, as the microscopic simulation tool that can mimic the real dynamics of traffic networks.

In modern traffic intersections, the traffic environment combines both HDVs and CAVs. In addition, connectivity-enabling technologies such as V2X and IoT are integrated. However, these recent technologies are still forming a hot research spot that requires a multitude of extra controllability and new designs to make them applicable in real-time scenarios. The objective of this paper is to utilize the V2X and CAVs technologies for improving traffic intersections which have local impacts on the CAVs and global impacts on the traffic network. As summarized in Table 1, it demonstrates the distinctive features of the proposed solution compared with

TABLE 1. Related works for the eco-driving solutions compared with the proposed approach.

Ref.	Mixed-traffic Env.	Speed control	Signal Control	Consider vehicle impacts	Consider network impacts	Multi-lanes
[21]	✓	×	✓	×	✓	✓
[22]	✓	✓	×	✓	×	✓
[23]	✓	✓	×	✓	×	✓
[24]	✓	✓	×	✓	×	×
[25]	✓	×	✓	×	✓	×
Proposed Solution	✓	✓	✓	✓	✓	✓

existing approaches across multiple key metrics, including mixed-traffic environment handling, speed control, and signal control, as well as the consideration of vehicle and network impacts, and multi-lane scenarios. The primary contributions of this study are summarized as follows:

1) **Integrated Framework for Traffic Optimization:** This paper proposes a comprehensive framework that employs connectivity technologies to enhance traffic networks both at the micro and macro scales. Distinct from previous works such as [8] and Kamal et al. [26], which only focus on global impacts, or Fayazi and Vahidi [27], which concentrates solely on the effects of CAVs, our approach is dual-faceted. On the micro level, it introduces a vehicle control layer optimizing CAVs' driving behaviors to reduce fuel consumption, and improve average delay and travel time. On the macro level, it adapts traffic signals according to vehicle flow rates, thereby enhancing intersection throughput and reducing average delays and queue lengths.
2) **Generic Platform for Research Advancement:** The framework is built upon a versatile platform that encourages further research exploration. It tackles the signal timing control dilemma through a unique data-driven and simulation-based approach, resulting in a Machine Learning (ML) model that is thoroughly developed, trained, analyzed, and evaluated to produce optimal traffic signals based on traffic flow and Market Penetration Rate (MPR) of CAVs. Furthermore, it introduces an Eco-driving adaptive speed algorithm for the effective longitudinal and lateral control of CAVs. This generalizable solution lays the groundwork for solving a broad spectrum of traffic dilemmas.
3) **Superior Performance and Promising Implications:** Through empirical comparisons, this paper showcases the significant improvements and superior performance of our proposed framework against other recent and fundamental solutions. These results highlight the framework's potential to significantly impact traffic management practices in the transportation industry.

The rest of this paper is organized as follows. Section II describes the methodology used for both the signal-timing control layer and vehicle-control layer of the proposed Eco-driving framework. Then, Section III details the experiments' setup and the simulation environment followed by Section IV which presents the final results. Finally, Section V concludes the paper.

II. METHODOLOGY

This Section introduces the methodologies used for developing the eco-driving framework for a smart AIM. Figure 1 illustrates the participating actors of the proposed framework. In the vehicle-control layer, the CAVs receive the signal phase and timing (SPaT) information and speed limits of the road from the AIM. Then, they process the received information and compute the corresponding optimal control actions such as the acceleration value and lane-change decision to clear the intersection minimizing the energy consumption, average vehicle delay, and travel-time. On the other hand, in the signal-timing control layer, the AIM utilizes the information it receives from the RSUs such as the vehicles' flow rates and the MPR of CAVs. Then, it processes these data and computes the optimal green signal times given to each section of the intersection to minimize the total-vehicle average delay and maximize the throughput of the intersection.

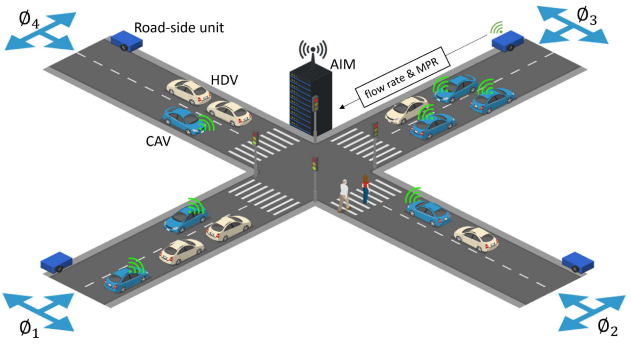


FIGURE 1. Typical intersection environment with V2X.

The proposed Eco-driving framework covers both the signal-timing control layer and vehicle-control layer as shown in Figure 2, which introduces the overall structure of the proposed solution.

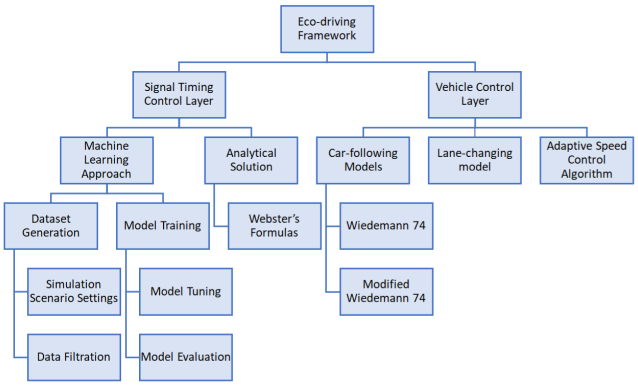


FIGURE 2. The proposed Eco-driving framework for both traffic signal layer, and vehicle control layer.

A. SIGNAL-TIMING CONTROL LAYER

Initially, a typical 4-leg intersection that permits all movements for every signal phase namely, a ''standard-four-phase'' is utilized in the literature in [28], [29], and [30] and is also considered in this paper as demonstrated in Figure 1. Every single-phase sequence is composed of the following states (All-red, Red/Amber, Green, Amber, and All-red). To find the optimal green signals for the intersection, a data-driven solution is used.

1) DATA-SET GENERATION

A data-set generation process is adopted in this paper due to the lack of data-sets that cover the needed traffic scenarios (such as different cases of MPR of CAVs and multiple flow rates per each intersection entry). The PTV-VISSIM micro-simulation is utilized to generate the data-set by using the COM-interface and Python scripting. Algorithm 1 is used for generating the data-set.

Algorithm 1 Data-Set Generation

```
Cycle_lengths ← [40 : 180]
Saturation_flow ← 2800
Total_Flows ← [0.3 ∗ Saturation_flow : Saturation_flow]
MPR ← [0 : 1]
green_times_list ← generate_green_times()
flow_ratios_list ← generate_rates()
for c in Cycle_lengths do
    for green_times in green_times_list do
        for flow_ratios in flow_ratios_list do
            Run_simulation(c, flow_ratios, green_times)
            save_results()
        end
    end
end
```

The following procedure is the complete sequence for the data-set generation process.

1) Initially, the simulation environment is built as in Section III.
2) The scenario variables and settings are defined as shown in Table 2. For every simulation run, a scenario setting is applied to the simulation environment.

TABLE 2. The scenario settings for the simulation runs.

Parameter	Range	Units
F	$r_1 \times S \times N_L$	v/h
f_i	$r_2 \times F$	v/h
MPR	$[0, 100]$	%
CT	$[50, 180]$	sec
g_i	$\{CT\frac{f_i}{F}, CT\frac{1}{4}, R1, R2\}$	sec

3) The total traffic F is the sum of all flow rates per each intersection's entry. F is selected based on a percentage $r_1 \in \{0.4, 0.5, .., 1\}$ of the traffic saturation rate S which is set to be 1900 veh/hr per lane.
4) The flow rate per entry f_i where $i \in \{1, 2, 3, 4\}$ is selected based on a percentage $r_2 \in [0.1, 0.7]$ of F in a way that:

$$\sum_{i=1}^{4} f_i = F \tag{1}$$

5) After every simulation run, the script records the evaluation results such as vehicle average delay, and fuel consumption.
6) A filtration step is applied by finding the optimum green times combination and MPR for every set of flow rates that give the minimum delay and fuel consumption.
7) The resultant green times of the filtered data-set will be used as ground-truth values during the training of the ML model. Table 3 presents an example of the generated data-set after filtration.

2) MACHINE LEARNING MODEL

An artificial neural network (ANN) is used to build a regression model that maps the input data (flow rates per each approach f_i and the MPR of CAVs) into output data of continuous real-number values (the green signal times). The activation function is a rectified linear-unit (ReLU). The merit is that ReLU maps negative values to zero, converges quickly and decreases the number of firing neurons. Due to the difference in ranges of the input features of the model, they are all normalized in a range of [0, 1]. Normalization helps the algorithm to converge faster. The hyperparameters such as the number of hidden layers, the number of neurons per layer, and learning rate values are determined after several experiments of tuning. Nine experiments are conducted to find the ANN model's optimal architecture. Each experiment selects a number of hidden layers combined with a number of neurons. The loss is plotted as shown in Figure 3. The architecture of 3 layers with 32 units is chosen as it has a fewer number of neurons and reduces the overfitting and the generalization error for testing data. The loss function for the

TABLE 3. An example of the generated data-set after filtration. The headings are ***CL***: cycle-length, ***F***: total-traffic flow, $\{f_1..f_4\}$: traffic per entry, $\{g_1..g_4\}$: green-time per signal phase, ***D***: average delay, and ***FC***: fuel consumption.

CL	F	f_1	f_2	f_3	f_4	MPR	g_1	g_2	g_3	g_4	D	FC
70	2240	224	224	672	1120	1	5.4	5.4	16.2	27	13.30	29.47
60	1120	112	224	224	560	0.4	4.4	8.8	8.8	22	13.43	15.53
50	1400	140	420	420	420	0.6	3.4	10.2	10.2	10.2	13.49	19.50
70	2240	448	448	448	896	0.8	10.8	10.8	10.8	21.6	13.53	30.08
90	2520	252	252	252	1764	1	7.4	7.4	7.4	51.8	13.56	33.62

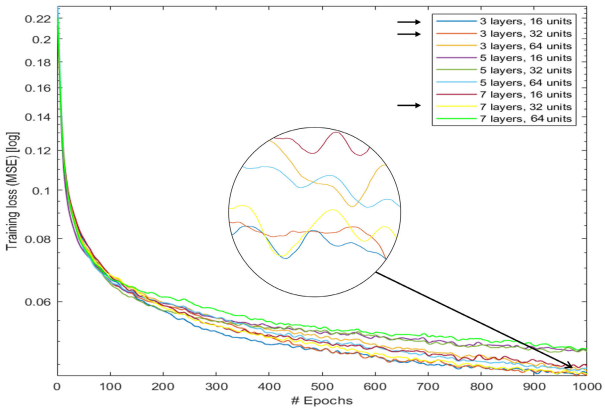


FIGURE 3. Results of 9 experiments to find the optimal ANN architecture for the learning model.

training model is selected to be the mean square error:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_i')^2 \tag{2}$$

where y_i is the ground truth value of the green time, and y_i' is the predicted values from the ANN model. The metric used in the model is the mean absolute error (MAE), and the used optimizer is the adaptive moment estimation (ADAM).

3) ANN MODEL TRAINING AND TUNING

The data-set is divided into three parts; 70% for training, 15% for validation, and 15% for testing. The validation data-set is used to track the performance of the model during the training. As seen in Figure 4, the performance of the initial model is found to be unacceptably low. To improve the performance, the following procedures are considered:

- **K-fold cross-validation:** This technique is utilized to represent a general view of the model performance (not limited to a specific range of data). First, the data-set is divided into K sections to be trained on K identical models. Second, each ANN is trained on $K-1$ sections of the data while evaluating the remaining sections. Finally, the validation score is the average of the K validation scores. After applying the K-fold cross-validation technique, the hyper-parameters are tuned such as the number of layers, and the number of neurons per layer.
- **Weights Regularization:** This technique can be enforced by adding a penalty function (cost function) to the loss function preventing the weights getting too large. There are two levels of regularization; first level $\mathcal{L}_1$, and second level $\mathcal{L}_2$.

$$\mathcal{L}_1 = \mathcal{L} + \lambda \sum_i |\omega_i| \tag{3}$$

$$\mathcal{L}_2 = \mathcal{L} + \lambda \sum_i |\omega_i^2| \tag{4}$$

$\mathcal{L}$ in Equations (3, 4) represents the old loss function, while $\mathcal{L}_1$ and $\mathcal{L}_2$ describe the new loss function. λ is a regularization parameter. $\mathcal{L}_1$ acts by forcing many weights to be zero, which ends up with sparse weights. $\mathcal{L}_2$ acts by forcing all weights to be small while some of them can reach zeros as well. It is clear that the model performance has been improved as the gap between the training loss and validation loss is reduced significantly as seen in Figure 4b.

The model is tested using the testing data, and the performance is evaluated. The scored **loss (MSE) of 0.0045 and MAE of 0.0414** are achieved. It should be known that generally a good score of loss starts with $\mathcal{L} \leq 0.5$. Justifications on the sufficient values of training loss is available in [31].

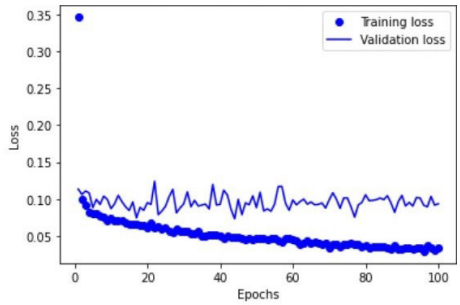
4) COMPUTATIONAL COMPLEXITY

The complexity of the model is categorized into two parts: 1) training, and 2) testing. When training an ANN model, the complexity of the algorithms such as feedforward and backpropagation are considered. It can be estimated by:

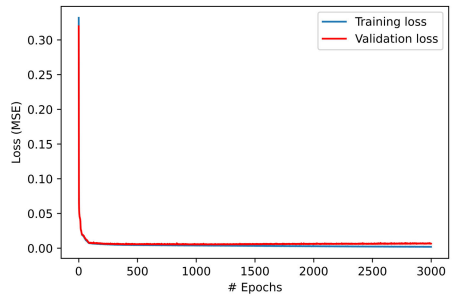
$$O\left(nt \times \sum_{i=1}^{K-1} ||m_i|| \times ||m_{i+1}||\right) \tag{5}$$

where n refers to the count of epochs, m is the samples used for training, k is the count of layers, and $||m_i||$ is the count of units in the i^{th} layer. In computing the Big-O notation, the constants in Equation(5) are cancelled resulting in:

$$O(nt) \tag{6}$$



(a) Before tuning



(b) After tuning

FIGURE 4. The performance (loss function) of training and validation before and after tuning.

It should be noted that the impacts of the complexity of the training model are not influencing the model performance in the deployment stage as the training is performed only once and then the model is deployed for usage and testing. The model's time complexity for deployment is found by:

$$O\left(\sum_{i=1}^{K-1} ||m_i|| \times ||m_{i+1}||\right) = \text{constant time} \tag{7}$$

5) UNCERTAINTY EVALUATION WITH SENSITIVITY ANALYSIS

For regression models, it is important to study the model's robustness against the input noise. Nowadays, the CAVs are still under development and the MPR is still low in smart cities. This MPR can be a result of an estimation model or collected data from sensors and measurements. As a result, in our case, it is very important that our model's sensitivity for the CAVs MPR remains to be very low. Consequently, a sensitivity analysis based on variance using the approach in [32] is investigated.

For any prediction model formulated as $Y = f(t_1, t_2, \ldots, t_k)$, a first order of variance-based sensitivity for variable t_i is determined by:

$$V_{t_i}\left(E_{\bar{t}_i}(Y|t_i)\right) \tag{8}$$

where t_i denotes the i^{th} parameter and $\bar{t}_i$ is the other parameters but not t_i. The average of output Y for all $\bar{t}_i$, when t_i is constant, is called the internal expectation operator. The external variance is computed for all values of t_i. In this case, the first-order index of sensitivity is determined by:

$$S_i = \frac{V_{t_i}\left(E_{\bar{t}_i}(Y|t_i)\right)}{V(Y)} \tag{9}$$

The first-order index indicates the contribution of an input parameter to the output regardless its effects on other input parameters. If an interaction with other input parameters is considered, then the overall participation of every input parameter, S_{Ti} is:

$$\begin{aligned} S_{Ti} &= \frac{E_{\bar{t}_i}(V_{t_i}(Y|\bar{t}_i))}{V(Y)} \\ &= 1 - \frac{V_{t_i}(E_{\bar{t}_i}(Y|t_i))}{V(Y)} \end{aligned} \tag{10}$$

The Saltelli sampler is utilized as implemented in SALib Python package [34]. Due to the normalization of the features, a range of (0, 1) is selected as bounds for the inputs to the sensitivity analysis. Figure 5 illustrates the contribution of the uncertainty in MPR of CAVs measurements to the predicted green times. The total order is greater than the first order as it describes the interaction among other variables (traffic flow in each intersection's entry). Although there is little difference among the four splits, the average of the first-order and total order of sensitivity index for MPR of CAVs are significantly low (0.006) indicating the model's sensitivity to measurements' errors in the estimations of MPR of CAVs in the traffic network.

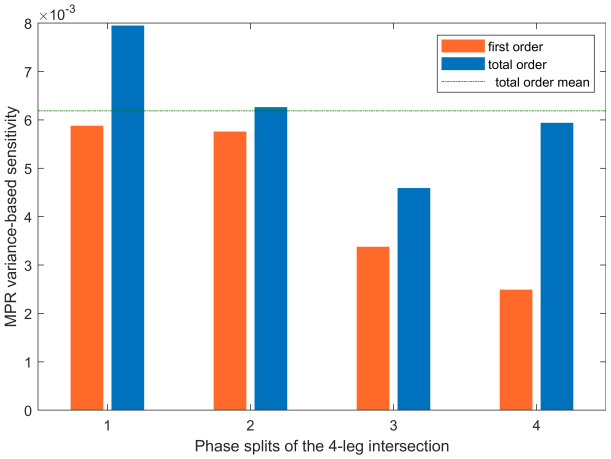


FIGURE 5. Sensitivity analysis of the ANN model for the parameter MPR.

6) OTHER MACHINE LEARNING TECHNIQUES

To verify that the proposed ANN approach is a suitable and proper ML algorithm that fits our research problem, seven experiments are conducted in this research using Linear Regression (LR), K-Nearest Neighbors (KNN), Decision Tree (DT), and Random Forest (RF), direct multi-output regression using SVR, and chained multi-output regression using SVR. The evaluation methods of the performance of the seven models were mean square error (MSE) and mean

absolute error (MAE). Table 4 shows a comparison among the mentioned ML algorithms. The results show that the performance of the ANN approach outperformed the other algorithms with an MAE of 0.0414 and MSE of 0.0045.

7) ANALYTICAL SOLUTION: BASELINE

In the analytical solution, to calculate the four phases for the signal controller, first, the cycle length is defined as the total time when all phases are completed for one cycle. The lost time is the time not being completely utilized during the cycle length. The total lost time is

$$t_L = t_{sl} + t_{cl} \tag{11}$$

where t_L is the overall lost time in seconds, t_{sl} is the lost time in start-up in seconds, and t_{cl} is the lost time for clearance in seconds. Webster's optimum cycle length is estimated as in [7]:

$$C_{opt} = \frac{1.5 * L + 5}{1.0 - \sum_{i=1}^{n} (v/s)_{ci}} \tag{12}$$

where C_{opt} is the optimal cycle length to reduce the average delay, L is the total lost time for cycle, and $(v/S)_c i$ represents a flow ratio for a critical lane group i, where S describes the saturation flow rate and is approximated depending on the geometry of the intersection as 1900 veh/hr per intersection's entry, and n represents the count of critical lane groups.

To find the critical lane group, each leg of the intersection has two lane groups, (left-turn) and (head and right-turn). For simplicity, the two lane-groups are assumed to have the same percentage of the routing decisions. The critical lane-group will be the one with the maximum flow rate of the two lane-groups per phase, which will be $0.5 * V_i/S$. After computing the optimal cycle length, the allocation of green times in the cycle is considered as follows:

$$g_i = \left(\frac{v}{s}\right)_{ci} \left(\frac{C}{X_i}\right) \tag{13}$$

where, g_i represents the effective green time for phase i, X_i represents the ratio for lane group i, and C represents the cycle length in seconds. By using Equations (11-13), the analytical optimal cycle length is computed and then, the green times are computed.

B. VEHICLE-CONTROL LAYER

In this layer, the CAVs dynamical behaviors such as longitudinal or lane-change behaviors are controlled to achieve the minimum vehicle average delay and fuel consumption. A low-level platform, shown in Figure 6, is deployed for running an adaptive speed control algorithm besides a car-following model that manages the vehicle to reach the intersection within the green signal period.

1) ADAPTIVE SPEED CONTROL ALGORITHM

The adaptive-speed control algorithm is proposed to find the optimal desired speed passed to the car-following model. The car-following model should provide the driving behavior

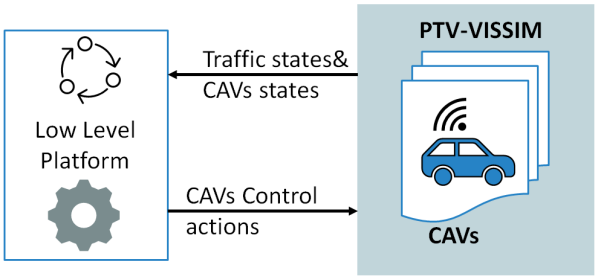


FIGURE 6. The low-level platform is connected with the simulation environment.

of each vehicle in terms of the given desired speed in a longitudinal direction with respect to the leading vehicle in the same lane. The objective is to make the distance from the CAV to the signal head zero when the signal state is green.

The distance to the signal head of the intersection in terms of the position in the current lane can be found by:

$$d = \begin{cases} L_i - P_j(t) & L_i \geq P_j(t) \\ 0 & L_i < P_j(t) \end{cases} \tag{14}$$

where L_i is the location of the signal head on lane i, and $P_j(t)$ is the current position of the vehicle CAV_j. If there is no signal on the current lane or the vehicle passed already the signal, the distance is 0. To calculate the time until the next green phase, t_0^{g+} should start taking into account cycle length C_L, current cycle-time $C(t)$, Green start-time t_0^g, and Green end-time t_f^g.

$$t_0^{g+} = \begin{cases} C_L - (C(t) - t_0^g) & t_0^g \leq C(t) \leq t_f^g \\ C_L - C(t) + t_0^g & C(t) \geq t_f^g \\ t_0^g - C(t) & Otherwise \end{cases} \tag{15}$$

To calculate the time until the next red phase starts, the cycle length C_L, current cycle-time $C(t)$, Green start-time t_0^g, and Green end-time t_f^g should be taken into account.

$$t_0^{r+} = \begin{cases} t_f^g - C(t) & C(t) < t_0^g \\ C_L - (C(t) - t_f^g) & C(t) \geq t_f^g \\ t_f^g - C(t) & Otherwise \end{cases} \tag{16}$$

The maximum velocity to arrive at the next green start is found by:

$$V_{max} = \frac{d}{t_0^{g+}} \tag{17}$$

If the vehicle drives faster, it would arrive at the signal head before the next green signal time. The minimum velocity to arrive at the next green end-time is found by:

$$V_{min} = \frac{d}{t_0^{r+}} \tag{18}$$

If the vehicle drives slower, it would not catch the next green end-time.

The optimal speed in this scenario is bounded by the maximum or minimum speed limits related to the urban road

TABLE 4. Performance comparison among tested machine learning algorithms.

#	LR	KNN	DT	RF	SVR_direct	SVR_chained	ANN
MSE	7.499	5.224	5.399	3.033	14.083	14.001	0.0414
MAE	1.625	1.108	0.887	0.865	2.458	2.454	0.0045

guidelines. The overall adaptive speed control algorithm used to find the optimal desired speed is described in Algorithm 2.

Algorithm 2 Adaptive Speed Control Algorithm

```
Function OptimalSpeedMin(minSpeed, desSpeed):
    if minSpeed < desSpeed then
        optimalSpeed = desSpeed
    end
    else
        optimalSpeed = -1
    end
    return optimalSpeed
Function OptimalSpeedMax(maxSpeed, desSpeed):
    if maxSpeed > desSpeed then
        optimalSpeed = desSpeed
    end
    else
        optimalSpeed = maxSpeed
    end
    return optimalSpeed
∀CAV_i ∈ {CAVs}, ∀δt ∈ t
for CAV_i in CAVs do
    V_min = getVminRedEnd( CAV_i)
    V_max = getVmaxGreenStart(CAV_i)
    DesSpeed = getDesSpeed( CAV_i )
    minSpeed = getRoadMinSpeed ()
    if V_min > V_max then
        optimalSpeed = OptimalSpeedMin( V_min, DesSpeed)
        if optimalSpeed == -1 then
            optimalSpeed = OptimalSpeedMax( V_max, DesSpeed)
        end
    end
    else
        optimalSpeed = max(min( V_max, DesSpeed), minSpeed)
    end
end
setVehiclesSpeed(optimalSpeed)
```

2) THE CAR-FOLLOWING MODELS

In [35], the authors compared the state-of-the-art car-following models in microscopic simulations and their applications which helped in developing the car-following model in this paper. For a mixed-traffic environment, it is necessary to define two behaviors for the two types of vehicles (HDVs and CAVs). For HDVs, Wiedemann-74 car-following model [36] is utilized as in [37]. For CAVs, a modified version of Wiedemann-74 model is proposed and used.

3) THE WIEDEMANN-74 CAR-FOLLOWING MODEL

The Wiedemann-74 model is also called the psychophysical model that utilizes perceptual thresholds in terms of relative distance and speed between leader-follower pairs of vehicles. The driver reacts only when the thresholds are reached. An example scenario for Wiedemann-74 model is illustrated in Figure 7. The blue line represents the decision path of an oncoming vehicle with a higher speed than the leader. As a result, it is getting closer until passing the point (P1) at which a deceleration perceptual threshold θv is activated as the driver realizes that the vehicle's speed is greater than the leader. However, a natural human behavior is the inability to follow the leader's speed precisely. This leads to increasing the spacing between the two vehicles until passing the point (P2) at which an acceleration perceptual threshold σv is activated. The driver again tries to follow the leader's speed by acceleration. This process is resumed as shown in Figure 7 at the unconscious reaction region.

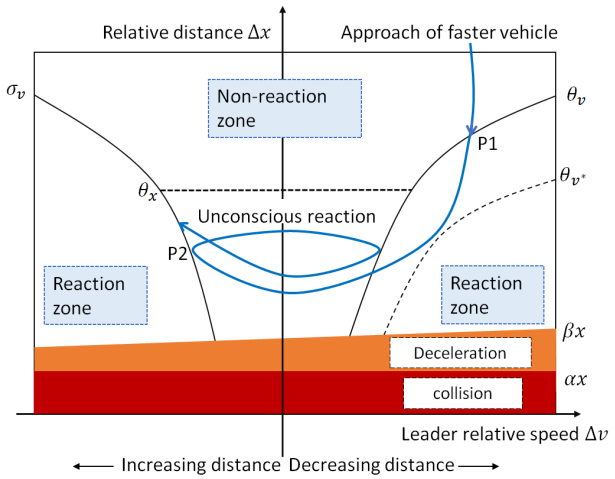


FIGURE 7. The parameters and thresholds utilized in this paper for the Wiedemann-74 car-following model illustrative graph [36], [38].

In Wiedemann-74, the threshold of the desired standstill distance αx is defined in [35] and [36] as follows:

$$\alpha x = L_{n-1} + \alpha x_{add} + R_1^n \alpha x_{mult} \tag{19}$$

where (n) refers to the ego vehicle and $(n-1)$ for the front vehicle. L_{n-1} is the length of the front vehicle, and αx_{add} and αx_{mult} describe additive and multiplicative components of the safety distance respectively. R_1^n is a random variable with a normal distribution to describe that a part of the desired standstill distance relies on the preferred safety requirement of the driver. The desired minimum following distance threshold η is calculated as follows:

$$\eta = \alpha x + \beta x \tag{20}$$

$$\beta x = (\beta x_{add} + \beta x_{mult} R_1^n)\sqrt{v} \tag{21}$$

where βx_{add} and βx_{mult} are additive and multiplicative components and they are calibration parameters. The follower

vehicle's speed v in the threshold's equation (21) at lower speed differences is found as:

$$v = \begin{cases} v_{n-1} & v_n > v_{n-1} \\ v_n & v_n \leq v_{n-1} \end{cases} \quad (22)$$

where v_{n-1} is the front vehicle's speed.

The threshold θx is used for the maximum following distance. The event occurs when the spacing between the follower-leader pair is increasing. Then, an action of acceleration is taken. The range of the threshold distance is in $[1.5 : 2.5]$ of βx, which is denoted by:

$$\theta x = \alpha x + \beta x * \gamma x \quad (23)$$

$$\gamma x = \gamma x_{add} + \gamma x_{mult}(R_D - R_2) \quad (24)$$

where γx_{add} and γx_{mult} are calibration parameters. R_D, and R_1 are random variables. The perception threshold θv is activated when there is a difference in speeds as mentioned in the earlier scenario. θv is defined as follows:

$$\theta v = \left(\frac{\Delta x - L_{n-1} - \alpha x}{\zeta x} \right)^2 \quad (25)$$

$$\zeta x = \zeta x_{const} \, (\zeta x_{add} + \zeta x_{mult} * (R1 + R2)) \quad (26)$$

where ζx_{const}, ζx_{add}, and ζx_{mult} are calibration parameters. The θv^* is a threshold that is similar to the θv but it recognizes the over-decaying in speed difference. On the other hand, if the speed difference increases, the threshold of σv is activated as the driver's speed is less than the leader's.

$$\sigma v = \theta v^* \, (-\sigma v_{add} - \sigma_{mult} * NR_3) \quad (27)$$

TABLE 5. The calibrated ranges and values for utilized parameters of Wiedemann-74 model [35], [36], [37].

Parameter	Value
αx_{add}	1.25 (m)
αx_{mult}	2.5 (m)
βx_{add}	2.0 (m)
βx_{mult}	1.0 (M)
γx_{add}	1.5 (m)
γx_{mult}	0.55 (m)
σv_{mult}	1.5
σv_{add}	1.5
ζx	40
A_{null}	0.1 (m/s^2)
R_D	N(0.5, 0.15)
R_1	N(0.5, 0.15)
R_2	N(0.5, 0.15)
R_4	N(0.5, 0.15)
A_{max}	$3.5 - (3.5/40)v$
A_{min}	$-20 - (1.5/60)v$

The typical values of all mentioned calibration parameters are found in Table 5.

4) THE MODIFIED WIEDEMANN-74 MODEL

In our proposal, the CAVs adopted the Wiedemann-74 model with the same calibrated parameters mentioned in Table 5 but thanks to the high speed of computations in CAVs, some parameters are modified as shown in Table 6.

TABLE 6. The configured parameters for HDVs and CAVs.

Driving behavior parameters	HDVs	CAVs
Average standstill distance	2 (m)	1 (m)
Multiplicative component of safety distance	3 (m)	0 (m)
Additive component of safety distance	2 (m)	1.5 (m)
Connectivity	No	Yes
Reaction time	(2 ± 0.3) (s)	0 (s)

5) LANE-CHANGE MODEL

The lane-change decision plays a crucial role in the driver's behavior. VISSIM utilizes Sparmann's lane-change model as considered in [39]. The model is categorized into two classes of decisions: 1) necessary lane-change decision and 2) free-lane change decision. In the first class, it is a must for the drivers to change their lanes based on their routing plan. In the second class, the decision is taken based on some criteria that improve the vehicle's performance such as increasing the speed and keeping the standstill distance at the same time [37]. In the micro-simulation software PTV-VISSIM, the vehicle that wants to take the lane-change decision is called an "own vehicle" and the vehicle in the other lane, which will be turned into a follower after the process of lane-change is complete, is called a "trailing vehicle". The maximum acceptable deceleration that is achieved by the follower vehicle is controlled as a parameter by the software tool. Tables 7 and 8 show the parameters and their values utilized in this paper.

TABLE 7. Simulation parameters for necessary lane-change decision.

Parameter	Own vehicle	Trailing vehicle
Maximum deceleration	$-3.5m/s^2$	$-3m/s^2$
$-1m/s^2$ distance	100 m	100 m
accepted deceleration	$-1m/s^2$	$-1m/s^2$

III. SIMULATION AND EXPERIMENTS

The methodology used for representing the environment is a multimodal microscopic traffic simulation. The PTV-VISSIM, a commercial simulation platform used in many real-world projects [37], is utilized. The software tool provides a traffic flow model, which moves each participant according to a sophisticated movement model.

TABLE 8. Other simulation parameters for lane-change.

Parameter	Value
Waiting time before diffusion	60 s
Min. clearance (front/rear)	0.5 m
Safety distance reduction factor	0.6
Maximum deceleration for cooperative braking	$-3m/s^2$

A. IMPLEMENTATION OF ISOLATED SIGNALIZED INTERSECTION

A signalized intersection with four legs for each direction is introduced. Each leg contains two links and two lanes per link as shown in Figure 8. Two types of vehicles into the traffic network with two different driver behaviors are defined as shown in Figure 9. For the HDVs, the traditional Wiedemann-74 psychophysical driver behavior with the calibrated parameters in Table 5 is used. On the other hand, for CAVs, the modified version of Wiedemann-74 is used with the parameters in Table 6.

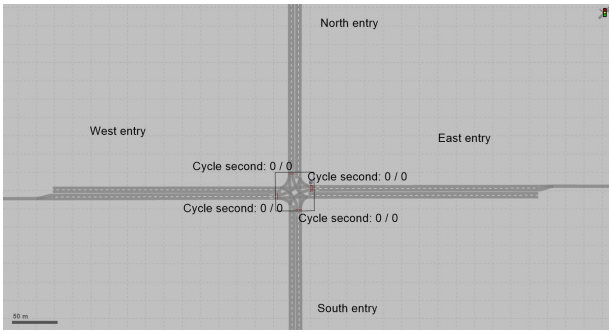


FIGURE 8. The traffic intersection in 2D-layout in PTV-VISSIM.

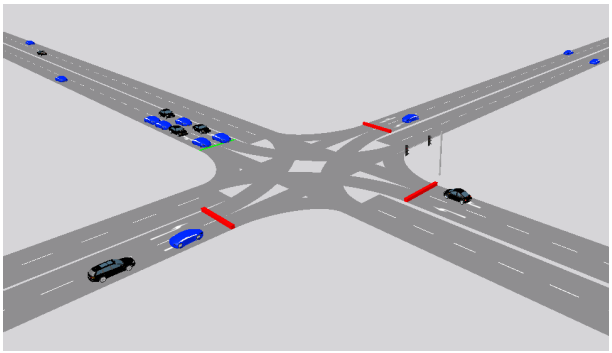


FIGURE 9. The mixed-traffic environment inside the intersection in 3D-layout where blue refers to CAVs, and black refers to HDVs.

B. EXPERIMENTS SETUP FOR TESTING THE ML MODEL

First, nine scenarios are set as shown in Table 9 to test the performance of the ANN model against the analytical solution by Webster's formula. There are two metrics to assess the performance of traffic networks, 1) Traffic level, which is the total traffic flows for the intersection and, 2) traffic imbalance, which represents how much the traffic flows of each approach in the intersection are similar to each other. Regarding the MPR of the CAVs, two comparisons are conducted: 1) when *MPR* of CAVs is 0% for a comparison purpose of evaluating the performance of the ANN model against the analytical solution that deals only with 0% of MPR of CAVs. The second comparison is for testing the performance of the model against the analytical solution by Webster's formulae while increasing the MPR of the CAVs from 0% to 100%.

TABLE 9. Scenario settings.

#	F	f_1	f_2	f_3	f_4	Traffic level	Traffic imbalance
1	1200	300	300	300	300	low	low
2		400	400	200	200	low	medium
3		200	200	200	600	low	high
4	2000	500	500	500	500	medium	low
5		600	600	400	400	medium	medium
6		200	200	400	1200	medium	high
7	2800	700	700	700	700	high	low
8		900	900	500	500	high	medium
9		800	800	800	600	high	high

The signal control program is essential for managing the traffic signals at each signal head within an intersection. This program adheres to a specific sequence of signal phases, which include: {All-red, Red/Amber, Green, Amber, and All-red} in a complete cycle. To establish a signal control program, the sequence initiates from the green phase times as determined by the proposed ANN model. These green phase times are crucial for defining the onset of green signals within the total cycle length.

Given are the start time of the green phase for each approach GS_i, the duration of the All-red phase x_1, the Red/Amber phase duration x_2, the duration of the green phase G_i, and the Amber phase duration x_3. The start times GS_i for the green phases are computed using the following equations:

$$GS_1 = x_1 + x_2 \tag{28}$$
$$GS_2 = GS_1 + G_1 + x_3 + x_1 \tag{29}$$
$$GS_3 = GS_2 + G_2 + x_3 + x_1 \tag{30}$$
$$GS_4 = GS_3 + G_3 + x_3 + x_1 \tag{31}$$

Upon establishing these start times, the model is set to facilitate a comparative study through simulation. Algorithm 3 represents of the approach used to calculate the start times for each signal phase, reflecting the logical flow described by the equations above.

This structured algorithm ensures the calculated start times for the green signals effectively manage the transition

Algorithm 3 Calculate Signal Start Times

1: **Input:** Green times $G[1..4]$, All-red time x_1, Red/Amber time x_2, Amber time x_3
2: **Output:** Green start times $GS[1..4]$
3: $GS[1] \leftarrow x_1 + x_2$ **for** $i = 2to4$ **do**
4: $GS[i] \leftarrow GS[i-1] + G[i-1] + x_3 + x_1$
end

between traffic signal phases, optimizing the flow of traffic through the intersection.

IV. RESULTS

In this section, the results are shown for both the ML approach for developing optimal signal control and the adaptive speed control algorithm for the CAVs.

A. RESULTS OF OPTIMAL SIGNAL TIMING: (GLOBAL IMPACTS)

The first comparison is represented in both Figure 10 and Figure 11, which shows the average delay and the total fuel consumption with an MPR of 0% of CAVs respectively. The bar charts show that the ANN model is working properly and similarly to the analytical solution with a slight improvement in all scenarios when all vehicles in the traffic network are HDVs. The means of the nine scenarios in terms of average delay are shown in Figure 10 and fuel consumption are shown in Figure 11 and it also shows that the ANN model performance is better than or equal to the optimal analytical solution when MPR of CAVs is 0%. However, when they are compared together in the case of increasing the MPR of CAVs, the ANN model outperformed the analytical solution significantly. The charts in Figure 12 and Figure 13 demonstrate the same previously mentioned 9 scenarios but in each scenario the MPR of CAVs is increased from 0% to 100%.

Interestingly, it is clear that while Webster's solution (analytical one) is constant in every scenario. The proposed framework (the ANN model and the adaptive speed algorithm) improves the traffic network metrics when exposed to the various values of MPR of CAVs. It should be noticed that the proposed solution is valid for a mixed-traffic environment, and it would reach its maximum benefit and exploitation when the MPR of CAVs is 100%.

Furthermore, to deduct the quantity of improvements for the proposed solution against the traditional analytical solution. a comparison is carried on in terms of the mean of the earlier proposed nine scenarios with the various values of MPR. As shown in Tables 10, 11, the most common assessments of traffic networks are considered. These tables represent the percentages of improvement for the proposed framework.

B. RESULTS OF THE CAVS LONGITUDINAL CONTROL

The adaptive speed control algorithm is used to enable the CAVs to change their speeds depending on the received information such as the signal phase and timing (SPaT) information. The algorithm also considers the speed limits of the road where the CAVs drive through. To illustrate the performance of the algorithm, an experiment is conducted by recording the travel times of the first 100 CAVs entering

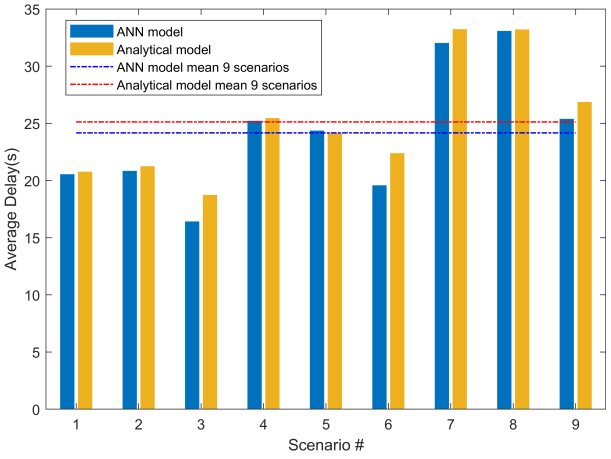


FIGURE 10. The average delay in seconds for 0% MPR of CAVs.

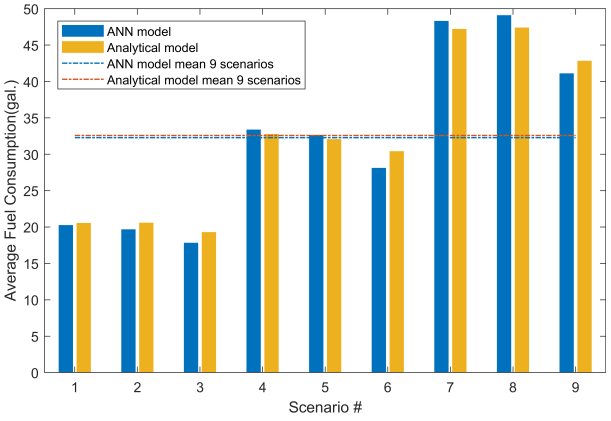


FIGURE 11. The fuel consumption in gallons results for 0% MPR of CAVs.

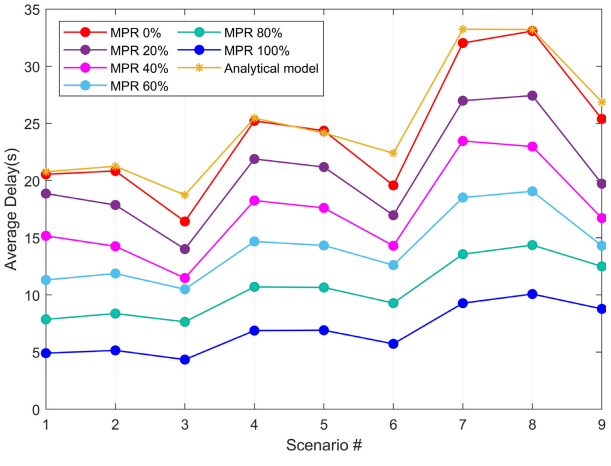


FIGURE 12. The average delay in seconds for [0-100%] MPR of CAVs.

TABLE 10. Results of the proposed eco-driving framework against the baseline. A: The proposed Eco-driving framework, W: baseline framework based on Webster's formulas, I: the percentage of improvements (reduction).

MPR	Avg. Delay			Fuel Consumption			Stops Per Vehicle		
	A	W	I	A	W	I	A	W	I
0%	24.1652	25.1231	4%	32.2803	32.58	1%	0.8049	0.8224	2%
20%	20.5487		18%	29.5513		9%	0.691		16%
40%	17.137		32%	26.6788		18%	0.5361		35%
60%	14.1306		44%	24.0827		26%	0.3948		52%
80%	10.5499		58%	21.3638		34%	0.2515		69%
100%	6.896		73%	18.8003		42%	0.1131		86%

TABLE 11. Results of the proposed eco-driving framework against the baseline. A: The proposed Eco-driving framework, W: baseline framework based on Webster's formulas, I: the percentage of improvements (reduction).

MPR	Stops Delay			Emissions CO			Avg. Q. Length		
	A	W	I	A	W	I	A	W	I
0%	17.8251	18.7124	5%	2256.4	2277.3	1%	12.187	12.809	5%
20%	12.1155		35%	2065.6		9%	9.5443		25%
40%	8.2555		56%	1864.8		18%	7.3021		43%
60%	5.6302		70%	1683.4		26%	5.4689		57%
80%	3.02		84%	1493.3		34%	3.5545		72%
100%	0.9474		95%	1314.1		42%	1.9422		85%

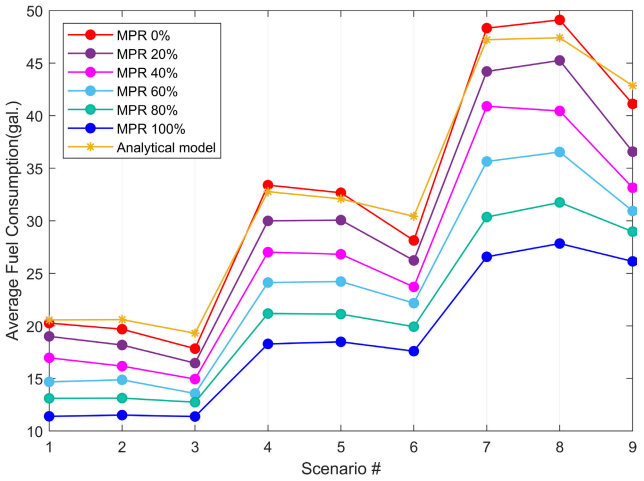


FIGURE 13. The fuel consumption in gallons for [0-100%] MPR of CAVs.

the intersection control area from all intersection entries (e.g., south, east, west, and north) as shown in Figure 14. The red circle in the graph represents the moment that a CAV enters the intersection track while the green circle shows the moment that the arrival of a CAV to the intersection signal head. The signal display for each entry is fixed above the plot to show the current state of the traffic signal for every CAV. It should be noted that the proposed framework could guarantee that all CAVs are able to clear the intersection at the starting time of a green signal or within the green signal period.

By analyzing this figure, it is concluded that two main cases occur with the CAVs while crossing the intersection. The two cases are 1) when it is feasible for the CAV to catch the nearest green signal, and 2) when it is infeasible to catch the nearest green signal. To illustrate these two cases, two different CAVs are selected from the set of the CAVs and their speed profiles are shown in Figures 15 and 16.

It is worth mentioning that the advised speed from the proposed algorithm is presented as the desired speed in graphs. The speed limit in the urban area is set to $60km/h$. However, the proposed algorithm caps the desired speed at $50km/h$ for safety and allows tolerance to any error that comes from measurements or the nonlinear dynamics of the vehicle's trajectory in real-life. A comparison of both cases is demonstrated in Table 12.

It is clear from the speed graph in Case 2 that the desired speed, obtained through the algorithm, is reduced from $50km/h$ to $20km/h$ to optimize the vehicle performance in terms of fuel consumption, vehicle delays, and vehicle stops. The optimal speed advised by the algorithm enhances the engine performance by not stopping a long time waiting for the green signal and being in the idling state without avail.

V. CONCLUSION

Signalized intersections play a crucial role in affecting traffic networks' efficiency and safety. The proposed framework provides a comprehensive study that considers a data-driven

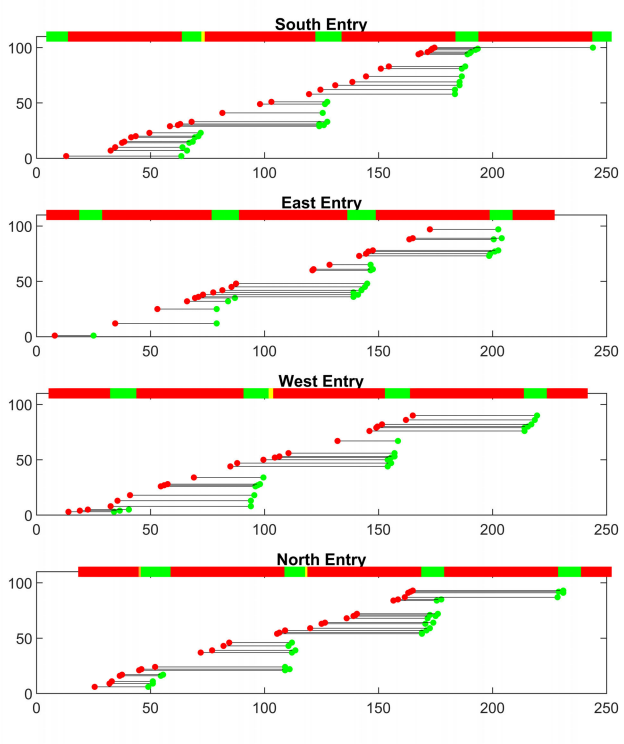


FIGURE 14. The travel time of 100 CAVs entered the intersection from four directions. The x-axis is the time in seconds and y-axis is the vehicle ID. The red circle is the entrance moment, the green circle is the arrival moment.

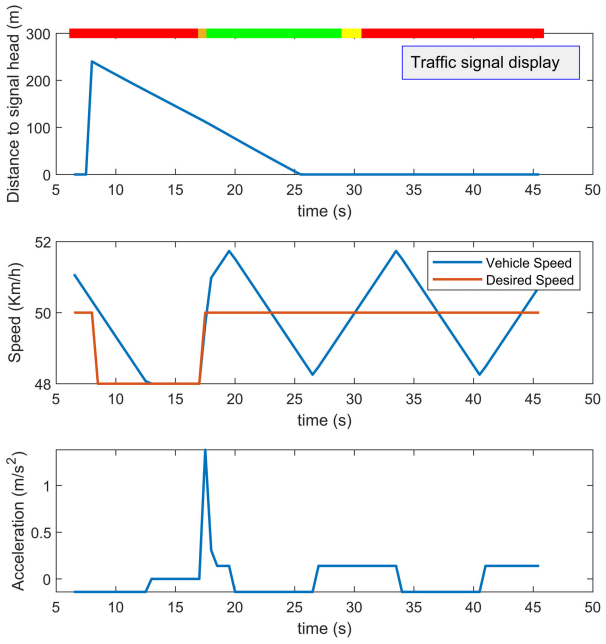


FIGURE 15. Speed profile for case 1: Feasible to catch the nearest green signal.

approach for signal timing control, and an adaptive speed algorithm for connected autonomous vehicles control. Data-driven approaches are suitable and more adaptive to address multiple variations of traffic scenarios such as varying values

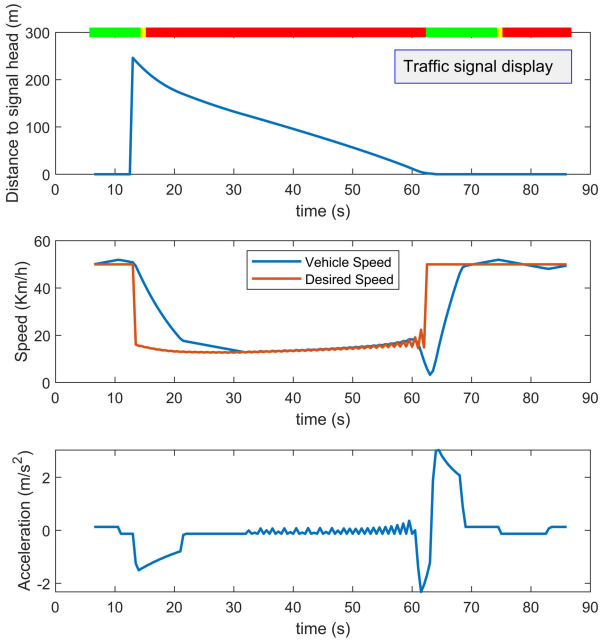


FIGURE 16. Speed profile for case 2: Infeasible to catch the nearest green signal.

TABLE 12. A comparison of the two cases of CAVs utilizing the vehicle-control layer.

Parameter	Case 1	Case 2
current signal received	red	green
feasibility to cross within the green period	yes	no
reaction behavior	keep at maximum	slow-down
$t_{entrance}$	7.5 (s)	13 (s)
$t_{arrival}$	25 (s)	61 (s)
travel time	18 (s)	48 (s)
average energy saving	42%	26%
average delay saving	73%	44%

of the MPR. The proposed technique can be customized in a real-time traffic network with its own geometrical conditions and traffic levels. The ANN model proved its efficiency against the traditional analytical solution provided by Webster's formulae. With the MPR of 0% of CAVs, the proposed solution could achieve similar or slightly better performance than the analytical solution (e.g., a reduction in the average delay is 4%). However, when the MPR is increased to its maximum 100%, there was a significant improvement in the traffic network behavior in terms of average delay (73%), average fuel consumption (42%), stops per vehicle (86%), stops delay (95%), emissions (42%), and average queue length (85%).

Future research directions include expanding the dataset to incorporate vulnerable road users, such as pedestrians and cyclists, thus broadening the model's applicability. Additionally, the adaptive speed control algorithm presents opportunities for further refinement through the exploration of more sophisticated control action tuning methods. These

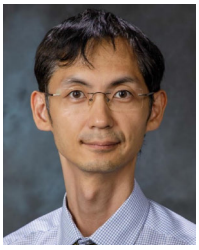
advancements promise to further optimize traffic network performance, underscoring the potential of our proposed framework to significantly contribute to the field of intelligent transportation systems.

REFERENCES

[1] C. European, "Traffic safety basic facts 2012: Junctions," Directorate Gen. Mobility Transp., Eur. Union, Brussels, Belgium, Tech. Rep., 2012.
[2] E.-H. Choi, *Crash Factors Intersection-Related Crashes: An On-Scene Perspective*. U.S. Department of Transportation, National Highway Traffic Safety Administration, 2010.
[3] X. Wang, H. Yu, C. Nie, Y. Zhou, H. Wang, and X. Shi, "Road traffic injuries in China from 2007 to 2016: The epidemiological characteristics, trends and influencing factors," *PeerJ*, vol. 7, p. e7423, Aug. 2019.
[4] L. Blincoe, T. R. Miller, E. Zaloshnja, and B. A. Lawrence, "The economic and societal impact of motor vehicle crashes, 2010 (revised)," Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. DOT HS 812 013, 2010.
[5] M. Grote, I. Williams, J. Preston, and S. Kemp, "Including congestion effects in urban road traffic CO_2 emissions modelling: Do local government authorities have the right options?" *Transp. Res. D, Transp. Environ.*, vol. 43, pp. 95–106, Mar. 2016.
[6] A. Ahmad, "Enhancing Internet of Things (IoT) strategies for autonomous electric vehicles (AEV) with existing transportation infrastructure and road users," Master's thesis, Dept. Elect. Eng. Comput. Sci., Khalifa Univ., United Arab Emirates, 2022.
[7] F. Webster, *Traffic Signal Settings* (Road Research Technical Paper). London, U.K.: H. M. Stationary Office, 1958.
[8] Y. Wu, H. Chen, and F. Zhu, "DCL-AIM: Decentralized coordination learning of autonomous intersection management for connected and automated vehicles," *Transp. Res. C, Emerg. Technol.*, vol. 103, pp. 246–260, Jun. 2019.
[9] J. Lee, B. Park, and I. Yun, "Cumulative travel-time responsive real-time intersection control algorithm in the connected vehicle environment," *J. Transp. Eng.*, vol. 139, no. 10, pp. 1020–1029, Oct. 2013.
[10] J. N. Barkenbus, "Eco-driving: An overlooked climate change initiative," *Energy Policy*, vol. 38, no. 2, pp. 762–769, Feb. 2010.
[11] Y. Feng, K. L. Head, S. Khoshmagham, and M. Zamanipour, "A real-time adaptive signal control in a connected vehicle environment," *Transp. Res. C, Emerg. Technol.*, vol. 55, pp. 460–473, Jun. 2015.
[12] M. A. S. Kamal, J. Imura, A. Ohata, T. Hayakawa, and K. Aihara, "Control of traffic signals in a model predictive control framework," *IFAC Proc. Volumes*, vol. 45, no. 24, pp. 221–226, Sep. 2012.
[13] J. Niittymäki and M. Pursula, "Signal control using fuzzy logic," *Fuzzy Sets Syst.*, vol. 116, no. 1, pp. 11–22, Nov. 2000.
[14] Y. Gong, M. Abdel-Aty, J. Yuan, and Q. Cai, "Multi-objective reinforcement learning approach for improving safety at intersections with adaptive traffic signal control," *Accident Anal. Prevention*, vol. 144, Sep. 2020, Art. no. 105655.
[15] I. Lamouik, A. Yahyaouy, and M. A. Sabri, "Smart multi-agent traffic coordinator for autonomous vehicles at intersections," in *Proc. Int. Conf. Adv. Technol. Signal Image Process. (ATSIP)*, May 2017, pp. 1–6.
[16] Y. Mo, M. Wang, T. Zhang, and Q. Zhang, "Autonomous cooperative vehicle coordination at road intersections," in *Proc. IEEE Int. Conf. Commun. Syst. (ICCS)*, Dec. 2018, pp. 192–197.
[17] H. Nguyen, L. Kieu, T. Wen, and C. Cai, "Deep learning methods in transportation domain: A review," *IET Intell. Transp. Syst.*, vol. 12, no. 9, pp. 998–1004, Nov. 2018.
[18] Z. Li, M. V. Chitturi, D. Zheng, A. R. Bill, and D. A. Noyce, "Modeling reservation-based autonomous intersection control in VISSIM," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2381, no. 1, pp. 81–90, Jan. 2013.
[19] S. Chen and D. J. Sun, "An improved adaptive signal control method for isolated signalized intersection based on dynamic programming," *IEEE Intell. Transp. Syst. Mag.*, vol. 8, no. 4, pp. 4–14, 2016.
[20] Z. Wang, G. Wu, and M. J. Barth, "Cooperative eco-driving at signalized intersections in a partially connected and automated vehicle environment," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 5, pp. 2029–2038, May 2020.
[21] Y. Du, W. ShangGuan, and L. Chai, "A coupled vehicle-signal control method at signalized intersections in mixed traffic environment," *IEEE Trans. Veh. Technol.*, vol. 70, no. 3, pp. 2089–2100, Mar. 2021.
[22] H. Dong, W. Zhuang, B. Chen, G. Yin, and Y. Wang, "Enhanced eco-approach control of connected electric vehicles at signalized intersection with queue discharge prediction," *IEEE Trans. Veh. Technol.*, vol. 70, no. 6, pp. 5457–5469, Jun. 2021.
[23] Y. Jeong and K. Yi, "Target vehicle motion prediction-based motion planning framework for autonomous driving in uncontrolled intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 1, pp. 168–177, Jan. 2021.
[24] S. Li, K. Shu, Y. Zhou, D. Cao, and B. Ran, "Cooperative critical turning point-based decision-making and planning for CAVH intersection management system," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 11062–11072, Aug. 2022.
[25] B.-K. Xiong and R. Jiang, "Speed advice for connected vehicles at an isolated signalized intersection in a mixed traffic flow considering stochasticity of human driven vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 11261–11272, Aug. 2022.
[26] M. A. S. Kamal, T. Hayakawa, and J.-i. Imura, "Development and evaluation of an adaptive traffic signal control scheme under a mixed-automated traffic scenario," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 2, pp. 590–602, Feb. 2020.
[27] S. A. Fayazi and A. Vahidi, "Mixed-integer linear programming for optimal scheduling of autonomous vehicle intersection crossing," *IEEE Trans. Intell. Vehicles*, vol. 3, no. 3, pp. 287–299, Sep. 2018.
[28] M. Bashiri, H. Jafarzadeh, and C. H. Fleming, "PAIM: Platoon-based autonomous intersection management," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 374–380.
[29] M. Pourmehrab, L. Elefteriadou, S. Ranka, and M. Martin-Gasulla, "Optimizing signalized intersections performance under conventional and automated vehicles traffic," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 7, pp. 2864–2873, Jul. 2020.
[30] M. Bashiri and C. H. Fleming, "A platoon-based intersection management system for autonomous vehicles," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 667–672.
[31] G. Shmueli, "To explain or to predict?" *Stat. Sci.*, vol. 25, pp. 289–310, Jan. 2011.
[32] A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola, "Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index," *Comput. Phys. Commun.*, vol. 181, no. 2, pp. 259–270, Feb. 2010.
[33] T. Iwanaga, W. Usher, and J. Herman, "Toward SALib 2.0: Advancing the accessibility and interpretability of global sensitivity analyses," *Socio-Environ. Syst. Model.*, vol. 4, p. 18155, May 2022.
[34] J. Herman and W. Usher, "SALib: An open-source Python library for sensitivity analysis," *J. Open Source Softw.*, vol. 2, no. 9, p. 97, Jan. 2017.
[35] H. U. Ahmed, Y. Huang, and P. Lu, "A review of car-following models and modeling tools for human and autonomous-ready driving behaviors in micro-simulation," *Smart Cities*, vol. 4, no. 1, pp. 314–335, Mar. 2021.
[36] Wiedemann and Reiter, "The simulation system mission, background and actual state," CEC, Brussels, Belgium, Tech. Rep. V1052, 1992.
[37] PTV Group, Karlsruhe, Germany. (2022). *PTV VISSIM 2022 User Manual*.
[38] M. Saifuzzaman and Z. Zheng, "Incorporating human-factors in car-following models: A review of recent developments and research needs," *Transp. Res. C, Emerg. Technol.*, vol. 48, pp. 379–403, Nov. 2014.
[39] E. Fransson, "Driving behavior modeling and evaluation of merging control strategies—A microscopic simulation study on sirat expressway," Master's thesis, Dept. Commun. Transp. Syst., Linköping Univ., Linköping, Sweden, 2018.

ABDULRAHMAN AHMAD (Student Member, IEEE) received the B.Sc. degree (Hons.) in computer and systems engineering from Minia University, Minia, Egypt, in 2016, and the M.Sc. degree in electrical and computer engineering with the Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates, in 2023. He is currently pursuing the Ph.D. degree in electrical engineering and computer science with Khalifa University. Since 2021, he has been on academic leave from his position as a Teaching and Research Assistant with the College of Engineering, Minia University, to focus on his doctoral research. His research interests include intelligent transportation systems, with a particular emphasis on the motion planning and control of autonomous vehicles. He has a strong interest in leveraging artificial intelligence, robotics control, optimization techniques, and machine learning to advance the field of autonomous systems. His work aims to contribute to the development of smarter, more efficient, and safer transportation solutions.

AMEENA S. AL-SUMAITI (Senior Member, IEEE) received the B.Sc. degree in electrical engineering from United Arab Emirates University, Abu Dhabi, United Arab Emirates, in 2008, and the M.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2010 and 2015, respectively. She was a Visiting Assistant Professor with MIT, Cambridge, MA, USA, in 2017. She is currently an Associate Professor with the Advanced Power and Energy Center and the Department of Electrical Engineering and Computer Science, Khalifa University, Abu Dhabi. She is also leading the Smart-OR Laboratory. Her research interests include stochastic processes, intelligent systems, transportation engineering, and autonomous driving. She is among the 2021 list of top 2% scientists of the world as per the results of 2022 and 2023. She is a receiver of many local and international awards and fellowships among them is the Arab-American Frontiers Fellowship. She is a National Expert Program 3.0 Alumnae.



YOUNG-JI BYON (Member, IEEE) was born in Seoul, South Korea, in 1979. He received the B.A.Sc. degree in mechanical engineering and the M.A.Sc. and Ph.D. degrees in civil engineering from the University of Toronto, Canada, in 2003, 2005, and 2011, respectively. From 2009 to 2010, he was a Visiting Researcher with the University of Chile, Chile. From 2010 to 2011, he was a Postdoctoral Research Fellow with the University of Calgary, Canada. From 2012 to 2023, he was an Assistant Professor, an Associate Professor, and the Associate Chair with the Department of Civil Infrastructure and Environmental Engineering, Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates. He is currently a Professor of engineering and the Founding Director of the Engineering Program, Northwestern College, IA, USA.

KHALIFA AL HOSANI (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering from the University of Notre Dame, Notre Dame, IN, USA, in 2005 and 2007, respectively, and the Ph.D. degree in electrical and computer engineering from The Ohio State University, Columbus, OH, USA, in 2011. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, Khalifa University, Abu Dhabi, United Arab Emirates. He is also the Co-Founder with the Power Electronics and Advanced Sustainable Energy Center Laboratory, ADNOC Research and Innovation Center (now the Power Electronics and Sustainable Energy (PEASE) Research Laboratory), Khalifa University. His research interests include nonlinear control, sliding mode control, renewable energy systems modeling and control, and transportation engineering.