Traffic Prediction and Optimization at Signalized Intersections with Connected Vehicles: A Hybrid Approach

School of Graduate Studies

Hamilton, Ontario

Canada

December 2023

TRAFFIC PREDICTION AND OPTIMIZATION AT SIGNALIZED INTERSECTIONS WITH CONNECTED VEHICLES: A HYBRID APPROACH

McMaster University

TRAFFIC PREDICTION AND OPTIMIZATION AT SIGNALIZED INTERSECTIONS WITH CONNECTED VEHICLES: A HYBRID APPROACH

By Xiaoyan Sun, M.CE.

A Thesis Submitted to the School of Graduate Studies in Partial Fulfilment of the Requirements for

the Degree Master of Engineering

McMaster University MASTER OF ENGINEERING (2023) Hamilton, Ontario (History)

TITLE: Traffic Prediction and Optimization at Signalized Intersections with Connected Vehicles: A Hybrid Approach

AUTHOR: Xiaoyan Sun, M.CE. (McMaster University)

SUPERVISOR: Assistant Professor Hao Yang

NUMBER OFPAGES: x, 80

Abstract

With the evolution of Connected Vehicle (CV) technology, the exploration of CV-enabled traffic control systems emerges as a novel strategy to address urban traffic congestion and enhance the operational efficiency of signalized intersections. In a mixed traffic environment comprising both CVs and non-CVs, leveraging multi-resolution CV data for more precise prediction of future traffic conditions has become a critical factor in improving the effectiveness of traffic control systems. This thesis proposes a novel hybrid approach that seamlessly integrates traditional traffic flow models with deep learning techniques to predict and optimize traffic at signalized intersections with CVs.

The proposed method first utilizes a Long Short-Term Memory (LSTM) Neural Network model based on CV data to predict the in-flow rates at intersections. Subsequently, a shockwave theory is applied to the predicted in-flow rates for accurate queue profile prediction. Finally, a signal optimization algorithm is developed to search for optimal phase sequences and durations within a forward time window to minimize vehicle delay. The simulation platforms are built for both a virtual and a real-world intersection, respectively, to evaluate the effectiveness of this hybrid approach in predicting queue profiles and optimizing signal timings. The results demonstrate that the proposed hybrid model performs well in predicting total delay at intersections under various market penetration rates (MPRs) of CVs and traffic demand levels. Furthermore, in a comparative analysis with the actuated control and fixed-time control, the proposed control algorithm is proven to outperform them significantly under medium and high demand levels. Under low demand levels, the proposed control algorithm has effectiveness similar to the actuated control but remains superior to the fixed-time control. Additionally, the results indicate that the effectiveness of this algorithm improves with higher MPRs and is still better than that of the actuated control, even under lower MPRs.

Key words: Connected Vehicle; Queue Profile Prediction; Signal

Optimization; Long Short-Term Memory Neural Network

Acknowledgement

I would like to express my sincere gratitude to my supervisor, Professor Yang, for his invaluable guidance, unwavering support, and professional expertise throughout my academic journey. His consistent support and immense patience have been crucial to my development, and his passion for academia has been a deep source of encouragement. I am profoundly thankful for the opportunity to study under his guidance, and I appreciate the knowledge and skills I've gained during this learning experience.

I would also like to express my heartfelt thanks to my family and friends for their unwavering love, encouragement, and support throughout my educational journey. Their consistent support and motivation have served as the driving force behind my accomplishments, and reaching this significant milestone would have been impossible without their presence.

Moreover, I want to express my thanks to my schoolmates for their constructive feedback and suggestions, which greatly enriched the quality of this research.

Finally, I extend my thanks to all those who have contributed to my successful completion of studies at McMaster University.

Table of Contents

List of Figures

List of Tables

Background

1.1 Introduction

In the past few decades, there has been a notable escalation in traffic congestion in major urban areas, primarily attributed to the movement of both people and goods. The Greater Toronto and Hamilton Area (GTHA) in Canada, in particular, grapples with this issue, incurring an annual cost of approximately six billion dollars [1]. Meanwhile, within the United States, urban residents were burdened with a remarkable increase of 6.9 billion additional hours spent on travel and an additional consumption of 3.1 billion gallons of fuel in 2011, resulting in a staggering congestion cost of $16 trillion [2]. Several critical factors contribute to these congestion problems, notably the inefficiencies in traffic signal timing control and the excessive traffic demand for road space. Innovative traffic control systems based on the detection of real-time traffic states are being developed by researchers to address these challenges and enhance traffic efficiency in urban areas. These control systems could detect vehicles, gather traffic information, and predict future traffic state (traffic volume, queue length, etc.). On this basis, traffic signal optimization algorithms are implemented to generate corresponding signal control parameters in real-time, such as green signal ratio, phase duration, phase sequence, and cycle length [3-5]. Therefore, the effectiveness of traffic control systems depends on the accuracy of traffic state prediction and the capabilities of signal timing optimization algorithms.

The existing traffic signal control systems mainly rely on loop detectors or video cameras to perceive traffic conditions and estimate the queue length and delay. However, information obtained from fixed detectors is just the instantaneous position of a vehicle instead of a direct measurement of vehicle mode or state (location, speed, acceleration) [6]. Consequently, traffic state prediction based on this instantaneous data lacks the ability to predict short-term variations in continuous traffic flow. Additionally, this instantaneous information also limits the ability of the signal optimization algorithm to reveal the mechanisms of interaction between control objectives and dynamic changes in traffic flow, affecting the control effectiveness of systems [7]. On the other hand, the existing traffic signal control systems usually consist of multiple sensors; thus, the installation and maintenance costs are relatively high, and the malfunction of one of the sensors will significantly degrade the performance of the entire system [8]. In summary, it is evident that the development of traffic signal control systems based on fixed-location detectors has encountered insurmountable bottlenecks.

The development of the Connected Vehicles (CV) technology provides an effective solution to overcome the above-mentioned bottlenecks. In contrast to fixed detectors, vehicles in a CV environment can communicate with each other (V2V) and roadside infrastructures (V2I) [9]. The CV environment can provide massive, dynamic, and continuous CV data regarding the vehicle's position, speed, and space headway in real time with better accuracy [10], which reflects a dynamic perception of the vehicle's state in both time and space. However, processing this multi-resolution spatiotemporal data to make a more accurate traffic state prediction is a new challenge. Due to the fact that signal optimization operations heavily rely on the accuracy of queue length or delay predictions [11], we urgently need a robust method capable of handling fluctuations in traffic demand and various Market Penetration Rates (MPRs) of CVs. In recent years, many machine learning models, especially deep learning models, have been widely developed and deployed for traffic estimation and prediction due to the availability of multi-resolution traffic data[12-15]. The Long Short-Term Memory (LSTM) neural network model is one of the most popular deep learning models in this field [16,17], highly favoured for its excellent ability to remember historical traffic state information for much longer periods and capture abrupt discontinuities in traffic flow [18].

On the basis of existing research, this thesis integrates traditional traffic models with traffic in-flow rates predicted by the deep learning model to establish a traffic control system enhanced by CV data, which enables fine-grained queue profile prediction and traffic signal optimization in a mixed environment of CVs and non-CVs. The proposed system will be capable of performing the following: (1) Predict the future queue profile and the delay associated with the queue based on the predicted in-flow rate. (2) Utilize the queue profile information at the intersection to optimize the sequence and duration of signal phases with the objective of minimizing vehicle delay in real time. It is noteworthy that the model and algorithm proposed in this thesis can effectively leverage all available CV data without the necessity of additional fixed detectors as supplementary data sources.

Overall, compared with existing traffic control systems, the system proposed in this thesis could eliminate the reliance on fixed detectors. Through the comprehensive utilization of CV data, achieved by integrating traditional traffic models with deep learning approaches, the research develops a refined queue profile prediction model and a more efficient, adaptive signal optimization algorithm. These advancements aim to enhance the traffic flow at a single signalized intersection, representing a significant improvement compared to existing systems.

1.2 Objectives

The detailed objectives are shown below:

(a) Develop a hybrid approach that integrates the conventional traffic model with predicted in-flow rates from a deep learning model, leveraging data from CVs. This integration seeks to enhance the accuracy of future queue profile predictions.

(b) Present a real-time signal optimization algorithm grounded in the predicted future queue profile information. The objective is to minimize vehicle delays at signalized intersections with a mix of CVs and non-CVs

Literature Review

2.1 Research on queue length prediction

2.1.1 Methods using only CV (probe vehicle) data

Some early studies in this direction have used wireless location technology to build probe vehicle-based traffic monitoring systems. These studies used wireless communication technologies and Global Position System (GPS) to record the trajectories of a portion of vehicles (probe vehicle) in the traffic network, which provide their dynamic information of speed, location, and travel time to estimate the queue length.

The traditional probe vehicle-based methods are constrained by the trajectory resolution level and penetration rate of sampling vehicles. Typically, these methods involve dividing the roads into grids and estimate the average traffic state (e.g., average speed and average flow rate) within each grid based on data collected by probe vehicles [19-21]. Due to the limited resolution and data volume, most probe vehicle-based methods cannot be applied for real-time queue length estimation or prediction. In contrast, CV-based methods can collect data at high resolutions, allowing for the characterization of the trajectory of each sampled vehicle or derivation from adjacent unsampled vehicles [22-23]. Therefore, the real-time data collected by CVs is able to be employed for real-time queue length estimation or prediction.

Several studies used probe vehicle data to estimate queue length from probability theory and statistics. Comert & Cetin [24] used the location information of probe vehicles in queues at isolated under-saturated intersections; the study developed an analytical formula based on conditional probability distribution to estimate the expected queue length and its variance. Given the penetration of probe vehicles and the distribution of queue length, their method could estimate the queue length of each cycle only by the position of the last probe vehicle in the queue. The research also discussed the influence of probe vehicle market penetration on the estimation accuracy in this literature. However, the practical application of this method has some difficulties. A later study [25] considered the time when probe vehicles entered the queue. Their work could be extended to spatial and temporal dimensions and provided formulations to quantify the estimation error of queue length using probe vehicle data.

Similarly, Viti and Van Zuylen [26] developed a probabilistic method for modelling the dynamics and uncertainties of vehicle queues at intersections controlled by the fixed-time and actuated signal controls. Comert [27] further developed real-time queue length estimation models and analyzed the estimation error under different MPRs of CVs and volume-to-capacity ratio levels. In their assumption, the arrival pattern of vehicle follows a Poisson distribution with known arrival rates. Moreover, the MPRs of CVs and phase duration are also assumed to be known values. On the basis of these main assumptions, the study derived the probability distribution of the probe vehicles' queuing position and queuing joining times. For the cases with overflow queues, approximate models were established, and analytical expressions for the mean and variance of queue length estimators are provided for the cases without overflow queues. Based on VISSIM microscopic simulation results, this model could reduce the estimation error to less than 4% under all volume-to-capacity ratios and MPRs of CVs. Comert [28] assumed a point-queue model without explicitly consideration of space. This method assumed that the sequence of probe vehicles in the queue was known. Besides, the study summarized a series of methods for queue length estimation under the assumption of Poisson distribution and systematically evaluated different estimators.

The disadvantages of Comert's probabilistic analytical method lie in its reliance on the locations of the queue locations and the time when vehicles joined the queue, which may not be directly obtained from CV data. Hao et al. [29] proposed a real-time queue length distribution estimation method for signalized intersections based on Bayesian networks. This method first defined the virtual trip lines at the upstream and the downstream of the intersection, from which the travel times of sample vehicles could be collected. Subsequently, this study classified traffic conditions into seven scenarios using sample travel times and established a Bayesian network model for each case to estimate vehicle index by modifying a predefined three-layer Bayesian network [30]. Following the computation of conditional probabilities for latent variables based on the estimated sample travel times and vehicle indices, the study ultimately derived the distribution of queue lengths for each scenario. By conducting field experiments and validating with simulation data, it was demonstrated that this approach displays enhanced accuracy and robustness when contrasted with previously developed linear fitting methods and queue localization approaches.

Some studies used probe vehicle or CV data to estimate the queue length based on shockwave theory. Cheng et al. [31] developed a shockwave-based cycle-by-cycle queue length estimation model utilizing high-resolution and extensive vehicle trajectory data. Initially, they constructed a trajectory model based on LWR theory. Subsequently, they introduced a threshold-based algorithm to identify critical points that signify changes in vehicle dynamics. The real-time queue length was then determined by fitting the curves of the critical points associated with queue-forming and discharging waves. The effectiveness of the method was validated through simulations and NGSIM data, demonstrating an average absolute percentage error of approximately 20% in estimating maximum queue length across various scenarios.

Similarly, Cetin [32] also proposed a method to identify the critical points in the shockwave profile. Utilizing data from the first and last observed probe vehicles in each cycle, this approach formulated a method to estimate queue length at signalized intersections. Although this approach did not necessitate probe observations for every cycle, obtaining results proved challenging if there are less than two probe vehicles could be observed in oversaturated conditions.

Ban et al. [33] introduced an alternative method based on shockwaves for real-time estimation of queue length at signalized intersections. The proposed model relied on travel time and delay measured by CVs passing through the signalized intersection. They proposed that critical pattern changes in travel time correlated with alterations in signal timing and queue length. On the basis of the assumptions that vehicle arrivals were uniform and the signal timing was known, this study identified critical points within a cycle where the queue was maximized, minimized, or cleared, and constructed the queue length curve using the delay patterns. Field experiments and simulations demonstrated that favorable outcomes could be achieved under high MPR scenarios (e.g., above 30%). However, the necessity for at least two observed probe vehicles per cycle might impact the feasibility of this approach.

Ramezani and Geroliminis [34] proposed an shockwave profiles estimation method at signalized intersections in urban networks. They assumed given vehicle arrival distribution, location, and velocity and used a threshold-based classifier to classify the data into moving and stopped classes. Then, they considered the estimation of the queue front for each cycle a constrained least squares problem and employed piecewise linear function to identify the tail of the queue. The feasibility and robustness of proposed approach were tested through numerical analysis of both NGSIM data and simulation data.

Hao et al. [35] proposed an approach based on kinematic equations. This approach could estimate the location of vehicles in the queue based on the time of vehicles passing through the intersection. Assuming that one vehicle's acceleration and deceleration rates were constant, they established specific kinematic equations for the through and left-turn vehicles, respectively, and considered over-saturation cases. The kinematic equation was designed to estimate the sample vehicle's position in the queue and its time of joining the queue. The results of field tests and simulation indicated that this approach outperformed the optimization-based method proposed by a prior study [33]. One of the primary factors contributing to the enhanced performance was the avoidance of assuming uniform arrival patterns. Wang et al. [36] introduced a novel queue profile estimation approach at a signalized intersection by building an integer programming model that conforms to the temporal and spatial propagation of shockwaves to estimate the queue profile. The proposed model took the location and speed of probe vehicles and the start time of red as input and then analyzed the collected information to acquire the corresponding queue profile. Unlike previous studies that approximated the queue profile using triangles or polygons, this model could detect the queue profile of any shape. Simulation and empirical data were utilized to evaluate the proposed model. The results showed that the model obtained satisfactory results when the penetration rate was only 10% to 20% and the sampling interval was 20 to 30 seconds.

The abovementioned methods utilizing probe vehicle or CV data exhibit certain notable constraints. Some studies relied on satisfying certain assumptions, such as known arrival rates and signal plan, which may not be feasible in practical implementation. Most shockwave-based studies did not sufficiently consider the internal dynamics of queue formation. In most studies, the performance at lower market penetration levels (e.g., <10%) was merely acceptable but not satisfactory

[37].

2.1.2 Methods using the combination of sensor data and CV data

Both fixed sensor data and probe or CV data have some disadvantages. First, fixed sensors cannot recognize detailed information about a particular vehicle. They can only collect traffic data at discrete spatial points. Second, probe vehicles and CVs are still at a low level of market penetration, resulting in unstable estimation accuracy of existing models. With advances in information technology and sensing, researchers aim to combine probe vehicle or CV data with fixed sensor data to achieve greater efficiency and better estimation accuracy.

Li et al. [38] proposed an event-based approach utilizing data from CVs and loop detectors to estimate real-time queue length. The study investigated the estimation accuracy at different MPRs and proposed a data-fusion method combining probe vehicle trajectory data and loop detector data. The case study based on VISSIM simulation data showed that the event-based estimation algorithm achieved good results. When the MPR is 50%, MAPE is below 18%, and the value of MAPE is still acceptable even with a low MPR of 10%.

Cai et al. [39] proposed a method to estimate real-time cycle-by-cycle queue length at signalized intersections by utilizing the upstream fixed sensors data and the trajectory of probe vehicles. They analyzed three cases based on the spatial relationships between the tail of the queue, the location of the sensor and the vehicle stop or start-up position. The proposed method could identify critical breakpoints by fusing fixed and mobile sensor data to indicate key queue formation and dissipation states. Based on LWR theory, the maximum queue length estimation models of three cases were conducted. Their algorithm was applicable to various traffic conditions, including oversaturated and undersaturated conditions. They tested three maximum queue length estimation models using empirical data, and the results indicated average percentage errors of 11.60% and 9.98% for the first two cases, respectively. In contrast, the third case exhibited an average percentage error of 26.40%. In a later study, Wang et al. [40] provided an approach to estimate vehicle queue length using multi-source detection data under under-saturation and over-saturation, respectively. The LWR theory was applied to the queue dynamics modelling in this research. Then, the analytical formulations were developed to calculate the maximum and minimum queue lengths by fusing the data of probe vehicles and fixed detectors. Ground truth data was used to verify the result of numerical experiments. The results showed that this model was relatively effective for estimating the maximum queue length, with MAPE of 17.09% and 12.28% in two tests. However, it was not ideal for estimating the residual queue length.

Shahrbabaki et al. [41] developed a novel data fusion approach for second-by-second estimation of queue length. The data required for this method included the in-flow rate collected from a fixed detector and the position and speed of the CVs. When estimating the position of the tail of the queue, they provided a probability-based method to reduce the errors caused by low MPRs, which made the proposed approach more robust to different MPRs of CVs. Based on traffic flow theory, a nonlinear function was established to determine the number of vehicles in the queue according to the rear of the queue and the average speed of CVs. Three types of toad segments with different MPRs of CVs were tested and demonstrated in a real micro-simulation environment. The results showed that the proposed approach was efficient for high-resolution estimation, even in measurement noise. In addition to traditional fixed sensor data, license plate recognition (LPR) data is also utilized with probe vehicle data to estimate queue length. Tan et al. [42] proposed a lane-based queue length estimation method based on the combination of LPR data and probe vehicle trajectory. The research first matched the LPR data with the probe vehicle data. The aim is to restrict the two-dimensional probability density distribution of discharge headway. Then, the lane-based queue length in the condition of under-saturation and the initial queue in the condition of over-saturation with the maximum possibility were derived by applying Bayesian theory, respectively. Both simulation and empirical data were used to evaluate the performance of the proposed model. The results showed that this method could accurately estimate the queue length under under-saturation and over-saturation conditions and obtain reliable estimation even under a low penetration rate (3%).

Comert and Cetin [43] provided a model for queue length estimation based on CVs as well as range sensors such as light detection and ranging camera at signalized intersections. The proposed model is a simple input-output model, so that it is simple to implement and adopt cyclic queues. They validated this model with both simulation data and field experiment data. Compared with queue length estimations without range sensors, the results of simulation data showed that the proposed model could reduce estimation errors by about 25% and that utilizing the field test data demonstrated an error reduction of nearly 10%

2.2 Research on signal optimization based on CVs

In the context of traffic signal optimization in CV environments, the optimization objectives typically include minimizing queue length, travel time, delay, or other metrics while also being subject to constraints such as maximum/minimum cycle length or phase duration. The control variables are usually signal timings or phase sequences, and the environmental input typically consists of vehicle arrivals.

According to the control schemes, CV-based traffic signal control can be further categorized into three types: actuated traffic signal control, platoon-based traffic signal control and planning-based traffic signal control based on CVs data [48]. For these three control types, the approach to signal optimization might be slightly different. For instance, the actuated signal control might solely extend or reduce green light durations based on estimated traffic flow rate or vehicle arrivals, while planning-based approaches will optimize phase sequence, phase duration and cycle length within a particular forward time window.

The distinctions among these three control methods are also reflected in the level of detail in their predictions of future traffic states [48]. The actuated traffic signal control enhanced by CVs data will predict some relatively coarse and aggregated traffic measures (e.g., average flow rate, queue length, total delay) and make control decisions based on these predictions. The platoon-based signal control will identify vehicle platoons instead of individual vehicles and predict their arrivals or trajectories, thereby making adjustments to the signal timing plan more manageable. The planning-based method will consider the arrival or trajectory of each individual vehicle and optimize signal timing or phase within a forward time horizon through the adoption of more accurate and complex models. Among these three control methods, the actuated signal control exhibits better robustness and flexibility, while the planning-based methods have the capability to construct more refined traffic flow models and optimal control strategies.

2.2.1 Actuated Traffic Signal Control Based on CVs Data

Actuated control can dynamically adjust signal timing in response to changes in real-time vehicle arrivals. The traditional actuated traffic signal control relies on the collection of information using loop detectors installed several meters upstream from the stop line. However, the information gathered by loop detectors is inaccurate and limited in spatial scope. Therefore, Yin et al. [49,50] proposed relevant rough models to describe traffic states, but these models cannot adequately represent changes in traffic demand and vehicle arrivals. However, the emergence of CV technology has addressed these shortcomings. Based on accurate location information from arriving CVs, we can promptly adjust the phase sequence and duration.

For instance, Gradinescu et al. [51] developed an actuated control algorithm based on CVs. Utilizing the vehicle information obtained through V2I (Vehicle-to-Infrastructure) and I2I (Infrastructure-to-Infrastructure) communication technologies, they estimated the traffic flow rate for each approach and then calculated the optimal cycle length for the next cycle by the Webster method. The green time was assigned to each movement based on the saturation levels, subject to constraints on maximum/minimum green time and minimum pedestrian crossing time. Their simulation results indicate that this approach can reduce delay time and fuel consumption compared with fixed-time signal control.

2.2.2 Platoon-based Traffic Signal Control Based on CVs Coordination

The platoon-based traffic signal control involves clustering vehicles into platoons according to certain rules. The leader vehicle of the platoon is responsible for obtaining the movement status of platoon members and interacting with roadside infrastructure. The signal controller then optimizes signal timing based on information such as the length of the platoon and the predicted arrival time of the platoon. With the advent of V2X technology, the rational division of queues becomes more practically applicable, enabling the generation of optimal platoon-based signal timing plans accordingly [52,53].

Pandit et al. [54] leveraged real-time speed and location information transmitted by vehicles in the vehicular ad-hoc network (VANET) and used online algorithms to make scheduling decisions through the central control center. They treated the intersection control problem as a job scheduling problem for the central processor, considering groups of vehicles with similar lengths as unit jobs and establishing a model with intersection delay as the optimization objective. The Oldest Job First (OJF) algorithm was utilized to determine the release order based on the arrival schedule of vehicles. The experimental results demonstrate that, in comparison to fixed timing, this strategy can significantly reduce delays only in scenarios with lower traffic volumes. Moreover, the algorithm necessitates the central controller to have access to a complete vehicle arrival schedule, limiting its applicability to the fully CVs environment.

He et al. [55] introduced a dynamic process model based on platoons to manage traffic signals on the main road. They utilized collected sample vehicle data and a headway-based identification algorithm to recognize the vehicle platoon approaching the intersection. Subsequently, they regarded the signal control problem as a mixed-integer programming problem. By leveraging platoon information rather than individual vehicle data, the number of variables in the mixed-integer programming calculations was reduced, making it relatively easier to solve. The model demonstrated good applicability under both oversaturated and non-saturated traffic conditions.

However, due to the simplification of individual vehicles into platoons, the platoon-based methods may only generate suboptimal strategies. Additionally, the effectiveness of signal control may significantly depend on the performance of algorithms used to identify vehicle platoons. Further research is needed on how to accurately define, model, and aggregate platoons based on actual traffic flow.

2.2.3 Planning-based Traffic Signal Control Based on CVs Data

Planning-based methods typically predict traffic states within a future time window and then construct optimization models with objectives such as traffic flow efficiency or fuel economy. The optimization models are usually formulated as integer nonlinear programming problems, and various methods that involve approximating and reformulating are often utilized to solve them. Dynamic programming (DP) is one of the most widely used techniques for reformulating and solving such control problems [56-58].

Feng et al. [57] proposed a real-time traffic signal control method designed for CV environments. They first estimated the states of unequipped vehicles based on speeds and positions of known CVs and then generated a comprehensive predicted arrival table within a forward time window. Subsequently, they developed a two-level optimization model to generate the optimal signal timing and phases. The upper level employed dynamic programming to determine the minimum and maximum feasible barrier group lengths, while the lower level was formulated as an integer programming problem aiming to minimize either total vehicle delay or queue length. Simulation results in VISSIM indicated that the model performed well even under low MPRs compared with the actuated control. When minimizing vehicle delay was the objective, the delay was lower; however, when minimizing vehicle queue length was the goal, the performance across phases was more balanced.

Li and Ban [59] proposed another DP-based method with the goal of minimizing fuel consumption and travel time within a fixed cycle length. They constructed a DP model by treating each phase as a stage and estimating the fuel consumption and travel time within that stage. By introducing the end-stage cost and a branchand-bound regulator into the DP model, they ensured that the optimal solution of the model would adhere to a fixed cycle length. Simulation results indicated an improvement in control performance compared with the actuated control methods. The above planning-based methods primarily focus on signal optimization control at individual intersections. For the optimization problem of signal control in multiple intersections, Priemer et al. [60] conceptually introduced the application of V2I communication technology in the signal control of multiple intersections for the first time. They assumed that the length of the communication area could reach up to 300m. The communication data included vehicle ID, position, speed, timestamp, and other traffic information collected by in-vehicle devices. Due to communication distance limitations, the optimization step was set to 5s, and the optimization time window was 20s. They used dynamic programming and exhaustive enumeration methods to find the optimal solution. Simultaneously, they analyzed different MPRs of CVs. By optimizing control for multiple intersections, the results showed that compared to TRANSYT-7F, it could reduce delays by 24% and increase vehicle speed by 5%.

Beak et al. [61] developed a two-level approach for signal optimization and coordination at a traffic corridor. At the intersection level, dynamic programming (DP) was employed to generate the optimal phase duration while considering coordination constraints among multiple intersections. At the corridor level, a mixed-integer linear program could provide the optimal offset, which was then fed back to the intersection level as coordination constraints. Simulation results demonstrate that the proposed approach could decrease both the average delay and the number of stops for the entire corridor compared with the traditional actuated-coordinated signal control methods.

To alleviate the complexity and computational load associated with network-wide optimization problems, thereby enhancing the efficiency of signal coordination optimization on a larger-scale road network, a distributed control method has been proposed. Isiam et al. [62] proposed a distributed signal coordination optimization method in a CV environment. The basic idea of this method is to independently optimize individual intersections within control units, significantly reducing the complexity of coordinated control. By coordinating with neighbouring intersections, the distributed control method could avoid finding local optimal solutions. The results show that compared to the actuated coordinated control, this method can improve intersection throughput by 1%-5% and reduce travel time by 17%-48%.

Planning-based signal control has the ability to predict future traffic states and determine the optimal solutions within a specified forward time window, resulting in enhanced optimization effects. However, careful consideration is required when selecting the length of the prediction horizon. A longer prediction horizon allows for the utilization of more future information, but it may come with increased computational burden and prediction errors. Conversely, a shorter prediction horizon may expedite computations but could potentially lead to reduced control performance due to the lack of future traffic information [48].

2.3 Research Gap

According to the literature review of previous studies, it is evident that relatively rich results in the research of queue profile prediction and signal optimization under the CV environment have been achieved. However, this thesis contributes to the literature from the following perspective:

(a) The traditional traffic model and deep learning model are integrated to predict the queue profile, which combines the benefits of both methods.

(b) To estimate the status of non-CVs, some existing studies still require supplementary data sources such as loop detectors or cameras, or rely on satisfying certain assumptions such as known arrival rates. This thesis achieves a relatively high prediction accuracy of queue profiles and develops an effective signal optimization performance solely through the use of CV data.

(c) Most existing studies only utilize partial CV information, such as the first and last stopped CVs, when predicting the queue lengths in a mixed traffic environment. However, this thesis fully and directly utilizes all CVs information within the communication range, enabling a comprehensive prediction of queue profile changes in the future.

2.4 Arrangement of chapters

This thesis focuses on the queue profile prediction and signal optimization in a mixed traffic environment. The whole thesis is organized as follows.:

Chapter 1 is the introduction, which introduces the current background of traffic signal control systems and analyzes the advantages of CV technology.

Chapter 2 reviews the existing studies related to queue length prediction and signal optimization under CV environment, and analyzes the purpose of this research topic.

In Chapter 3, the basic theory of LSTM Neural Networks is introduced, and an encoder-decoder LSTM model structure is proposed. After that, the shockwave theory is presented to describe the formation and discharging of vehicle queues, and the queue profile prediction model based on the predicted future in-flow rate is introduced. Then, we introduce a signal optimization method based on Dynamic Programming with the objective of minimizing the total vehicle delay within the optimization time window.

Chapter 4 first shows the numerical experiment results of the proposed LSTM model, including the process of model selection and final prediction performance. Then, we test the performance of the proposed queue profile prediction model and signal optimization algorithm at a virtual intersection and a real-world intersection using the INTEGRATION simulator. The results are analyzed under different MPRs of CVs and various traffic demand levels.

Chapter 5 provides a comprehensive summary of the entire thesis, presents the conclusions of this study, and outlines potential directions for future research.

Methodology

Figure 3.1 indicates the framework of the whole signal control system. First, an encoder-decoder LSTM model is employed to predict future in-flow rates. Then, based on the predicted future in-flow rates, the LWR model is applied to analyze shockwave propagations to predict the queue profile. Finally, a signal optimization algorithm based on DP is developed with the predicted queue profiles, which can allocate the optimal phase sequence and duration to minimize total vehicle delay at the signalized intersection.

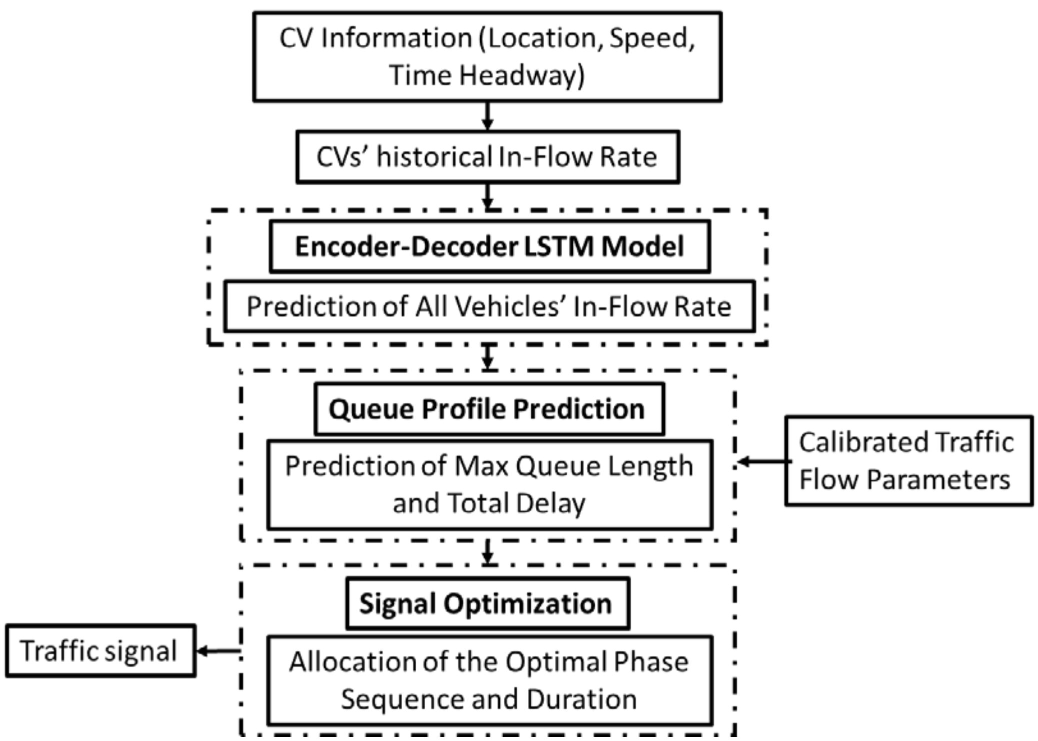


Figure 3.1 The framework of the signal control system

3.1 LSTM-based in-flow rate prediction

This research aims to build an LSTM model that can predict all vehicles' future in-flow rate based on CVs' historical in-flow rate, which means the inputs and outputs of this LSTM model will be derived from two different but correlated time series datasets. In order to better capture the temporal relationship between these two datasets, a kind of Encoder-Decoder LSTM structure is utilized. The workflow is shown in Figure 3.2.

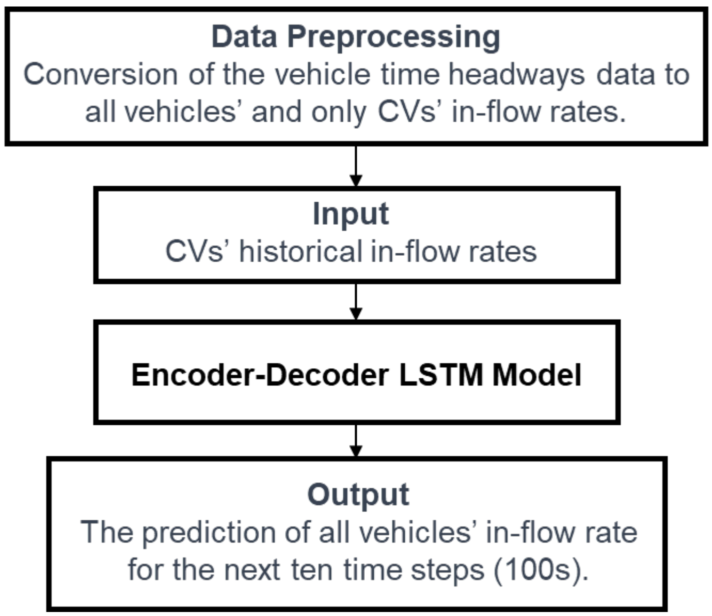


Figure 3.2 Workflow of traffic in-flow rate prediction

3.1.1 Review of LSTM network

Deep learning has presented distinctive opportunities for addressing more intricate issues in transportation. In the research of traffic flow prediction, since traffic flow data can be considered a time series data, we should analyze the inter-dependency relationship between historical traffic flow data from previous time periods and predicted traffic flow for the target period. However, Recurrent Neural Network (RNN), the typical deep learning technique, faces challenges in effectively solving long-term dependency issues, and its performance in short-term traffic flow prediction is not ideal [63]. Therefore, this study adopts an improved Long Short-Term Memory (LSTM) network based on the original RNN for traffic flow prediction, which is capable of handling data with temporal or sequence structures and varying input lengths [64].

An LSTM network consists of three layers: the input layer, the recurrent hidden layer, and the output layer. The structure of the LSTM model's hidden layer is illustrated in Figure 3.3. It primarily includes three gate control units: the input gate, the forget gate, and the output gate, along with a memory cell unit. Unlike traditional neural networks, LSTM features a memory cell divided into two parts: the short-term state or hidden state ($h_{(t)}$) and the long-term state or cell state ($c_{(t)}$). The hidden state ($h_{(t)}$) represents the output of the layer at a specific time step (t), while the cell state ($c_{(t)}$) preserves information to capture long-term dependencies among the current hidden state and the previous hidden states across time. As the information traverses from left to right, the cell state undergoes memory forgetting through the forget gate and subsequently incorporates new memories through an addition operation.

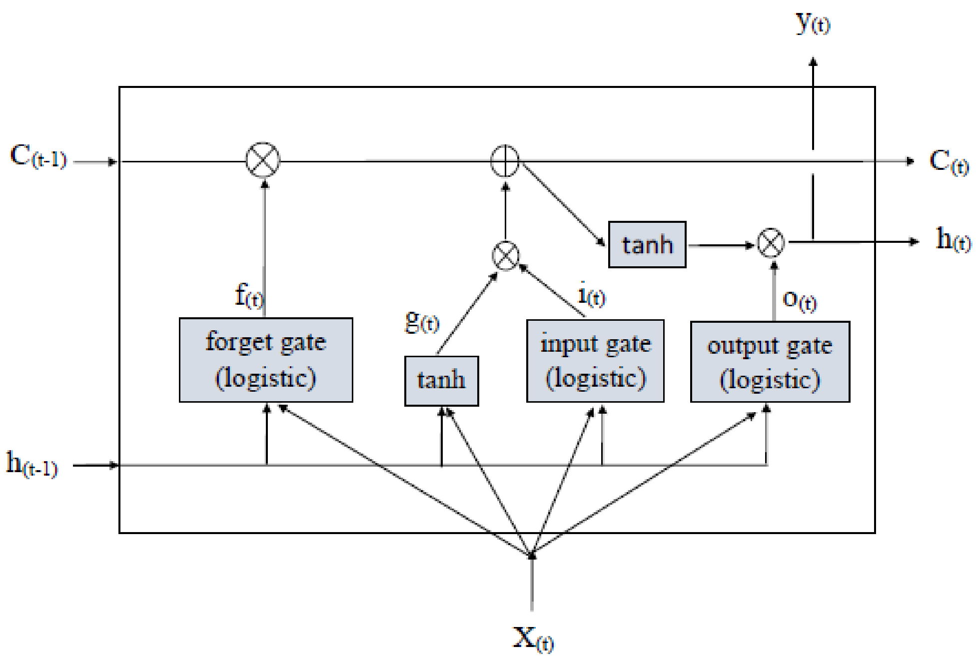


Figure 3.3 The complete structure of an LSTM hidden layer

As illustrated in Fig. 3.3, the input vector ($X_{(t)}$), the previous hidden state ($h_{(t-1)}$) and the previous cell state ($c_{(t-1)}$) are inputted into the LSTM hidden layer. The memory cell input ($g_{(t)}$) at the given time step (t) is calculated based on the input vector ($X_{(t)}$) and the previous hidden state ($h_{(t-1)}$) through a Tanh activation function. The three gate control units employ Sigmoid activation functions, resulting in outputs ranging from 0 to 1. When the output is 0, the gate is in a closed state, indicating that information cannot pass through, and the processed information is discarded at this point. When the output is 1, the gate is in an open state, signifying that any vector multiplied by it remains the vector itself, allowing the information to pass through and propagate to the next layer. Among them, the purpose of the forget gate $f_{(t)}$ is to prevent information that has little or even no impact on the model output from being passed to the next layer. By controlling which parts of the cell state should be thrown away, the forget gate determines how the cell state of the current time step filters the information of the cell state of the previous time steps. The input gate $i_{(t)}$ is responsible for determining which components of the input should be incorporated into the cell state, thereby ensuring the retention and subsequent transmission to the next layer of data information that holds substantial influence on the final output. The output gate $o_{(t)}$ is employed to regulate the impact of cell memory on the current output, determining which components of the cell state should be accessed and provided as output ($h_{(t)}$) at the current time step (t). The mathematical formulation of these operations is expressed as follows:

Input gate control unit:

$$i_{(t)} = \sigma\left(W_i^t \cdot \left[h_{(t-1)}, X_{(t)}\right] + b_i\right) \tag{1}$$

Forget gate control unit:

$$f_{(t)} = \sigma\left(W_f^t \cdot \left[h_{(t-1)}, X_{(t)}\right] + b_f\right) \tag{2}$$

Output gate control unit:

$$o_{(t)} = \sigma\left(W_o^t \cdot \left[h_{(t-1)}, X_{(t)}\right] + b_o\right) \tag{3}$$

The memory cell input:

$$g_{(t)} = \tanh\left(W_g^t \cdot \left[h_{(t-1)}, X_{(t)}\right] + b_g\right) \tag{4}$$

where, W_i, W_f, W_o, $W_g$are the weight coefficients of each of the gates and b_i, b_f, b_o, b_g are the bias terms for each of the gates. Each weight coefficient is composed of the weight matrix of ($h_{(t-1)}$) and the weight matrix of ($X_{(t)}$). σ represents the sigmoid function and $\tanh$ represents the hyperbolic tangent function. $\otimes$ is the Hadamard product operator. Moreover, the cell and hidden states are computed using the following equations:

The cell state:

$$c_{(t)} = f_{(t)} \otimes c_{(t-1)} + i_{(t)} \otimes g_{(t)} \tag{5}$$

The hidden state:

$$h_{(t)} = o_{(t)} \otimes \tanh\left(c_{(t)}\right) \tag{6}$$

The training of the LSTM neural network can be realized with the algorithm of back propagation through time (BPTT) [65], which updates the weight coefficient and the bias term of each gate control unit and the memory cell input through each training.

3.1.2 LSTM framework for in-flow rate prediction

In this research, we can obtain the time headway of all vehicles and the time headway of CVs, a subset of all vehicles' headway, from the vehicle trajectory data. Because the flow rate is equal to the inverse of the mean time headway, we can calculate all vehicles' in-flow rate and CVs' in-flow rate, respectively. Actually, traffic flow rate is considered a type of time series data; thus, we aim to develop an LSTM-based time series forecasting model, which is essentially a supervised learning problem. We use CVs' historical in-flow rate as the input and all vehicles' future in-flow rate as the output to capture the correlation between these two different but related time series datasets. To more accurately approximate the real underlying mapping between these two datasets, we utilize a kind of "stacked encoder-decoder LSTM model", and the structure is shown in Figure 3.4.

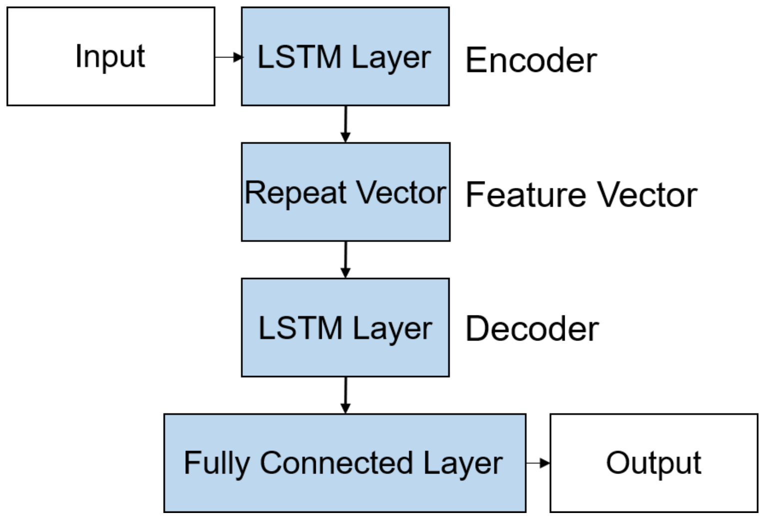


Figure 3.4 The framework of Encoder-Decoder LSTM model

The construction of an Encoder-Decoder LSTM model involves a straightforward process. In this approach, all time steps first get put through the encoding layer, generating a set of hidden states (one for each time step). These hidden states are then encapsulated by a feature vector to hold the sequential information and serve as inputs for the decoding LSTM layer. Finally, the decoding layer sequentially decodes the results of the encoder to assist the model in making predictions. This structure is capable of forecasting variable-length output sequences and designed explicitly for sequence-to-sequence problems. In this research, the utilization of this structure could help us more accurately approximate the real underlying mapping—from the historical in-flow rates sequence estimated solely based on CV information to another in-flow rates sequence of all vehicles in the future.

In this approach, a Vanilla LSTM layer function is used as the encoder, responsible for comprehending and interpreting the input sequence. The encoder generates a fixed-length vector, representing the model's interpretation of the input sequence. Following the encoder, the fixed-length output is replicated multiple times, corresponding to each required time step in the output sequence. This represents the model's internal representation of the input sequence. Subsequently, this replicated sequence is fed into the decoder, which is also an LSTM layer. Unlike traditional models that only output the end of the sequence, the decoder outputs the entire sequence. Each neuron unit in this layer produces a value for every element in the output sequence. Finally, a fully connected layer is employed to interpret each time step in the output sequence before the final output. The same fully connected layer and output layer are applied to process each time step in the decoder output sequence by wrapping the model's output segment in a TimeDistributed wrapper.

3.2 Queue length and total delay prediction with traffic flow model

This research proposes a queue length and total delay prediction model based on the LWR traffic flow model at the intersection. Once we get the predicted all vehicles' in-flow rate from the LSTM model and the pre-calibrated parameters of the fundamental diagram, we can easily calculate the shockwave propagations to predict the queue profile. Then, the prediction of maximum queue length and total delay are calculated based on the predicted queue profile. The workflow is shown in Figure 3.5.

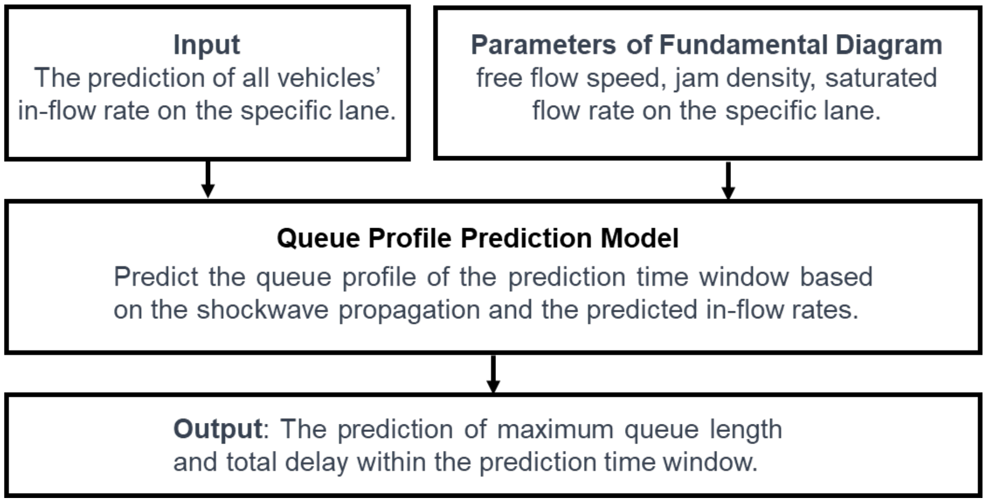


Figure 3.5 Workflow of queue profile prediction

3.2.1 Shockwave analysis

LWR traffic flow model is one traditional kinematic wave model which assumes that traffic flow is a function of density at any point on the road [66]. From the LWR model, we can derive the traffic shockwave theory, which is defined as the propagation of an abrupt change in traffic stream density [67], and the wave velocity can be calculated by the following equation:

$$u_w = \frac{\Delta \mathrm{q}}{\Delta k} = \frac{q_2 - q_1}{k_2 - k_1} = \frac{k_2 u_2 - k_1 u_1}{k_2 - k_1} \tag{7}$$

where q_1, k_1, u_1 are the traffic flow, density and velocity of the upstream region and q_2, k_2, u_2 are the traffic flow, density and velocity of the downstream region. This research focuses on traffic shockwaves generated from a triangular fundamental diagram, as shown in Figure 3.6, where v_f is the free flow speed, (q_m, k_m) is the flow and density under the capacity and k_j is the jam density. In addition, (q_a, k_a) is a given traffic state of the arrival traffic stream. The slope of the line connecting any two points (i.e., two different traffic states) on the triangular fundamental diagram gives the shockwave speed.

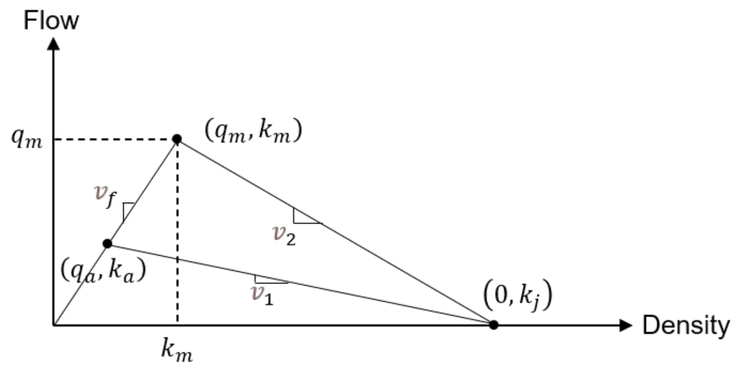


Figure 3.6 The triangular fundamental diagram

At signalized intersections, various traffic shockwaves are generated due to the stop-and-go traffic caused by signal changes [68]. The interaction of these shockwaves results in the formation and dissipation of vehicle queues, as shown in Fig. 3.7 for more universal oversaturated conditions. If vehicles arrive at the intersection when the signal is red, they are forced to stop to wait for a green indication, which creates different traffic states between the arrival and the stopped traffic stream. Such abrupt change of traffic states forms a queuing shockwave $v_{n,i}$ propagating from the stop line to the upstream of the intersection with the speed:

$$v_{n,i} = \frac{0-q_a^{n,i}}{k_j-k_a^{n,i}} \tag{8}$$

where n is the cycle number and i indicates the ith traffic state of the arrival traffic stream. This research assumes the traffic state of the arrival traffic stream is stable within a small time interval (e.g., 10 seconds), so the arrival vehicles have piece-wise arrival rates: $q_a^{n,i}$ and $k_a^{n,i}$ are the arrival flow rates and density of the ith traffic state. The dash-dot lines in Fig. 3.7 represent the shockwaves between any two consecutive traffic states of the arrival traffic stream. From the triangular fundamental diagram, it can be deduced that all these shockwaves have the same velocity v_f. When these shockwaves meet the queuing shockwave, a series of turning points are formed on the piece-wised queuing shockwave curve.

The queuing shockwave $v_{n,i}$ keeps moving to the upstream of the intersection. At the beginning of the effective green T_g^n, the queue begins to be released under saturation flow rate, forming the second shockwave, which is defined as the discharge shockwave at the stop line moving upstream with the speed:

$$v_2 = \frac{q_m - 0}{k_m - k_j} \tag{9}$$

In general, the speed of the discharge shockwave v_2 is higher than v_1, so the discharge wave will catch the queuing wave at time T_{max}^n, when this approach has the maximum queue length. Once these two shockwaves meet, a third shockwave, defined as the departure shockwave, is generated and propagates from the queue tail toward the stop line with the speed:

$$v_3 = \frac{q_m - q_a^{n,i}}{k_m - k_a^{n,i}} = v_f \tag{10}$$

At the start of the red time of the next cycle, if the queue is unable to completely discharge, a residual queue emerges, constituting the fourth shockwave propagating upstream with the speed:

$$v_4 = \frac{0 - q_m}{k_j - k_m} \tag{11}$$

Shockwave v_4 describes the residual queue forming process. Waves v_3 and v_4 have inverse directions so that they will meet at time T_{min}^{n+1} when the approach has minimum queue length, i.e., the residual queue. As soon as the two waves meet, a new queuing wave of the $n + 1$th cycle is generated.

The shockwave motion outlined above illustrates the cyclical process of the formation and discharging of the vehicle queue at a signalized intersection. In summary, the queues commence accumulating when the red time starts, reaching the minimum queue length (i.e., residual queue) shortly after. As the green time starts, the queues begin to discharge and achieve the maximum length sometime after the start.

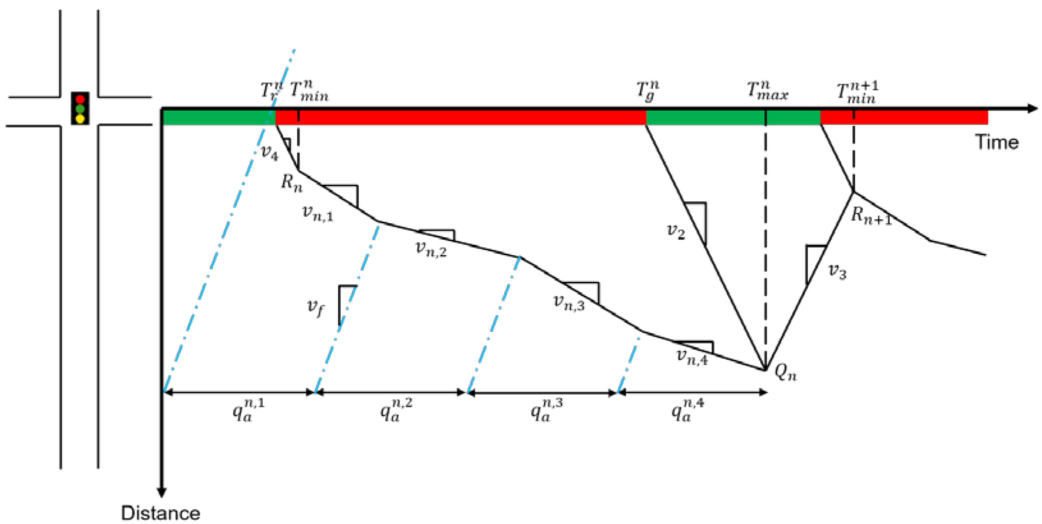


Figure 3.7 Shock wave propagation

3.2.2 Queue profile prediction

Generally, our goal is to predict the total delay or maximum queue length, which can be easily calculated once we obtain the queue shockwave profile. The queue shockwave profile is a polygon in the x-t plane enclosed by the queuing shockwave and discharge shockwave, as indicated by the red shaded area in Figure 3.8 for

Link to Table of Contents

oversaturated conditions. In this research, the key parameters of the triangular fundamental diagram have been pre-calibrated based on historical data for each approach, so the slopes v_2, v_3 and v_4 are known constants. Therefore, once we can obtain the predicted arrival traffic flow $q_a^{n,i}$, it is straightforward to predict the queue shockwave profile. The total delay D_n and the maximum queue length Q_n could be calculated as follows:

$$Q_n = v_4(T_{min}^n - T_r^n) + \sum v_{n,i}(T_i^n - T_{i-1}^n) \tag{12}$$

$$D_n = \left(\frac{v_4(T_{min}^n - T_r^n)^2}{2} + \sum \frac{(Q_{i-1}^n + Q_i^n)(T_i^n - T_{i-1}^n)}{2} - \frac{v_2(T_{max}^n - T_g^n)^2}{2}\right) \cdot k_j \tag{13}$$

Where T_i^n represents the time when the ith queuing shockwave ends and Q_i^n represents the location where the it ends. If $i = 0$, we have $T_0^n = T_{min}^n$ and $Q_0^n = R_n$ for oversaturated conditions, or $T_0^n = T_r^n$ and $Q_0^n = 0$ for undersaturated conditions. If $i = I$ where I denotes the last segment of queuing shockwave, we have $T_I^n = T_{max}^n$ and $Q_I^n = Q_n$.

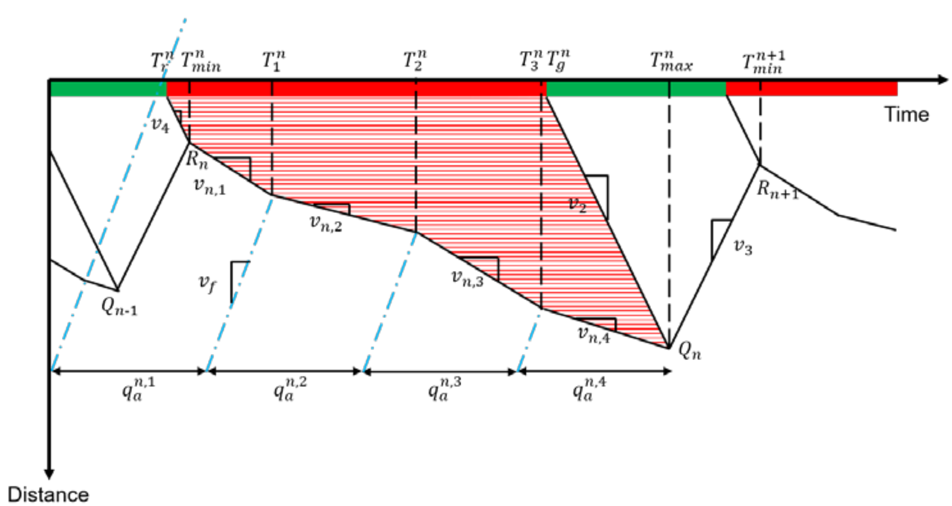


Figure 3.8 Queue Profile at the intersection

3.3 Development of Signal Optimization Algorithm

This research proposed a signal optimization algorithm based on dynamic programming, which can allocate the optimal phase sequence and duration within the optimization time window. In this case, the calculation of the performance measure is based on the predicted total delay from the queue profile prediction model. The workflow is shown in Figure 3.9.

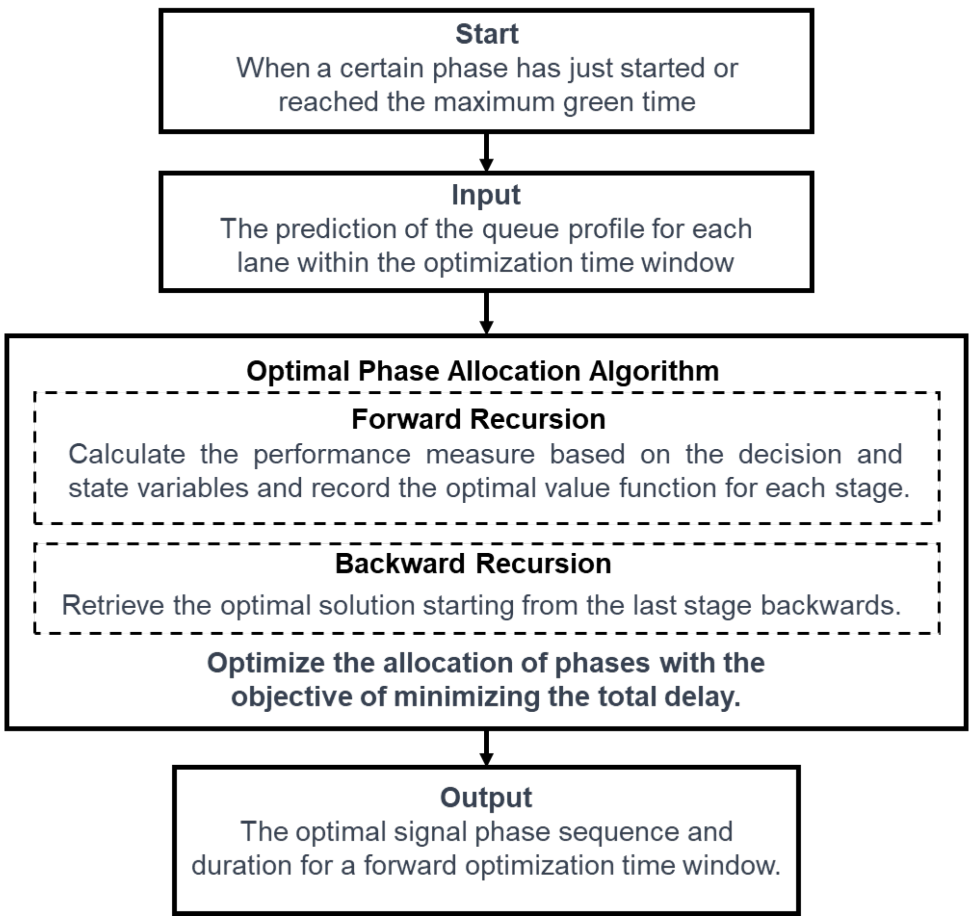


Figure 3.9 Workflow of signal optimization

3.3.1 Optimal phase allocation algorithm

The phase allocation algorithm can assign the sequence and duration of signal phases based on the given initial phase at start time 0 and the predicted in-flow rate within the optimization period [0, T] in order to optimize specific performance metrics (such as delay or queue length). This research will utilize dynamic programming (DP) to optimize signal phase allocation. DP is a mathematical method for optimizing decision processes, which can decompose the given decision problem into a series of subproblems. It will calculate appropriate measures of effectiveness recursively and then combine the solutions of subproblems to obtain the solution of the original problem [69]. Because many subproblems are usually similar, DP attempts to solve each subproblem only once to reduce the computational burden. Once the solution to a given subproblem is determined, it will be memorized and stored so that it can be directly accessed the next time the same subproblem needs to be solved without redundant calculations. Clearly, allocating signal phases within the optimization period is a typical DP problem. Therefore, DP is applied to each phase defined as a combination of non-conflicting movements through the intersection. Figure 3.10 shows one possible set of four phases, including eight movements at an intersection.

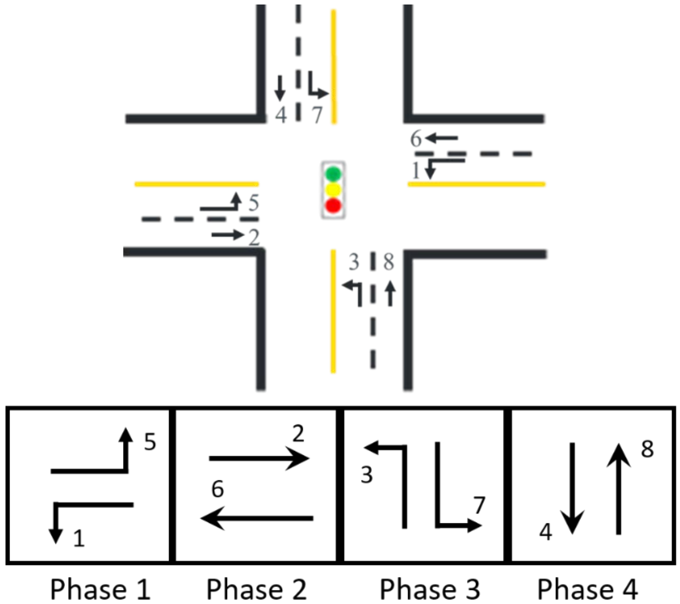


Figure 3.10 Sample signalized intersection and phases setting

The following notation is used to define the algorithm:

P: Set of phases. $|P|$ denotes the total number of phases.

ϕ: Index of individual phase.

T: Total number of time steps in the optimization time window, in seconds.

g: Minimum green time

G: Maximum green time

r: Clearance interval which is the total of the yellow and all red clearance times.

j: Index of DP stages

x_j: Control variable denoting the green time of stage j.

s_j: State variable denoting the total number of time steps allocated to stage j.

$X_j(s_j)$: Set of feasible control decisions, given state variable s_j.

$f_j(s_j, x_j)$: Performance measure at stage j, given state variable s_j and control variable x_j.

$v_j(s_j)$: Value function given state variable s_j which represents the cumulative performance measure for the current and all previous stages.

Given a value of the state variable s_j, the discrete set of the feasible control variable x_j can be calculated by:

$$X_j(s_j) = \begin{cases} \{0\}, if\ s_j - r < \ g \\ \{0, g, g+1, \dots, G\}, if\ s_j - r \geq \ G \\ \{0, g, g+1, \dots, s_j - r\}, otherwise. \end{cases} \tag{14}$$

The relationship between two continuous stages is described as follows:

$$s_{j-1} = s_j - h_j(x_j) \tag{15}$$

Where

$$h_j(x_j) = \begin{cases} 0, if\ x_j = 0 \\ x_j + r, otherwise \end{cases} \tag{16}$$

Figure 3.11 indicates this relationship between s_{j-1}, s_j and x_j.

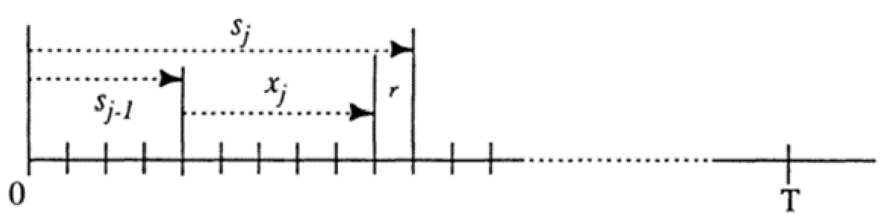


Figure 3.11 Relationship between two consecutive stages

The developed DP is solved utilizing a forward and a backward recursion. The forward recursion is used to calculate the value function, which represents the cumulative performance measure of all prior stages based on each decision and the corresponding state variables, and it records the optimal value function and the control variable as the solution of each state. The Backward recursion, on the other hand, starts from the last stage and recovers the optimal decisions by determining the optimal solution of each state and the associated optimal control variable backwards. The specifics of the forward and backward recursion are outlined as follows:

(a) Forward recursion

The forward recursion allocates time to phases as stages in the DP problem. Considering each phase in a cycle as a stage, this algorithm starts with an initial phase and plans as many phases as necessary to obtain the optimal solution. In fact, this algorithm allows skipping any phase by assigning $x_j = 0$ to the corresponding stage when it is beneficial to do so, which ensures the flexibility of the phase sequence. In addition, this algorithm does not generate a fixed cycle length and does not pre-set an exact number of stages, thereby accommodating the variability in traffic demands.

The initial value function v_0 for forward recursion is set to 0. Starting from the first stage, the forward recursion calculates and records the optimal ${x_j}^*(s_j)$ for each possible s_j. The performance measure $f_j(s_j, x_j)$ is computed based on total delay or maximum queue length from the proposed queue profile prediction model. The stopping criterion for the forward recursion is when the value functions for stage j and its preceding $|P| - 1$ stages are all equal. At this point, all $|P|$ phases have been considered, and repeating these phases can not achieve a greater value function. The process of forward recursion is described below:

Step 1: Set $j = 1$ and $v_0 = 0$.

Step 2: for $s_j = r, r+1, \dots, T$

$$v_j(s_j) = Min_{x_j}\big(f_j(s_j, x_j\ ) + v_{j-1}(s_{j-1})\big|x_j \in X_j(s_j)\big)$$

Record $x_j{}^*(s_j)$ as the optimal solution in Step 2.

Step 3: if (j < $|P|$), $j = j + 1$, and repeat from step 2.

else if $\big(v_{j-k}(T) = v_j(T)\big)$ for all $k \le |P| - 1$, STOP.

else $j = j + 1$, and repeat from step 2.

(b) Backward recursion

During the process of forward recursion, the optimal decision $x_j{}^*(s_j)$ corresponding to each s_j has been found and recorded. The stopping rule is that the last $|P| - 1$ stages do not lead to a further increase in the total value function. Therefore, in the backward recursion, we can trace back from Stage $J - (|P| - 1)$ to obtain the optimal trajectory of state variables s_j^* and the corresponding optimal control variable $x_j{}^*(s_j^*)$, where J represents the index of the last stage when the forward recursion stopped. The backward recursion is summarized as follows:

Step1: $s^*_{J-(|P|-1)} = T$

Step2: for $j = J - (|P| - 1), J - (|P| - 1) - 1, \dots, 1$

Read $x_j{}^*(s_j^*)$ from the memorized table computed in the forward recursion

If $j > 1$, $s^*_{j-1} = s^*_j - h_j\left(x_j{}^*(s_j^*)\right)$

3.3.2 Calculation of performance indices

In Step 2 of the forward recursion, for given values of s_j and x_j, it is necessary to calculate the corresponding performance measure $f_j(s_j, x_j\ )$, and compare the performance measure values of all possible values of s_j and x_j to identify the optimal decision. Both total delay and maximum queue length can be considered as the performance indices. The calculation of these performance indices depends on the queue profile prediction discussed in Chapter 3.2.2. For better illustration, we introduce the following notation:

$\phi(j)$: The index in the set of phases P associated with stage j.

$L_{\phi,j}$: A function of s_j and x_j denoting temporary queue profile.

$Q_{\phi,j}$: A function of s_j denoting permanent queue profile.

In this study, we decompose the optimization period [0, T] into a series of sub-intervals [s_{j-1}, s_j], $j = 1,2, \ldots$, to calculate performance indices. For the given s_j and x_j, s_{j-1} is calculated by Eq. (2), thus the time interval [s_{j-1}, s_j] can be regarded as a function of s_j and x_j.

The steps we use to calculate the $Q_{\phi,j}$ are as follows:

(a) Given s_j calculate the temporary queue profile $L_{\phi,j}(s_j, x_j)$ for each $\phi \in P$ and $x_j \in X_j(s_j)$.

(b) Based on the temporary queue profile $L_{\phi,j}(s_j, x_j)$, we can calculate the performance measure $f_j(s_j, x_j)$ which is used in the minimization in step 2 of the Forward Recursion.

(c) Identify $x_j^*(s_j)$ according to the minimization in step 2 of the Forward Recursion. Then the corresponding temporary queue profile $L_{\phi,j}\left(s_j, x_j^*(s_j)\right)$ is recorded as the permanent queue profile $Q_{\phi,j}(s_j)$.

Once we get the permanent queue profile $Q_{\phi,j}(s_j)$, we can use the method proposed in chapter 3.3,2 to calculate the total delay under this queue profile, and the performance measure associated with time interval $[s_{j-1}, s_j]$ is calculated by adding the total delay associated with each phase $\phi \in P$.

Result and Discussion

This section is dedicated to assessing the accuracy of the proposed queue profile prediction and the effectiveness of the signal optimization system in both a virtual intersection and a real-world intersection.

4.1 Case study setting

This section will present two case studies designed to assess the performance of the delay prediction and signal optimization algorithm on the single intersection.

(a) Case Study 1: A virtual intersection

In the first case study, we make a comprehensive simulation of traffic conditions around a four-legged intersection. In this intersection, each approach has an exclusive left-turn lane and a through and right-turn lane. The speed limits, capacity and jam density of all roads are 60km/h, 1600 veh/hr/lane and 160 veh/km, respectively. The length of the upstream communication area is set as 300m. Figure 4.1 shows the simulation setting of traffic demand for each approach under the demand level = 1 and the setting of 4 phases. The basic fixed-time SPaT plan is calculated based on Webster method for the signal optimization.

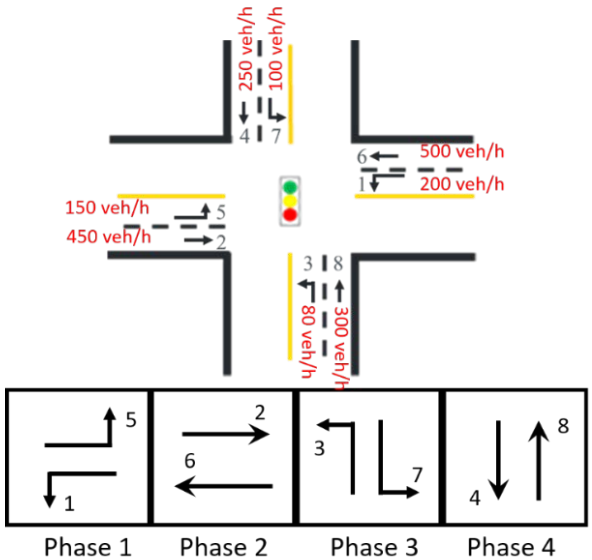


Figure 4.1 The simulation setting of traffic demand for each approach

(b) Case Study 2: A real-world intersection at FM 528

In the second case study, a real-world intersection, FM 528 located in Houston, Texas, is selected as the testbed.

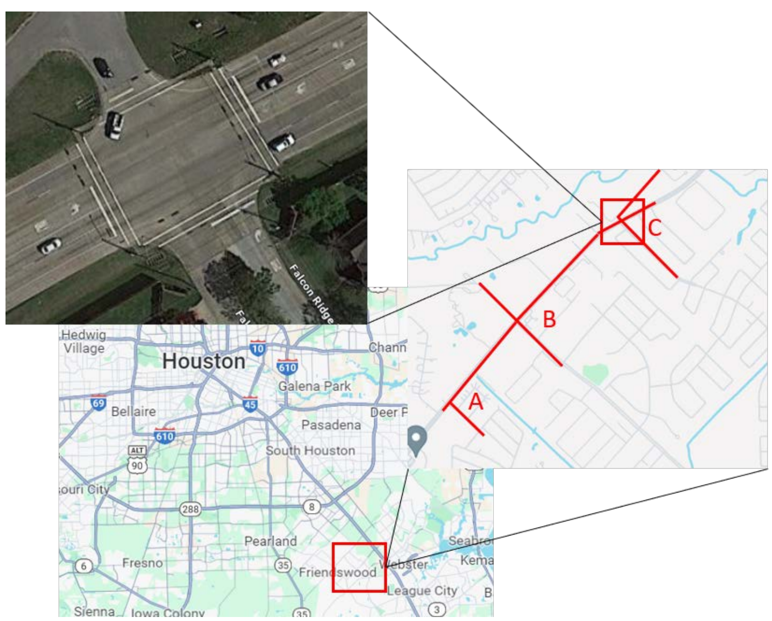


Figure 4.2 Intersection C on FM 528 in Houston, TX

Figure 4.2 shows the location and the geometry of this real-world intersection on FM 528. Further details about this intersection are elaborated below:

(a) Each approach at Intersection C is equipped with two lanes. On the east and west approaches, an additional dedicated left-turn lane is provided, while the rightmost lane on both approaches accommodates through and right-turn movements.

(b) The major road has a speed limit of 72 km/h, while the minor road has a speed limit of 48 km/h.

(c) Currently, the SPaT plan of the actuated signal control system implemented at this intersection, as well as in-flow rates on each approach, are provided by Texas Department of Transportation Houston District Office. Typically, the green extension time for the main road ranges from 1.5 to 2 seconds, while it varies between 2 to 4 seconds for the minor road. Additionally, the green split for the main road traffic fluctuates between 0.48 and 0.64.

4.2 Overview of INTEGRATION microscopic traffic simulator

In this subsection, we provide a brief introduction to INTEGRATION microscopic traffic simulator and elucidate its advantages. The choice of the micro-traffic simulator INTEGRATION, developed by Virginia Tech, as the simulation platform for this study is justified for the following reasons:

(a) Open source:

INTEGRATION serves as an open-source platform, enabling users to seamlessly apply their developed traffic control systems to existing traffic networks. Moreover, the algorithm demonstrates excellent applicability across various operational scenarios in real-world urban intersection configurations.

(b) Well-calibrated model:

INTEGRATION is an integrated simulation and traffic assignment model that conducts traffic simulations by tracking the movement of individual vehicles from origin to destination every 1/10th of a second [70]. This provides a substantial amount of information about the motion state of vehicles. Additionally, the model allows for continuous variation of traffic flow density along a road, providing a flexible representation of spatiotemporal changes in traffic conditions, which aids in the analysis of shockwave propagation. INTEGRATION adopts Van Aerde's fundamental diagram [71] to depict the relationship between flow rates, density, and speed, which is illustrated in Figure 4.3.

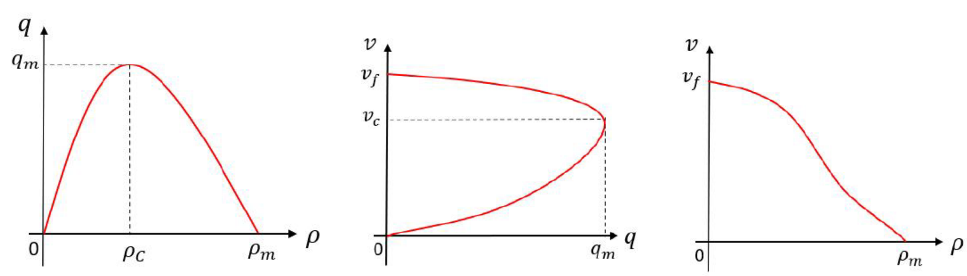


Figure 4.3 The Van Aerde fundamental diagram

Moreover, INTEGRATION employs Van Aerde's car-following model to describe the responsive behavior of the following vehicle due to changes in the motion state of the leader vehicle. It integrates both the Greenshields car-following model and the Pipes car-following model, effectively addressing the limitations of these two models. In terms of lane-changing models, the lane-changing model utilized in INTEGRATION is derived from the Gipps's lane-changing model. It categorizes lane-changing behaviours into autonomous lane change and forced lane change based on whether there are better traffic conditions or if the driver have to change lanes, providing a more realistic simulation of vehicle lane-changing behaviours.

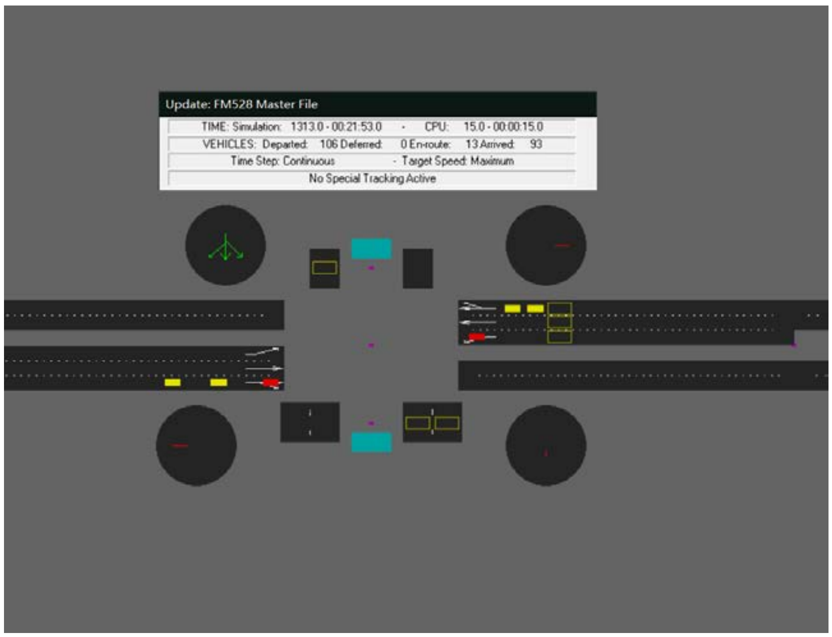


Figure 4.4 INTEGRATION Software Interface for Intersection C on FM528

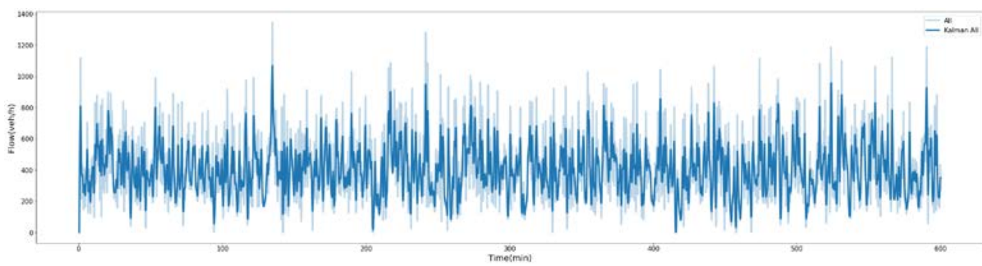
(c) Computational efficiency:

Compared to existing commercial simulators, INTEGRATION exhibits higher computational efficiency, enabling a substantial number of simulations to be conducted in a shorter timeframe. This is crucial for evaluating the effectiveness and applicability of the proposed traffic control system under various traffic scenarios.

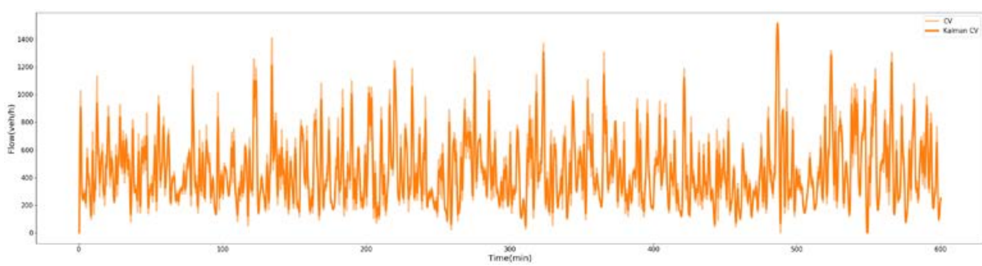
Link to Table of Contents

4.3 Training and testing of the LSTM-based in-flow rate prediction model

In this section, we mainly focused on the in-flow rate prediction for north through movements of the virtual intersection. The 24-hour vehicle trajectory data is obtained through simulation for training and testing the LSTM model. From the raw vehicle trajectory data, the time headway of each vehicle, when it is 300 meters from the stop line, can be obtained for both all vehicles and only a subset of CVs. Based on the relationship between time headway and traffic flow rate, the real in-flow rate calculated from all vehicles and the flow calculated only from CVs can be obtained, respectively. The in-flow rate calculated only from CVs will be used as the input variable, while the corresponding in-flow rate of all vehicles at the next time step is employed as the output variable to train the LSTM model. A sliding window method, that is, the utilization of prior time steps to predict the next time step, is applied to divide the entire dataset into a series of samples. Before the training process, we use Kalman smoother to smooth the in-flow rate data and reduce noises. Figure 4.5(a) and Figure 4.5(b) respectively illustrate the temporal variations in the in-flow rates calculated solely based on CVs or all vehicles before and after data cleaning and smoothing under the MPR of 60%. The light-colored part in the figure represents the in-flow rate data before processing, while the dark-colored part represents the in-flow rate after processing. It can be seen that the Kalman smoother effectively removes the noise of the original data and retains the trend of in-flow rate variation over time. Through the comparison of Figure 4.5(a) and Figure 4.5(b), we observe a similar trend in variation of all vehicles' in-flow rate and CVs' in-flow rate, despite some differences at extreme values. In fact, CVs' in-flow rate is derived from the time headways of a subset of all vehicles. Therefore, there must be some correlation between these two time series datasets, which is expected to be captured by the LSTM model.



(a) All vehicles' in-flow rate



(b) CVs' in-flow rate

Figure 4.5 In-flow rate before and after using Kalman smoother under the MPR of 60%

4.3.1 Hyperparameter Selection of LSTM Model

Neural network model optimization typically requires continuous testing and adjustment based on empirical experience. Optimal settings for various hyperparameters are chosen through comparisons. In this study, the dataset has been divided into three distinct subsets. Specifically, 80% of the data has been allocated to serve as the training set, while the validation set and the test set each constitutes 10% of the total dataset. Table 4.1 presents the initial values of parameters selected based on previous research and preliminary observations. According to the experience, this study employs a dropout ratio of 0.2 for the dropout layer. The activation functions used in both the LSTM and dense layers are all the Rectified Linear Unit (Relu) functions, chosen for their superior gradient propagation and efficient computational performance [59]. The model is optimized using the Adam optimizer, with the mean square error (MSE) serving as the chosen loss function. Initial distribution for the number of neurons in the LSTM layer, batch size, and the number of time steps in one sample are provided. Considering their impact on the predictive performance of the LSTM model on the validation set, these three parameters are individually adjusted, and the optimal parameter combination is ultimately chosen.

Table 4.1 The initial values of the parameters

Parameter	Values
Number of nerons in the encoder layer	$\{16,32,64,128,192,256,320\}$
Number of nerons in the decoder layer	$\{16,32,64,128,192,256,320\}$
Dropout	0.2
Batch Size	$\{16,32,64,96,128,160,192,224,256,288\}$
Optimizer	adam
Loss function	MSE
Activation Function	relu

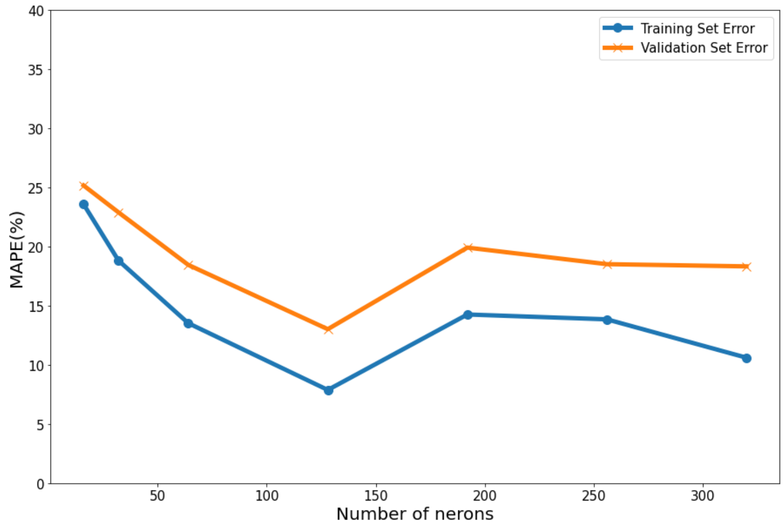


Figure 4.6 The impact of the number of LSTM layer nerons on model performance

The selection for the best number of LSTM layer nerons is based on MAPE in the validation set. From Figure 4.6, it can be observed that when the number of LSTM layer nerons is set to 128, the model's MAPE value is the lowest. Beyond this point, as the number of neurons increases, the MAPE starts to rise. Within the range of neuron values between 192 and 320, with the increase in the number of LSTM layer nerons, the MAPE generally tends to decrease, but it always remains higher than the MAPE value obtained when the number of neurons is 128. Therefore, the optimal choice for the number of LSTM layer nerons in this study is 128.

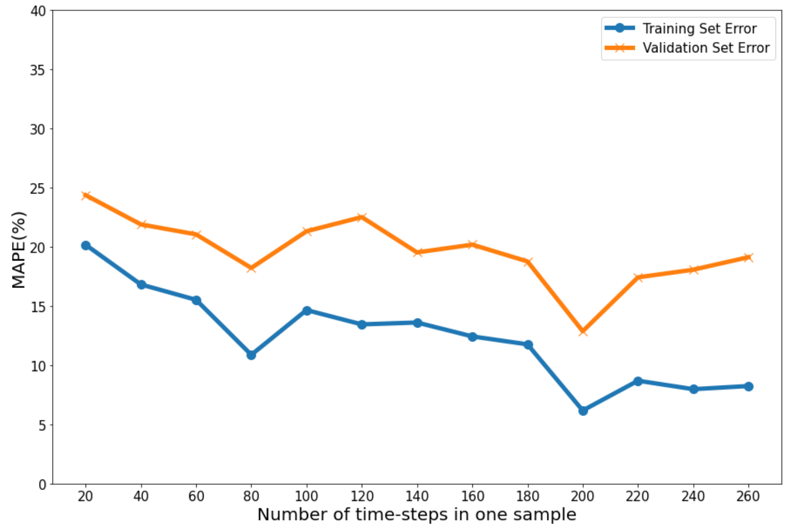


Figure 4.7 The impact of the length of historical data on model performance

The number of time steps in one sample represents the length of historical data. As observed from Figure 4.7, the model's performance on the validation set varies with changes in the number of time steps. Due to the excellent memory capabilities inherent in the LSTM neural network used in this model, the output at the current time step is related to the historical data information from any past temporal step. Consequently, as the historical data sequence becomes longer, there is a gradual reduction in the MAPE values of both the training set and validation set, indicative of an enhancement in the model's performance. The optimal performance on the validation set is achieved when the time steps reach equal to 200. However, beyond this point, when the time steps in one sample exceed 200, there is a further increase in the MAPE on the validation set. This could be attributed to the model excessively memorizing details within the training data, leading to suboptimal performance on unseen data. Therefore, this study selects the number of time steps corresponding to the minimum MAPE metric as the length of historical data for each sample, which is 200.

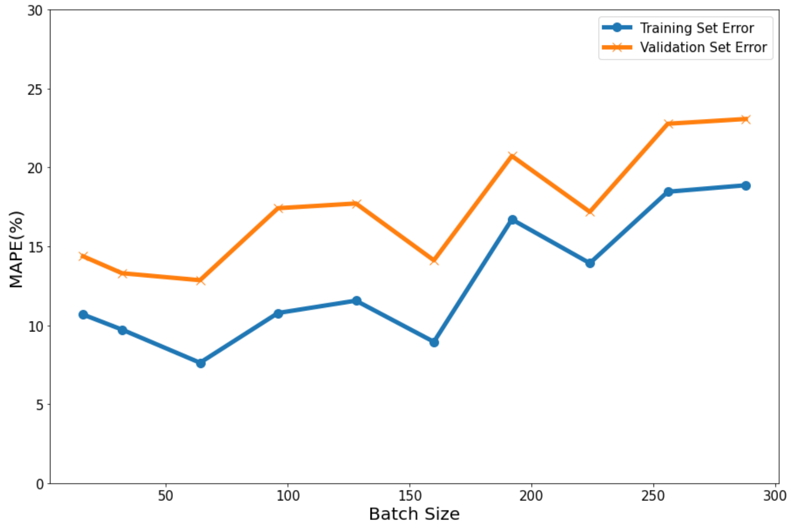


Figure 4.8 The impact of the batch size on model performance

Regarding the batch size, when the value is too small, the model's convergence speed will slow down, and oscillations frequently occur during the training process. Conversely, when the value is too large, it imposes an excessive burden on computational devices, causing the experimental device's memory capacity to be insufficient to meet computational demands. Additionally, the runtime for one epoch, that is, training all samples once and updating model parameters, becomes longer with larger batch sizes. Consequently, more epochs will be needed to achieve the same accuracy, leading to an overall slower model efficiency and increased time consumption. Therefore, within a certain range, increasing the batch size can accelerate convergence speed, reduce oscillations, and enhance model operational efficiency, but using excessively large batch sizes should be avoided.

The batch sizes within the range of [16, 288] are compared. MAPE was used as the evaluation metric to observe the model's predictive performance under different batch size values. As shown in Figure 4.8, the model's predictive results change with the variation in batch size. As the batch size increases from 16 to 64, the MAPE metric shows a decreasing trend, reaching its minimum at a batch size of 64. Subsequently, with further increases in batch size, the MAPE value exhibits an overall upward trend, indicating a significant increase in errors. Although the MAPE value at a batch size of 160 is similar to the value at a batch size of 64, considering the issues related to convergence speed and computational burden mentioned earlier, this study sets the batch size for the LSTM model to 64.

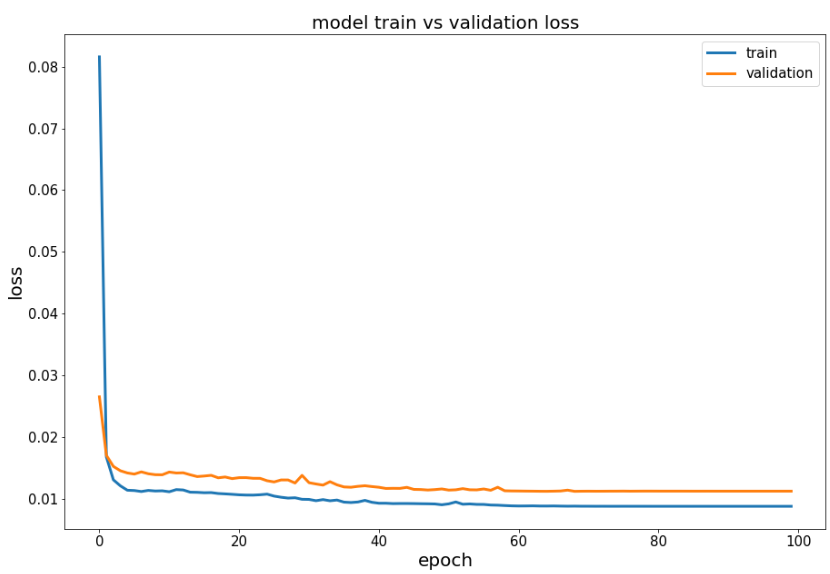


Figure 4.9 Variations of the Training and Validation Loss

With 100 epochs of training, the fluctuations of the training and validation loss for the model with the best setting of hyperparameters are illustrated in Figure 4.9.

The technique of reducing the learning rate is employed to avoid oscillating performance or non-convergence caused by excessively large or small learning rates via motoring the loss function value of the validation set with the following criterion: if the model's performance does not improve after ten epochs (patience), the learning rate will be reduced to 10% of its original value. As observed in Figure 4.9, the validation loss is always higher than the training loss, and when the number of epochs exceeds 70, both the training loss and validation loss remain unchanged, which verifies the convergence of the model.

4.3.2 In-flow rate prediction

From Figure 4.10, it can be seen that the trained LSTM model has an excellent performance in capturing the variations of all vehicles' in-flow rates over time under a MPR of 60%. Overall, the predicted in-flow rates are very close to the real in-flow rates, although some extreme values are not accurately predicted. Additionally, quantitative calculations were performed on the model's prediction results. The MAPE value is 13.115%, indicating that the accuracy of the model's prediction is close to 87%, demonstrating excellent predictive performance.

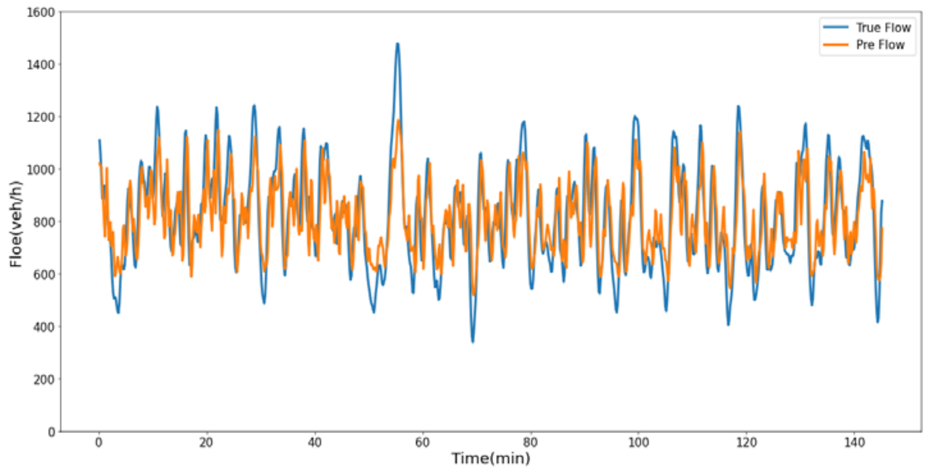


Figure 4.10 Actual and predicted in-flow rate under a MPR of 60%

4.4 Intersection queue profile prediction

In this section, the proposed queue profile prediction model is applied to the virtual intersection simulation environment. The length of the communication range is set as 300 meters. Once a CV enters the communication range, crucial information such as the current location, speed, and time headway of this vehicle will be collected. The proposed algorithm can provide real-time predictions for the future queue profile, which means that each time a prediction is needed, all collected data will be converted into the historical in-flow rate based on only CVs at the location 300 meters away from the stop line, serving as input for the LSTM model. Then, the LSTM model will predict all vehicles' in-flow rate for the next ten time steps, with each time step being 10 seconds. Therefore, the total length of the prediction window is 100 seconds. Once we get the predicted in-flow rate for all vehicles, the queue profile could be calculated based on the propagation of shockwaves.

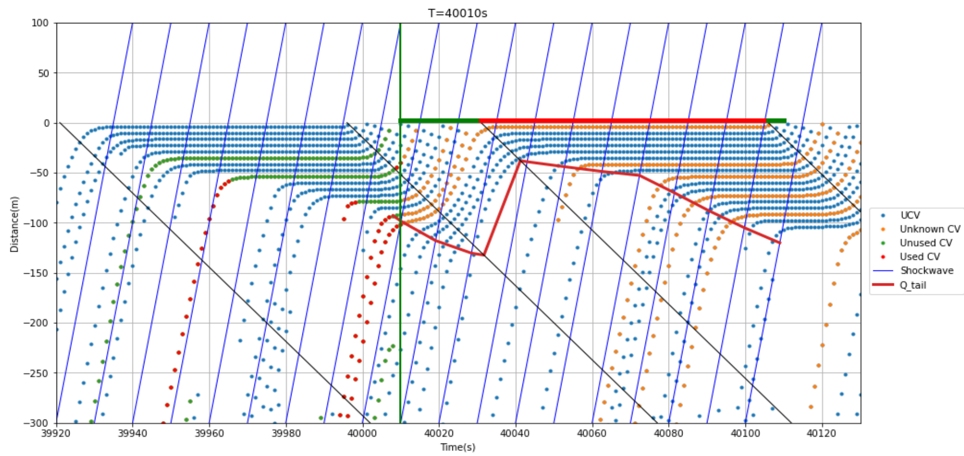


Figure 4.11 An illustrative example of queue profile prediction

Figure 4.11 shows an illustrative example of queue profile prediction for the oversaturated condition in the x-t plane. The green vertical line represents the current time. The points of four different colours represent four types of vehicles' locations at each second (non-CVS, unknown CVs, CVs that are utilized, and CVs that are not utilized). The red line represents our prediction of the location of the queue tail within the prediction time window, and the blue line shows the shockwaves between any two consecutive traffic states of the arrival traffic stream. In general, the path of the queue tail in the x-t plane aligns closely with the envelope line formed by the points representing stopped vehicles, that is, the line connecting each vehicle's location where it joins the queue.

4.4.1 The effect of MPR on total delay prediction

In this subsection, we examine the impact of MPRs of CVs on the total delay prediction under the traffic demand = 1. Figure 4.12 shows the comparison between the predicted total delay and the true total delay in the next 100 seconds of each timestep. It can be clearly seen that the trend of the predicted line almost matches the observed line except at some peaks, which demonstrates that the proposed method can accurately predict how the total delay changes in the future, but there will be some errors in predicting extreme values of the delay. This may be attributed to the inability of the constructed LSTM model to accurately predict the extreme values of the in-flow rates. As shown in Table 4.2, when MPR=20%, the model has the worst performance on the total delay prediction with a MAPE of 24.08%. This is because, at this point, there are not enough CVs approaching the intersection to provide the required information. With the elevation of MPR, there is a gradual reduction in the value of MAPE, signifying a progressive enhancement in the precision of the total delay prediction. When MPR=80%, we can achieve the accuracy of the total delay prediction as high as 86%.

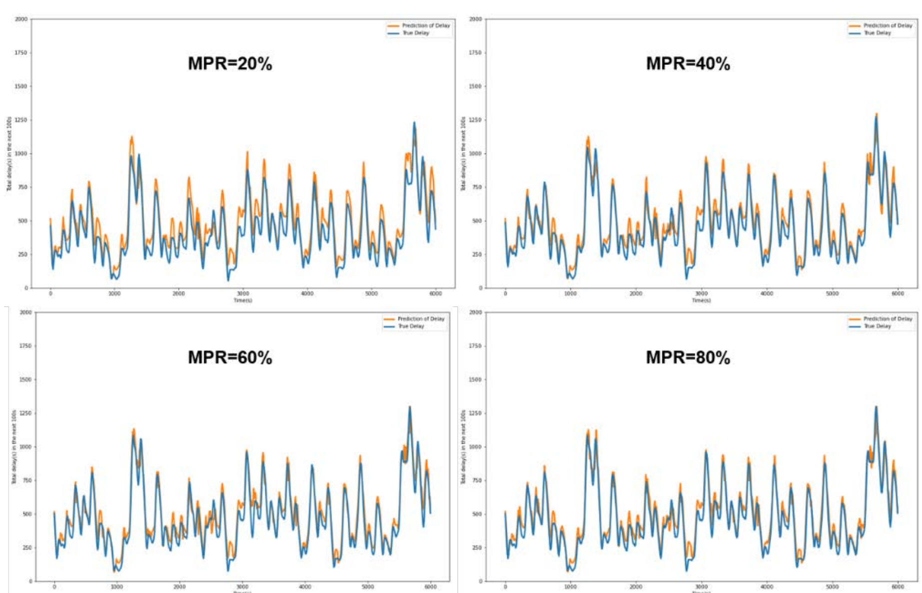


Figure 4.12 The comparison between the predicted and the true total delay under different MPRs

Table 4.2 The MAPE of total delay prediction under different MPRs

MPR	20%	40%	60%	80%
MAPE	24.08%	17.74%	14.35%	13.95%
MAE	78.62s	60.18s	57.52s	49.09s

4.4.2 The effect of demand level on total delay prediction

In this subsection, a sensitivity analysis of different demand levels under a scenario with MPR=60% is conducted to evaluate the accuracy of total delay prediction under different traffic conditions. The traffic demand for each approach shown in Figure 4.1 is used as the base case (demand level=1) and ranges from 0.6 to 1.2. The results summarized in Figure 4.13 and Figure 4.14 illustrate that the model attains the minimum MAPE value, which is 10.69% when the demand level is set to 1.1. At lower demand levels, due to the relatively limited total number of vehicles approaching the intersection, there is a lower count of CVs, leading to higher prediction errors. With the gradual increase in traffic demand, the accuracy of predictions also improves progressively, reaching its peak at saturated or slightly oversaturated demand levels. However, as the traffic demand at the intersection further becomes oversaturated, multiple rolling queues occur, resulting in the formation and propagation of more shockwaves at the intersection. This increases the complexity of computing the queue profile, leading to a decrease in prediction accuracy.

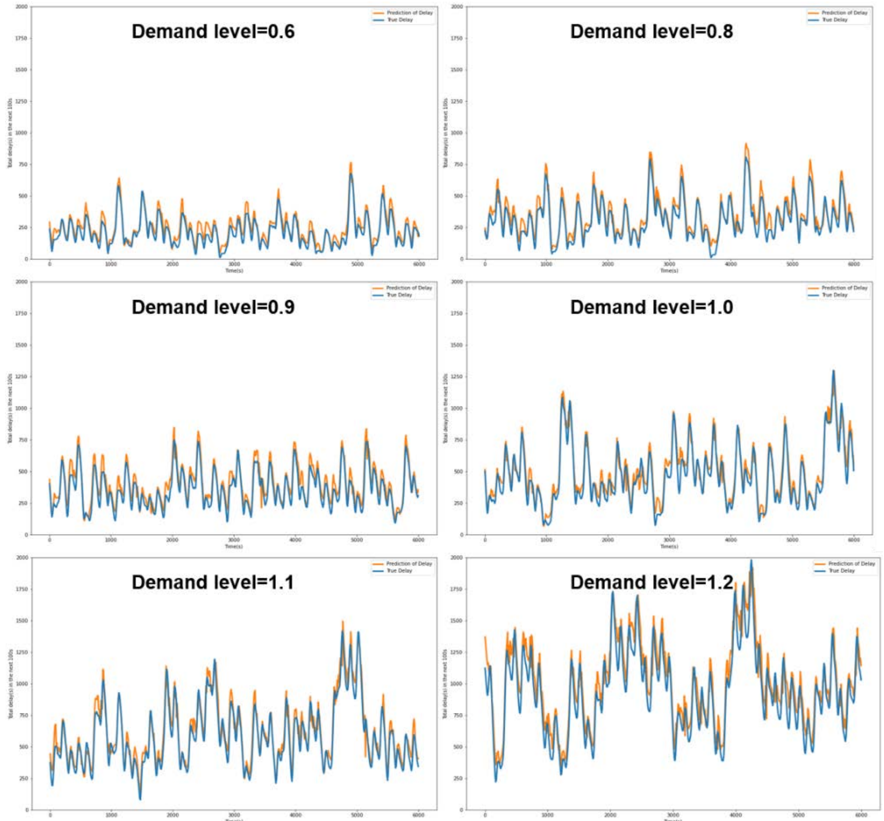


Figure 4.13 The comparison between the predicted and the true total delay under different traffic demands

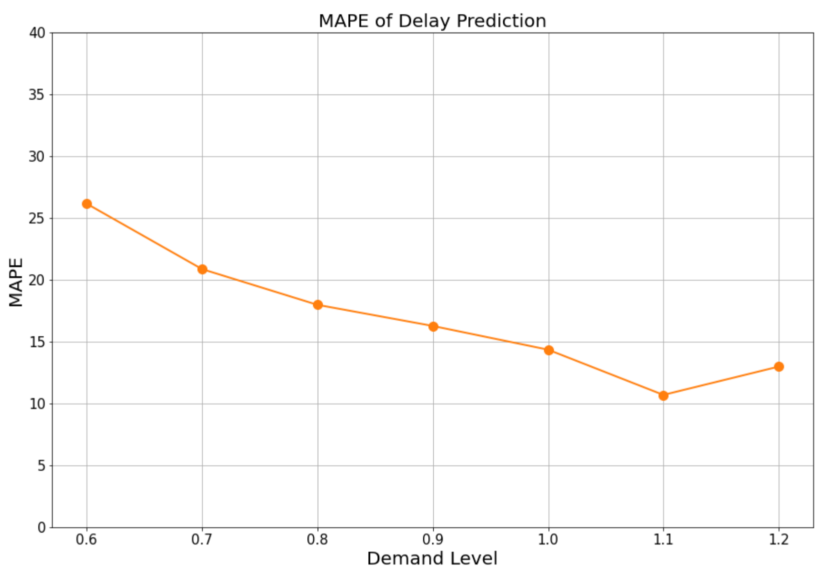


Figure 4.14 The MAPE of total delay prediction under different traffic demand levels

4.5 Performance of Signal Optimization

In this section, the performance of the proposed signal optimization method under different demand levels and MPRs is firstly evaluated on the simulation of the virtual intersection. Then, to test the effectiveness of the proposed method in a more realistic environment, the sensitivity analyses of different demand levels and MPRs are also made on the simulation of a real-world intersection.

4.5.1 The effect of demand level on the performance of signal optimization

This subsection selects a scenario with MPR=60% and analyzes the average travel time delay per vehicle under different control methods corresponding to different traffic demand levels.

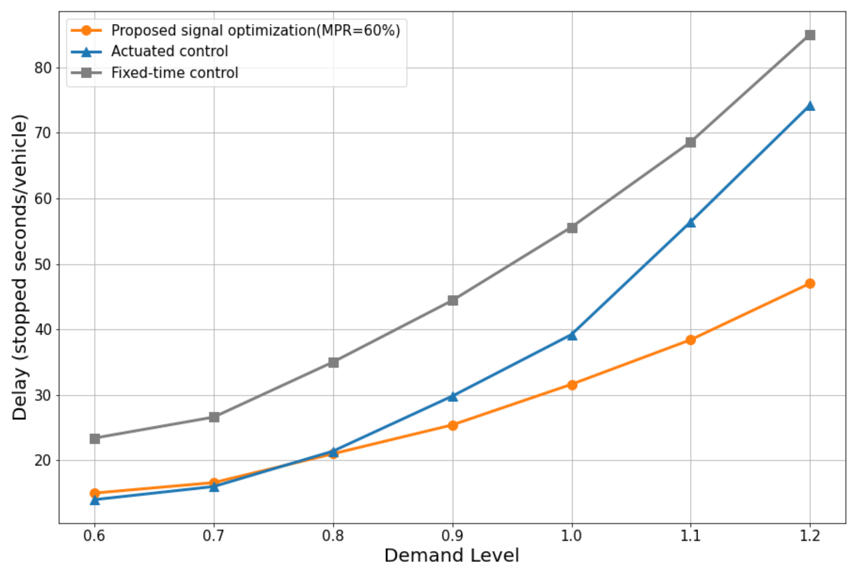


Figure 4.15 The average delay of various control methods under different demand levels

When the MPR reaches 60%, considered a relatively high penetration rate, the proposed signal optimization method demonstrates effective performance in this scenario. It can be observed that different control methods exhibit distinct performances under different traffic demand levels. As the demand level increases from undersaturation to oversaturation, the average delay gradually rises, with a more pronounced rate of increase. Notably, when the demand level exceeds 0.9, there is a significant increase in average delay.

Among these three control methods, fixed-time control exhibits the worst performance, while both actuated control and the proposed optimized control consistently outperform fixed-time control across all demand levels. Under low demand levels, all control methods result in minimal delays, and there is little difference between actuated control and the proposed optimized control. However, as the intersection gradually becomes saturated, the average delay increases and the contribution of CVs to the control method becomes significant. Comparatively, the proposed optimized control exhibits substantial advantages over actuated control, particularly under high demand levels. In summary, while actuated control performs well at low demand levels, its optimization effect weakens as the demand level increases gradually. In contrast, the proposed optimized control method consistently delivers superior performance and mitigates traffic congestion.

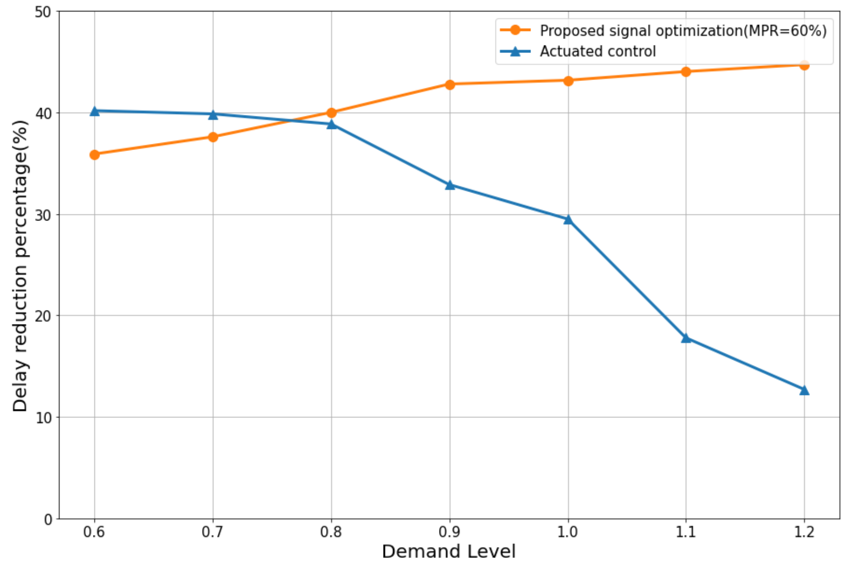


Figure 4.16 The reduction in average delay of various control methods compared with fixed-time control

Figure 4.15 shows that both the actuated control and the proposed optimized control outperform the fixed-time control. Figure 4.16 describes the reduction in average delay for various control methods compared with fixed-time control. When the demand level is below 0.7, actuated control has a larger reduction of average delay. At this point, despite the high penetration rate, the number of vehicles is still low, resulting in uneven vehicle distribution and inaccurate in-flow rate prediction. When the demand level exceeds 0.7, the advantages of the proposed optimized control gradually become significant, while the effectiveness of actuated control weakens. This is because as the traffic flow rate increases, the number of queued vehicles gradually rises. Consequently, once the location of the queue tail exceeds the loop detector, the start-stop motion of vehicles, influenced by the queuing waves, would interfere with actuated control. In contrast, the effectiveness of the proposed optimized control remains relatively stable, with a reduction in average delay ranging from 30% to 50%.

4.5.2 The effect of MPR on the performance of signal optimization

Figures 4.17 indicates the percentage reduction in average delay achieved by the proposed signal optimization model compared with actuated control under different MPRs in three scenarios with different demand levels.

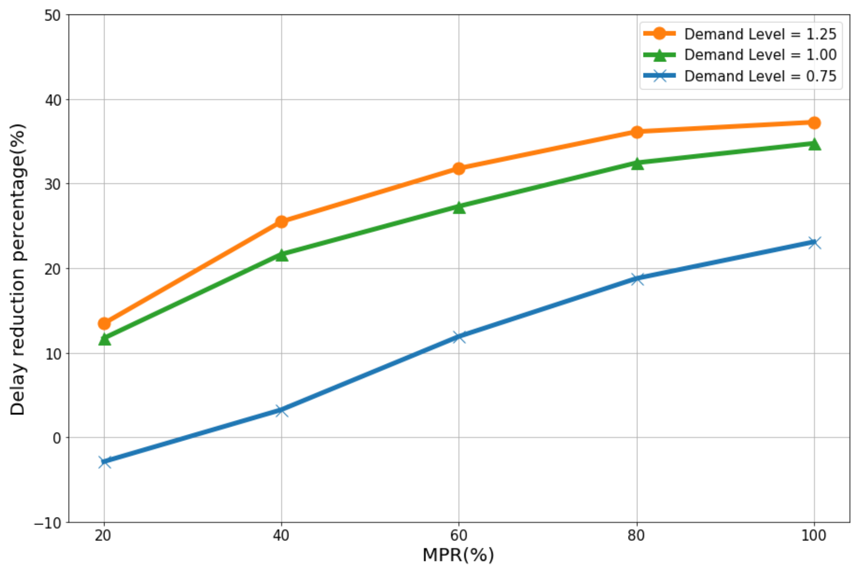


Figure 4.17 The delay reduction compared with the actuated control under different MPRs with three different demand levels

(a) The virtual intersection with a demand level of 1.25

In the oversaturated condition with a traffic demand of 1.25, the traffic volume at the intersection has exceeded its maximum capacity, resulting in higher delays. However, due to the large traffic volume and the existence of long queues, there is an obvious delay reduction compared to actuated signal control. The optimization effect at the lowest MPR=20% is also around 13%. As the MPR increases, the delay further decreases, but the magnitude of this reduction gradually diminishes. After the MPR reaches 80%, the proposed signal optimization model has essentially maximized its effectiveness. Further increasing the MPR does not lead to significant additional improvements. At an MPR =100%, where all vehicles are CVs, the optimization effect is at its best, with a percentage reduction in delay reaching 37.25%.

(b) The virtual intersection with a demand level of 1

When the demand level is set as 1, the intersection is in the saturated condition, and the proposed signal optimization model still has an ideal performance. At the minimum MPR of 20%, the percentage reduction in delay has already exceeded 10%, and it can reach a maximum of 36% at MPR=100%. Similar to the situation at a demand level of 1.25, there is no significant increase in the optimization effect once the MPR reaches 80%.

(c) The virtual intersection with a demand level of 0.75

When the demand level is 0.75, the intersection is in the undersaturated condition. At this point, the in-flow rate is relatively low. Thus, even under fixed-time signal control, queues formed during red time can be easily released during green time. Consequently, the benefits derived from the proposed signal optimization model, which aims to minimize total delay, are not substantial. With lower MPR, the delay introduced by the proposed optimization method is slightly higher than that of actuated signal control. This may be attributed to the lack of sufficient CV information, leading to less precise predictions of the in-flow rate. As MPR gradually increases, there is an overall decreasing trend in average delay, reaching a reduction of 20.21% compared to actuated control when MPR is 100%. However, regardless of the MPR value, the proposed signal optimization model consistently outperforms fixed-time signal control.

4.5.3 Performance of signal optimization on FM 528

This subsection analyzes the effect of different demand levels and MPRs on the performance of the proposed signal optimization on a real-world intersection of FM 528.

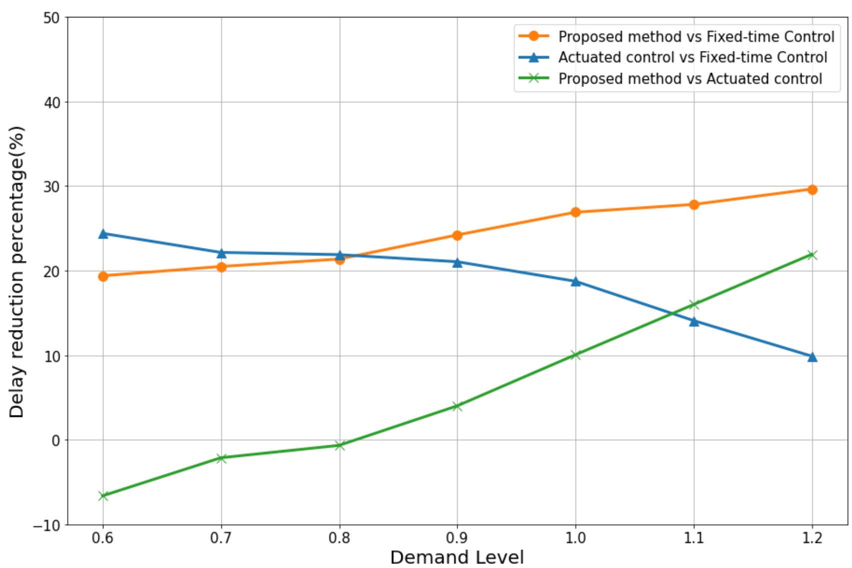


Figure 4.18 The delay reduction among various control methods under different demand levels at the real-world intersection

Figure 4.18 illustrates the delay reduction percentage among various control methods under different demand levels with MPR=20%. Similar to the virtual intersection in Section 4.5.1, both actuated control and the proposed signal optimization method outperform fixed-time control. With a higher demand level, the reduction in the average delay of the proposed signal optimization method is larger, while the optimization effect of actuated signal control gradually diminishes. Different from the virtual intersection, this sensitivity analysis is conducted on a real-world intersection in a scenario with low MPR. The variable in-flow rate and low MPR at the real-world intersection may lead to a decrease in the accuracy of delay predictions, thereby affecting the performance of the proposed signal optimization method. Therefore, the percentage reduction in average delay achieved by the proposed method compared with fixed-time control only ranges from 20% to 30%.

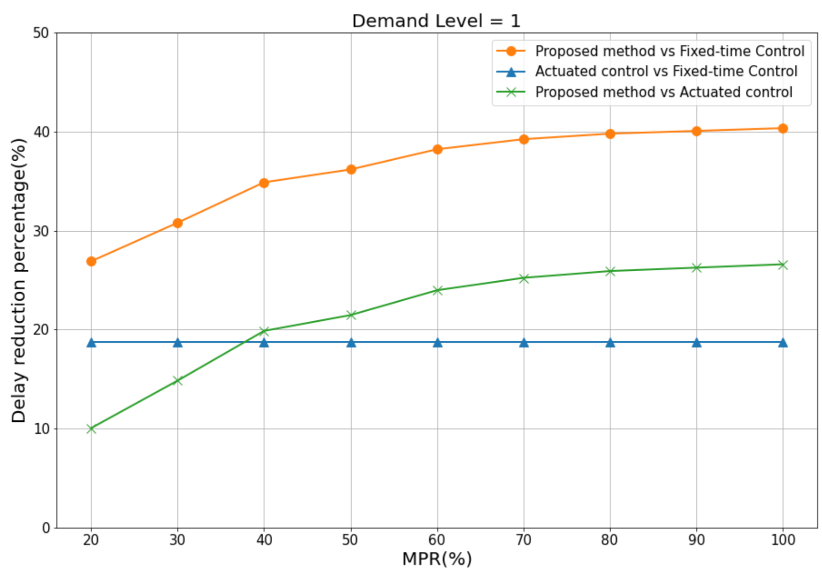


Figure 4.19 The delay reduction among various control methods under different MPRs at the real-world intersection

Figure 4.19 depicts a sensitivity analysis illustrating the impact of MPRs on the percentage reduction in average delay among various control methods, with a demand level set to 1. Similar to the virtual intersection, with higher MPRs, the reduction in average delay at the intersection is larger, and there is no significant increase in the optimization effect once the MPR reaches 80%. Although the performance at the real-world intersection is slightly inferior to that at the virtual intersection, the proposed signal optimization method still effectively reduces delay. Under different MPRs, the proposed method always outperforms the fixed-time control and the actuated control, and the percentage reduction in delay achieved by the proposed method compared with the actuated control ranges from 10% to 26%.

Conclusions

In this thesis, we develop a hybrid model to predict intersection queue profiles and delays under a mixed CV environment. First, a deep learning-based in-flow rate predictive model is proposed using the LSTM model. An encoder-decoder LSTM model is employed to more accurately approximate the real underlying mapping—from the historical in-flow rate estimated solely based on CV information to the in-flow rate of all vehicles in the future. Second, based on the predicted future in-flow rates, the LWR model is applied to analyze shockwave propagations at the single signalized intersection to predict the queue profile for both undersaturated and oversaturated conditions. Moreover, a signal optimization algorithm based on DP is developed with the predicted queue profiles, which can allocate the optimal phase sequence and duration to minimize vehicle delays within the optimization time window. The primary research conclusions and key innovations are summarized, and potential directions for future research are outlined in this chapter. To test the effectiveness of the proposed models, two case studies, a virtual four-legged intersection and a real-world intersection, are simulated by INTEGRATION microscopic traffic simulator, respectively. The proposed LSTM model exhibits effective performance in predicting the in-flow rate for north through movements at the virtual intersection, achieving an accuracy of 87%. Building on this success, sensitivity analyses are conducted to evaluate the influence of different MPRs and demand levels on the performance of both the queue profile prediction model and the signal optimization model at the virtual intersection. In terms of queue profile prediction performance, the analysis reveals that higher MPRs correspond to smaller errors in total delay prediction, with optimal performance observed under saturated or slightly oversaturated demand levels. Specifically, under conditions of relatively high MPR (60%) and slightly oversaturated traffic (demand level = 1.1), the MAPE for delay prediction is impressively low at 10.69%.

Moreover, the outcomes of the sensitivity analysis, aimed at evaluating the advantages of the proposed signal optimization algorithm in mitigating average vehicle delays, reveal that the reduction in vehicle delay is more pronounced with higher MPRs of CVs at the virtual intersection. The algorithm can achieve the optimal control effect when the MPRs reach 80%. At the virtual intersection with demand level=1, the proposed algorithm can reduce vehicle delays by up to 36% compared with the actuated control. In addition, the sensitivity analysis of the demand level at the virtual intersection shows the proposed algorithm has a relatively stable performance under various traffic conditions, while the effectiveness of the actuated control weakens at higher demand levels due to the location of the queue tail exceeding the loop detector. Finally, the case study of a real-world intersection on FM 528 proves that the proposed signal optimization algorithm can still effectively reduce vehicle delay in a more realistic traffic environment. At low MPR (20%), the percentage reduction in average delay achieved by the proposed method compared with fixed-time control ranges from 20% to 30% under various demand levels. Moreover, the proposed method also has a significant advantage compared to the actuated control at the relatively higher MPRs.

With the continuous development of CV technology, the model and signal optimization method proposed in this thesis holds significant reference value for the deployment of advanced traffic control in the real world and related research. In practical application, the proposed methods can be universally implemented in any single signalized intersection. The queue profile prediction model is applied to each lane of every approach at the intersection based on the collected CVs' historical in-flow rates. The signal optimization algorithm can generate any phase sequence and duration, minimizing total vehicle delay at the intersection by considering predicted queue profiles for each lane. Consequently, the proposed queue profile prediction model and signal optimization algorithm can be seamlessly transferred to any single signalized intersection, irrespective of its geometry and number of signal phases. While applying the system to another signalized intersection requires recalibrating traffic flow parameters and training the LSTM model, the practical implementation only needs to perform these tasks at a single intersection and then extend them to other intersections with similar traffic environments. For instance, in the case of multiple intersections along an arterial road displaying analogous in-flow rate patterns and traffic flow parameters, calibrating parameters and training the LSTM model at one intersection is adequate for deploying the proposed method at all intersections on that arterial road. While this may result in a slight reduction in prediction accuracy and optimization effectiveness, it ensures the portability and applicability of the proposed methods.

However, the work in this thesis still has several limitations:

(a) The CVs are able to communicate with each other to form a vehicle platoon with smaller space headways and time headways, thus, several consecutive CVs in the queue could be released at a larger saturation flow rate. This different driving behaviours of CVs and non-CVs have not been considered in this thesis.

(b) In this research, the MPR of CVs is a crucial factor influencing both prediction accuracy and optimization effectiveness. Although the sensitivity analyses of MPRs have been conducted regarding the accuracy of total delay prediction and the performance of signal optimization, this research has yet to delve further into the direct impact of MPRs on the accuracy of in-flow rates prediction. This aspect should be subjected to more in-depth analysis in future research.

(c) Although this research has assessed the performance of the proposed signal optimization method at a relatively low MPR of 20%, further investigation is warranted to determine the minimum MPR at which the method can still effectively reduce delays.

In addition, traffic control at intersections in the intelligent vehicle-road network environment still faces tremendous challenges due to the complexity of its control and the diversity of scenarios. Future research will build upon the foundation laid in this paper and focus on the following tasks:

(a) Build a multivariate LSTM model. This thesis has currently developed a univariate LSTM model to predict the future in-flow rate of all vehicles at the single intersection, which can be extended to a multivariate LSTM model. By introducing additional variables such as the speed of CVs and the out-flow rate at neighbouring intersections, the model can learn richer patterns and trends, thereby improving its predictive performance for future scenarios and adapting to a broader range of application scenarios."

(b) Consider the coordination of multiple intersections. This thesis only analyzes the queue profile prediction and signal optimization method for the single intersection under the CV environment. In subsequent research, it is advisable to consider extending the analysis to CV-based signal coordination on traffic corridors and even to network-wide CV-based signal control. This expansion aims to further enhance the traffic efficiency on a main traffic corridor or the large-area road network.

(c) Consider other types of vehicles. In this thesis, we make the assumption that all vehicles on the road are passenger cars. However, in the intelligent vehicle-road network environment, it is worth noting that information regarding various vehicle types can be obtained and incorporated into the analysis. Further research could consider integrating the coordination and priority of special vehicle types, such as buses, large trucks, and emergency vehicles, into the current signal optimization algorithms.

(d) Consider additional traffic participants. Within the intelligent human-vehicle-road tripartite communication network, the traffic control system could achieve a globally optimal solution by coordinating the control of all traffic participants. Therefore, future research can further consider the demands of non-motorized vehicles and pedestrians for intersection spatiotemporal resources, integrating the traffic safety concerns and travel efficiency of different traffic participants into signal optimization objectives.

(e) Consider vehicle-to-vehicle communication. This paper primarily models and optimizes the traffic signal control system based on the data collected in the CV environment. By exchanging speed and position information with each other, CVs can pass the intersection with a smaller space headway between each other, thereby enhancing the efficiency of spatiotemporal resource utilization at the intersection. Thus, future research could further explore the impact of vehicle-to-vehicle communication on the effectiveness of signal optimization.

Citations and References

[1] Dachis B. Cars, congestion and costs: A new approach to evaluating government infrastructure investment. CD Howe Institute Commentary. 2013;385.

[2] Schrank D, Eisele B, Lomax T, et al. 2015 urban mobility scorecard. 2015.

[3] Webster F V. Traffic signal setting. Road Research Technical Paper. 1958;39.

[4] Gartner N H, Little J D C, Gabbay H. Optimization of traffic signal settings by mixed-integer linear programming: Part I: The network coordination problem. Transportation Science. 1975;9(4):321-343.

[5] Gartner N H. Optimal traffic assignment with elastic demands: a review part II. Algorithmic approaches[J]. Transportation Science. 1980;14(2):192-208.

[6] Feng Y, Head K L, Khoshmagham S, et al. A real-time adaptive signal control in a connected vehicle environment. Transportation Research Part C: Emerging Technologies, 2015;55:460-473.

[7] Yao Z, Shen L, Liu R, et al. A dynamic predictive traffic signal control framework in a cross-sectional vehicle infrastructure integration environment. IEEE Transactions on Intelligent Transportation Systems. 2019;21(4):1455-1466.

[8] Feng Y. Intelligent traffic control in a connected vehicle environment. The University of Arizona, 2015.

[9] Zheng J, Liu H X. Estimating traffic volumes for signalized intersections using connected vehicle data. Transportation Research Part C: Emerging Technologies. 2017;79:347-362.

[10] Zheng F, Jabari S E, Liu H X, et al. Traffic state estimation using stochastic Lagrangian dynamics. Transportation Research Part B: Methodological, 2018;115:143-165.

[11] Rahman R, Hasan S. Real-time signal queue length prediction using long short-term memory neural network. Neural Computing and Applications. 2021;33:3311-3324.

[12] Yuan Y, Zhang Z, Yang X T, et al. Macroscopic traffic flow modeling with physics regularized Gaussian process: A new insight into machine learning applications in transportation. Transportation Research Part B: Methodological. 2021;146:88-110.

[13] Zhang Z, Yang X T, Yang H. A review of hybrid physics-based machine learning approaches in traffic state estimation. Intelligent Transportation Infrastructure, 2023;2.

[14] Zhang Z, Yuan Y, Yang X. A hybrid machine learning approach for freeway traffic speed estimation. Transportation research record. 2020;2674(10):68-78.

[15] Hodge V J, Krishnan R, Austin J, et al. Short-term prediction of traffic flow using a binary neural network. Neural Computing and Applications. 2014;25:1639-1655.

[16] Yang B, Sun S, Li J, et al. Traffic flow prediction using LSTM with feature enhancement. Neurocomputing. 2019;332:320-327.

[17] Wang J, Chen R, He Z. Traffic speed prediction for urban transportation network: A path based deep learning approach. Transportation Research Part C: Emerging Technologies. 2019;100:372-385.

[18] Polson N G, Sokolov V O. Deep learning for short-term traffic flow prediction[J]. Transportation Research Part C: Emerging Technologies. 2017;79:1-17.
[19] He Z, Zheng L, Chen P, et al. Mapping to cells: A simple method to extract traffic dynamics from probe vehicle data. Computer-Aided Civil and Infrastructure Engineering. 2017;32(3):252-267.
[20] Jiang Z, Chen X M, Ouyang Y. Traffic state and emission estimation for urban expressways based on heterogeneous data. Transportation Research Part D: Transport and Environment. 2017;53:440-453.
[21] Ran B, Song L, Zhang J, et al. Using Tensor Completion Method to Achieving Better Coverage of Traffic State Estimation from Sparse Floating Car Data. PLoS ONE. 2016;11(7).
[22] Wan N, Vahidi A, Luckow A. Reconstructing maximum likelihood trajectory of probe vehicles between sparse updates. Transportation Research Part C: Emerging Technologies. 2016;65:16-30.
[23] Xie X, van Lint H, Verbraeck A. A generic data assimilation framework for vehicle trajectory reconstruction on signalized urban arterials using particle filters. Transportation research part C: emerging technologies. 2018;92: 364-391.
[24] Comert G, Cetin M. Queue length estimation from probe vehicle location and the impacts of sample size. European Journal of Operational Research. 2009;197(1):196-202.
[25] Comert G, Cetin M. Analytical evaluation of the error in queue length estimation at traffic signals from probe vehicle data. IEEE Transactions on Intelligent Transportation Systems. 2011;12(2):563-573.
[26] Viti F, Van Zuylen H J. The dynamics and the uncertainty of queues at fixed and actuated controls: A probabilistic approach. Journal of Intelligent Transportation Systems. 2009;13(1):39-51.
[27] Comert G. Simple analytical models for estimating the queue lengths from probe vehicles at traffic signals. Transportation Research Part B: Methodological. 2013;55:59-74.
[28] Comert G. Queue length estimation from probe vehicles at isolated intersections: Estimators for primary parameters. European Journal of Operational Research. 2016;252(2):502-521.
[29] Hao P, Ban X J, Guo D, et al. Cycle-by-cycle intersection queue length distribution estimation using sample travel times. Transportation research part B: methodological. 2014;68:185-204.
[30] Hao P, Sun Z, Ban X J, et al. Vehicle index estimation for signalized intersections using sample travel times. Procedia-Social and Behavioral Sciences. 2013;80:473-490.
[31] Cheng Y, Qin X, Jin J, et al. Cycle-by-cycle queue length estimation for signalized intersections using sampled trajectory data. Transportation research record. 2011;2257(1):87-94.
[32] Cetin M. Estimating queue dynamics at signalized intersections from probe vehicle data: Methodology based on kinematic wave model. Transportation research record. 2012;2315(1):164-172.

[33] Ban X J, Hao P, Sun Z. Real time queue length estimation for signalized intersections using travel times from mobile sensors. Transportation Research Part C: Emerging Technologies. 2011;19(6):1133-1156.

[34] Ramezani M, Geroliminis N. Queue profile estimation in congested urban networks with probe data. Computer-Aided Civil and Infrastructure Engineering. 2015;30(6):414-432.

[35] Hao P, Ban X, Whon Yu J. Kinematic equation-based vehicle queue location estimation method for signalized intersections using mobile sensor data. Journal of Intelligent Transportation Systems. 2015;19(3):256-272.

[36] Wang Z, Zhu L, Ran B, et al. Queue profile estimation at a signalized intersection by exploiting the spatiotemporal propagation of shockwaves. Transportation research part B: methodological. 2020;141:59-71.

[37] Yin J, Sun J, Tang K. A Kalman filter-based queue length estimation method with low-penetration mobile sensor data at signalized intersections. Transportation Research Record. 2018;2672(45):253-264.

[38] Li J Q, Zhou K, Shladover S E, et al. Estimating queue length under connected vehicle technology: Using probe vehicle, loop detector, and fused data. Transportation research record. 2013;2356(1):17-22.

[39] Cai Q, Wang Z, Zheng L, et al. Shock wave approach for estimating queue length at signalized intersections by fusing data from point and mobile sensors. Transportation Research Record. 2014;2422(1):79-87.

[40] Wang Z, Cai Q, Wu B, et al. Shockwave-based queue estimation approach for undersaturated and oversaturated signalized intersections using multi-source detection data. Journal of Intelligent Transportation Systems. 2017;21(3):167-178.

[41] Shahrbabaki M R, Safavi A A, Papageorgiou M, et al. A data fusion approach for real-time traffic state estimation in urban signalized links. Transportation research part C: emerging technologies. 2018;92:525-548.

[42] Tan C, Liu L, Wu H, et al. Fuzing license plate recognition data and vehicle trajectory data for lane-based queue length estimation at signalized intersections. Journal of Intelligent Transportation Systems. 2020;24(5):449-466.

[43] Comert G, Cetin M. Queue length estimation from connected vehicles with range measurement sensors at traffic signals. Applied Mathematical Modelling. 2021;99:418-434.

[44] Cheek M T. Improvements to a queue and delay estimation algorithm utilized in video imaging vehicle detection systems. Texas A&M University, 2007.

[45] Shirazi M S, Morris B. Vision-based vehicle queue analysis at junctions. 2015 12th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). IEEE. 2015;1-6.

[46] Jiang T, Cai M, Zhang Y, et al. Fast video-based queue length detection approach for self-organising traffic control. IET Intelligent Transport Systems. 2019;13(4):670-676.

[47] Al Okaishi W, ZAARANE A, SLIMANI I, et al. Vehicular queue length measurement based on edge detection and vehicle feature extraction. Journal of Theoretical and Applied Information Technology. 2019;97(5):1595-1603.

[48] Guo Q, Li L, Ban X J. Urban traffic signal control with connected and automated vehicles: A survey. Transportation research part C: emerging technologies. 2019;101:313-334.
[49] Yin Y, Li M, Skabardonis A. Offline offset refiner for coordinated actuated signal control systems. Journal of transportation engineering. 2007;133(7):423-432.
[50] Yun I, Park B. Stochastic optimization for coordinated actuated traffic signal systems. Journal of Transportation Engineering. 2012;138(7):819-829.
[51] Gradinescu V, Gorgorin C, Diaconescu R, et al. Adaptive traffic lights using car-to-car communication. 2007 IEEE 65th vehicular technology conference-VTC2007-Spring. IEEE. 2007;21-25.
[52] Liang X, Guler S I, Gayah V V. Signal timing optimization with connected vehicle technology: Platooning to improve computational efficiency. Transportation Research Record. 2018;2672(18):81-92.
[53] Lioris J, Pedarsani R, Tascikaraoglu F Y, et al. Platoons of connected vehicles can double throughput in urban roads. Transportation Research Part C: Emerging Technologies. 2017;77:292-305.
[54] Pandit K, Ghosal D, Zhang H M, et al. Adaptive traffic signal control with vehicular ad hoc networks. IEEE Transactions on Vehicular Technology. 2013;62(4):1459-1471.
[55] He Q, Head K L, Ding J. PAMSCOD: Platoon-based arterial multi-modal signal control with online data. Transportation Research Part C: Emerging Technologies. 2012;20(1):164-184.
[56] Chen S, Sun D J. An improved adaptive signal control method for isolated signalized intersection based on dynamic programming. IEEE Intelligent Transportation Systems Magazine. 2016;8(4):4-14.
[57] Feng Y, Head K L, Khoshmagham S, et al. A real-time adaptive signal control in a connected vehicle environment. Transportation Research Part C: Emerging Technologies. 2015;55:460-473.
[58] Sen S, Head K L. Controlled optimization of phases at an intersection. Transportation science. 1997;31(1):5-17.
[59] Li W, Ban X J. Traffic signal timing optimization in connected vehicles environment. 2017 IEEE Intelligent Vehicles Symposium (IV). IEEE. 2017;1330-1335.
[60] Priemer C, Friedrich B. A decentralized adaptive traffic signal control using V2I communication data. 2009 12th international ieee conference on intelligent transportation systems. IEEE. 2009;1-6.
[61] Beak B, Head K L, Feng Y. Adaptive coordination based on connected vehicle technology. Transportation Research Record. 2017;2619(1):1-12.
[62] Al Islam S M A B, Hajbabaie A. Distributed coordinated signal timing optimization in connected transportation networks. Transportation Research Part C: Emerging Technologies. 2017;80:272-285.
[63] Lipton Z C, Berkowitz J, Elkan C. A critical review of recurrent neural networks for sequence learning. arXiv preprint arXiv:1506.00019. 2015.

[64] Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation. 1997;9(8):1735-1780.
[65] Gers F A, Schmidhuber J, Cummins F. Learning to forget: Continual prediction with LSTM. Neural computation. 2000;12(10):2451-2471.
[66] Lighthill M J, Whitham G B. On kinematic waves II. A theory of traffic flow on long crowded roads. Proceedings of the royal society of london. series a. mathematical and physical sciences. 1955;229(1178):317-345.
[67] Stephanopoulos G, Michalopoulos P G, Stephanopoulos G. Modelling and analysis of traffic queue dynamics at signalized intersections. Transportation Research Part A: General. 1979, 13(5): 295-307.
[68] Liu H X, Wu X, Ma W, et al. Real-time queue length estimation for congested signalized intersections. Transportation research part C: emerging technologies. 2009;17(4):412-427.
[69] Sen S, Head K L. Controlled optimization of phases at an intersection. Transportation science. 1997;31(1):5-17.
[70] Van Aerde M, Rakha H. INTEGRATION© Release 2.30 for Windows: User's Guide–Volume I: Fundamental Model Features. M. Van Aerde & Assoc., Ltd., Blacksburg, 2007.
[71] Van Aerde M. Single regime speed-flow-density relationship for congested and uncongested highways. 74th Annual Meeting of the Transportation Research Board, Washington, DC. 1995;6.