DYNAMIC TRANSIT SIGNAL PRIORITY

by

WAEL EKEILA

B.Sc., The American University of Sharjah, 2002

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

in

THE FACULTY OF GRADUATE STUDIES

(CIVIL ENGINEERING)

THE UNIVERSITY OF BRITISH COLUMBIA

March 2006

Abstract

Transit Signal Priority (TSP) is a popular Traffic Demand Management strategy used to enhance the performance of transit systems by modifying the signal control logic to give transit signal priority through signalized intersections. Conventional TSP strategies used in most cities have been shown to offer significant benefits in minimizing the delays of transit vehicles. However, there have been several concerns about the shortcomings of conventional TSP strategies which limited their applications. The main concern is the potential negative impact on the cross street traffic. Another concern is the static nature of conventional TSP strategies and the lack of responsiveness to real-time traffic and transit conditions.

This thesis describes the development and evaluation of a dynamic Transit Signal Priority (TSP) control system which has the ability to provide signal priority in response to real-time traffic and transit conditions. The dynamic TSP system consists of three main components: a virtual detection system, a dynamic arrival prediction model, and a dynamic TSP algorithm. The methodology followed to develop the system consisted of three main steps. The first was to develop a microsimulation model that would be used to test and evaluate the performance of the dynamic TSP system. In the model, Automatic Vehicle location (AVL) was used as the virtual detection system. The second step was the development of several bus arrival prediction models using linear regression and neuro-fuzzy methods. Techniques such as Kalman and Bayes filters were used to refine the prediction. The last step was the development of a dynamic TSP algorithm that would decide what TSP strategy to use and when to apply it. The dynamic TSP system was tested and compared to the conventional TSP system using the microsimulation model. Scenarios with varying simulation parameters and traffic volumes were tested. Results showed that when an accurate prediction model was used, the dynamic TSP system outperformed the conventional one. The dynamic TSP system could be further enhanced if better arrival prediction models are used, more TSP strategies are evaluated, and a larger scale of implementation is studied.

Table of Contents

List of Tables

List of Figures

Acknowledgement

At last, the day has come and I am submitting my thesis and graduating. In the past couple of years I had my good and bad days. With my friends and colleagues around me these days passed smoothly and happily.

At first I would like to thank the City of Vancouver for the opportunity of working with them on the Vancouver's Streetcar project. I would specially like to thank Scott Edwards and Dale Bracewell for their help and support during all the time I spent in the city.

I would also like to thank my supervisor Tarek Sayed who was like a big brother to me. His continuous support and advice helped me not only with my studies but even with my personal life.

In my years of studies, I have met a lot of people that are now some of my best friends. I would never forget the days and nights we spent together studying, working, and having some fun. Thanks Daria, Kevan, Shafat, Wok, and Paul for all these nice moments.

In my stay in Canada, I have made very close friends who I would like to thank for all their support and help. I would like to thank Kareem, Amr, Mohammed Ammar, Najeeb, Khalid, Mohammed El Faramawy, Nagham, Abdelsalam, Tarek, Yehya, Ibraheem and Aly. I would like to thank Kareem a lot for all the time and effort he made and for all the times we spent together. However, the real appreciation and thanks are to the ones who maybe were far away from me yet closest with their feelings and hearts. Special thanks to my parents for all their support, help, and prayers throughout my stay in Canada and my whole life.

CHAPTER 1 INTRODUCTION

1.1 General Overview

The need for transportation facilities is always increasing. In every city, the transport system is analogous to the veins and arterials in the body facilitating the movement of people and freight. The Transport system affects the economy, safety, security, social life, and environment of each city. As a result, traffic and transportation engineers are making significant efforts to optimize the use of existing traffic networks.

Most big and developed cities in the world suffer from congestion. Congestion has significant effects on the society, companies, individuals, and environment. Congestion has several negative impacts such as delay in traffic movement, waste of resources, loss in productivity, and increase in vehicles' gas emissions and fuel consumption (Lomax & Turner, 2001).

Due to the limited space and funds needed for further increase in the capacity of road networks, deployment of Travel Demand Management strategies became a necessity rather than an option. Travel Demand Management (TDM), also called Mobility Management, helps solve congestion problems by optimizing the use of existing road networks. TDM helps people and communities meet their transport needs in the most efficient way, which often reduces total vehicle trips. TDM gives higher priority to travel based on the value and cost of each trip. TDM strategies are categorized into four major groups: policy reformation strategies, tolling strategies, parking and land management strategies, and strategies favoring transit and non-motorized modes (Victoria Transportation Institute, 2005). Transit systems play a very important role in the whole transport system. Many new and advanced traffic demand management strategies, such as Transit Signal Priority, tend to favor and promote the use of transit systems since they value person trips over vehicle trips and are friendlier to the environment.

With significant advances in telecommunications and computing technologies, Intelligent Transportation Systems (ITS) revolutionized the usage of TDM strategies in terms of the technologies used and the significant benefits gained. ITS uses advanced detection and monitoring systems to facilitate and optimize the application of many TDM strategies. As a result, Transit Signal Priority techniques have gained popularity in both research fields and real-life applications. Transit Signal Priority is a system that attempts to improve transit effectiveness by giving priority to transit vehicles at signalized intersections. This research aims to develop a dynamic transit signal priority system that would outperform the conventional transit signal priority system currently used in many cities.

1.2 Problem Statement

Transit Signal Priority has been widely used in different cities in North America, Europe, and Asia. Transit Signal Priority has been very useful in facilitating the movement of transit vehicles through intersections. Conventional TSP strategies used in most cities have been shown to offer significant benefits in minimizing the delays of transit vehicles. However, there have been several concerns about the shortcomings of conventional TSP strategies which limited their applications. The main concern is the potential negative impact on the cross street traffic. Another concern is the static nature of conventional TSP strategies and the lack or responsiveness to real-time traffic and transit conditions.

Conventional TSP systems are mainly composed of fixed check-in and check-out detectors that are located close to the intersection; usually the check-in detector is located 50 to 100 meters away from the stop bar. The detectors are integrated with a simple algorithm that adjusts the signal timings and phases. Once transit vehicles are detected, conventional TSP systems would make a decision on whether to provide signal priority and then apply a TSP strategy. However, a critical disadvantage to the TSP operation may occur without real-time transit travel time prediction. Such uncertainty in transit travel times may consequently render the signal priority control ineffective (Lee et al., 2005). As well, due to the location of the check-in detectors, the time available to take the decision and apply a TSP solution is short. If more time is available an optimal TSP strategy could be selected and applied. TSP strategies could be applied over several cycles to reduce delays of transit and cross street vehicles and to minimize the effects on the signal coordination system.

1.3 Thesis Objectives

This research aims to develop a "Dynamic" or "Adaptive Transit Signal Priority System". The dynamic transit signal priority system is mainly composed of three components which are: an Automatic Vehicle Location detection system, a transit arrival prediction model, and a TSP priority strategy selection algorithm. The main objectives of this research are as follows:

- To use AVL as a virtual detection system rather than the check-in and check-out detectors used in the conventional transit signal priority system.
- To develop several transit arrival prediction models using different variables and techniques.
- To enhance the developed models using various filters.
- To study, analyze, and evaluate the accuracy of the developed prediction models.
- To develop an algorithm that would use the virtual detection system and the arrival prediction model to improve the operation of TSP.
- To test the developed algorithm using different prediction models and filters at different exposure rates.

- Finally, to perform a comparison between conventional and dynamic TSP systems.

1.4 Thesis Scope

The thesis is mainly composed of three components that are integrated together in an algorithm. The first component is the development of arrival prediction models using different techniques and filters. The data used was obtained from the Vancouver's Streetcar Project where a microsimulation model was built for the area surrounding the streetcar path in Vancouver. Using this data, several arrival prediction models were developed using linear regression analysis and neuro-fuzzy models. Kalman Filter and Empirical Bayes techniques were applied to the models to refine the prediction and produce lower variances. The second component is the development of a priority strategy selection system that uses the arrival prediction model to determine the arrival time and the signal phase at the time of arrival. Based on these, a priority strategy would be selected and the time to apply it would be determined. Finally, the third component is testing the developed algorithm in a simulation setting with a modeled virtual detection system using four different algorithms and under two different exposure rates.

1.5 Thesis Structure

The thesis includes eight chapters, a reference section, and appendices. Chapter 1 includes the introduction, problem statement, thesis's objectives, and thesis's scope. Chapter 2 includes a literature review of transit signal priority, arrival prediction models, and previous work on arrival prediction models and TSP applications. Chapter 3 discusses the VISSIM microsimulation software, its characteristics, and the modeling of the simulated intersection that is used to test the developed algorithms. Chapter 4 introduces the developed arrival prediction models and their analysis. Chapter 5 describes the developed algorithms and their components. Chapter 6 presents the simulation results for the different algorithms and exposure rates. Chapter 7 analyzes, compares, and discusses the results. And finally, the conclusions and recommendations are given in chapter 8.

CHAPTER 2 LITERATURE REVIEW

2.1 Transit Signal Priority

Transit Signal Priority can be defined as an operational method that facilitates the movement of transit vehicles, such as buses and streetcars, through signalized intersections by adjusting the signal control logic. Frequently, priority and vehicle preemption are mistakenly assumed to be similar. Although both methods facilitate the movement of vehicles, preemption interrupts the signal operation while priority changes or modifies the signal operation. Furthermore, preemption is usually used for high priority situations where emergency vehicles are used and the main concern is to facilitate a safe movement of the vehicles through the signal with no consideration to the resulting delays. In the following subsections, TSP concepts, design criteria, and strategies are discussed.

2.1.1 Signal Control Types

To understand the different concepts used in transit signal priority design, the types of signal controllers used should be described. There are two types of signal controllers used in signal design: fixed-time signals and real-time signals (Roess et. al., 1998).

- Fixed-time signals, also called pre-timed signals, are designed based on the average traffic conditions where the signal timing plan is applied regardless of the actual traffic conditions. Historical records of traffic volumes are used to design the phases and timing plan of the signals.

- Real-time signals are designed based on continuous update of information. Several detection methods have been developed and used due to the significant advancements in telecommunication technologies. Real-time signals are considered the most desirable and flexible signals since the signal timing plans are responsive to the actual traffic conditions.

2.1.2 Priority Concepts

At a signal, priority can be awarded to a transit vehicle in different methods. How, when, and where to apply transit signal priority is what differentiates between priority concepts. Based on the method and data used by the TSP algorithm to give transit priority, active and passive priorities are defined. Moreover, direct and indirect priorities are dependent on where to apply signal priority. And finally, conditional and unconditional priorities are defined based on when to award signal priority. The priority concepts as described by (Chada & Newland, 2002) are as follows:

- *Active & Passive Priorities*: Active priority awards priority at a signal using a detection system that detects transit vehicles as they approach the intersection. This concept is popular since the TSP algorithm uses real traffic conditions. On the other hand, passive priority gives priority at a signal using historic data of transit vehicles arrivals without using a detection system. Passive priority is rarely used. It would be used at signalized intersections that are so close to transit vehicles origins where they will most probably be running on schedule. Since no detection system would be required, they also might be used when the available budget is small.

- *Direct & Indirect Priorities*: Direct priority awards priority to a transit vehicle at the local scale or as it reaches close to the signalized intersection. Priority solutions are provided locally to the transit vehicle by modifying the signal's logic. This concept is also popular and is used often with active priority. However, indirect priority gives priority to transit vehicles at a network scale or when they are a few intersections away. It adjusts the timings of all the signals ahead of the transit vehicles. At highly congested time periods, it attempts to reduce high traffic volumes that are ahead of transit vehicles. Indirect priority requires a sophisticated detection system for both transit and non-transit vehicles.
- *Conditional & Unconditional Priorities*: Conditional priority gives transit vehicles priority under certain limits and conditions. Limits include the transit vehicle's occupancy, time headway, and delay. The main objective of this concept is to operate the network at the most efficient way where priority would not greatly affect non-transit vehicles. On the other hand, unconditional priority provides priority to a transit vehicle whenever it is detected with no consideration to any limits. The main aim of this concept is to reduce travel times of transit vehicles. This

makes the transit system operate at a better level of service which encourages more people to use it.

2.1.3 Priority Design Criteria

TSP design criteria are based on the following parameters: the transit vehicle's delay, time headway, occupancy, location in queue, and the time since the last priority was provided (Chada & Newland, 2002).

- The schedule based criterion gives priority according to the transit schedules and the delay times of the transit vehicles. The main objective of this criterion is to keep the transit system on schedule.
- The headway based criterion gives priority based on the time headways between consecutive buses. The main aim of this criterion is to reduce bunching between buses and therefore to reduce the waiting time of the transit users.
- The occupancy based criterion gives priority based on the occupancy rates of the transit vehicles. The main aim of this criterion is to reduce the delay per person, thus giving higher priority to transit vehicles with high occupancies.
- The queue based criterion gives priority based on the queue length in front of the transit vehicles. The main objective of this criterion is to reduce the stopping and delay times of transit vehicles at signalized intersections.

- The priority time frequency based criterion gives priority based on the time since the last priority was provided at a certain intersection. This criterion aims to reduce the effect of TSP on cross street traffic.

2.1.4 Priority Strategies

Several transit signal priority strategies have been developed through various case studies and projects done on transit signal priority at different cities in North America. However, the type and characteristics of such strategies are dependent on the selected priority concept (ITS America, 2002).

2.1.4.1 Passive Priority Strategies

Passive priority strategies are based on the schedules of the transit vehicles. They are developed based on the assumption that transit vehicles adhere to the planned schedule. They are easy to implement and require low investment since no detection system is used. Passive priority strategies include:

- *Green Adjustment*: There are two types of green adjustment strategies: extending the green phase; and truncating the red phase. Depending on the arrival time of a transit vehicle, the signal timing is modified so that a transit vehicle receives a green signal when it arrives at the intersection.
- *Phase Splitting*: In this strategy, the signal phase is split into two equal phases as shown in Figure 2-1. This strategy would not affect the cycle

length nor the cross streets traffic. Although the start up time loss for the phase is doubled, the waiting time of a transit vehicle over a signal is halved. The phase should be long enough so that when divided the new split phases will have satisfactory clearance and green times. This strategy is suitable at intersections with medium to high transit vehicle volumes and where the non-transit vehicle volumes are medium to low.

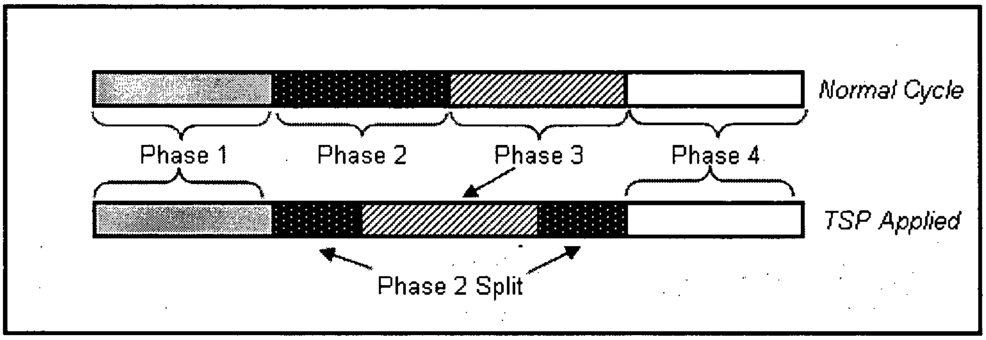

Figure 2-1 The Phase Splitting Strategy

- *Cycle Length Reduction*: This strategy is similar to the phase splitting strategy. However, in this strategy the cycle length is reduced as shown in Figure 2-2. This reduces the stopping time of the transit vehicles; however a higher portion of the cycle now is wasted as loss time. The loss time is due to the all red clearance times and the start up delays at the beginning of each phase. This strategy is suitable at intersections with medium to high transit vehicle volumes and with medium to low volumes of non-transit vehicles on all approaches.

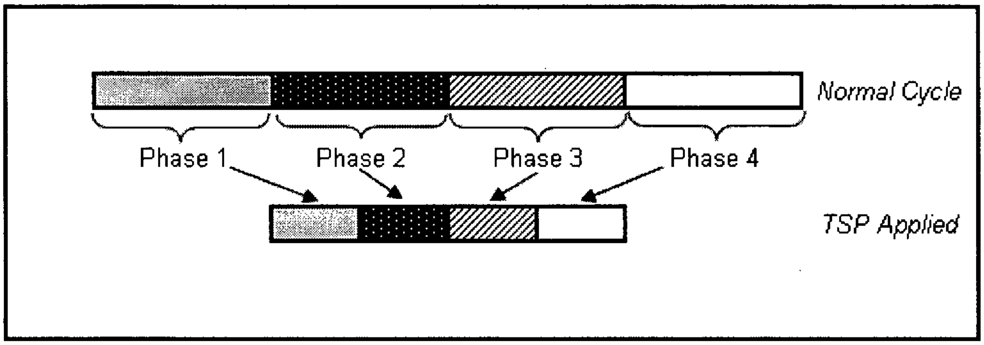

Figure 2-2 The Cycle Length Reduction Strategy

- *Transit Coordination*: Based on the schedule of the transit vehicles, the offsets of the signals along the route of the transit vehicles are designed for signal coordination. The transit vehicle path might not be through an arterial. In this case coordination is designed along the **path** of the transit vehicles. However this strategy is not that effective due to the difficulty of predicting the stoppage time of the transit vehicles at the different stations along the route.
- *Queue Jumps*: This strategy is only suitable at intersections where transit vehicles have designated lanes. Based on the transit vehicles schedule the transit vehicles are given early green times to be able to jump in front of the vehicle queues. Figure 2-3 illustrates the strategy.

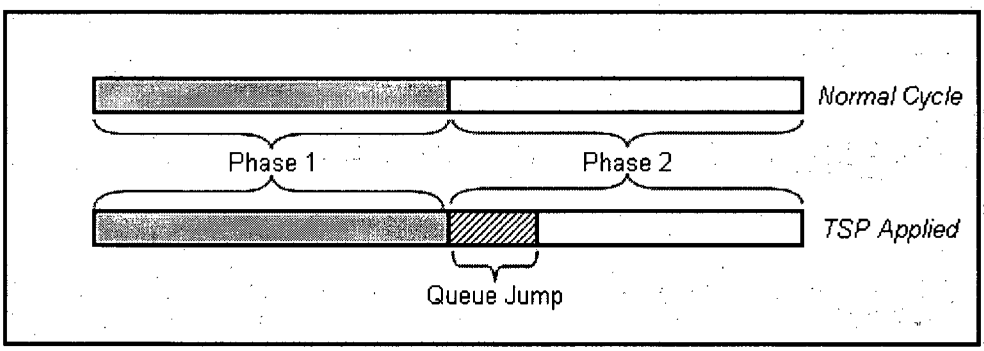

Figure 2-3 The Queue Jump Strategy

2.1.4.2 Active Priority Strategies

Active priority strategies are based on real time conditions and are better than the passive strategies since they are responsive to traffic conditions. However they require greater investment due to the implementation of a detection system. Active priority strategies include:

- *Green Extension*: This strategy is used when a transit vehicle is detected during its green phase. The green phase is extended, if needed, until the transit vehicle passes the intersection. A maximum extension limit is used to limit the impact on cross street vehicles. This priority strategy is one of the most popular strategies used in two-phase signals, since it allows the transit vehicle to pass the intersection without any extra losses in start-up and clearance. Figure 2-4 illustrates the strategy clearly.

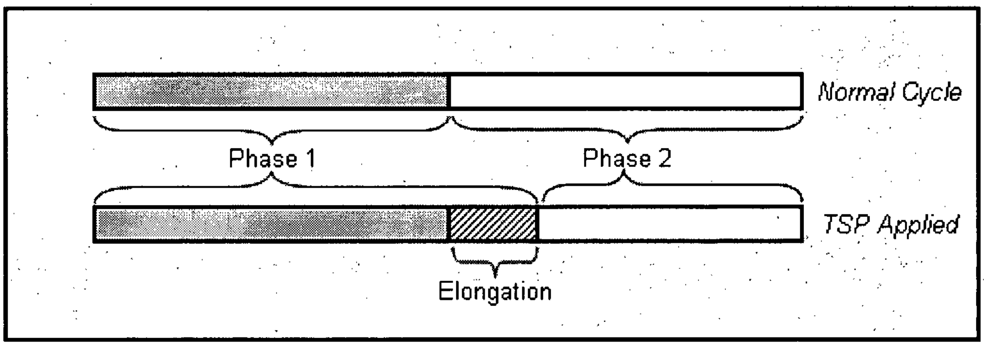

Figure 2-4 The Green Extension Strategy

- *Red Truncation (Early Green)*: This strategy is used when a transit vehicle is detected during its red phase. The cross street green signal is truncated, if the green phase duration is greater than the minimum green. Then an inter-phase is started to shift the transit signal phase into green. This priority strategy is also one of the most popular strategies used in two-phase signals. Figure 2-5 illustrates the strategy.

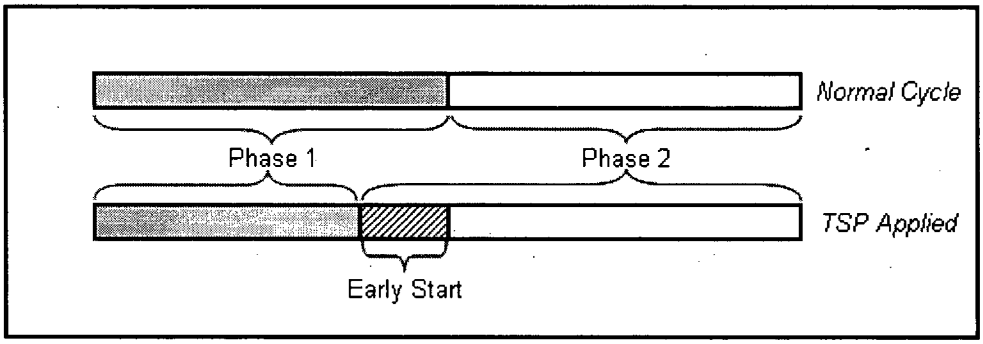

Figure 2-5 The Red Truncation Strategy

- *Phase Insertion*: In this strategy, a new phase is added to the cycle to give priority to the transit vehicle. A good example is when a transit vehicle needs to turn left on a two phase signal. A protected left turn phase is added to the signal cycle. Figure 2-6 illustrates the strategy clearly.

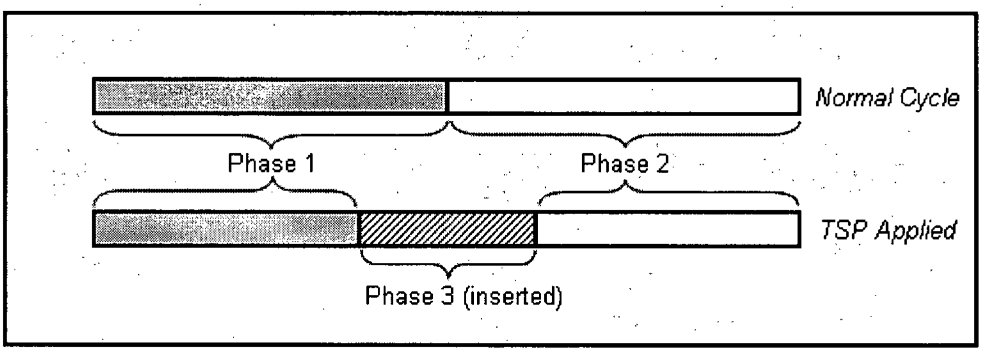

Figure 2-6 The Phase Insertion Strategy

- *Phase Rotation or Substitution*: This strategy is very effective in multiphase signals. The signal controller rotates or substitutes some signal phases in the cycle. Changing the sequence of the phases would not affect other vehicles yet it gives the desired transit vehicles the priority they require. However, this strategy might affect the coordination of the signals. A good example is of a signal with protected left turns. The left turn phases can be substituted with their thru movements to provide the transit signal priority needed. Figure 2-7 illustrates the strategy.

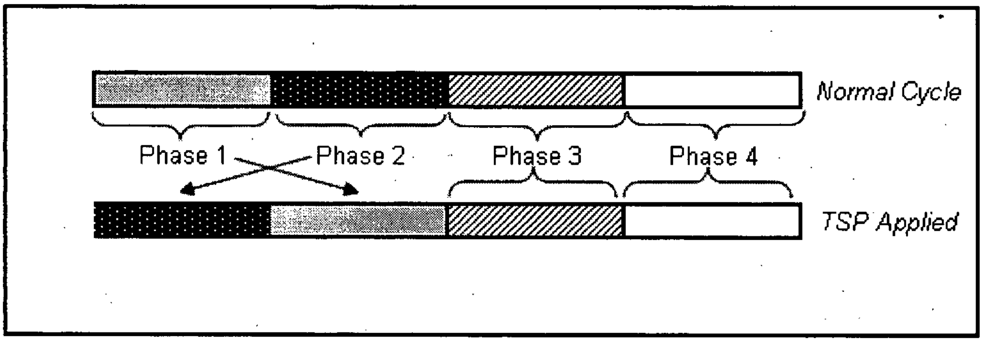

Figure 2-7 The Phase Rotation or Substitution Strategy

- *Queue Jumps*: This strategy is similar to that used in the passive priority systems; however the early green time is given to the detected transit vehicles.

2.1.5 Signal recovery

Applying transit priority strategies might have some negative effects on the cross street traffic and signal coordination. This leads into the discussion of signal recovery which consists of two concepts; signal compensation and offset recovery.

In certain TSP strategies, the additional green time given to the desired phase is taken from other phases if the signal cycle time is to be kept the same. This might cause some delays and queues on the other phases especially if transit signal priority is awarded frequently. Signal compensation is the solution for such a problem. Signal compensation is implemented by mainly adjusting either the cycle length where the TSP strategy is applied, or by adjusting the green times of the affected phases in the following cycles, or by doing both together. The way it is implemented depends on the frequency of priority requests.

In addition, when a transit signal priority strategy is applied, the offsets of the signals are altered. This alteration would most probably disrupt the coordination of the signals on an arterial. To solve this problem, offset recovery should be employed. Offset recovery tries to recover the offset of the signal by adjusting the cycle lengths of the next two or three cycles.

Signal recovery is usually applied on a network scale. To apply signal recovery with its two categories, signal controllers should try to optimize the usage of the signals based on the bus schedules, the different coordination schemes, and the real-time information such as queue discharge rates and transit arrival times.

2.1.6 Local, Arterial, & Network TSP

Transit signal priority can be applied at different scales. It can be applied at an isolated intersection, along a specific arterial with signalized intersections, and over a network of intersections.

Transit signal priority at an isolated intersection requires a basic detection system. Whenever a transit vehicle arrives at an intersection and is detected, a request is sent to the signal controller system. The signal controller in turn applies a strategy that is suitable for the current running phase and the required priority movement. In conventional TSP systems two sets of detectors are usually used. The first is a check-in detector that detects the approaching transit vehicle. The second is a check-out detector that detects the clearance of the transit vehicle from the intersection. The area between the check-in and the check-out detectors is called the detection zone or area.

Moving to a higher scale, transit signal priority can be applied over a group of signalized intersections along an arterial. The same detection system used at the local scale can be utilized. When a transit vehicle approaches the first intersection, the first check in-detector can send requests to all the signal controllers downstream. If long queues exist at the downstream intersections, then the signal controller tries to dissipate them to ease the implementation of transit signal priority when transit vehicles arrive.

At the highest and most sophisticated scale, transit signal priority can be applied at a network scale. At this scale, in addition to the non-transit vehicle queue detectors, a more complex and advanced detection system should be implemented such as the Automatic Vehicle Location systems. The AVL system monitors the movement of the transit vehicles at all times and sends all the data to the signal control centre which integrates this transit data with the non-transit flow data and the planned transit vehicles schedule to provide transit signal priority when possible and beneficial. At this level, various priority strategies and different priority concepts, such as conditional and indirect priorities, can be applied together in a way that optimizes the network performance. The signal control centre attempts to facilitate the movement of transit vehicles through the network while minimizing effects on non-transit vehicles. Applying transit signal priority at a network level is the most efficient method compared to the local and arterial levels. However, it needs considerable investment due to the sophisticated detection system needed. Moreover, developing such a control centre needs a lot of work and research, since a highly sophisticated program should be developed. This program would contain the algorithms needed by the signal control centre to modify and adjust the signal design of all the signalized intersections in the network in accordance with the continuous information update.

2.2 Arrival Prediction Models

Transit vehicles operations are dependent on traffic and ridership conditions. Due to the importance of their arrival times to transit users as well as engineers and planners, transit arrival prediction models have attracted interests of many researchers and engineers. Several methods have been applied in order to predict and enhance the prediction models. Methods include historical averaging, linear regression, neural networks, and Kalman filters. In the following subsections, some of the previously listed methods are described and summarized.

2.2.1 Linear Regression Models

Linear regression is defined as a method of estimating the conditional expected value of one variable given the values of some other variables. The variable of interest is conventionally called the *dependent variable*. The other variables are called the *independent variables*. The dependent and independent variables may be scalars or vectors. If the independent variable is a vector, one speaks of *multiple linear regression*.

Most linear regression models use the least-square method to fit the regression line. In the least-squares method, the best-fitting line for the observed data is calculated by minimizing the sum of the squares of the residuals from each data point to the line (if a point lies on the fitted line exactly, then its vertical deviation is 0). Because the deviations are first squared, then summed, there are no cancellations between positive and negative values.

The significance of the used variables and the fitness of the model can be checked using different statistics such as the R^2, T-value, and F-value. The R^2, coefficient of determination, is the proportion of a sample variance of a response variable that is explained by the predictor variables when a linear regression is done. The higher the R^2 the better is the relationship between the independent variables and the dependent variable. To check whether each of the independent variables is significant or not, the T-test is usually used. The T-value for each variable is calculated by dividing the coefficient of the variable by its standard error. For a specific confidence interval and degrees of freedom, if the T-value of the variable is greater than T-critical, then the variable is significant. Finally, the F-test is used to determine whether the observed relationship between the dependent and independent variables occurs by chance. For a specific confidence interval and degrees of freedom, if the F-value of the model is much greater than F-critical, then it is extremely unlikely that an F-value this high occurred by chance.

2.2.2 Artificial Neural Networks & NEU-Fuzzy Models

Artificial Neural Networks are a subfield of the artificial intelligence technology that has been used more frequently due to the advancements in computing and processing. ANNs have gained considerable popularity in solving complex problems and models where the nonlinear relation between the independent and dependent variables is nonlinear.

ANNs are mainly composed of a large number of neurons that are distributed into layers. Each neuron in a layer is connected to all other neurons in the preceding and succeeding layers. Each link has a weight that is a numerical estimate of the relation strength. A neuron receives inputs from all neurons in the preceding layer and sends its outputs to all neurons in the succeeding layer (Sayed & Razavi, 2000).

There are different structures or architectures for ANNs. The choice of the architecture mainly depends on the type of problem and on the available data. Architectures include the Standard Back Propagation Network (SBP), the Kohonen Unsupervised Network, the Radial Basis function Network (RBF), and the NEU-fuzzy Network (NEU-Frame Manual, 1997).

Although the use of ANNs in modeling complex problems is powerful and flexible, the relationship between the input and output variables are kept inside the model and are difficult to interpret by the modeler; ANNs have been always criticized as "black boxes". Moreover, ANNs will use all input variables in its weighting process regardless of their significance. Thus it is the role of the modeler to eliminate all insignificant input variables in order to produce a more efficient model (Sayed & Razavi, 2000).

Fuzzy logic, which is another subfield in the artificial intelligence technology, overcomes these shortcomings of ANNs and has gained popularity in modeling complex systems. Fuzzy logic is an extension of Boolean logic dealing with the concept of partial truth. Whereas classical logic holds that everything can be expressed in binary terms (0 or 1, black or white, yes or no), fuzzy logic replaces Boolean truth values with degrees of truth. The main characteristic of fuzzy logic is that relationships between variables can be linguistically interpreted using logical operators such as "AND" and "OR".

NEU-Fuzzy models emerged as a combination of the transparent and linguistic representation of the fuzzy systems with the learning ability of the ANNs. NEU-Fuzzy models have gained considerable popularity since they produce a set of linguistic rules that can be analyzed, enable knowledge extracted from the data to be combined with expert knowledge, produce a solution based on minimum error, discard unnecessary input variables, and can be constrained by fixed rules (Maier et al., 2000).

2.2.3 Kalman Filter

Kalman filter is an optimal linear recursive data processing algorithm. Using initial estimates, Kalman filter predicts and adjusts the parameters of the model with each new measurement. Kalman filter is optimal with respect to virtually any creation that makes sense. One aspect of its optimality is that it incorporates all information given to it. The word recursive in the description means that unlike certain data processing concepts, Kalman filter does not require all previous data to be kept in storage and reprocessed every time a new measurement is taken (Maybech, 1979).

Kalman filter processes all available measurements, regardless of their precision, to estimate the current value of the parameters with use of knowledge of the system and measurement device dynamics, the statistical description of the system noises, measurement errors, and uncertainty in the dynamics models, and any available information about initial conditions of the parameters. All of these characteristics made it very popular in many research fields and applications, particularly in the areas of control systems, assisted navigations, and communication technologies. Kalman filter performs optimally on a problem when the system can be described through a linear model and in which the system and measurement noises are infinite and Gaussian. Under these three conditions, Kalman filter is considered one of the best filters to be used. Even when some of these restrictions are relaxed, Kalman filter outperforms many other filters (Maybech, 1979).

2.2.4 Bayesian Statistics

In Bayesian statistics, parameters are considered random variables having probability distributions. These probabilities measure "degree of belief". The rules of probability (Bayes' theorem) are used to revise one's belief, given the observed data. Bayesian analysis can be performed on different statistical problems such as a binomial proportion, a normal mean, the difference between normal means, the difference between proportions, and a simple linear regression model. (John, 1989)

The Bayesian approach is in contrast to the concept of frequency probability. In frequency probability, probability is held to be derived from observed or imagined frequency distributions or proportions of populations. The difference has many implications for the methods by which statistics is practiced when following one model or the other, and also for the way in which conclusions are expressed. When comparing two hypotheses and using some information, frequency methods would typically result in the rejection or non-rejection of the original hypothesis with a particular degree of confidence, while Bayesian methods would suggest that one hypothesis was more probable than the other or that the expected loss associated with one was less than the expected loss of the other. (Thomas et. Al., 1992)

Bayes' theorem is often used to update a given variable in light of new evidence. Bayesian statistics uses both prior and new or existing information. Usually something is known about possible parameter values before the experiment is performed, and it is wasteful not to use this prior information. Due to this advantage of the Bayes' theorem, it is used by many researchers and practitioners to develop prediction models that make use of all available information such as historic, current, and experts' opinions.

2.3 Early Research & Work on Transit Arrival Prediction Models

Researchers and planners have given considerable attention to the development of transit arrival prediction models for decades since they have many essential uses in traffic engineering and urban planning. Transit arrival prediction models are used to give transit planners and users real time information that help in planning and adjusting their schedules. Moreover, arrival prediction models can be utilized in the application of dynamic and advanced transit control operations and techniques such as dynamic transit signal priority. The following subsections will review the work and the research undertaken in developing arrival prediction models, updating them with real-time data, and using them in TSP applications.

2.3.1 Static Arrival Prediction Models

Farhan A. et al. (2002) developed three models, a historical average model, a linear regression model, and an artificial neural network model based on data collected in May 2001 of Route 5 in Toronto's downtown area. The historical average model was developed using the average travel times between points along the route. The regression model was developed based on variables which included distance, average bus speed, dwell time at bus stops, and delay at intersections. Finally, the artificial neural network model was developed using the same variables of the regression model. The ANN model used the Time Lag Recurrent Network structure since it had shown to give the best results regarding accuracy and errors and since the bus trip along the different stops on its route was considered a series of sequential information. The study showed that the artificial neural network model had the lowest errors followed by the regression model and the historical average model.

Z. Wall & D.J. Dailey (1999) developed a model which included an algorithm of two sequential steps. The first step was tracking and estimating the location of the transit vehicle using the real-time AVL data. The second step was predicting the arrival time of the transit vehicle. The prediction model was developed using historical data and conditional probability of travel time knowing the distance left for the transit vehicle to get to the desired intersection. Their results showed that the predicted arrival times were useful for transit planners and riders since the arrival times could be predicted for 15 minutes to one hour in advance with errors ranging between 2 to 5 minutes.

2.3.2 Dynamic Arrival Prediction Models

In recent studies, the arrival prediction models were improved by the use of real-time data updates incorporated through different types of filters. Shalaby A. & Farhan A. (2004) extended their work and improved the ANN prediction model by using real-time AVL and APC updates. The developed model was composed of two algorithms based on the Kalman filter method. The first used AVL data to predict the transit vehicles' running time, while the second algorithm used the APC data to predict the transit vehicles' dwelling time at a transit stop. The study was based on the assumption that the dwelling time at a stop is highly affected by the arrival time of the transit vehicle at the stop and thus should be separated from the running time. Summing the output of both algorithms will give a prediction of the bus arrival time. The running prediction model was developed based on both historical travel time data collected through the AVL system and real-time travel time data update. The advantage of this model was that it captured any changes in traffic conditions. The dwelling time model was developed in a similar way where passenger arrival rates were used instead of running times.

In a similar study, Chen M. et al (2004) developed a dynamic arrival prediction model using real-time APC update. The developed arrival prediction model was composed of two components. The first component was the artificial neural network model. The ANN model was preferred in this study since it could describe complex systems without the need to formulate a mathematical equation to relate the different variables to the travel time. Four different variables were used in the model: day of the week, time of the day, weather, and segment. The second component, which made the prediction model dynamic, is the Kalman Filter algorithm. As a bus passes a specific stop, the real arrival time was obtained and used within the Kalman Filter algorithm to predict the arrival times at the downstream stops.

As part of the California PATH Program, Zhou et. al. (2004) developed an arrival prediction model that used real-time bus location and bus wheel speed information with historical AVL data. The arrival time predictor consisted of two models: a historical model that predicts the arrival time based solely on the historical data, and an adaptive model that adjusts its filter gain based on the real-time AVL data. The estimates generated by these two models were combined in a weighted average. The weight distribution was calibrated according to error variances obtained from the historical and adaptive models. The historical model relied solely on the historical data, assuming that bus operation speed between two nodes can be modeled as a constant with an uncertainty. The least-square regression method was used off-line to calibrate the constant average speed and the variance of an error term, with which the arrival time and its associated variance could be calculated. In addition, a historical dwell time model was developed to consider the dwell time distribution at each bus stop. For each of the bus stops, the mean and variance of the observed dwell times were calculated and these two parameters were updated as more AVL data were made available. Finer tunings were obtained using real-time speed and location data in the adaptive model. For this, a recursive least-square regression method was developed with its filter gain adjusted based on the real-time AVL data.

2.3.3 Arrival Prediction Models & TSP Applications

As an outcome of the California PATH Program, Liu H. et al. (2003) attempted to integrate between adaptive signal control techniques with advanced transit signal priority techniques. The priority awarded to a transit vehicle would be dependent on a weighting factor given to each transit priority call. A bus with priority would be converted into a relevant number of passenger vehicles using the weighting factor. The weighting factor was determined by the traffic demand and queuing conditions of every approach at an intersection and on the lateness of the transit vehicle. The computed weighting factor would be used to recalculate the signal timings and splits keeping into consideration the signal's parameters such as minimum and maximum greens, permissive start and end times, and force-off green times.

The use of arrival prediction models in TSP operation strategies is relatively a new idea. Lee J. et al. (2005) tried to use on-line microsimulation-based arrival prediction models in developing dynamic transit signal priority systems. The method consisted of two major components: an on-line microsimulation travel time prediction model and a priority operation model. The prediction model was designed as follows: when a transit vehicle is detected, it activates the prediction model to retrieve the currently implemented signal timing information and the traffic data from the upstream and downstream sensors. The developed prediction model would get the traffic counts and the detection times of each vehicle from the sensors in the intersection approach. The simulation model was designed to represent each vehicle as a separate object so that the model would be able to describe each vehicle's movements and the interactions between vehicles. It would be affected by several factors such as driving behavior, adjacent vehicle behavior, and other environmental and surrounding factors. The developed prediction model was validated using

PARAMICS through comparing the predicted bus travel times to the actual travel times. Later on, the priority operation model was used. The priority operation model consisted of a library of six priority plans that would be evaluated by the arrival prediction model. After evaluating the travel time of each priority plan, the most appropriate plan with the least travel time would be chosen and sent to the signal controller.

In summary, there has been interesting work done in the development of arrival prediction models. Also, the use of real-time data to update and refine the developed models has gained considerable popularity among researchers. Finally, although the use of arrival prediction models in TSP applications is relatively new and has several challenges such as the arrival prediction models' variance and accuracy, it has potential in the advancement of dynamic TSP operations.

CHAPTER 3 THE MICROSIMULATION MODEL

Microsimulation models are computer models that operate at the level of the individual behavioral entity. Such models simulate large representative populations of these low-level entities in order to draw conclusions that apply to higher levels of aggregation. This type of model is distinct from aggregate models whose explanatory variables already represent collective properties.

In traffic microsimulation models, the movements of individual vehicles traveling around road networks are determined by using car following, lane changing, and gap acceptance theories. They have been accepted by the transportation community for the evaluation and development of road traffic management and control systems.

Macrosimulation models provide an aggregated representation of traffic, typically expressed in terms of total flows per hour. In such models, all vehicles of a particular group follow the same rules of behavior.

By contrast, micro-simulation models provide a better representation of actual driver behavior and network performance. They are the only modeling tools available with the capability to examine certain complex traffic problems such as intelligent transportation systems, complex junctions, shockwaves, and effects of incidents. In addition, there is the appeal to users of the powerful graphics offered by most software packages that show individual vehicles traveling across networks that include a variety of road categories and junction types. This visual representation of problem and solution in a simple format can be a powerful way to gain more understanding of complex strategies and systems.

3.1 Simulation Software

3.1.1 VISSIM

VISSIM is a microscopic simulation program and one of the most powerful tools available for simulating multi-modal traffic flows, including cars, trucks, buses, heavy rail, trams, LRT, bikes and pedestrians. It provides the flexibility to model any type of geometric configuration or unique driver behavior encountered within the transportation system (VISSIM 4.0 Manual, 2004).

VISSIM has been used to analyze networks of sizes ranging from individual intersections to entire metropolitan areas. Within these transportation networks, VISSIM is able to model all roadway classifications from freeways to driveways. VISSIM has a variety of network entities which include transit vehicles, bicycles and pedestrians.

In VISSIM, data can be reported for any time period or an interval within that time period. Data can be reported for any point-location in the network, for an intersection, along any path and/or for the entire network. Data can be aggregated by mode or by vehicle class. It can also be reported for an individual vehicle. Numerous measures of effectiveness can be reported from VISSIM including delay, speed, density, travel time, stops and queues.

3.1.2 VAP

VAP (Vehicle Actuated Programming) is an optional add-on module of VISSIM for the simulation of programmable, phase or stage based, traffic actuated signal controls. The control logic is described in a text file using a C++ interface programming language. During VISSIM simulations or in the test mode, VAP interprets the control logic commands and creates the signal control commands for the VISSIM network. At the same time, various detector variables reflecting the current traffic situation are retrieved from the simulation and processed in the logic (VISSIM 4.0 Manual, 2004).

3.2 The Simulated Intersection

3.2.1 Layout

In order to test and compare the developed algorithms, an intersection was modeled as shown in Figure 3.1. Each of the four approaches consists of two lanes each with a width of 3.5 meters.

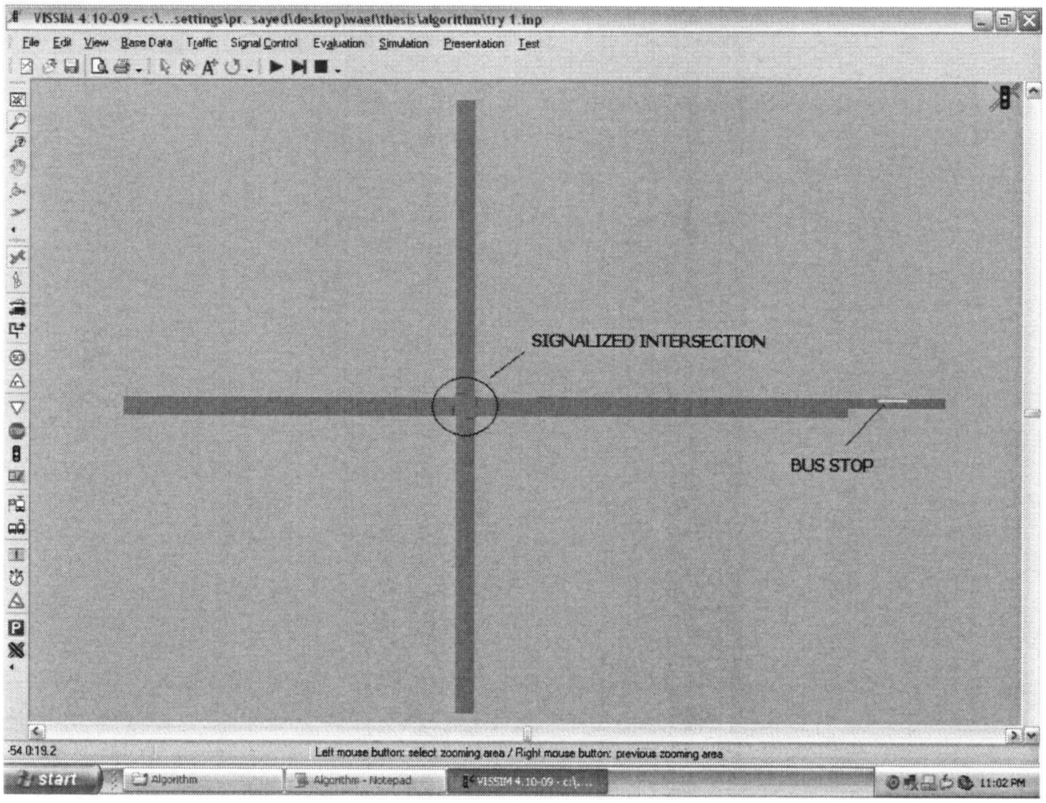

Figure 3-1 The Simulated Intersection's Layout

3.2.2 Signal Control

A two-phase signal was used to model the signal logic. The signal had a cycle length of 65 seconds with green time duration of 29 seconds for the first phase and of 26 seconds for the second phase. The amber time for each phase was taken as 4 seconds, and the all-red time was taken as 1 second. Figure 3.2 illustrates the signal cycle in detail.

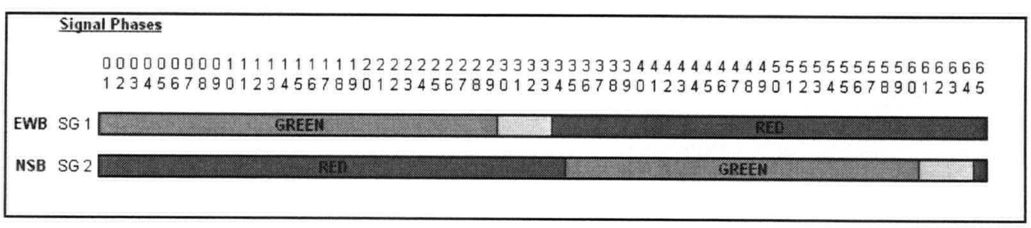

Figure 3-2 The Simulated Intersection's Signal Phases

In order to program the signal controller logic on VAP, an interstage description file was created. In the interstage file, the following information was defined:

- Signal groups. A signal group is defined as a group of signal heads that show exactly the same color at all times.
- Signal stages. A signal stage is composed of one or more signals groups that all show green at the same time.
- Starting stage
- Signal interstages. An interstage is defined as a transition stage where a signal moves from one stage to another, which usually includes the amber and all red times.

3.2.3 Right of Way

To designate the right of way for conflicting movements, priority rules were developed. A priority rule consists of one or more conflict markers (green bars) that are associated with a stop line (red bar). The stoppage of vehicles at the stop line depends on two conditions; the minimum gap time and the minimum distance headway. In order for a vehicle not to stop at a stop line or to pass it after stopping at it, the conditions of all its corresponding conflict points should be fulfilled.

The gap time condition is considered more important at free flow conditions or when traffic is moving at a relatively high speed. The gap time is calculated every time step in the simulation based on the time an approaching vehicle will require to reach the conflict point, assuming that the vehicle will continue traveling at its current speed. If the current gap time is less than the defined minimum gap time for that conflict point, the corresponding stop line will stop any vehicle approaching it.

On the other hand, the minimum distance headway condition is more relevant at congested conditions where traffic moves slowly and queues. The minimum distance headway can be defined as the length of the conflict area. During the simulation the headway is determined based on the distance between the conflict point and the first approaching vehicle. If the current distance headway is less than the defined minimum distance headway for that conflict point, the corresponding stop line will stop any vehicle approaching it.

When modeling the intersection, three sets of priority rules were defined for each of the following cases:

- Left turning vehicles at the intersection since the signal controller is assumed to have a permitted left turning phase rather than a protected one.
- Right turning vehicles at the intersection since vehicles were allowed to take a right turn on a red signal (Right Turn on Red).
- Clearance zone where no vehicle was allowed to enter the intersection if a vehicle or a group of vehicles still occupy the intersection from a previous phase.

3.2.4 Driver Characteristics

The driver and vehicle characteristics were chosen to be the same as those of the Vancouver's Streetcar Project. The passenger vehicles were modeled with the following characteristics:

- Maximum acceleration = 3.5 m/s^2
- Maximum deceleration = 7.5 m/s^2
- Traveling Speed = 48 to 58 km/hr
- Length = 4.5 m
- Width = 1.5 m

3.2.5 Transit System

In the model, one transit line was modeled with a transit stop located 350 meters upstream of the west bound approach. The transit vehicles were modeled with the following characteristics:

- Maximum acceleration = 3.5 m/s^2
- Maximum deceleration = 7.5 m/s^2
- Traveling Speed = 25 to 30 km/hr
- Length = 11.54 m
- Width = 2.5 m

Based on the Vancouver's streetcar Project, the dwelling time was defined as a normal distribution with a mean of 15 seconds and a standard deviation of

3 seconds. Dwelling time is defined as the time a transit vehicle waits at a transit stop, which includes the time boarding and deboarding passengers.

3.2.6 Detection (AVL System)

One of the dynamic components of this research is the use of AVL as a detection system rather than that of the conventional system, which is composed of fixed check-in and check-out detectors that are located on the road. The use of the AVL system as a detection system in dynamic TSP has several advantages such as:

- AVL systems already exist in many transit vehicles. There is nearly no investment required for the implementation of a new detection system.
- AVL systems give continuous detection regarding the location of the bus, unlike the fixed detectors, where the bus is detected at a certain location of the street. AVL can be thought of as a virtual detector.
- AVL systems can also send other data that may help in the application of dynamic TSP. Data may include passenger occupancy, dwelling time at stations, average speed between any two points, and bus adherence to schedule.
- AVL systems have a variety of detection criteria. They can provide time-step detection (where a transit vehicle can be detected every specific time interval), distance-step detection (where a transit vehicle can be detected as it travels a specific distance), and event-based

detection (where a transit vehicle can be detected at specific locations such as intersections and bus stops).

- AVL systems can be used to implement dynamic TSP on an arterial scale or even a network scale. All the transit vehicles movements over the network are known due to the continuous detection of the AVL system. This is one of its great advantages since it facilitates the work of the dynamic TSP algorithm in optimizing the movement of transit vehicles throughout the network.

In the simulated intersection, the only way to model the AVL detection system was as a distance-step detection system since VAP does not have the option of continuous detection of a certain vehicle. As shown in Figure 3.3, the AVL system was modeled as a group of detectors that are located 10 meters away from each other. The length of the detectors was taken as zero so that the detectors would act as a point detector. As the front wheels of the transit vehicle passes over a detectors a single impulse would be sent to the signal controller.

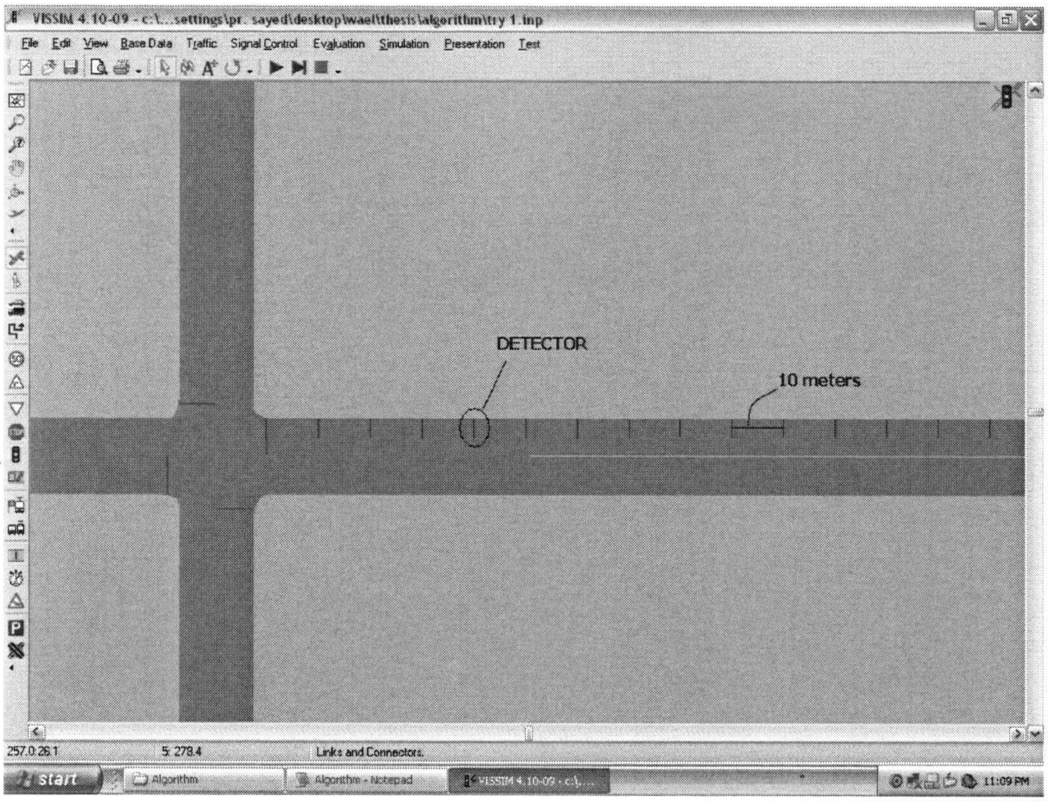

Figure 3-3 Modeling of AVL Detection

CHAPTER 4 DEVELOPMENT OF ARRIVAL PREDICTION MODELS

This chapter describes the development of arrival prediction models which are used to predict the arrival times of transit vehicles at a signalized intersection. This chapter describes the data, the development methods and procedures, and the developed models.

4.1 Data

Two sets of data were used in the development of the arrival prediction models. The first data set was obtained from a microsimulated network and was used in developing a highly accurate arrival prediction model that would be used to validate the dynamic algorithms. The second data set consisted of real-time data and was used in developing real-time arrival prediction models.

4.1.1 Microsimulated Data

A highly accurate arrival prediction model was needed to validate the performance of the developed algorithms. Therefore, the data used for the model development was obtained from the simulated intersection, which was the apparatus for testing the TSP systems. The prediction would be highly accurate since the transit vehicles in the simulation would all run through the same environment and traffic conditions. The validation process is discussed in detail in chapter 6.

The data contained distance, dwelling time, and travel time. The data was obtained from three simulation runs each with a different exposure level (750, 1000, 1200 vehicles/hr). Travel time measuring points were located every 100 meters for 800 meters as shown in Figure 4.1. In each microsimulation run, travel time measurements were collected for four different busses. In total, 96 data points were available for the development of the model.

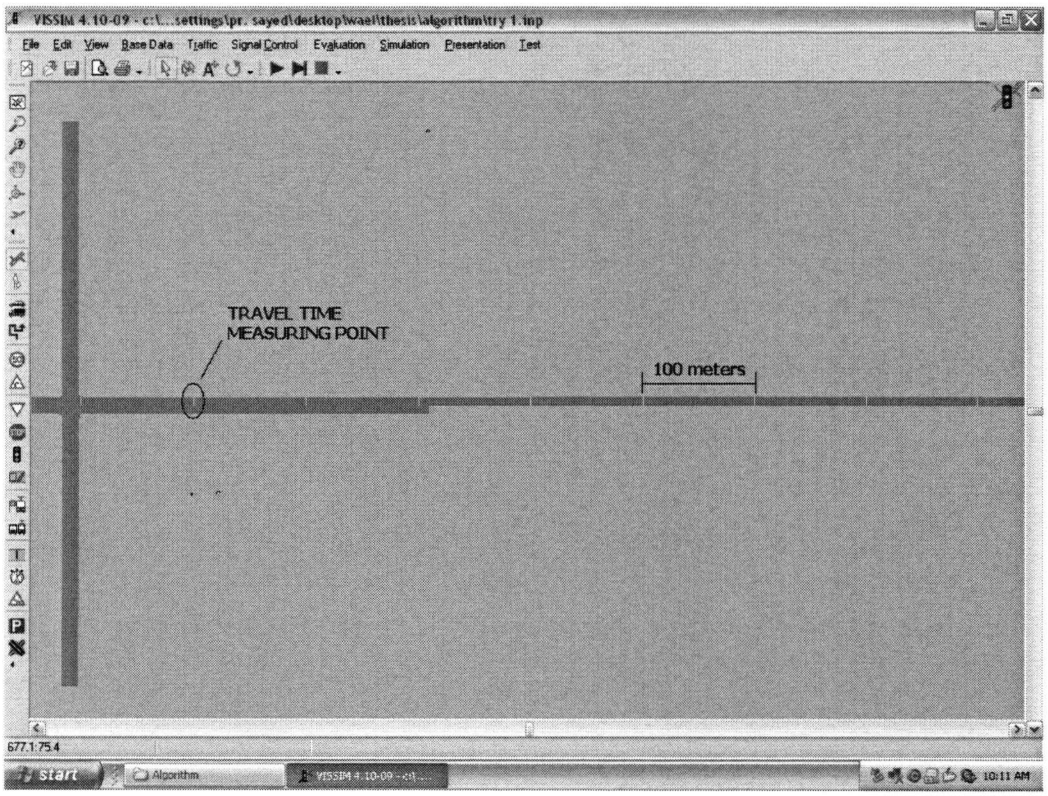

Figure 4-1 Travel Time Measurement Points

4.1.2 Real-Time Data

The real-time arrival prediction models are models that can be used for networks that have similar characteristics. The real-time models were tested to check whether their use in Dynamic TSP systems would be successful. Several real-time arrival prediction models would be compared and tested for their performance.

The data used for the development of the real-time arrival prediction models was obtained from the Vancouver's Streetcar Project. The Streetcar Project studied eight transit lines that served the eastern part of Downtown Vancouver and the False Creek area. For this thesis, the data was obtained from line # 8 that travels between Marine Drive and Downtown Vancouver through Main Street. Figure 4.2 shows the part of the line that has been chosen for data collection. It consists of fourteen segments; seven of them northbound, and the other seven southbound.

This portion of roadway was chosen since the segment characteristics cover several aspects. Although all of them have three thru lanes, they vary in characteristics such as:

- Segment's length
- Traffic volumes
- Signal type, some are actuated while the others are pre-timed
- Cycle length and green times

- Number of bus stops, some have none, while others have one or two bus stops.
- Location of bus stops (near-side, mid-block, or far-side)

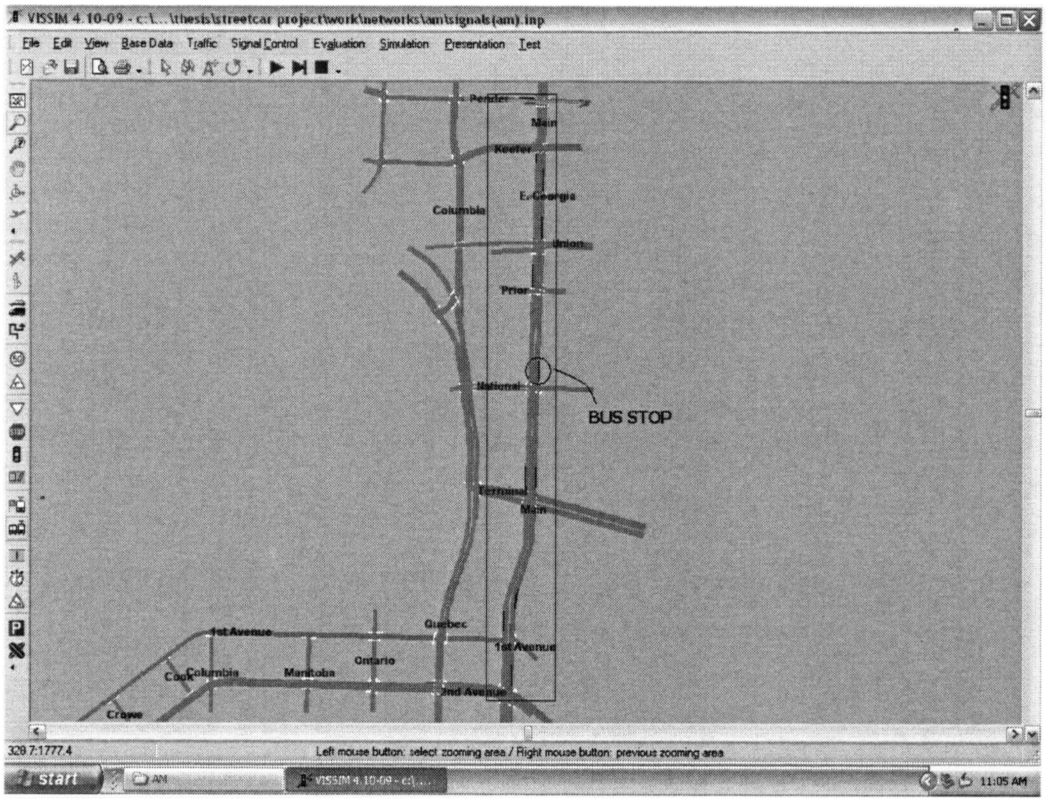

Figure 4-2 Chosen section of Line # 8

The travel time data used was obtained from the results of the AM peak simulation. The AM peak model was simulated to measure the travel times of one bus through the 14 segments for five different exposure rates. Travel time measurements were done for the whole length of each segment. 70 data points were used in the development of the real-time prediction models.

4.2 Linear Regression Models

4.2.1 Software Package

Microsoft EXCEL 2003 was used and its built-in function LINEST to develop linear regression arrival prediction models. This function calculates the fit for a straight line by using the least squares method to calculate to fit the data, and returns an array that describes the line. The equation for the line is:

$$y = m_1x_1 + m_2x_2 + ... + m_nx_n + b \tag{4-1}$$

In the developed models, y is the predicted time which is a function of the independent variables x_i. The constant b is the y-intercept.

4.2.2 Variables & Development Procedure

In previous research on transit arrival prediction models, several variables were used which include distance, dwelling time, and average speed. In this thesis, some new variables were considered. Several arrival prediction models were developed using various combinations of these variables. The variables included:

- DISTANCE: defined as the distance traveled by the transit vehicle to the intersection (unit = meters)
- DWELL: defined as the dwelling time of the transit vehicle at a stop (unit = seconds)

- VOLUME: defined as the hourly vehicle flow (unit = vehicles per hour)
- GREEN: defined as the length of the green time phase for the transit vehicle approach (unit = seconds)
- CYCLE: defined as the cycle length of the downstream signal where TSP is to be applied (unit = seconds)
- g/C: defined as the ratio of green time over the cycle length (unitless)
- STOPS: defined as the number of bus stops in the segment (unitless)
- QUEUES: defined as the number of times a transit vehicle stops due to queues infront of it before it passes the downstream signal (unitless)
- SPD_IN: defined as the average speed of the vehicle in a segment including the time it spends at a transit stop (unit = km/hr)
- SPD_EX: defined as the average speed of the vehicle in a segment excluding the time it spends at a transit stop (unit = km/hr)

The forward procedure (Sawalha, 2002) was used in developing the transit arrival prediction models. In this procedure, variables are added to the model one at a time. If a variable is significant is retained. Variables are tested for significance using the two-tail T-test.

4.2.3 Real-time Prediction Models

Several real-time prediction models were developed and studied. However, only four models were chosen for analysis since all their variables were significant. All the models predicted travel time except for Model 4 which predicted running time instead. Real running times were calculated by subtracting dwelling times form real travel times. The four developed models are presented below:

Table 4-1 Model 1

Prediction Model	r^2	*Variables*	*Coefficients*	*t-ratio*
1	0.934	DWELL	1.886	7.127
		DISTANCE	0.190	9.161
		CONSTANT	0.000	-

Table 4-2 Model 2

Prediction Model	r^2	*Variables*	*Coefficients*	*t-ratio*
2	0.936	g/C	10.97	1.586
		DWELL	1.827	6.911
		DISTANCE	0.172	7.386
		CONSTANT	0.000	-

Table 4-3 Model 3

Prediction Model	r^2	*Variables*	*Coefficients*	*t-ratio*
3	0.944	SPD_EX	-2.302	12.957
		VOLUME	0.005	2.561
		DWELL	1.006	6.828
		DISTANCE	0.212	14.988
		CONSTANT	38.363	-

Table 4-4 Model 4

Prediction Model	r^2	*Variables*	*Coefficients*	*t-ratio*
4	0.675	g/C	-65.430	3.380
		DISTANCE	0.187	9.341
		CONSTANT	43.958	-

4.2.4 Simulation-based Prediction Model

Using the microsimulation data, a highly accurate arrival prediction model was developed using two variables (DWELL and DISTANCE). This arrival prediction model would be used to validate the developed TSP algorithm. The model is presented below in Table 4-5.

Table 4-5 Simulation-based Prediction Model

Prediction Model	r^2	*Variables*	*Coefficients*	*t-ratio*
Simulation Based	0.999	DWELL	1.479	24.132
		DISTANCE	0.135	69.455
		CONSTANT	2.274	-

4.2.5 Discussion

When developing the linear arrival prediction models, many issues were observed regarding the variables, their significance, and the linear models. The issues included the following points:

- To consider the effect of traffic conditions on the prediction of travel time, traffic volume was used as a variable in the prediction models. However, traffic volume surprisingly showed to be insignificant in most of the developed models.
- In most of the developed models, the intercept had very high values even in the models having high r^2. As transit vehicles get closer to the intersection, some significant variables such as distance have low values. And as a result low travel time values are expected. However, with the high constant values in the models, the predicted travel times

were high at locations close to the intersection. This led to developing models with their constants restricted to zero.

- Travel time is expected to have an inverse relationship with the g/C ratio since the more green time in a cycle the less the delay of the bus. However, the g/C variable showed a direct relationship with travel time in Model 2 which was considered one of the good developed models.
- Using the average speed, excluding the time at the bus stops, gave very good results as shown in Model 3. Speed had an inverse relationship with travel time which is logical. This shows that average speed well indicates the effects of traffic conditions on transit vehicles.
- In models 1 and 2, dwelling time had a coefficient of almost 2, which seems an exaggeration of its effect on travel time. Therefore, to get rid of the dwell time effect, running time was predicted rather than travel time in Model 4.

4.3 NEU-Fuzzy Models

A NEU-Fuzzy model is a combination of fuzzy systems and artificial neural networks. NEU-Fuzzy models, unlike artificial neural networks, help the modeler interpret the relationship between input and output variables through linguistic rules. In this thesis, NEU-Fuzzy models were used to develop several transit arrival prediction models.

4.3.1 Software Package

The NEU-Frame software package was used in order to develop the NEU-Fuzzy models. NEU-Frame is an integrated group of Intelligence Technology tools that include Neural Networks and Neuro-fuzzy logic that allows the modeler to use the power of neural nets in developing prediction models. NEU-Fuzzy allows the user the opportunity to evaluate and edit the fuzzy rules during or after training. Developing a fuzzy system is an iterative procedure involving training the network and then editing it with expert knowledge until a solution is developed.

The ability of NEU-Fuzzy to help its user understand the problem at hand and to clarify the role played by the input variables is important. Coupling this with the ability to generate solutions incorporating expert knowledge makes NEU-Fuzzy one of the most powerful and valuable tools available to the modeler. The advantages of NEU-fuzzy are summarized as follows:

- It produces rule sets which enable analysis of the decisions.
- It enables knowledge extracted form data to be combined with expert knowledge.
- It runs an optimizing process that discards the variables that produces minimal influence on the network structure.
- Its models can be constrained by fixed rules.

4.3.2 Network Structure

In order to develop the prediction models, a network template was designed as shown in Figure 4-3. The template contained three datasheets and a NEU-fuzzy network. The first datasheet was named "Training Data" and it contained the training inputs and targets that were needed to train the NEU-fuzzy network. The second datasheet was named "Query Input" and it contained the data that would be queried by the trained network. The third datasheet was named "Query Output" and it would contain the queried results computed by the trained network.

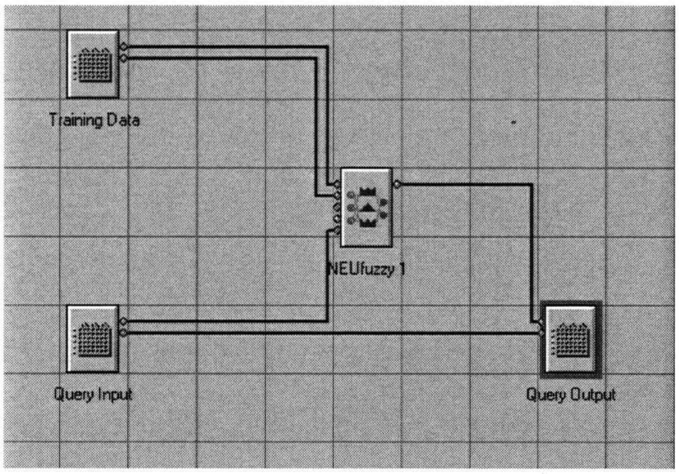

Figure 4-3 NEU-Fuzzy Network Template

The NUE-fuzzy network consists of three layers as shown in Figure 4-4. The first layer contains all the input variables. Each variable is defined as a group of fuzzy sets as shown in Figure 4-5. The second layer consisted of some sub-networks. In these sub-networks, the various variables are weighted and correlated to each other. Finally, the third and last layer contains the output results. The output results are also defined as a group of sets.

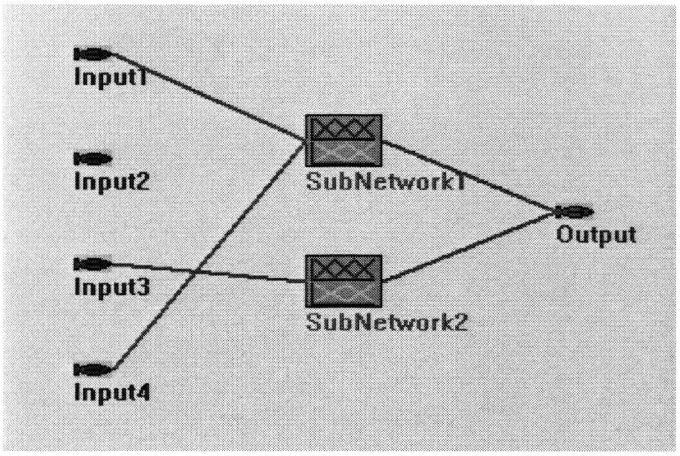

Figure 4-4 NEU-Fuzzy Network Layers

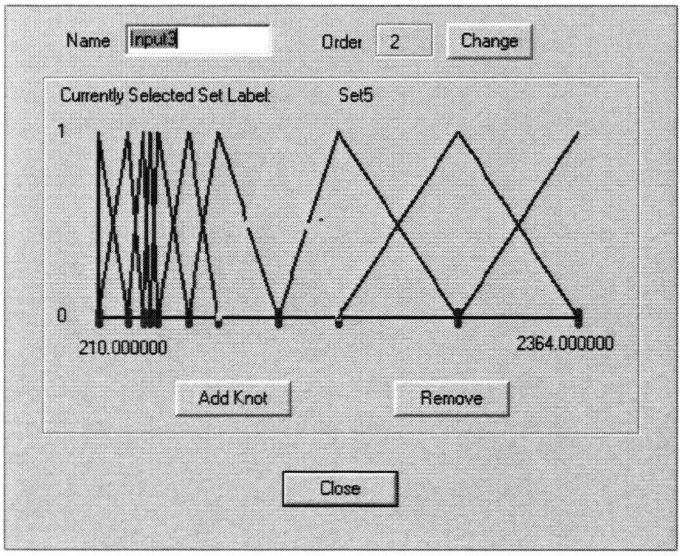

Figure 4-5 Input Variable Sets

The software uses the training data to (1) define the different sets of the input variables and output results, (2) define the links between the input variables and the sub-networks, and (3) decide on the optimum number of sub-networks.

When the network is trained, a list of fuzzy rules is defined. The modeler can modify and change the defined sets, the links, and the number of sub-networks based upon experts' knowledge until a desired model is reached. Usually this is not required as in most cases the network produced is near optimal.

4.3.3 Developed Models

Using the data obtained from the Vancouver's Streetcar Project, four models were developed and modified. From the 70 data points available, 56 were used for training the network, and the other 14 were queried in order to evaluate and check the performance of the developed models. The variables used were the same as the ones used in the linear prediction models.

Model A

In this model, only two variables were used which are DISTANCE and DWELL. This model is similar to Model 1 of the linear prediction models. Figure 4-6 and Table 4-12 show the model and the list of developed fuzzy rules.

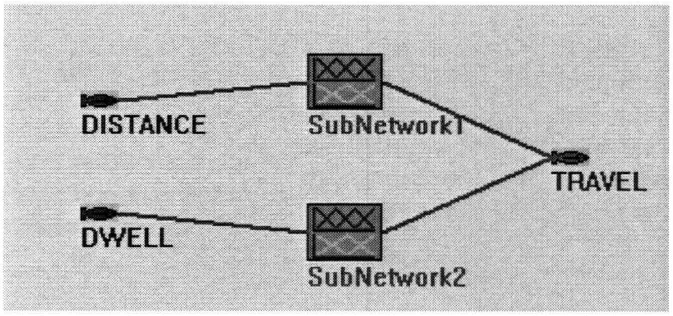

Figure 4-6 Model A

Table 4-6 Fuzzy Rules (Model A)

1: IF DISTANCE is LOW THEN TRAVEL is LOW (0.93) OR TRAVEL is HIGH (0.07)
2: IF DISTANCE is HIGH THEN TRAVEL is LOW (0.04) OR TRAVEL is HIGH (0.96)
3: IF DWELL is LOW THEN TRAVEL is LOW (1.00)
4: IF DWELL is HIGH THEN TRAVEL is HIGH (1.00)

Model B

In this model, a third variable was used which is g/C. This model is similar to Model 2 of the linear prediction models. Figure 4-7 and Table 4-13 show the model and the list of developed fuzzy rules.

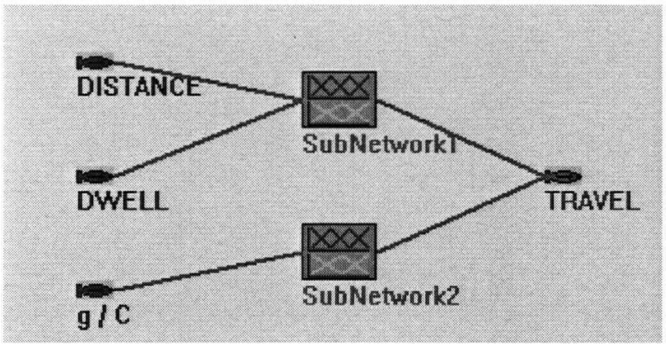

Figure 4-7 Model B

Table 4-7 Fuzzy Rules (Model B)

1: IF DISTANCE is LOW AND DWELL is LOW THEN TRAVEL is LOW (0.35) OR TRAVEL is HIGH (0.65)
2: IF DISTANCE is HIGH AND DWELL is LOW THEN TRAVEL is LOW (0.29) OR TRAVEL is HIGH (0.71)
3: IF DISTANCE is LOW AND DWELL is HIGH THEN TRAVEL is LOW (0.16) OR TRAVEL is HIGH (0.84)
4: IF DISTANCE is HIGH AND DWELL is HIGH THEN TRAVEL is HIGH (1.00)
5: IF g / C is SET1[1] THEN TRAVEL is LOW (0.23) OR TRAVEL is HIGH (0.77)
6: IF g / C is SET2 THEN TRAVEL is LOW (0.15) OR TRAVEL is HIGH (0.85)
7: IF g / C is SET3 THEN TRAVEL is LOW (1.00)
8: IF g / C is SET4 THEN TRAVEL is LOW (0.23) OR TRAVEL is HIGH (0.77)
9: IF g / C is SET5 THEN TRAVEL is LOW (0.26) OR TRAVEL is HIGH (0.74)
10: IF g / C is SET6 THEN TRAVEL is LOW (0.40) OR TRAVEL is HIGH (0.60)
11: IF g / C is SET7 THEN TRAVEL is LOW (0.29) OR TRAVEL is HIGH (0.71)
12: IF g / C is SET8 THEN TRAVEL is LOW (0.29) OR TRAVEL is HIGH (0.71)

[1] Variable values increase as moving from Set 1 to Set 8

In this model, the defined sets of g/c and several of its fuzzy rules were doubted. In reality, as g/C increases, travel time should decrease, and visa versa. Moreover, although the fuzzy rules regarding distance and dwelling time seemed reasonable, there is no clear correlation between them. Thus the model was modified by:

- Adding an additional sub-network linked only to the DWELL variable.
- Deleting the link between the DWELL variable and sub-network 1 that is linked to the DISTANCE variable.
- Redefining the sets of the input variable g/C.

Figure 4-8 and Table 4-14 show the modified model and the new list of developed fuzzy rules.

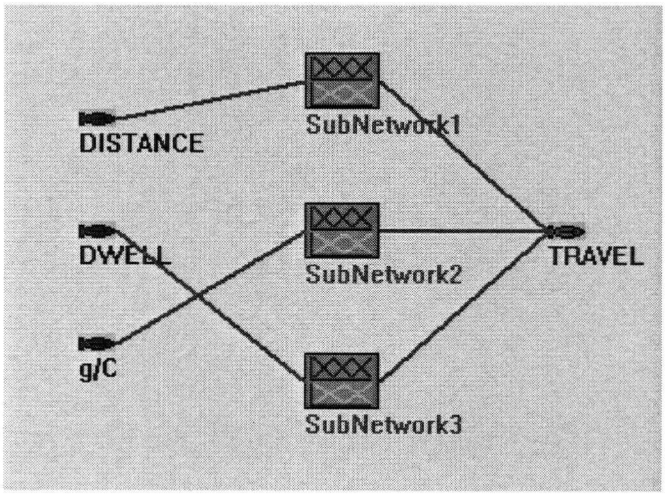

Figure 4-8 Model B (Modified)

Table 4-8 Fuzzy Rules (Model B Modified)

1: IF DISTANCE is LOW THEN TRAVEL is LOW (0.87) OR TRAVEL is HIGH (0.13)
2: IF DISTANCE is HIGH THEN TRAVEL is LOW (0.15) OR TRAVEL is HIGH (0.85)
3: IF g/C is LOW THEN TRAVEL is LOW (0.36) OR TRAVEL is HIGH (0.64)
4: IF g/C is HIGH THEN TRAVEL is LOW (0.93) OR TRAVEL is HIGH (0.07)
5: IF DWELL is LOW THEN TRAVEL is LOW (1.00)
6: IF DWELL is HIGH THEN TRAVEL is HIGH (1.00)

Model C

In this model, four variables were used which are DISTANCE, DWELL, VOLUME, and SPD_EX. This model is similar to Model 3 of the linear prediction models. Figure 4-9 shows the model.

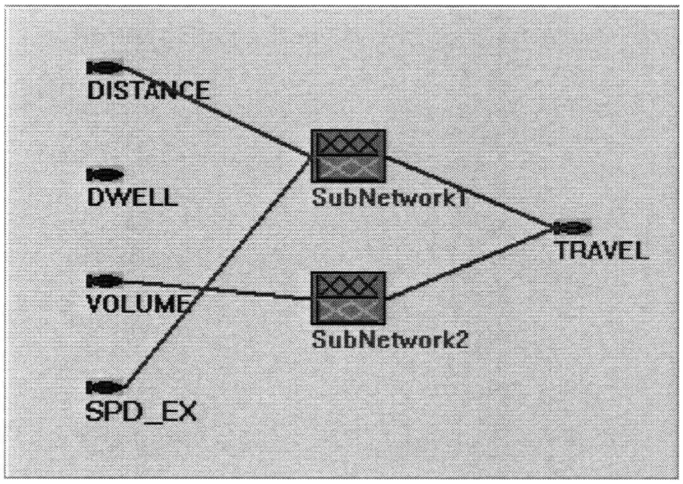

Figure 4-9 Model C

The model gave 59 fuzzy rules all of them seemed to be reasonable, especially when DISTANCE and SPD_EX were shown to be correlated. However, the network assumed that the DWELL variable had insignificant effect on travel time. Thus, a new model was developed by adding an extra sub-network and linking it to the DWELL variable as shown in Figure 4-10.

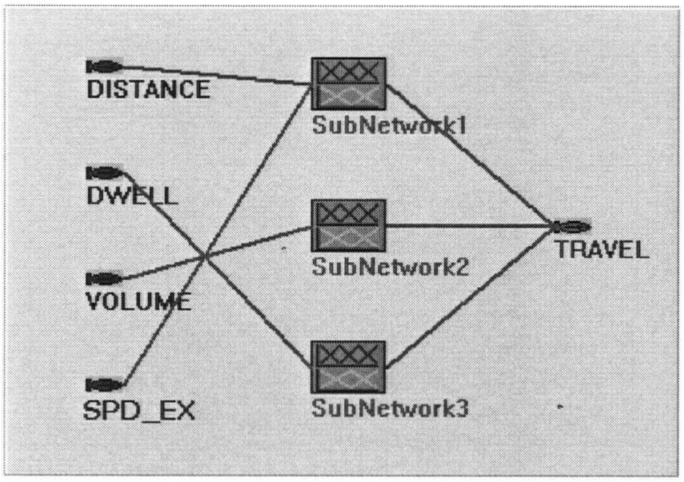

Figure 4-10 Model C (Modified)

Model D

In this model, two variables were used which are DISTANCE and g/C to predict running time rather than travel time. This model is similar to Model 4 of the linear prediction models. Figure 4-11 shows the model.

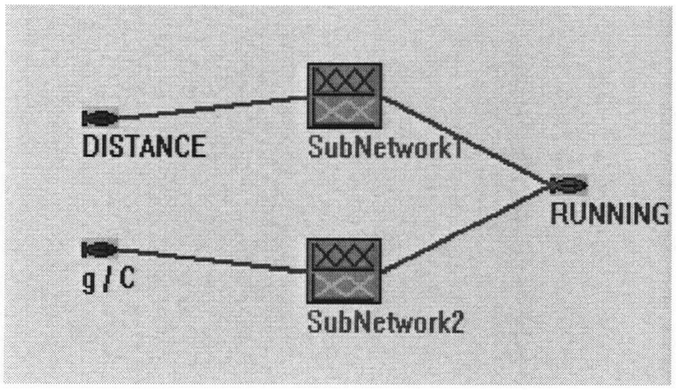

Figure 4-11 Model D

The model gave 15 fuzzy rules. Most of them are not reasonable. Thus a new model was developed by redefining the sets of both variables. Figures 4-12 and 4-13 show an example of the modified sets. Table 4-14 shows the new list of developed fuzzy rules.

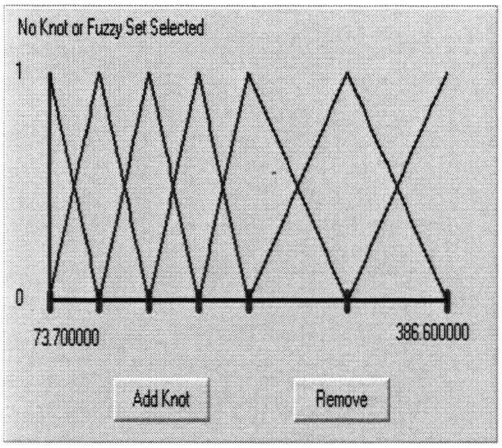

Figure 4-12 Old Sets

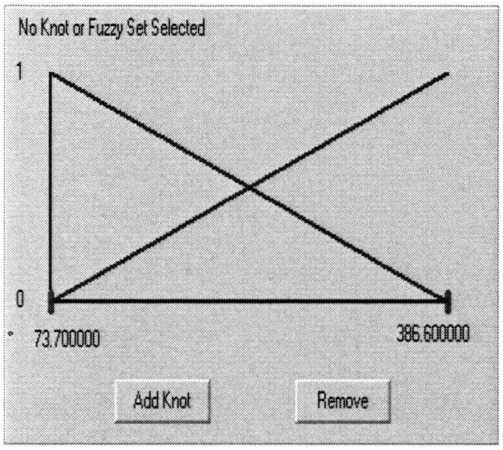

Figure 4-13 New Sets

Table 4-9 Fuzzy Rules (Model D Modified)

1: IF DISTANCE is LOW THEN RUNNING TIME is LOW (1.00)
2: IF DISTANCE is HIGH THEN RUNNING TIME is HIGH (1.00)
3: IF g / C is LOW THEN RUNNING TIME is LOW (0.44) OR RUNNING TIME is HIGH (0.56)
4: IF g / C is HIGH THEN RUNNING TIME is LOW (0.94) OR RUNNING TIME is HIGH (0.06)

4.3.4 Discussion

Using the NEU-Fuzzy networks, four different models were developed and modified. Several issues regarding the used variables and the networks structures were observed and they included the following points.

- As the case in linear prediction models, traffic volumes showed to be insignificant in NEU-Fuzzy models and were not used in any of the developed arrival prediction models.
- When using the g/C variable in the different models, the input sets were always defined into multiple sets which could not be explained.

Moreover, the developed fuzzy rules related to the g/C variable were not logical. The g/C variable was expected to have an inverse linear relationship with travel time. However, the fuzzy rules showed a direct relationship. This is maybe due to the different effects of g/c on travel time. The effects would vary depending on the arrival phase (red or green) and on the queue lengths.

- As the case in linear prediction models, a good prediction model (Model C), was developed when the average speed variable was used. Moreover, the correlation defined between the distance and the speed variables seemed reasonable and valid. This also shows that average speed is a good indicator of traffic conditions.
- In Model C, the dwelling time variable was surprisingly considered insignificant. Thus, it was modified in order to incorporate the effect of dwelling time.

4.4 Models Evaluation & Comparison

All the eight models developed using linear regression and neuro-fuzzy logic were evaluated and compared. In this evaluation, four prediction error measurements were computed to test the prediction models performance.

4.4.1 Error Measurements

The travel time obtained from the simulations was taken to be the true value, and the one from the arrival models was taken to be the predicted value. The four error measurements were:

- *The Mean Squared Error (MSE),* measures the mean of the squares of the deviations from the true value and is calculated as follows:

$$\varepsilon_{ms} = \frac{1}{N}\sum_{t}\left[X_{true}(t) - X_{pred}(t)\right]^2 \tag{4-2}$$

- *The Mean Relative Error,* indicates the expected error as a fraction of the measurement and is calculated as follows:

$$\varepsilon_{mr} = \frac{1}{N}\sum_{t}\left|\left(\frac{X_{true}(t) - X_{pred}(t)}{X_{true}(t)}\right)\right| \tag{4-3}$$

- *The Root Squared Relative Error,* captures large prediction errors and is calculated as follows:

$$\varepsilon_{rs} = \sqrt{\frac{1}{\sum_{t} X_{true}(t)}\sum_{t}\left(\frac{X_{true}(t) - X_{pred}(t)}{X_{true}(t)}\right)^2 X_{true}(t)} \tag{4-4}$$

- *The Maximum Relative Error,* captures the maximum prediction error and is calculated as follows:

$$\varepsilon_{\max} = MAX\left|\left(\frac{X_{true}(t) - X_{pred}(t)}{X_{true}(t)}\right)\right| \tag{4-5}$$

Models 1 & A

These two models used two variables which were DISTANCE and DWELL. Table 4-10 shows the error measurements of the regression model and the NEU-Fuzzy model.

Table 4-10 Errors of Models 1 & A

	Regression	NUE Fuzzy
ε_{ms}	335.928	274.916
ε_{mr}	0.276	0.292
ε_{rs}	0.300	0.281
ε_{max}	0.681	1.032

Models 2 & B

These two models used three variables which were DISTANCE, g/C, and DWELL. Table 4-11 shows the error measurements of the regression model, the NEU-Fuzzy model, and the adjusted NEU-Fuzzy model.

Table 4-11 Errors of Models 2 & B

	Regression	NUE Fuzzy	NEU (Adj)
ε_{ms}	299.176	309.929	275.991
ε_{mr}	0.292	0.212	0.235
ε_{rs}	0.297	0.249	0.247
ε_{max}	0.960	0.390	0.585

Models 3 & C

These two models used four variables which were DISTANCE, VOLUME, SPD_EX and DWELL. Table 4-12 shows the error measurements of the regression model, the NEU-Fuzzy model, and the adjusted NEU-Fuzzy model.

Table 4-12 Errors of Models 3 & C

	Regression	NUE Fuzzy	NUE (Adj)
ε_{ms}	149.654	8.517	56.747
ε_{mr}	0.214	0.033	0.061
ε_{rs}	0.211	0.051	0.112
ε_{max}	0.640	0.206	0.468

Models 4 & D

These two models used two variables to predict running time rather than travel time. The used variables were DISTANCE and g/C. Table 4-13 shows the error measurements of the regression model, the NEU-Fuzzy model, and the adjusted NEU-Fuzzy model.

Table 4-13 Errors of Models 4 & D

	Regression	NUE Fuzzy	NUE (Adj)
ε_{ms}	336.419	1105.959	357.787
ε_{mr}	0.319	0.287	0.296
ε_{rs}	0.344	0.577	0.348
ε_{max}	0.697	1.584	0.682

4.4.2 Discussion

An error analysis was conducted on both linear and NEU-fuzzy arrival prediction models. Several observations were made regarding the results and the performances of the models. The observations are summarized below:

- The results of the error analysis showed that the NEU-fuzzy models outperformed the linear regression models in most cases. This is reasonable since the assumption that all variables have linear relationship with travel time may not be valid.
- The modified NEU-fuzzy models had small relative mean error differences from basic NEU-fuzzy models when. This is because the changes made were only related to the correlation between the different variables and to the definition scheme of the input variables. Input variables were defined as sets ranging from LOW to HIGH.

- In Models 3 and C, where speed was used a variable, both the regression and the NEU-fuzzy models had the lowest error values compared to the other models.
- Model C gave smaller errors when compared to the other models, which supports the significance of the average speed variable.
- Both the regression and NEU-fuzzy models that predicted running time rather than travel time gave the highest error values when compared to the other models. This shows that dwelling time has several effects on travel time such as effects on the accelerations and decelerations of transit vehicles.
- In most of the developed models, the probability of having a prediction error increased when the dwell time is equal to zero. This supports what was stated earlier regarding the several effects of dwelling time on travel time. When dwelling time was taken equal to zero, any other effects of dwelling time were neglected, and therefore prediction errors increased.

CHAPTER 5 ALGORITHMS DEVELOPMENT

5.1 General

The third component of this research was to develop an adaptive algorithm that provides transit vehicles with the most suitable transit priority strategy at the signalized intersections. This algorithm would make use of the AVL system and the transit arrival prediction models developed earlier.

The algorithms were developed on VAP. During a VISSIM simulation the VAP control program is called by VISSIM every simulation time step after the vehicle movements. Usually the entire program is executed, even when an interstage, defined earlier as the transition from one stage to another, is running. This allows detector values to be collected and evaluated during an active interstage.

Five different algorithms were developed:

1) Classic Algorithm: An algorithm that was used to run the conventional TSP system, which relied mainly on check-in and check-out detectors and used simple green extension or red truncation strategies.
2) Dynamic Algorithm: An algorithm that was used to run the dynamic TSP system that utilizes the modeled AVL system and the developed arrival prediction models.

3) Adjusted Dynamic algorithm: A modification done to the Dynamic Algorithm to check the performance of the cycle extension strategy.
4) Bayes Algorithm: An algorithm that was used to run the dynamic TSP system. In this algorithm however, the predicted travel times would be updated with recent travel time data using Bayes Technique.
5) Kalman Algorithm: An algorithm that was used to run the dynamic TSP system. In this algorithm however, the predicted travel times would be updated with recent travel time data using Kalman Filter.

5.2 Classic Algorithm

The classic algorithm was designed in order to compare the performance of the conventional TSP system to that of the newly developed dynamic TSP system. Conventional TSP systems are usually composed of a fixed detection system and an algorithm that provides priority to transit vehicles when detected. Figure 5-1 presents the flowchart of the classic algorithm. The structure of the classic algorithm consists of two components: a detection system and TSP decision scheme. The two components of the algorithm are described in detail in the following subsections.

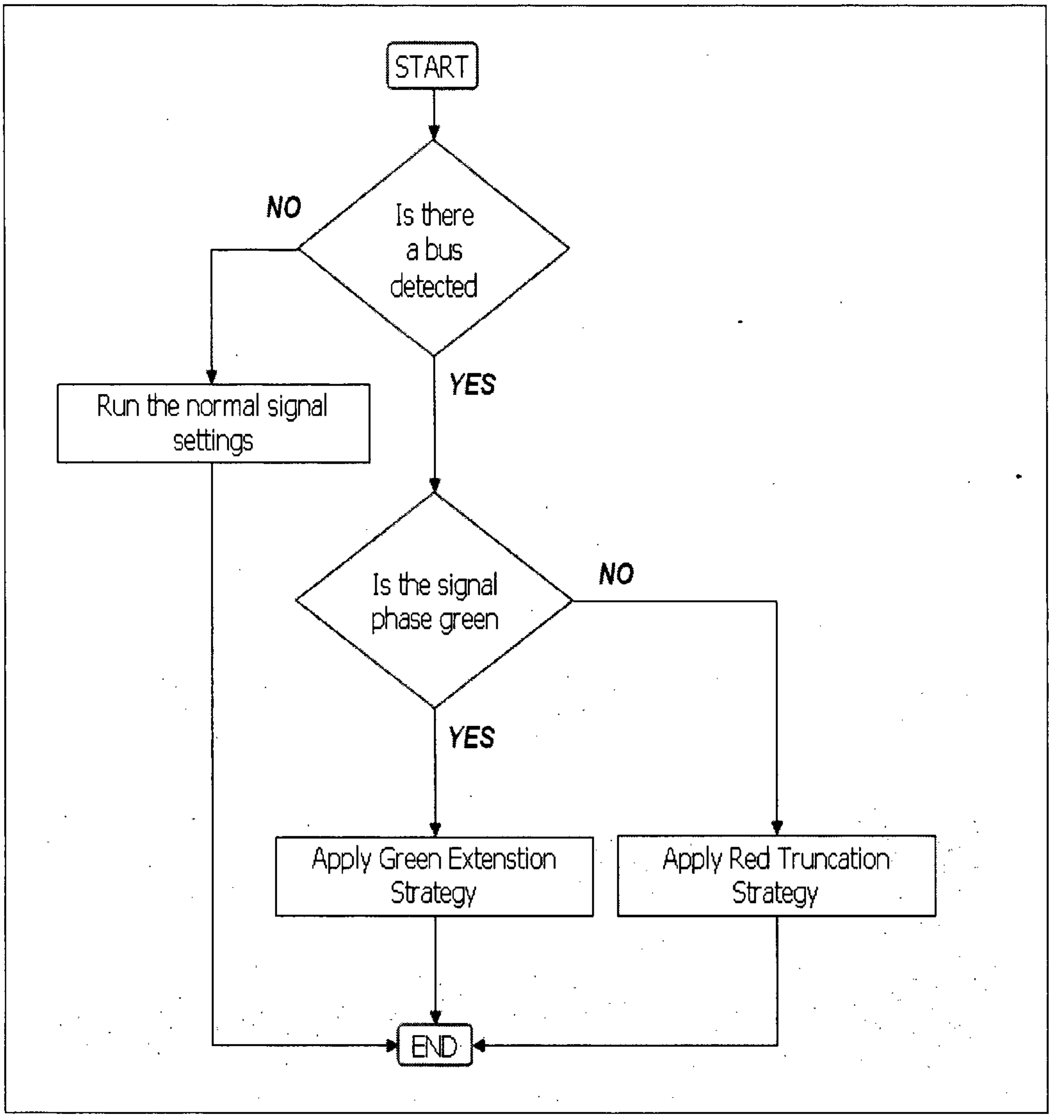

Figure 5-1 Classic Algorithm Flowchart

5.2.1 Detection System

The detection system of the classic TSP system, illustrated clearly in Figure 5-2, was composed of two detectors:

- A check-in detector that was located 50 meters away from the signal's stop line. Its main operation was to notify the traffic signal when a bus is approaching.

- A check-out detector that was located at the signal's stop line. It notified the traffic signal when the bus entered the intersection and passed the signal's stop line.

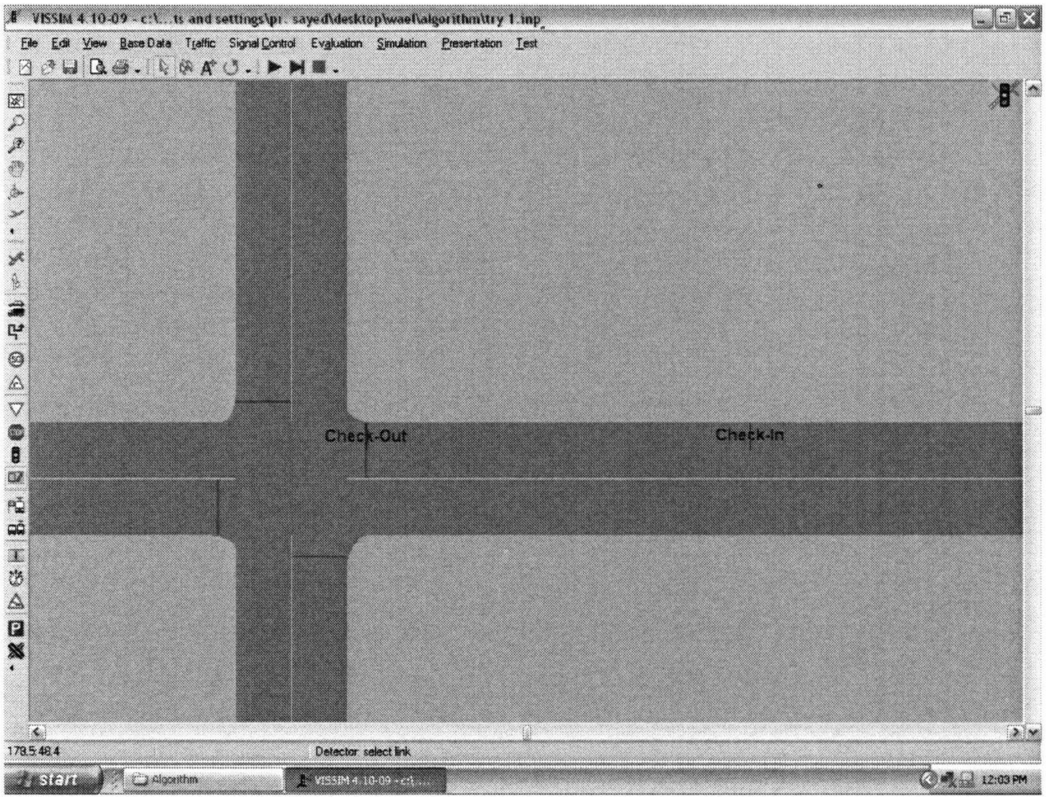

Figure 5-2 Classic Detection System

5.2.2 Decision & TSP Solutions

With the conventional TSP system, the decision would be instantly taken as a transit vehicle passes over the check-in detector. If no detection occurred, the algorithm would run the normal signal settings. When taking the decision, one of two TSP strategies would be chosen:

- Green Extension: If the signal phase is green when the bus is detected, the green phase will be extended until the bus checks out or the maximum extension limit is reached. In order to achieve offset recovery, the succeeding green phase of the opposing approach will be reduced with a time interval equal to the extension.
- Red Truncation: If the signal phase is yellow or red when the bus is detected, the red phase will be truncated with a time interval equal to the maximum extension, taking into consideration the minimum signal requirements for pedestrians. In order to achieve offset recovery, the green phase will be extended with a time interval equal to the maximum extension.

5.3 Dynamic Algorithms

In this thesis four dynamic algorithms were developed (Dynamic, Adjusted Dynamic, Bayes, and Kalman Algorithms). The four algorithms would be used to test the performance of the developed dynamic TSP system. The algorithms share the main structure; however they differ in the computational procedures and data assignments. Figure 5-2 presents the flowchart of the four dynamic algorithms.

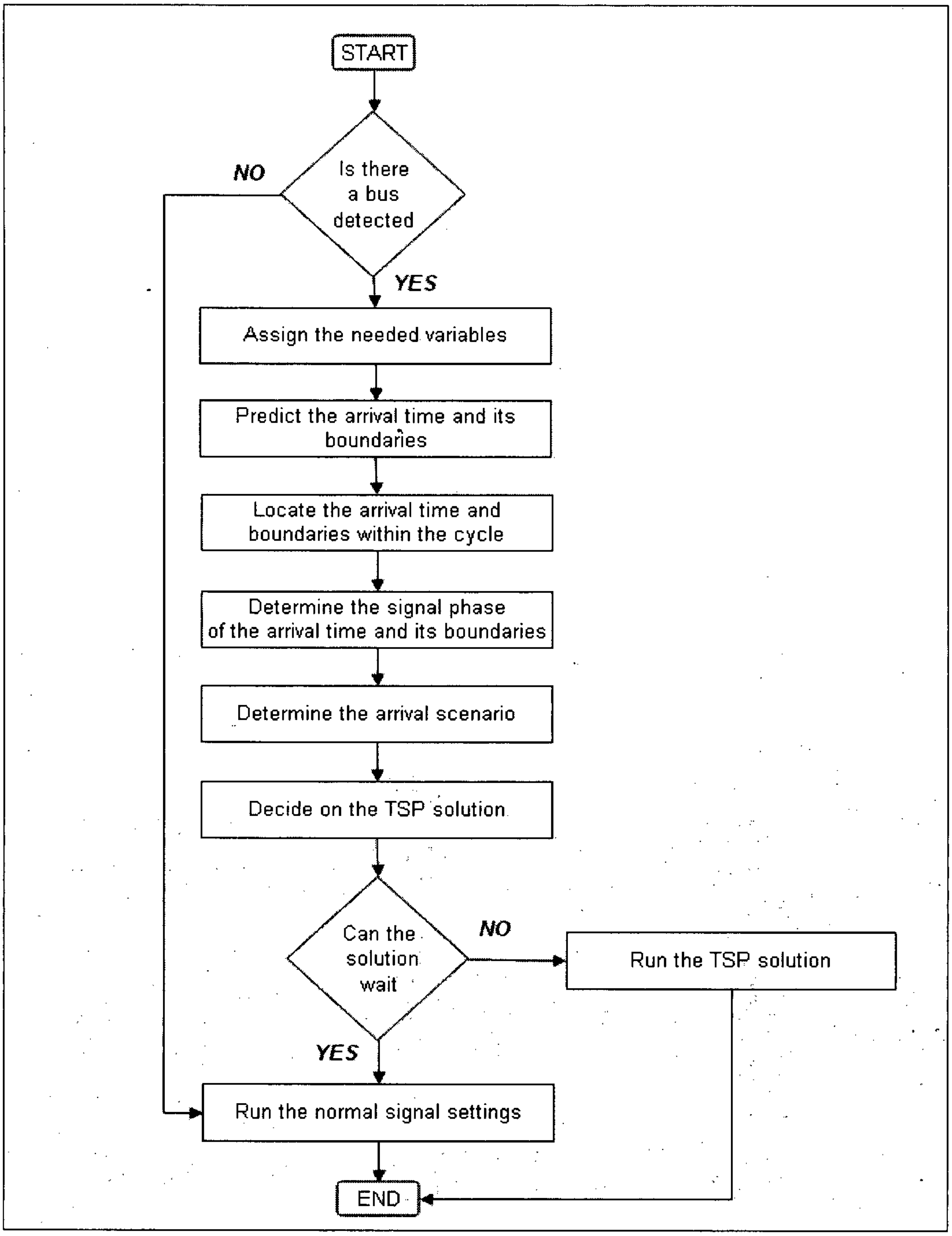

Figure 5-3 Dynamic Algorithms Flowchart

As seen in the flowchart, the algorithms are composed of seven major sequential steps. Each of these steps is described and discussed in the following subsections.

5.3.1 Transit Vehicle Detection

The detectors in the simulated intersection were modeled in a way that resembles the performance of an AVL system. In all four algorithms, whenever a transit vehicle was detected, several variable values were determined and used in the prediction models.

In the Bayes Algorithm, whenever a transit vehicle was detected, the travel time of the previous bus measured from the detector to the signal stop line would be retrieved. The previous bus travel time would be used by Bayes Technique to adjust the model's predicted travel time.

In the Kalman Algorithm, whenever a transit vehicle was detected, the average travel time of all previously simulated busses and their variance would be retrieved and used by Kalman Filter to adjust the model's predicted travel time.

5.3.2 Arrival Time Prediction

This is one of the most crucial steps in the algorithm, and is what differentiates the developed Algorithms. In the Dynamic and the Adjusted Dynamic Algorithms, one of the prediction models developed earlier was used without any adjustments to predict the transit arrival times.

Lower and upper boundaries of the arrival time were defined to determine the arrival scenarios in a later stage. The time length between the predicted arrival time and any of its boundaries was defined as the *boundary length*. The lower and upper boundaries, in some cases, would be about one standard deviation from the arrival time. Sayed T. & Zein S. (1998) estimated the variance of a linear prediction model as shown in Equation 5-1. This eqцation was used in the algorithms to calculate the variance of the prediction models.

$$Var = s_d^{\,2}\left(1+\frac{1}{n}\right)+s_{a1}^{\,2}\left(x_1 i-\overline{x}_1\right)^2+.....+s_{an}^{\,2}\left(x_n i-\overline{x}_n\right)^2 \qquad (5\text{-}1)$$

Var = is the variance of the prediction model

s_d = is the standard error in y (the predicted dependent value)

s_a = is the standard error of an independent variable

n = is the total number of used independent variables

x_i = is the value of this variable used in the prediction model

$\overline{x}$ = is the average value of this variable

In the Bayes Algorithm, the arrival time prediction was composed of three steps. The first step was to calculate the arrival time and its variance from the prediction model. The second step was to calculate alpha, which was defined as the degree of confidence in the models predicted value, as shown in Equation 5-2. Finally, alpha would be used with the predicted and previous arrival times, as shown in Equation 5-3, to calculate the arrival time that would be used in the algorithm.

$$\alpha = \frac{1}{1+\left(\frac{Var_{Model}}{P_{Model}}\right)} \tag{5-2}$$

$$P_{Bayes} = \alpha P_{Model} + (1-\alpha) P_{Previous} \tag{5-3}$$

In the Kalman Algorithm, the arrival time prediction was composed of three steps. The first step was to calculate the arrival time and its variance from the prediction model. The second step was to calculate average time and variance of all previous simulated busses. Finally, both arrival times and variances would be used together, as shown in Equation 5-4, to calculate the arrival time that would be used in the algorithm.

$$P_{Kalman} = \frac{Var_{Previous}}{Var_{Model} + Var_{Previous}} P_{Model} + \frac{Var_{Model}}{Var_{Model} + Var_{Previous}} P_{Previous} \tag{5-4}$$

5.3.3 Arrival Time Allocation

After predicting the arrival time and the lower and upper boundaries, the location of the arrival time and its boundaries within the cycle were determined. In all the four algorithms, the arrival time allocation process was similar and performed in four steps.

(1) The arrival time, in seconds, was added to the current cycle second.

(2) The number of cycles left until arrival was calculated.

(3) The location of the arrival time within the cycle was determined by subtracting the cycles from the sum of the arrival and current times.

(4) The same procedure was repeated for each boundary.

5.3.4 Signal Phase Determination

The signal phase at which the transit vehicles would arrive at were determined whether green, yellow, or red. In VAP a signal phase is defined using parameters that include GE (Green End), YE (Yellow End), and RE (Red End). In a cycle, a signal phase can be arranged in one of three scenarios as shown in Figure 5-4.

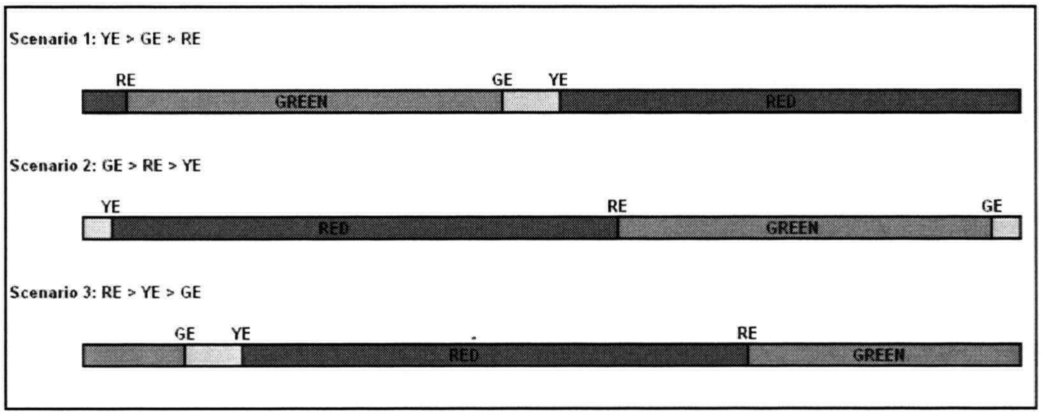

Figure 5-4 Phase Scenarios

Knowing the three parameters GE, YE, and RE the phase scenario was determined. The phases of each the arrival time, the lower boundary, and the upper boundary would be also determined.

5.3.5 Arrival Scenario Determination

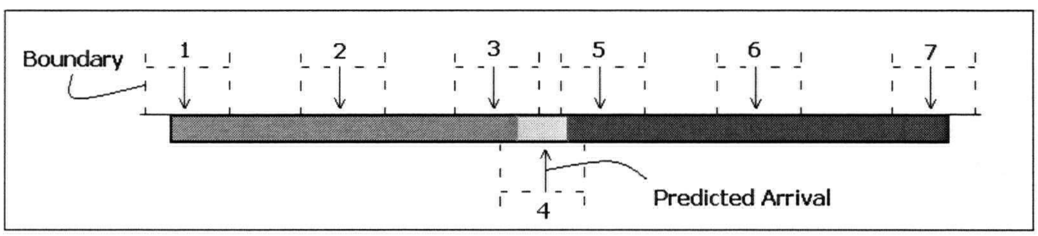

Figure 5-5 Arrival Scenarios

As shown in Figure 5-5, seven different arrival scenarios were defined:

- SCENARIO 1: A scenario where the predicted arrival and its upper boundary are located in the green phase, while the lower boundary is located in the preceding red phase.
- SCENARIO 2: A scenario where the predicted arrival and its lower and upper boundaries are all located in the green phase.
- SCENARIO 3: A scenario where the predicted arrival and its lower boundary are located in the green phase, while the upper boundary is located in the succeeding yellow or red phase.
- SCENARIO 4: A scenario where the predicted arrival is located at the yellow phase.
- SCENARIO 5: A scenario where the predicted arrival and its upper boundary are located in the red phase, while the lower boundary is located in the preceding yellow or green phase.
- SCENARIO 6: A scenario where the predicted arrival and its lower and upper boundaries are all located in the red phase.
- SCENARIO 7: A scenario where the predicted arrival and its lower boundary are located in the red phase, while the upper boundary is located in the succeeding green phase.

In the four dynamic algorithms, after determining the signal phase of the arrival time and its boundaries, the corresponding arrival scenario would be determined. Based on these seven arrival scenarios, the type of solution or TSP strategy would be chosen.

5.3.6 TSP Solutions

The type of solution or TSP strategy chosen was based on the determined arrival scenario. In this research, three TSP solutions were used and are described as follows:

- SOLUTION 1: This solution is based on the Green Extension Strategy. In this strategy, a green extension will be given to the green phase until the bus checks-out the signal line or the maximum extension is reached. Offset recovery is applied in this solution where the succeeding opposing green phase will be reduced with a time interval equal to the extension.
- SOLUTION 2: This solution is based on the Red Truncation Strategy. In this strategy, the red phase will be truncated earlier so that the bus arrives during the green phase. Offset recovery is also applied in this solution where the green phase will be extended with a time interval equal to the extension.
- SOLUTION 3: This solution is based on the Cycle Extension Strategy. In this strategy, the cycle will be extended to have a length of one and a half of the normal cycle length in order to make sure that the transit vehicle will arrive during a green phase and in order to retain signal coordination. The solution will be executed for two cycles, replacing

three cycles of normal cycle length, then the cycle will be reduced again to its normal length.

In each of the four dynamic algorithms and for each scenario, one of the three defined solutions would be used. Table 5-1 shows the scenarios and their corresponding solutions for each algorithm.

Table 5-1 Scenarios & Solutions

	DYNAMIC	***DYN. ADJ***	***BAYES***	***KALMAN***
Scenario 1	SOLUTION 2	SOLUTION 2	SOLUTION 2	SOLUTION 2
Scenario 2	DO NOTHING	DO NOTHING	DO NOTHING	DO NOTHING
Scenario 3	SOLUTION 1	SOLUTION 1	SOLUTION 1	SOLUTION 1
Scenario 4	SOLUTION 1	SOLUTION 1	SOLUTION 1	SOLUTION 1
Scenario 5	SOLUTION 1	SOLUTION 1	SOLUTION 1	SOLUTION 1
Scenario 6	SOLUTION 3	SOLUTION 2	SOLUTION 2	SOLUTION 2
Scenario 7	SOLUTION 2	SOLUTION 2	SOLUTION 2	SOLUTION 2

Although in Scenario 2 the bus is always expected to arrive during the green phase, SOLUTION 1 was programmed to be used only if the bus had unexpected excessive delays where an extension would be given to the phase. Moreover, SOLUTION 3 was only used in the Dynamic Algorithm in Scenario 6. This is the only difference between the Dynamic and Adjusted Dynamic Algorithms. This adjustment was made to check the performance and effect of this unpopular strategy on the intersection and algorithm at different exposure rates. Although presented theoretically in many papers, the application of the Cycle Extension Strategy in real life projects is rare.

In order to understand the way the different solutions work with the scenarios, Figure 5-6 illustrates the solutions of the different scenarios for a period of three normal cycle lengths.

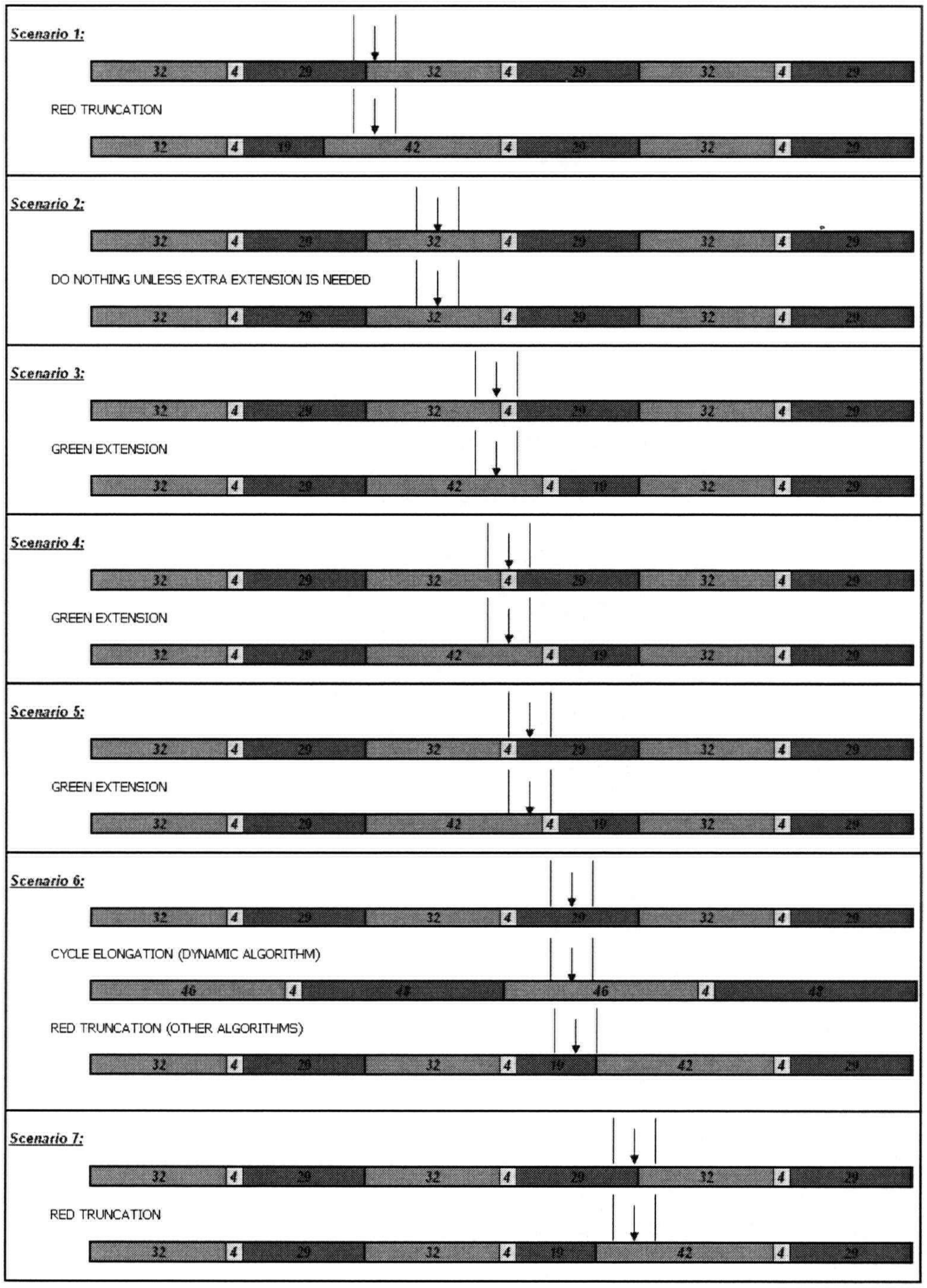

Figure 5-6 Scenarios & Solutions

Although the cycle extension strategy seems more beneficial than the red truncation strategy, its effects on the queues and algorithm are unpredicted.

5.3.7 Taking The Decision

The time at which the decision is taken is one of the most important and dynamic components of the developed algorithms. After running all the preceding steps and determining the most suitable TSP solution, the algorithms would check whether to apply the TSP solution immediately or delay the application. If the algorithm would wait, all the preceding steps would be repeated again to check whether the previously determined arrival scenario and TSP solution were still valid. This component of the dynamic algorithms tries to choose the optimum TSP solution taking into consideration any traffic changes that might cause changes in the arrival scenario. When to take the decision was based on the arrival scenario and on the nature of the chosen TSP solution. The developed decision time limits for the seven arrival scenarios are listed in Table 5-2 and illustrated in Figure 5-7.

Table 5-2 Scenarios' Time Limits

Scenario	Decision Time Limits
Scenario 1	At the start of the preceding red phase
Scenario 2	N / A
Scenario 3	At the start of the current green phase
Scenario 4	At the start of the preceding green phase
Scenario 5	At the start of the preceding green phase
Scenario 6 (SOLUTION 3)	At the start of the second preceding green phase

Scenario 6 (SOLUTION 1)	At the start of the current red phase
Scenario 7	At the start of the current red phase

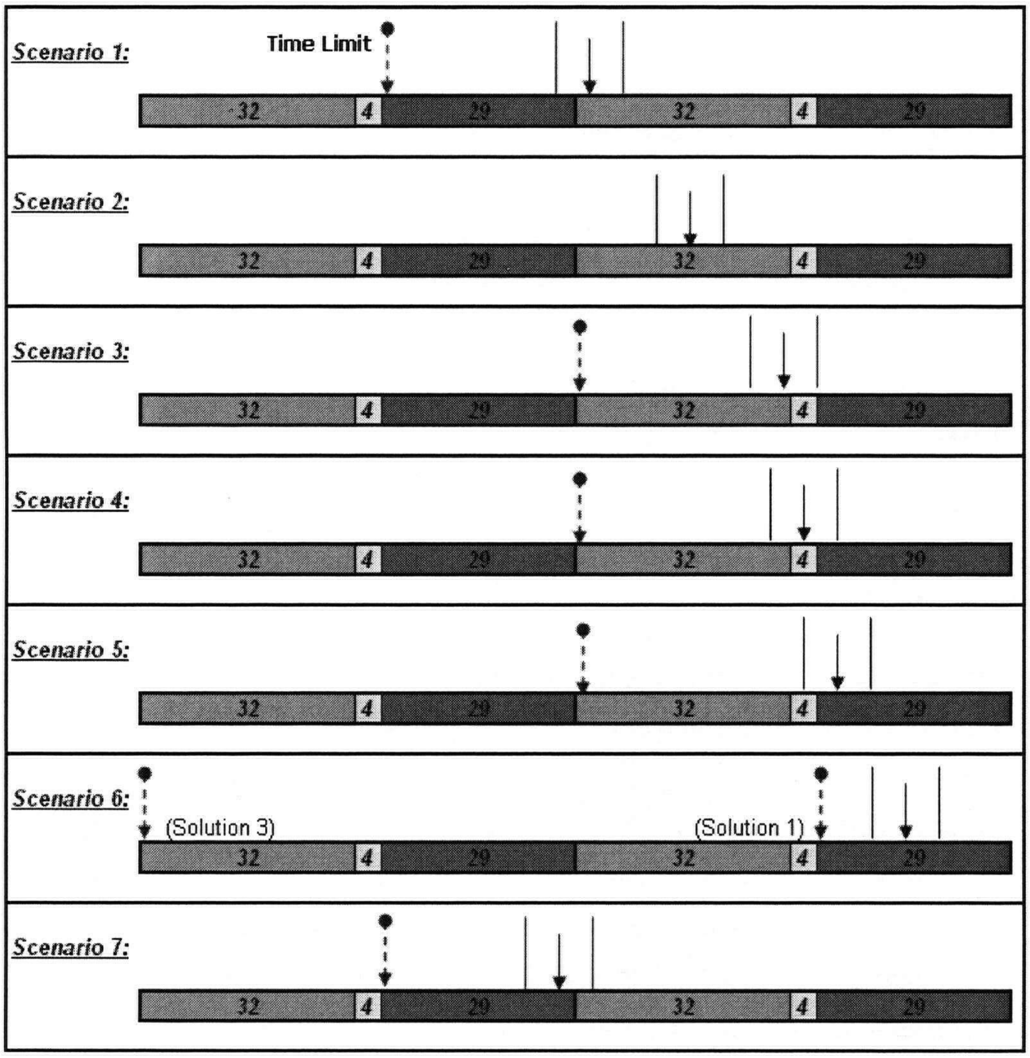

Figure 5-7 Time Limits

The dynamic algorithm runs the normal signal settings until a decision is taken. When a decision is taken, none of the previously described steps will be executed, except for detection, until the solution ends and offset recovery is achieved. When the next bus arrives, the algorithm will be ready to grant it the priority it needs.

5.3.8 Discussion

As described earlier, four different algorithms were developed. While developing these algorithms some observations were made regarding their structures and the characteristics of their sequential steps. The observations are summarized below:

- Only the linear regression models were used in the algorithms to predict the bus arrival times. The use of the NEU-fuzzy models might give better results. However, they need to be linked to the algorithm by an OLE object that requires a high speed processor that would be able to run both software packages at the same time.
- Due to the advanced detection system used by the algorithms, which rely on AVL systems as virtual detectors, the arrival scenario of the detected bus would change depending on the existing traffic conditions. This dynamic response is used by the algorithms to choose the most suitable TSP solution.
- In the developed algorithms, only three types of solutions or TSP strategies were used. However, as shown in the literature review, there are a number of other strategies that could be used with the dynamic algorithms.

- The assumed decision time limits are dependent on the arrival scenario, the type of TSP solution to be applied, and on other factors such as minimum green times and maximum allowable extension.
- In the developed algorithms, offset recovery was taken into consideration. However, signal compensation for side streets traffic was not considered.

CHAPTER 6 SIMULATIONS RESULTS & DISCUSSION

This chapter presents the results of the simulations and their discussion. Two sets of results are presented: the first set is for the algorithms validation process and the second set is for the performance testing of the real-time transit arrival prediction models.

6.1 Algorithms Validation

As discussed earlier, a highly accurate arrival prediction model was developed to validate the performance of the algorithms. The highly accurate arrival prediction model was developed using data obtained from the simulated intersection.

6.1.1 Methodology

For each of the five algorithms, 3 different simulations were executed, each with a different seed value, using two different exposure rates, 750 and 1000 vehicles/hour, for a total of 30 simulation runs. The simulation period was set to 2 hours (7200 seconds). In this time period, 40 different busses would pass through the intersection.

6.1.2 Results

The results included the delay time of the buses at the intersection. In VISSIM, delay is defined for a vehicle whenever its speed drops below 5 km/hr. The results also included the sum of delay of all busses, the average delay, and the standard deviation for each algorithm. The results are shown in Tables 6-1 and 6-2.

Table 6-1 Validation Results (Exposure = 750 vehs/hr)

	Classic	*Dynamic*	*Adj Dyn*	*Bayes*	*Kalman*
SUM	*214.4*	*141.2*	*178.1*	*196.9*	*180.8*
AVG	*5.4*	*3.5*	*4.5*	*4.9*	*4.5*
S. D.	*7.0*	*3.8*	*5.9*	*6.5*	*5.8*

Table 6-2 Validation Results (Exposure = 1000 vehs/hr)

	Classic	*Dynamic*	*Adj Dyn*	*Bayes*	*Kalman*
SUM	*232.7*	*229.3*	*203.1*	*215.0*	*218.4*
AVG	*5.8*	*5.7*	*5.1*	*5.4*	*5.5*
S. D.	*6.9*	*5.5*	*5.8*	*6.1*	*6.0*

6.1.3 Discussion

Using the simulation-based highly accurate arrival prediction model, the developed algorithms were checked for their validity. Issues regarding the performance of the dynamic algorithms are described below:

- All the developed dynamic algorithms produced less average delay than the classical algorithm. This supports the hypothesis that the Dynamic TSP system outperforms the Classical TSP system.
- At medium traffic flow levels (750 vehicles/hour), the Dynamic Algorithm outperformed the Adjusted Dynamic Algorithm. However at

high traffic flow levels (1000 vehicles/hour), the Adjusted Dynamic Algorithm outperformed the Dynamic Algorithm.

- The cycle extension strategy was effective medium traffic flow levels. However, at high traffic flow levels, the strategy would cause high queues in the subsequent phases that might cause delays for succeeding busses.
- Although Kalman Filter was used in earlier projects and gave good prediction results (Shalaby & Farhan, 2004), in this research Kalman and Bayes Algorithms gave results with higher delays than the Dynamic and Adjusted Dynamic Algorithms. Kalman and Bayes Algorithms, as in previous projects, used previous bus travel times in predicting the new travel times. Previous travel times are not accurate enough to be used in TSP application. The previous bus might have arrived at a green or red phase. Therefore, the travel time of the previous bus did not really help in determining the travel time of the current bus since the arrival phase and the queue length could be different.

6.2 Real-time Prediction Models

After validating the dynamic algorithms, the performance of the real-time prediction models were tested. Four different models were selected and analyzed earlier. From these models, Models 1 and 2 would only be tested. Model 4 did not provide accurate results and therefore was not tested. And although model 3 gave the best results, it was not tested because it was hard to predict the variable SPD_EX. The variable was defined earlier as the average speed of the bus throughout the whole segment, excluding the time spent at bus stops.

6.2.1 Methodology

For each of the five algorithms, 2 different simulations were executed, each with a different seed, at two different exposure rates, 750 and 1000 vehicles/hour, for a total of 64 simulation runs The algorithms were also tested for two different boundary lengths, 5 and 10 seconds. Boundary length, B, was defined earlier as the time length between the predicted arrival time and its upper or lower boundary. Changes in the boundary length might cause significant changes in the transit arrival scenario, which in turn would affect the time and type of TSP decision. The simulation period was set to 2 hours (7200 seconds). In this period, 40 different busses would pass through the intersection.

6.2.2 Results

The results included the delay time of the buses at the intersection. The results also included the sum of delay of all busses, the average delay, and the standard deviation for each algorithm. The results are shown in Tables 6-3 through 6-10.

Table 6-3 Model 1 (Exposure = 750veh/hr, B = 5s)

	Classic	Dynamic	Adj Dyn	Bayes	Kalman
SUM	222.3	197.9	236.3	199.9	225.6
AVG	5.6	4.9	5.9	5.0	5.6
S. D.	7.3	5.3	6.4	6.4	6.3

Table 6-4 Model 1 (Exposure = 750veh/hr, B = 10s)

	Classic	Dynamic	Adj Dyn	Bayes	Kalman
SUM	222.3	178.6	183.7	187.7	225.6
AVG	5.6	4.5	4.6	4.7	5.6
S. D.	7.3	6.0	5.9	6.3	6.3

Table 6-5 Model 1 (Exposure = 1000veh/hr, B = 5s)

	Classic	Dynamic	Adj Dyn	Bayes	Kalman
SUM	244.7	247.3	287.8	240.4	284.0
AVG	6.1	6.2	7.2	6.0	7.1
S. D.	7.2	6.0	6.6	6.6	6.7

Table 6-6 Model 1 (Exposure = 1000veh/hr, B = 10s)

	Classic	Dynamic	Adj Dyn	Bayes	Kalman
SUM	244.7	215.9	211.2	223.6	181.1
AVG	6.1	5.4	5.3	5.6	4.5
S. D.	7.2	6.4	5.7	6.3	5.2

Table 6-7 Model 2 (Exposure = 750veh/hr, B = 5s)

	Classic	Dynamic	Adj Dyn	Bayes	Kalman
SUM	222.3	234.0	262.2	199.9	225.6
AVG	5.6	5.8	6.6	5.0	5.6
S. D.	7.3	6.3	6.4	6.4	6.3

Table 6-8 Model 2 (Exposure = 750veh/hr, B = 10s)

	Classic	Dynamic	Adj Dyn	Bayes	Kalman
SUM	222.3	189.5	215.6	187.7	225.6
AVG	5.6	4.7	5.4	4.7	5.6
S. D.	7.3	5.8	6.3	6.3	6.3

Table 6-9 Model 2 (Exposure = 1000veh/hr, B = 5s)

	Classic	Dynamic	Adj Dyn	Bayes	Kalman
SUM	244.7	270.7	305.3	240.4	284.0
AVG	6.1	6.8	7.6	6.0	7.1
S. D.	7.2	6.7	6.7	6.6	6.7

Table 6-10 Model 2 (Exposure = 1000veh/hr, B = 10s)

	Classic	Dynamic	Adj Dyn	Bayes	Kalman
SUM	244.7	239.0	283.7	223.6	181.1
AVG	6.1	6.0	7.1	5.6	4.5
S. D.	7.2	6.0	6.7	6.3	5.2

6.2.3 Discussion

Several observations were made regarding the performance of some of the real-time prediction models and are described below:

- In the Dynamic and the Adjusted Dynamic Algorithms, Model 2 produced better results than those of the classical system only at medium traffic flow levels (750 vehicles/hour) when B was taken equal to 10 seconds.
- In the Dynamic Algorithm, Model 1 produced better results than those of the classical system at both exposure rates when the boundary length was taken equal to 10 seconds, and at medium traffic flow levels (750 vehicles/hour) when B was taken equal to 5 seconds.
- In the Adjusted Dynamic Algorithm, Model 1 produced better results than those of the classical system at both exposure rates when the boundary length was taken equal to 10 seconds.

- In general, Model 1 outperformed Model 2 in all the developed dynamic algorithms. This may be due to the ambiguous effect of g/C on travel time.
- When the value of the boundary length increased, better results were achieved.
- When the exposure level decreased, better results were achieved. This is valid since higher traffic volumes would cause more interaction with transit vehicles, which in turn would increase unanticipated delays.
- Both the Bayes and Kalman Algorithms models produced the same results when they have the same exposure rate and boundary length values. This shows that the algorithms relied solely on the previous data rather than the prediction models in predicting the travel times. This was mainly due to the high variances of the prediction models.

CHAPTER 7 CONCLUSION & RECOMMENDATIONS

7.1 Conclusion

Traffic simulation techniques have been used since the early days of the development of traffic theory. The ever-increasing power of personal computers and search for ITS solutions to growing urban transport problems has led to the emergence of a number of Travel Demand Management strategies such as Transit Signal Priority.

In most of the big cities in the world, Transit Management Systems became a necessity rather than an option. With the extremely high demand on trips and the fixed capacity of the existing road networks, urban planners and traffic engineers focused on developing and improving transit facilities and movements. Thus, Transit signal Priority became one of the most popular ITS solutions. With TSP, priority in the traffic network is awarded to person trips rather than vehicle trips.

In this thesis, a Dynamic TSP system has been developed to replace the Conventional TSP system used in most cities. The Dynamic TSP system outperforms the Conventional TSP system due to its three dynamic components that make use of real-time data updates. The three dynamic components are:

- an advanced virtual detection system (AVL system)
- a dynamic transit arrival prediction model
- a dynamic algorithm that decides on the TSP strategy to be applied

Several transit arrival models have been studied and developed. Models were developed using some newly proposed variables. Simulation data and real life data, obtained from Vancouver's New Streetcar Project, were used in the models development. The models were developed using two methods, Linear Regression and NEU-Fuzzy Artificial Neural Networks. The models were refined in some of the developed dynamic algorithms by using Bayes Technique and Kalman Filter. In total eight models were developed and evaluated. The error analysis showed that some newly used variables such as average speed improved the accuracy of the prediction models. In addition, results showed that the neuro-fuzzy models outperformed the linear regression models.

With Microsimulation modeling, the performance of the Dynamic TSP system was tested over the simulated intersection. Four dynamic algorithms were developed and tested. The algorithms used the AVL system to be continuously updated with the location of the desired bus. Then using an arrival prediction model, the algorithm would recursively check the predicted arrival time of the bus until it reaches a limit where a decision should be taken. Based on the arrival scenario of the bus, an appropriate TSP solution would be chosen and applied to the signal.

The four dynamic algorithms were tested under various scenarios that varied the prediction models, the simulation parameters, and the traffic volumes. Results showed that when an accurate prediction model was used, the dynamic TSP system outperformed the conventional one.

Finally, the main contribution of this thesis is the development of the dynamic algorithm that has employed arrival prediction models in TSP applications. Using this algorithm, different detection systems could be used, different arrival prediction models could be evaluated, and different TSP strategies and solutions could be tested and applied.

7.2 Recommendations

This thesis has provided several research ideas that could be studied later on to enhance and improve the developed dynamic TSP system. Improvements can be done to the three dynamic components of the TSP system. The use of AVL system as a detection system is a challenge by itself. Advancements and research in this field would help a lot not only in TSP applications but also in other ITS technologies. Using the AVL system to send other valuable data, such as number of occupants and bus schedule adherence, might benefit the dynamic TSP system in enhancing the TSP algorithm and options.

Moreover, as seen from the results, advancements in the arrival prediction models will significantly improve the dynamic TSP system. As the accuracy of the arrival prediction models increase, better TSP decisions could be taken at earlier times. The use of new variables that measures traffic congestion such as queue length might be very beneficial. Moreover, some of the variables used in the development of the transit arrival prediction models, such as the g/C ratio and the average speed, showed promising results and should be further studied and understood. For instance, the ability to predict average speed will help in developing more accurate arrival prediction models especially through NEU-Fuzzy Artificial Neural Networks.

Finally, the dynamic TSP algorithm itself can be enhanced in many ways. For instance, a bigger database of TSP strategies and solutions can be used based on newly defined scenarios. In this thesis only three TSP solutions were tested. The use of more TSP solutions might provide more benefits to both transit and non-transit vehicles. In addition, the algorithm can include more flexible and effective offset recovery and signal compensation strategies. Signal recovery strategies could be applied over multiple signal phases, which would minimize the effects on other vehicles. Finally, a more advanced and complex algorithm can be developed for a network of intersections rather than a single intersection. By linking the signals into one algorithm, an optimal dynamic TSP system could be developed that aims to balance transit priority needs and non-transit vehicles delay and progression.

References

- California PATH Program (2004) Development of Adaptive Transit Signal Priority Systems, University of California at Berkeley, USA
- Chada, S. & Newland, R. (2002), Effectiveness of Bus Signal Priority, National Center For Transit Research, University of South Florida, USA
- Chen, M., Liu, X., Xia, J. & Chien, S. (2004) A Dynamic Bus-Arrival Time Prediction Model Based on APC Data, Computer-Aided Civil and Infrastructure Engineering, Vol.19, pp.364-376
- Ekeila, W. (2005), Vancouver's Streetcar Microsimulation Report, University of British Columbia & City of Vancouver, BC, Canada
- Farhan, A., Shalaby, A. & Sayed, T. (2003) Bus Travel Time Prediction Using AVL and APC, University of Toronto, Canada
- Intelligent Transportation Society of America (2002) An Overview of Transit Signal Priority, USA
- Lee, J., Shalaby, A., Greenough, J., Bowie, M. & Hung, S. (2005) Advanced Transit Signal Priority Control Using On-line Microsimulation-Based Transit Prediction Model, TRB 2005 Annual Meeting, [CD-ROM]
- Lin, G., Liang, P., Schonfeld, P. & Larson, R. (1995) Adaptive Control of Transit Operations, University of Maryland, USA
- Liu, H., Skabardonis, A. & Zhang, W. (2003) A Dynamic Model For Adaptive Bus Signal Priority, TRB 2003 Annual Meeting, [CD-ROM]

- Lomax, T., Turner, S., Hallenbeck, M., Boon, C. & Margiotta, R. (2001) Traffic Congestion and Travel Reliability, Federal Highway Administration, USA
- Maier, H., Sayed, T. & Lence, B. (2000) Forecasting Cyanobacterial Concentrations Using B-Spline Networks, Journal of Computing in Civil Engineering, July 2000
- Maybeck, P. (1979) Stochastic Models, Estimation, and Control, Vol. 1, Ch. 1, pp. 1-15, Academic Press, New York, USA
- NEU-Frame Manual (1997), Neural Computer Sciences, United Kingdom
- Roess, R., McShane, W. & Prassas, E. (1998) Basic Principles of Intersection Signalization, Traffic Engineering, Second Edition, Ch. 16, pp. 401-423, Prentice Hall, New Jersey, USA
- Sawalha, Z. (2002) Traffic Accident Modeling: Statistical issues and safety applications, University of British Columbia, BC, Canada
- Sayed, T., Navin, F. & Abdelwahab, W. (1997) A Countermeasure-Based Approach for Identifying and Treating Accident Prone Locations, Canadian Journal of Civil Engineering, Vol. 24, pp. 683-691
- Sayed, T. & Razavi, A. (2000) Comparison of Neural and Conventional Approaches to Mode Choice Analysis, Journal of Computing in Civil Engineering, January 2000
- Sayed, T. & Zein, S. (1998) Traffic Conflict Standards for Intersections, Transportation Planning and Technology, Vol. 22, pp. 309-323, BC, Canada

- Shalaby, A. & Farhan, A. (2004) Prediction Model of Bus Arrival and Departure Times Using AVL and APC Data, Journal of Public Transportation, Vol. 7, No. 1
- Victoria Transport Policy Institute (2005) Traffic Demand Management, Online TDM Encyclopedia, http://www.vtpi.org/tdm
- VISSIM 4.0 Manual (2004), PTV, Germany
- Wall, Z. & Dailey, D. (1999) An Algorithm For Predicting the Arrival Time of Mass Transit Vehicles Using Automatic Vehicle Location Data, TRB 1999 Annual Meeting, [CD-ROM]
- Wikipedia Online Encyclopedia (2005) Linear Regression, http://en.wikipedia.org/wiki/Linear_regression