A Sustainable and Equitable Transportation System Design Under Connected Vehicle and Big Data Driven Environment

By

Farzana Rahman Chowdhury

Dissertation
Submitted to the Faculty of
The University of Texas at Arlington
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Transportation Engineering
in the Department of Civil and Environmental Engineering

The University of Texas at Arlington

August 2022

A Sustainable and Equitable Transportation System Design Under Connected Vehicle and Big Data Driven Environment

By
APPROVAL PAGE
Farzana Rahman Chowdhury

Approved:

_____
Dr. Pengfei (Taylor) Li
(Committee Chair)

_____
Dr. Stephen P. Mattingly
(Committee Member)

_____
Dr. Kyung Kate Hyun
(Committee Member)

_____
Dr. Xianfeng Terry Yang
(Committee Member)

Name: Farzana Rahman Chowdhury
Date of Degree: June 28, 2022

Institution: The University of Texas at Arlington

Major Field: Transportation Engineering

Major Professor: Pengfei (Taylor) Li

Title of Study: A Sustainable and Equitable Transportation System Design Under Connected Vehicle and Big Data Driven Environment

Candidate for Degree of Doctor of Philosophy

ABSTRACT

As the population increases day by day, several cities are having trouble dealing with the increasing demand for a sustainable and equitable transportation system. At the same time, application of newer technology is also increasing . By definition, a sustainable transportation system indicates a method that includes the needs of each individual road user and mode of transportation. A sustainable transportation system also supports the compatibility of the system with the change in demand, regional development, and available resources. In a sustainable and equitable transportation system, integrating the virtual world and the physical world is crucial since it makes the overall system dynamic. This integration includes data monitoring and analysis from the virtual and the real world, developing new strategies, and applying simulation to evaluate the proposed methods. The goal of the sustainable and equitable transportation system is to generate a sustainable system that benefits all road users, i.e., both vehicles and pedestrians. Therefore, transportation sustainability and equitability can be divided into three main focuses (i) sustainable and equitable intersection design in mixed traffic conditions (ii) sustainable and equitable system for pedestrian detection ,and (iii) network performance analysis within the big data-driven environment. In this dissertation, a sustainable and equitable transportation system is presented in a connected vehicle and big data driven environment. The initial objective develops a congestion-aware heterogeneous

connected automated vehicle cooperative scheduling problem at intersections with the objective to present a method that can provide a systematic approach to the green request accommodations with different priorities at intersections. The Mixed Integer Linear Programming (MILP) formulation is developed in the context of discrete space-time and phase-time networks whose choice variables are space-time with respect to individual vehicles and phase-time. An efficient search algorithm based on the "Arrival and Departure Curves (A-D curves)" for real-time applications is also built. Three experiments are conducted to validate the proposed MILP formulation and search algorithm. The simulation-based performance evaluation for the congestion-aware Connected and Automated Vehicles (CAV) scheduling reveal promising results for real-world applications in the future.

Later, a novel dynamic flash yellow arrow (D-FYA) solution using the LIDAR-based tracking technique is developed. It can address the safety concerns in the FYA while recovering the permissive left-turn capacity after the concurrent pedestrians are cleared. Depending on the pedestrian volumes, the corresponding FYA with each cycle will either start as scheduled, be postponed, or be canceled within each cycle. The proposed D-FYA was deployed at an intersection next to the campus of the University of Texas at Arlington, and its real-time D-FYA decisions in the field were verified for over 100 traffic signal cycles through simultaneous observation in the field. The proposed D-FYA solution is further evaluated within an "ATC-cabinet-in-the-loop" traffic signal simulation platform to compare its mobility performance with another two permissive left-turn strategies: (I) "Protected + Permissive left turn (PPLT)" and (II) "PPLT with Minus-pedestrian-phase". The experiment results reveal the D-FYA is accurate and adaptive compared to the other two permissive left-turn strategies.

Furthermore, the dissertation also presents an innovative framework for travel demand forecasting. The current practice of travel demand forecasting in DFW is the classic "four-step" method based on household surveys and traffic counts. Like the connected vehicle trajectories,

the emerging new traffic data bring both opportunities and challenges. The novel data sets reveal much more information about the traveler than before and pave the way for enhanced accurate and high-fidelity travel demand forecasts. On the other hand, the traditional travel demand forecast cannot take advantage of the total power of such data sets. The inconsistency of various data sets and heterogeneous data quality impact fusing the emerging traffic data with the traditional ones. This task explores the innovative framework of travel demand forecasting based on connected vehicle data using state-of-the-art big data analytics and high-performance computing to address these issues.

DEDICATION

I dedicate this research work to

My Parents

Waysur Rahman Chowdhury

Nazma Rahman Chowdhury

And

My Beloved Husband

Jobaidul Alam Boni

ACKNOWLEDGEMENTS

Since I started my engineering studies, doing a Ph.D. has been a dream for me, and it would not have been possible without consistent reinforcement from many people. First, I would like to express my sincere gratitude to my supervisor and mentor, Dr. Pengfei (Taylor) Li, for his continuous guidance while pursuing my Ph.D. degree. He always kept motivating me with his wise advice whenever I lost hope and had difficulties in my studies and personal affairs. He provided me with this unique opportunity to excel in Transportation engineering by sharing his intelligence, skills, and techniques throughout the journey. The time I spent working with Dr.Li was one of the most important and memorable times of my entire life. He was always open to hearing about new ideas and provided all available resources to execute the work. To date, he is the go-to person for me whenever I face any difficulties.

I would also like to thank Dr. Stephen Mattingly for allowing me to work by his side for two different projects where I learned more about applying some specific tools, which has added some value to my overall progress. I also wish to acknowledge the NCTCOG modeling group members. Special appreciation to Dr. Hong Zheng, Francisco Torres and Dr. Gopindra Nair for providing me with valuable suggestions and insides that eventually expedited my research.

Infinite gratitude to my family – my parents, in-laws, and siblings for always being my powerhouse and being there with me no matter what the situation is. I believe I didn't let them down. Special thanks to my husband, Jobaidul Alam Boni, for encouraging me to be focused during our difficult times. Without his patience and understanding, I wouldn't be able to reach this point.

In addition, my deepest appreciation to a few dearest friends and lab mates, including but not limited to Farhina Haque, Slade Wang, and Swastik Khadka, for their endless encouragement and compassionate ear that made my Ph.D. journey pleasant.

Last but not the least I attribute all my achievements to the Almighty Allah.

Farzana Rahman Chowdhury

The University of Texas at Arlington, August,2022

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 Overview

With the increase in complexity in the urban transportation system, the need for a more diverse and sustainable alternative for network performance analysis has increased. As the population increases day by day, several cities are having trouble dealing with the increasing demand for a sustainable and equitable transportation system. At the same time, application of newer technology is also increasing .By definition, a sustainable transportation system indicates a method that includes the needs of each individual road user and mode of transportation. A sustainable transportation system also supports the compatibility of the system with the change in demand, regional development, and available resources *(1)*.

On the other hand, an equitable transportation system is referred to as a system in which every traveler, drivers, pedestrians, and bicyclists have equitable mobility and safety. Recently, the concept of equitable transportation systems has been driven by the emerging artificial intelligence and big data analytics to deal with the surfacing equity issue in transportation planning and operations. In a sustainable and equitable transportation system, integrating the cyber space and the physical world is crucial since it makes the overall system dynamic. This integration includes data monitoring and analysis from the cyber space and the real world, developing new strategies, and applying simulation to evaluate the proposed methods. The goal of the sustainable and equitable transportation system is to generate a sustainable system that

includes benefit for all road users, i.e., both vehicles and pedestrians. Therefore, transportation sustainability and equitability can be divided into three main focuses:

- Sustainable and equitable intersection design in mixed traffic conditions
- Sustainable and equitable system for pedestrian detection
- Network performance analysis within the big data-driven environment

1.2 Sustainable and equitable intersection design in mixed traffic condition

Because of this coexistence of mixed traffic in the network, ensuring the safe operation of the Connected Automated Vehicles (CAVs) is also becoming one of the prime factors during the implementation process (1). An important application of CAV technology at intersections is to schedule green lights for all approaching vehicles so that the intersection capacity can be further improved. With the increasing application of this technology, intersection design is becoming more challenging. The existence of CAVs has an impact on the background traffic. In addition, CAVs can be classified into two different groups: (i) high priority CAVs (i.e., emergency vehicles) and (ii) low priority CAVs (i.e., Connected transit vehicles or passenger cars). An equitable intersection needs to consider both these high-priority and low-priority CAVs present in the roadway to serve green requests in the future. In the real world, CAVs can notify their arriving times in advance to the downstream intersections along their paths using automated navigation systems and advanced wireless communication networks. These advantages allow expanding the CAV scheduling horizon at intersections to a few minutes and considering both CAVs and traditional human-driving vehicles. Long-range communications will become possible for CAVs soon after the emerging cellular-V2X technologies are in place. The expanded time horizon offers the downstream intersections with sufficient time to gradually adjust their control plans, cooperatively organize the background traffic and consequently reduce interruption to the background traffic. By contrast, the existing CAV

scheduling strategy for signalized intersections mostly aims at shorter horizons (e.g., 10~20 s before CAVs arrive at intersections). Optimizing and adjusting traffic control plans within such a short period will inevitably bring interruptions and delays to general vehicles.

1.3 Sustainable and equitable system for pedestrian detection

Pedestrian detection (PD) technology is another aspect of an equitable transportation system. These PD devices can collect information about the location, behavior, waiting time, speed, and perception reaction time of pedestrians at the intersection. Therefore, it can reduce the application of traditional push buttons. A large portion of pedestrian-involved crashes occurs at signalized intersections. Thus, providing adequate protection and mitigating measures for crossing pedestrians is vital. It is important to study up-to-date pedestrian behaviors to design safer pedestrian facilities and an equitable transportation system. In the 21st century, transportation developments in the United States were largely focused on making the driver's experience as expedient and frictionless as possible while providing little protection or technological advancements for the needs of pedestrians. Guides for traffic controls at intersections have been developed for stewarding safety since the 1930s, but the occurrence of pedestrian fatalities occurring at intersections has remained relatively flat for the past two decades, with no strong upward or downward trends. Even it is seen that although traffic fatalities have generally been trending downwards, pedestrian deaths represent an increasing portion of overall traffic fatalities.

The surface transportation system is experiencing rapid changes today. Not only is travel demand increasing, but also the travel modes are diversified. People have more choices for travel other than traditional vehicles, from self-driving cars to e-scooters. There are many initiatives toward smart infrastructure and intelligent vehicles at federal, state, and municipal levels to accommodate these new trends. While these efforts are modernizing the transportation

system, issues of “equitable safety” are surfacing. According to the National Highway Traffic Safety Administration (NSHTA) report, pedestrian fatalities have increased by 44% from 2010 to 2019. In 2019, 6,590 pedestrians died of traffic crashes, the highest in 30 years (National Highway Traffic Safety Administration 2020). Unfortunately, these saddening facts suggest that walking or biking on the street is less safe today. While most of our efforts are devoted to improving mobility and safety for vehicles, the safety for pedestrians on roads is left far behind. Choosing walking or cycling over vehicles is not just a matter of choice but a matter of complex social-economic standings. Many underserved and low-income residents have to walk or bike in their lives. Technologies should not only serve those who can afford them but also those who need them. Smart transportation is smart only if it provides equitable safety for all road users. In particular, the vulnerable pedestrians should be paid the most attention to.

Pedestrian safety is a critical prerequisite to promoting walkability in cities since their mobility information is minimal at intersections. We know some people are willing to cross the intersection through a push-button. However, we do not know how many people wish to cross and how long they have waited. Excessively long waiting times will make pedestrians lose respect for traffic lights. It was observed a lot during the experiment (e.g., Jaywalking). In this research, the pedestrian tracking process is divided into three zones named as:

(i) pedestrian waiting zone,

(ii) entering zone and

(iii) cross end zone.

All these zones are shown in figure 1.1.

Figure 1.1 Pedestrian tracking process

With a modern detection technique with perception algorithms, the pedestrian behaviors at intersections will be tracked to understand such as

- Waiting time before crossing (ped-delay)
- Perception-reaction time to the onset of WALK
- Crossing speed distribution

There are many aspects to pedestrian safety improvement. From the perspective of technologies. The efforts can be categorized into four levels, as shown in Fig. 1.2

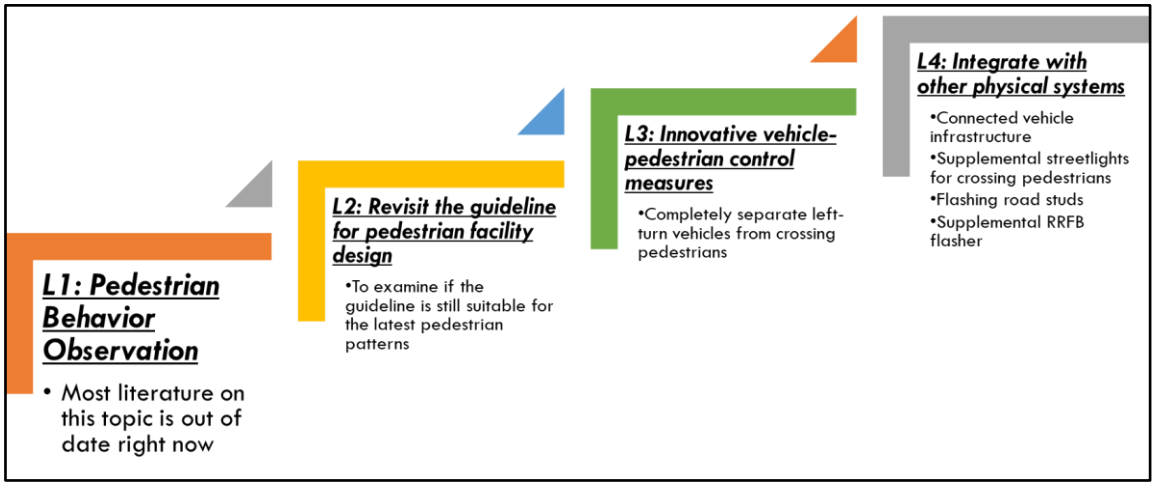


Figure 1.2 Four levels of pedestrian safety improvement

Level 1: Observe pedestrian behaviors: Pedestrian data are mostly composed of counts today. While the pedestrian counts reflect the needs for pedestrian facilities, they do not necessarily represent pedestrian safety. As such, it is necessary to collect pedestrian behavioral data for better-informed decision-making toward pedestrian safety improvement.

Level 2: Revisit the design guideline for pedestrian facility design. With more pedestrian behavioral data, it becomes possible to inspect the effectiveness of existing pedestrian facilities and validate the current design guideline. Level 2 measures primarily aim at planning.

Level 3: Novel control measures to improve pedestrian safety: this level involves real-time pedestrian behavioral data collection and real-time pedestrian protection, such as reducing pedestrian conflicts with vehicles. Level 3 measures are primarily aimed at operations, and the proposed D-FYA system belongs to this level.

Level 4: Integrate with other physical systems. At this level, multiple physical systems will be integrated to protect pedestrians further. For instance, the D-FYA system can be coupled with a lighting system to provide supplemental lights for crossing pedestrians at night. Level-4 solutions are rare today, but those novel solutions may be highly effective in protecting pedestrians.

Dynamic Flash Yellow Arrow (D-FYA):

This dissertation proposes a dynamic flash yellow arrow (D-FYA) mechanism based on a state-of-the-art LIDAR tracking system to protect concurrent crossing pedestrians fully. In category, it falls into the Level-3 pedestrian protection activities as defined above. The significant benefit of this new D-FYA method is to separate the concurrent crossing pedestrians from permissive left-turn vehicles while using all safe permissive left-turn capacities. This feature is especially beneficial when a phase duration is much longer than the required pedestrian crossing time.

1.4 Network performance analysis within the big data-driven environment

Successful development of an Equitable system requires some primary information. For example, the number of unoccupied vehicles on the roadway in the future, use of energy, and overall impact of the existing traffic on the environment are significant concerns to address. The system needs to be balanced according to that. To get this information, the future travel demand prediction result is necessary. Nonetheless, predicting the travel demand itself is a challenging task since it requires a good number of available data sources to get the real-time traffic behavior in a roadway. With the application of big data and connected vehicles, now it has become much more manageable. It is now possible to get real-time traffic movement and behavior in a network from the connected vehicle trajectory data. Now the future travel demand can be predicted to develop a sustainable and equitable traffic system.

Recent advancements in connected vehicles have increased the level of penetration and consistency of ping intervals of probes. Nowadays, high-fidelity vehicle trajectory data is readily available from commercial sources and can provide a near-real-time, cost-effective way to assess traditional travel time and delay characteristics on a corridor and approach-level performance at the intersection. Over 400 billion vehicle position records are already generated each month in the United States. Connected vehicle (CV) data has been made available recently by several transportation planning and traffic engineering companies. The data delivered by these companies provide important spatial-temporal characteristics of a significant sample of the vehicles that travel in a specific zone. Therefore, our research objective is to find potential uses of connected vehicles and develop an innovative framework for travel demand forecasting.

In this dissertation, an innovative framework for travel demand forecasting is going to be developed. This task aims to explore an innovative framework to forecast the travel demand in the DFW area. The current practice of travel demand forecasting in DFW is the classic “four-

step" method based on household surveys and traffic counts. The emerging traffic data, like the connected vehicle trajectories, bring opportunities and challenges. The novel data sets reveal much more information about the traveler than before and pave the way for enhanced accurate and high-fidelity travel demand forecasts.

On the other hand, the traditional travel demand forecast cannot take advantage of the total horsepower of such data sets. The inconsistency of various data sets and heterogeneous data quality are other issues of fusing the emerging traffic data with the traditional ones. This task will explore the innovative framework of travel demand forecasting based on connected vehicle data using state-of-the-art big data analytics and high-performance computing to address these issues.

1.5 Dissertation Objective

This dissertation presents a sustainable and Equitable Transportation System Design Under Connected vehicles and Big Data-Driven Environment. It focuses on the following specific objectives:

- Development of a congestion-aware heterogeneous connected automated vehicles cooperative scheduling problem at intersections. This objective aims to present a method that can provide a systematic approach to the green request accommodations with different priorities at intersections.
- Application of dynamic flash yellow arrow along with pedestrian detection technique to ensure pedestrian safety and reduce queue length and delay.
- Application of connected vehicle trajectory data in travel demand forecasting.

1.6 Structure of the Dissertation

The remaining part of the dissertation is structured as below:

- Chapter II discusses a comprehensive literature review on all the research objectives. Here, initially, a review of the previous work on connected automated vehicle scheduling problems is discussed. Secondly, a brief review of the application of pedestrian detection technique is presented, and finally, a review of the previous work application of connected vehicle trajectory data is presented.
- In chapter III presents a method that can provide a systematic approach to the green request accommodations with different priorities at intersections.
- In chapter IV, A new pedestrian behavior capturing system is developed using Dynamic Flash Yellow Arrow. Also, pedestrian detection will be deployed to examine the current pedestrian-related traffic signal timing guideline and provide data-driven recommendations for necessary updates to the existing design guideline.
- Chapter V presents an innovative framework for travel demand forecasting using connected vehicle trajectory data. This chapter explores an innovative framework to forecast the travel demand in the DFW area.
- Finally, in chapter VI, the dissertation is concluded by discussing the overall findings, citing the limitations, and providing directions for future studies.

CHAPTER II

LITERATURE REVIEW

This section is going to cover the comprehensive review of the existing works of three individual project objective.

2.1 Comprehensive review of the existing work on connected automated vehicles scheduling problem

The traditional intersection automation systems can be classified as fixed-time, actuated traffic signal system and adaptive traffic signal system. The actuated traffic signal system is a traffic control system, which uses the current demand and operations information collected from the detectors within the intersection. The adaptive traffic signal system is developed to respond with real-time dynamic vehicle arrivals. Such a system can adjust the signal timing plans in real time based on current traffic conditions, road capacity, and some real-time performance objectives.

Using the CAV technology, the communication between vehicle-to-vehicle (V2V) and vehicles to infrastructure (V2I) has become the new source of information to control the system[1, 2]. Autonomous vehicles use more than 10Hz connectivity for faster V2V interaction and V2I forecast [3]. Earlier, the traffic signal and vehicles are considered individually to control the network performance. Using V2I technology to share information among individual vehicles has made multiple vehicles coupled with traffic control [4-6].

The CAV green requests scheduling at intersections have also been studied. CAVs can report their speeds, locations, headways, steering angles, and accelerations. The real-time vehicle trajectories can also be predicted accordingly. Feng and his colleagues propose a real-time adaptive phase allocation algorithm by using connected vehicle data to optimize the phase sequence and duration[7-10].Taking vehicle locations and speeds as inputs, the developed algorithms first construct an arrival table and then minimizes the total delay caused by the green request accommodations. The arrival table also contains the estimation of the unequipped vehicle information, referred to as the "Estimation of Location and Speed" or (EVLS) algorithm. Feng et al. also present a theoretical formulation for joint optimization of vehicle trajectories and traffic timing in which vehicles are assumed to communicate with each other along with the road-side unit [9].

Priemer and Fredrich [11] develop the concept of a decentralized adaptive traffic signal control system for V2I communication data to minimize the total queue length. Through limiting the range of information exchange between the signal controllers and phase transition (maximal 20s), the system uses loop detectors to locate the vehicles in front of the stop line.

To solve the traffic signal optimization problem with the CAV environment, Dresner and Stone [12] propose an autonomous intersection management (AIM) policy. Intersections are divided into grids as a type of space-time resource. The algorithm will first examine the availability of the required resources to reserve the resources for individual vehicles. The AIM algorithm grants the resource in a First-In-First-Out (FIFO) manner. The AIM policy is further extended with more flexibilities later for the mixed traffic environments with both CAVs and traditional vehicles [13] and path-based AIM policy [14]. Datesh et al. propose an algorithm for platooned traffic [15]. Lee et al. propose a control algorithm based on CV data in which the authors define the cumulative travel time responsive (CTR) data from the moments when a vehicle enters the approach to the present moment [16]. By monitoring the queue length at the

downstream and adjusting offsets and splits of the upstream, Venkatanarayana et al. propose a signal control algorithm to avoid the oversaturated condition [17].

Another type of green request scheduling is based on classic nonlinear control delay modes as defined in Highway Capacity Manual (HCM) and the queue and shockwave profiles at a signalized intersection[18].This method is often used to estimate the traffic delay and bus delay under various traffic signal timings, such as the literature due to Li et al.[19], Liu et al.[20] and Han et al. [21]. To compare the transit buses with regular vehicles, the delays of buses and regular are often normalized into the passenger delays such as the literature [22, 23].

2.2 Comprehensive review of the existing work on pedestrian behavior capturing system

The pedestrian detection (PD) technology is essential in the traffic control system since it helps to activate pedestrian crossing signals as well as to extend the crossing time if the pedestrian is already present in the crosswalk. A rich body of literature on the PD technologies is available in which multiple solutions are presented. The discussed PD technologies include video image processing, infrared cameras, radar, and LIDAR sensors. The back-end algorithms for identifying pedestrians and their behaviors are mostly based on clustering and machine learning techniques. Specifying the detection area is essential because it provides an adequate passing zone on the intersection and sidewalk and identifies the waiting zone. This literature in focused on the following two aspects:

i. Pedestrian detection technique.

ii. Application of pedestrian detection technique in existing work.

2.2.1 Pedestrian Detection using Camera

In pedestrian detection techniques using a video camera, the whole crowd is identified, and the approximate number of the pedestrian is decided in surveillance video. The basic assumption of large-scale detection is that it will be considered the crowded region when any region is moving forward. Although several methods for this detection are available, background modeling and background subtraction are mostly used.

2.2.1.1 Application of detection using camera

Kilambi et al. used the Gaussian density method for estimating the number of people in a group [24]. He used both heuristic learning methods and shape models to identify the variation of captured data. The basic assumption of his model is there is an average statistical distance maintained between the members of any crowd. His projection method can find each blob area obtained from foreground segmentation in the world coordinate using camera calibration information. Although the method can lower the issues generated due to the moving objects of different height (i.e., vehicles) other than the human height, it cannot give the crowd's motion trajectory information. Besides, the blobs data is not capable of giving the actual crowd size. However, Chan et al.[25] and Yoshinaga et al.[26] identifies this limitation in later work. He use a blob descriptor to find the crowd size. He used background subtraction on the PETS2006 (PETS2006) dataset using the Parzen density estimation method, where the number of pedestrians is estimated using a neural network. Although the pixel values in his model are observed in the massive frames, the developed neural network cannot always give correct estimation. Chan et al. developed a modified formulation for surveillance video technology [27]. The database used to prove his concept is a one-hour video recorded by a stationary digital camcorder. Later he used Bayesian Poisson Regression for counting crowd size. Bhuvaneshwar et al. proposed a systematic approach for counting and detecting pedestrians at an intersection

using a video camera [28]. In this study, median filtering and thresholding were applied to identify the difference between the moving objects at the intersection using the height and area occupied by the object. He proposed a shadow removal algorithm for the detection and removal of the object from the frame. His system gives a general idea about the number and location of pedestrians at the intersection.

Although a large-scale (macroscopic) detection technique uses a video camera to identify the crowd size without the information of the location of the pedestrians, the position of the camera, camera angles, and features can create an error in results. This the reason the application of large-scale (macroscopic) detection is limited[29].

2.2.2 Pedestrian Detection using Thermal Camera/Passive Infrared

Another method of pedestrian detection is using thermal camera and passive infrared. Both Thermal cameras and passive infrared (PIR) sensors use passive detection of infrared light. The infrared light is capable of sensing shorter wavelength of 8-14 micrometers. Although when thermal images are used for pedestrian detection, the actual size and color information cannot be collected. Moreover, change in weather also impacts the outcome since the thermal sensors visualize temperature radiation from the objects in the images.

2.2.2.1 Application of Detection using Thermal Camera/Passive Infrared

John et al. calibrated and analyzed the image from the thermal camera (FLIR far-infrared camera) and visible camera (IDS visible camera) to perform pedestrian detection [30]. In his approach, two types of cameras are initially calibrated using a heated calibration rig and further used PSO algorithm for estimating affine transformation. The algorithm works in three ways, initially by using the calibration information, primary grid points are created. In the second and third approaches, objects and pedestrians are detected. For pedestrian tracking, the

unclothed regions of the human body are captured using a thermal camera because of having a relatively higher heat signature. Thus, the intensity and size-based thresholds are used to identify the human face blob in the frame, and the centroid information is used to detect the trajectory. At the same time, the trajectory is identified by the visible camera using background subtraction. Nonetheless images are a pair only when it is available in both cameras. Baek et al. proposed a thermal position intensity histogram (TPIHOG) for pedestrian detection at night using a thermal camera [31]. He used a combined TPIHOG and additive kernel support vector machine (AKSVM) to perform nighttime pedestrian detection better. In his work, Kim et al. developed a multi-stage cascade learning device for pedestrian detection at night time or in a location of lower light [32]. In his proposed approach, he estimated the distance between the detected pedestrian area and the infrared camera location with the information of the position of the pedestrian who is detected in the real-world environment in the 2D thermal image.

2.2.3 Pedestrian Detection using Active Infrared

Active infrared sensor is another method of pedestrian detection. Those sensors effuse an infrared light beam to the receiver located across a pedestrian path. If any pedestrian enters that path, the beam is blocked and thus one pedestrian count is added to the record. Although, the limitation of active infrared detection is that cannot identify pedestrians and bicyclists separately. Also, the range of detection location is also very small.

2.2.3.1 Application of Detection using Active Infrared

Because of its limitations, active infrared is generally used for pedestrian-only trails, where the pedestrian path is constrained and classification is not necessary (Kothuri et al. [33]).

2.2.4 Detecting using LIDAR Sensor

Light Detection and Ranging (LiDAR) sensors is a remote sensing technique widely used in many areas, including transportation. Although the application of LiDAR in transportation is mostly focused on autonomous vehicles, it is also applied to other road users like pedestrians for detecting and tracking while implemented in the field.

2.2.4.1 Application of Detection using LiDAR Sensor

Zhao et al. proposed a pedestrian tracking approach using multiple single row LiDAR sensors [34]. In his approach, real-time pedestrian behavior data from a wide area are collected by scanning, then moving objects are extracted. He used the Kalman filter for developing a tracking algorithm to identify pedestrian trajectories. In a later work, he applied a network of horizontal LiDAR sensors to monitor vehicle and pedestrian movement entering a large crowded intersection [35]. He used data clustering techniques in an integrated special and temporal data association framework to find the moving object and motion trajectory at the intersection. Nonetheless, the clustering was conducted manually without considering the same object entered the database from different sensors. Moreover, a few critical parameters were estimated based on experience, making the study lack generality and weak adaptability.

Zhao et al. presented a systematic approach for tracking and detecting pedestrians at an intersection using infrastructure-based LiDAR sensors [36]. The foremost step of the methodology is the background filtering of the collected data. After that, the objects are classified into pedestrian and vehicle, and object clustering and tracking are conducted with the speed and trajectory of each object data from the sensor. In a separate study, he used a deep auto encoder- artificial neural network (DA-ANN) for predicting the behavior of the pedestrians along the sidewalk using roadside LiDAR sensors [37]. His developed model

initially gathers pedestrian trajectory data from the roadside sensor and performs data extraction, partitioning, feature extraction, and model evaluation.

Lv et al. developed a systematic approach to extract high-resolution traffic data from roadside LiDAR sensors to get and extracted the trajectory information from the speed distance profile (SDP) of the road user to reduce vehicle-pedestrian conflicts [38]. Wu et al. used high-resolution micro traffic data (HRMTD) from LiDAR sensor based on the spatial distribution of laser points, which filters both static and moving background efficiently [39]. He used one background filtering method named 3Ddensity-statistic-filtering (3D-DSF) for efficiently separating static and dynamic backgrounds. In his study, Combs et al. identified the range for pedestrian sensors [40]. He estimated the maximum numbers of pedestrian fatalities that could be avoided if the system were converted into an automated vehicle environment. Grassi et al. developed a method based on data extraction and data fusion to detect the pedestrians and classify them depending on their movement direction using both the LiDAR sensor and video camera [41]. In his study, he classified the data without tracking or movement analysis. Ansari et al. developed a hybrid pedestrian detection technology to identify both moving and static pedestrians by incorporating both 3D LiDAR data and vision sensors for data clustering [42]. Visible image maps are generated from those that cluster for finding a common reason of interest. Furthermore, the pedestrians are identified using the Color-based Histogram of Oriented Gradients (HOG) feature along with the Local Self-similarity (LSS) feature provided in the Support Vector Machines (SVM) classifier.

By using two-dimensional LiDAR data and monocular camera image, Bu et al. proposed end-to-end neural network architecture for pedestrian detection where an image-based orientation detection technology is used to get the actual orientation of pedestrian from the 2D image [43]. He also proposed a Regional Proposed Network (RPN) for the non-oriented pedestrian data and Predictor Net for predicting oriented pedestrian.

Soundrapandiyan et al. [44] and Tang et al. [45] proposed an Offline Adaptive pedestrian detection using a neural network and collecting data from the sensor and video detection database. Soundrapandiyan et al. performed background modeling of the image collected from the thermal camera, and pedestrian detection is conducted by local adaptive thresholding using the parameter from the input image; on the other side Tang et al.[45] used controlled convolutional neural network (CCNN) architecture and modulating neural network (MNN) for detecting pedestrian in a location. CCNN works on adaptively generating a priority classifier, which is later dynamically adjusted by MNN.

2.2.5 Detecting using Radar Sensor

Radio detection and ranging (RADAR) is an active senor with a wide span of usable wavelengths (100m to 4mm). Because of the longer wavelength, it can cover more objects. Nevertheless, longer wavelengths produce lower resolution sensor data.

2.2.5.1 Application of Detection using Radar Sensor

In his works Manston et al. used Radar advanced driver assistance system (ADAS) features and were used in some the aforementioned PUFFIN crossings to detect pedestrians moving the crosswalk [46] .When necessary, a dual antenna system can provide a curbside detection zone and a crosswalk detection zone. Limitations of radar include susceptibility to error from rainfall, though a 13 GHz radar has improved upon this limitation from earlier 24 GHz models. Radar can be used to detect pedestrians up to 30 meters away, though sensors for commercial application generally specify a range of 18 meters.

In this literature review lays a comprehensive discussion on the different pedestrian detection technology and their application in some previous works.

2.3 Comprehensive review of the existing work on travel demand forecasting using connected vehicle trajectory data

In recent years, connected vehicle technology (CVT) has been considered the next big innovation platform for ITS. Over a wireless communication network, connected vehicles (CVs) will reliably share the traffic condition data with surrounding vehicles through vehicle-to-vehicle (V2V) communication and with transportation infrastructures through the vehicle-to-infrastructure (V2I) communication. Once the CV on-board unit compiles traffic data (e.g., vehicle position, number of brakes applied, etc.) in an autonomous way at predetermined intervals, they transmit the data to roadside units (RSUs). These data are subsequently processed yet again to produce the additional vehicle kinetics data (including average speed, acceleration, etc).[47]. The validity of this vehicle-generated data for incident detection, congestion identification, vehicle routing, and improving energy efficiency has already been investigated in prior study[48-53]. Different state departments of transportation (DOTs) have identified CVT as a potential tool for future congestion management and monitoring [54]. This application architecture demonstrates how various physical objects (i.e., entities) are connected to create real-time operational strategies using CV data. Once the roadside units (RSUs) have obtained vehicle status data from adjacent CVs, including data on traffic condition monitoring, they transfer the information to the traffic management center (TMC). The parameters are set by TMC to regulate the flow of information from the RSUs about the traffic situation. The maintenance and construction center, the transportation information center, and the emergency management center will receive the estimated traffic condition data from TMC for further action (e.g., traffic information dissemination to road users, emergency road maintenance, etc.). The most important task in real-time roadway network condition evaluation is estimating traffic density [55, 56]. Other traffic metrics, such as speed or path flow, can serve as stand-in signs

for a clogged situation. However, it has been determined that the most important factor in determining traffic congestion is density [57]. Traditionally embedded inductive loop detectors, surveillance cameras, and hybrid approaches (such as the loop detector-probe vehicle, loop detector-chase-car method, etc.) have been used to conduct real-time traffic operational analysis through density estimate [58, 59].

The origin and destination (O-D) of the created journey are estimated using path flow estimation. To determine the current traffic path flow and traffic operational state, many investigations have been carried out. Vehicle speed [60, 61], vehicle spacing (i.e., the distance between two vehicles) [62], and density have all been used to assess traffic operational circumstances. Qui et al. noted that in order to create a high-performance traffic management system, traffic density was necessary. Meanwhile, other research discovered that the single most important factor in influencing traffic congestion and path flow was density [63]. Density estimation using (a) single devices and (b) hybrid devices are the two subcategories of wired density estimation technologies. Single devices, such as loop detectors [64], video cameras, and microphones [65], estimate density. Inductive loop detectors are used in a vehicle re-identification system that Hernandez et al. developed to forecast the journey periods of re-identified vehicles in real-time [57]. The findings revealed that the mean absolute percentage error for both congested and non-congested condition detection was less than 4% after comparing with video camera data. Based on the difference between the various spectrum contents of the noise signals, Tyagi et al. classified the states of vehicle traffic into overcrowded, medium-flow, and free-flow circumstances using the cumulative auditory signal collected by microphones.[65]. The acoustic signal segments were classified using the Bayes classifier, and they discovered excellent classification accuracy (almost 95 %). Utilizing a discriminative support vector machine classifier improves classification accuracy. But this approach cannot extract microscopic traffic speed. Numerous hybrid devices have been

employed for density estimate in addition to these solo devices. This study used speed data gathered over the course of more than 37 hours and 250 freeway segments in Los Angeles by chase cars (a chase car is an instrumented vehicle that records the distance between the vehicle itself and another target vehicle representing typical driving behaviors in the study area; this distance is then used to determine speed and acceleration data for the target vehicle for each second[66, 67]) .To calculate segment density, data is aggregated for each segment once every 15 minutes after being matched with the information gathered by a loop detector (such as speed, count, occupancies, etc.).

Anand et al. collected data on flow and journey time using digital movies and GPS-equipped probe vehicles, respectively[66]. Here, the mean absolute percentage error of the prepared model is varied from 0.9 to 15.5%. Wireless communication is used by new CVT-equipped automobiles to gather and transmit traffic data to nearby vehicles and infrastructure (e.g., traffic signal, RSU). The efficacy of traffic data gathering using wireless V2V and V2I-based communication for density estimate has been examined in a number of studies. Barrachina et al. used roadmap topology features and beacon signals from CVs and RSUs (i.e., utilizing V2V and V2I, respectively) to estimate the density in a study. This study examined the traffic conditions in various cities using the ns-2 simulator[68].

The simulated data were used to create a mathematical relationship, which produced an average relative error for V2V-based density estimation of 1.02 %. The average relative error for V2I-based density estimation was 3.04 %. To calculate the street-junction ratio, which was a variable in the regression analysis model the authors used to estimate density, this approach depended on looking at the corridor maps. According to Barrachina et al., combining V2V with V2I would result in greater accuracy. In a different investigation, vehicle clustering was discovered to be a successful method for density estimate using both V2V and V2I communication [69]. The authors of this study created an algorithm to assess density using

vehicle clusters. In a simulation environment, the suggested approach was verified. Analysis showed that the program correctly predicted the thickness at various speeds of roadway traffic. To record route flow, Caceres et al. used several cell phones as probes[70] . The authors used anonymous cell phone data to approximate the vehicle number moving from one area (i.e., service area under the coverage of a set of base stations) to another. The experimental results revealed an absolute relative inaccuracy of 17% when compared to loop detector data. The created technique can also be applied to non-real-time estimations. DOTs must gather precise data from all throughout the state for real-time traffic management. Loop detectors are a common tool used by state DOTs to gather statistics on traffic.

Yang et al. [71] employed two alternative models in their investigation to estimate the offline OD using data from probe vehicles. When the distribution of probe vehicle ratios was homogeneous across different OD pairings, the results showed that both proposed models enhanced the current OD pattern to a comparable degree. The proposed models, however, have a limitation to get beyond this restriction to estimate the traffic assignment matrix because the probe OD ratios only provides an approximation of the traffic flow assignment.

Nie et al. presented a least square technique for path flow estimation. He et al combined a decouple estimator for finding the O-D trip matrix [72]. In his formulation he incorporated the measurement errors of traffic link counts and future path flow matrix. The advantage of his work is that it determined the equilibrium assignment mapping by exogenously identifying the optimal paths conforming to a user-equilibrium state. The result obtained from the numerical experiment shows that, using link travel counts gives them an O-D matrix which concurs with UE state .

CHAPTER III

CONGESTION-AWARE HETEROGENEOUS CONNECTED AUTOMATED VEHICLES COOPERATIVE SCHEDULING PROBLEM AT INTERSECTIONS LITERATURE REVIEW

3.1 Overview

In this section the methodology and results of the development of a congestion-aware heterogeneous connected automated vehicles cooperative scheduling problem at intersections is discussed.

Green accommodating mechanism at intersections: two types of green accommodations can be used for CAVs in practice include: (a) *Green Extension* to hold the current green until the green request sender crosses; (b) *Red Truncation* to terminate the green on other approaches early and turn on the corresponding green for the request sender. Fig. 3.1 illustrates these two types of strategies. Fig. 3.1A illustrates a green extension so that an approaching CAV can cross the intersection within the current cycle whereas Fig. 3.1B illustrates a red truncation so that the CAV can cross the intersection without stopping. In addition, if a high-priority green request is granted, the current traffic control mechanism will unconditionally cancel any active green accommodations for low-priority requests.

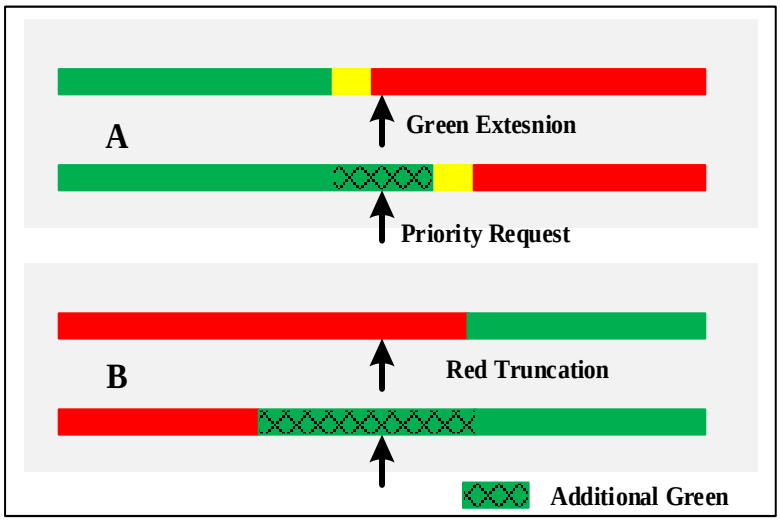


Figure 3.1 Green extension and red truncation to serve CAVs' green requests

The above strategies have been practiced for decades to serve transit priority and/or preemption. It contains little systematic optimization given the short notice (15 to 20 s before they arrive at stop bars). As such, those strategies may be inflexible in tackling complicated situations, such as more intense and heterogeneous green requests. In addition, when the traffic control system over emphasizes on CAV green requests, it may harm the mobility of background traffic. Most related literature in the past focuses on scheduling homogeneous green requests while the issue of background traffic and the heterogeneous nature of CAVs are not addressed enough.

Representation of heterogeneous priority of CAV green requests: in most literature, vehicles requesting a traffic signal priority are multiplied with a large weight in problem formulations to ensure that letting those vehicles cross earlier will create additional benefits. By contrast, we define two attributes in light of the phase-time network concept, a linear traffic control representation (Li et al. 2015) to describe the green requests by heterogeneous CAVs: the length of the time window and the "benefit" of accommodating this request. High-priority green requests should have strict (i.e., short) time windows, and the benefit of servicing a high-

priority request should be high. If the benefit of serving that time window is very high to outnumber the resulting delays, then the high-priority request will be served with certainty. By contrast, the low-priority requests have longer time windows to accommodate randomness due to unforeseeable congestions, and the service benefit is relatively low. The phase-time network is constructed as a forward, acyclic network. The outbound arcs from any node are defined according to the feasible next phase and valid range of split (green+ yellow+ all red clearance). Without loss of generality, one second is adopted as the intersection automation resolution.

Phase-time network construction: the phase-time network is a forward acyclic graph to represent traffic control plan over a time period. Traffic control formulation based on the phase-time network can keep the target problem's linearity [73]. Without loss of generality, let us assume a simple fixed phasing sequence as in Fig. 2b. The valid duration of all four phases is from 2 s to 4 s. at any phase-time node (p, t), 3 phase-time arcs leaves: $(p, t, p', t + 2)$, $(p, t, p', t + 3)$ and $(p, t, p', t + 4)$ where p' is the valid next phase (e.g $\Phi 1 \rightarrow \Phi 2$). If the ending time of a phase-time arc is beyond the time horizon, then it will be directly connected to the super sink node Z.

Following this logic, we construct a phase-time network with the time horizon of 7 seconds in Fig. 3.2c. As an example, the list of outbound arcs (p, t, p', τ) at the phase-time node *(1,1)* include *(1,1,2,3), (1,1,2,4), (1,1,2,5), (1,1,2,6).* Each outbound arc can be interpreted as *"Phase 1 starts at t=1, after yellow and all-red clearance, it turns over the green to phase 2 at t=min (T,τ), where τ=3,4,5 and T is the time horizon"*. Both blue and read phase-time paths are feasible for traffic control but only the blue phase-time path can serve the CAV green request within the required time window. As such, only the blue phase-time path is feasible in this context.

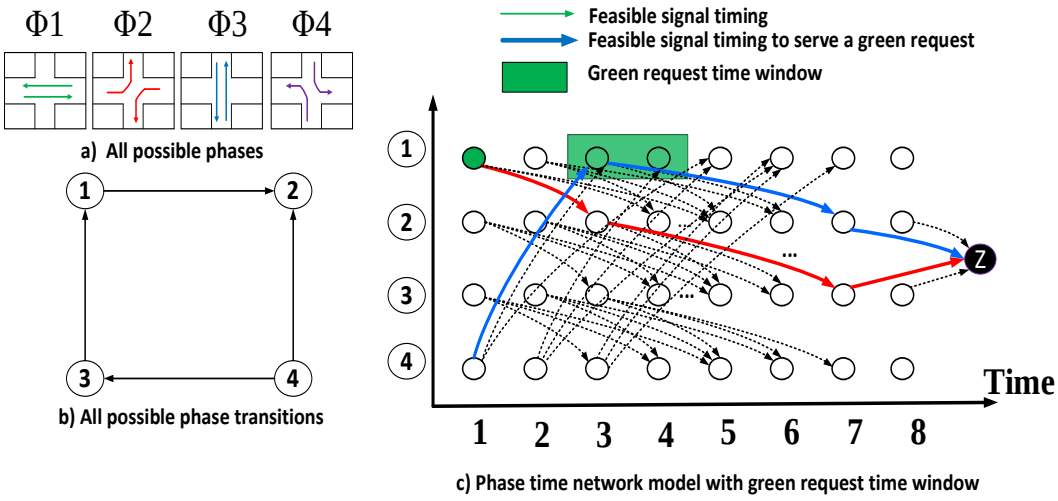


Figure 3.2 Illustration of phase-time network construction and timing plan representation considering a CAV green request time window

To further help readers understand the proposed concept, we analogize serving a CAV green request in the phase-time network to picking up a passenger in a road network, a green request can be interpreted as an additional constraint that a passenger at Φp must be picked up within a time window while a "virtual vehicle" is seeking the shortest path in the phase-time network (i.e., optimal phasing configuration). This problem can be viewed as a traveler salesperson problem with time windows. Fig. 3.3 demonstrates this analogy. High-priority CAVs include on-duty ambulance and fire trucks should be unconditionally served and therefore a short time window is defined with large serving benefits whereas those low-priority CAV like buses have longer time windows to reflect their tolerance of slight delays.

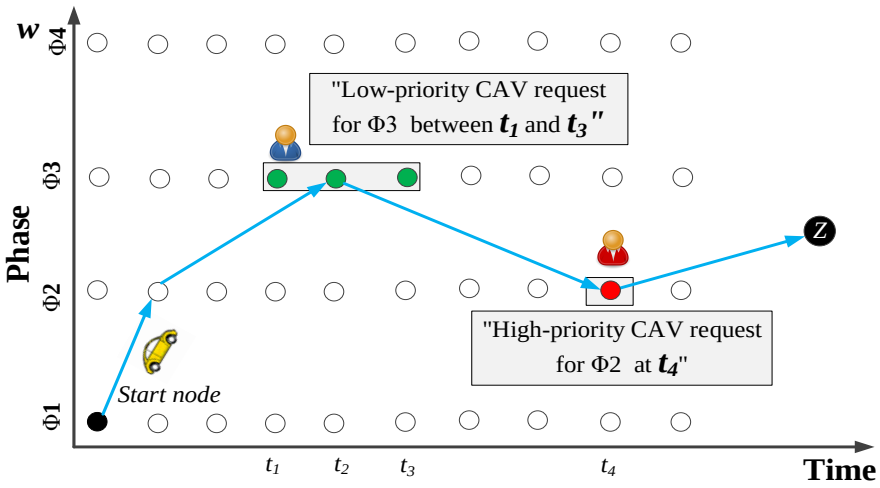


Figure 3.3 Analogy between a multimodal traffic control strategy on the phase-time network and the vehicle routing problem with pickup time windows

Although the traveler salesperson problem with time windows is an NP-hard problem in the worst case, the formulated problem in this context can be efficiently solved due to the optimization horizon and simplicity of phase-time network (i.e., forward and acyclic). Within minutes of the time horizon, the traffic signal timings could be adjusted at small steps to service multiple CAV green requests.

3.2 Methodology and problem formulation

3.2.1 Constructing a phase-time network for congestion-aware heterogeneous CAV scheduling at intersections.

The heterogeneous CAV green requests can be constructed on top of the original phase-time network. Whenever an approaching CAV sends green request to downstream intersections, the transmitted message contains the phase accompanied with a time window. The message can be interpreted as a series of nodes in the phase-time network. As shown in Fig. 3.4, green request one can be served under two conditions: (a) the corresponding green phase starts early and hold longer until the time window starts, similar with green extension

(blue-dash phase-time arcs); (b) green starts during the time windows (red-dash and black-solid phase-time arcs). All those phase-time arcs that can serve a CAV green request is referred to as the "admissible phase-time arcs to a CAV request". If the time windows of two green requests overlap, some phase-time arcs can service multiple requests (the black-solid phase-time arcs in Fig. 3.4).

The arc costs for general phase-time arcs are estimated according to the incurred background traffic delays which will be described in detail later. For those phase-time arcs admissible to the green request(s), additional benefits will be added as N×(-λ). (-λ) denotes a negative integer representing a reward to serve a green request and N denotes the number of green requests which can be served by that traffic control operation.

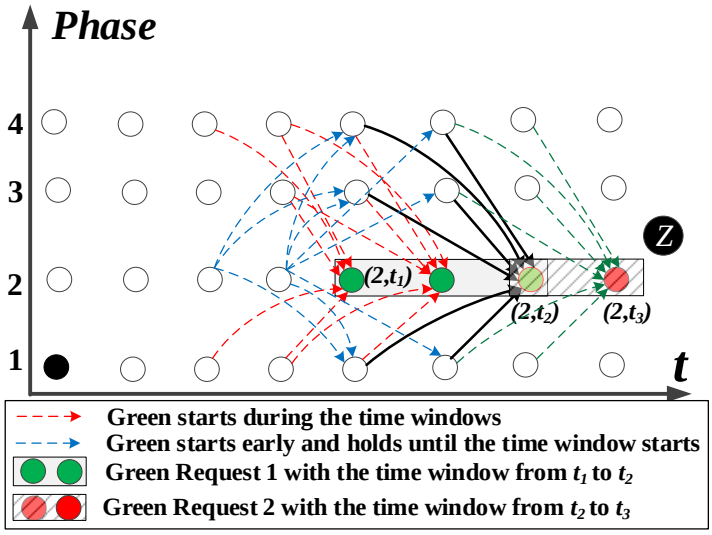


Figure 3.4 Illustration of admissible phase-time arcs to CAV green requests

3.2.2 A MILP formulation for congestion-aware heterogeneous CAV scheduling at intersections.

A MILP formulation for the congestion-aware heterogeneous CAV scheduling is proposed. The objective is to find a feasible traffic control plan that can serve all green requests while the background traffic is not severely interrupted.

Table 3.1 Parameters and variables for the new MILP formulation

Notations for road network and space-time network	
$G_o(N_o, A_o)$	Road network
$G(N, A)$	Space-time network
N_o, A_o, A_s	N_o : set of road network nodes; A_o : set of road network links; A_s: road links controlled by traffic lights (open if the corresponding phase is green (active); red (inactive) otherwise)
N, A	N : set of space-time network nodes; A : set of space-time network arcs
A_k	phase-time arc set admissible to the CAV request k
$i, j, (i, j)$	$i, j \in N_o, (i, j) \in A_o$
t, s, τ, h, H	Time indices; H: time horizon
t_0^v	Departure time of v
$FFTT_{(i,j)}$	Free flow travel time on link $(i, j), \forall (i, j) \in A_o$
$SR_{(i,j)}$	Saturation rate of $(i, j), \forall (i, j) \in A_o$
$L(i, j)$	The storage capacity of $(i, j) \in A_o$
$(i, t), (j, s), (i, t, j, s)$	$(i, t), (j, s) \in N, \forall (i, t, j, s) \in A$; *if* $(i, j) \in A_o, s = t + FFTT_{(i,j)}$; if $i = j$ (waiting arc), $s = t + 1$
v_1 , v_2	v_1 : Set of regular vehicles; v_2: Set of CAVs.
v, V	Vehicle $v \in V$; $v = v_1 + v_2$
$o(v), d(v)$	Origin and destination of v
$c_{(v,i,j)}, \mathbb{C}$	$c_{(v,i,j)}$: total free-flow path travel time if (i, j) is the last link of v's path; otherwise, 0; $\mathbb{C} = \{c_{(v,i,j)}\}, \forall v \in V$

$m(p,i,j)$	Mapping matrix from a traffic control phase p to its corresponding controlled signal link(s); always equal to 1 if (i,j) is a regular link otherwise can be either 1 if p is the current green or 0 otherwise.
Notations for phase-time network	
$\Psi(P,T)$	Phase-time network
p,P	Set of phases, $P=\{p\}$
$(p,t),(p',t')$	Nodes in Ψ, $p,p' \in P, t,t' \in T$.
$(p_o,0),(p_z,H)$	Origin (current phase) and destination vertex (ending phase) in Ψ
(p,t,p',t')	A phase-time edge in Ψ, representing: "phase p starts green at t, after yellow and all-red clearance, turns over green to phase p' at t' "; note $p \neq p'$ because there are no waiting arcs in the phase time networks
M,λ	Large positive number $M \gg \lambda \gg 1$.
N (p, τ, p', h)	the number of CAV requests can be serviced by (p,τ,p',h)
$\wp(k)$	The phase number which k requests for a CAV service
A_k	phase-time arc set admissible to the CAV request k (see Fig. 4)
C(p)	The clone phase of p
I_p	Set of cloned phase-time arcs from phase p to any cloned phase $C(p'), p' \in P$
c_0	The constant cost for each phase transition
Variables	
$x_{(v,i,t,j,s)} \in X$	Equal to 1 if v enters link (i,j) at t and leaves at s; otherwise, 0
$y_{(p,\tau,p',h)} \in Y$	Equal to 1 if and only if the phase-time arc (p,τ,p',h) is selected, otherwise 0. When$y_{(p,\tau,p',h)}=1$, it can be interpreted as phase p starts at τ and turns over green to p' after yellow and all-red clearance.

Please note that, even though the decision variables have 4 indices, they are two-dimension arc-time-indexed variables. The MILP formulation in light of the phase-time network is described as follows.

$$Minimize\ Z_1 = \sum_{v\in V}\sum_{s\le H}\sum_{1\le t}\sum_{(i,j)\in A_0}((s - t_0^v) \times \varphi_{(v,i,j)} \times x_{(v,i,t,j,s)}) -$$

$$\sum_{v\in V}\sum_{(i,j)\in A_o} c_{(v,i,j)} + \sum_{(p,\tau,p',h)\in \Psi}(c_{o_{(p,\tau,p',h)}} - N_{(p,\tau,p',h)} \times \lambda) y_{(p,\tau,p',h)} \quad (1)$$

3.2.2.2 Constraint for Traffic Dynamics

While constructing the constraints for traffic dynamics, to model the traffic control and the interaction between traffic control and traffic dynamics, the links in the network are considered of two types; i) Regular links whose capacity is not controlled by infrastructures like traffic signal system, and ii) Control links which are controlled by infrastructures like traffic control system. Fig 3.5 shows an example of a physical network with regular and control link.

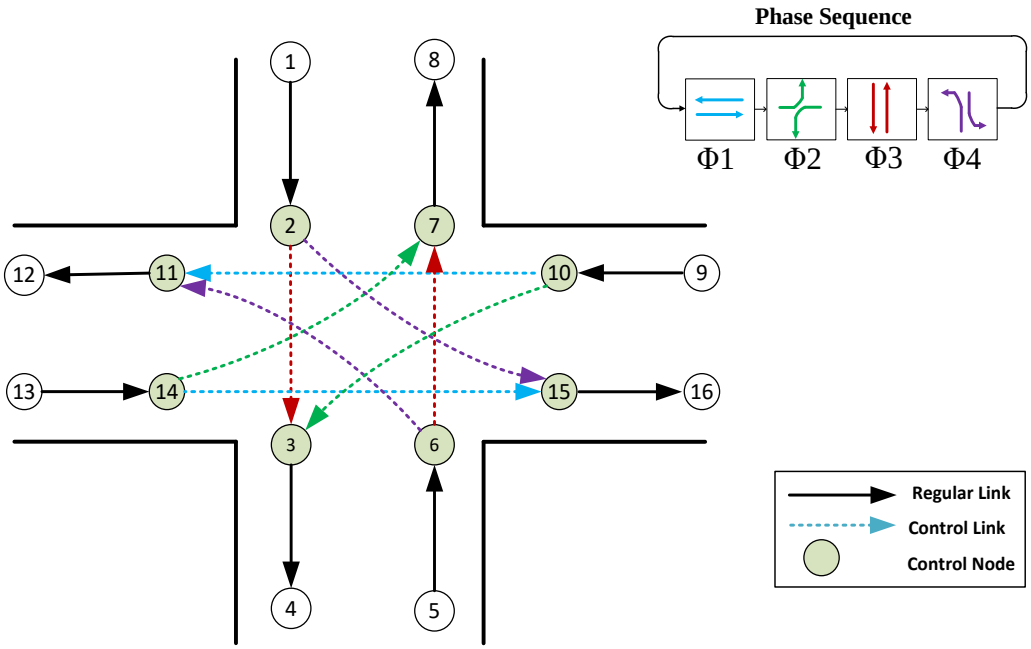


Figure 3.5 Constructing resilient phase-time network for heterogeneous TSP scheduling problem

Regular link capacity constraints

$$\sum_{v\in V} x_{(v,i,t,j,s)} \le SR_{(i,j)}, t,s\sim[1,H], \forall (i,j) \in \{A_o - A_s\}, p \in P$$

(2a)

Control link capacity constraints

$$\sum_{v\in V} x_{(v,i,t,j,s)} \le m(p,i,j) \times SR_{(i,j)} \times \sum_{(p,\tau,p',h)\in\Psi} y_{(p,\tau,p',h)}, t,s\sim[1,H], \forall (i,j) \in A_s, p \in P$$

(2b)

If the capacity of (i,j) is controlled phase p, then $m(p,i,j)$=1. When p is green, $y_{(p,\tau,p',h)} = 1$. Therefore, the RHS of (2) is equal to $SR_{(i,j)}$; when p is red, then the RHS of (2) is equal to 0.

Road link storage constraint

$$\left(\sum_{0\le\tau\le t}\sum_{v\in V} x_{(v,i,\tau,j,\tau+FFTT_{(i,j)})}\right) - \left(\sum_{0\le\tau\le t}\sum_{v\in V}\sum_{(j,\tau,i,\tau+FFTT_{(j,i)})\in A} x_{(v,j,\tau,i,\tau+FFTT_{(i,j)},}\right) \le$$

$$L_{(i,j)}, \quad \forall (i,j),(j,i) \in A_o, t\sim[1,\mathcal{H}] \qquad (3)$$

(3) define that the number of vehicles (i.e., the difference between cumulative arrivals and cumulative departures at any time) on a link must less than the link's storage capacity.

Flow conservation constraint at space-time network

$$\sum_{(i,t,j,s)\in A} x_{(v,i,t,j,s)} - \sum_{(j,s,i,s')\in A} x_{(v,j,s,i,s')} = \begin{cases} -1;\ (j,s) = o_v \\ 1;\ (j,s) = d_v \\ 0;\ otherwise \end{cases}, \forall v \in V, \forall (j,s) \in N \qquad (4)$$

Constraint (4) ensures all vehicles cross the intersections in a feasible solution.

3.2.2.3 Constraint for traffic control modeling in phase-time network

Flow conservation constraint at phase-time network

$$\sum_{(p,\tau,p',h)\in\Psi} y_{(p,\tau,p',h)} - \sum_{(p',h,p,h')\in\Psi} y_{(p',h,p,\,h')} = \begin{cases} -1;\ (p',h) = (p_o,0), \\ 1;\ (p',h) = (p_z,H), \\ 0;\ otherwise \end{cases} for\ \forall (p',h) \in \Psi \qquad (5)$$

Multiple-service prohibition for one CAV request

$$\sum_{(p,\tau,p',h)\in A_k}\left(y_{(p,\tau,p',h)}\right) = 1, \forall k \in K \quad (6)$$

A_k is the set of all admissible phase-time arcs to the green request k. Through "pre-building" the green request time windows, each admissible phase-time has a benefit. If a time window (for the low-priority green request) is long, (6) will prevent a traffic control solution from take the admissible arcs multiple times to maximize the solution benefit.

3.2.3 A resilient version of MILP formulation for congestion-aware heterogeneous CAV scheduling at intersections.

Z_1 can be solved with most commercial or open-source MIP solvers. Still, the MILP formulation may not contain feasible solutions if multiple CAV green requests are placed. In that case, secondary optimal solution should be provided by declining some requests appropriately. A challenge is that most MIP solvers reveal little information if a problem has no feasible solutions. To address this issue, the MILP formulation is modified into a resilient version. The idea of the resilient MILP formulation is to ensure that the MILP formulation will always have a mathematically feasible solution and from the solution it can be told how many CAVS green requests cannot be served. The proposed resilient MILP formulation starts with expanding the original phase-time network model by introducing the "clone" nodes and arcs in 3 steps:

Algorithm 1: Generating a resilient phase-time network from a standard phase-time network.

Step 1: For each control phase p, a "clone" phase is created. For any(p,t), a clone phase$(C(p),t)$ is generated.

Step 2: Duplicate all phase-time arcs inbound to a control phase p with CAV green requests. The new arcs connect from the same origin node to the corresponding clone nodes. For example, to duplicate a phase-time arc (p, τ, p', h) inbound to (p', h), the corresponding clone arc will be$(p, \tau, C(p'), h)$. (See the green arcs and purple arcs in Fig. 3.6)

Step 3: Duplicate all outbound phase-time arcs from a normal phase and connect to the corresponding clone phases. For example, to duplicate a phase-time arc (p, t, p', h) outbound from(p, t), a clone arc $(p, t, C(p'), h)$ is generated. (See red arcs and blue arcs in Fig.3.6)

The arc costs of all new clone phase-time arcs are set to very large cost M. Therefore, the clone phase will not be visited unless there are no feasible solutions to the original MILP formulation.

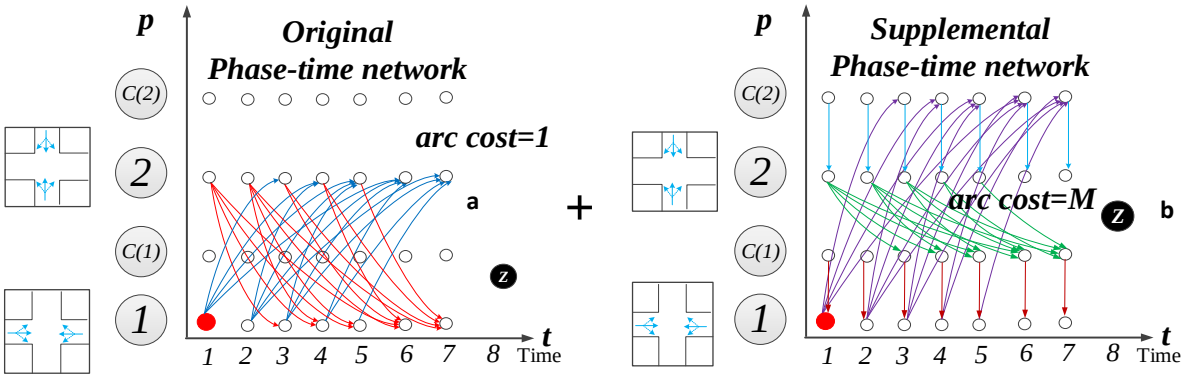


Figure 3.6 Construction of the resilient phase-time network for this problem

In the resilient phase-time network, Constraints (7) replace (6) to prevent multiple services to the same green request. If a CAV green request k cannot be served (i.e., no arcs in A_k can be selected selected), the first term of (7) will be zero and the second term must be 1, meaning the corresponding clone phase must be visited once. Assume the green request k requests control phase p' or $\wp(k) = p'$, then the constraints can be written as:

$$\sum_{(p,\tau,p',h)\in A_k} y_{(p,\tau,p',h)} + \sum_{(p,\tau,p',h)\in I_{\wp(k)}} y_{(p,\tau,\wp(k),h)} = 1;\ \forall k \in K \qquad (7)$$

According to the definitions, if $c_{(p,\tau,p',h)} = M$ (clone arcs), then $N_{(p,\tau,p',h)} = 0$ (no real green request service can come from a clone phase). Constraints (7) require that if a green request k cannot be served by its admissible phase-time arcs in A_k, then it must be served by one of clone phase-time arcs in $I_{C(\wp(k))}$ once and generate a large cost M. In other words, if any clone arc in $C(\wp(k))$ is selected in the final solution, it means the corresponding CAV green request is not served. The advantage of the resilient MILP is that it always generates a mathematically feasible solution. If any clone phase-time arc is selected, it is a sign of infeasible solution in the real word and can easily be identified which request is declined through finding out the clone phase-time arcs.

3.2.4 An efficient approximation of traffic dynamics with cumulative vehicle counting curves for real-time applications.

In this section, an efficient approximation to represent traffic dynamics for real-time applications is designed. For traffic control problem at individual intersections, the control delay on each approach can be represented by the areas between vehicle cumulative arrival and departure curve (the "A-D curves"). The analysis of A-D curves is a vital method to estimate the control delay at one approach [74]. The A-D curve method uses the queuing theory to calculate delays based on the number of individual vehicles approaching toward the intersection and crossing the stop line. Fig. 3.7 illustrates two different scenarios of an intersection under two timing plans. In Fig.3.7a the green times of phase 1 and phase 2 are appropriate. The vehicles arriving during red can be released within one cycle. By contrast, in Fig. 3.7b the green time of phase 1 is much shorter than the green time of phase 2. When the A-D curves are used to estimate the incurred delay under a given traffic control operation. As shown Fig. 3.7a when phase 1 is activated from τ to h. The added traffic control delays on all approaches are the shadowed area. At individual intersections, the simple A-D curves will

generate the same traffic control delay for each phase-time arc as the traffic dynamics model in space-time networks.

Assuming the A curves (i.e., vehicle arrival profiles) are prior known, then for each phase operations, its incurred delay can be estimated with the changes of D curve before and after this operation. In the meanwhile, the constant phase transition cost and the possible benefits for serving CAV green requests can also be estimated. The simple A-D curve method for traffic control delays allows for seeking the least-cost path algorithmically in the phase-time network, considering both traffic delays and CAV green request accommodations. Although the above formulation and solution algorithm is deterministic, we can adopt the *"Rolling Horizon"* to continuously update the status of CAV green requests. Since the CAV green requests are based on the estimated arriving times at intersections which are highly dynamic. The method of Rolling Horizon is to solve a stochastic problem by continuously solving a series of deterministic problems. Specifically, although the time horizon of this optimization is for minutes, the results will be applied for only 3-4 phase transitions. After that, the CAV green requests and A-D curves are updated, and a new problem will be solved.

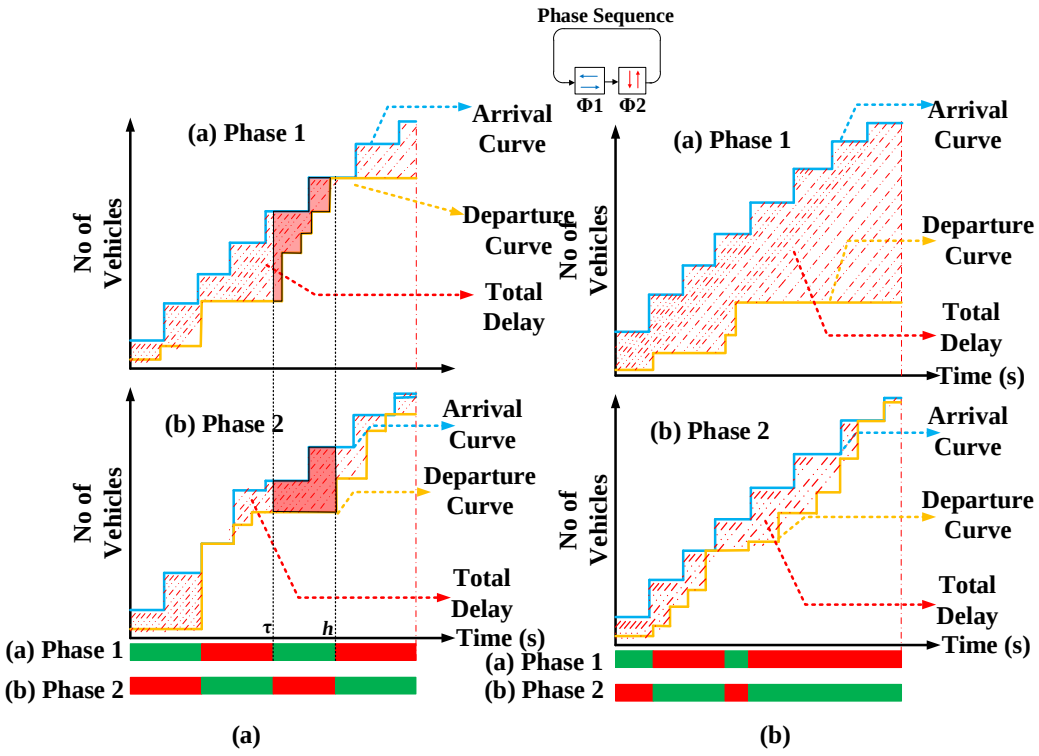


Figure 3.7 Construction of the resilient phase-time network for this problem

3.3 Numerical Example

Three experiments are conducted in this section: (I). Solving a simple example of the proposed MILP formulation with GAMS MIP solver. The main purpose of this experiment is to validate the MILP formulation; (II). Solving another example problem with the phase-time network and cumulative vehicle counting curves to prove the concept; (III). Evaluate the performance of Experiment (II) in a continuous, real-time manner within a high-fidelity microscopic simulation environment to bridge between the theoretical research and implementation in the future.

3.3.1 Experiment I: solving a sample problem with the GAMS solver.

As shown in Fig. 3.8A, an intersection network containing 16 nodes and 16 links, among those 16 links, 8 are normal links represented in solid lines with travel time 10s, and

the other 8 are control links in dash lines of different colors with travel time 2s. The traffic signal is controlled by 4 phases, the minimum green is 3 s, maximum green is 8 s, and all-red clearance plus yellow is set as 2 s. Within a horizon of 120 s, a total of 80 regular vehicles plan to cross the intersections, and they are released into the network with little competition for green lights when arriving at intersections. In addition, 15 approaching CAVs send green requests to the traffic control system within the time horizon, 9 of which are high-priority with a 1-second time window while the other 6 are low-priority with 10-second time window. The CAV green requests and their time windows are illustrated in Fig. 3.8B.

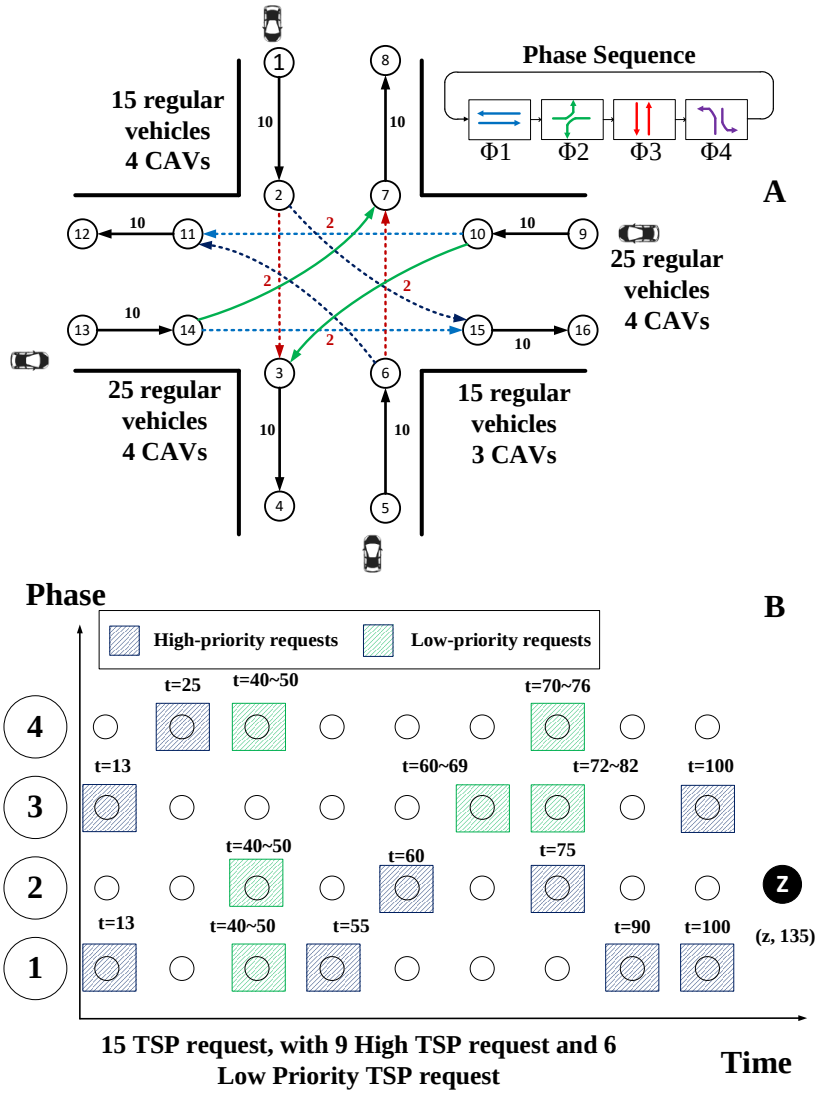


Figure 3.8 Example Settings for Experiment I

It takes the MIP solver about 16 s to find the optimal solution, and the optimal phase-time traffic control plan is displayed in Fig. 3.9. The clone phase 3 is visited, and it means that at least one CAV green requests are declined. After examining the resulting CAV trajectories (see Fig. 3.9), three CAV green requests are declined and wait at the intersections: CAV 9, CAV 11, and CAV 15. This is expected because those three CAV requests are set to compete with other requests, and it is impossible to serve all the requests. For instance, two conflicting high-priority CAV green requests are placed at t=13 with a 1-s time window, and both cannot be served for sure.

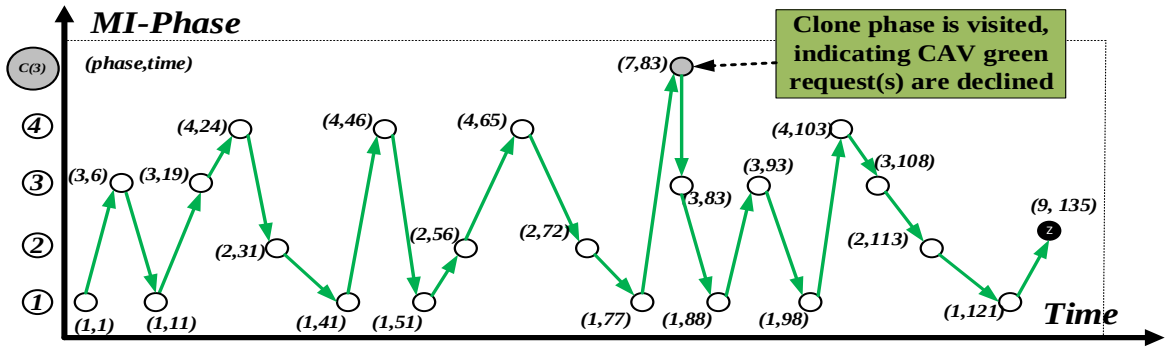


Figure 3.9 Optimal congestion-aware CAV scheduling in the resilient traffic-time network

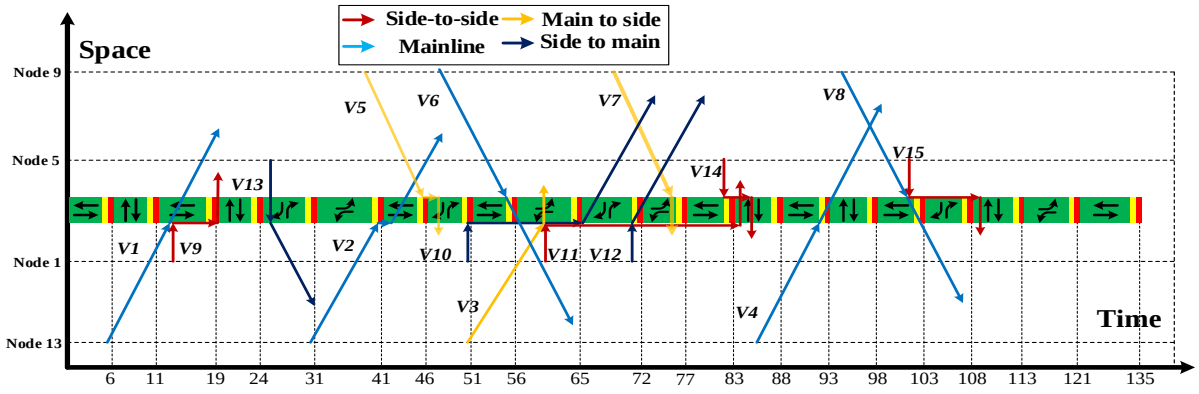


Figure 3.10 CAV trajectories at the intersection under the congestion-aware CAV scheduling

To find the control delay, the actual arrival time of the vehicle at any node and subtracted that from the free flow arrival time is measured. Therefore, the final solution reveals that the background 80 regular vehicles generate 263 seconds control delay (3.28 seconds/vehicle), and the maximum control delay is 8s. Hence, it can be concluded that the background traffic is not interrupted due to the intensive CAV services.

3.3.2 Experiment II: Congestion-aware CAV scheduling with vehicle cumulative counting curves in the phase-time network

In Experiment II, the proposed delay calculation based on the A-D curves in phase-time networks is evaluated. The purpose of this experiment is to validate the proposed method through a simple example. At an urban four-leg signalized intersection as in Fig.3.8a, each approach has one lane. Assume the traffic control plan contains 4 phases and the travel demands on each phase are 150 vehicles per hour for phase 1; 700 vehicles per hour for phase 2; 200 vehicles per hour for phase 3 and 1,100 vehicles per hour for phase 4. The saturate capacity for each phase is 1,800 vehicles per hour. Assuming there are 13 heterogeneous CAV green requests within 120 s. Seven are high-priority green requests and six are low-priority green requests. The requested phases and time windows are shown in Fig. 3.8b. Obviously, the CAV green requests are set excessive, and some are expected to be rejected. For the sake of argument, an arc penalty is also set for violating the fixed phase sequence ($\Phi1 \rightarrow \Phi2 \rightarrow \Phi3 \rightarrow \Phi4$) as 100 and the rewards for serving a high-priority green request and low-priority green request are 1,000 and 100 respectively. Since the benefits of serving green requests are set to be equal with or much higher than the penalty, the phasing sequence can be violated if the background traffic delay can be significantly reduced by doing so or CAV green request(s) can be served. The phasing sequence is barely maintained in this experiment given the intensity of the CAV green requests and arbitrarily set penalties and benefits.

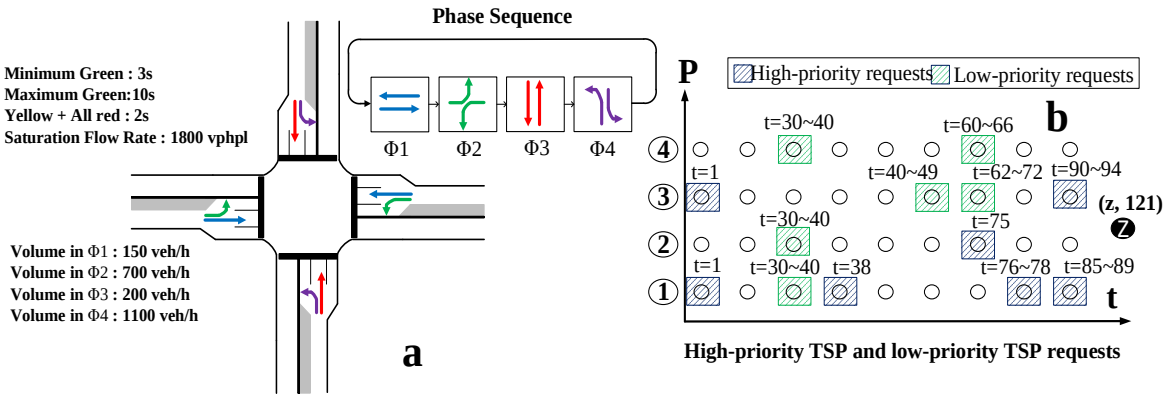


Figure 3.11 Problem configuration for Experiment II

Based on Algorithm 2, the optimal traffic signal timing in the phase-time path is displayed in Fig. 3.11. and Fig. 3.12. Fig. 3.12 evaluates how the CAV green requests are served. We can see that seven high-priority green requests are served but 3 low-priority green requests are declined due to their challenging time windows. Fig. 3.13 focuses on the traffic mobility under the optimal traffic control plan through the A, D curves of each phase. Fig. 3.13 reveals that there are no severe cycle failures or residual queues after 120 s while most CAV green requests are served. If a CAV green request is declined, that vehicle should be notified immediately. That CAV can either reschedule a new route to other intersections or bear with the possible delays at this intersection.

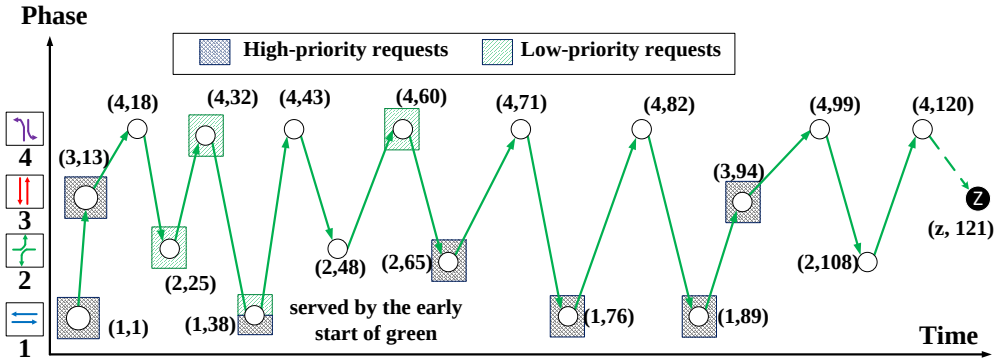


Figure 3.12 Optimal traffic control plan in phase-time network

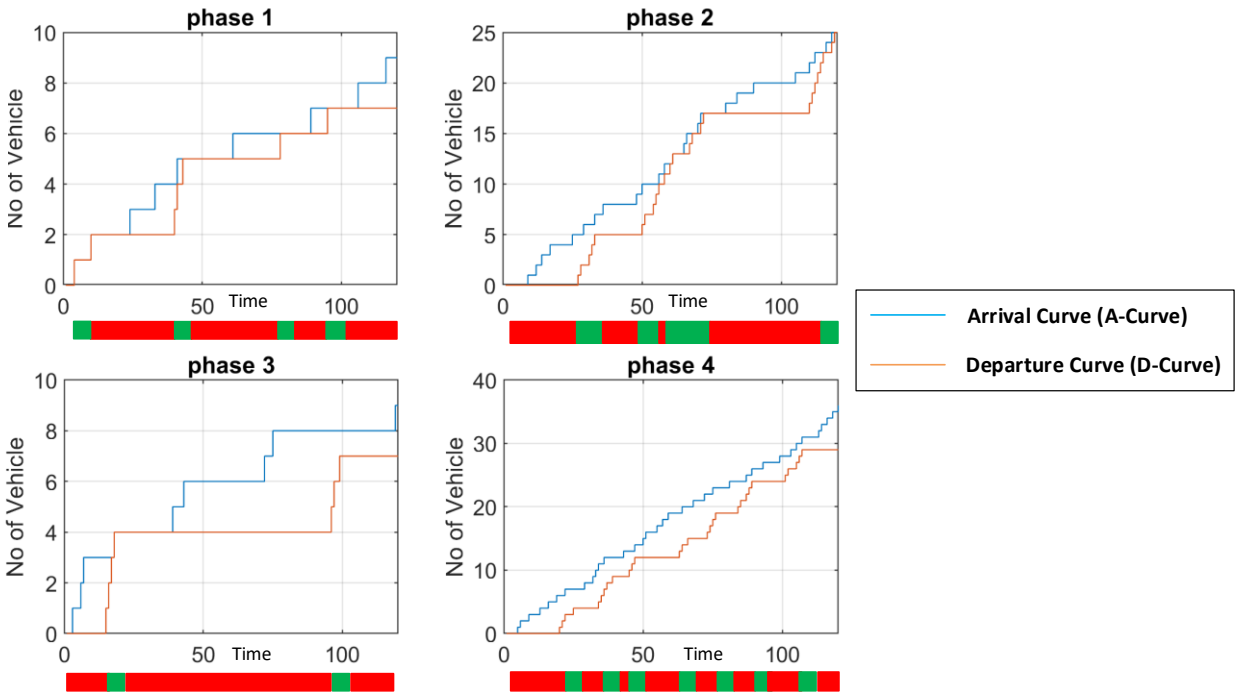


Figure 3.13 A-D curves on each phase under the optimal traffic signal timing plan

3.3.3 Experiment III: Real-time congestion-aware CAV green request scheduling at intersections with the microscopic traffic simulator

Experiment III is conducted within a high-fidelity simulation environment. The purpose of this experiment is to prove the concept of real-time heterogeneous scheduling for real-world applications. Experiment III provides a bridge from the theoretical contribution in this dissertation with real-world deployment. Each simulation scenario is run multiple times. Li and Mirchandani (Li and Mirchandani 2016) propose a hardware-in-the-loop simulation concept to evaluate novel traffic control algorithms. They propose to host novel traffic control logics on a hardened signal board computer (SBC) and then override the inherent control logic for intersection automation in simulation. This concept will ensure any novel control logic's implement ability in the field. The simulation environment is set up in PTV VISSIM 11.0 with a high-fidelity software-in-the-loop signal emulator. The CAV scheduler (Algorithm II) is hosted in a single-board computer communicating with the traffic signal emulator. The real-

time traffic control scheduling will be performed based on the latest projected vehicle arrivals and all the CAV green requests within the time horizon. Without loss of generality, it is assumed all CAVs request either phase-time phase P3 (NEMA phase Φ2, Φ6) or P8 (NEMA phase Φ4, Φ8).

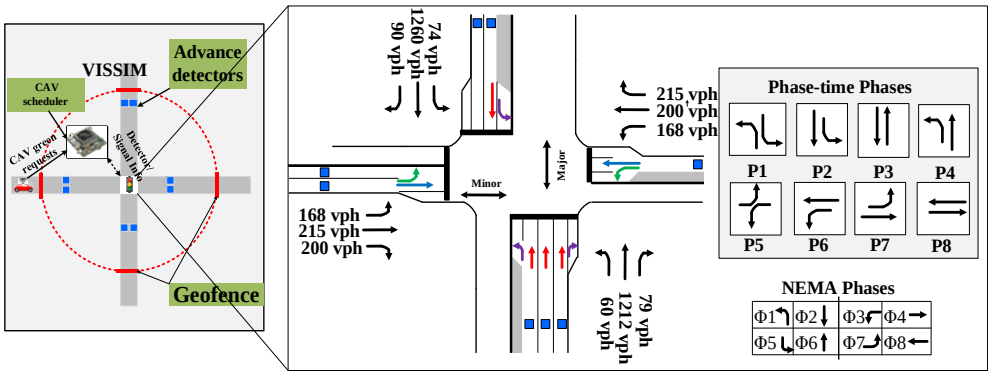


Figure 3.14 Layout of simulated intersection

Fig. 3.14 illustrates the simulation architecture for this experiment, the proposed congestion aware the CAV scheduler retrieves vehicle arrivals via advance detectors (located about 20 s from the stop lines) and receives location reports from CAVs via VISSIM COM (to emulate the CAV's on-board units) on each phase. Whenever a CAV reaches the location 30 s away from the stop line, it will report its requested phase and time windows to the CAV scheduler. Based on the received CAV green requests, the CAV scheduler will generate the set of admissible phase-time arcs for each request (Fig.3.3). Heterogeneous CAVs requesting high-priority and low-priority crossing are set as 2% and 2% of total vehicles, respectively (i.e., 96% are the background vehicles).

Periodically, the CAV scheduler seeks the least-cost phase-time path in the newly constructed phase-time network according to the values of three terms on each arc: constant phase transition cost, benefit (if it is an admissible arc for CAV green requests) and incurred traffic delays. The optimal phasing sequence will then be sent to the signal emulator and

override the baseline actuated traffic signal plan. The host single board computer has powerful CPUs and sufficient memory and so the optimization can be finished with 2-3 seconds which can meet the real-time requirements. The standard NEMA traffic signal emulator in the simulation engine is set to a "local-free" mode and the latest optimal timing plan out of the proposed algorithm can be implemented through three standard commands: Force-off, Hold and Omit to extend, terminate, and skip those preprogrammed phases. There is no need for timing plan transition like in the time-based traffic signal priority.

Although the time horizon for traffic control optimization each time is set as 90 sec, the number of phases on the least-cost phase-time path varies because their durations change. To apply the "rolling horizon" methods, the traffic control plan will be re-optimized after the first two phase-time control operations (i.e., the first two transitions) end. Therefore, the interval between two consecutive optimizations is dynamic.

3.3.3.1 Performance of CAV green request accommodation

Performance of CAV green request accommodation

A fully connected (i.e., fully adaptive) phase-time network for the control algorithm is constructed. The constructed phase-time network allows for phase transitions from one phase to another phase. Fig. 3.15 shows the resulting phase-time path within an hour and all the CAV green requests. Totally 142 CAV green requests are placed within an hour, and 134 (87%) CAVs have received green priority to cross the intersections.

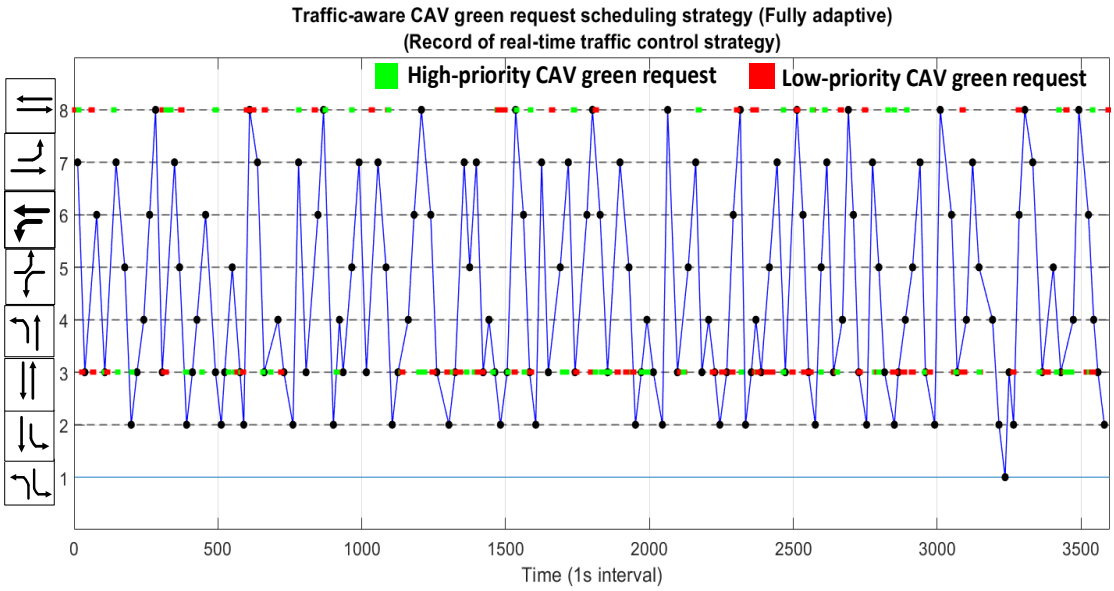


Figure 3.15 CAV green request accommodation with a fully adaptive phasing sequence

We can see that a fully adaptive phasing sequence will accommodate the CAV green requests with efficiency, but the service rates on approaches seems imbalanced. P1 in Fig. 3.15 is only selected once within a simulation hour while P3 and P8 are frequently because the volumes of both traffic and CAV green requests on P3 and P8 are high and so the algorithm must go to those two phases frequently to serve the approaching CAVs and regular vehicles. Note that the results in Figs. 3.15 are retrieved from the microscopic traffic simulation. Random maneuvers of individual vehicles were observed, making the vehicle arrivals more complicated than the models based on the A-D curves. It is observed that some vehicles slow down to yield or change lanes. These phenomena change CAVs' estimated arriving times at stop lines (i.e., the green request timing windows). It is also observed that vehicle startup loss varies from cycle to cycle. These realistic phenomena are beyond the capacity of the proposed traffic control algorithm. Nonetheless, we do see the proposed control algorithm proactively responds to arriving vehicles, and there are no residue queues most of the time.

3.3.3.2 Traffic queue length and delay evaluation under the proposed CAV scheduling

The resulting queue length and control delay on each approach. Fig. 3.16 shows the radar diagram of queue length and control delay analysis based on VISSIM evaluation outputs is also compared. As expected, after offer priorities for intensive CAVs, the resulting control delays are not as good as the optimal actuated traffic control plan. Nonetheless, traffic delays are well considered in the proposed control algorithm. The queue length and control delay on each approach (North bound through, left; South bound through, left; East bound through; West bound through, left) is acceptable and not significantly deteriorated.

Although the simulation design does not consider the pedestrian, it does not lose the generality because the pedestrian crossings can be easily satisfied by adding the following conditions when the traffic control plan is re-optimized. Specifically, the algorithm can check the following two conditions before re-optimization. If a pedestrian has been placed on a phase by the time of re-optimization, the minimum green on that phase should be adjusted as the bigger value of minimal green for vehicles and WALK plus clearance time for pedestrians. This setting will guarantee the green, if that phase is selected by the algorithm, will not end before the pedestrians have crossed the intersection. If a pedestrian phase is already active, the re-optimization should hold until that pedestrian phase is over.

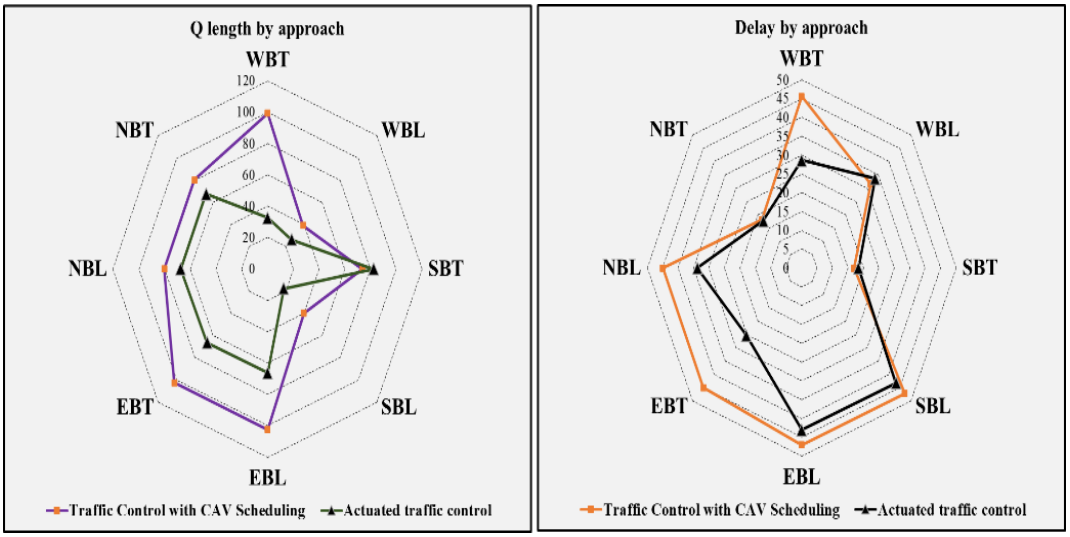


Figure 3.16 Queue length and control delay analysis under three control strategies

The proposed algorithm in this chapter assumes that the CAVs does not know the dynamic traffic control optimization results and maintain their driving behaviors as before. This assumption will certainly hold for connected vehicles but may not necessarily for automated vehicles, especially when the automated vehicles in the future can dynamically adjust their behaviors according to the newly incoming surround information, such as the traffic signal control. In that case, it would be necessary to conduct a joint optimization for traffic control scheduling and vehicle trajectories.

3.4 Summary

With the increase of CAV penetration rates, the competition of CAV green requests will become severe. Frequent CAV service at intersections also needs to consider the possible interruption to the background traffic. To address this issue, a new method to solve the congestion-aware CAV scheduling problem at intersections to balance the traffic mobility and CAV green requests is presented. A mixed-integer linear programming based on the phase-time network and space-time network is formulated to provide a theoretical foundation for this

problem. Compared with the previous literature on phase-time network modeling, the original phase-time network to a so-called “resilient phase-time network” model is augmented. The resilient phase-time network will guarantee a mathematically feasible solution when not all CAV green requests can be served due to competing time windows. From the outcome of the resilient phase-time network, we can tell which request is declined. By contrast, the MIP solver will leave little information for improvement if this problem is not solvable in the original phase-time networks.

CHAPTER IV

DEVELOPING A TRACKING-BASED DYNAMIC FLASH YELLOW ARROW (D-FYA) STRATEGY FOR PERMISSIVE LEFT-TURN VEHICLES TO IMPROVE THE PEDESTRIAN SAFETY AT INTERSECTIONS

4.1 Overview

Flash yellow arrow (FYA) was widely adopted to indicate permissive left-turn movements after a related research was conducted and concluded that the FYA would improve traffic safety [75]. Nonetheless, the FYA mechanism does not separate permissive left-turn vehicles from concurrent crossing pedestrians. As a result, pedestrian crashes reportedly increased at certain locations after the implementation of FYA. To address this issue, agencies either turn the FYA off or adopt a special feature in some brands of traffic signal controllers, referred to as “minus pedestrian”. The concept is temporarily suppressing the FYA for a cycle if the corresponding pedestrian phase is called. Fig. 2 shows the concepts of FYA and the “minus pedestrian”.

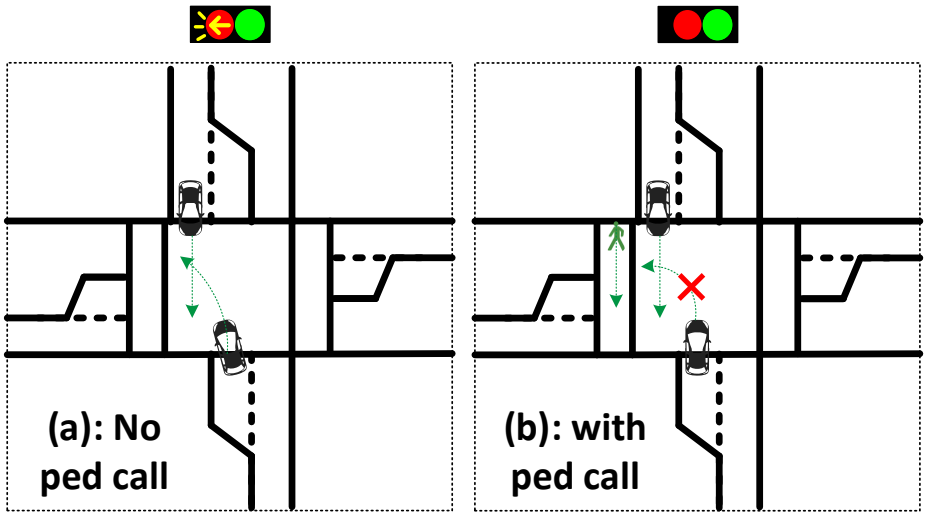


Figure 4.1 demonstrations of FYA and “minus pedestrian”

Although the “minus pedestrian” feature separates left-turn vehicles from concurrent crossing pedestrians, it also eliminates all the permissive left-turn capability for that cycle. This mechanism often creates excessive left-turn queues during peak hours when both pedestrian volumes and left-turn vehicle volumes are high. To address this issue, a new dynamic FYA or D-FYA based on a LIDAR-based pedestrian tracking system is designed. As shown in Figure 4.2, concurrent crossing pedestrians have conflict with left-turn vehicles only when they are within the so called “hazard zone”.

“Three-zone” pedestrian tracking with LIDAR sensors: In reality, many pedestrians push the pedestrian buttons and then choose to cross or “jaywalk” before the “WALK” sign starts. As a result, neither pedestrian phase nor FYA suppressing are needed for that cycle. In addition, the D-FYA can (and should) only protect those pedestrians who completely follow the traffic regulations because protecting both legitimate and illegitimate crossing pedestrians will considerably interrupt traffic signal operations. To address these issues, a “three-zone” method to filter and only track those legitimate crossing pedestrians is designed. As shown in Fig. 4.2. A pedestrian need to enter the wait zone first and push the pedestrian button in order

to be considered legitimate. The two waiting zones of each pedestrian phase are defined as "far-end" (Zone 1) and "near-end" (Zone 1') according to their relative locations to the left-turn vehicles. During WALK, if a pedestrian in Zone 1 and/or 1' enters the boundary zones (Zone 2 and 3), then this pedestrian is considered a legitimate pedestrian. If the same pedestrian reaches the other end, then this pedestrian crossing is considered finished. In case that the pedestrian button is pushed but no legitimate pedestrians enter the intersection, the pedestrian request is then considered void and ignored. The three-zone method will filter out those "jaywalking" pedestrians.

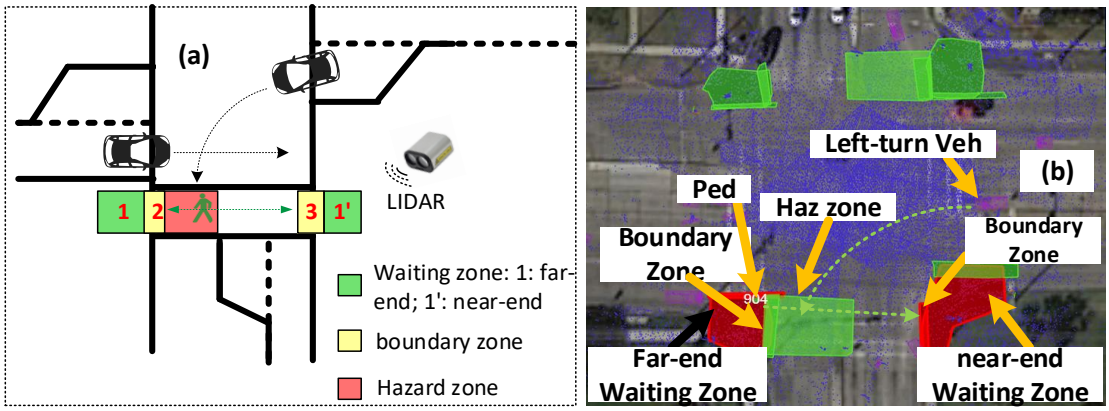


Figure 4.2 Three-zone pedestrian detection method at intersections: a: demonstration b: zone settings in the field in references of WB left-turn vehicles (City of Irving, Tx)

Dynamic Flash Yellow Arrow (D-FYA) based on pedestrian tracking: The D-FYA algorithm is elaborated as follows:

When a traffic signal green phase starts: Reset all the FYAs as originally programmed.

When this signal phase enters yellow and all-red: The proposed D-FYA algorithm will check the following items in sequence:

STEP 1: Check if this phase has a concurrent pedestrian phase. If yes, go to Step 2. If no, STOP

STEP 2: Check if the pedestrian button is pushed. If yes, go to Step 3. If no, STOP

STEP 3: Examine the existence of pedestrians in far-end and near-end waiting zones. There are two scenarios:

(i) No pedestrians are detected at either waiting zones, then D-FYA algorithm will keep the original FYA settings. Then go to STEP 4.

(ii) Pedestrians are detected at one or two waiting zones, then the D-FYA algorithm will suspend the programmed FYA temporarily. Then go to STEP 4.

When green or WALK starts, the D-FYA algorithm will check STEP 4 through 6 to make the final decision on FYA for this cycle.

STEP 4: At this step, there are four possibilities for pedestrians to enter intersection from two sides of waiting zones.

(i) During the WALK time, if pedestrians in the far-end waiting zone (e.g., Zone 1 in Fig.1) enter the intersection (e.g., zone 2 in Fig. 1) but no pedestrians in the near-end waiting zone (e.g., Zone 1' in Fig. 2) enter (e.g., zone 3 in Fig. 1). The FYA is suspended until all pedestrians have left the “hazard zone” (See Fig. 1). Then the FYA is re-activated until the current phase ends.

(ii) During the WALK time, if pedestrians in the near-end waiting zone enters the intersection while no pedestrians in the far-end waiting zone enters, then the FYA is suspended until all near-end pedestrians reach the other side of intersection (e.g., enter the boundary zone on the other side). Then the FYA is re-activate until the current phase ends.

(iii) During the WALK time, if pedestrians enter the intersection from both sides, then the FYA is suspended until all pedestrians reach the other side.

(iv) During the WALK time, if no pedestrians enter from either side, then the FYA is activated until the current phase ends.

Step 4 is the final step of this algorithm for each phase.

Discussion:

1. Note that the decisions on FYA at Step 3 is temporary because it is possible that a detected person in the waiting zones may not mean to cross, or a pedestrian may mistakenly push a pedestrian button. The final decisions of keeping or suspending a FYA will be determined after the green/WALK starts.
2. Note that the decision on activating FYA or not is made once and only once with each cycle to avoid confusing drivers and pedestrians.
3. If a pedestrian "jaywalks" so to get out of the boundary zone when reach the other side of intersection LIDAR sensors lose tracking it. The missing pedestrian will be allocated with a longest time walk time beyond which this person is considered to have crossed.
4. The proposed D-FYA is particularly effective when the opposing green is much longer than the needed pedestrian crossing time. Once all pedestrians are cleared, the FYA is re-activated and can provide a significant permissive capacity for left-turn vehicles. By contrast, the current "minus pedestrian" mechanism will unconditionally suppress the FYA all through the cycle even if no pedestrians cross or all pedestrians have crossed the intersection during a short time.

4.2 Analysis of permissive left-turn capacity under D-FYA

In this section, we analyze the changes to the permissive left capacity with the D-FYA as opposed to that with the PPLT under different scenarios. A traffic scenario in this context is composed of the duration of D-FYA, opposing through traffic volumes and the number of lanes, and the corresponding pedestrian volume. After the protected left-turn phase is over, the FYA will start together with the green for the opposing through traffic. The left-turn vehicles will begin to seek acceptable gaps to maneuver. While the queue of opposing traffic is being

discharged, then the left-turn vehicles cannot find the gaps due to the small headways. After the queuing vehicles are cleared, the left-turn vehicles will be able to find acceptable gaps to cross. If the permissive left-turn strategy is D-FYA, then the flash yellow arrow may start on time, be delayed, or even canceled, depending on the presence of pedestrians. It can be formulated as follows. Table 4.1 shows the notation for the formulation.

Table 4.1 Notation for Analysis of permissive left-turn capacity under D-FYA

Notations for Analysis of permissive left-turn capacity under D-FYA	
C	Cycle length (sec)
G	Green duration of opposing through traffic
s	Saturation rate (vehicle per hour per lane)
q	Volume of opposing through traffic (veh per hour per lane)
p	Volume of concurrent crossing pedestrian (ped per hour)
T	Time window for permissive left turn (sec)
T'	Time window for permissive left-turn under D-FYA (sec)
t_c	Queue clearing time (sec)
h	Headway (sec)
h_a	Acceptable gap for left turning (sec)
cap_{permLT}	Capacity during the permissive protected left turn (veh per hour per lane)
$\overline{cap_{permLT}}$	Capacity during FYA (veh per hour per lane)

As shown in Fig. 4.3, t_c is the time for clearing the queue of opposing through traffic and it can be calculated as:

Total arrivals during red and queue clearing time: $q \times (C - G + t_c)$

Total departures during the queue clearing time: $s \times t_c$

Then total arrivals are equal to total departures when the queue is cleared.

$$q \times (C - G + t_c) = s \times t_c \quad (1)$$

Therefore,

$$t_c = \frac{(C-G)}{\left(\frac{s}{q}-1\right)} \tag{2}$$

During t_c, left-turn vehicles cannot find acceptable gaps. The permissive time window for the left-turn vehicles with a cycle is:

$$T = G - t_c = G - \frac{(C-G)}{\left(\frac{s}{q}-1\right)} = \frac{(sG-qC)}{(s-q)} \tag{3}$$

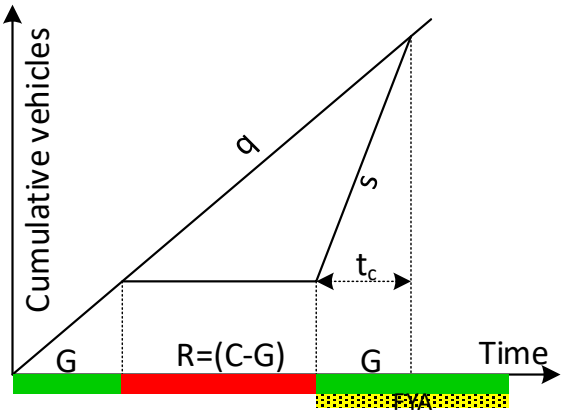


Figure 4.3 Queue clearing time calculation with the cumulative counting curves

Assume new opposing through vehicles arrive randomly, then the headway between arrivals can be approximated by an exponential distribution. The CDF function of the headway h is

$$F_{(h,q)} = \begin{cases} 1 - e^{-qh} & h \geq 0 \\ 0 & h < 0 \end{cases} \quad \text{(Multiple lanes)} \tag{4}$$

$$F_{(h,q)} = \begin{cases} 1 - e^{-qh} & h \geq h_{safe} \\ 0 & h < h_{safe} \end{cases} \quad \text{(Single lane)} \tag{4-a}$$

From (4) and (4-a), after the queue is cleared, the mean headway in seconds will be $\left(\frac{3600}{q}\right)$ seconds and the expected number of gaps of opposing through traffic during the permissive

left-turn time window will be $\frac{T}{\left(\frac{3600}{q}\right)}$. We can also estimate that the probability that headway is equal to or greater than the acceptable gap is:

$$F_h\{h > h_a\} = 1 - (1 - e^{-qh_a}) = e^{-qh_a} \quad (5)$$

So, the maximal left-turn capacity during the permissive time window will be

$$cap_{permLT} = \frac{T}{\left(\frac{3600}{q}\right)} \times e^{-qh_a} = \frac{(sG-qC)}{(s-q)} \times q \times e^{-qh_a} \quad (6)$$

The average pedestrian arrivals per cycle n_p can be calculated as

$$n_p = \frac{p}{(\frac{3600}{C})} \quad (7)$$

When pedestrians are only on the near side or on both sides (See Fig. 4.3), then they will use all the walk and pedestrian clearance time to cross the intersection. During that period, the D-FYA will indicate a red arrow for left-turn vehicles. After the pedestrian clearance timer expires, FYA will be displayed. As such, the remaining permissive time window T'will be

$$T' = T - t_{WALK} - t_{PC} = \frac{(sG-qC)}{(s-q)} - t_{WALK} - t_{PC} \quad (8)$$

And the permissive left-turn capacity is

$$\frac{T'}{\left(\frac{1}{q}\right)} \times (1 - e^{-qh_a}) = \left(\frac{(sG-qC)}{(s-q)} - t_{WALK} - t_{PC}\right) \times q \times e^{-qh_a} \quad (9)$$

When pedestrians are only on the far side, then they will take about 50% of pedestrian clearance time to cross the "hazard zone", then the D-FYA will start the flash yellow arrow for left-turn vehicles and so the permissive left-turn capacity in this case is

$$\left(\frac{(sG-qC)}{(s-q)} - -t_{WALK} - \frac{t_{PC}}{2}\right) \times q \times e^{-qh_a}$$

(10)

If $n_p \leq 1$, then the presence probability of one crossing pedestrian with each cycle will be n_p and the pedestrian can appear either on the near side or far side with equal (50%)

probabilities (See Fig. 4.3). The expected permissive left-turn capacity under D-FYA can be estimated as

$$\overline{cap_{permLT}} = \frac{\left(\left(\left(\frac{(sG-qC)}{(s-q)}--t_{WALK}-t_{PC}\right)\times q\times e^{-qh_a}\right)+\left(\left(\frac{(sG-qC)}{(s-q)}--t_{WALK}-\frac{t_{PC}}{2}\right)\times q\times e^{-qh_a}\right)\right)}{2} \quad (11)$$

If $n_p > 1$, then we can assume there are more than one pedestrian every cycle and they can be all on the near side, all on the far side, or both sides with equal (33%) probability. The expected permissive left-turn capacity of the D-FYA can be estimated as

$$\overline{cap_{permLT}} = \frac{\left(2\times\left(\left(\frac{(sG-qC)}{(s-q)}--t_{WALK}-t_{PC}\right)\times q\times e^{-qh_a}\right)+\left(\left(\frac{(sG-qC)}{(s-q)}--t_{WALK}-\frac{t_{PC}}{2}\right)\times q\times e^{-qh_a}\right)\right)}{3} \quad (12)$$

Table 4.2 Traffic settings for the permissive left-turn capacity calculation

	Permissive LT			**D-FYA (np<1)**			**D-FYA (np>=1)**		
Cycle length	110	110	110	110	110	110	110	110	110
opposing through green (s)	34	40	46	42	48	54	42	48	54
saturation rate (vphpl)	1500	1500	1500	1500	1500	1500	1500	1500	1500
the volume of opposing through traffic (veh per hour p)	400	500	600	400	500	600	400	500	600
Crossing ped volumes (ped per hour)	100	200	300	100	200	300	100	200	300
Through queue clearing time	3	3	3	3	3	3	3	3	3
time window for permissive LT	14	30	12	14	30	12	14	30	12
Acceptable gaps for permissive LT	8	8	8	8	8	8	8	8	8
pedestrian clearance time (sec)	5	5	5	5	5	5	5	5	5
pedestrians walk time (sec)	10	10	10	10	10	10	10	10	10

Table 4.3 shows the sensitivity results of capacity under different conditions.

Table 4.3 The sensitivity results of capacity under different conditions

Single Lane			
Oppose Th Volume (v/h/l)	**Perm LT Cap in PPLT**	**D-FYA (np<1)**	**D-FYA (np>=1)**
400	1632	903	797
500	1434	717	598
600	1027	642	513
Multiple Lane			
Oppose Th Volume (v/h/l)	**Cap of Perm LLT**	**D-FYA (np<1)**	**D-FYA (np>=1)**
400	1046	579	511
500	823	535	466
600	527	329	264

Fig. 4.4 shows the permissive left-turn capacity under different capacities.

Discussion:

1. From Eq. (6), if the opposing through traffic volume q is high, the permissive left-turn capacity will be close to zero. In that case, the traffic signal timing should only use a protected left-turn strategy to discharge the left-turn vehicles.
2. From Eq. (11) and Eq. (12), if the mainline green is much longer than the walk and pedestrian clearance time or even almost equal to the cycle length, then the D-FYA will reserve significant permissive left-turn capacities while separating the pedestrians from left-turn vehicles. By contrast, the PPLT with minus pedestrian phase will not reserve any permissive left-turn capability when pedestrians arrive with every cycle.

3. The above analysis is limited to isolated intersections because it assumes random arrivals of opposing through traffic after the queue is cleared. If an intersection is on coordination, then exponentially distributed headway for new arriving vehicles may be no longer valid because upstream vehicles will arrive in platoons. The analysis of acceptable gaps for coordinated intersections must be empirically performed.

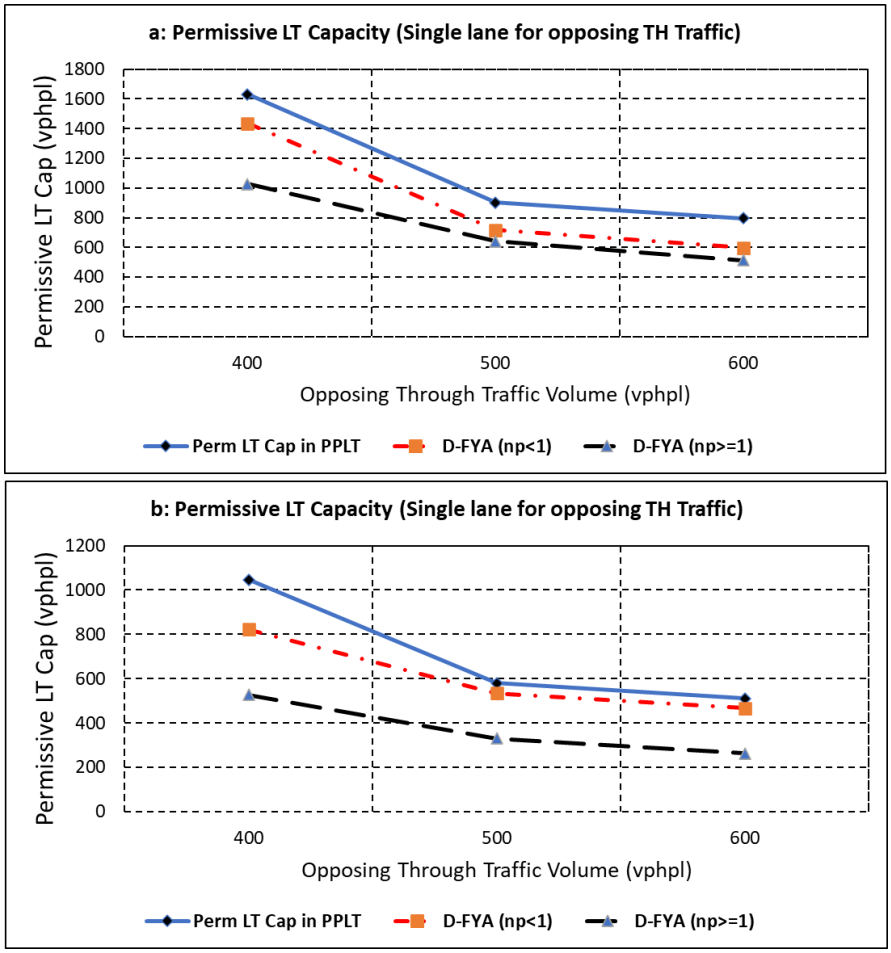


Figure 4.4 The sensitivity of capacity under different conditions: (a) single lane (b) multiple lanes

4.3 Case Study I: Evaluation of D-FYA's performance using the "Emulation-in-the-field" traffic signal simulation framework

In this experiment, we evaluated the performance of the proposed D-FYA algorithm in the field by verifying its real-time decisions according to the observed pedestrian behaviors in the field. The experiment design is referred to as the "emulation-in-the-field" framework. It

means all the traffic signal inputs and pedestrian behaviors are instantaneously collected in the real world to drive the D-FYA decision makings whereas the D-FYA decisions are not implemented but reported to the observers for verification. The purpose of this experiment is to evaluate the algorithm's reliability and accuracy in the field. The selected intersection is Cooper Street at the UTA Blvd, a major intersection connecting two urban campuses of the University of Texas at Arlington. The daily pedestrians crossing Cooper Street (mainline) range from 1,000 to 1,500 in a school day. The phasing sequence and pedestrian tracking zones are shown in Fig.4.5. There are four flash yellow arrows on all four approaches.

Whenever a phase starts, the D-FYA algorithm will run and report its findings (e.g., the presence of waiting pedestrians) and decisions (e.g., suppressing or activating an FYA) on the console screen. At the same time, a researcher verified the reported decisions according to their observations in the field based on the expected decisions according to the algorithm. The observation was carried out over 100 signal cycles with pedestrian crossings. Table 4.4 demonstrates how the D-FYA decisions were recorded and verified, using 5 cycles as an example.

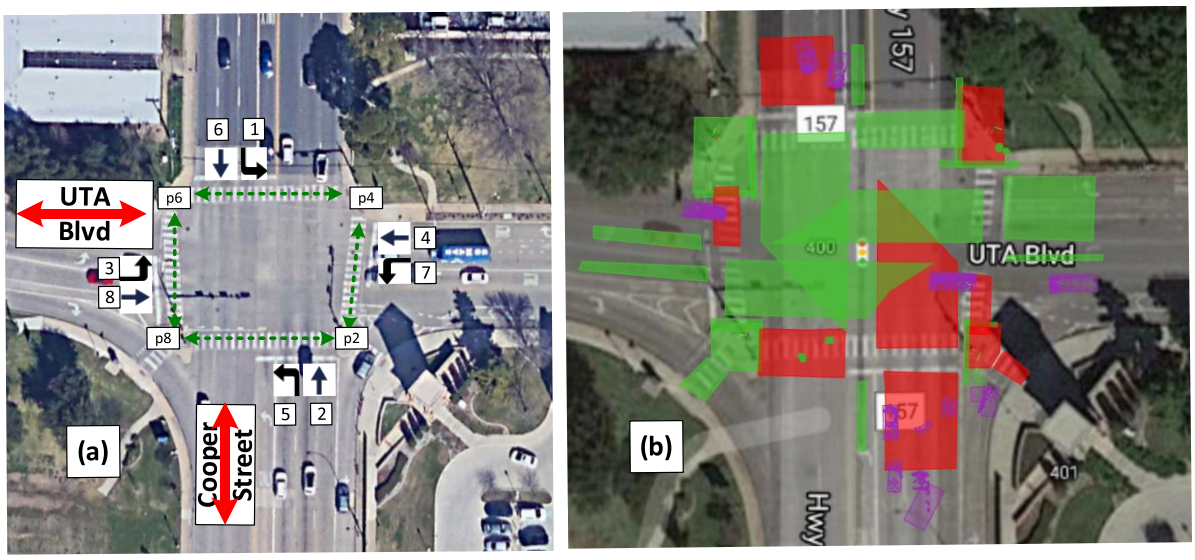


Figure 4.5 Phasing sequence (Fig. 4-a) and pedestrian sensing zone layout (Fig.4-b) at the Cooper Street at UTA Blvd, Arlington, Tx

Table 4.4 Records of emulation-in-the-field to verify the D-FYA strategy

Signal Cycle	Corresponding signal phases	near-end ped presence	far-end ped presence	Both ends ped presence	No ped presence	FYA started as scheduled	FYA delayed	FYA cancelled	Comment
1	8	1	0	0	0	0	0	1	1*
2	4	1	0	0	0	0	1	0	1*
3	4	0	0	0	1	1	0	0	2*
4	4	1	0	0	0	0	1	0	1*
5	4	0	1	0	0	0	1	0	1*

Note: 1*: verified by the researcher in the field; 2* verified a ped phase call but the pedestrian presence

The case study was conducted for 100 cycles in the field. There were 70 cycles that at least one pedestrian phase was called. Among those 70 cycles, 25 cycles only had near-end pedestrians, 25 cases with far-end pedestrians, and 9 cases with pedestrians on both sides. Comparing what the D-FYA was reported on the screen and what we observed in the field, we concluded that the D-FYA algorithm could make correct decisions in 93 cycles out of 100

cycles. Table 4.4 summarizes the D-FYA's performance under various scenarios. Table 4.5: Performance summary of D-FYA algorithm under different scenarios.

Table 4.5 Performance summary of D-FYA algorithm under different scenarios

Cycles with no ped calls	**Cycles only with near end peds**	**Cycles only with far-end peds**	**Cycles with both-end peds**	**Cycles with ped calls but no ped presence**	**The accuracy rate of the D-FYA algorithm**
30	25	25	9	11	93%

After finishing the experiment in the field, we further analyzed the recorded video and identified the possible reasons for incorrect D-FYA decisions. In those failed cases, the pedestrians either leaned to traffic light poles or multiple pedestrians stood too close for the LIDAR tracking algorithm to separate them effectively. This accuracy rate should further increase if the LIDAR tracking algorithm can improve in the future.

4.4 Case Study II: Mobility Evaluation of the D-FYA strategy using the "Cabinet-In-the-Loop" traffic signal simulation platform

In the second case study, we evaluate the mobility performance of the D-FYA as opposed to the other two common permissive left-turn strategies: (I) protected + permissive left turn (PPLT); and (II) protected + permissive + minus-ped-phase. The first strategy is to show the left-turn vehicles with a green arrow followed by a flash yellow arrow whereas the second strategy is to show the left-turn vehicles with a green arrow first and then examine if a pedestrian call is placed. If so, then a red arrow is displayed until the end of opposing green. Otherwise, the flash yellow arrow is activated.

The intersection of the West Walnut Hill Ln at the North Belt Line Rd in the City of Irving, Texas was selected to develop a simulation model. Fig. 4.6 shows the movements and phasing sequence.

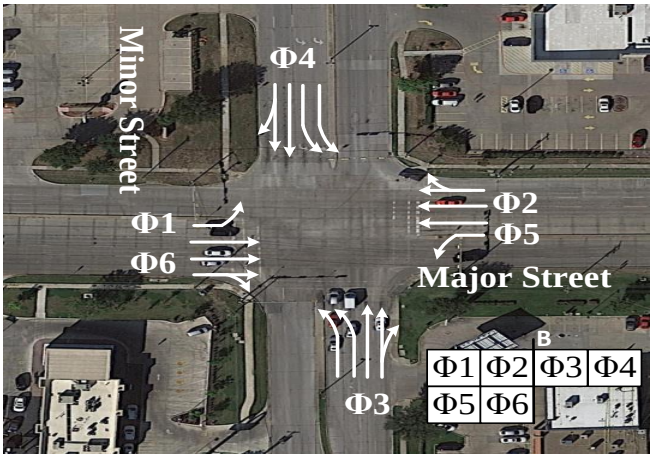


Figure 4.6 Layout of intersection for the 2nd case study

Cabinet-in-the-loop traffic signal simulation platform: the "minus-ped-phase" feature is not available in traffic signal controllers until very recently. So, it is not yet supported by any traffic signal simulation engine. To keep a high-fidelity and fair comparison, we developed a cabinet-in-the-loop traffic signal simulation platform for this experiment. As shown in Fig. 4.7, two control units (CU) are coupled with the VISSIM simulation engine. The first CU is a fully scaled traffic signal assembly. Through the input and out serial ports of the assembly, we retrieved the latest traffic signal status in the traffic signal controller, and we then sent it into VISSIM simulation via the provided traffic signal control API. On the other hand, the real-time detector status in the simulation is collected via the signal control API and then sent into traffic signal assembly via its input serial port. The hardware traffic signal controller will decide according to the detector inputs, including the FYA and minus pedestrian phase for the FYA.

A challenge in this experiment is that pedestrian tracking is not straightforward in simulation. To address this issue, we developed a second virtual controller in simulation for the D-FYA strategy. Its logic is to issue a red arrow if there are crossing pedestrians (i.e., the blue detectors are occupied by pedestrians) otherwise, it will issue a green arrow. The virtual controller issues red light only when the pedestrian phase is activated, so pedestrians (if any) enter the

intersection. The simplified D-FYA algorithm will not lose its generality since pedestrians have no random exceptions in a simulation like jaywalking.

As shown in Fig. 4.7, the blue detectors are configured to detect concurrent crossing pedestrians. Two signal heads, controlled by the hardware controller and by the virtual D-FYA controller, respectively, are placed in sequence for the left turn vehicles. The permissive left-turn vehicles can seek gaps and enter only if neither traffic signal head is red. As an illustration, when the opposing (SB in Fig. 4.7) traffic light turns green with the concurrent pedestrian phase, the hardware traffic signal controller will turn the first signal head to a flash yellow arrow. In the meanwhile, if the virtual controller detects the presence of crossing pedestrians, it will turn red, preventing vehicles from entering the intersection. If the virtual controller does not detect the pedestrian presence, it will indicate a green arrow. A flash yellow arrow and a green arrow will allow left-turn vehicles to enter the intersection during the permissive left-turn phase. This configuration can, in essence, start, delay, or cancel a programmed FYA within a cycle.

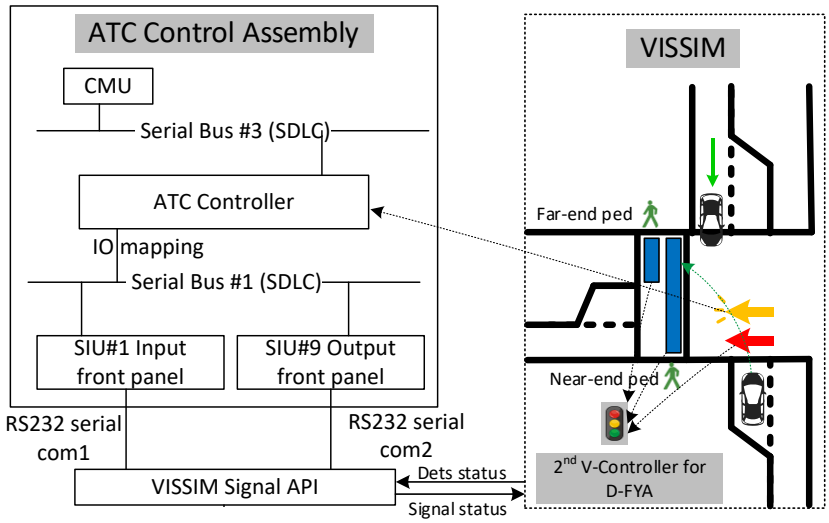


Figure 4.7 The architecture of Cabinet-in-the-loop traffic signal simulation for the D-FYA evaluation

Without loss of generality, the mainline vehicle and concurrent crossing pedestrian volumes are set as low, medium, and high to evaluate the performance of three permissive left-turn strategies (see Table 4.6). The experiment also excluded the possibility of "starvation" by extending the mainline left-turn lanes to ensure the mainline traffic was not affected by different permissive left-turn strategies.

Table 4.6 Vehicle and pedestrian volumes for different scenarios

Volume	**Southbound**			**Northbound**			**Westbound**			**Eastbound**			**Pedestrian**
	L	T	R	L	T	R	L	T	R	L	T	R	
Low	75	200	30	75	200	30	300	500	120	300	500	130	100
Medium	75	200	30	75	200	30	500	500	120	500	500	130	200
High	75	200	30	75	200	30	750	500	120	750	500	130	350

Nine simulation scenarios are generated with the combination of available vehicle and pedestrian volumes. They are referred to as:

1. LVLP: low vehicle volumes and low pedestrian volumes.
2. LVMP: low vehicle volumes and medium pedestrian volumes.
3. LVHP: low vehicle volumes and high pedestrian volumes.
4. MVLP: medium vehicle volumes and low pedestrian volumes.
5. MVMP: medium vehicle volumes and medium pedestrian volumes.
6. MVHP: medium vehicle volumes and high pedestrian volumes.
7. HVLP: high vehicle volumes and low pedestrian volumes.
8. HVMP: high vehicle volumes and medium pedestrian volumes.
9. HVHP: high vehicle volumes and high pedestrian volumes.

Fig. 4.8 shows the mainline left-turn queue length (in feet) comparison among three permissive left-turn strategies. It reveals that the mobility performance of D-FYA is between

the PPLT and “PPLT with Minus-pedestrian-phase” in most cases. In some cases, the D-FYA is much better than the “PPLT+Minus-Ped-phase” (e.g., the MVHP scenarios) in mobility while separating the left-turn vehicles and pedestrians. When the opposing through traffic and pedestrian volumes are both high, all three permissive left-turn strategies will degrade to the protected-only left-turn strategies (e.g., the HVHP scenario) because the left-turn vehicles cannot find the acceptable gaps. A similar pattern also shows in the delay analysis (Fig. 4.9).

Discussion II

From the simulation results, it can be concluded that for both low vehicle and medium traffic conditions, PPLT and D-FYA has better performance over PPLT+Minus-Ped-Phase strategy whereas the D-FYA and PPLT with Minus-Pedestrian-Phase have the same pedestrian protection. Although, when both vehicle and pedestrian volumes increase to a high level, all three permissive left-turn strategies show similar delays and queue lengths to the protected-only left-turn strategy. This is because the left-turn vehicles cannot find the acceptable gaps during FYA. It implies that we may need to prohibit any permissive left-turn strategies under certain scenarios.

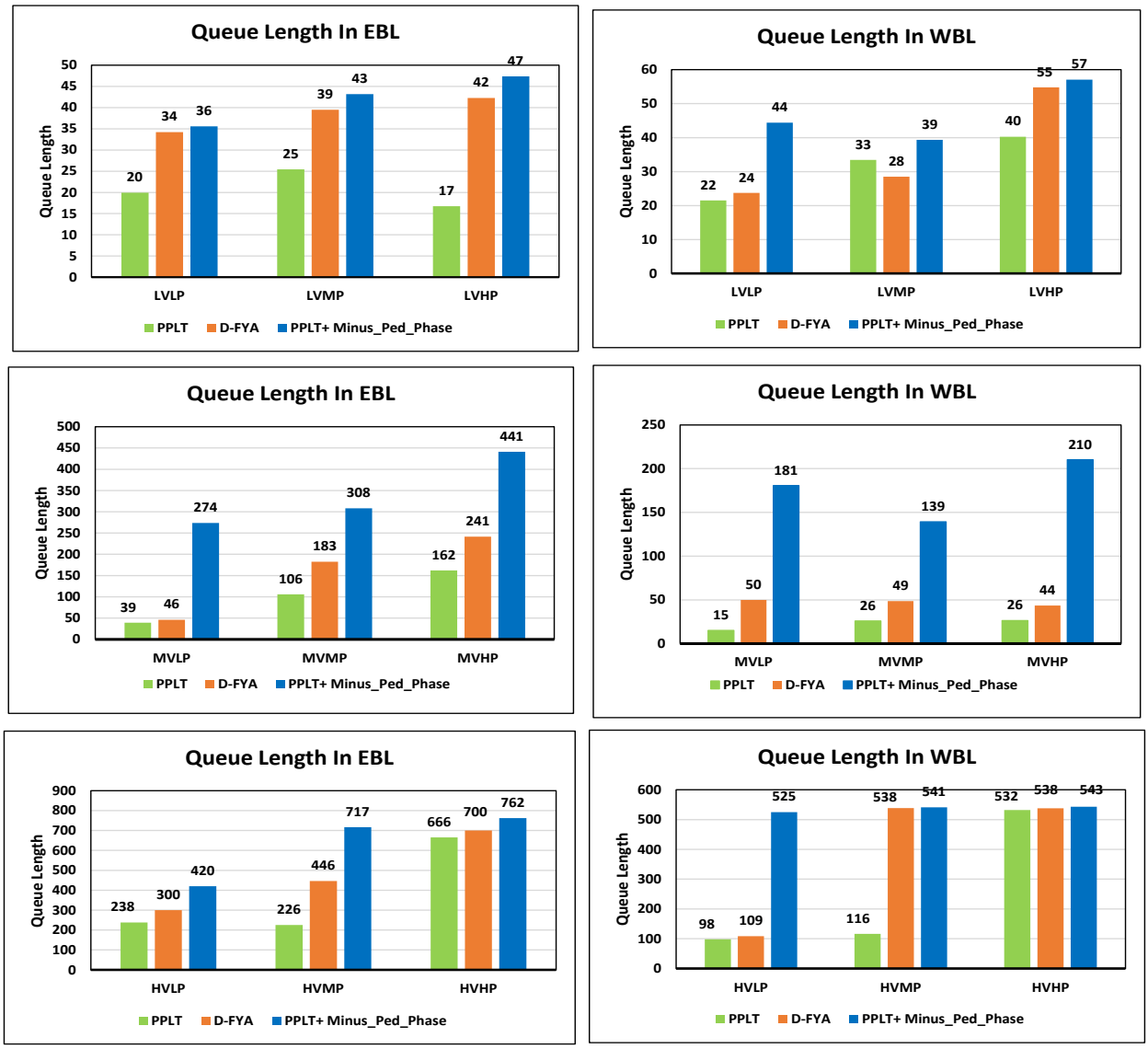


Figure 4.8 The mainline left-turn queue length comparison under various scenarios

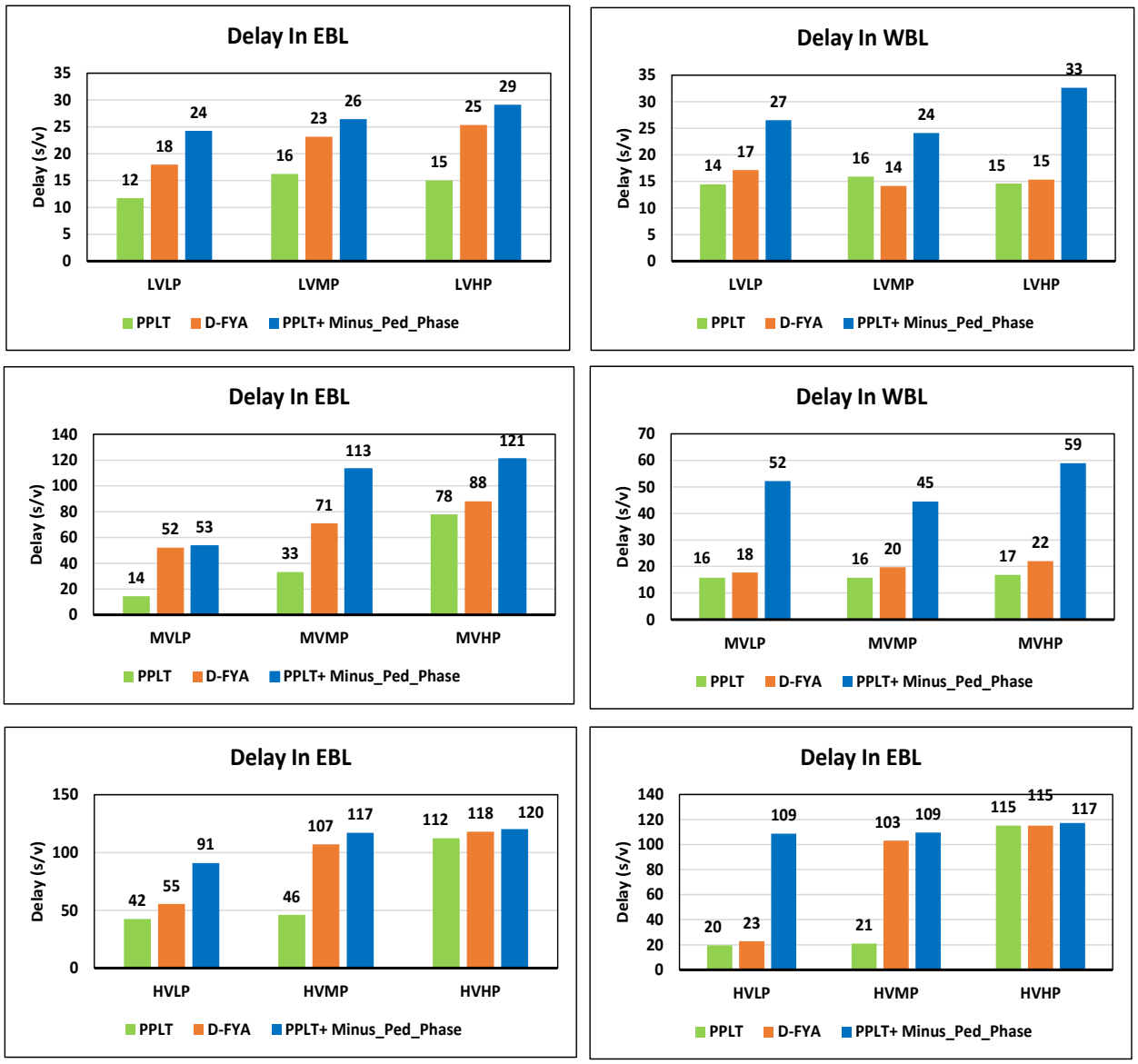


Figure 4.9 The mainline left-turn delay comparison under various scenarios

4.5 Summery

In this study, we develop a novel dynamic flash yellow arrow (D-FYA) mechanism to leverage the permissive left-turn capacity and crossing pedestrians' safety based on pedestrian tracking technologies. The research outcome is to address the reported potential safety hazards after the flash yellow arrow (FYA) permissive left-turn strategy is widely deployed. Through a novel "emulation-in-the-field" traffic signal control framework, we verified the resilience of the proposed D-FYA algorithm to random pedestrian behaviors and mitigations to inaccurate

pedestrian detections. In addition, in a controlled simulation environment, we further evaluated all three permissive left-turn strategies: protected-permissive left turn (PPLT), D-FYA, and PPLT with minus-pedestrian-phase. We concluded that the proposed D-FYA-based pedestrian tracking would be more efficient than the PPLT with a minus-pedestrian phase. At the same time, it can effectively solve the issue of pedestrian safety. It was also found that when the opposing through traffic became highly, all three permissive left-turn strategies degraded to the protected-only control strategy, leading to high delays and long queues.

In the future, we plan to introduce more features into the D-FYA strategy, considering the concurrent crossing of pedestrians and the opposing through traffic. As revealed in the experiment, it would be better to dynamically cancel and recover the FYA according to the volume of opposing through traffic. It may reduce the possibility of collisions between left-turn vehicles and opposing through vehicles.

Acknowledgment

This study is a part of a research project titled "*Using LIDAR sensors to study pedestrian behaviors and safety improvement at signalized intersections*", sponsored by the National Institute for Transportation and Communities, a national university transportation center hosted at Portland State University. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors. They do not necessarily reflect the official views or policies of the above organizations, nor do the contents constitute a standard, specification, or regulation of these organizations.

Data Availability statement

Some or all data, models, or code generated or used during the study are available in a repository online in accordance with funder data retention policies. The simulation model for the 2nd case study can be downloaded at https://github.com/pflee2002/D-FYA-VISSIM-Model

CHAPTER V

APPLICATION OF CONNECTED VEHICLE TRAJECTORY DATA IN TRAVEL DEMAND FORECASTING

5.1 Overview

When it comes to the question of strategic planning of an equitable transportation network, a traditional four-step planning method has been used for a long time (fig 5.1). Among these four steps, dynamic Origin-Destination (OD) estimation conducted in trip distribution is vital while planning and predicting future networks. An accurate prediction of these OD metrics is essential to ensure the optimal use of available resources in the transportation network. This prediction gives the road user a brief idea about the route choice along with helping the transportation professionals in the implantation of technology to envisage future needs within the system.

Traditionally, the four-step transportation model works, as shown in figure 5.1 where traffic assignment is the final step for the network design.

However, a new approach is presented in this research objective where the OD matric of the vehicles in the network using the connected vehicle's trajectory data is going to be predicted. In other words, by using traffic assignment results and network performance, the path flow distribution of the entire network will be predicted. Therefore, an innovative framework for travel demand forecasting is going to be developed. The goal of this chapter is to explore an innovative framework to forecast the travel demand in the DFW area. The current

practice of travel demand forecasting in DFW is the classic "four-step" method based on household surveys and traffic counts. The emerging new traffic data, like the connected vehicle trajectories, bring both opportunities and challenges. The novel data sets reveal much more information about the traveler than before and pave the way for enhanced accurate and high-fidelity travel demand forecasts.

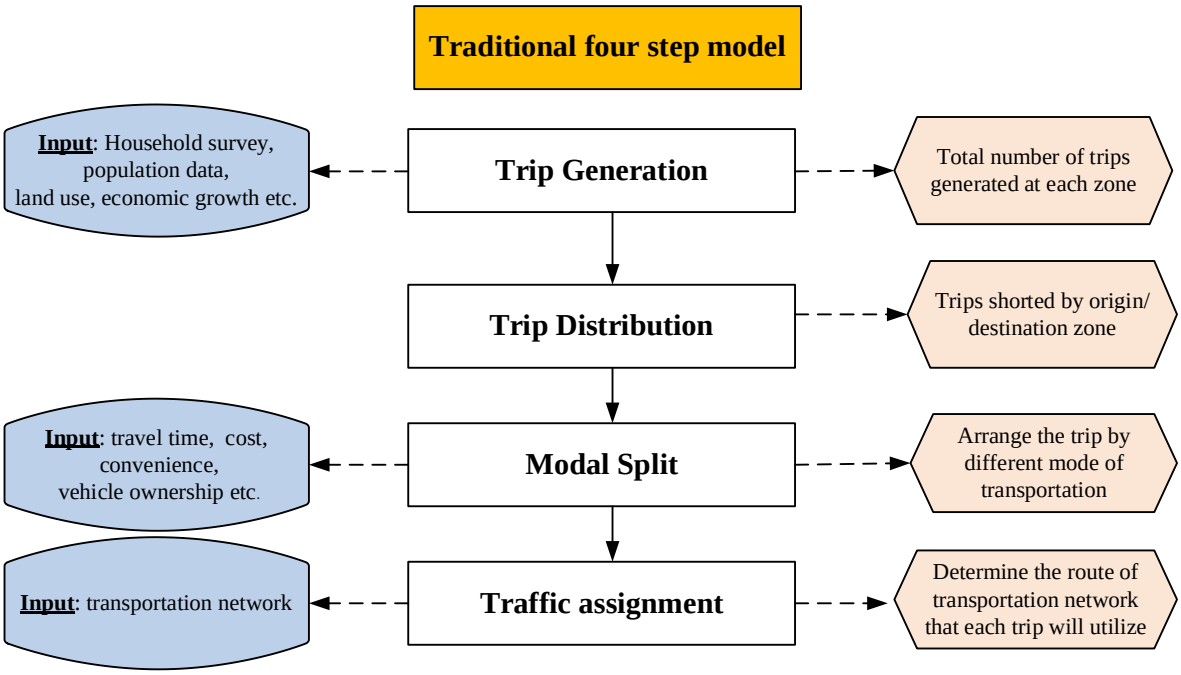


Figure 5.1 Traditional four step transportation model

On the other hand, the traditional travel demand forecast cannot take advantage of the total horsepower of such data sets. The inconsistency of various data sets and heterogeneous data quality are other issues of fusing the emerging traffic data with the traditional ones. This chapter will explore the innovative framework of travel demand forecasting based on connected vehicle data using state-of-the-art big data analytics and high-performance computing to address these issues.

5.2 Formulation for the path flow estimation model :

A nonlinear optimization formulation for path flow estimation is proposed in this section. The objective is to find the minimum Mean Square Error value for the 100% link counts and estimated path flow value. Below table 5.1 shows the parameters and variables for the nonlinear optimization function.

Table 5.1 Parameters and variables for the nonlinear optimization formulation

Notations for Non- Linear Optimization Formulation	
A	Set of road link nodes
P	Set of all possible path from origin to destination
$c_{(i,j)}$	100% link count (known Link volume)
w_p	Observed (partial) Wejo path flow
p	Volume of concurrent crossing pedestrian (ped per hour)
$t_{(i,j)}$	Link travel time on each link
$t'_{(i,j)}$	Free flow travel time on each link
$m_{(i,j,p)}$	Node to path mapping (map-matching)
$cap_{(i,j)}$	Capacity of each link
x_p	Decision Variable-(adjustment factors for each path)

Non- Linear Optimization Formulation Using Mean Square Error Method

Objective function:

$$\min Z = \sum_{(i,j)\epsilon A}(c_{(i,j)} - \sum_{p\epsilon P} w_p x_p m_{(i,j,p)})^2 \quad (1)$$

In eq (1) , the mean square error formulation is presented. Here, w_p represents the wejo path flow value which varies from 3 to 5% when comparing with the actual path flow values. x_p here is our decision variable which indicates the adjustment factor for each path.

Subjected to:

BPR function $\sum_{(i,j)\epsilon A}(t'_{(i,j)} \left[1 + \alpha \left(\frac{\sum_{p\epsilon P} w_p x_p m_{(i,j,p)}}{cap_{(i,j)}}\right)^{\beta}\right]) \quad = t_{((i,j)\epsilon A)}$ (2)

$1 \leq x_p \leq 33 \;\; for\; p\epsilon P$ (3)

Eq (2) is the representation of the standard BPR (Bureau of Public Road) function, BPR function is used to compute link travel time for each road link. Here, $t'_{(i,j)}$ indicated the free flow travel time for each link. The term $w_p x_p m_{(i,j,p)}$ indicates the estimated path flow value and $t_{((i,j)\epsilon A)}$ indicates the link travel time for each road link.

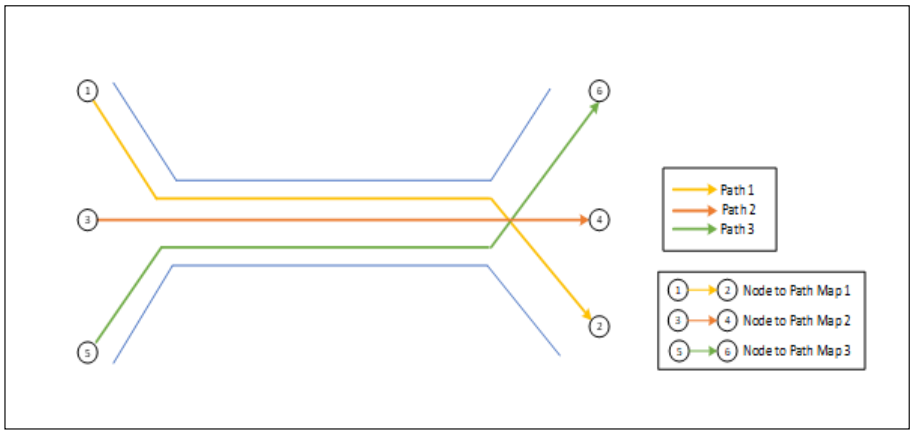


Figure 5.2 Node to path mapping $m_{(i,j,p)}$

Eq (3) is used to restrict the limit if the adjustment factor x_p. The limit used here is obtained by observation. Fig 5.2 show how node to path mapping $m_{(i,j,p)}$ is conducted in this research.

5.3 Numerical Experiment for Model Validation

In this section, the results collected from different sets of experiments are presented. The experiments are set to validate the proposed path flow estimation model. The experiment is designed for two different sets. In the initial set of experiments, only one O-D pair is considered.

Again, in experiment I, three different path flow condition is considered. The three different conditions are as follows:

I. Low path flow

II. Medium path flow

III. High path flow

In the second experiment, two different O-D pair is selected to check the sensitivity of the model under multiple O-D situations.

5.3.1 Formulation Validation Experiment I: Considering One O-D Pair

5.3.1.1 Scenario I: Low path flow condition

For validating the proposed path flow estimation model, the first experiment is conducted in a low path flow value. The below fig 5.3 (a) shows the 100% link counts for each connected node. And fig 5.3(b) shows the path flow under low path flow conditions for one selected O-D pair node 1 to node 9. To be mentioned here, or the selected O-D pair, six possible paths are detected. The highest expected path flow value for this experiment is 349 veh/hr.

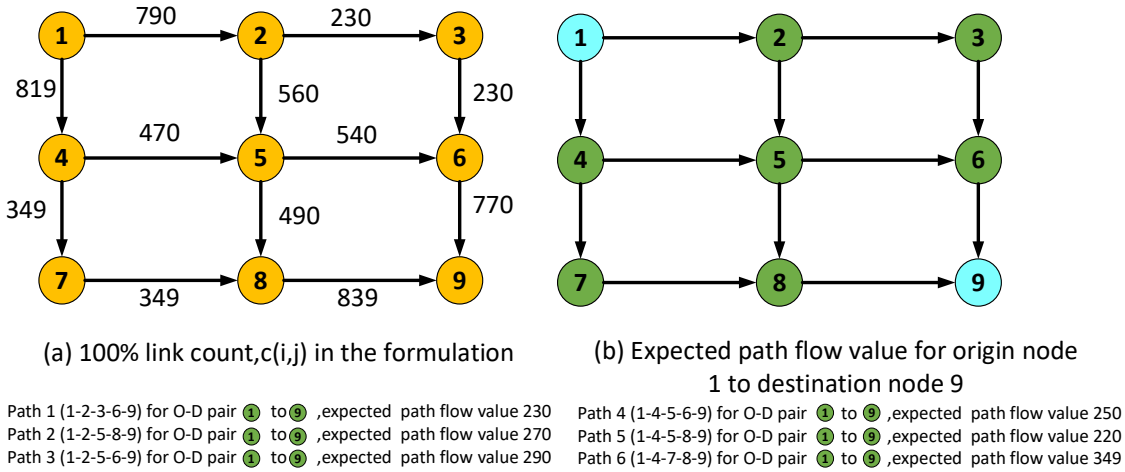


Figure 5.3 (a) Tropology of the network with 100% link counts for scenario I (b) Real O-D matrix with expected path flow value.

The below table 5.2 and fig 5.5 shows the results from the scenario.

Table 5.2 Scenario I: Low path flow condition

Path ID	Wejo %	Expected Path Flow	Estimated Path Flow	% Difference
Path 1	3.5	230	231	0.43%
Path 2	4.5	270	306.0848835	13.36%
Path 3	4.5	290	256.9150883	11.41%
Path 4	5	250	278.0848836	11.23%
Path 5	3	220	188.9151416	14.13%
Path 6	4	349	341	2.29%

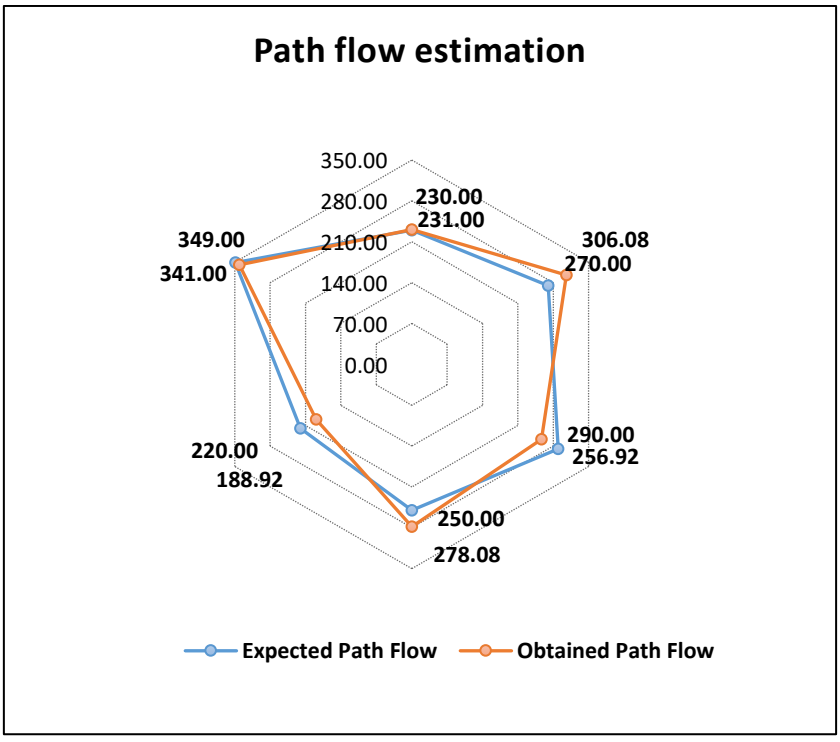
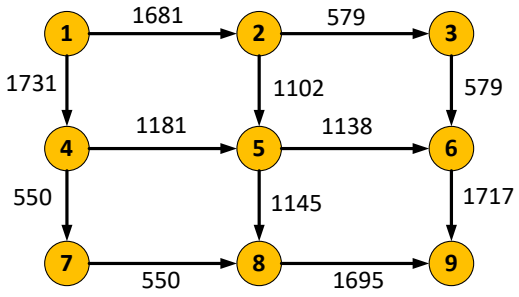


Figure 5.4 Rader plot showing the expected and estimated path flow value

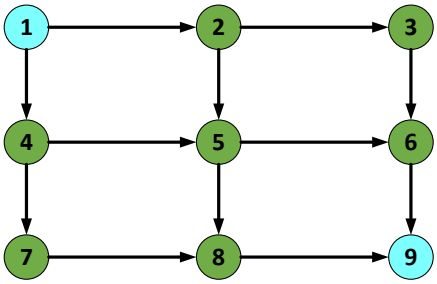
The result indicates that difference between is the expected path flow value and estimated path flow from the model is very satisfying . The highest difference between these two criteria is found 14.13% where the expected path flow is 220 veh/hr and estimated value is 189 veh/hr.

5.3.1.2 Scenario II: Medium path flow condition

For scenario II, medium path flow condition is considered. Fig 5.5(a) shows 100% link counts for each links. In fig 5.5 (b),the expected path flow value for origin 1 to destination 9 is showed. The highest value of path flow in this condition is 618 veh/hr.



(a) 100% link count,c(i,j) in the formulation



(b) Expected path flow value for origin node 1 to destination node 9

Path 1 (1-2-3-6-9) for O-D pair 1 to 9 ,expected path flow value 579
Path 2 (1-2-5-8-9) for O-D pair 1 to 9 ,expected path flow value 582
Path 3 (1-2-5-6-9) for O-D pair 1 to 9 ,expected path flow value 520

Path 4 (1-2-3-6-9) for O-D pair 1 to 9 ,expected path flow value 618
Path 5 (1-2-5-8-9) for O-D pair 1 to 9 ,expected path flow value 563
Path 6 (1-2-5-6-9) for O-D pair 1 to 9 ,expected path flow value 550

Figure 5.5 (a) Tropology of the network with 100% link counts for scenario II (b) Real O-D matrix with expected path flow value.

The below table 5.3 and fig 5.6 shows the results from the scenario II.

Table 5.3 Scenario II: Medium path flow condition

Path ID	Wejo %	Expected Path Flow	Estimated Path Flow	% Difference
Path 1	3.5	579	567.00	2.07%
Path 2	4.5	582	461.00	20.79%
Path 3	4.5	520	525.00	1.35%
Path 4	5	618	612.00	0.97%
Path 5	3	563	561.00	0.36%
Path 6	4	550	550	0.00%

The result indicates that difference between is the expected path flow value and estimated path flow from the model is very close.

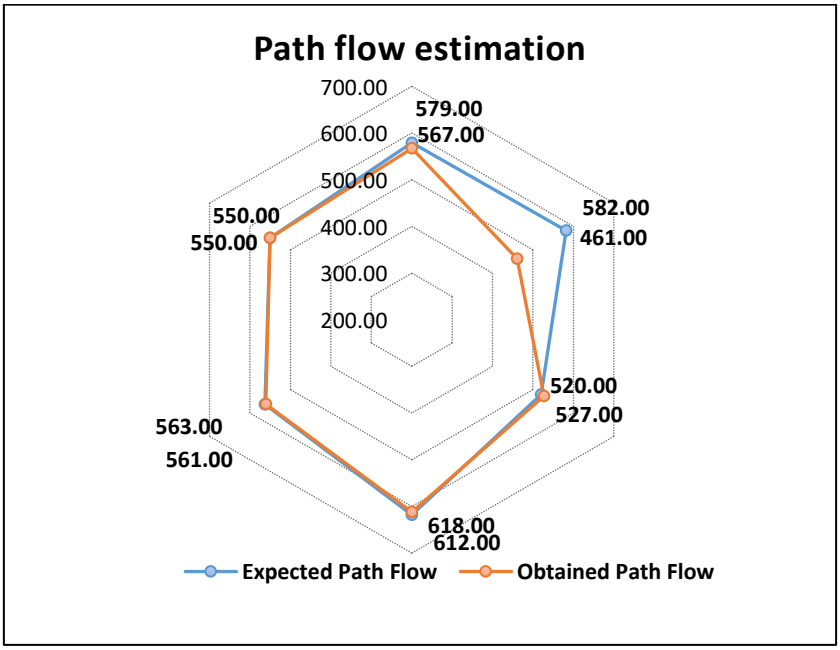


Figure 5.6 Rader plot showing the expected and estimated path flow value

The highest difference between these two criteria is found 20.79% where the expected path flow is 582 veh/hr and estimated value is 461 veh/hr. The lowest difference is found as small as 0% where estimated value of path flow matches the expected path flow value.

5.3.1.3 Scenario III: High path flow condition

This third experiment is conducted for high path flow value. The below fig 5.7 (a) shows the 100% link counts for each connected node. And fig 5.7 (b) shows the path flow in high path flow condition for one selected O-D pair node 1 to node 9. The highest expected path flow value for this scenario is 690 veh/hr.

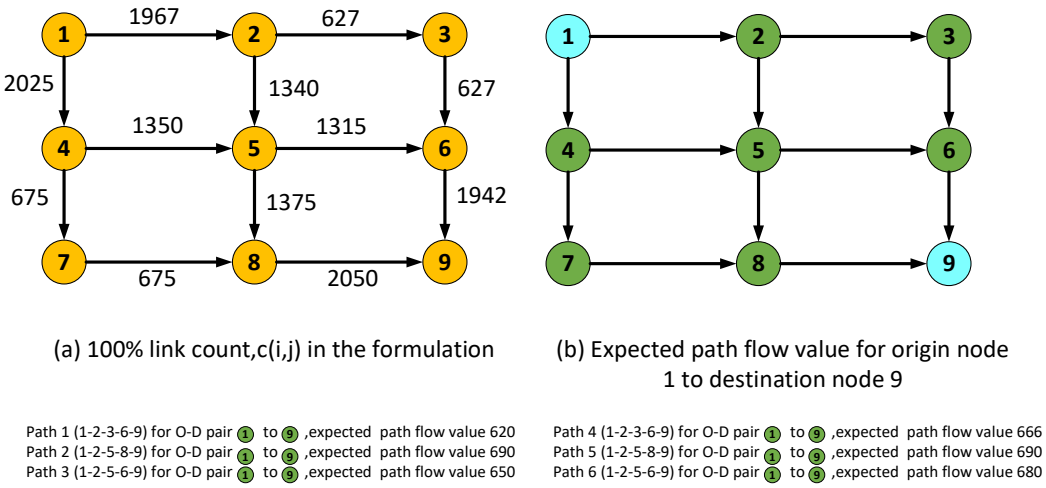


Figure 5.7 (a) Tropology of the network with 100% link counts for scenario III (b) Real O-D matrix with expected path flow value.

The below table 5.4 and fig 5.7 shows the results from the scenario.

Table 5.4 Scenario III: High path flow condition

Path ID	Wejo %	Expected Path Flow	Estimated Path Flow	% Difference
Path 1	3.0	620	627	1.13%
Path 2	4.0	690	749.0268568	8.55%
Path 3	5.0	650	590.9731432	9.08%
Path 4	4.0	666	724.0268568	8.71%
Path 5	3.0	690	625.9733822	9.28%
Path 6	4.0	680	675	0.74%

The result indicates that difference between is the expected path flow value and estimated path flow from the model is very satisfying . The highest difference between these two criteria is found 9.28% where the expected path flow is 690 veh/hr and estimated value is 626 veh/hr.

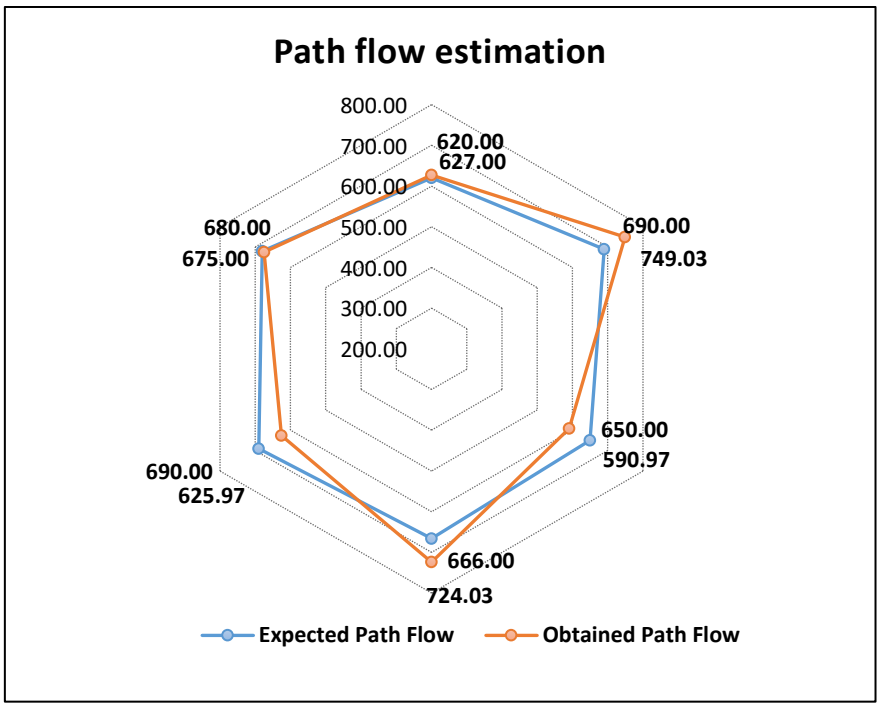


Figure 5.8 Rader plot showing the expected and estimated path flow value

From these three experiments it can be observed that at the higher path flow condition, the estimated path flow is the closer to the expected value.

5.3.2 Formulation Validation Experiment II: Considering Multiple O-D Pair

In this experiment, instead of using only one O-D pair, multiple O-D pair is considered. As show in the below figure, here two O-D pair is considered, where node 2 and 1 has respective destination node 9 and 8.

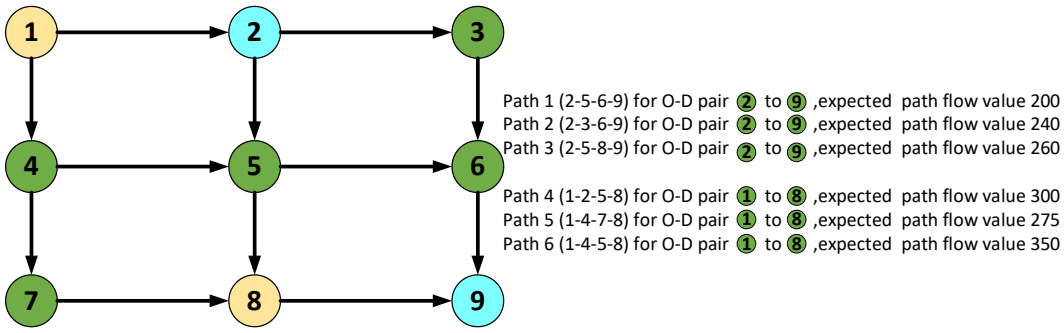


Figure 5.9 Numerical experiment for multiple O-D pair

Table 5.5 Experiment II: Multiple O-D pair condition

O-D pair	Path ID	Wejo %	Expected Path Flow	Estimated Path Flow	% Difference
2-9	Path 1	3.0	200	199.99	0.00%
	Path 2	3.5	240	228.57	4.76%
	Path 3	3.0	260	266.67	2.56%
1-8	Path 4	4.0	300	300	0.00%
	Path 5	4.5	275	266.67	3.03%
	Path 6	5.0	350	360	2.86%

The table 5.5 shows the outcome of the proposed path flow model. The result shows that with multiple O-D pair in the network, the result is still very satisfying. The maximum difference between expected and estimated path flow value is found for O-D pair 2-9 which is 4.76% and minimum value found is 0%.

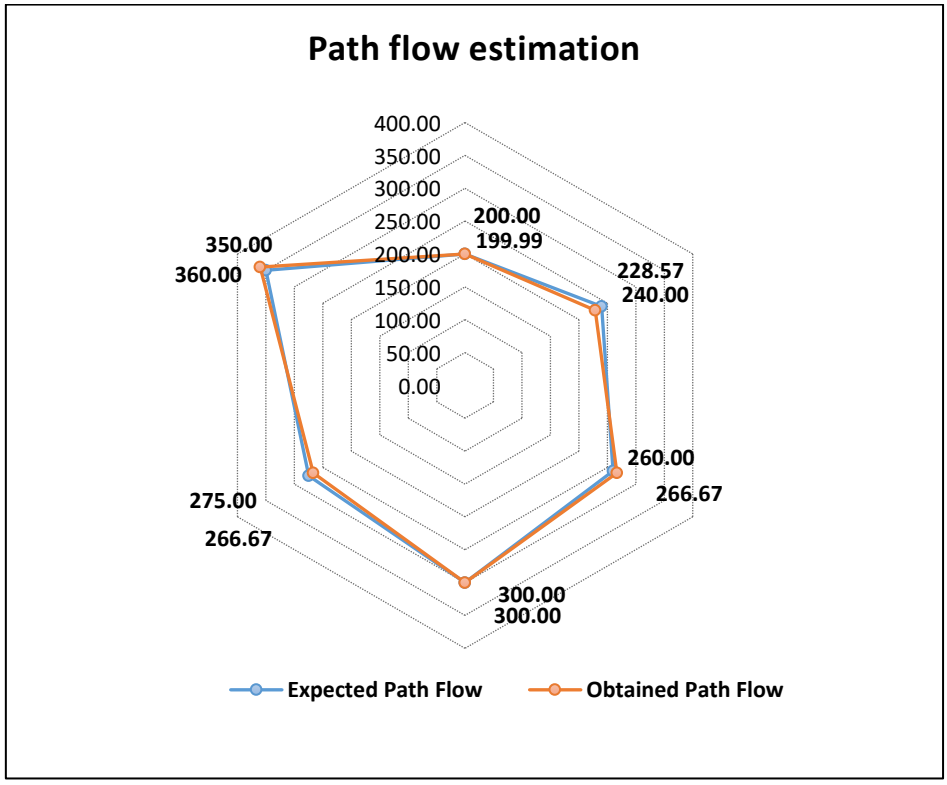


Figure 5.10 Rader plot showing the expected and estimated path flow value

5.4 Summary

This chapter is focused on developing an innovative formulation for path flow estimation in travel demand forecasting based on connected vehicle data (Wejo data). One Non-Linear Optimization Formulation Using Mean Square Error Method is presented in the chapter where the objective function is to find the minimum difference between 100% link counts and estimated path flow value. The wejo data is 3 to 5% of the actual path flow value. Three numerical experiments is conducted for low expected path flow value, medium expected path flow value, and high expected path flow value. The result obtained from the experiment

is very promising. In low path flow condition, the highest difference between expected and estimated path flow value is 14.13%, in medium path flow condition, the highest difference is found 20.79% and in high path flow condition the highest difference is found 9.28%.

These experiments are done considering one O-D pair. Another experiment is conducted in the chapter where multiple O-D pair is used instead of using one O-D pair. The result shows that the maximum difference between expected and estimated path flow values is found for O-D pair 2-9, which is 4.76%, and the minimum value found is 0%. Therefore, applying multiple O-D into the proposed model is also very satisfying.

In the future, a computational graph for path flow estimation will be developed for large-scale network data. The algorithm will check the applicability of the proposed model in big data-driven environment to estimate traffic assignment and travel demand forecasting.

CHAPTER VI

CONCLUSION AND FUTURE WORK

6.1 Overview of the Research

Increasing population and demand is creating need for a sustainable and equitable transportation system over time. To address this issue, in this research work, a sustainable and equitable transportation system for connected vehicle and big data driven environment. This research proposed a sustainability and equitable of transportation system by addressing intersection design in mixed traffic condition, pedestrian safety at inter section by proposing a tracking-based dynamic flash yellow arrow (D-FYA) strategy for permissive left-turn vehicles and finally analyzing the network performance within the big data driven environment. The summary of the findings from each of the three research goals are discussed in this section.

6.2 Summary of Congestion-Aware Heterogeneous Connected Automated Vehicles Cooperative Scheduling Problems at Intersections

The competition of CAV green request becomes more essential if the CAV penetration rate increases. CAV request at the intersection also needs to be overcome the possible interruption of background traffic in mixed traffic condition.

6.2.1 Conclusion

This research proposed a new method to find the solution of the congestion aware CAV scheduling problem at intersection. This solution also works on balancing the mobility of traffic when an active CAV green request is placed. A mixed-integer linear programming based on

the phase-time network and space-time network is formulated to provide a theoretical foundation for this problem. We also presented a “resilient phase-time network” based on the original phase time network. The resilient phase-time network will guarantee a mathematically feasible solution when not all CAV green requests can be served due to competing time windows. From the outcome of the resilient phase-time network, we can tell which request is declined. By contrast, the MILP solver will leave little information for improvement if this problem is not solvable in the original phase-time networks.

6.2.2 Future Recommendation

Only the isolated intersection with a limited percentage of priority requests is the subject of this study. In the future, the model's sensitivity can be verified by optimizing a long corridor with numerous connected intersections. The long corridor's results can be used in extensive network modeling. In summary, it is possible to investigate whether the suggested model can be used for actual development.

6.3 Summary of Dynamic Flash Yellow Arrow Along with Pedestrian Detection Technique to Improve Pedestrian Safety

In this study, we develop a novel dynamic flash yellow arrow (D-FYA) mechanism to leverage the permissive left-turn capacity and crossing pedestrians’ safety based on pedestrian tracking technologies.

6.3.1 Conclusion

The research outcome is to address the reported potential safety hazards after the flash yellow arrow (FYA) permissive left-turn strategy is widely deployed. Through a novel “emulation-in-the-field” traffic signal control framework, we verified the resilience of the proposed D-FYA algorithm to random pedestrian behaviors and mitigations to inaccurate pedestrian detections. In addition, in a controlled simulation environment, we further evaluated

all three permissive left-turn strategies: protected-permissive left turn (PPLT), D-FYA, and PPLT with minus-pedestrian-phase. We concluded that the proposed D-FYA-based pedestrian tracking would be more efficient than the PPLT with a minus-pedestrian phase. At the same time, it can effectively solve the issue of pedestrian safety. It was also found that when the opposing through traffic became highly, all three permissive left-turn strategies degraded to the protected-only control strategy, leading to high delays and long queues.

6.3.2 Future Recommendation

In the future, we plan to introduce more features into the D-FYA strategy, considering the concurrent crossing of pedestrians and the opposing through traffic. As revealed in the experiment, it would be better to dynamically cancel and recover the FYA according to the volume of opposing through traffic. It may reduce the possibility of collisions between left-turn vehicles and opposing through vehicles.

6.4 Summary of Application of Connected Vehicle Trajectory Data in Travel Demand Forecasting

This study is focused on developing an innovative formulation for path flow estimation in travel demand forecasting based on connected vehicle data (Wejo data).

6.4.1 Conclusion

One Non-Linear Optimization Formulation Using Mean Square Error Method is presented in the chapter where the objective function is to find the minimum difference between 100% link counts and estimated path flow value. The wejo data is 3 to 5% of the actual path flow value. Three numerical experiments is conducted for low expected path flow value, medium expected path flow value, and high expected path flow value. The result obtained from the experiment is very promising. In low path flow condition, the highest difference between expected and estimated path flow value is 14.13%, in medium path flow condition, the highest

difference is found 20.79% and in high path flow condition the highest difference is found 9.28%.

These experiments are done considering one O-D pair. Another experiment is conducted in the chapter where multiple O-D pair is used instead of using one O-D pair. The result shows that the maximum difference between expected and estimated path flow values is found for O-D pair 2-9, which is 4.76%, and the minimum value found is 0%. Therefore, applying multiple O-D into the proposed model is also very satisfying.

6.4.2 Future Recommendation

In the future, a computational graph for path flow estimation will be developed for large-scale network data. The algorithm will check the applicability of the proposed model in big data-driven environment to estimate traffic assignment and travel demand forecasting.

REFERENCE

[1] Q. Guo, L. Li, and X. J. Ban, "Urban traffic signal control with connected and automated vehicles: A survey," *Transportation research part C: emerging technologies,* vol. 101, pp. 313-334, 2019.

[2] C. Yu, Y. Feng, H. X. Liu, W. Ma, and X. Yang, "Integrated optimization of traffic signals and vehicle trajectories at isolated urban intersections," *Transportation Research Part B: Methodological,* vol. 112, pp. 89-112, 2018.

[3] W. Knight, "Car-to-car communication," *Technology Review,* vol. 118, no. 2, pp. 38-39, 2015.

[4] S. I. Guler, M. Menendez, and L. Meier, "Using connected vehicle technology to improve the efficiency of intersections," *Transportation Research Part C: Emerging Technologies,* vol. 46, pp. 121-131, 2014.

[5] W. Sun, J. Zheng, and H. X. Liu, "A capacity maximization scheme for intersection management with automated vehicles," *Transportation research procedia,* vol. 23, pp. 121-136, 2017.

[6] B. Xu *et al.*, "Distributed conflict-free cooperation for multiple connected vehicles at unsignalized intersections," *Transportation Research Part C: Emerging Technologies,* vol. 93, pp. 322-334, 2018.

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

[8] Y. Feng, K. L. Head, S. Khoshmagham, and M. Zamanipour, "A real-time adaptive signal control in a connected vehicle environment," *Transportation Research Part C: Emerging Technologies,* vol. 55, pp. 460-473, 2015.

[9] Y. Feng, C. Yu, and H. X. Liu, "Spatiotemporal intersection control in a connected and automated vehicle environment," *Transportation Research Part C: Emerging Technologies,* vol. 89, pp. 364-383, 2018.

[10] X. Kong, X. Song, F. Xia, H. Guo, J. Wang, and A. Tolba, "LoTAD: Long-term traffic anomaly detection based on crowdsourced bus trajectory data," *World Wide Web,* vol. 21, no. 3, pp. 825-847, 2018.

[11] C. Priemer and B. Friedrich, "A decentralized adaptive traffic signal control using V2I communication data," in *2009 12th International IEEE Conference on Intelligent Transportation Systems,* 2009: IEEE, pp. 1-6.

[12] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *Journal of artificial intelligence research,* vol. 31, pp. 591-656, 2008.

[13] T.-C. Au, S. Zhang, and P. Stone, "Semi-autonomous intersection management," in *AAMAS*, 2014, pp. 1451-1452.

[14] M. Hausknecht, T.-C. Au, and P. Stone, "Autonomous intersection management: Multi-intersection optimization," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011: IEEE, pp. 4581-4586.

[15] J. Datesh, W. T. Scherer, and B. L. Smith, "Using k-means clustering to improve traffic signal efficacy in an IntelliDrive SM environment," in *2011 IEEE Forum on Integrated and Sustainable Transportation Systems*, 2011: IEEE, pp. 122-127.

[16] J. Lee, B. Park, and I. Yun, "Cumulative travel-time responsive real-time intersection control algorithm in the connected vehicle environment," *Journal of Transportation Engineering*, vol. 139, no. 10, pp. 1020-1029, 2013.

[17] R. Venkatanarayana, H. Park, B. L. Smith, C. Skerrit Jr, and N. W. Ruhter, "Application of IntelliDrive℠ to address oversaturated conditions on arterials," 2011.

[18] P. G. Michalopoulos and G. Stephanopoulos, "Oversaturated signal systems with queue length constraints—I: Single intersection," *Transportation Research*, vol. 11, no. 6, pp. 413-421, 1977.

[19] M. Li, Y. Yin, W. B. Zhang, K. Zhou, and H. Nakamura, "Modeling and implementation of adaptive transit signal priority on actuated control systems," *Computer-Aided Civil and Infrastructure Engineering,* vol. 26, no. 4, pp. 270-284, 2011.

[20] H. Liu, A. Skabardonis, W.-b. Zhang, and M. Li, "Optimal detector location for bus signal priority," *Transportation Research Record*, vol. 1867, no. 1, pp. 144-150, 2004.

[21] X. Han, P. Li, R. Sikder, Z. Qiu, and A. Kim, "Development and evaluation of adaptive transit signal priority control with updated transit delay model," *Transportation Research Record*, vol. 2438, no. 1, pp. 45-54, 2014.

[22] E. Christofa, K. Ampountolas, and A. Skabardonis, "Arterial traffic signal optimization: A person-based approach," *Transportation Research Part C: Emerging Technologies*, vol. 66, pp. 27-47, 2016.

[23] W. Ma, W. Ni, L. Head, and J. Zhao, "Effective coordinated optimization model for transit priority control under arterial progression," *Transportation Research Record*, vol. 2366, no. 1, pp. 71-83, 2013.

[24] P. Kilambi, E. Ribnick, A. J. Joshi, O. Masoud, and N. Papanikolopoulos, "Estimating pedestrian counts in groups," *Computer Vision and Image Understanding*, vol. 110, no. 1, pp. 43-59, 2008.

[25] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008: IEEE, pp. 1-7.

[26] S. Yoshinaga, A. Shimada, and R.-i. Taniguchi, "Real-time people counting using blob descriptor," *Procedia-Social and Behavioral Sciences,* vol. 2, no. 1, pp. 143-152, 2010.

[27] A. B. Chan and N. Vasconcelos, "Bayesian poisson regression for crowd counting," in *2009 IEEE 12th international conference on computer vision*, 2009: IEEE, pp. 545-551.

[28] V. Bhuvaneshwar and P. B. Mirchandani, "Real-time detection of crossing pedestrians for traffic-adaptive signal control," in *Proceedings. The 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No. 04TH8749)*, 2004: IEEE, pp. 309-313.

[29] B. Li, Q. Yao, and K. Wang, "A review on vision-based pedestrian detection in intelligent transportation systems," in *Proceedings of 2012 9th IEEE international conference on networking, sensing and control*, 2012: IEEE, pp. 393-398.

[30] V. John, S. Tsuchizawa, Z. Liu, and S. Mita, "Fusion of thermal and visible cameras for the application of pedestrian detection," *Signal, Image and Video Processing,* vol. 11, no. 3, pp. 517-524, 2017.

[31] J. Baek, S. Hong, J. Kim, and E. Kim, "Efficient pedestrian detection at nighttime using a thermal camera," *Sensors,* vol. 17, no. 8, p. 1850, 2017.

[32] J. Kim, "Pedestrian Detection and Distance Estimation Using Thermal Camera in Night Time," in *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, 2019: IEEE, pp. 463-466.

[33] S. Kothuri, K. Nordback, A. Schrope, T. Phillips, and M. Figliozzi, "Bicycle and pedestrian counts at signalized intersections using existing infrastructure: opportunities and challenges," *Transportation research record,* vol. 2644, no. 1, pp. 11-18, 2017.

[34] H. Zhao and R. Shibasaki, "A novel system for tracking pedestrians using multiple single-row laser-range scanners," *IEEE Transactions on systems, man, and cybernetics-Part A: systems and humans,* vol. 35, no. 2, pp. 283-291, 2005.

[35] H. Zhao *et al.*, "Detection and tracking of moving objects at intersections using a network of laser scanners," *IEEE transactions on intelligent transportation systems,* vol. 13, no. 2, pp. 655-670, 2011.

[36] J. Zhao, Y. Li, H. Xu, and H. Liu, "Probabilistic prediction of pedestrian crossing intention using roadside LiDAR data," *IEEE Access,* vol. 7, pp. 93781-93790, 2019.

[37] J. Zhao, H. Xu, J. Wu, Y. Zheng, and H. Liu, "Trajectory tracking and prediction of pedestrian's crossing intention using roadside LiDAR," *IET Intelligent Transport Systems,* vol. 13, no. 5, pp. 789-795, 2019.

[38] B. Lv, R. Sun, H. Zhang, H. Xu, and R. Yue, "Automatic Vehicle-Pedestrian Conflict Identification With Trajectories of Road Users Extracted From Roadside LiDAR Sensors Using a Rule-Based Method," *IEEE Access,* vol. 7, pp. 161594-161606, 2019.

[39] J. Wu, H. Xu, Y. Sun, J. Zheng, and R. Yue, "Automatic background filtering method for roadside LiDAR data," *Transportation Research Record,* vol. 2672, no. 45, pp. 106-114, 2018.

[40] T. S. Combs, L. S. Sandt, M. P. Clamann, and N. C. McDonald, "Automated vehicles and pedestrian safety: exploring the promise and limits of pedestrian detection," *American journal of preventive medicine,* vol. 56, no. 1, pp. 1-7, 2019.

[41] A. P. Grassi, V. Frolov, and F. P. León, "Information fusion to detect and classify pedestrians using invariant features," *Information fusion,* vol. 12, no. 4, pp. 284-292, 2011.

[42] M. El Ansari, R. Lahmyed, and A. Trémeau, "A Hybrid Pedestrian Detection System based on Visible Images and LIDAR Data," in *VISIGRAPP (5: VISAPP)*, 2018, pp. 325-334.

[43] F. Bu, T. Le, X. Du, R. Vasudevan, and M. Johnson-Roberson, "Pedestrian Planar LiDAR Pose (PPLP) Network for Oriented Pedestrian Detection Based on Planar LiDAR and Monocular Images," *IEEE Robotics and Automation Letters,* vol. 5, no. 2, pp. 1626-1633, 2019.

[44] R. Soundrapandiyan and P. C. Mouli, "Adaptive pedestrian detection in infrared images using background subtraction and local thresholding," *Procedia Computer Science,* vol. 58, no. 1, pp. 706-713, 2015.

[45] S. Tang, M. Ye, C. Zhu, and Y. Liu, "Adaptive pedestrian detection using convolutional neural network with dynamically adjusted classifier," *Journal of Electronic Imaging,* vol. 26, no. 1, p. 013012, 2017.

[46] K. Manston, "The challenges of using radar for pedestrian detection," *Traffic Engineering & Control,* vol. 52, no. 7, 2011.

[47] Y. Ma, M. Chowdhury, M. Jeihani, and R. Fries, "Accelerated incident detection across transportation networks using vehicle kinetics and support vector machine in cooperation with infrastructure agents," *IET intelligent transport systems,* vol. 4, no. 4, pp. 328-337, 2010.

[48] R. Claes, T. Holvoet, and D. Weyns, "A decentralized approach for anticipatory vehicle routing using delegate multiagent systems," *IEEE Transactions on Intelligent Transportation Systems,* vol. 12, no. 2, pp. 364-373, 2011.

[49] J. D. Crabtree and N. Stamatiadis, "Dedicated short-range communications technology for freeway incident detection: Performance assessment based on traffic simulation data," *Transportation research record,* vol. 2000, no. 1, pp. 59-69, 2007.

[50] Y. He, M. Chowdhury, Y. Ma, and P. Pisu, "Merging mobility and energy vision with hybrid electric vehicles and vehicle infrastructure integration," *Energy Policy,* vol. 41, pp. 599-609, 2012.

[51] R. Long Cheu, H. Qi, and D.-H. Lee, "Mobile sensor and sample-based algorithm for freeway incident detection," *Transportation research record,* vol. 1811, no. 1, pp. 12-20, 2002.

[52] Y. Ma, M. Chowdhury, A. Sadek, and M. Jeihani, "Real-time highway traffic condition assessment framework using vehicle–infrastructure integration (VII) with artificial intelligence (AI)," *IEEE Transactions on Intelligent Transportation Systems,* vol. 10, no. 4, pp. 615-627, 2009.

[53] H. Qi, R. Cheu, and D. Lee, "Freeway incident detection using kinematic data from probe vehicles," in *9th World Congress on Intelligent Transport SystemsITS America, ITS Japan, ERTICO (Intelligent Transport Systems and Services-Europe)*, 2002.

[54] S. M. Khan, K. C. Dey, and M. Chowdhury, "Real-time traffic state estimation with connected vehicles," *IEEE Transactions on Intelligent Transportation Systems,* vol. 18, no. 7, pp. 1687-1699, 2017.

[55] H. C. Manual, "HCM2010," *Transportation Research Board, National Research Council, Washington, DC,* vol. 1207, 2010.

[56] T. Z. Qiu, X.-Y. Lu, A. H. Chow, and S. E. Shladover, "Estimation of freeway traffic density with loop detector and probe vehicle data," *Transportation Research Record,* vol. 2178, no. 1, pp. 21-29, 2010.

[57] S. Hernandez, A. Tok, and S. G. Ritchie, "Density Estimation Using Inductive Loop Signature Based Vehicle Re-Identification and Classification," 2013.

[58] T. Choe, A. Skabardonis, and P. Varaiya, "Freeway performance measurement system: operational analysis tool," *Transportation research record,* vol. 1811, no. 1, pp. 67-75, 2002.

[59] R. Mao and G. Mao, "Road traffic density estimation in vehicular networks," in *2013 IEEE Wireless Communications and Networking Conference (WCNC)*, 2013: IEEE, pp. 4653-4658.

[60] W. Shi and Y. Liu, "Real-time urban traffic monitoring with global positioning system-equipped vehicles," *IET intelligent transport systems,* vol. 4, no. 2, pp. 113-120, 2010.

[61] S. Tao, V. Manolopoulos, S. Rodriguez Duenas, and A. Rusu, "Real-time urban traffic state estimation with A-GPS mobile phones as probes," *Journal of Transportation Technologies,* vol. 2, no. 1, pp. 22-31, 2012.

[62] T. Seo, T. Kusakabe, and Y. Asakura, "Traffic state estimation method using probe vehicles equipped with spacing measurement system," in *International Symposium on Recent Advances in Transport Modelling*, 2013.

[63] D. Ni, "Determining traffic-flow characteristics by definition for application in ITS," *IEEE Transactions on Intelligent Transportation Systems,* vol. 8, no. 2, pp. 181-187, 2007.

[64] A. Ramezani, B. Moshiri, B. Abdulhai, and A. Kian, "Estimation of free flow speed and critical density in a segmented freeway using missing data and Monte Carlo-based expectation maximisation algorithm," *IET control theory & applications,* vol. 5, no. 1, pp. 123-130, 2011.

[65] V. Tyagi, S. Kalyanaraman, and R. Krishnapuram, "Vehicular traffic density state estimation based on cumulative road acoustics," *IEEE Transactions on Intelligent Transportation Systems,* vol. 13, no. 3, pp. 1156-1166, 2012.

[66] A. Anand, G. Ramadurai, and L. Vanajakshi, "Data fusion-based traffic density estimation and prediction," *Journal of Intelligent Transportation Systems,* vol. 18, no. 4, pp. 367-378, 2014.

[67] L. Yu, Z. Wang, and Q. Shi, "PEMS-based approach to developing and evaluating driving cycles for air quality assessment," Center for Transportation Training and Research, Texas Southern University, 2010.

[68] J. Barrachina *et al.*, "V2X-d: A vehicular density estimation system that combines V2V and V2I communications," in *2013 IFIP Wireless Days (WD)*, 2013: IEEE, pp. 1-6.

[69] M. D. Venkata, M. M. Pai, R. M. Pai, and J. Mouzna, "Traffic monitoring and routing in VANETs—A cluster based approach," in *2011 11th international conference on ITS telecommunications*, 2011: IEEE, pp. 27-32.

[70] N. Caceres, L. M. Romero, F. G. Benitez, and J. M. del Castillo, "Traffic flow estimation models using cellular phone data," *IEEE Transactions on Intelligent Transportation Systems,* vol. 13, no. 3, pp. 1430-1441, 2012.

[71] X. Yang, Y. Lu, and W. Hao, "Origin-destination estimation using probe vehicle trajectory and link counts," *Journal of Advanced Transportation,* vol. 2017, 2017.

[72] Y. Nie, H.-M. Zhang, and W. Recker, "Inferring origin–destination trip matrices with a decoupled GLS path flow estimator," *Transportation Research Part B: Methodological,* vol. 39, no. 6, pp. 497-518, 2005.

[73] P. Li, P. Mirchandani, and X. Zhou, "Solving simultaneous route guidance and traffic signal optimization problem using space-phase-time hypernetwork," *Transportation Research Part B: Methodological,* vol. 81, Part 1, pp. 103-130, 11// 2015, doi: https://doi.org/10.1016/j.trb.2015.08.011.

[74] M. E. Pitstick, "Measuring delay and simulating performance at isolated signalised intersections using cumulative curves," presented at the Trasnsportation Research Board, 1989.

[75] D. A. Noyce, A. R. Bill, and M. A. Knodler Jr, "Evaluation of the Flashing Yellow Arrow (FYA) Permissive Left-Turn in Shared Yellow Signal Sections," 2014.

APPENDIX A

APPLING PROPORTION MATRICES OF LINK COUNT FOR PATH FLOW ESTIMATION

In Appendix A the path link proportion matrix used by Nie et al. *(72)* is applied in the proposed model for path flow estimation in section 5.2. For this experiment, we used the same network in section 5.3.1.1 (Scenario I: Low path flow condition). The below figure show shows the original network. To apply the proportion matrix method, we calculated the link flow proportion of each link. At the same time, the path flow proportion is also calculated. The below figure shows the original network before calculating the link flow and path flow proportion.

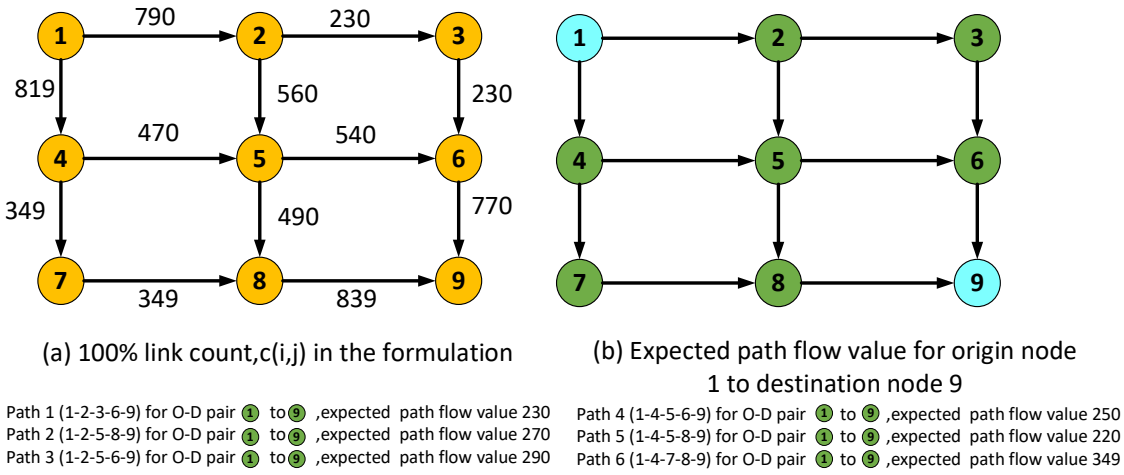


Figure 6.1 Path flow model using link proportion matrix

Table 6.1 shows the results of the path flow estimation model using the link and path proportion matrix.

Table 6.1 Path flow estimation using link and path proportion matrix

Path ID	Wejo %	Expected Path Flow	Estimated Path Flow	% Difference
Path 1	3.0	0.143	0.167	14.45%
Path 2	3.5	0.168	0.182	6.90%
Path 3	3.0	0.180	0.319	43.52%
Path 4	4.0	0.155	0.137	13.64%
Path 5	4.5	0.138	0.185	25.99%
Path 6	5.0	0.217	0.381	43.12%

The result of the experiment is represented in figure 6.2. From the result, it can be identified that when using the link and path flow proportion matrix instead of using the actual counts, the results are not as good as earlier.

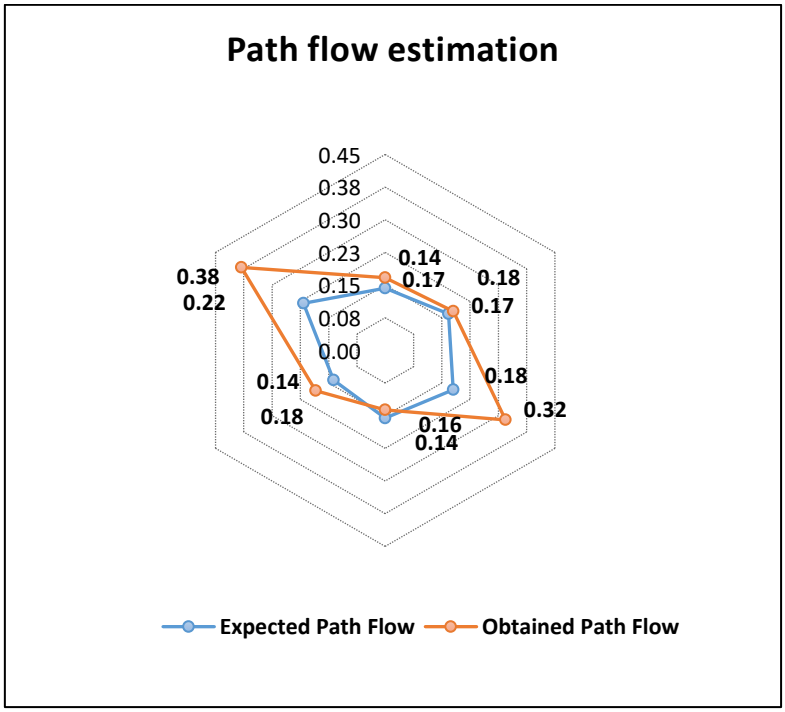


Figure 6.2 Rader plot showing the expected and estimated path flow value

We show the maximum difference between the estimated and expected path flow is as big as 43.52%. Such a big difference doesn't contribute to estimating the path flow value of the network using the wejo data. Therefore, we can conclude that when we try to estimate path flow value, using 100% link count is a more efficient way rather than using link proportion matrix.