A Kalman Filter-based Dynamic Model for Bus Travel Time Prediction

By

Abdulaziz Aldokhayel

Thesis submitted to the Faculty of Graduate and Postdoctoral Studies In partial fulfillment of the requirements for the degree of

Master of Applied Science (M.A.Sc)

in

Electrical and Computer Engineering

School of Electrical Engineering and Computer Science

Faculty of Engineering

University of Ottawa

uOttawa

Abstract

Urban areas are currently facing challenges in terms of traffic congestion due to city expansion and population increase. In some cases, physical solutions are limited. For example, in certain areas it is not possible to expand roads or build a new bridge. Therefore, making public transpiration (PT) affordable, more attractive and intelligent could be a potential solution for these challenges. Accuracy in bus running time and bus arrival time is a key component of making PT attractive to ridership. In this thesis, a dynamic model based on Kalman filter (KF) has been developed to predict bus running time and dwell time while taking into account real-time road incidents. The model uses historical data collected by Automatic Vehicle Location system (AVL) and Automatic Passenger Counters (APC) system. To predict the bus travel time, the model has two components of running time prediction (long and short distance prediction) and dwell time prediction. When the bus closes its doors before leaving a bus stop, the model predicts the travel time to all downstream bus stops. This is long distance prediction. The model will then update the prediction between the bus's current position and the upcoming bus stop based on real-time data from AVL. This is short distance prediction. Also, the model predicts the dwell time at each coming bus stop. As a result, the model reduces the difference between the predicted arrival time and the actual arrival time and provides a better understanding for the transit network which allows lead to have a good traffic management.

Key words: Intelligent Public Transit Systems, travel time prediction, bus arrival time prediction, Kalman filter

Acknowledgement

All praises and thanks are for the Almighty Allah for giving me the strength to pursue this thesis.

I would like to express my gratitude to my supervisor, Professor Hussein Mouftah, for his expertise, understanding, patience, and support during my journey. Professor Mouftah is always there whenever I need his advice and help. It has been an honour to be his student during the master's program.

A special thanks to my siblings for their unflagging love and endless support throughout my life. Also, I would like to thank my in-laws for their support; they are a blessing in my life.

I am grateful to the Saudi Arabian Cultural Bureau in Canada for their financial support during my studies here in Canada. Finally, I would like to thank my friends here in Canada and back home.

Dedication

To the pure spirit of my father, Suliman,

Who believes in me. May Allah bless your soul.

To my mother, Hailah,

Your prayers, love, and encouragement are a light upon my life.

To my lovely wife, Taghreed,

For your unconditional love, endless support, and the sacrifices you have made.

Contents

List of Figures

List of Tables

List of Acronyms

ANN	Artificial Neural Network
APC	Automatic Passenger Counters
APTS	Advanced Public Transportation Systems
AVLS/ AVL	Automatic Vehicle Location System
AVM	Automatic Vehicle Monitoring
CBDs	central business districts
DSS	Decision Support Systems
EAS	Exploitation Aid System
E-bus	Electric Bus
ESCAP	United Nations Economic and Social Commission for Asia and the Pacific
ESS	Exploitation Support System
EV	Electric Vehicles
FTA	Federal Transit Administration
GIS	Geographic Information Systems
GPS	Global Positioning System
GTHA	Greater Toronto and Hamilton Area
IoT	Internet of things
IPTS	Intelligent Public Transportation Systems
ITS	intelligent transportation system
KF	Kalman Filter
KNN	k-Nearest Neighbors

PT	Public Transportation
QoS	Quality of Service
RTPIS/TIS	Real Time Passenger information Systems
SOV	Single-Occupancy Vehicle
TBST	Total Bus Stop Time
TIS	Traveler Information Systems
UN	United Nations
WHO	World Health Organization

List of Symbols

a	Loop-gain
AAT	Actual arrival time of the previous bus
ADT	Actual departure time of a bus from a bus stop
APw	Alighting passengers who use wheelchairs
AST	Alighting serving time for passengers
B_t	Control input parameters matrix that affect vector u_t
BST	Boarding serving time for passengers
C	Bus's maximum capacity
Dis	Fixed distance between two bus stops
dis	Mini link distance
DT	Dwell time
e	Kalman filter-error
F_t	State transition matrix of the process from the state at *t-1* to the state at *t*
g	Kalman filter-gain
H	Transformation matrix from state vector of the estimation into the measurement domain
$HAAP$	Historical average number of alighting passengers
HAP	Historical average passenger arrival rate
HAS	Average speed of the historical data for the link
hs	Observed speed for the link from the previous days
l	Interval between two bus stops that are located next to each

m	Door number
MAE	Mean absolute error
ml	Interval period between two AVL updates
N	Number of historical days
n	Number of the bus stops on the route
OAS	Average speed of the entire past mini link/links
obs	Observed "actual" speed of bus at the link
P_t	Estimated state error covariance
$\bar{P}_t$	Variance of the a priori estimate
PAP	Predicted number of alighting passengers
$Pass_{left}$	Number of passengers that are left behind at the bus stop due to the bus being at full capacity
PAT	Predicted arrival time of a bus at bus stop
$Pdis$	Past distance from the last bus stop to the bus's current location
PDT	Predicted departure time
PH	Predicted headway
$PPAR$	Predicted passenger arrival rate to the bus stop
PPN	Predicted number of passengers that board
PRT	Predicted running times for remaining distance
PS	Predicted speed for the link
Pw	Wheelchair passengers' arrival rate prediction
Q	Process noise covariance
R_t	Measurement noise covariance

rd	Remaining distance to the next bus stop
RE_{max}	Maximum relative error
RE_{mean}	Mean relative error
RE_{rs}	Root squared relative error
$RMSE$	Root mean squared error
S	Bus's actual number of passengers when it leaves the bus stop
TRT	Total running time
t_{update}	Frequently time of AVL sending update
u_t	Control input at time *t*
v_t	Measurement noise
VAR_{in}	Historical data variance
VAR_{out}	Prediction variance
w_t	Process's white noise
x_t	State vector of the estimation at time *t*
$\hat{x}_t$	Predicted state estimate
z_t	Actual measurement of *x* at time *t*
β	Alighting serving time for passengers
ρ	Boarding serving time for passengers
σ	Length of time for the door to open and close

Chapter 1

1 Introduction

1.1 Overview

One of the main infrastructures of today's smart city, a current hot topic, is intelligent transportation system (ITS). Therefore, the modes of transport are improving day by day to become safer, less expensive, and more powerful. Revolutions in technology adapt new features to the transportation field. As a result of these revolutions, new connections between cities and nations have an impact on lifestyles, the global economy, and the environment[1][2][3][4].

In modern societies, transportation is the lifeblood of human daily needs; however, in most cities the transportation systems have become increasingly complicated due to the increasing number of vehicles on the roads[5]. While transportation infrastructures cannot meet the requirements of the daily commute, basic road infrastructures are limited in some areas. As a result of these limitations, the lives of people are at risk with World Health Organization (WHO) considering road injury to be one of the top ten causes of death and the leading cause of death for young people aged 15- 29 worldwide [6][7]. In This problem cannot be solved by physical expansion of the transportation systems network such as by building new bridges or roads or expanding existing roads, especially in busy cities or downtown areas. Instead, incorporating new technologies to reduce the traffic problems and to increase the efficiency of the transportation systems network could lead to alternative solutions. One such technology is intelligent transportation system (ITS). According to Intelligent Transportation Systems Society of Canada (ITS Canada), ITS is defined as incorporating new technologies, such as computer application, surveillance, sensors, and communication systems, into transportation systems for the purpose of reducing commute time, lowering costs, expending less energy, and reducing the number of traffic accidents [8][9].

In the 1990s, ITS was first introduced in the United States. In recent years, ITS has become an attractive topic for researchers and developers [6][10]. Advanced Public Transportation Systems (APTS), also called Intelligent Public Transportation Systems (IPTS), is one branch of ITS which aims to make public transportation (PT) systems more reliable. To achieve this reliability, the networks of public transportation are controlled, the performance of the network is kept in good condition, the scheduling of public transportation is improved, and users are updated with the information they need to plan their trips. Also, IPTS aims to assist operators in maintaining control over the transit network [11][12][13][14][15]. To ensure that APTS performs at a high level of quality, five categories of different technologies are used [12]:

i. Automatic Vehicle Location Systems (AVL) provide the control unit with real-time information about the vehicles such as location, direction, speed, and delay. This information helps the operators and the ridership to make better decisions. More details are provided in section 2.2.

ii. Traveler Information Systems (TIS) allow the ridership to have access to PT information such as bus arrival, departure, and travel time and scheduled timetable. Therefore, riders can plan their trips.

iii. Automatic Passenger Counters (APC) are used to count the number of passengers who are boarding the bus. More details are provided in section 2.2.

iv. Geographic Information Systems (GIS) allow passengers and operators to track buses or trains on the map by knowing their location and time[16].

v. Decision Support Systems (DSS) assist public transportation operators in making good decisions when there are unexpected situations or abnormal events. It also reduces the time that needed by the operators to take the decision [17][18].

1.2 Thesis Motivation

In recent years, as more people move to urban areas, city expansion has been causing commuting calamity. United Nations Economic and Social Commission for Asia and the Pacific (ESCAP) expects that urban growth in Asia will increase from 48 percent to 64 percent by 2050. On other report from United Nations (UN), the world's urban population is going to increase by 2.5 billion people by 2050, with 2.9 billion extra vehicles. With this growth in population, public transportation's (PT) ridership will also increase each year. For example, a transit system for a big city like Toronto, which has the third largest ridership in North America, served 538.1 million passengers in 2016; in other words, it served more than 1.7 million passengers each weekday. In 2003, TTC served only 405.4 million passengers. Therefore, over 13 years, the number of TTC passengers increased by 32.8%. Correspondingly, traffic congestion in Toronto has worsened. A report from City of Toronto shows that in 2006 for Greater Toronto and

Hamilton Area (GTHA), the annual cost of traffic congestion to commuters was $3.3 billion. This cost is compounded by the average commuter in GTHA facing an average of 81 hours of delay yearly. Increased traffic congestion and ridership affects the quality of life for residents. One solution to this problem is to develop a PT system that has a high quality of service (QoS) in order to decrease the number of cars in the streets. A reliable PT system will attract ridership and instill confidence in its value for daily use [19][20][21][22].

QoS is measured by many parameters, including accuracy of the bus's running time. If the bus's real running time is close to the time provided by the TIS, then the ridership is more likely to be satisfied. Riders become frustrated when the bus or the train does not arrive on time, and then they start looking for alternative ways of commuting such as car sharing, owning personal vehicles, and taking taxis. An experiment in Okayama, Japan reduced 63% of the average bus stop wait time by applying an arrival time prediction system. This system reduced the passengers' wait time at the bus stop by six minutes because the passengers knew the bus's predicted arrival time. The results of this experiment show that more than 60% of Okayama's residents believe that the running time prediction system has made them more willing to use PT instated of private cars. Therefore, adapting an accurate system to predict travel time is a key component of attracting ridership, instilling confidence in passengers, and improving PT's QoS. Also, an accurate system encourages people to use PT as their first choice for commuting as long as it respects their time [23][24][25].

Making PT highly reliable and efficient not only benefits the ridership but also has a positive impact on the economy, the environment, and energy. First, when PT is running well, the economy benefits in direct and indirect ways such as increased sales of PT tickets, more job positions, better wages for PT workers, and increased value of properties that are located next to

PT stations. Second, PT positively impacts the environment in many ways as shown by the following example: the Federal Transit Administration (FTA) indicates that the average single-occupancy vehicle (SOV) produces more greenhouse gas emissions per passenger mile than that produced by PT fleets. Third, another study shows some of the impact of PT on saving energy. It has been proven that increasing the PT ridership in the top five U.S. cities by 10% would save over 85 million gallons of gas per year [26][27][28].

By recognizing the value of internet of things (IoT) and the need for smart cities, PT companies should augment the intelligence of their infrastructures and develop the ability to interact with the ridership. Accurate prediction of running and dwell time are features that need improvement and attention to ensure that PT networks meet the requirements of future lifestyles as societies move toward smart cites. Finally, it is unfortunate that some PT companies have not yet installed Global Positioning System (GPS) in their fleet.

1.3 Thesis Objective

The thesis aims to develop a model that predicts the running and arrival times of the bus in order to manage the PT traffic and to reduce the difference between the predicted and actual travel time. This model benefits the following:

- Network planners, who update and review the route plans and create bus scheduling, are helped by the model to know which areas of the network need improvement or where they should add a new route. If the running time measurement is accurate, then network planners can identify the busiest stops and where delays occur.

- Decision-makers, who monitor the PT network, are helped by the model to perform the right actions at the right time in order to keep the PT network on track. For example, since the model predicts the arrival time of the buses at any bus stop, decision-makers know when and where they can add a new bus along the route to reduce the delay time.
- The ridership have more accurate knowledge of bus arrival time so that they do not need to depend on the bus's scheduled time and wait so long for the bus. Also, the model helps the riders to plan their trip according to the bus's real time so that they can arrive at their destination on time.

1.4 Thesis Contribution

This thesis aims to provide a dynamic model that predicts the running and arrival times of buses. The model, which is based on Kalman filter, is comprised of two main parts: bus running time prediction and bus dwell time prediction. The main contributions of the model proposed in this thesis are as follows:

- The model predicts the bus's total running time to the downstream stops every time the bus departs a bus stop. This means that the predictions are always updated based on the most recent data that arrives through AVL/APC systems.
- To produce a more accurate arrival time, the model continuously predicts the running time while the bus is moving between the bus stops. Based on these predictions, the model adjusts the long-distance prediction. Therefore, the system can react to any incidents that occur and incorporate new data into the running time prediction.

- The model not only considers the boarding passengers when it predicts the dwell time at a bus stop, but it also considers the number of alighting passengers and the number of passengers with wheelchairs. The model also takes into account how many passengers are on board and how many passengers are boarding and alighting. Then, the model shows how much space is left on the bus and how many passengers are left behind at the bus stop because the bus is too full.

1.5 Thesis Outline

The thesis is organized as follows: Chapter 1 introduces the thesis and presents the motivation, objective, and contribution of this work. Chapter 2 presents a literature review with three main sections. Section 1 is an introduction to the chapter, section 2 contains a survey on the technologies that are used to predict the travel time of the bus, section 3 presents a survey on models that predict the travel time of the bus, and section 4 reviews models that have been developed to predict the bus dwell time. The last section is a conclusion for Chapter 2. Chapter 3 outlines the system model. Section 1 provides an overview of Kalman filter, section 2 presents the proposed model's algorithms, and section 3 shows how these algorithms work. Chapter 4 presents the simulations results. Section 1 is an introduction to the chapter, section 2 provides the sublimation sittings and assumptions, section 3 shows the performance results, and section 4 presents the model performance evaluation. Finally, section 5 concludes the chapter. Chapter 5 is the thesis conclusion and the future research.

Chapter 2

2 Literature Review

2.1 Introduction

PT companies, like other companies, seek their customers' trust, confidence, and satisfaction. Therefore, one aspect that makes the PT system network more reliable and the first choice of ridership is an accurate real-time information system to provide ridership with predicted arrival time, departure time, and travel time of buses and trains. In fact, travellers want to plan their trip according to real-time information, not a scheduled timetable, because of incidents that affect the bus's scheduled time (e.g. bad weather conditions, construction on the road, unexpected collisions, etc.) [29]. The causes of frequent bus travel-time changes can be categorized into two groups of demand and capacity (see Figure 2.1) [30][31].

Predicting the bus travel time is the key component of Real Time Passenger Information System (RTPIS/TIS). RTPIS is a system that provides passengers with the real time of the bus's or train's arrival or departure via text messages, smart phone application, or social media. Also, RTPIS provides PT systems with the live location of the bus through GPS. This information is not only useful to ridership; it also helps operators, planners, and drivers. In addition, this information is important in planning upgrades to the PT network [32].

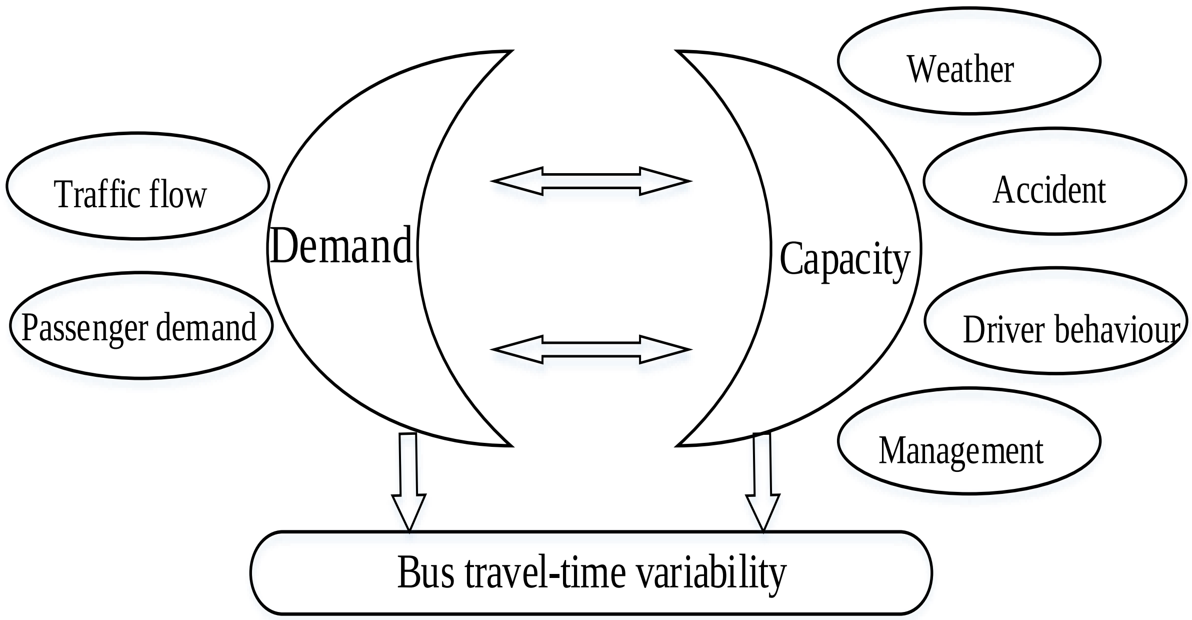

Figure 2.1 Demand and capacity-related determinants of bus travel-time variability

This chapter surveys three main topics of interest in this thesis. Section 2 contains an overview of the technologies that are used to predict the travel time of the bus, namely Automatic Vehicle Location Systems (AVLS/AVL) and Automatic Passenger Counters (APC). Understanding these technologies is necessary to comprehend how predictive travel time models and algorithms work to collect the data. Section 3 presents a survey of the research studies that have been conducted to develop models and algorithms that are used to predict the travel time of the bus. Dwell time models and algorithms are reviewed in section 4. Finally, section 5 concludes the chapter.

2.2 Technologies Integrated with Bus Travel Time Prediction Models

2.2.1 Automatic Vehicle Location (AVL)

Automatic Vehicle Location system (AVL), which is also referred to as Automatic Vehicle Monitoring (AVM), Exploitation Aid System (EAS), or Exploitation Support System (ESS), is used to track the location and the speed of the vehicle automatically so that the vehicle's activities can be computerized to facilitate the operators' duties of analyzing the system network. AVL systems are commonly used by transit agencies and delivery companies to track their fleet at any time on any route so that the time of arrival or delivery can be estimated [12][33][34][35][36].

There is more than one way in which AVL systems can work according to the technologies that are used to operate AVL system. Nowadays, GPS-based and signpost "active/passive" are the most common technologies used on AVL. There are other technologies that are less common, such as Ground-Based Radio and Dead-Reckoning. Specifically, GPS-based AVL depends upon orbiting satellites that transmit a signal down to the GPS receiver located in vehicles. The receiver equipment must receive signals from at least three different satellites before determining the vehicle's location by triangulation. Then, the receiver sends the vehicle's location and the other information to the control centre through wireless communication medium as shown in Figure 2.2 [37][38][39].

The advantages of AVL GPS-based are as follows[38]:

- It can operate anywhere as long as there are satellite signals;

- No roadside infrastructures need to be installed or maintained; and
- It determines the location very accurately.

However, the disadvantage of GPS-based AVL is that the satellites' signals can be blocked by tunnels, high-rise buildings, or trees.

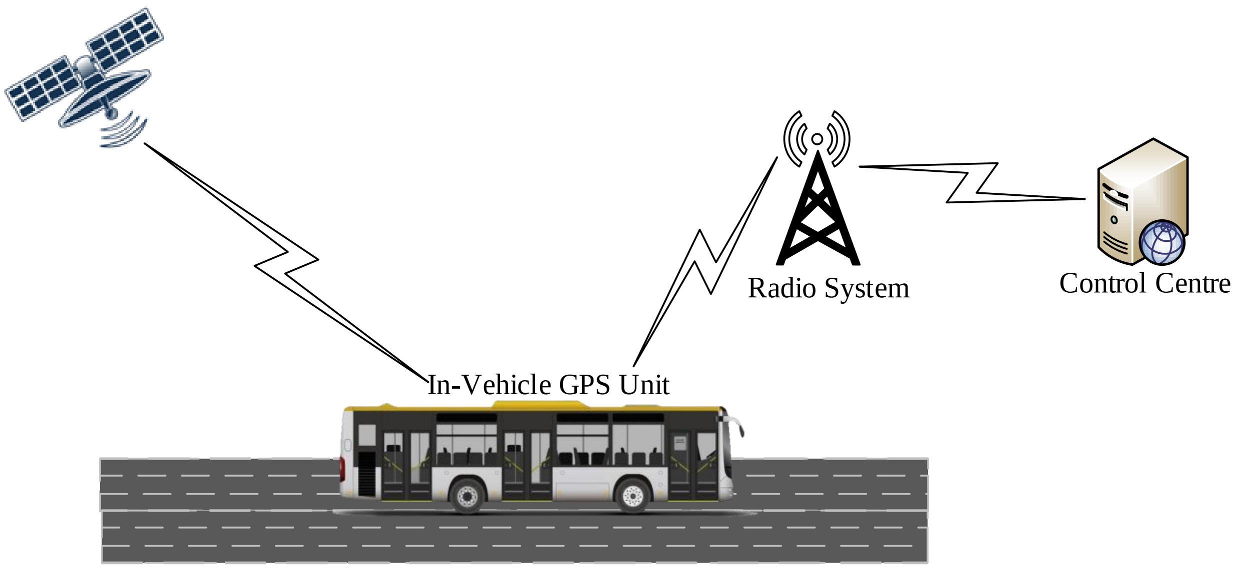

Figure 2.2 GPS-based automatic vehicle location system

Signpost-based AVL system, on the other hand, tracks the location of a vehicle by the two different methods of active and passive. The active method works based on signposts (beacons) along the route that transmit a unique signal. Then, the vehicle (i.e. bus, train) reads the signal when it passes by the signpost. After that, the vehicle transmits the information to the control centre, as shown in Figure 2.3 [40]. In contrast, the passive method works based on the vehicle sending a unique signal while it moves along the route. Then, signposts that are located along the route receive the signal when the vehicle passes by them. After that, the signpost transfers the information to the control centre, as shown in Figure 2.4 [40][41].

Signpost-based AVL system has the following advantages:

- This well-established technology works well in tunnels; and
- Fewer dedicated radio frequencies are required.

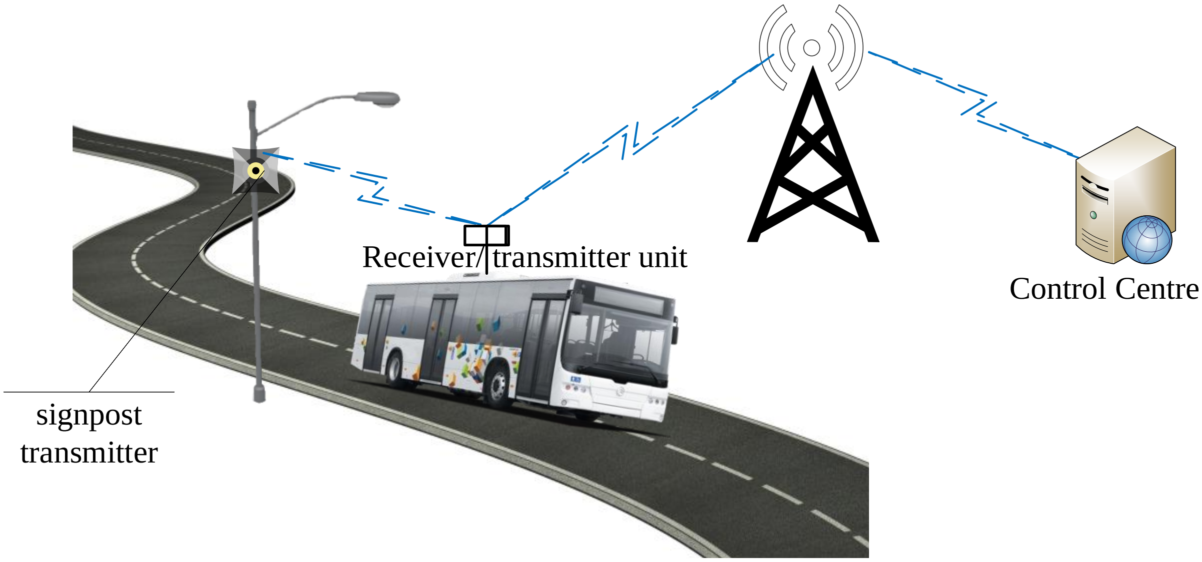

Figure 2.3 Communication processes of signpost "active"

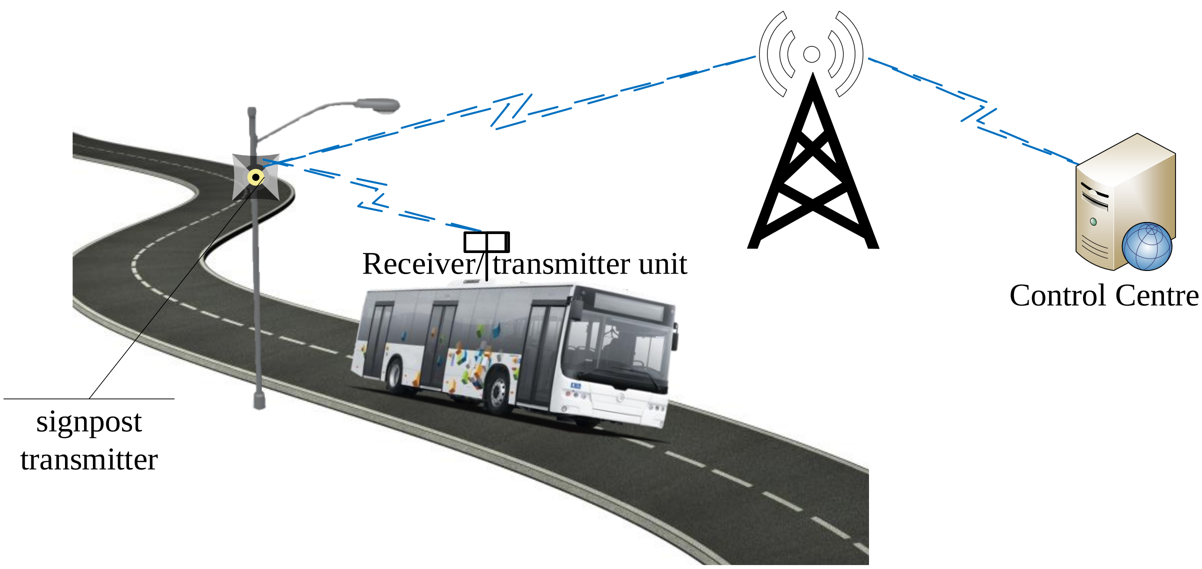

Figure 2.4 Communication processes of signpost "passive"

The disadvantages of it are as follows:

- Signposts have to be installed where AVL has been designed to work;
- The vehicle's location is tracked only when it passes by a signpost, not before or after; and
- Signpost-based AVL system is not an effective option when a fleet does not follow a specific route or when the route is blocked and the vehicles take a temporary route.

AVL system has several benefits that have encouraged its application by transit agencies to manage their fleet on the ground. These benefits are as follows:

- AVL helps transit agencies to collect real-time information about the fleet which helps to manage the fleet network;
- AVL helps ridership to receive updated information of the bus's arrival, departure, and travel time;
- AVL decreases the overall cost of the operation; and
- In cases of emergency, AVL helps to ensure a better and faster response.

2.2.2 Automatic Passenger Counters (APC)

Automatic Passenger Counting (APC) system provides an important source of data that can improve the service quality of the public transportation (PT) system network. APC system counts the number of passengers on board buses or trains by recording boarding and alighting passengers' data without human intervention. In the past, the number of boarding and alighting passengers were not counted automatically; instead, collecting the ridership data was a manual process [12][35][36].

In general, APC technologies can be divided into two main categories of static and on-board systems. For the static systems, the equipment is installed in stations to count ridership. For the on-board systems, the equipment is installed inside the vehicles (i.e. buses and trains) [42]. [42]. Specifically, on-board APC systems contain three main components: sensors on the vehicle doors, on-board computer, and wireless communication antenna (see Figure 2.5) [43].

Two sensors at each door have infrared beams to count on-board passengers. When the beam breaks, this indicates that a passenger is boarding or alighting the bus depending on the order of the beam breaking. Another passenger-counting technique measures the weight of the vehicle and then estimates the number of passengers [44].

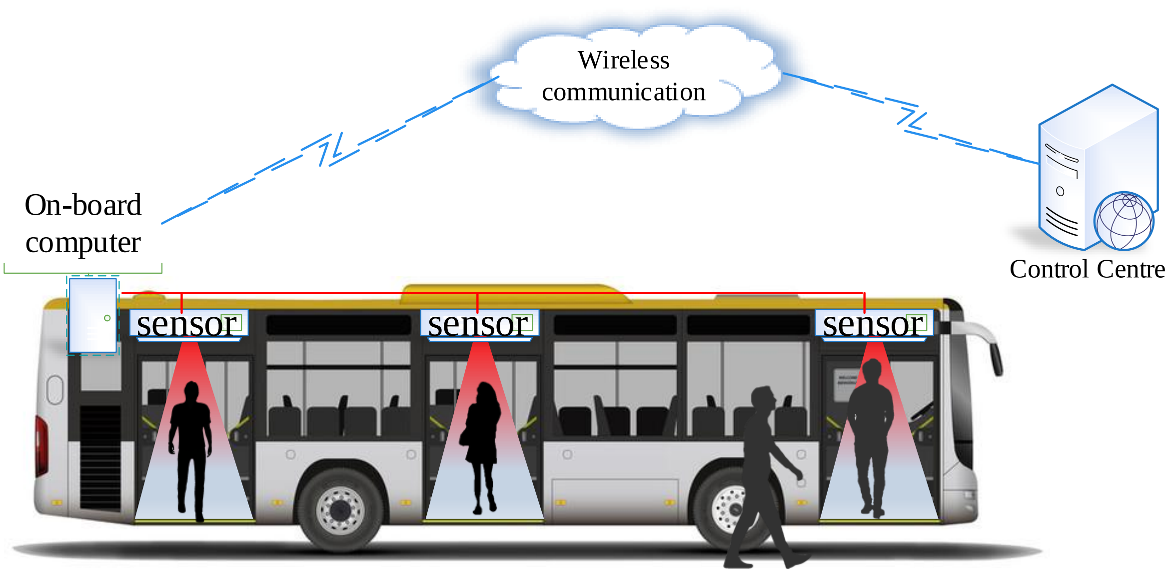

Figure 2.5 Communication process of on-board APC system

One of APC-static system's ways to count passengers is by using image processing. The authors of [45] proposed an algorithm to count the passengers. The algorithm has an input of images taken from the bus station. For each image, the algorithm detects the passengers' heads with an accuracy of 85%. However, the algorithm has difficulty detecting the head if the passenger wears a hat or clothing that is a colour similar to the head's colour.

The benefits of the APC system are as follows:

- Knowing the exact number of riders assists the operators in identifying which bus stops of the networks are most in demand;
- PT agencies can satisfy their customers by increasing the service at high-demand bus stops; and
- PT agencies can estimate their revenue.

2.3 Models and Algorithms to Predict Bus Travel Time

This section presents a review of travel time prediction models. The aim is to provide an overview of the existing models and technologies that predict the travel and arrival time of a bus and to identify which variables and factors have an effect on the travel time prediction. Initially, the variables and factors that affect prediction can be classified into five groups: *a)* passengers, such as the length of time that passengers take to board and alight the bus; *b)* infrastructure, such as number of stops on the route, distance between stops, number of traffic lights, and number of intersections; *c)* environment, such as weather, traffic patterns, and traffic accidents; *d)* driver's behaviour, such as driver's schedule and driving habits; and *e)* operation and management, such as travel time. The relationship between these variables and the bus's travel and arrival times can be explained by several models that have been developed for this purpose [46].

A variety of models have been developed and utilized in the past years to predict the travel and arrival times for ITS. These models can be classified into three categories, as shown in Figure 2.6 [47] [48]. Although many models and methods have been used to predict travel time, they are less popular than historical average models, regression models, Artificial Neural Network

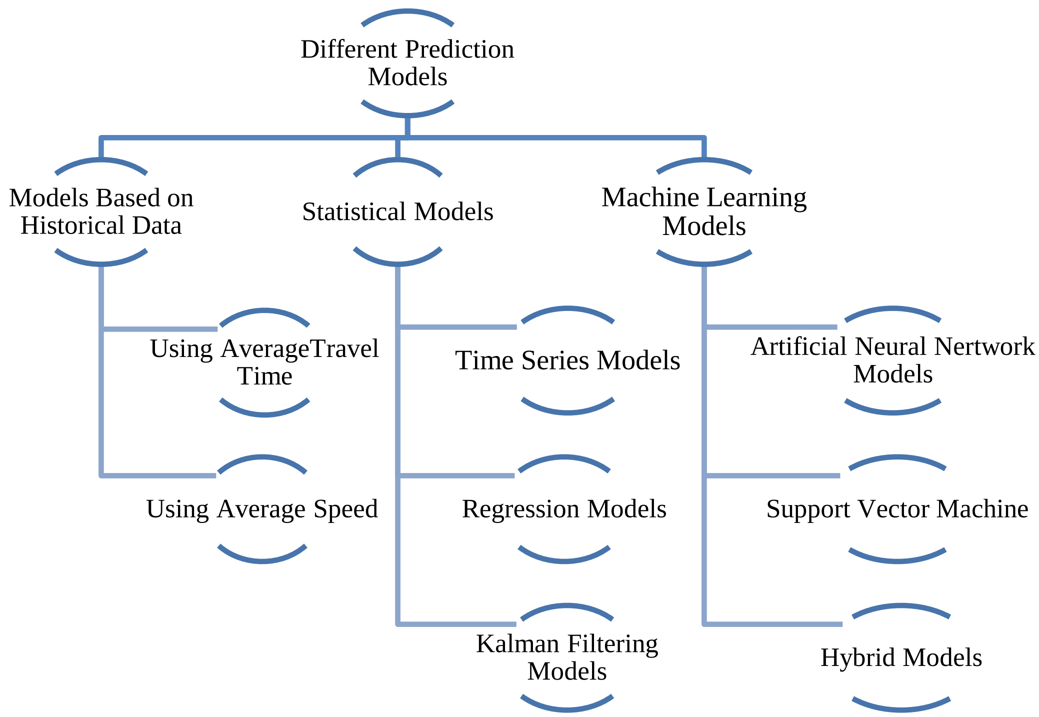

Figure 2.6 Different Travel Time Prediction Models [47]

(ANN) models, and Kalman filter (KF). Less popular models include Markov chains, k-nearest neighbour model, and support vector machines. KF has been proven to be perform better than other prediction techniques.[49][50][51]

2.3.1 Statistical Historical Average Models

Historical average models use the basic and uncomplicated method of predicting the bus's travel time from one point to another point. Travel times are predicted based on the historical data of that same day and same time period; therefore, the traffic conditions are assumed to be the same for every period in the day. The authors of [52][53] have recognized the phenomenon that the traffic conditions follow the same daily and weekly patterns. This leads to the assumption that historical average data of any traffic conditions in the past can provide a reasonable prediction for the same traffic on the same day and time in the future. Thus, an historical average model can give good results in areas that do not have busy traffic [24][54]. In fact, an extensive collection of historical data is required [55]. On the other hand, the ridership does not usually worry about the arrival time and the travel time of the bus under normal traffic conditions like they do when the traffic is unusual [56]. The authors of [57] proposed a model that depends upon the APC historical data. The model has two major components: to predict the travel time between two bus stops that are next to each other, and to search for the next bus's arrival time by using an algorithm.

The authors of [58] proposed a model based on historical data by using vehicle location and timestamp as an input for the model. The arrival time for a bus at a bus stop is based on the arrival time at the previous bus stop and the historical data for the particular stop and time. The model has three modules of data collection, data analysis, and pre-processing. Specifically, data collection collects the data to build a historical data bank, data analysis searches historical data for incomplete data, and pre-processing adds the missing data. Since the model is completely dependent on past data, it does not respond to emergency incidents that may occur on the bus route.

Historical average models are used in combination with many other models as in in [49][59][50].

2.3.2 Regression Models

Regression models depend upon a mathematical relationship between a set of independent variables and a single dependent variable, for example bus arrival, travel time, and distance between two bus stops [60][61]. Despite the ability of regression models to anticipate the effect of independent variables on the dependent variable, a set of independent variables is required that are not correlated. This requirement limits the application of regression models due to the difficulties of coming up with correlated variables [31].

In [62], the authors used a Bayesian regression approach to develop a travel time prediction model for the purpose of estimating the travel time in central business districts (CBDs). The authors installed video cameras in four different types of streets: *a)* two-way street with two lanes in each direction and permitted parking; *b)* two-way street with one lane in each direction and no parking; *c)* one-way street that has three lanes and no parking; and *d)* one-way street that has three lanes for all kinds of traffic, one lane for transit vehicles only, and no parking. The proposed model of [62] estimates the travel time based on the movement of vehicles that flow through, turn right, and turn left. Also, the model estimates the number of intersections in each kilometre, percentage of stopped vehicles at each link, percentage of heavy vehicles, and flow of transit buses. As a result of their development, the predicted travel time of one-way streets has the smallest error value. These models are expensive because cameras must be installed in every route; furthermore, these cameras require maintenance. As well, the need to install cameras in every route makes it difficult to establish a new route. This slows the expansion of the PT network.

A nonlinear regression model was developed by the authors of [63] to predict bus delays. The authors divided each bus's route into links and then determined the variables that cause the delay. These variables are link length, number of bus stops in the link, number of buses in the link, bus efficiency ratio estimates, number of stops made by a bus on the link, traffic density on the link, traffic flow, heavy vehicle density, number of passengers on each link, and number of traffic signals in each kilometre. The authors of [63] concluded that the length of link, number of bus stops, and bus efficiency ratio estimates are the independent variables that have the most significant effect on bus travel time. Moreover, the other independent variables have a somewhat lesser effect upon the travel time of the bus.

A multivariate linear regression model was developed by [64] to predict the travel time between two bus stops. The authors considered the following variables as independent: number of stops, number of boarding and alighting passengers, dwell times, distance, and weather conditions. Since the model uses APC data as an input to predict the travel time and the dwell time, APC was installed on the buses. However, the APC data is only transferred to the data centre for analysis after the bus has completed the run and returned to the garage. As result, the passenger will not receive a real-time update of the bus's arrival time. This makes this model less efficient. Therefore, the model does not provide the ridership with real-time information. The model does not respond to cases of emergency by updating the passengers about the delay.

2.3.3 Artificial Neural Network (ANN) Models

Artificial Neural Network (ANN) models have the ability to solve complex nonlinear relationships; moreover, their variable inputs are not required to be independent. The authors of [65] developed two ANN models, link-based and stop-based, to predict the real-time bus's arrival. In the link-based ANN, the distance between two bus stops is divided into links; therefore, the model predicts travel time to any stop by adding the travel times of the links between bus stops. On the other hand, stop-based ANN predicts the travel time by aggregating certain data, such as the means and standard deviations of volumes, speeds, and delays on the links that are located between the two bus stops. The authors of [66] conclude that stop-based ANN model gives more accurate results when there are multiple intersections between stops. Meanwhile, link-based ANN model's results become more accurate when the intersections between stops are few.

The authors of compared three prediction models: historical, regression, and ANN. The authors used AVL data in their study and found that ANN outperformed the historical and regression models. The factors that they considered in their model are arrival time of the bus at the previous bus stop, traffic congestion, and dwell time at the previous bus stop. Similarly, the authors of [67] used ANN model with GPS data only and found that ANN model produces more accurate results than historical model. A model that uses only AVL or GPS has less accuracy than a model that uses extra technologies such as APC.

2.3.4 Kalman Filtering-based Models

In 1960, R. E. Kalman proposed an algorithm [68] that uses a series of data collected over time to estimate the current state of the system and to predict the future state of the system by using observed and historical data [69]. In other words, this algorithm predicts what is going to happen based on what has been observed [70] This ability is one of the strengths of Kalman filter (KF). One application of KF is its use in transit systems to predict the bus's travel time. In PT systems, KF uses historical data and/or real-time data collected by technologies, such as AVL and APC, to predict the travel and arrival times of the bus and the passengers' rate of arrival to the station or the bus stop.

In [59], the authors proposed an algorithm that aims to present real-time departure information to help inform riders who are waiting at any transit station. The authors used AVL data and historical data to predict the arrival time of the bus. Their algorithm has two main components of tracking and prediction, as shown in Figure 2.7 [59]. The first component tracks the current location of the vehicles by using KF model. This component has two inputs: AVL data and control input. AVL data provides the location of the bus and the control input represents the relationship between the vehicle and the driver. The second component predicts the bus's

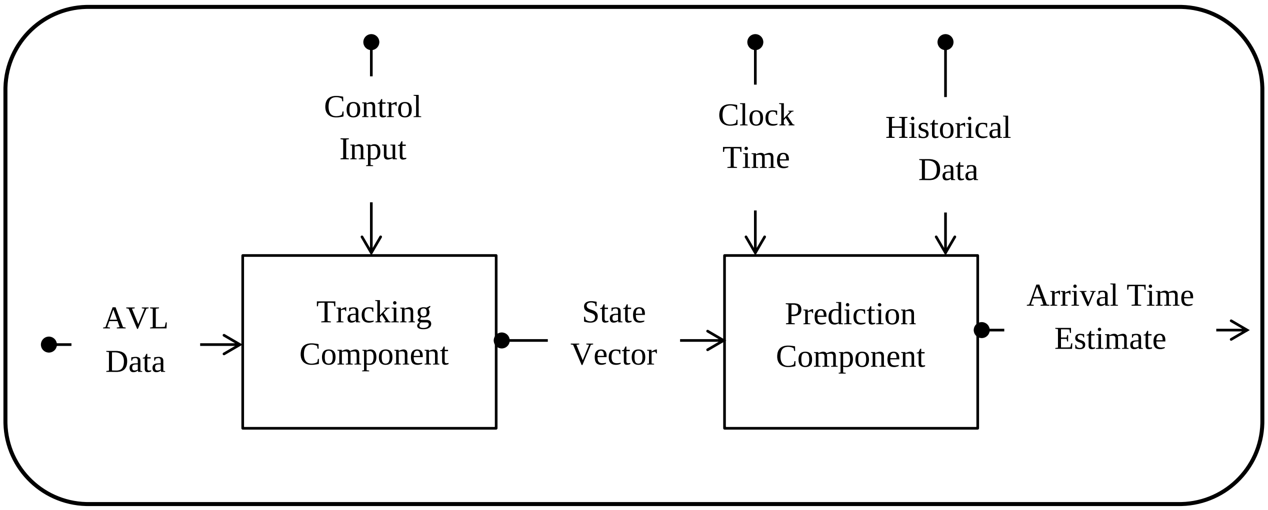

Figure 2.7 Block Diagram of the Wall and Dailey's prediction algorithm

arrival time through statistical estimation technique. This component relies on three inputs of vehicle location, current clock time, and historical data. The algorithm calculates the predicted arrival time by adding predicted travel time to the current time. As a result of the proposal, the algorithm could predict the arrival time of the bus with an error rate of 12% at 15 minutes before the bus arrives. However, this algorithm considers the dwell time as an independent variable. In fact, it is not an accurate assumption to consider the bus's dwell time an independent variable because the dwell time affects the bus's headway, which leads to more passengers waiting for the bus at the next bus stop.

The authors of [71] developed a system by using KF instead of static averaging algorithm. The authors called this new system Countdown. This system, which measures the arrival time estimations, was tested on the bus system of London city. Specifically, the bus route is divided into links. Each link has a number of bus stops and between every two links there is a radio beacon. The buses are also equipped with an on-board radio. Every 30 seconds, buses send data such as their location and completed links to the central computer. The central computer uses an algorithm based on KF to track the bus's location. Next, the ridership is provided with information about whether or not the bus will be late and the estimated arrival time. The model of [71] shows the error distribution improving by up to 7% and a significant decrease on the maximum absolute error. This model's need for infrastructure, such as a radio beacon, makes it costly; however, the model can be reapplied by using new technologies (i.e. AVL, APC, and GPS).

KF was used on [50] to develop a model that could predict the bus's arrival time by using data collected by AVL and APC. The model takes into account the impact of the bus operators' efforts to make a schedule recovery. The data that comes from AVL, APC, and historical trips helps to simplify the recovery of the schedule. The model initially uses time table data and then continues making predictions based on observed data. At the current stop, the model predicts the arrival time at all stops to the downstream stop. When there is updated data, the prediction is adjusted according to these updates. This process repeats itself until the bus reaches the final stop. Therefore, the prediction error is higher at the final stops because these stops are farther away from the bus's starting location. However, the difference between the predicted arrival time and the actual arrival time at the final destination decreases as the bus comes closer to the final stop. The model of [50] is not computationally intensive which makes it easier to apply to the real world transit system. Moreover, driver behaviour is one of the factors that affect travel time.

The authors of [72] used KF in the second element of their model. Their aim is to develop a dynamic model to predict bus arrival time, take into account various factors that affect the bus's travel time, and add real-time information to make the prediction more accurate. The model has two main elements: ANN and KF. KF is used to adjust the arrival time prediction that has already been predicted by ANN by using real-time information. Moreover, ANN requires time to complete the process training. At the same time there is a stream of information that adds to the database while the bus is running, and this information is transferred from APC. The fact that ANN does not have a dynamic feature to adjust the prediction according to new information from the trip means that the model does not consider unexpected incidents that may occur on the bus route, such as construction and accidents that have an impact on the arrival time prediction. Therefore, KF-based algorithm is used to improve ANN's prediction by considering the impacts of such incidents. The authors of [72] conclude that the dynamic algorithm, KF, always outperforms the comparable ANN model.

The authors of [49] improved the algorithm of [71] by replacing the previous predicted running time in the KF prediction equation with the observed (actual) running time [73]. the authors of [49] developed a model to predict the bus travel time by using KF technique. The input data was collected by AVL and APC. The model consisted of two KFs: one to predict the running time, and the other to predict the dwell time at the bus stop. The authors divided the route into five links with each link having two to eight bus stops. The link starts and ends by consecutive time point stops (checkpoints). Predictions of running time and dwell time are only made at checkpoints. One feature of this model is that the bus's dwell time is predicted separately from the prediction of the bus's running time, which helps to detect the effects of the bus's earliness or lateness at the bus stop. The authors used historical data and real-time data to predict the running time. The historical data consisted of the running time records of the bus for the same period of time in the last three days. For real-time data, the authors use the observed running time data of the previous bus that just been received from AVL. On the other hand, the dwell time is predicted first by using KF to predict the passengers' arrival rate. Specifically, the model uses the historical passengers' arrival rate for the last three days and the observed passengers' arrival rate of the previous bus that had just been provided by APC. Then, the passengers' predicted arrival rate is multiplied by the predicted headway (the time between the arrival of two buses at the bus stop) and the passenger boarding time (2.5 sec/passenger was assumed). The authors of [49] also developed an historical average model, a regression model, and a time lag recurrent neural network model; however, the dwell time in each of these models is included in the link travel time. The authors concluded that the KF technique performs better than historical average model, regression model, and time lag recurrent neural network model in terms of accuracy and the dynamic ability of the KF technique to update itself based on new data.

2.3.5 Other Models to Predict the Bus's Travel Time

Predicting the travel time is not limited to the previous models' techniques; in fact, some researchers have used other mathematical ways to develop algorithms. In [74], the authors developed a model to predict the travel time. The authors divided the model into two main parts: arrival time at the next station from the bus's current position, and arrival time at the following two or more stations from the bus's current position. For the first part, the model estimates the arrival time at the next station from the bus's velocity. For the second part, the model divides the bus's travel time into three components: *a)* bus's running time, *b)* bus's dwell time at bus stops, and *c)* delay caused by traffic lights. The travel time from the bus's current position to the targeted bus stop is divided into symmetric subsections. For each subsection, the running time, dwell time, and traffic lights' delay are estimated. The model estimates the running time of the bus by measuring the average speed of traffic every five minutes. Then, the average travel time for each subsection is calculated. The bus's location is identified through GPS; therefore, the bus's travel time through each subsection is predicted based on the average speed of traffic and the bus's location. Next, the total travel time is predicted by adding the average travel time of these subsections. The bus's travel time is calculated depending on the average speed of traffic with new updates every five minutes. The accuracy of this model in downtown areas or other busy areas is not as good as it is in areas that have stable traffic. For example, if any incident occurs on the bus's route, the model will only detect the delay after the GPS update which could take up to five minutes.

The model presented in [75] shares some similarities with the previous model. The model has two main parts. The first part, short-distance prediction, predicts the bus's arrival time up to the next three bus stations from the current bus's position. This arrival time is estimated, as in the second part of the previous model. However, the second part, long-distance prediction, predicts the bus's arrival time at the stations that are located after the next three stations. For long-distance prediction, the authors of [75] estimated the bus's arrival time based on real-time information of road condition and historical data. The historical data was classified into groups according to the travel time period of the day and week. The model works as follows: To calculate the arrival time of the bus, for instance at the seventh station, the model first estimates the bus's travel time to the first three stations as if it is a short-distance prediction. Then, the model adds this travel time to the historical bus's travel time data from the fourth station to the seventh station. Like the other models that depend heavily on historical data, this model is not ideal for busy areas or for a city that has changeable weather because it does not take into account the factors that affect road conditions at the time of the current day. Indeed, the ridership always needs to have the travel time updated, especially when the traffic is abnormal.

The authors of [76] proposed a model to predict the bus's arrival time based on historical data and real-time data. The model has two algorithms: short-distance prediction based on the real-time traffic conditions, and long-distance prediction based on k-Nearest Neighbors (KNN). The short-distance prediction algorithm predicts the arrival time at the bus stops that are located in the next three kilometres. The total running time of the line is divided into link travel time and dwell time. The average speed of the same link for the last ten minutes is calculated and updated every two minutes. Therefore, the arrival time can be predicted from real-time data and delay time. On the other hand, long-distance prediction is made for bus stops that are more than three kilometres away from the bus's current location based on KNN. The model outperforms ANN and KNN alone with no real-time data in terms of the algorithm accuracy and efficiency.

2.4 Dwell Time Prediction Models

Dwell time plays a key role in the bus's total travel time prediction which is defined as the length of time that the transit vehicle takes at the bus stop or station in order to service the passengers who board or alight the transit vehicle. The time extends from when the doors are opened until when they are closed [77] [78][79]. Some models do not predict dwell time separately from the running time when they predict the total travel time or the arrival time, so they just ignore the effectiveness of the dwell time. Indeed, the dwell time consumes up to 26%

of the bus's total travel time, especially at busy bus stops or during the rush hour [80]. The factors that have an effect on dwell time are listed in the *Transit Capacity and Quality of Service Manual* as follows: the volume of boarding and alighting passengers, method of fare payment, in-vehicle passengers' movement, and the space of the bus stop [81]. Other factors which affect the dwell time are time of day, type of route, and type of vehicle [82].

In [80], the authors developed four linear and nonlinear models to estimate the bus's dwell time by using APC. Model A, which is a simple linear model, considers dwell time as the dependent variable and the total number of boarding and alighting passengers as the independent variables. Model B, which is a multivariate linear model, considers dwell time as the dependent variable and the number of boarding and alighting passengers separately as the independent variables. This model needs the passengers to be defined as either boarding or alighting, and it assumes that the rates of boarding and alighting are independent. Model C, which is a multivariate nonlinear model, treats the number of boarding, alighting, and standee passengers as independent variables. Also, it looks at the relation between the boarding and alighting rates and crowding level as inverse. Model D, which is a nonlinear model, counts the relation between the dwell time and total number of passengers as nonlinear. In conclusion, model C generally outperformed the other models.

The authors of [75] predicted the dwell time for the next three coming stops by calculating the average dwell time of the bus at any one of these three stations for the past hour. They calculated the average dwell time as follows: the summation of departure times of the bus at any station minus the arrival times of the same bus for the buses that pass by the station in the last hour divided by the number of buses that pass by the station. The dwell time prediction in this model depends on the last hour's data; therefore, if an incident occurs on the route for only one hour, it will affect the dwell time for the next hour. That means that the predicted dwell time will be affected for two hours. Also, the model does not take into account the number of boarding and alighting passengers which changes from hour to hour.

The author of [83] used six multiple regression models to estimate the difference between bus's dwell time when different payment methods are used and other factors such as the age of the passenger and the effects of interactions between the boarding, alighting, and standing passengers. Also, the author measured the effect of the bus having a high floor level, a low floor level, and steps by its doors. At each tested bus stop, the following information was recorded: the time when the doors opened and closed, the number of boarding passengers and their payment methods (i.e. prepaid card, cash, cash with change back, student pass, or tickets that need to be stamped by the driver), and the number of alighting passengers and their age (i.e. students, adults, or seniors). The study shows three scenarios for the dwell time of a bus to board 100 passengers: it takes 4.4 minutes if the passengers pay outside the vehicle such as at the station entrance; it takes 7.7 minutes if the passengers pay by cash or prepaid card with magnetic strip; and it takes 19.8 minutes if the passengers pay by cash. Therefore, this shows the effect of passengers' payment methods on dwell time. When payment method is upgraded, then dwell time decreases by more than 22%. Also, the study proves that the buses with steps require longer dwell times than those which do not have steps. As well, the study shows that senior passengers take a longer time to board and to alight the bus than younger passengers. Finally, friction between boarding, alighting, and standing passengers causes a longer dwell time. This was the first study to compare the effects of fare payment methods and passenger age on bus dwell time.

The authors of [84] proposed a model to estimate the bus dwell time that contains an algorithm to predict boarding and alighting passengers and an algorithm to estimate bus dwell time. The first algorithm uses historical data and information about the previous bus on the current day to predict the number of boarding and alighting passengers. Also, the algorithm considers the bus's capacity by counting alighting passengers, boarding passengers, and passengers who could not find space on the bus. The second algorithm considers two situations: boarding from the front door and alighting from the rear door; and boarding and alighting from all doors. The algorithm also considers the crowding affect. In comparison to previous models used for the same route, the model provides better estimation — especially in areas with a high occupancy of vehicles.

In [85], the authors developed a model to estimate bus dwell time by using four different time series-based methods of random walk, exponential smoothing, moving average, and autoregressive integrated moving average. AVL system was used to collect the historical data. The historical data was classified based on three categories of time of day: morning peak, morning inter peak, and evening peak. For the bus stops that were chosen to be in the study, some are located in CBD and others are located in non-CBD areas. The four models were ranked according to their accuracy, simplicity, and robustness. Overall, the moving average model outperformed the other three methods for bus stops that are located in CBD areas. Overall, exponential smoothing model shows better results than other models in terms of robustness. For non-CBD bus stops, autoregressive integrated moving average model outperforms other models overall; however, moving average model shows the worst performance for non-CBD bus stops.

In [78], the authors defined a new variable that affects the dwell time. The total bus stop time (TBST) is the time from when the bus manoeuvres into the bus stop until the bus leaves the bus stop and successfully merges to the traffic stream. The TBST includes the dwell time. For this study the authors selected 30 bus stops that are located near intersections and another 30 bus stops that are located at mid-blocks in order to prove the effect of the new variable. The data was collected manually on weekdays during three periods: 7:00 a.m. to 10:00 a.m., 12:00 p.m. to 2:30 p.m., and 4:00 p.m. to 6:00 p.m. The data that was collected includes bus stop ID number, bus route number, arrival time of the bus at the bus pad, number of passengers alighting and boarding, time for the doors to open and close, length of time for the bus to leave the bus pad after the doors have closed, whether the parking next to the bus stop is permissible or not, number of lanes in the street, and bus pad's length. The data was collected under normal weather conditions. The authors developed a regression model based on the time of day and the type of bus stop. TBST was defined based on the following parameters: dwell time, number of boarding passengers, parking allowance in the street, number of approach lanes, bus pad's length, and number of alighting passengers. According to the study's results, the bus's dwell time was between 20 and 29 seconds at bus stops at intersections and TBST was between 42 and 67 seconds. On the other hand, the dwell time at bus stops mid-block was between 17 and 19 seconds and TBST was between 31 and 36 seconds. This study depends on manually recorded data because there are differences between AVL/APC data and field data. This makes the calculation and estimation of TBST not practical for the operators. However, this approach would be more beneficial if it depended on automatic recording of data without human interference.

The authors of [86] proposed a model to predict bus dwell time at a bus station based on KNN algorithm. The authors took into account two main factors that affect bus dwell time: passengers' demands and traffic conditions. Passengers' demands make the bus spend a longer time at the bus station serving passengers, and traffic conditions affect the bus's headway which leads to more passengers waiting for the bus at the bus station. The authors assumed that there is a similarity between the current dwell time at a bus station and the previous day's dwell time at the same bus station at the same time of day. The model uses the dwell time of the bus at the upstream as an input to the KNN algorithm, and this was estimated first. Also, the model uses historical data divided into four groups: morning peak on weekdays, off-peak on weekdays, evening peak on weekdays, and weekends. Finally, the model predicts bus dwell time at downstream stations as follows: *a)* find the historical data recorded by GPS, *b)* find the upstream data, and *c)* calculate the dwell time by using KNN. The model performs better than average dwell time method and KNN method based on total data. When the GPS data is processed, any data with a speed of zero and within 50 metres from the bus station is considered to be dwell time data. Therefore, the stations that are located next to traffic lights or intersection are less accurate in their historical dwell time data because the time the bus stops at the intersection might be considered a part of the bus station's dwell time if the intersection is less than 50 metres away.

In [87], a model was developed to estimate the bus's dwell time and time lost serving the stop. The model defined three factors that have an effect on the dwell time and cause the serving stop to lose time. These factors are bus's acceleration and deceleration times, serving boarding and alighting passengers, and bus's dead time at bus stop. Specifically, the model assumed that the bus deceleration rate is 1.2m/s^2 and the acceleration rate is 1.0m/s^2. The authors used a polynomial model incorporating kinematics to estimate the bus's acceleration starting from the second its doors are closed. Then, the model predicts the time to serve boarding and alighting passengers by assuming that the doors' opening and closing takes two to five seconds. After that, the model defines the door/doors as being used for boarding, alighting, or both. Then, the serving time for each boarding or alighting passenger is multiplied by the number of boarding or alighting passengers. The busiest door, which takes a longer time to serve passengers, is taken into account while the rest of the door/doors are ignored. The third part, which is the bus's dead time at the bus stop, has four components. These components are average delay for re-entering the traffic, bus stop failure time, boarding lost time, and traffic signal delay. The model considers re-entering the traffic as a function of capacity and the degree of saturation. The total estimated bus's dwell time and time lost serving stop is the summation of the three factors. This model does not depend on technologies such as AVL and APC that can help with real-time information in a dynamic model.

2.5 Summary

This chapter presents a review of AVL/APC systems' technologies used in APTS to provide real-time data information to the control centre. Section 2.3 offers a literature review of the models that predict bus's travel, running, and arrival times. This section also presents an overview of the most common models that are used to predict travel time, namely historical average models, regression models, Artificial Neural Network (ANN), and Kalman filtering (KF). In addition, the chapter presents a literature review on bus travel time prediction models that do not follow these techniques. According to this literature review, historical average models are not accurate when the traffic conditions are not normal. Also, for short-term prediction, KF model outperforms regression models and ANN models because it is dynamic with simple calculation. KF model performs very well as a dynamic model, because it does not take a long time to make predictions and it does not go through learning and training stages. The last section in Chapter 2 outlines a survey of the models that are used to predict bus's dwell time. The next chapter, Chapter 3, presents the methodology of the thesis's proposed model.

Chapter 3

3 System Model

3.1 Kalman Filter Overview

Kalman filter (KF) is an optimal estimator method that aims to remove information that is known to have error, noise, or uncertainty from useful information. In fact, KF is one of the greatest discoveries of the twentieth century in the field of statistical estimation theory. KF helps to control any dynamic system by estimating what will happen. First, from the observed data, KF understands what the system has done in the past. Next, based on this information, KF predicts the future stages for the system. One feature of KF is that it does not require the entire data to be input in order to initiate the estimation. Instead, KF starts estimating by using the data that is available at that time. Essentially, KF estimates by understanding the variation or the uncertainty of these data inputs, and KF continues to update the estimation as new data arrives. Another feature of KF is its fast estimation process due to its simple equations. Therefore, KF is a good option for tracking a dynamic system such as ITS or IPTS. In fact, KF has many applications. These include locating a mobile station in mobile communication [88]; in a power system in harmonic distortion and voltage events [89]; in robot vision [90]; in a navigation system [91][92]; and in finance[93] [70][94][95].

KF is comprised of three necessary calculations: Kalman gain, current estimate, and new error in the estimate. Figure 3.1 illustrates how KF works [94].

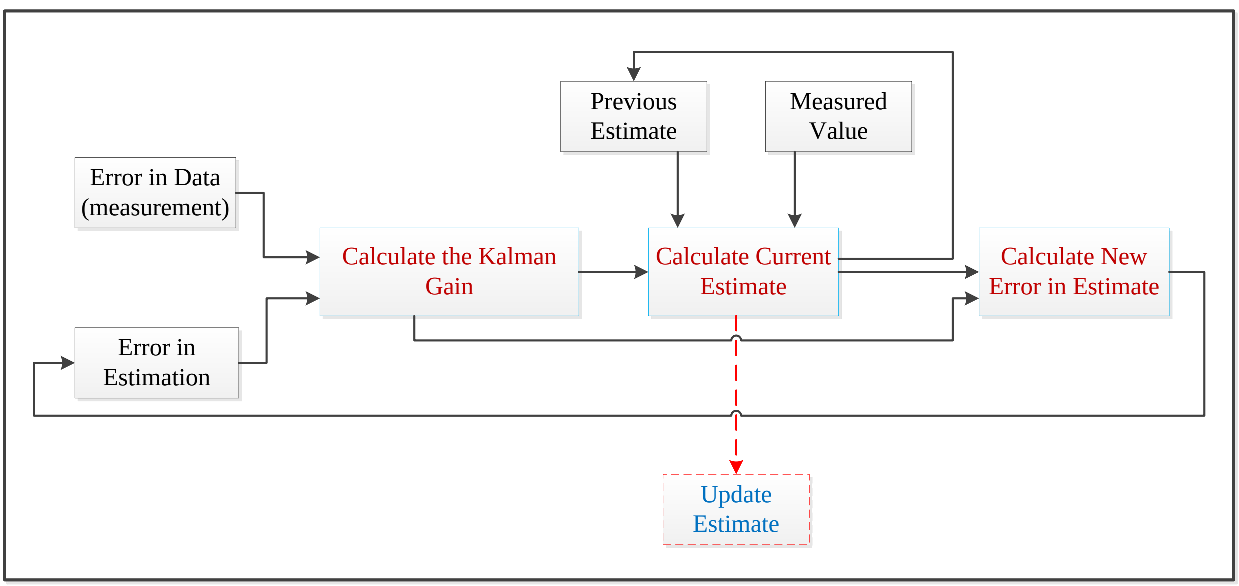

Figure 3.1 General block diagram of Kalman Filter [94]

3.1.1 Kalman Filter Equations

At any time, KF model assumes that the state of a system at time *t* has evolved from the previous state at time *t-1* according to the following equation (3.1)[96]:

$$\boldsymbol{x_t} = \boldsymbol{F_t x_{t-1}} + \boldsymbol{B_t u_t} + \boldsymbol{w_t} \tag{3.1}$$

Where:

x_t is the state vector of the estimation (e.g., position, velocity, heading) at time *t;*

u_t is any control input (e.g. in case of tracking a vehicle speed, u_t is steering angle, throttle setting, braking force) at time *t*;

F_t is the state transition matrix of the process from the state at *t-1* to the state at *t;*

B_t is the control input parameters matrix that affect vector u_t;

w_t is the process's white noise for each parameter in the state vector with known covariance; and

t is time indices with t= 0, 1, 2…

Alternatively, the measurements or observations of the system can be modelled according to the following form:

$$\boldsymbol{z_t} = \boldsymbol{H_t x_t} + \boldsymbol{v_t} \tag{3.2}$$

Where:

z_t is the actual measurement of *x* at time *t;*

H is the transformation matrix from state vector of the estimation into the measurement domain; and

$v_t$is the measurement noise that is assumed to be with zero mean and known covariance.

Figure 3.2 which shows the dynamics and observation (measurement) model, represents equations (3.1) and (3.2).

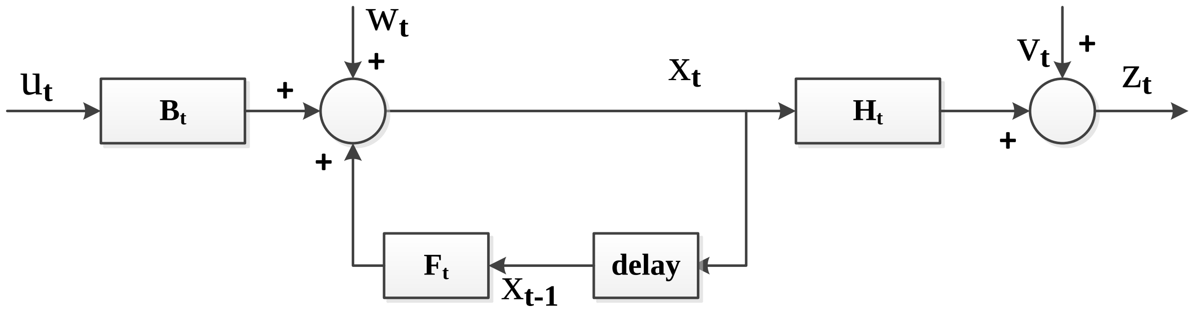

Figure 3.2 Dynamics and observation model

Like any system, the system presented in Figure 3.3 has an input u_t, an output z_t, and the state that needs to be estimated x_t. While equations (3.1) and (3.2) represent this model, they do not reflect the true value x_t because they are measurement noise v_t and process noise w_t. Since this does not reflect the true value of x_t, the mathematical model of the system (prediction) might help to estimate x_t. u_t can be used as the input of the system's mathematical model to give an estimation of the system's output; therefore, the predicted state estimate $\hat{x}_t$ can be calculated. In fact, the estimation is imperfect — not because of the mathematical shortage, but because of the process noise w_t. At this point, KF can play a role in obtaining an estimation of x_t, even if the measurement and process noise are there. By combining z_t and $\hat{x}_t$, this estimation is close to perfect. The optimal estimate is found by multiplying the prediction $\hat{x}_t$ and the measurement z_t probability. The equation(3.3) is the result of this multiplication [96][97].

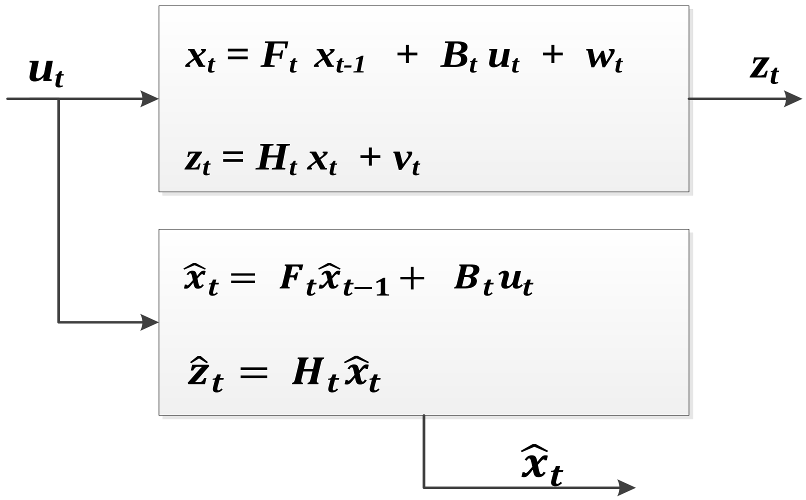

Figure 3.3 Prediction state and measurement state block diagram

$$\boldsymbol{\hat{x}_t = F_t \hat{x}_{t-1} + B_t u_t + G\left(z_t - H_t(F\hat{x}_t + B_t u_t)\right)} \tag{3.3}$$

The first part of equation (3.3) ($\boldsymbol{F_t \hat{x}_{t-1} + B_t u_t}$) predicts the current state by using the state estimation from the previous time step and the current input. This is called the priori estimate because it is calculated before the current measurement is taken. The priori estimate is denoted as $\boldsymbol{\hat{\hat{x}}_t}$. Therefore, equation (3.3) can be rewritten in a new form as in equation (3.4). The second part of equation (3.3) ($\boldsymbol{G\left(z_t - H_t(F\hat{x}_t + B_t u_t)\right)}$) uses the measurement combined with the prediction to update the priori estimate. The result, which is called posteriori estimate, is equation (3.4).

$$\boldsymbol{\hat{x}_t = \hat{\hat{x}}_t + G\,(z_t - H_t \hat{\hat{x}}_t)} \tag{3.4}$$

The process of KF basically consists of the two following parts:

1. Predictions (time update): Predictions are responsible for estimating the a priori state estimate and the covariance of the prior error estimates for the next time step *(t)* from the previous time step *(t-1)*. The time update can be formulated as:

$$\hat{\hat{\boldsymbol{x}}}_{\boldsymbol{t}} = \boldsymbol{F}_{\boldsymbol{t}}\hat{\boldsymbol{x}}_{\boldsymbol{t-1}} + \boldsymbol{B}_{\boldsymbol{t}}\boldsymbol{u}_{\boldsymbol{t}} \tag{3.5}$$

$$\overline{\boldsymbol{P}}_{\boldsymbol{t}} = \boldsymbol{F}_{\boldsymbol{t}}\boldsymbol{P}_{\boldsymbol{t-1}}\boldsymbol{F}^{\boldsymbol{T}} + \boldsymbol{Q} \tag{3.6}$$

Where:

$\overline{P}_t$ is the variance of the a priori estimate;

$\boldsymbol{P}_{\boldsymbol{t}}$ is the estimated state error covariance matrix at time *t*; and

Q is the process noise covariance associated with control input noise.

2. Measurements update (correction): Measurements improve the prediction process by updating the computed Kalman gain, posteriori state estimate, and posteriori state error covariance. The measurement process step uses the priori estimates from equation (3.5) and then updates these estimates by using equation (3.7) to find a posteriori estimation of the state and the error covariance. The equations of the measurements update are as follows:

$$\boldsymbol{G}_{\boldsymbol{t}} = \frac{\overline{\boldsymbol{P}}_{\boldsymbol{t}}\,\boldsymbol{H}^{\boldsymbol{T}}\boldsymbol{H}_{\boldsymbol{t}}^{\boldsymbol{T}}}{\boldsymbol{H}_{\boldsymbol{t}}\,\overline{\boldsymbol{P}}_{\boldsymbol{t}}\,\boldsymbol{H}_{\boldsymbol{t}}^{\boldsymbol{T}} + \boldsymbol{R}_{\boldsymbol{t}}} \tag{3.7}$$

$$\hat{\boldsymbol{x}}_{\boldsymbol{t}} = \hat{\hat{\boldsymbol{x}}}_{\boldsymbol{t}} + \boldsymbol{G}_{\boldsymbol{t}}\,(\boldsymbol{z}_{\boldsymbol{t}} - \boldsymbol{H}_{\boldsymbol{t}}\hat{\hat{\boldsymbol{x}}}_{\boldsymbol{t}}) \tag{3.8}$$

$$\boldsymbol{P}_{\boldsymbol{t}} = \overline{\boldsymbol{P}}_{\boldsymbol{t}} - \boldsymbol{K}_{\boldsymbol{t}}\,\boldsymbol{H}_{\boldsymbol{t}}\,\overline{\boldsymbol{P}}_{\boldsymbol{t}} \tag{3.9}$$

Where:

G_t is Kalman filter gain; and

R_t is the covariance of the measurement noise.

Therefore, the time update forecasts the future state estimate, and then the forecasted estimate is adjusted by actual measurements. In equation (3.8), if the gain is high, KF place more weight on the measurements. In contrast, if the gain is low, KF place more weight on prediction. Figure 3.4 shows the KF model block diagram. In this diagram the upper part is the filter model and the lower part is the filter process [98].

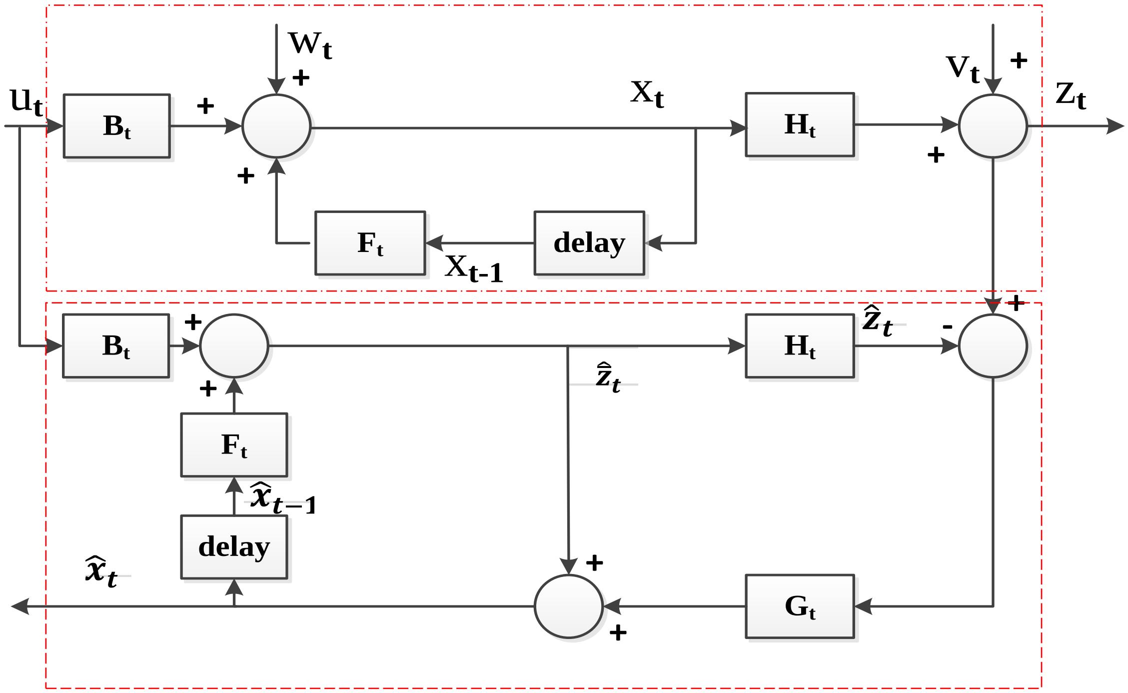

Figure 3.4 KF model block diagram; The upper part is the filter model and the lower part is the filter process.

3.2 The Proposed Travel Time Prediction Model

The bus's total travel time is the summation of the bus's running time between bus stops and the bus's dwell time at these bus stops. To obtain a more accurate travel time prediction, the proposed model predicts the running time and the dwell time separately because the dwell time consumes a significant amount of the bus's travel time. Thus, three algorithms are proposed in this thesis to predict the bus's total travel time. The long-distance prediction algorithm and short-distance prediction algorithm are used to predict the running time. The dwell time algorithm is used to predict the amount of time that the bus spends at each bus stop. Furthermore, these algorithms are KF-based which is reliant on historical data and AVL and APC systems' real-time data. In fact, KF is widely used in many research areas, particularly in the autonomous and navigation fields. KF is advantageous because of its ability to make predictions and adjustments for the model with each new measurement and for its ease of use in terms of computational algorithms. AVL and APC systems keep the model dynamic while strengthening the system to react to any incidents that may occur on the bus route such as bad weather conditions, road construction, and accidents. We assumed, wireless communication means are perfect.

3.2.1 Historical Data

As mentioned above, historical data is one of two sources that the model relies on. When the data centre receives the bus's data through AVL and APC systems, this data is stored in different categories for efficient use. The data that is recorded from AVL systems includes the bus's location, speed, and arrival and departure times at bus stops. On the other hand, the APC system provides data about the number of boarding and alighting passengers and the number of times that the bus wheelchair ramp is used.

It is assumed that the traffic patterns differ from day-to-day during the week; however, the traffic pattern of a day is similar to the traffic pattern of the same day of the previous week. Therefore, the data centre classifies AVL/APC data in the historical database where it is stored in categories. Each category contains the data for one day (i.e. Mondays, Tuesdays, Wednesdays…etc.) and the data of that day is divided into subcategories. Each subcategirt represents a period of time on that day. For example, the data for Mondays is organized into hourly subcategories (e.g., from 01:00 p.m. to 02:00 p.m.). This continues for the rest of the day's time periods and for all of the days of the week.

3.2.2 Bus Running Time Prediction Algorithms

To predict the bus's running time, the model has two algorithms. These are long- and short-distance prediction algorithms. The following two subsections explain these two algorithms.

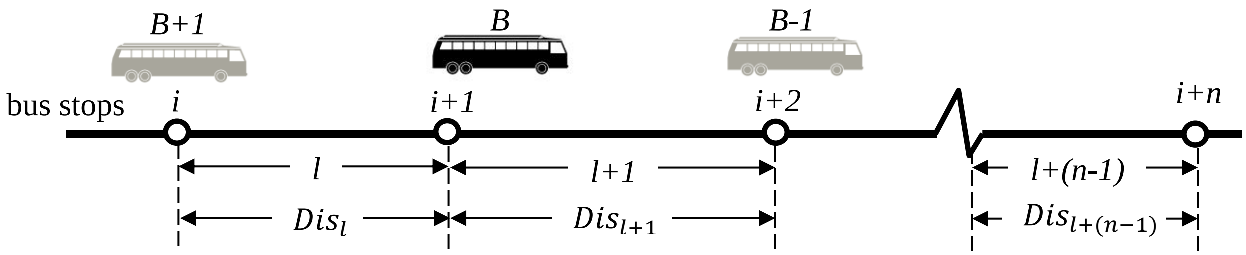

Figure 3.5 A Bus Route Schematic

3.2.2.1 Long-Distance Prediction Algorithm

The moment when a bus, for example bus (*B)*, closes its doors at bus stop (*i+1)* and departs, the long-distance prediction algorithm predicts the running time. The running time is the time the bus takes to travel from one stop to the next stop. This does not include the dwell time of the bus up to the downstream stop. This algorithm uses the historical average speed data for each link (*l+(n-1))* and the observed speed of the previous bus *(B-1)* for the same link on the same day. The link is the interval between two bus stops that are located next to each other as Figure 3.5 illustrates. The KF equations for the running time prediction algorithm have been structured as follows [49],[71] and[99]:

$$\boldsymbol{g_{l+1} = \frac{e_l + VAR_{out}}{VAR_{in} + VAR_{out} + e_l}} \tag{3.10}$$

$$\boldsymbol{a_{l+1} = 1 - g_{l+1}} \tag{3.11}$$

$$\boldsymbol{e_{l+1} = VAR_{in} * g_{l+1}} \tag{3.12}$$

$$\boldsymbol{PS_{l+1} = a_{l+1} * obs_{B-1,l+1} + g_{l+1} * HAS_{l+1}} \tag{3.13}$$

Where:

g is the filter-gain;

e is the filter-error;

VAR_{out} is the prediction variance;

VAR_{in} is the historical data variance;

a is the loop-gain;

$PS_{l+1}$is the predicted speed for the link (*l+1*);

$obs_{B-1,l+1}$ is the observed "actual" speed of bus *(B-1)* at link (*l+1*);

HAS_{l+1} is the average speed of the historical data (for the same day and time period) for the link (*l+1)*; and

n is the number of the bus stops on the route.

Since each link has its own historical bus speed data which differs from any other link, VAR_{in} must be calculated for each link by using the data of the previous days.

$$\boldsymbol{VAR_{in} = VAR\ [\ hs_1, hs_2, hs_3, \cdots, hs_N\ ]} \tag{3.14}$$

Where:

hs is the observed speed for the link from the previous days; and

N is the number of historical days.

For a random variable, the variance is defined as follows:

$$\boldsymbol{E(X) = E[(X - E[X])^2]} \tag{3.15}$$

$$\boldsymbol{E(X) = HAS = \frac{hs_1 + hs_2 + hs_3 + \cdots + hs_N}{N}} \tag{3.16}$$

In order to calculate $\boldsymbol{VAR_{in}}$, the Δ for each historical day's speed *N* must be calculated as follows:

$$\Delta_{\boldsymbol{1}} = \boldsymbol{(hs_1 - HAS)^2} \tag{3.17}$$

$$\Delta_{\boldsymbol{2}} = \boldsymbol{(hs_2 - HAS)^2} \tag{3.18}$$

$$\vdots$$

$$\Delta_{\boldsymbol{N}} = \boldsymbol{(hs_N - HAS)^2} \tag{3.19}$$

Now VAR_{in} is calculated as in the following equation:

$$\boldsymbol{VAR_{in}} = \frac{\Delta_{\mathbf{1}} + \Delta_{\mathbf{2}} + \Delta_{\mathbf{3}} + \cdots + \Delta_{\boldsymbol{N}}}{\boldsymbol{N}} \tag{3.20}$$

Now, VAR_{out} is the only one left without calculation because VAR_{out} depends on the variance of the prediction and the corresponding future observation which are not yet available. To obtain these data, both the prediction and the trip must be made. However, once they have been made, then there is no need to predict the running time. Thus, if the prediction is a good prediction, then the VAR_{out} equals the VAR_{out} and they are equal to *VAR*. Therefore, equations (3.10) and (3.12) can be formulated as:

$$\boldsymbol{g_{l+1}} = \frac{\boldsymbol{e_l + VAR}}{\boldsymbol{e_l + 2 \times VAR}} \tag{3.21}$$

$$\boldsymbol{e_{l+1} = VAR * g_{l+1}} \tag{3.22}$$

Now, when the bus closes its doors before departing a bus stop, the long-distance prediction algorithm predicts the running time between each coming link to the downstream stop by using equations (3.21), (3.11),(3.22) and (3.13) respectively as follows:

$$\boldsymbol{PRT_{B,(i+1,i+2)}} = \frac{\boldsymbol{Dis_{l+1}}}{\boldsymbol{PS_{l+1}}} \tag{3.23}$$

Where:

$PRT_{B,(i+1,i+2)}$ is the predicted running time from bus stop *(i+1)* to *(i+2)* for bus *B*; and

Dis_{l+1} is the fixed distance between two bus stops *(i+1)* and bus stop *(i+2)*.

Thus, the bus's predicted arrival time at the next bus stop depends on the actual departure time of the bus from the previous bus stop which AVL system provides. The following equation calculates the bus's predicted arrival time at the next bus stop:

$$\boldsymbol{PAT_{B,(i+2)} = ADT_{B,(i+1)} + PRT_{B,(i+1,i+2)}} \tag{3.24}$$

Where:

$PAT_{B,(i+2)}$ is the predicted arrival time of bus *B* at bus stop *(i+2)*; and

$ADT_{B,(i+1)}$ is the actual departure time of bus *B* from bus stop *(i+1)*.

3.2.2.2 Short-Distance Prediction Algorithm

The short-distance prediction algorithm predicts the bus's running time from its current location to the upcoming bus stop. Specifically, the AVL systems that are installed in every bus frequently send updated data on the bus's location to the control centre (i.e. every 15 or 20 seconds) and this time denotes t_{update}. The interval period between two updates is called a mini link *(ml)*, as shown in Figure 3.6, so every link *(l)* has many *(ml)*. Then, the KF-based algorithm predicts the time remaining before reaching the next bus stop.

After t_{update} of the bus leaving a bus stop, the AVL system sends an update ($N_{AVL\ update}$) of the bus's location to the control centre. The algorithm calculates the past distance which is the distance from the last bus stop *(i)* or the bus's location at the time of the last update to the bus's location when it sends the update. At this moment the last mini link's average speed (observed average speed of the last mini link) can be calculated. Also, the observed average speed of the past mini link/links can be calculated. The algorithm uses the last mini link's observed speed as the observed speed and the average speed/speeds of the past mini link/links as the historical average speed. After that, the model predicts the remaining time for the link's remaining distance. The follow equations show how the algorithm works (assuming that the bus is at the 2^{nd} *AVL update* heading toward bus stop $i+1$).

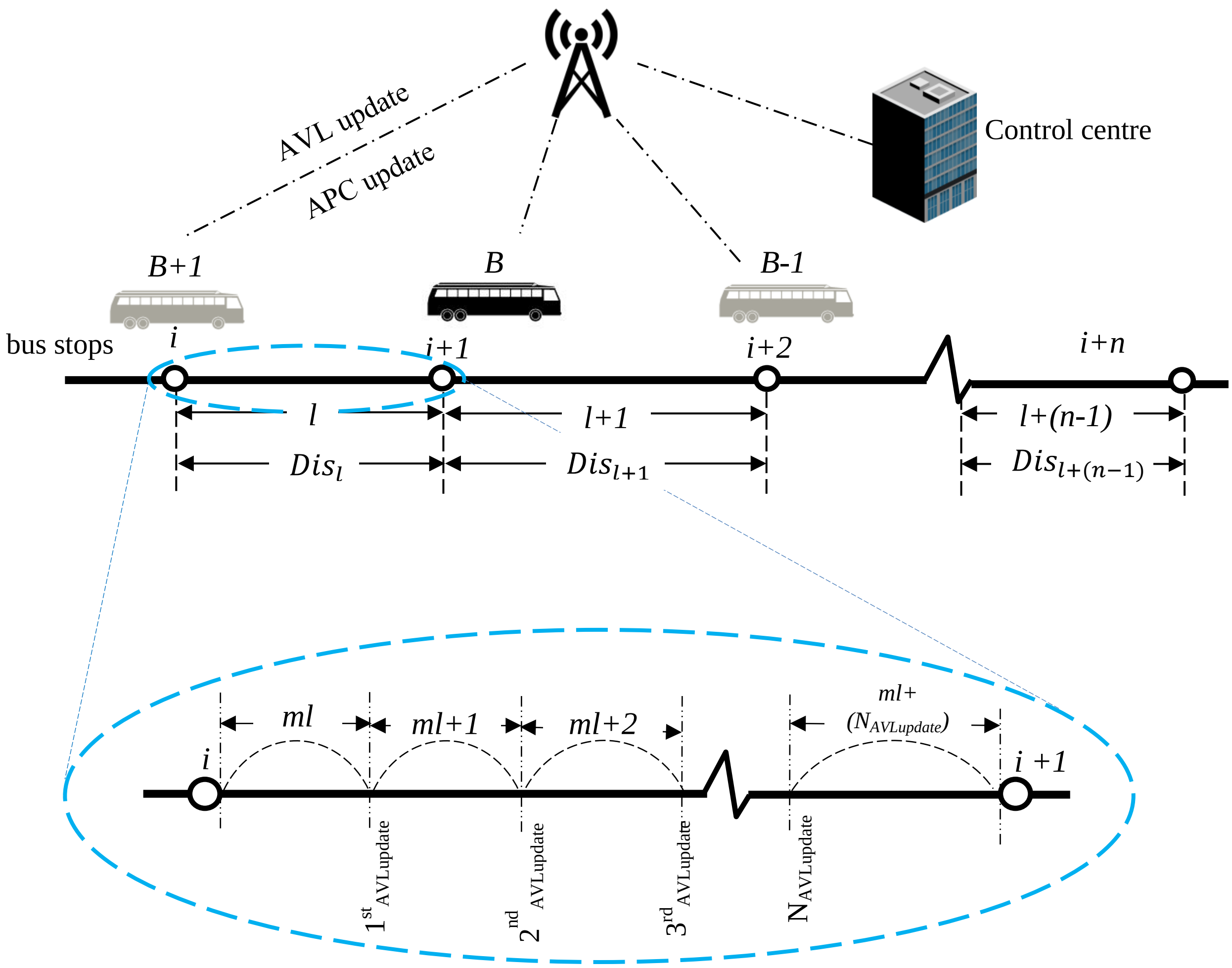

Figure 3.6 Mini Links (ml) Schematic

$$g_{rd} = \frac{e_{ml+1} + VAR}{e_{ml+1} + 2 \times VAR} \tag{3.25}$$

$$e_{ml+1} = VAR * g_{rd} \tag{3.26}$$

$$a_{rd} = 1 - g_{rd} \tag{3.27}$$

$$PS_{rd} = a_{rd} * obs_{ml+1} + g_{rd} * OAS \tag{3.28}$$

Where:

PS_{rd} is the predicted speed of the remaining distance to the next bus stop (from *2nd AVL update* to the *(i+1)*);

obs_{ml+1} is the observed average speed of the last mini link (*ml+1*); and

OAS is the average speed of the entire past mini link/links *(ml & ml+1)* speed.

The observed average speed for the last mini link (obs_{ml+1}) can be calculated in the following steps. The AVL system provides the past distance from the last bus stop to the bus's current location ($Pdis_{2nd_{AVL\,update}}$). The distance of the last mini link can be formulated as follows:

$$\boldsymbol{dis_{ml+1} = Pdis_{2nd_{AVL\,update}} - Pdis_{1st_{AVL\,update}}} \quad (3.29)$$

Where:

dis_{ml+1} is the distance of *(ml+1)*.

Thus,

$$\boldsymbol{obs_{ml+1} = \frac{dis_{ml+1}}{t_{update}}} \quad (3.30)$$

Where:

t_{update} is the frequent time that AVL sends an update to the control centre.

Then, the variance (VAR) can be calculated as follows:

$$\boldsymbol{VAR = VAR\left[\, obs_{ml}, obs_{ml+1}, obs_{ml+2}, \cdots, obs_{ml+NAVLupdate} \,\right]} \quad (3.31)$$

The average speed for all past mini links is calculated as:

$$\boldsymbol{OAS = \frac{obs_{ml} + obs_{ml+1} + obs_{ml+2} + \cdots + obs_{ml+NAVLupdate}}{N_{AVL\,update}}} \quad (3.32)$$

To calculate *VAR*, the Δ for each past mini link/links has to be calculated as in equations (3.33) to (3.36). However, Δ_1 will equal zero because the observed speed at (obs_{ml}) equals *OAS* at the *1st* $_{AVL\ update}$. Therefore, obs_{ml} will be equal to the average of obs_{ml} and the historical average speed of the entire link *(HAS)*.

$$\Delta_1 = \left(\frac{\boldsymbol{HAS} + \boldsymbol{obs_{ml}}}{\boldsymbol{2}} - \boldsymbol{OAS}\right)^{\boldsymbol{2}} \tag{3.33}$$

$$\Delta_2 = (\boldsymbol{obs_{ml+1}} - \boldsymbol{OAS})^{\boldsymbol{2}} \tag{3.34}$$

$$\Delta_3 = (\boldsymbol{obs_{ml+2}} - \boldsymbol{OAS})^{\boldsymbol{2}} \tag{3.35}$$

$$\vdots$$

$$\Delta_{N_{AVL\,update}} = \left(\boldsymbol{obs_{ml+NAVLupdate}} - \boldsymbol{OAS}\right)^{\boldsymbol{2}} \tag{3.36}$$

Thus,

$$\boldsymbol{VAR} = \frac{\Delta_1 + \Delta_2 + \Delta_3 + \cdots + \Delta_{N_{AVL\,update}}}{\boldsymbol{N_{AVL\,update}}} \tag{3.37}$$

Now, the speed for the remaining distance PS_{rd} from the bus's current location to the next stop can be predicted by using equation (3.28). This means that the predicted running time to the next bus stop can be also calculated. However, the algorithm first checks the remaining distance $Rdis$ to the next bus stop *(i+1)* equation (3.38). If the remaining distance is greater than zero, then the algorithm predicts the running times for the remaining distance PRT. Otherwise, the bus has already arrived at the next bus stop. The algorithm also checks the $N_{AVL\,update} \times t_{update}$. If $N_{AVL\,update} \times t_{update}$ becomes greater than the bus's headway, then the algorithm considers the bus's status to be out of control.

$$\boldsymbol{Rdis_{N_{AVL\,update},i+1}} = \boldsymbol{Dis_l} - \boldsymbol{Pdis_{2nd_{AVL\,update}}} \tag{3.38}$$

$$\boldsymbol{PRT_{N_{AVL\,update},i+1}} = \frac{\boldsymbol{Rdis_{N_{AVL\,update},i+1}}}{\boldsymbol{PS_{rd}}} \tag{3.39}$$

When the bus arrives at the next stop, the summation of $obs_{ml+NAVLupdate}$ is the observed speed for the link *(l)*. This speed is recorded on the database as historical to be used as hs_N in next week's prediction for the same day and time period.

3.2.3 Bus Dwell Time Prediction Algorithm

The dwell times for a bus at the upcoming bus stops are predicted every time the bus closes its doors and departs a bus stop. The algorithm predicts the number of boarding and alighting passengers and the number of passengers with wheelchairs. The algorithms use the historical data from the same day and time period from the previous weeks and the data from the last trip on the same day to predict the dwell time. The following equations show how the dwell time prediction algorithm works:

$$\boldsymbol{g_{i+1} = \frac{e_i + VAR}{e_i + 2 \times VAR}} \tag{3.40}$$

$$\boldsymbol{e_{i+1} = VAR * g_{i+1}} \tag{3.41}$$

$$\boldsymbol{a_{i+1} = 1 - g_{i+1}} \tag{3.42}$$

$$\boldsymbol{PPAR_{i+1} = a_{i+1} * obs_{B-a.i+1} + g_{i+1} * HAP_{i+1}} \tag{3.43}$$

Where:

$PPAR_{i+1}$ is the predicted passenger arrival rate to the bus stop;

$obs_{B-a.i+1}$ is the actual passenger arrival rate for the previous bus *(B-1)* at bus stop *(i+1);* and

HAP_{i+1} is the historical average passenger arrival rate (for the same day and time period) at bus stop *(i+1).*

Now that the passenger arrival rate to the bus stop has been predicted, it is possible to estimate the number of boarding passenger after knowing the bus's headway. The following equation, (3.44) , is used to predict the bus's headway:

$$\boldsymbol{PH_{B-1,B} = PAT_{B,(i+2)} - AAT_{B-1,(i+2)}} \tag{3.44}$$

Where:

$PH_{B-1,B}$ is the predicted headway between bus *B-1* and *B* at bus stop *(i+2)*; and

$AAT_{B-1,(i+2)}$ is the actual arrival time of the previous bus *B-1* at the bus stop *(i+2)*.

The predicted number of passengers that board PPN_{i+2} at bus stop *(i+2)* is expressed in the following equation:

$$\boldsymbol{PPN_{i+2} = PPAR_{i+2} \times PH_{B-1,B}} \tag{3.45}$$

Thus, predicting the arrival rate of passengers who use wheelchairs follows the same method that is used to predict the arrival rate of other passengers but with a change in the input of the algorithm. The input data of wheelchair passengers' arrival rate prediction *(Pw)* consists of the historical data of the wheelchair passengers' arrival rate to the bus stop and the observed wheelchair passengers' arrival rate data on the last trip. Moreover, every time the bus's wheelchair ramp is used, the APC system records a passenger with a wheelchair boarding or alighting the bus.

The prediction for boarding passengers with wheelchairs $PPNw_{i+2}$ at bus stop *(i+2)* is calculated by the following equation:

$$\boldsymbol{PPNw_{i+2} = Pw_{i+2} \times PH_{B-1,B}} \tag{3.46}$$

Also, the prediction for alighting passengers without wheelchairs is calculated by the following equations:

$$\boldsymbol{g_{i+1} = \frac{e_i + VAR}{e_i + 2 \times VAR}} \tag{3.47}$$

$$\boldsymbol{e_{i+1} = VAR * g_{i+1}} \tag{3.48}$$

$$\boldsymbol{a_{i+1} = 1 - g_{i+1}} \tag{3.49}$$

$$\boldsymbol{PAP_{i+1} = a_{i+1} * obs_{B-a.i+1} + g_{i+1} * HAAP_{i+1}} \tag{3.50}$$

Where:

PAP_{i+1} is the predicted number of alighting passengers at bus stop (i+1);

$obs_{B-a.i+1}$ is the observed number of alighting passengers at bus stop (i+1) from the previous bus (B-1); and

$HAAP_{i+1}$ is the historical average number of alighting passengers at bus stop (i+1).

To calculate the predicted number of alighting passengers who use wheelchairs *APw*, the same equations are used by changing the input data.

Due to the bus's limited capacity, the bus may eventually run out of space. The model calculates the capacity of the bus as follows, assuming that each wheelchair is equal to 3 passengers.

$$\boldsymbol{S_{B,(i+2)} = S_{B,(i+1)} - (PAP_{i+2} + 3 * APw_{i+2}) + (PPN_{i+2} + 3 * PPNw_{i+2})} \tag{3.51}$$

$$\boldsymbol{Pass_{left\,(i+2)}} = \begin{cases} \boldsymbol{0} & , \quad \boldsymbol{S_{B,(i+2)} \leq C_B} \\ \boldsymbol{S_{B,(i+2)} - C_B} & , \quad \boldsymbol{S_{B,(i+2)} > C_B} \end{cases} \tag{3.52}$$

Where:

$S_{B,(i+1)}$ is the bus's *B* actual number of passengers when it leaves the bus stop *(i+1)*;

C_B is the bus's *B* maximum capacity; and

$Pass_{left}$ is the number of passengers that are left behind at the bus stop due to the bus being at full capacity.

Then, when the bus departs the bus stop, the actual number of passengers *(i+2)* is:

$$\boldsymbol{S_{B,(i+2)}} = \begin{cases} \boldsymbol{S_{B,(i+2)}} & , \quad \boldsymbol{S_{B,(i+2)} \leq C_B} \\ \boldsymbol{C_B} & , \quad \boldsymbol{S_{B,(i+2)} > C_B} \end{cases} \tag{3.53}$$

Now, the dwell time prediction is the summation of the service time at the busiest door for the alighting and boarding passengers. At each door, the alighting serving time is calculated as follows:

$$\boldsymbol{AST_{m,(i+2)} = (PAP_{i+2} \times \beta) + (APw_{i+2} \times \beta_w)} \tag{3.54}$$

Where:

m is the door number;

$AST_{m,(i+2)}$ is the alighting serving time for passengers at door m at bus stop *(i+2)*;

β is the alighting serving time for passengers; and

β_w is the alighting serving time for passengers with wheelchairs.

Then, the serving time for boarding passengers at each door is calculated as follows:

$$\boldsymbol{BST_{m,(i+2)} = \big((PPN_{i+2} - Pass_{left}) \times \rho\big) + (PPNw_{i+2} \times \rho_w)} \tag{3.55}$$

Where:

$BST_{m,(i+2)}$ is the boarding serving time for passengers at door m at bus stop *(i+2)*;

ρ is the boarding serving time for passengers; and

ρ_w is the boarding serving time for passengers with wheelchairs.

The dwell time $DT_{B,(i+2)}$ for bus B at bus stop *(i+2)* is:

$$\boldsymbol{DT_{B,(i+2)} = \sigma + \max_{1-m}\{BST_{m,(i+2)} + AST_{m,(i+2)}\}} \tag{3.56}$$

Where:

σ is the length of time for the door to open and close.

3.3 How the Model Works

This section shows how previous algorithms work together to predict the bus's total running time to the downstream stop and how the model adjusts the prediction.

Now, from the two previous sections, the predicted running time and the dwell time are predicted, so the total running time $TRT_{B,(i+1,i+2)}$ from a bus stop (i.e. *(i+1)*) to when the bus closes its doors and departs from the next bus stop *(i+2)* can be calculated. The total running time is the summation of the predicted running time between the two stops $PRT_{B,(i+1,i+2)}$ and the dwell time at the stop $DT_{B,(i+2)}$.

$$\boldsymbol{TRT_{B,(i+1,i+2)} = PRT_{B,(i+1,i+2)} + DT_{B,(i+2)}} \tag{3.57}$$

Next, the predicted departure time $PDT_{B,(i+2)}$ for bus B from bus stop *(i+2)* can be predicted by the following equation:

$$\boldsymbol{PDT_{B,(i+2)} = PAT_{B,(i+2)} + \ DT_{B,(i+2)}} \tag{3.58}$$

The model follows the same process for the upcoming bus stops. In other words, the model predicts the running time for the link *(l)* and the dwell time at the bus stop *(i+1)* that is located at the end of the link. Next, the model calculates the next link *(l+1)* and the dwell time at the bus stop that is located at the end of this link. This process continues for all of the upcoming links and stops. The model repeats this process every time the bus departs a bus stop. While the bus is running from one bus stop to the next bus stop, the model keeps adjusting the running time prediction to the next stop using short-distance prediction algorithm. The total running time is then updated.

Chapter 4

4 Simulation Results

4.1 Introduction

Using the model proposed in Chapter 3, this chapter presents the prediction of the travel time's results. The simulation environment was built using MATLAB. Section 4.2 presents the simulation sitting and assumptions, section 4.3 presents the performance results, and section 4.4 presents the results' validation.

4.2 Simulation Setting and Assumptions

The bus simulated route is shown in Figure 4.1. The length and the maximum speed of each link are presented in Table 4.1. The route is assumed to be located in a busy area (i.e. downtown) and the bus is assumed to be running during peak time (07:45:00 AM). The bus headway is also assumed to be five minutes (300 seconds). The assumptions are based on a publish data from the bus service provider. The incidents that might occur on the bus's route are assumed to follow Poisson distribution.

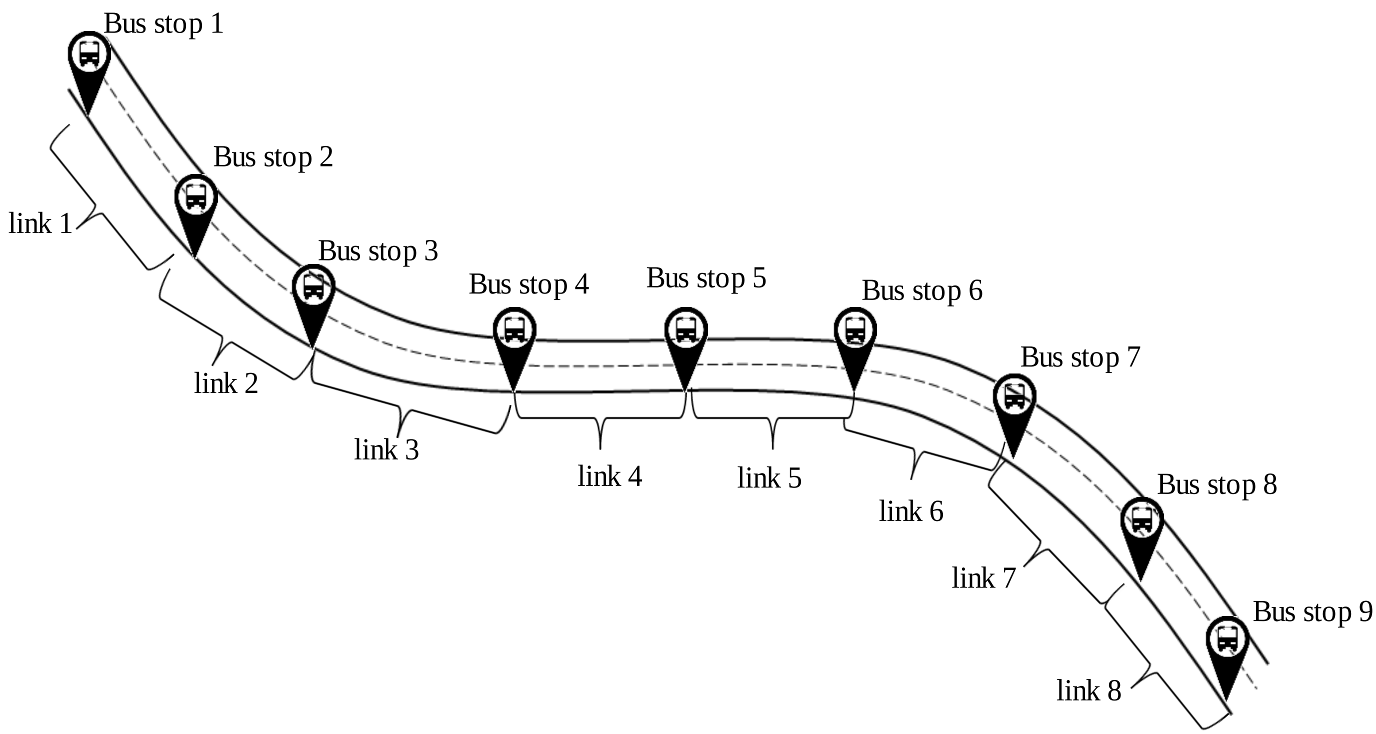

Figure 4.1 Map of the bus route

Table 4.1 Length and the maximum speed of each link

Link	Length (m)	Maximum speed (km/h)
Link 1 (from bus stop 1 to bus stop 2)	650	40
Link 2 (from bus stop 2 to bus stop 3)	700	60
Link 3 (from bus stop 3 to bus stop 4)	2100	80
Link 4 (from bus stop 4 to bus stop 5)	800	60
Link 5 (from bus stop 5 to bus stop 6)	980	60
Link 6 (from bus stop 6 to bus stop 7)	870	60
Link 7 (from bus stop 7 to bus stop 8)	500	40
Link 8 (from bus stop 8 to bus stop 9)	790	40

Based on *Transit Capacity and Quality of Service Manual*, the parameters presented in Table 4.2 were used in the simulation.

Table 4.2 Simulation Parameters

Parameter	Value
Bus Capacity	Bus length 60 ft., low floor level, 3 passenger doors, 64 seated passengers, and 24 standees
Alighting passengers	25% of alighting passengers leave the bus from the front door and the remainder of alighting passengers leave the bus from the two rear doors equally.
Boarding passengers	Passengers board equally from all doors.
Serving time for alighting passengers	2.5 seconds per passenger alighting from the front door, and 1.75 seconds per passenger alighting from the rear doors
Serving time for boarding passenger	2.75 seconds per passenger for all doors
Serving time for alighting and boarding passengers with wheelchairs	30 to 45 seconds
Time for the doors to open and close	4 seconds

4.3 Performance Results

The model is comprised of two main parts, namely long- and short-distance predictions. The model starts with the long-distance prediction which is then adjusted by the short-distance prediction. Therefore, the long-distance prediction is presented here first followed by the short-distance prediction. As the long-distance prediction is a combination of the running time and the dwell time, the running time prediction results are presented as long- and short-distance predictions.

4.3.1 Running Time Prediction

The running time is the length of time it takes the bus to travel from one bus stop to another. The dwell time is temporarily ignored and the bus's headway is assumed to be 5 minutes (300 seconds). Table 4.3 shows the length of time that the bus takes to arrive at each downstream bus stop (i.e. bus stop 2 or bus stop 3) when the bus is at bus stop 1. The running time to any bus stop on the downstream is the accumulated running time of the links that are located between the bus's location and the destination bus stop. For example, the predicted running time for the bus to travel from bus stop 1 to bus stop 4, as shown in Table 4.3, is 206.23 seconds which is the summation of the running time of link 1 (61.45 seconds), link 2 (104.80 – 61.45 seconds), and link 3 (206.23 – 104.80 seconds). Table 4.4, Table 4.5, Table 4.6, Table 4.7, Table 4.8, Table 4.9 and Table 4.10 show also the running times from bus stops 2, 3, 4, 5, 6, 7 and 8 to downstream bus stops/stop.

Table 4.3 Running time from bus stop 1 to the downstream bus stops

Running time from bus stop 1	Running time (seconds)
to bus stop 2	61.45
to bus stop 3	104.80
to bus stop 4	206.23
to bus stop 5	259.46
to bus stop 6	322.29
to bus stop 7	378.94
to bus stop 8	427.31
to bus stop 9	511.19

To predict the running time from bus stop 2 to the downstream bus stops, the same data that are used to predict the running time from bus stop 1 to the downstream bus stops are used again because there are no changes between the historical data and the observed data.

Table 4.4 Running time from bus stop 2 to the downstream bus stops

Running time from bus stop 2	Running time (seconds)
to bus stop 3	43.35
to bus stop 4	144.78
to bus stop 5	198.00
to bus stop 6	265.93
to bus stop 7	322.59
to bus stop 8	370.96
to bus stop 9	454.84

However, the data changes at some point due to the arrival of new data (the observed data of the previous bus). The links and bus stops that were located more than the headway (300 seconds) from the bus's location are now located less than 300 seconds from the bus's location. The data of this link has been changed because of the newly arrived data of the previous bus. Therefore, the model must predict the running time in that link.

The bus's predicted running time from bus stop 1 (Table 4.3) to bus stop 6 is 322.29 seconds (more than the bus's headway). The predicted running time becomes 265.93 seconds (less than the bus's headway) when the bus is at bus stop 2 (Table 4.4). Due to the change in data, the running time for link 5 is change from 62.83 seconds (322.29 – 259.46) to 67.93 seconds (265.93 – 198).

The running time for link 6 is 56.66 seconds (322.59 – 265.93) when the bus is at bus stop 2 (Table 4.4) When the newly observed data arrives, the running time of the link is updated to become 59.66 seconds (282.24 – 222.58) as shown in Table 4.5.

Table 4.5 Running time from bus stop 3 to the downstream bus stops

Running time from bus stop 3	Running time (seconds)
to bus stop 4	101.43
to bus stop 5	154.66
to bus stop 6	222.58
to bus stop 7	282.24
to bus stop 8	330.62
to bus stop 9	414.49

Table 4.6 Running time from bus stop 4 to the downstream bus stops

Running time from bus stop 4	Running time (seconds)
to bus stop 5	53.23
to bus stop 6	121.16
to bus stop 7	180.82
to bus stop 8	235.07
to bus stop 9	318.95

The running time for link 7 is 48.38 seconds (330.62 – 282.24) when the bus is at bus stop 3 (Table 4.5). When the newly observed data arrives, the running time of the link is updated to become 54.25 seconds (235.07 – 180.82) as shown in Table 4.6.

Table 4.7 Running time from bus stop 5 to the downstream bus stops

Running time from bus stop 5	Running time (seconds)
to bus stop 6	67.93
to bus stop 7	127.59
to bus stop 8	181.84
to bus stop 9	262.25

The running time for link 8 is 83.88 seconds (318.95 – 235.07) when the bus is at bus stop 4 (Table 4.6). When the newly observed data arrives, the running time of the link is updated to become 80.41 seconds (262.25 – 181.84) as shown in Table 4.7.

Table 4.8 Running time from bus stop 6 to the downstream bus stops

Running time from bus stop 6	Running time (seconds)
to bus stop 7	59.66
to bus stop 8	113.91
to bus stop 9	194.32

Table 4.9 Running time from bus stop 7 to the downstream bus stops

Running time from bus stop 7	Running time (seconds)
to bus stop 8	54.25
to bus stop 9	134.66

Table 4.10 Running time from bus stop 8 to the downstream bus stop

Running time from bus stop 8	Running time (seconds)
to bus stop 9	80.41

4.3.2 Long-Distance Prediction

Every time the bus departs a bus stop, the long-distance prediction predicts the travel time to the downstream stops. The following scenario is used in the simulation to obtain the results. The bus's headway is five minutes (300 seconds), which means that every five minutes a bus starts running on the route. The bus (B) starts running at 07:50:00 AM and the model predicts its travel time to the downstream stop (bus stop 6). The previous bus (B-1), which is the bus that is ahead of B, starts running at 07:45:00 AM.

Table 4.11 Long-distance prediction for bus B when it departs bus stop 1

Bus B at stop 1	Bus B's predicted travel time to the downstream stop	Previous bus's (B-1) observed travel time to the downstream stop
Departure from stop 1 "observed"	*7:50:37*	7:45:26
Arrival at stop 2	7:51:40	7:46:37
Departure from stop 2	7:52:21	7:47:12
Arrival at stop 3	7:53:06	7:47:55
Departure from stop 3	7:53:41	7:48:22
Arrival at stop 4	7:55:39	7:50:47
Departure from stop 4	7:55:52	7:51:17
Arrival at stop 5	7:56:45	7:52:10
Departure from stop 5	7:56:58	7:52:29
Arrival at stop 6	7:58:01	7:53:31
Departure from stop 6	7:58:14	7:53:44

Table 4.11 shows the predicted the travel time for bus (B) from the moment when it is ready to depart bus stop 1 to the moment when it departs the downstream bus stop (bus stop 6). Table 4.11 also shows the previous bus's (B-1) travel time to the same stops. The departure time from bus stop 1 (7:50:37) is the observed time that is known by AVL/APC systems; however, the remaining arrival times and departure times are predicted times. On the other hand, the previous bus's (B-1) travel times are observed times. It is evident that the previous bus's data is actually from two buses. When bus B departs bus stop 1 at 7:50:37, the previous bus B-1 has not yet arrived at bus stop 4 as it is currently running on link 3. While the first part of the data (from departure from stop 1 to arrival at stop 3) is bus B-1's data, the second part of the data (from bus

B-1's arrival at stop 3 to departure from stop 6) is bus B-2's data. In fact, these data are considered to be from the previous bus because B-2 is the last bus to have travelled there.

Table 4.12 Long-distance prediction for bus B when it departs bus stop 2

Bus B at stop 2	Bus B's predicted travel time to the downstream stop	Previous bus's (B-1) observed travel time to the downstream stop
Departure from stop 2 **"observed"**	*7:52:22*	7:47:12
Arrival at stop 3	7:53:07	7:47:55
Departure from stop 3	7:53:42	7:48:22
Arrival at stop 4	7:55:24	7:50:04
Departure from stop 4	7:55:37	7:50:34
Arrival at stop 5	7:56:30	7:51:25
Departure from stop 5	7:56:43	7:51:44
Arrival at stop 6	7:57:46	7:52:46
Departure from stop 6	7:57:59	7:52:59

Table 4.12 presents the predicted travel time to all downstream bus stops when bus B departs bus stop 2, while (7:52:22) is its observed departure. In Table 4.11, the departure is predicted to be (7:52:21). The most recent last prediction was completed at (7:50:37) when the bus departed bus stop 1. Now the new prediction is processed. Meanwhile, some changes in the previous bus's data between (7:50:37) and (7:52:22) are used in the prediction. In Table 4.11 the running times for link 3 are 145 seconds for the previous bus and 118 seconds for the predicted travel time. Due to the fact that the previous bus is still running, there is a change in the previous bus's data: the running time of link 3 is 102 seconds. According to this change, the predicted

travel time becomes 102 seconds. The observed data that are received by AVL/APC systems shows that the bus B takes 107 seconds to run link 3 as it is shown in Table 4.13.

Table 4.13 Long-distance prediction for bus B when it departures bus stop 3

Bus B at stop 3	Bus B's predicted travel time to the downstream stop	Previous bus's (B-1) observed travel time to the downstream stop
Departure from stop 3 "observed"	*7:53:42*	7:48:22
Arrival at stop 4	7:55:24	7:50:04
Departure from stop 4	7:55:37	7:50:34
Arrival at stop 5	7:56:30	7:51:25
Departure from stop 5	7:56:43	7:51:44
Arrival at stop 6	7:57:47	7:52:47
Departure from stop 6	7:58:02	7:53:00

The observed departure time for (7:53:42) bus B from bus stop 3, as shown in Table 4.13, followed the prediction presented in Table 4.12. The most recent prediction was made at (7:52:22) and the new prediction will be made at (7:53:42) if there is a change on the previous bus's data. The running time of link 5 is changes, as does the dwell time at bus stop 6.

Table 4.14 Long-distance prediction for bus B when it departs bus stop 4

Bus B at stop 4	Bus B's predicted travel time to the downstream stop	Previous bus's (B-1) observed travel time to the downstream stop
Departure from stop 4 "observed"	*7:55:42*	7:50:34
Arrival at stop 5	7:56:35	7:51:25
Departure from stop 5	7:56:48	7:51:44
Arrival at stop 6	7:57:52	7:52:47

Departure from stop 6	7:58:07	7:53:00

When bus B departs bus stop 3, the previous bus B-1 has already passed bus stop 6 and so there is no change to the previous bus's data. Hence, there is no change to the prediction made at bus stops 4 and 5 as presented in Table 4.14 and Table 4.15.

Table 4.15 Long-distance prediction for bus B when it departs bus stop 5

Bus B at stop 5	**Bus B's predicted travel time to the downstream stop**	**Previous bus's (B-1) observed travel time to the downstream stop**
Departure from stop 5 "observed"	*7:57:15*	7:51:44
Arrival at stop 6	7:58:19	7:52:47
Departure from stop 6	7:58:34	7:53:00

Table 4.16 Long-distance prediction for bus B when it departs bus stop 6

Bus B at stop 6	**Bus B's observed departure time from stop 6**	**Previous bus's (B-1) observed departure time from stop 6**
Departure from stop 6	7:58:32	7:53:00

Figure 4.2 shows bus B's predicted travel time for each prediction of the model. Figure 4.3 shows bus B's predicted travel time from bus stop 1 to the downstream stop in comparison to the observed travel time for the same trip. The confidence interval (CI) is very small that it would be not seen clearly on the curve and the method for calculating the CIs is presented in Appendix A.

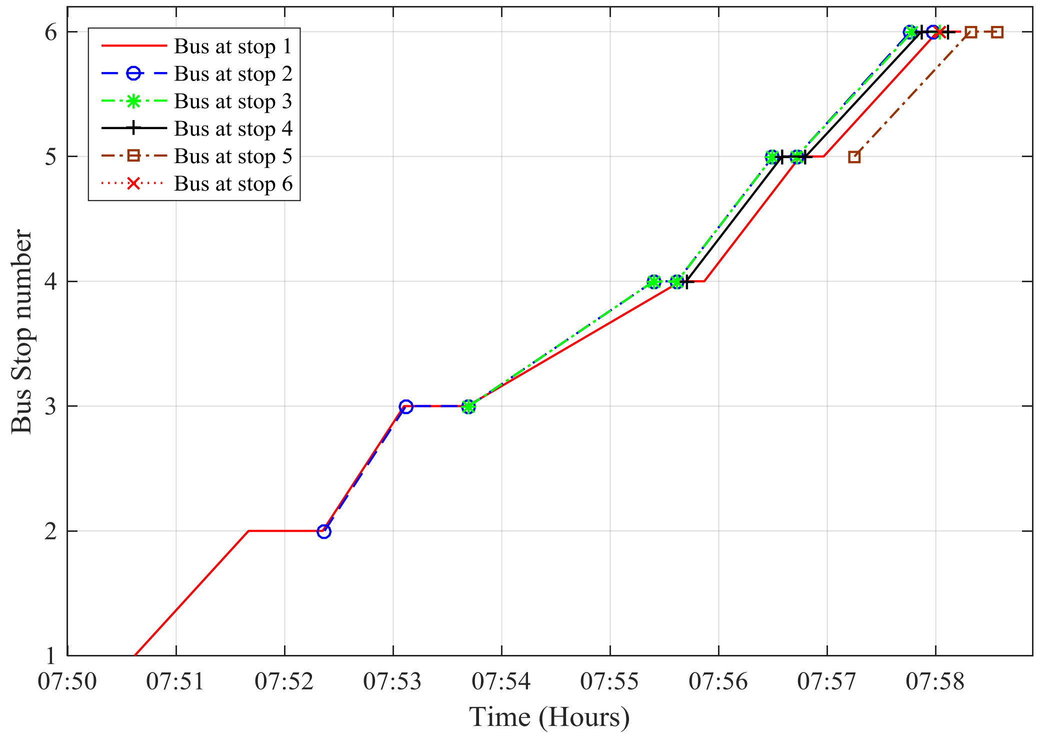

Figure 4.2 Bus B's predicted travel time at all of the downstream bus stops

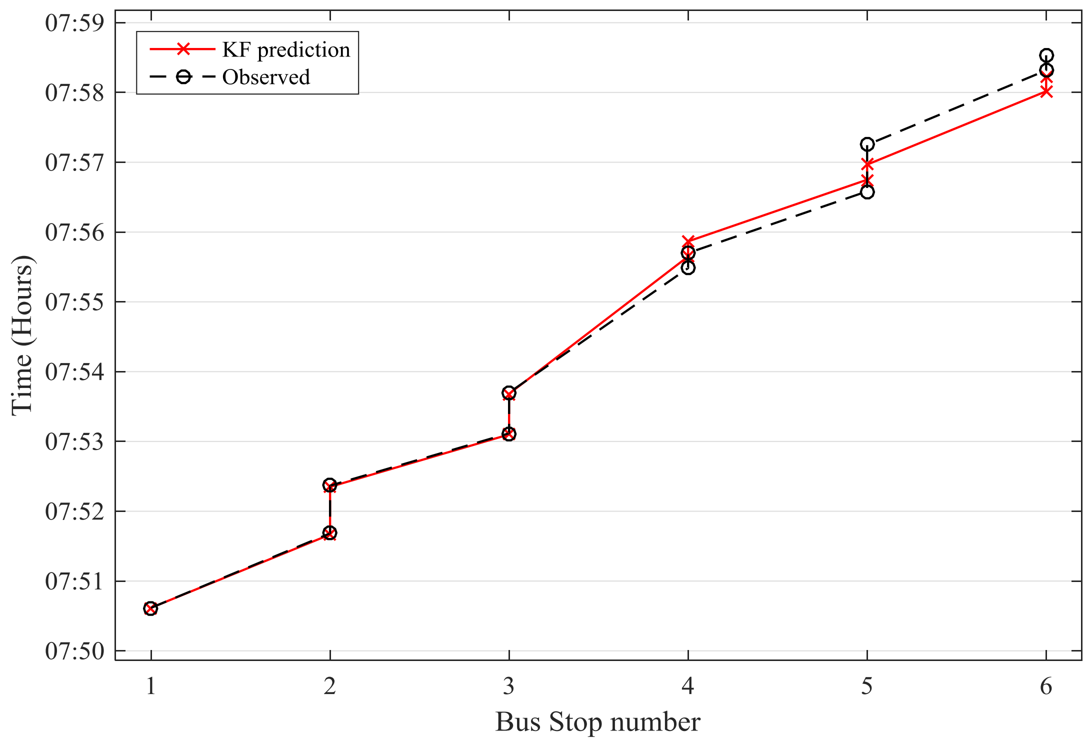

Figure 4.3 Bus B's travel time from stop 1 to downstream stop (observed vs. predicted)

4.3.3 Short-Distance Prediction

While the bus is running on the links, the AVL system sends every period of time t_{update} (assumed in the simulation to be 20 seconds). The model uses these data to predict the running time needed before arriving at the next bus stop. This adjusts the long-distance prediction.

To demonstrate the result of the long-distance prediction, the follow scenario is used. When the bus departs from bus stop 3, the model predicts the travel time to the downstream stop (as shown in Table 4.17). The bus, which is now running on link 3, is predicted to arrive at stop 4 at (7:55:33).

Table 4.17 Predicted travel time when the bus departs stop 3

Bus B at stop 3	**Bus B's predicted travel time to the downstream stop**

Arrival at stop 4	7:55:33
Departure from stop 4	7:55:52
Arrival at stop 5	7:56:43
Departure from stop 5	7:57:07
Arrival at stop 6	7:58:12
Departure from stop 6	7:58:30

Every 20 seconds, the AVL system sends an update with the bus's current information. Table 4.18 shows the first prediction while the bus is running.

Table 4.18 First prediction update while the bus is running on link 3

Bus B running on link 3	Bus B's predicted travel time to the downstream stop
Arrival at stop 4	7:55:33
Departure from stop 4	7:56:03
Arrival at stop 5	7:56:54
Departure from stop 5	7:57:18
Arrival at stop 6	7:58:23
Departure from stop 6	7:58:41

Table 4.19, Table 4.20, Table 4.21and Table 4.22 show the updated predictions for the remaining running time of link 3 before arriving at stop 4.

Table 4.19 Second prediction update while the bus is running on link 3

Bus B running on link 3	Bus B's predicted travel time to the downstream stop
Arrival at stop 4	7:55:30
Departure from stop 4	7:56:00
Arrival at stop 5	7:56:51
Departure from stop 5	7:57:15

Arrival at stop 6	7:58:20
Departure from stop 6	7:58:38

Table 4.20 Third prediction update while the bus is running on link 3

Bus B running on link 3	**Bus B's predicted travel time to the downstream stop**
Arrival at stop 4	7:55:27
Departure from stop4	7:55:57
Arrival at stop 5	7:56:48
Departure from stop 5	7:57:12
Arrival at stop 6	7:58:17
Departure from stop 6	7:58:35

Table 4.21 Fourth prediction update while the bus is running on link 3

Bus B running on link 3	**Bus B's predicted travel time to the downstream stop**
Arrival at stop 4	7:55:29
Departure from stop 4	7:55:59
Arrival at stop 5	7:56:50
Departure from stop 5	7:57:14
Arrival at stop 6	7:58:19
Departure from stop 6	7:58:37

Table 4.22 Fifth prediction update while the bus is running on link 3

Bus B running on link 3	**Bus B's predicted travel time to the downstream stop**
Arrival at stop 4	7:55:30
Departure from stop 4	7:56:00
Arrival at stop 5	7:56:51
Departure from stop 5	7:57:15
Arrival at stop 6	7:58:20
Departure from stop 6	7:58:38

Table 4.23 shows the observe arriving time of bus B to stop 4

Table 4.23 Bus B arrived to stop 4 "observed arriving"

Bus B arrived to stop 4	Bus B's predicted travel time to the downstream stop
Arrival at stop 4 "observed"	*7:55:29*
Departure from stop 4	7:55:52
Arrival at stop 5	7:56:43
Departure from stop 5	7:57:07
Arrival at stop 6	7:58:12
Departure from stop 6	7:58:30

Figure 4.4 shows the predicted running time versus the observed running time for a link comprised of twenty-five mini links. It is clear that the KF prediction changes according to changes in the observed data changes in order to predict a more accurate time.

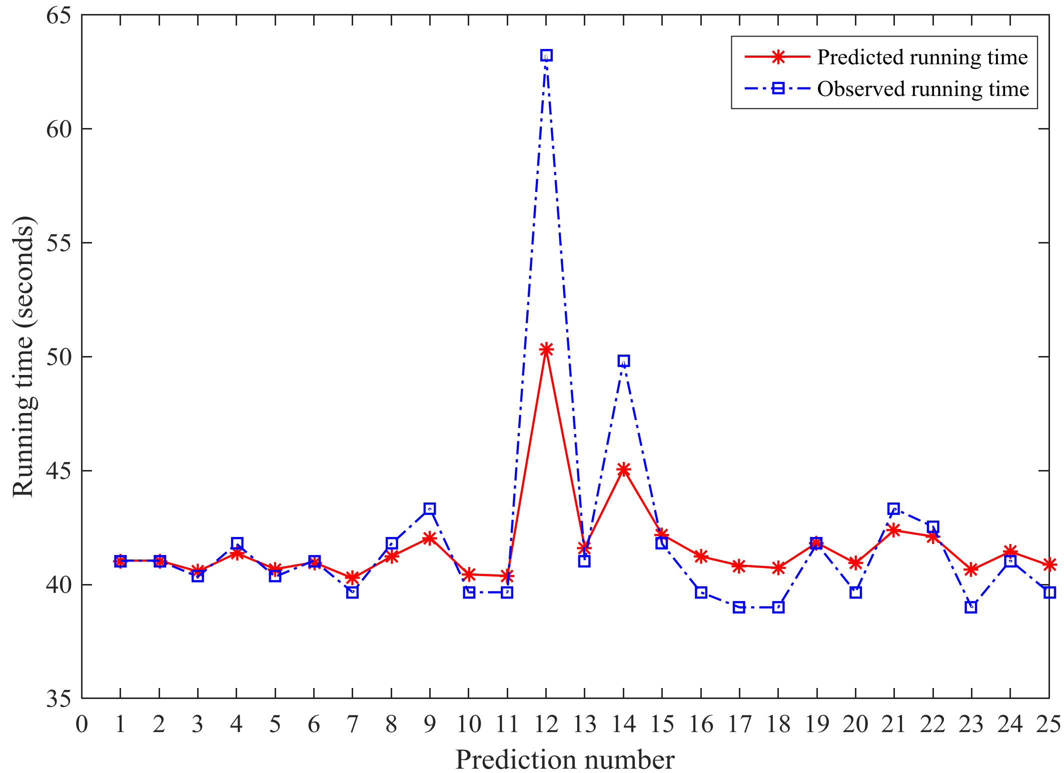

Figure 4.4 Predicted running time vs. observed running time

4.4 Model Performance Evaluation

The proposed model's results are evaluated through comparison to two other models: the model proposed in [49] denoted as Model A; and the model proposed in [84] denoted as Model B. The model proposed in this thesis is denoted as Model C.

- Model A vs. Model C

The following prediction error measurements are used to evaluate the travel time prediction's accuracy compared to observed travel time. The error indices are:

1. Mean relative error (RE_{mean})

$$RE_{mean} = \frac{1}{N}\sum_t \left|\left\{\frac{X_{true}(t) - X_{predicted}(t)}{X_{true}(t)}\right\}\right| \tag{4.1}$$

2. Root squared relative error (RE_{rs})

$$RE_{rs} = \sqrt{\frac{1}{\sum_t X_{true}(t)}\sum_t \left\{\frac{X_{true}(t) - X_{predicted}(t)}{X_{true}(t)}\right\}^2 X_{true}(t)} \tag{4.2}$$

3. Maximum relative error (RE_{max})

$$RE_{max} = max \left|\left\{\frac{X_{true}(t) - X_{predicted}(t)}{X_{true}(t)}\right\}\right| \tag{4.3}$$

Where:

N is the number of the sample;

$X_{true}(t)$ is the observed value at time t; and

$X_{predicted}(t)$ is the predicted value at time t.

Table 4.24 Relative error results of travel time prediction from model A and model C

Model	RE_{mean}	RE_{rs}	RE_{max}
Model A	0.028	0.036	0.077
Model C	0.022	0.030	0.24

Model C shows better error for all results except for RE_{max}; however, two links on model A show higher RE_{max} than model C.

- Model B vs. Model C

The dwell time prediction's accuracy is evaluated for model C and compared to model B. The following prediction error measurements are used:

1. Mean absolute error (MAE)

$$\boldsymbol{MAE} = \frac{1}{N}\sum_t \left|X_{true}(t) - X_{predicted}(t)\right| \tag{4.4}$$

2. Root mean squared error (RMSE)

$$\boldsymbol{RMSE} = \sqrt{\frac{1}{N}\sum_t \{X_{true}(t) - X_{predicted}(t)\}^2} \tag{4.5}$$

Table 4.25 Absolute error results of dwell time prediction for model B and model C

Model	MAE	RMSE	RE_{max}
Model B	3.96	5.98	20.4
Model C	3.60	5.61	76

Model C shows lower error than model B, with the exception of RE_{max}. Maximum relative error (RE_{max}) captures the maximum prediction error. For model C, wheelchair movement is considered when predicting the dwell time, and it is assumed the boarding or alighting time for the wheelchair is between 30 and 45 seconds. Therefore, model C always shows higher RE_{max}.

4.5 Summary

This chapter presents the simulation setting, assumptions, and simulation performance results. The results of the model's predicted bus travel time by using long-and short-distance prediction are presented here. Finally, the prediction error measurements are used to compare the model with two other models.

Chapter 5

5 Conclusion and Future Research

5.1 Concluding Remarks

This thesis has focused on the travel time prediction for a public transportation system by adopting a dynamic model that uses both real-time data and historical data while relying on Kalman filter. Time prediction is an essential platform for having a reliable ITS, which is a main infrastructure of smart cities.

The thesis has discussed the enabling technologies, AVL and APC systems, that are used to help predict a bus's travel time. Their advantages, weaknesses, and the way they work have been reviewed. It has been concluded that these technologies play a significant role in tracking a bus's location and number of boarding and alighting passengers. Therefore, the tsarist agencies should consider installing these technologies in their fleets for real-time monitoring of their network.

Next, this thesis conducted a literature review of the models and algorithms that are used to predict a bus's travel time, including statistical historical average model, regression models, artificial neural network (ANN) models, and Kalman filter model (KF). The thesis also reviewed other models that predict a bus's travel time. After that, the models that are used to predict a bus's dwell time at a bus stop were explored.

The proposed model is based on KF and it predicts the travel time by predicting the running time and the dwell time separately. The proposed model has two main parts: long-distance prediction and short-distance prediction. The long-distance prediction is made once every time the bus departs a bus stop. The model first predicts the running time for the coming link and then predicts the bus's dwell time at the bus stop that is located at the end of that link. Next, the model predicts the running time of the link that comes after the bus stop and so on up to the end of the downstream stop. The summation of the links' running time and bus stops' dwell time is the total travel time. The proposed model has two sources of data that are used for the prediction process: the previous bus's data as observed data and the average or mean of the historical data. The historical data is for the bus that has run the same link or waited at the same bus stop over the previous weeks for the same day and time period. On the other hand, the short-distance prediction works while the bus is running a link. Each time the AVL system sends the bus's information, the model predicts the remaining running time and the arrival time at the upcoming bus stops. Therefore, the link is divided into smaller links which are called mini links, and the long-distance prediction is adjusted accordingly. The data used here is comprised of the observed data of the last mini link and the average or mean of the past mini links. The short-distance prediction reacts to any incidents that may occur while the bus is running and the total travel time is updated accordingly.

A simulated environment was chosen and the settings and assumptions for performance analysis of travel time prediction were assigned. The simulated environment covered a route that has six bus stops. The simulated results showed that KF filter is an estimation tool that is able to update its predictions quickly. Using the average of the historical data instead of the last day's data lowers the risk of error, because abnormal data from the last day could affect the prediction.

The thesis introduced a model that predicts the travel model based on the most recent real-time data. The model reacts to incidents very quickly by updating its predictions at every bus stop departure and following every AVL update. The KF also uses the mean of the historical data instead of the last day's data. For the bus stop dwell time, the proposed model considers boarding passengers, alighting passengers, and passengers who use wheelchairs.

According to the simulation results, the overall conclusion is that the proposed model shows a lower risk of error than other models for travel time prediction and dwell time prediction.

5.2 Future Research

While the results are encouraging, a number of endeavours should be made in consideration of certain issues and to extend the work.

- The proposed model does not consider the bus's deceleration time that is needed to stop at a bus stop and the bus's acceleration time when it departs the bus station. Deceleration time and acceleration time could be considered in future research to improve the accuracy of the prediction.
- The proposed model was simulated for a single route and two buses, and it showed promising results. However, the model needs to be applied to an entire public

transportation network. Therefore, more strategies need to be addressed and developed in order to control an entire transit fleet.

- As an extension of this work, the bus's travel time prediction could be developed to control the charging schedule for electric buses (E-buses). To mitigate the concern of rising emission levels, E-buses have become a prime starting point to introducing electric vehicles (EV). An E-bus is a suitable solution for PT by using the features of both low energy consumption and long extending mileage [100][101][102][103]. Having an accurate prediction helps the system to determine when, where, and to what percentage the bus's battery needs to be charged. Knowing the travel time, the bus's consumption of energy and the charging stations that are planted along the bus's route facilitates the process of building the charging schedule and balancing the power load on the grid.

References

[1] H. Menouar, I. Guvenc, K. Akkaya, A. S. Uluagac, A. Kadri, and A. Tuncer, “UAV-enabled intelligent transportation systems for the smart city: Applications and challenges,” *IEEE Communications Magazine*, vol. 55, no. 3, pp. 22–28, 2017.

[2] M. Gohar, M. Muzammal, and A. Ur Rahman, “SMART TSS: Defining transportation system behavior using big data analytics in smart cities,” *Sustain. Cities Soc.*, vol. 41, pp. 114–119, 2018.

[3] Y. Liu, X. Weng, J. Wan, X. Yue, H. Song, and A. V. Vasilakos, “Exploring data validity in transportation systems for smart cities,” *IEEE Communications Magazine*, vol. 55, no. 5, pp. 26–33, 2017.

[4] S. H. Sutar, R. Koul, and R. Suryavanshi, “Integration of Smart Phone and IOT for development of smart public transportation system,” in *Proceedings of the International Conference on Internet of Things and Applications, IOTA*, 2016, pp. 73–78.

[5] R. Hernández, C. Cárdenas, and D. Muñoz, “Game theory applied to transportation systems in Smart Cities: analysis of evolutionary stable strategies in a generic car pooling system,” *Int. J. Interact. Des. Manuf.*, vol. 12, no. 1, pp. 179–185, 2018.

[6] M. Alam, J. Ferreira, and J. Fonseca, *Intelligent transportation systems: Dependable vehicular communications for improved road safety*, vol. 52. Springer, 2016.

[7] M. Alam, A. Rayes, X. He, M. Atiquzzaman, J. Lloret, and K. F. Tsang, “Guest Editorial Introduction to the Special Issue on Dependable Wireless Vehicular Communications for Intelligent Transportation Systems (ITS),” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 949–952, 2018.

[8] I. T. S. Canada, “Overview.” [Online]. Available: https://www.itscanada.ca/about/index/index.html. [Accessed: 09-Nov-2017].

[9] L. Zhuhadar, E. Thrasher, S. Marklin, and P. O. de Pablos, “The next wave of innovation—Review of smart cities intelligent operation systems,” *Comput. Human Behav.*, vol. 66, pp. 273–281, 2017.

[10] X. Xu, W. Wang, Y. Liu, X. Zhao, Z. Xu, and H. Zhou, “A bibliographic analysis and collaboration patterns of IEEE transactions on intelligent transportation systems between 2000 and 2015,” *Proc. IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2238–2247, 2016.

[11] J. M. Sussman, *ITS: A short history and a perspective on the future*. New York: Springer, 2005.

[12] S. Elkosantini and S. Darmoul, “Intelligent Public Transportation Systems: A review of architectures and enabling technologies,” in *proceedings of Interational Conference on Advanced*

Logistics and Transport (ICALT), 2013, pp. 233–238.

[13] S. Satunin and E. Babkin, “A multi-agent approach to Intelligent Transportation Systems modeling with combinatorial auctions,” *Expert Syst. Appl.*, vol. 41, no. 15, pp. 6622–6633, 2014.

[14] T. D. Camacho, M. Foth, and A. Rakotonirainy, “Pervasive Technology and Public Transport: Opportunities Beyond Telematics,” *IEEE Pervasive Comput.*, vol. 12, no. 1, pp. 18–25, 2013.

[15] T. Liu, A. Ceder, J. Ma, W. Guan, and L. Zhou, “Graphical Human-Machine Interactive Approach for Integrated Bus Transit Scheduling: Lessons Gained from a Large Bus Company,” *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 4, pp. 1023–1028, 2017.

[16] M. Daniel, *Handbook of behavioral and cognitive geography*. Cheltenham, Glos: Edward Elgar Publishing Limited, 2018.

[17] J. Kazak, J. van Hoof, and S. Szewranski, “Challenges in the wind turbines location process in Central Europe – The use of spatial decision support systems,” *Renew. Sustain. Energy Rev.*, vol. 76, pp. 425–433, 2017.

[18] V. Torretta, E. C. Rada, M. Schiavon, and P. Viotti, “Decision support systems for assessing risks involved in transporting hazardous materials: A review,” *Saf. Sci.*, vol. 92, pp. 1–9, 2017.

[19] C. Technology, D. Risk, and R. Division, *Intelligent Transportation Systems for Sustainable Development in Asia and the Pacific*. Bangkok, 2015.

[20] T. T. C. (TTC), “2016 Operating Statistics.” [Online]. Available: https://www.ttc.ca/About_the_TTC/Operating_Statistics/2016/index.jsp. [Accessed: 14-Nov-2017].

[21] Delcan and Lura Consulting, “City of Toronto Congestion management plan 2014-2018,” Toronto, 2013.

[22] R. Malekian, K. Wu, K. Steenhaut, R. H. Jacobsen, and M. A. Igartua, “Guest Editorial: Introduction to the Special Issue on Connected Vehicles in Intelligent Transportation Systems,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 7, pp. 2152–2155, 2018.

[23] C. Morton, B. Caulfield, and J. Anable, “Customer perceptions of quality of service in public transport: Evidence for bus transit in Scotland,” *Case Stud. Transp. Policy*, vol. 4, no. 3, pp. 199–207, 2016.

[24] H. Yu, R. Xiao, Y. Du, and Z. He, “A Bus-Arrival Time Prediction Model Based on Historical Traffic Patterns,” in *Proceedings of IEEE International Conference on Computer Sciences and Applications*, 2013.

[25] R. Zhang, W. Liu, Y. Jia, G. Jiang, J. Xing, H. Jiang, and J. Liu, “WiFi Sensing Based Real-time Bus Tracking and Arrival Time Prediction in Urban Environments,” *IEEE Sens. J.*, vol. 18, no. 11, pp. 4746–4760, 2018.

[26] E. C. Glen Weisbrod, EDRG; Derek Cutler and E. Duncan, “Economic Impact of Public Transportation Investment,” 2014.

[27] T. F. T. A. (FTA), “Transit’s Role in Environmental Sustainability,” 2016. [Online]. Available: https://www.transit.dot.gov/regulations-and-guidance/environmental-programs/transit-environmental-sustainability/transit-role. [Accessed: 04-Apr-2018].

[28] F. Li, Y. Yu, H. Lin, and W. Min, “Public bus arrival time prediction based on traffic information management system,” in *Proc. IEEE International Conference on Service Operations, Logistics, and Informatics (SOLI)*, 2011, pp. 336–341.

[29] P. Zhou, Y. Zheng, and M. Li, “How Long toWait?: Predicting Bus Arrival Time with Mobile Phone based Participatory Sensing,” *IEEE Trans. Mob. Comput.*, vol. 13, no. 6, pp. 1228–1241, 2014.

[30] J. Bates, J. Polak, P. Jones, and A. Cook, “The valuation of reliability for personal travel,” *Transp. Res. Part E Logist. Transp. Rev.*, vol. 37, no. 2–3, pp. 191–229, 2001.

[31] E. Mazloumi, G. Rose, G. Currie, and M. Sarvi, “An Integrated Framework to Predict Bus Travel Time and Its Variability Using Traffic Flow Data,” *J. Intell. Transp. Syst.*, vol. 15, no. 2, pp. 75–90, 2011.

[32] N. Prasad and S. P. Priyanka, “Performance of tracking public transport in heterogeneous networks,” in *Proceedings of the IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, 2015.

[33] F. Balbo and S. Pinson, “Using intelligent agents for Transportation Regulation Support System design,” *Transp. Res. Part C Emerg. Technol.*, vol. 18, no. 1, pp. 140–156, 2010.

[34] S. Olariu and M. C. Weigle, *Vehicular Networks: From Theory to Practice*. London, UK: Chapman & Hall/CRC, 2009.

[35] G. V. Sundar and B. G. Rajagopal, “IoT based passenger information system optimized for Indian metros,” in *Proceedings of the International Conference on Electronics, Communication and Aerospace Technology, ICECA*, 2017, pp. 92–96.

[36] G. Guido, A. Vitale, and D. Rogano, “Assessing public transport reliability of services connecting the major airport of a low density region by using AVL and GIS technologies,” in *Proceedings of the IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC)*, 2016.

[37] T. C. R. P. Multisystems, inc, United States. Federal Transit Administration and and T. D. Corporation., *Strategies for Improved Traveler Information.* Washington, D.C: Transportation Research Board., 2003.

[38] R. F. Casey and et al, *Advanced public transportation systems : the state of the art : update 200*.

Cambridge, MA: U.S. Dept. of Transportation, Federal Transit Administration, 2000.

[39] P. Mukheja, M. Kiran K, N. R. Velaga, and R. B. Sharmila, "Smartphone-based crowdsourcing for position estimation of public transport vehicles," *IET Intell. Transp. Syst.*, vol. 11, no. 9, pp. 588–595, 2017.

[40] C. H. Zhou and Z. G. Gao, "A real-time information system for BRT based on GPS/signpost compound navigation technology," in *Proceedings of the International Conference on Logistics Engineering and Intelligent Transportation Systems, (LEITS)*, 2010.

[41] M. Sandim, R. J. F. Rossetti, D. C. Moura, Z. Kokkinogenis, and T. R. P. M. Ŕubio, "Using GPS-based AVL data to calculate and predict traffic network performance metrics: A systematic review," in *Proceedings of the IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, 2016, pp. 1692–1699.

[42] J. Fihn and J. Finndahl, "A Framework for How to Make Use of an Automatic Passenger Counting System," 2011.

[43] "Telematics - Passenger Counting." [Online]. Available: http://www.actia.co.uk/passenger-counting/. [Accessed: 22-Jan-2018].

[44] R. Kovács, L. Nádai, and G. Horváth, "Concept validation of an automatic passenger counting system for trams," in *Proceedings of the 5th International Symposium on Applied Computational Intelligence and Informatics, SACI*, 2009.

[45] H. Yu, Z. He, and J. Liu, "A vision-based method to estimate passenger flow in bus," in *Proc. International Symposium on Intelligent Signal Processing and Communications Systems, ISPACS*, 2008.

[46] G. Chen, X. Yang, J. An, and D. Zhang, "Bus-Arrival-Time Prediction Models : Link-Based and Section-Based," *J. Transp. Eng.*, vol. 138, no. 1, pp. 60–66, 2012.

[47] R. Choudhary, A. Khamparia, and A. K. Gahier, "Real Time Prediction of Bus Arrival Time: A Review," in *Proceedings of The 2nd International IEEE Conference on Next Generation Computing Technologies*, 2016.

[48] W. Fan and Z. Gurmu, "Dynamic Travel Time Prediction Models for Buses Using Only GPS Data," *Int. J. Transp. Sci. Technol.*, vol. 4, no. 4, pp. 353–366, 2015.

[49] A. Shalaby and A. Farhan, "Bus Travel Time Prediction Model for Dynamic Operations Control and Passenger Information Systems," Washington, D.C, 2003.

[50] M. Chen, X. Liu, and J. Xia, "Dynamic Prediction Method with Schedule Recovery Impact for Bus Arrival Time," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 1923, pp. 208–217, 2005.

[51] A. Karbassi and M. Barth, "Vehicle route prediction and time of arrival estimation techniques for improved transportation system management," in *Proceedings of The Intelligent Vehicles*

Symposium. Proceedings, 2003.

[52] B. M. Williams and L. a. Hoel, "Modeling and Forecasting Vehicular Traffic Flow as a Seasonal ARIMA Process: Theoretical Basis and Empirical Results," *J. Transp. Eng.*, vol. 129, no. 6, pp. 664–672, 2003.

[53] B. A. Kumar, L. Vanajakshi, and S. C. Subramanian, "Bus travel time prediction using a time-space discretization approach," *Transp. Res. Part C Emerg. Technol.*, vol. 79, pp. 308–332, 2017.

[54] M. As and T. Mine, "Dynamic Bus Travel Time Prediction Using an ANN-based Model," in *Proceedings of the 12th International Conference on Ubiquitous Information Management and Communication*, 2018.

[55] R. Jeong and L. Rilett, "Prediction Model of Bus Arrival Time for Real-Time Applications," *Transp. Res. Rec. J. Transp. Res. Board*, no. 1927, pp. 195–204, 2005.

[56] L. Vanajakshi and L. R. Rilett, "Support Vector Machine Technique for the Short Term Prediction of Travel Time," in *in Proceedings of IEEE Intelligent Vehicles Symposium*, 2007.

[57] S. Cheng, B. Liu, and B. Zhai, "Bus arrival time prediction model based on {APC} data," in *Proc. 6th Advanced Forum onTransportation of China (AFTC)*, 2010.

[58] S. Maiti, A. Pal, A. Pal, T. Chattopadhyay, and A. Mukherjee, "Historical Data based Real Time Prediction of Vehicle Arrival Time," in *The proceedings of IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*, 2014.

[59] Z. Wall and D. J. Dailey, "Title : An Algorithm for Predicting the Arrrival Time of Mass Transit," in *in Proceedings of 78th Transp. Res. Board Annu. Meeting*, 1999.

[60] C. Bai, Z. R. Peng, Q. C. Lu, and J. Sun, "Dynamic bus travel time prediction models on road with multiple bus routes," *Comput. Intell. Neurosci.*, vol. 2015, no. 63, 2015.

[61] E. Mazloumi, S. Moridpour, G. Currie, and G. Rose, "Exploring the Value of Traffic Flow Data in Bus Travel Time Prediction," *J. Transp. Eng.*, vol. 138, no. 4, pp. 436–446, 2012.

[62] L. Frechette and A. Khan, "Bayesian Regression-Based Urban Traffic Models," *Transp. Res. Rec.*, vol. 1644, no. 1, pp. 157–165, 1998.

[63] A. Abdelfattah and A. Khan, "Models for Predicting Bus Delays," *Transp. Res. Rec.*, vol. 1623, no. 1, pp. 8–15, 1998.

[64] J. Patnaik, S. Chien, and A. Bladikas, "Estimation of Bus Arrival Times Using APC Data," *J. Public Transp.*, vol. 7, no. 1, pp. 1–20, 2004.

[65] S. I.-J. Chien, Y. Ding, and C. Wei, "Dynamic Bus Arrival Time Prediction with Artificial Neural Networks," *J. Transp. Eng.*, vol. 128, no. 5, pp. 429–438, 2002.

[66] S. I.-J. Chien, Y. Ding, and C. Wei, "Dynamic Bus Arrival Time Prediction with Artificial Neural Networks," *J. Transp. Eng.*, vol. 128, no. 5, pp. 429–438, 2002.

[67] Z. K. Gurmu and W. (David) Fan, “Artificial Neural Network Travel Time Prediction Model for Buses Using Only GPS Data,” *J. Public Transp.*, vol. 17, no. 2, pp. 45–65, 2014.

[68] R. E. Kalman, “A New Approach to Linear Filtering and Prediction Problems,” *J. basic Eng.*, vol. 82, no. 1, p. 35, 1960.

[69] J. Liu, W. Wang, X. Gong, X. Que, and H. Yang, “A hybrid model based on Kalman Filter and neutral network for traffic prediction,” in *Proceedings of the IEEE 2nd International Conference on Cloud Computing and Intelligent Systems (CCIS)*, 2012, pp. 533–536.

[70] M. S. Grewal, “Kalman Filtering,” in *International Encyclopedia of Statistical Science*, M. Lovric, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 705–708.

[71] E. M. REINHOUDT and S. A. VELASTIN, “A Dynamic Predicting Algorithm for Estimating Bus Arrival Time,” in *Presents of the 8Ih IFAC/IFIP/IFORS Symposium*, 1997, vol. 30, no. 8, pp. 1225–28.

[72] M. Chen, X. Liu, J. Xia, and S. I. Chien, “A dynamic bus-arrival time prediction model based on APC data,” *Comput. Civ. Infrastruct. Eng.*, vol. 19, no. 5, pp. 364–376, 2004.

[73] L. M. Kieu, A. Bhaskar, and E. Chung, “Benefits and issues for bus travel time estimation and prediction,” in *The proceedings of Australasian Transport Research Forum*, 2012, no. September, pp. 1–16.

[74] T. Zhu, F. Ma, T. Ma, and C. Li, “The prediction of bus arrival time using global positioning system data and dynamic traffic information,” in *In the Proceedings of Wireless and Mobile Networking Conference (WMNC), 4th Joint IFIP*, 2011.

[75] T. Zhu, J. Dong, J. Huang, S. Pang, and B. Du, “The bus arrival time service based on dynamic traffic information,” in *Proceedings of t6th International IEEE Conference on Application of Information and Communication Technologies (AICT)*, 2012.

[76] J. Dong, L. Zou, and Y. Zhang, “Mixed Model For Prediction OfBus Arrival Times,” in *The proceedings of theIEEE Congress on Evolutionary Computation*, 2013.

[77] B. Seyed, M. Hossein, A. Ismail, and A. Balali, “Evaluating Bus Dwell Time at Key Stops Using Automatic Data Collection Systems,” *Inst. Transp. Eng. ITE J.*, vol. 87, no. 11, pp. 45–50, 2017.

[78] S. Arhin, E. Noel, M. F. Anderson, L. Williams, A. Ribisso, and R. Stinson, “Optimization of transit total bus stop time models,” *J. Traffic Transp. Eng. (English Ed.*, vol. 3, no. 2, pp. 146–153, 2016.

[79] G. Chen, L. Huang, H. Liu, and X. Yang, “Effects of bunching at stops on bus dwell time,” *the Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems, ITSC*. 2013.

[80] R. Rajbhandari, S. Chien, and J. Daniel, “Estimation of Bus Dwell Times with Automatic

Passenger Counter Information," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 1841, pp. 120–127, 2003.

[81] Associates, Kittelson, and et al., *Transit capacity and quality of service manual*, 2nd ed. Washington, D.C.: Transportation Research Board of the National Academies, 2003.

[82] K. Dueker, T. Kimpel, J. Strathman, and S. Callas, "Determinants of Bus Dwell Time," *J. Public Transp.*, vol. 7, no. 1, pp. 21–40, 2004.

[83] A. Tirachini, "Bus dwell time: The effect of different fare collection systems, bus floor level and age of passengers," *Transp. A Transp. Sci.*, vol. 9, no. 1, pp. 28–49, 2013.

[84] C. Zhang and J. Teng, "Bus Dwell Time Estimation and Prediction: A Study Case in Shanghai-China," in *In the Proceedings of 13th COTA International Conference of Transportation Professionals (CICTP)*, 2013, vol. 96, pp. 1329–1340.

[85] S. Rashidi and P. Ranjitkar, "Estimation of bus dwell time using univariate time series models Soroush," *J. Adv. Transp.*, vol. 49, no. 1, pp. 139–152, 2014.

[86] J. Xin and S. Chen, "Bus Dwell Time Prediction Based on KNN," *Procedia Eng.*, vol. 137, pp. 283–288, 2016.

[87] C. Wang, Z. Ye, Y. Wang, Y. Xu, and W. Wang, "Modeling Bus Dwell Time and Time Lost Serving Stop in China," *J. Public Transp.*, vol. 19, no. 3, pp. 55–77, 2016.

[88] T. Jiang, N. D. Sidiropoulos, and G. B. Giannakis, "Kalman filtering for power estimation in mobile communications," *IEEE Trans. Wirel. Commun.*, vol. 2, no. 1, pp. 151–161, 2003.

[89] J. Barros, E. Pérez, R. I. Diego, M. De Apráiz, and E. Perez, "Application of Kalman Filtering in Power Systems : Harmonic Distortion," in *Kalman Filtering*, J. Gomez, Ed. New York, NY: Nova Science, 2011.

[90] S. Y. Chen, "Kalman filter for robot vision: A survey," *IEEE Trans. Ind. Electron.*, vol. 59, no. 11, pp. 4409–4420, 2012.

[91] O. M. Maklouf, Y. El Halwagy, M. Beumi, and S. D. Hassan, "Cascade Kalman filter application in GPSINS integrated navigation for car like robot," in *Proceedings of National Radio Science Conference*, 2009.

[92] Y. Geng and J. Wang, "Adaptive estimation of multiple fading factors in Kalman filter for navigation applications," *GPS Solut.*, vol. 12, no. 4, pp. 273–279, 2008.

[93] C. Wells, *The Kalman Filter in Finance*. Dordrecht: Springer Science & Business Media, 1996.

[94] "iLectureonline." [Online]. Available: http://www.ilectureonline.com/. [Accessed: 29-Mar-2018].

[95] Q. Li, R. Li, K. Ji, and W. Dai, "Kalman Filter and Its Application," in *Proceedings of the 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS)*, 2015.

[96] R. Faragher, "Understanding the basis of the kalman filter via a simple and intuitive derivation,"

IEEE Signal Process. Mag., vol. 29, no. 5, pp. 128–132, 2012.

[97] N. Thacker and A. Lacey, “Tutorial: The kalman filter,” in *Imaging Science and Biomedical Engineering Division*, Medical School, University of Manchester, 1998, pp. 133–140.

[98] S. Sepasi, R. Ghorbani, and B. Y. Liaw, “Improved extended Kalman filter for state of charge estimation of battery pack,” *J. Power Sources*, vol. 255, pp. 368–376, 2014.

[99] S. M. Bozic, *Digital and Kalman filtering: an introduction to discrete-time filtering and optimum linear estimation*, 2nd ed. London, UK: Edward Arnold, 1994.

[100] M. Rogge, E. van der Hurk, A. Larsen, and D. U. Sauer, “Electric bus fleet size and mix problem with optimization of charging infrastructure,” *Appl. Energy*, vol. 211, pp. 282–295, 2018.

[101] J. Wu, H. He, J. Peng, Y. Li, and Z. Li, “Continuous reinforcement learning of energy management with deep Q network for a power split hybrid electric bus,” *Appl. Energy*, vol. 222, pp. 799–811, 2018.

[102] C. Yang, W. Lou, J. Yao, and S. Xie, “On Charging Scheduling Optimization for a Wirelessly Charged Electric Bus System,” *Proc. IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 6, pp. 1814–1826, 2018.

[103] M. T. Sebastiani, R. Luders, and K. V. O. Fonseca, “Evaluating Electric Bus Operation for a Real-World BRT Public Transportation Using Simulation Optimization,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 10, pp. 2777–2786, 2016.

Appendix A

A. Confidence Interval

The confidence interval (CI) is used to provide a random interval to allow for the probability of estimated parameter in order to satisfy the requirement. Many methods can be used to calculate CI. Since the variance is known, the quantitative method to calculate CI is used here. A group of runs' results were collected, then the CI was applied to determine whether or not they had passed. The confidence level is 95% and the reliability factor $Z_{a/2}$ is 1.96.

To calculate CI, the mean or average of all runs (R_1, R_2,…, R_i) must be calculated through the following formula:

$$\overline{\boldsymbol{R}} = \frac{\sum_{i=1}^{n} \boldsymbol{R}_{n}}{\boldsymbol{n}} \tag{A 1}$$

The following formula is used to calculate the population variance σ^2

$$\sigma^2 = \frac{\sum_{i=1}^{n} (\boldsymbol{R}_i - \overline{\boldsymbol{R}})^2}{\boldsymbol{n}} \tag{A 2}$$

The standard deviation is calculated as follows:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(R_i - \bar{R})^2}{n}} \tag{A 3}$$

Equations A 4 and A5 show how the upper and lower bounds of confidence interval can be calculated:

$$\textbf{Confidence upper bounds} = \bar{R} + Z_{a/2} * \frac{\sigma}{\sqrt{n}} \tag{A 4}$$

$$\textbf{Confidence upper bounds} = \bar{R} - Z_{a/2} * \frac{\sigma}{\sqrt{n}} \tag{A 5}$$

The following tables A.1, A.2, A.3, and A.4 show the CI for the travel time prediction and the dwell time prediction.

Table A. 1 Relative error results of travel time prediction

Number of runs	RE_{mean}	RE_{rs}	RE_{max}
1000	0.023344	0.032253	0.15678
2500	0.022319	0.030194	0.251029
5000	0.023154	0.009509	0.190476
7500	0.023075	0.031528	0.289528
10000	0.022982	0.03139	0.31134
15000	0.022869	0.031149	0.22547
20000	0.022739	0.030732	0.322581
25000	0.022935	0.031319	0.28
35000	0.023068	0.031581	0.332627
45000	0.022865	0.03137	0.3147

Table A. 2 Confidence interval calculation for relative error

Variable	RE_{mean}	RE_{rs}	RE_{max}
$\bar{R}$	0.0229	0.0291	0.2675
n	10	10	10
CI lower bound	0.0228	0.0248	0.2303
CI upper bound	0.0231	0.0334	0.3046

Table A. 3 Absolute error results of dwell time prediction

Number of runs	MAE	RMSE	RE_{max}
1000	3.061	5.599911	42
2500	3.5872	5.390139	37
5000	2.9996	5.443896	71
7500	3.939467	6.128948	70
10000	3.5576	5.469863	75
15000	3.104933	5.629588	110
20000	3.5642	5.415339	76
25000	3.5936	5.550279	76
35000	3.594194	5.600362	83
45000	3.559403	5.53839	75

Table A. 4 Confidence interval calculation for absolute error

Variable	MAE	RMSE	RE_{max}
$\bar{R}$	3.4561	5.5767	71.5
n	10	10	10
CI lower bound	3.2703	5.4459	58.8609
CI upper bound	3.6420	5.7075	84.1391