2006-05

Final Report

A Nonlinear State Space Approach to Arterial Travel Time Prediction

Research

Technical Report Documentation Page

1. Report No.	2.	3. Recipients Accession No.
MN/RC-2006-05		

4. Title and Subtitle	5. Report Date
A Nonlinear State Space Approach to Arterial Travel Time Prediction	February 2006
	6.

7. Author(s)	8. Performing Organization Report No.
Jiann-Shiou Yang	

9. Performing Organization Name and Address	10. Project/Task/Work Unit No.
Department of Electrical and Computer Engineering University Of Minnesota 274 MWAH 10 University Dr Duluth, MN 55812	11. Contract (C) or Grant (G) No. (c) 81655 (wo) 130

12. Sponsoring Organization Name and Address	13. Type of Report and Period Covered
Minnesota Department of Transportation 395 John Ireland Boulevard Mail Stop 330 St. Paul, Minnesota 55155	Final Report 14. Sponsoring Agency Code

15. Supplementary Notes
http://www.lrrb.org/PDF/200605.pdf

16. Abstract (Limit: 200 words)

The study uses time series and the Kalman prediction techniques along with modern technology such as the Global Positioning System (GPS) for accurate data collection and analysis. A greater understanding of travel time will help facilitate traffic system performance monitoring, control, planning, and informed route decisions for motorists accessing information from changeable message sings (CMS). The models used for estimations include the autoregressive integrated moving average (ARIMA) and the autoregressive moving average (ARMA). The study collects travel data for the peak hours of travel (3:30-5:00 p.m.) over an eight-month period on the busiest section of Highway 194 in Duluth, Minnesota. The predictions were conducted over two weeks during the summer of 2005. Observed and predicted travel times are charted carefully and report evaluations determine the success of the study.

17. Document Analysis/Descriptors	18. Availability Statement
Travel time prediction, time series modeling, GPS test vehicle method	No restrictions. Document available from: National Technical Information Services, Springfield, Virginia 22161

19. Security Class (this report)	20. Security Class (this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	73	

A Nonlinear State Space Approach to Arterial Travel Time Prediction

Final Report

Prepared by:
Jiann-Shiou Yang, Ph.D.

Department of Electrical & Computer Engineering
University of Minnesota, Duluth

February 2006

Published by:
Minnesota Department of Transportation
Office of Research Services
Mail Stop 330
395 John Ireland Boulevard
St. Paul, MN 55155-1899

ACKNOWLEDGEMENTS

This project was conducted with funding provided from the Minnesota Department of Transportation (Mn/DOT). The author would like to thank graduate research assistant Archna Yadav and undergraduate research assistants Feng Qian, Jason Wollak and Daniel Billings, whose help was instrumental to the project development in this study.

TABLE OF CONTENTS

List of Tables

List of Figures

EXECUTIVE SUMMARY

The travel time variable is a good operational measure of the effectiveness of transportation systems. This information is needed to identify and assess operational problems along highway facilities. It is also necessary in traffic signal timing control coordination, incident detection, traffic assignment algorithms, and economic studies, etc. The ability to accurately predict freeway and arterial travel times in transportation networks is a critical component for many Intelligent Transportation Systems (ITS) applications.

This project studies the arterial travel time prediction (TTP) using the time series analysis and Kalman prediction techniques. Travel time is used as a performance measure because it is the most common way that drivers measure the quality of their trip and it is also a variable that can be directly measured. The availability of travel time data is essential for the development of "good" model(s) which can further affect the quality of TTP results. The data recorded, in general, can be considered and treated as a collection of observations made sequentially in time. Any quantity recorded over time yields a time series. Therefore, in the first part of this study, the time series models and analysis tools are first introduced. Particularly, we focus on the autoregressive integrated moving average (ARIMA) modeling because, in general, travel time data have non-stationary characteristics. The fundamental goal of using time series analysis is to understand the underlying mechanism that generates the observed data and, in turn, to forecast future values of the series.

Following the introduction of the time series models, we derive and formulate the state-space representation for the ARIMA model in terms of a properly defined state vector. The representation includes two equations (i.e., the state equation and the observation equation) which are further used in the Kalman recursive loop for TTP. To understand the basic principle of on-line estimation, the Kalman filter is briefly explained with its software implementation given. The combined state-space time series model and Kalman filtering are then used for arterial TTP purposes.

The test site chosen is a 3.7-mile State Highway 194 corridor (i.e., Central Entrance – Miller Trunk Highway) between Mesaba Avenue and Haines Road, one of the most heavily traveled and congested roadways in the Duluth area. The Global Positioning System (GPS) test vehicle technique, which involves collecting data with the aid of instrumented vehicles capable of receiving GPS signals for position and time information, is used to collect outbound peak hour (i.e., 3:30 pm -5 pm) traffic data. The vehicle tracking unit monitors a vehicle's location and travel time information. It utilizes the wireless data network to transmit data to the web server where the data (i.e., test vehicle ID, longitude, latitude, speed, direction, time stamp, date, etc.) can be accessed and the time stamp data is used to calculate section travel times. In calculating travel time, we use the Cohen-Sutherland line-clipping algorithm to check whether the test

vehicle entered an intersection.

By analyzing the section travel time data, the development of ARIMA modeling is then followed. The analysis includes data transformation and a study of the autocorrelation and partial autocorrelation function associated with the transformed data. The model order selection is based on the Akaike's and Bayesian information criteria while the model parameters are estimated via the Hannan-Rissanen (HR) algorithm. Following our model development, the model validation is further conducted via both the residual analysis and portmanteau lack-of-fit test. Furthermore, the state-space representations of the section models are also given.

Finally, based on the state-space models derived and real-time data measured, the Kalman recursion is used to conduct one-step-ahead TTP. In addition, we also investigate the correlation of adjacent road sections travel time data with the information properly weighted when predicting travel time. That is, the linear regression derived from data correlation is also incorporated with the model to improve our prediction results. This information is used only when the data correlation factor needs to be included. The performance evaluation includes the comparison of the observed and predicted values over different road sections on the corridor. We found that, in general, the ARIMA time series models produce reasonably good prediction results for most of the road sections studied. The predicted values are within the range of our observed travel times and they show very promising results.

Our study indicates the potential and effectiveness of using the time series modeling in the prediction of arterial travel time. Furthermore, the results presented here can be easily modified and used in short-term arterial TTP for other urban areas.

CHAPTER 1

INTRODUCTION

Travel time is the time required to traverse a route between any two points of interest and it is an important parameter that can be used to measure the effectiveness of transportation systems. This information is needed to identify and assess operational problems along highway facilities. It is also necessary in traffic signal timing control coordination, incident detection, traffic assignment algorithms, and economic studies, etc. In addition, the general public tends to think more in terms of travel time rather than volume in evaluating the quality of their trips. Therefore, travel times have always been of interest to traveler information researchers, planners, and public agencies as a key index in performance measure of traffic systems.

In recent years, many strategies based on advanced transportation technologies have been proposed to promote more efficient use of the existing roadway networks to ease congestion. Many of these systems require reliable prediction of travel times. Effective prediction of travel times is crucial to many advanced traveler information and transportation management systems. For instance, dynamic route guidance, in-vehicle information, congestion management and automatic incident detection systems can all benefit from accurate and implemental travel time prediction (TTP) techniques.

TTP refers to predicting and calculating the experienced travel time before a vehicle has traversed the arterial/freeway or route of interest. The ability to predict freeway and arterial travel times in transportation networks is a critical component for many Intelligent Transportation Systems (ITS) applications (e.g., Advanced Traffic Management Systems, In-vehicle Route Guidance Systems).

Travel time estimation and prediction has been an important research topic for decades. Many previous studies have been focused on predicting travel times on freeways using various methods, such as the time-series models [1], the Box-Jenkins techniques [2], the artificial neural network (ANN) models [3, 4], the parametric and non-parametric regression methods [5-8], the weighted moving average and cross correlation methods [9], the support vector machine method [10], etc. Traffic volume prediction was also reported using, for example, the adaptive filtering techniques [11], the seasonal autoregressive moving average and exponential smoothing method [12]. Some prediction models were developed using historic traffic data while others rely on real-time traffic information. Probe vehicles and geographic information system (GIS) technology were also reported to estimate the travel time (e.g., [13-15]).

Although research on travel time estimation for freeways is very rich, research on arterial travel time estimation is quite limited. Prediction of travel time is potentially more challenging for arterials than for freeways because vehicles traveling on arterials are subject not only to queuing delay but also to signal delays as well as delays caused by vehicles entering from the cross streets. Lin *et al.* formulated the arterial TTP problem into a Markov chain model but the nominal delay used in their model is based on the

existing delay formula for intersections [16]. It is known that many existing delay formulas perform poorly for over-saturated intersections. Sisiopiku and Rouphail focused on improving the performance by examining the use of detector output from simulation and field studies [17, 18]. However, their method is unable to predict travel time when the queues extend over the detector location. Artificial neural networks (ANNs) have also been used for prediction but they need a long time to learn the training data and the determination of the optimum architecture is simply a trial-and-error procedure. The Kalman filter algorithm was also used by Okutani and Stephanedes [19] to predict traffic volume in urban network. The advantage of using the Kalman filter is that it can update the parameter to make the predictor reflect the traffic fluctuation quickly. Compared with the Kalman filter algorithm, predicting travel time with ANNs may be less accurate if the future traffic patterns are not in the training samples.

Development of efficient methodologies for real-time measurement and estimation of travel time has been recognized as an important ITS component. It has also been identified by the Minnesota Department of Transportation (Mn/DOT) as one of the important issues for improving the safety and operational efficiencies of the traffic systems in the state of Minnesota.

This research focuses on the arterial TTP using time series analysis, modeling and the Kalman prediction techniques. The motivation of using this approach is because travel time data recorded can be considered as a collection of observations made sequentially in time and any quantity recorded over time yields a time series. Time series examples occur in a variety of fields ranging from economics to engineering and methods of analyzing time series constitute an important area of statistics. The fundamental goal of using time series analysis is to understand the underlying mechanism that generates the observed data and, in turn, to forecast future values of the series. This report is organized as follows:

In Chapter 2 we first introduce several important time series models and their properties. Particularly, we will focus on the autoregressive integrated moving average (ARIMA) modeling because, in general, travel time data (TTD) have non-stationary characteristics. We then derive the state-space representations for both the autoregressive moving average (ARMA) model and the ARIMA model. Since the state-space models will be used in the Kalman prediction, a summarized explanation of the Kalman filter together with its implementation are followed.

In Chapter 3, after describing the selected test site for TTP, we then explain the Global Positioning System (GPS) test vehicle technique used in our data collection. This method involves collecting data with the aid of instrumented vehicles capable of receiving GPS signals for position and time information. Real-time data access and travel time calculations will also briefly discussed in this chapter.

Chapter 4 starts the model development based on the analyzed data properties. The ARIMA time series model orders selection is based on the Akaike's and Bayesian information criteria, while the model parameters are estimated via the Hannan-Rissanen algorithm. The models developed are further validated via the residual analysis and

portmanteau lack-of-fit (PLOF) test. The state-space representation of these models are further given in this chapter.

Based on the results from the previous chapter, Chapter 5 will focus on TTP by presenting the performance evaluations on the test site, using the established models and the Kalman recursions to conduct one-step-ahead TTP. In addition, the effect of data correlation between adjacent road sections is also studied and included in our TTP via a proper weighting to improve the performance (i.e., a non-linear effect in the sense of mixing the weighted results). An experiment was conducted over a two-week period and the results are presented in this chapter. Our study indicates the potential and effectiveness of using the time series modeling and the Kalman filtering in the noisy environment to predict and estimate arterial travel times.

Finally, Chapter 6 gives the conclusion.

CHAPTER 2

TIME SERIES MODELING AND KALMAN RECURSIONS

The availability of TTD is essential for the establishment of good model(s) which can further affect the quality of TTP results. The data recorded, in general, can be considered and treated as a collection of observations made sequentially in time. Any quantity recorded over time yields a time series. A time series model for the observed data, say $\{x_t\}$, is a specification of the joint distributions of a sequence of random variables $\{X_t\}$ of which $\{x_t\}$ is postulated to be a realization. The term time series can mean both the data and the process of which it is a realization. The fundamental goal of time series analysis is to understand the underlying mechanism that generates the observed data and, in turn, forecast future values of the series. Time series analysis has been used in many areas such as economics, finance, etc. Time series analysis and Kalman filtering will be used in the development of travel time modeling and prediction.

2.1 Time Series Models

In this section, we briefly review several useful time series models. Time series modeling assumes that the value of the series X_t at time t depends only on its previous values and on a random noise. Therefore, if this dependence of X_t on its previous p values is linear, then X_t can be represented by

$$X_t = \Phi_1 X_{t-1} + \Phi_2 X_{t-2} + \ldots + \Phi_p X_{t-p} + Z_t \tag{2.1}$$

where $\Phi = (\Phi_1, \Phi_2, \ldots, \Phi_p)$ are the model parameters, called the autoregressive (AR) coefficients, and Z_t is the disturbance at time t. The process $\{Z_t\}$ is usually modeled as an independent and identically distributed (i.i.d.) white noise with zero mean and variance σ^2. That is, $E[Z_t] = 0$, $E[Z_t^2] = \sigma^2$ for all t, and $E[Z_t Z_s] = 0$ if $t \neq s$, where E[.] means the expectation. The process $\{X_t\}$ is said to be a moving average process of order q if X_t can be written as

$$X_t = Z_t + \theta_1 Z_{t-1} + \theta_2 Z_{t-2} + \ldots + \theta_q Z_{t-q} \tag{2.2}$$

where $\theta = (\theta_1, \theta_2, \ldots, \theta_q)$ are the moving average (MA) coefficients. In the above, p and q are the orders of AR(p) model and MA(q) model, respectively. By combining the AR and MV parts, we get a mixed autoregressive moving average (ARMA) process of order (p, q). That is,

$$X_t - \Phi_1 X_{t-1} - \Phi_2 X_{t-2} - \ldots - \Phi_p X_{t-p} = Z_t + \theta_1 Z_{t-1} + \theta_2 Z_{t-2} + \ldots + \theta_q Z_{t-q} \tag{2.3}$$

and this defines the ARMA(p, q) model. Note that any stationary ARMA process with a nonzero mean μ can be transformed into one with mean zero simply by subtracting the mean from the process. By introducing the back shift operator B, i.e., $B^i X_t = X_{t-i}$, then the ARMA (p, q) model can be written as

$$\Phi(B)\, X_t = \theta(B)\, Z_t \tag{2.4}$$

where $\Phi(B) = 1 - \Phi_1 B - \Phi_2 B^2 - \ldots - \Phi_p B^p$ and $\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \ldots - \theta_q B^q$.

Roughly speaking, a time series is stationary if the properties of one section of the data are much like those of any other section (e.g., no systematic changes in mean and variance, no strictly periodic variations, etc.). In practice, most time series we are facing are non-stationary. However, the stationary ARMA model can still be generalized to incorporate a special class of non-stationary time series models. For instance, if the observed time series is non-stationary, we can difference the series with X_t replaced by $(1-B)^d X_t$ where $(1-B)\, X_t = X_t - X_{t-1}$, $(1-B)^2\, X_t = (1-B)\, X_{t-1} = X_t - 2X_{t-1} + X_{t-2}$, etc. This operation is called differencing the time series. The ARMA model then becomes

$$(1-B)^d\, \Phi(B)\, X_t = \theta(B)\, Z_t \tag{2.5}$$

which is called the autoregressive integrated moving average (ARIMA) model and is expressed as ARIMA(p, d, q). In other words, any ARIMA(p, d, q) series can be transformed into an ARMA(p, q) series by differencing it d times and, thus, the analysis of an ARIMA process does not pose any difficulty as long as we know the number of times to difference the series. Clearly, the ARIMA process constitutes of three parts, an autoregressive part (AR), a differencing part (I), and a moving average part (MA). The differencing part is used to convert a non-stationary series into a stationary series. It removes the trend from the data. For details about introduction to time series models, please refer to any standard time series analysis books (e.g., [21-25]).

In time series analysis, it is very important to calculate the sample auto-covariance function (ACVF) and sample auto-correlation function (ACF) from the observed data of a given stationary process. The ACVF and ACF provide a useful measure of the degree of dependence among the values of a time series at different times and for this reason they play an important role when we consider the prediction of future values of the series in terms of past and present values. To find an appropriate model for the data observed we use the correlograms. A correlogram is a graph showing the time series ACF values against the lag h. From observing a correlogram sometimes we can get important information about the time series. For example, is the series stationary? If it is stationary, then is it AR(p), MA(q) or ARMA(p, q) type? What can be the order, i.e, the values of p and q for the series? It is known that for a series that fits MA (q) model, its correlogram should show a sharp cut-off after $h > q$, that is, the ACF becomes zero if $h > q$, a special feature of MA processes. If the correlogram doesn't cut-off sharply, and on the contrary, it decays either exponentially or sinusoidally or both, then it may suggest that the time series is either an AR (p) or ARMA(p, q) type. In this case the correlogram doesn't provide much information about the order of the series. So, we can pursue the partial correlogram (i.e., partial ACF vs. lag h) to see if any additional information can be extracted to find the proper order p. It can be shown that the partial ACF of an AR(p) process "cuts off" at lag p. Note that sample correlation functions do not always resemble the true correlation functions, in particular, when the number of data observed is small. Therefore, it should always be used with caution.

Another type of ARMA order selection is based on the so-called information criteria. The idea is to balance the risks of under fitting (i.e., selecting the orders smaller than the true orders) and over fitting (i.e., selecting orders larger than true orders). This is done by minimizing a penalty function, and the two commonly used functions are: $\ln \sigma^2+2(p+q)/n$ (i.e., the Akaike's Information Criterion (AIC)) and $\ln \sigma^2+(p+q) \ln (n)/n$ (i.e., the Bayesian Information Criterion (BIC)), where σ^2 is the estimated noise variance and n is the length of the data. For details regarding AIC and BIC criteria and order selection, please refer to [24, 25].

2.2 State Space Representations of ARMA and ARIMA Models

Although state-space representations are not unique, they can all be related via similarity transformation. In other words, the system characteristics remain unchanged under this transformation. In the following, we first present a state-space representation for the autoregressive process AR(p)

$$X_t = \Phi_1 X_{t-1} + \Phi_2 X_{t-2} + \ldots + \Phi_p X_{t-p} + Z_t \tag{2.6}$$

where t = 0, 1, 2, … and $\{Z_t\}$ is an i.i.d. zero-mean white noise with noise variance σ^2. The result for AR(p) will be served as a basis and it can then be easily extended to both ARMA and ARIMA processes. To express $\{X_t\}$ in state-space, let's introduce a p-dimensional state vector V_t as follows

$$V_t = \begin{bmatrix} X_{t-p+1} \\ X_{t-p+2} \\ \vdots \\ X_{t-1} \\ X_t \end{bmatrix}$$

then, it is easy to see that

$$V_{t+1} = \begin{bmatrix} X_{t-p+2} \\ X_{t-p+3} \\ \vdots \\ X_t \\ X_{t+1} \end{bmatrix}$$

Since $X_{t+1} = \Phi_1 X_t + \Phi_2 X_{t-1} + \ldots + \Phi_p X_{t-p+1} + Z_{t+1}$, the AR(p) model can be expressed as the following two equations

$$V_{t+1} = \begin{bmatrix} 0 & 1 & 0 & ..... & 0 \\ 0 & 0 & 1 & ..... & 0 \\ \vdots & \vdots & \vdots & ..... & \vdots \\ 0 & 0 & 0 & ..... & 1 \\ \phi_p & \phi_{p-1} & \phi_{p-2} & ..... & \phi_1 \end{bmatrix} V_t + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} Z_{t+1} \tag{2.7}$$

$$X_t = [0 \quad 0 \quad 0 \quad ... \quad 1] V_t \tag{2.8}$$

Note that Eq. (2.7) is called the state equation, while Eq. (2.8) is called the output (or observation) equation. Next, consider the ARMA(p, q) process defined by

$$X_t - \Phi_1 X_{t-1} - \Phi_2 X_{t-2} - \ldots - \Phi_p X_{t-p} = Z_t + \theta_1 Z_{t-1} + \theta_2 Z_{t-2} + \ldots + \theta_q Z_{t-q} \tag{2.9}$$

or $\Phi(B)\, X_t = \theta(B)\, Z_t$ where, as defined before, $\Phi(B) = 1 - \Phi_1 B - \Phi_2 B^2 - \ldots - \Phi_p B^p$ and $\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \ldots - \theta_q B^q$. Rewrite $X_t = \theta\,(B)\, [Z_t/\Phi(B)]$ and set $Z_t/\Phi(B) = U_t$, we have

$$\Phi(B) U_t = Z_t \quad \text{and} \quad X_t = \theta\,(B)\, U_t \tag{2.10}$$

Let r = max (p, q+1) then $\Phi_i = 0$ for i > p and $\theta_j = 0$ for j >q. Clearly, the 1st expression of Eq. (2.10) is simply an AR model with order r. Therefore, using the previous results (i.e., Eqs. (2.7), (2.8)) we immediately have the following expression of the state equation

$$V_{t+1} = \begin{bmatrix} 0 & 1 & 0 & ..... & 0 \\ 0 & 0 & 1 & ..... & 0 \\ \vdots & \vdots & \vdots & ..... & \vdots \\ 0 & 0 & 0 & ..... & 1 \\ \phi_r & \phi_{r-1} & \phi_{r-2} & ..... & \phi_1 \end{bmatrix} V_t + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} Z_{t+1} \tag{2.11}$$

where the state vector V_t is defined as

$$V_t = \begin{bmatrix} U_{t-r+1} \\ U_{t-r+2} \\ \vdots \\ U_{t-1} \\ U_t \end{bmatrix}$$

From the 2nd expression of Eq. (2.10), i.e., $X_t = \theta\,(B)\, U_t$, we form the observation equation

$$X_t = [\theta_{r-1} \quad \theta_{r-2} \quad \ldots \quad \theta_1 \quad 1]\, V_t \tag{2.12}$$

Thus, the state-space representation (i.e., Eqs. (2.11) and (2.12)) for the ARMA(p, q) model is obtained.

Since TTD in general is non-stationary, before modeling we need to transform the data so that it is "well-behaved". By this we mean that the transformed data can be modeled by a zero-mean, stationary ARMA type of process. For non-stationary data, a special way of filtering is simply to difference a given time series until it becomes stationary. Differencing is particularly useful for removing a trend. Because it is expected that TTD has non-stationary characteristics, it is reasonable to use the ARIMA modeling for TTP purposes. Based on the state-space representations for both AR(p) and ARMA(p, q) processes, let's find a state-space representation for the general ARIMA model described by

$$(1\text{-}B)^d\, \Phi(B)\, X_t = \theta(B)\, Z_t \tag{2.13}$$

Clearly, $\{X_t\}$ is an ARIMA process with $\{W_t = (1\text{-}B)^d X_t\}$ satisfying the ARMA model of $\Phi(B)\, W_t = \theta(B)\, Z_t$ That is, the time series $(1\text{-}B)^d X_t$ satisfies the ARMA(p, q) model. Following the similar approach, rewrite Eq. (2.13) as $(1\text{-}B)^d X_t = \theta(B)\, [Z_t/\Phi(B)] \equiv \theta(B)\, U_t$, and then we have

$$(1\text{-}B)^d X_t = \theta(B)\, U_t \qquad \text{and} \qquad \Phi(B) U_t = Z_t \tag{2.14}$$

Notice that the 2nd expression of Eq. (2.14) gives $U_t - \Phi_1 U_{t-1} - \Phi_2 U_{t-2} - \ldots - \Phi_p U_{t-p} = Z_t$, which further implies $U_{t+1} = \Phi_1 U_t + \Phi_2 U_{t-1} + \ldots + \Phi_p U_{t-p+1} + Z_{t+1}$. Again, let r = max (p, q+1) then $\Phi_i = 0$ for i > p and $\theta_j = 0$ for j >q. Now, implement the 2nd expression of Eq. (2.14) by letting

$$V_t = \begin{bmatrix} U_{t-r+1} \\ U_{t-r+2} \\ \vdots \\ U_{t-1} \\ U_t \end{bmatrix}$$

then we have

$$V_{t+1} = \begin{bmatrix} 0 & 1 & 0 & ..... & 0 \\ 0 & 0 & 1 & ..... & 0 \\ \vdots & \vdots & \vdots & ..... & \vdots \\ 0 & 0 & 0 & ..... & 1 \\ \phi_r & \phi_{r-1} & \phi_{r-2} & ..... & \phi_1 \end{bmatrix} V_t + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} Z_{t+1}$$

$$\square\ FV_t + W_t \tag{2.15}$$

The 1st expression of Eq. (2.14), $(1\text{-}B)^d X_t = \theta(B)\, U_t = U_t + \theta_1 U_{t-1} + \theta_2 U_{t-2} + \ldots + \theta_{r-1} U_{t-r+1}$, can be further written as

$$(1-B)^d X_t = [\theta_{r-1} \quad \theta_{r-2} \quad ... \quad \theta_1 \quad 1]\, V_t$$
$$\square\ GV_t \tag{2.16}$$

and we also know that

$$(1\text{-}B)^d X_t = X_t - (d,1)X_{t-1} + (d, 2)\, X_{t-2} - (d, 3)\, X_{t-3} + \ldots + (-1)^d\, (d, d)\, X_{t-d} \tag{2.17}$$

where the notation (d, j) means (d, j) = $\frac{d!}{j!(d-j)!}$ and d! = d(d-1)(d-2)…1. Define the state vector S_t as

$$S_t = \begin{bmatrix} Y_{t-d} \\ Y_{t-d+1} \\ \vdots \\ Y_{t-2} \\ Y_{t-1} \end{bmatrix}$$

then

$$S_{t+1} = \begin{bmatrix} 0 & 1 & 0 & ..... & 0 \\ 0 & 0 & 1 & ..... & 0 \\ \vdots & \vdots & \vdots & ..... & \vdots \\ 0 & 0 & 0 & ..... & 1 \\ (-1)^{d+1}(d,d) & (-1)^d(d,d-1) & (-1)^{d-1}(d,d-2) & ..... & d \end{bmatrix} S_t + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} (1-B)^d X_t$$

$$\square\ B\, S_t + A(1-B)^d X_t \tag{2.18}$$

Finally, combine Eqs. (2.15) and (2.18) by forming the state vector Γ_t defined as

$$\Gamma_t = \left| \frac{V_t}{S_t} \right| = \begin{bmatrix} U_{t-r+1} \\ U_{t-r+2} \\ \vdots \\ U_t \\ Y_{t-d} \\ \vdots \\ Y_{t-1} \end{bmatrix}$$

Then we have the state equation

$$\Gamma_{t+1} = \begin{pmatrix} F & 0 \\ AG & B \end{pmatrix} \Gamma_t + \left| \begin{matrix} W_t \\ 0 \end{matrix} \right| \tag{2.19}$$

Since

$$\begin{aligned} X_t &= (1\text{-}B)^d X_t + (d,1)X_{t-1} - (d,2) X_{t-2} + (d,3) X_{t-3} - \ldots + (-1)^{d+1} (d,d) X_{t-d} \\ &= GV_t + (d,1)X_{t-1} - (d,2) X_{t-2} + (d,3) X_{t-3} - \ldots + (-1)^{d+1} (d,d) X_{t-d} \end{aligned}$$

the observation equation becomes

$$X_t = \begin{bmatrix} G & (-1)^{d+1}(d,d) & (-1)^d(d,d-1) & (-1)^{d-1}(d,d-2) & \cdots & -(d,2) & (d,1) \end{bmatrix} \Gamma_t \tag{2.20}$$

Equations (2.19) and (2.20) are a state-space representation of the ARIMA(p, d, q) time series model. Note that if d = 1, then the matrices A and B reduce to 1. Based on the state-space representation derived above, the Kalman recursions technique will be used in the development of our TTP.

2.3 Kalman Filtering and Recursions

The Kalman filtering is a technique that can be used to recursively estimate state variables (e.g., travel time) from an observed time series $\{X_t\}$. This technique has been used extensively in many areas (e.g., navigation, guidance and control) with lots of practical applications reported in the literature (e.g., [26-28]). Its basic function is to provide estimates of the current state of the system. But it also serves as the basis for predicting future values of prescribed variables or for improving estimates of variables at earlier times. Since the introduction of the Kalman filter and

its initial applications in the early 1960's, many theoretical papers have been stimulated by problems encountered in applying the Kalman filter to practical problems. Due to the advances in digital computing, the Kalman filter has been the subject of extensive research and applications, particularly in the area of autonomous and assisted navigation. Recently, it has also been used in travel time estimation problems [29-31]. The Kalman filter basically is a set of mathematical equations that provides an efficient computational (recursive) means to estimate the state of a process, in a way that minimizes the mean of the squared error. The filter is very powerful in several aspects: it supports estimations of past, present, and even future states, and it can do so even when the precise nature of the modeled system is unknown. It tells how the past values of the input should be weighted in order to determine the present value of the output, that is, the optimal estimate. The two main features of the Kalman formulation and solution of the problem are (1) vector modeling of the processes under consideration, and (2) recursive processing of the noisy measurement (input) data. By providing a way of formulating the least squares filtering problem using state-space methods, the Kalman filter can solve many discrete-time, multi-input multi-output problems.

As can be seen from Section 2.2, many time series models (including the ARIMA model) can be cast into a state-space form with two equations (i.e., the state equation and the observation equation), in general, shown as

$$x_{k+1} = \phi_k x_k + w_k \tag{2.21}$$

where

$x_k = (n \times 1)$ state vector at time k

$\phi_k = (n \times n)$ state transition matrix relating x_k to x_{k+1} without a forcing function

$w_k = (n \times 1)$ vector representing a white noise sequence

The observation (measurement) of the process is

$$z_k = H_k x_k + v_k \tag{2.22}$$

where

$z_k = (m \times 1)$ measurement vector at time k

$H_k = (m \times n)$ matrix giving the connection between the measurement and the state vector at time k (under noise-free condition)

$v_k = (m \times 1)$ measurement error

The covariance matrices for the noise sequences $\{w_k\}$ and $\{v_k\}$are given by Q_k and R_k, respectively. And Q_k and R_k have the following properties: (a) $E[w_{ki}w_i^T] = Q_k$ for k = i and 0 for $k \neq i$; (b) $E[v_k v_j^T] = R_k$ for k =j and 0 for $k \neq j$, where the superscript T means the transpose and the symbol $E[\cdot]$ as mentioned in Section 2.1 represents the expected value. We assume that $\{v_k\}$ is a white sequence and is also uncorrelated with $\{w_k\}$, that is, $E[w_{ki}v_i^T] = 0$ for all k and i.

Assume that we have an initial estimate of the process at time k, and that this estimate is based on what is known about the process prior to k. This prior estimate will be denoted as $\hat{x}_k^-$ where "hat" denotes estimate and the superscript "-" is a reminder that this is the best estimate prior to assimilating the measurement at k. Now, define the estimation error to be

$$e_k^- = x_k - \hat{x}_k^- \tag{2.23}$$

and the associated error covariance matrix

$$P_k^- = E[e_k^- e_k^{-T}] = E[(x_k - \hat{x}_k^-)(x_k - \hat{x}_k^-)^T] \tag{2.24}$$

Then, a linear blending of the noisy measurement and the prior estimate can be chosen as

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H_k\hat{x}_k^-) \tag{2.25}$$

where $\hat{x}_k$ is the updated estimate and K_k is the blending factor. Note that the justification of the special form of Eq. (2.25) can be found in [27].

The optimal estimation problem is to find a particular K_k to minimize the performance criterion chosen to be the diagonal elements of the error covariance matrix P_k (to be given in Eq. (2.27)). Note that these diagonal terms represent the estimation error variances for the elements of x_k being estimated. There are several ways to solve this optimization problem (e.g., [26-28]). The optimal K_k, called the Kalman gain, is found to be

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \tag{2.26}$$

which minimizes the mean-square estimation error. In addition, the relationship between P_k and P_k^- can be further expressed as [27]

$$P_k = (I - K_k H_k) P_k^- \tag{2.27}$$

Since

$$\hat{x}_{k+1}^- = \phi_k \hat{x}_k \tag{2.28}$$

we can rewrite the expression for P_{k+1}^{-} as

$$P_{k+1}^{-} = \phi_k P_k \phi_k^T + Q_k \tag{2.29}$$

Note that Eq. (2.29) can be derived using Eqs. (2.23) and (2.24) with the index k replaced by k+1 together with Eqs. (2.21) and (2.28). With the results given above, the equations and the sequence of computational steps can now be summarized in Fig 2.1.

Our time series TTP, based on the minimization of the trace of P_k, can be summarized as follows:

Step 1: Initialization

Set $k = 0$ and let $E[x_0] = \hat{x}_0$ and $E[e_0^2] = P_0$

Step 2: Extrapolation

State estimate extrapolation: $\hat{x}_{k+1}^{-} = \phi_k \hat{x}_k$

Error covariance extrapolation: $P_{k+1}^{-} = \phi_k P_k \phi_k^T + Q_k$

Step 3: Kalman gain calculation

$$K_k = P_k^{-} H_k^T (H_k P_k^{-} H_k^T + R_k)^{-1}$$

Step 4: Update

State estimate update: $\hat{x}_k = \hat{x}_k^{-} + K_k (z_k - H_k \hat{x}_k^{-})$

Error covariance update: $P_k = (I - K_k H_k) P_k^{-}$

Step 5: Let $k = k + 1$ and go to Step 2 until the preset time period ends.

Note that Eqs. (2.19) and (2.20) are used in the formulation of Eqs. (2.21) and (2.22) with the state vector to be estimated is Γ_t.

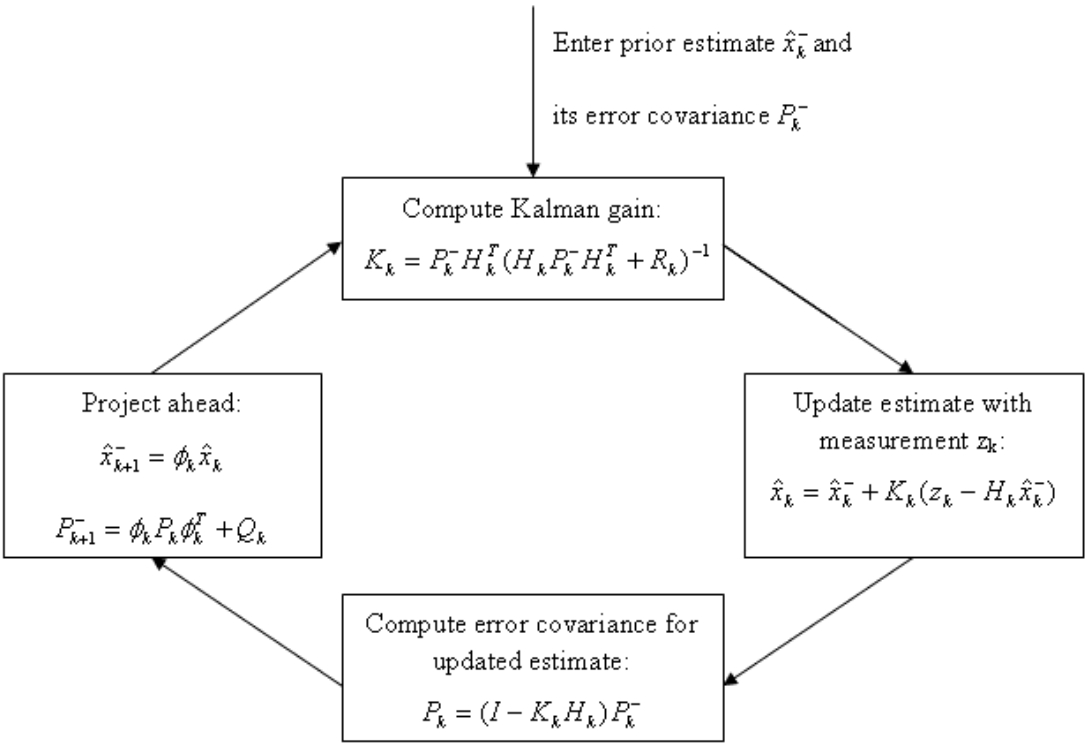

Fig. 2.1 The Kalman filter loop

CHAPTER 3
TRAVEL TIME DATA COLLECTION

Travel time is considered to be the total elapsed time of travel, including stops and delay, necessary for a vehicle to travel from one point to another over a specified route under existing traffic conditions. Travel time is an important measure of the performance and service quality of transportation systems. TTD are important for a variety of real-time and off-line transportation applications including traffic system performance monitoring, control, and planning. In addition, short-term travel time information through changeable message signs (CMS) is also useful for motorists to make their route choice decisions, to select different transportation modes, and to plan schedules. In this study, travel time is used as a parameter for performance measure due to the following: (a) it is the most common way that users measure the quality of their trip; (b) it is a variable that can be directly measured; (c) it is a simple measure to use for traffic monitoring. Currently, several methods (e.g., passive ITS probe vehicle method, license plate matching method, active test vehicle method) are available to measure TTD [20]. Since TTD collection with a Global Positioning System (GPS) unit has many advantages including reduction in staff requirements as compared to the manual method; reduction in human error; no vehicle calibration necessary as with the Distance Measuring Instrument (DMI) method; and relatively low operating cost after initial installation, etc., we use the active GPS test vehicle method to collect data (e.g., [20, 32, 33]). In this chapter, after describe the project study area information the GPS test vehicle technique will be briefly introduced and a description of the equipment we used will follow. The data collected is mainly used for our modeling purpose based on the techniques described in the previous chapter. The ultimate goal is to predict travel time with reasonable accuracy.

3.1 The Project Study Area

The test site we chose is the Miller Hill corridor on Minnesota State Highway 194 (Central Entrance - Miller Trunk Highway) between Mesaba Avenue and Haines Road, one of the most heavily traveled and congested roadways in the Duluth area. Miller Hill is a main traffic corridor and also a transit route in Duluth. This corridor is of great importance not only to Duluth's economy but to the region's as well. Therefore, the ability to better understand the traffic flow along this corridor will provide for better overall traffic management and better traveler information. The area map showing the test site is given in Fig. 3.1. Along this 3.7-mile corridor, there are eight road sections separated by signalized intersections. The section distance is given in Table 3.1. The outbound speed limit on this corridor is 30 miles per hour from Mesaba Avenue to Mall Entrance and then becomes 40 miles per hour from Mall Entrance to Haines Road. The inbound speed limit is 30 miles per hour for the entire road stretch. Figure 3.2 is a simplified area map showing the cross streets and intersections.

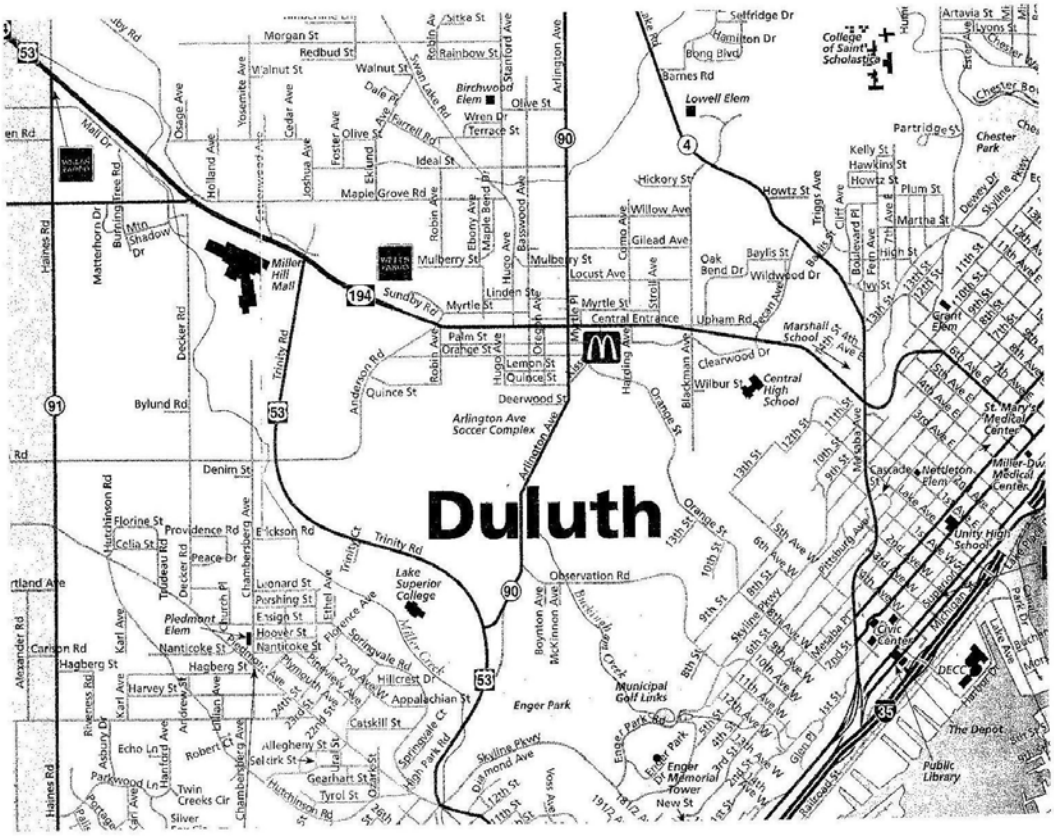

Fig. 3.1 The area map showing the project study area

Road Section	Distance (miles)
Mesaba Avenue – Pecan Avenue	0.5
Pecan Avenue – Arlington Avenue	0.8
Arlington Avenue – Basswood Avenue	0.1
Basswood Avenue – Anderson Road	0.4
Anderson Road – Mall Drive	0.3
Mall Drive – Trinity Road	0.3
Trinity Road – Maple Grove Road	0.5
Maple Grove Road – Haines Road	0.8
Total	3.7

Table 3.1 Section distance along the corridor

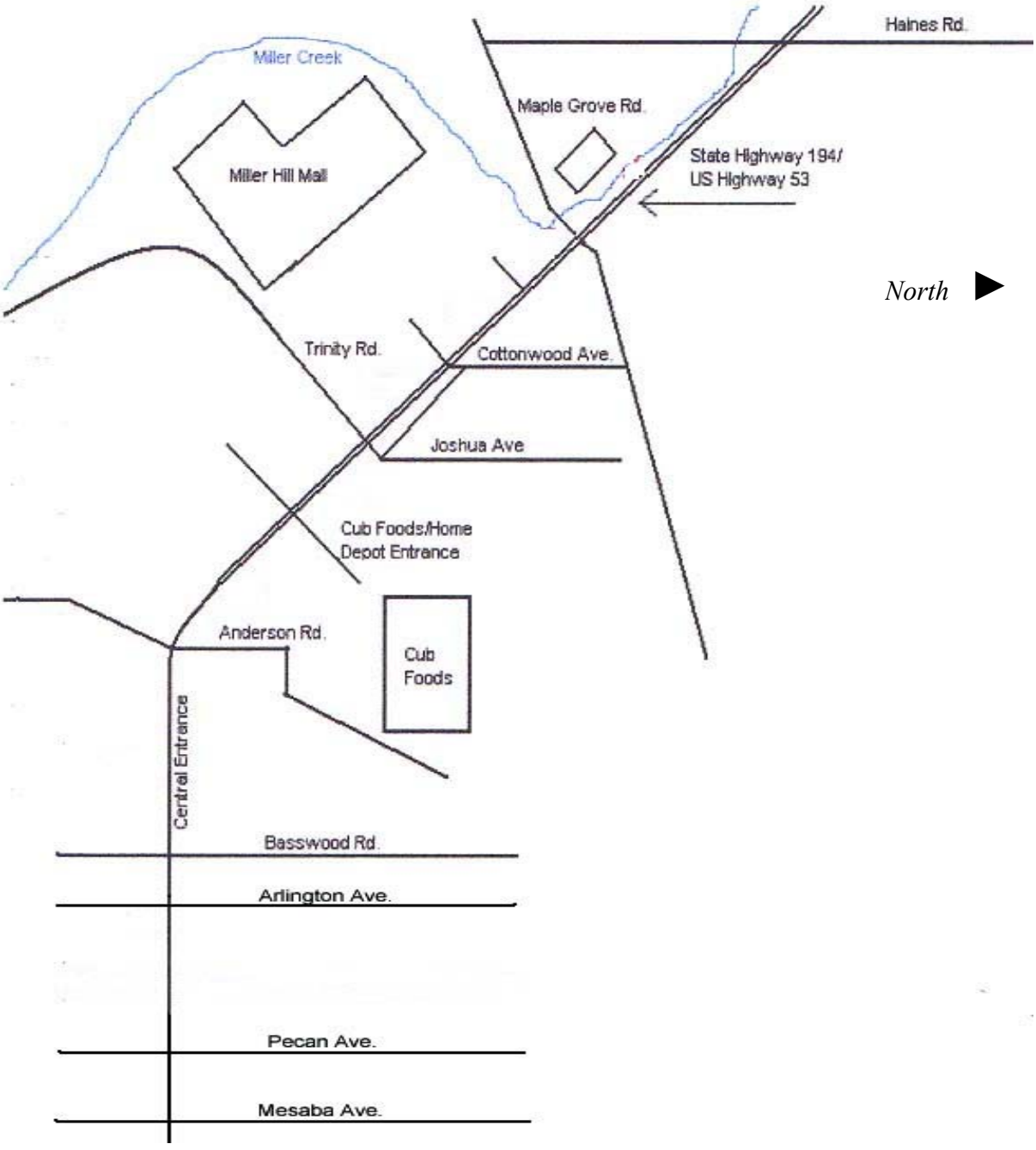

Fig. 3.2 A simplified area map showing the intersections

3.2 The GPS Test Vehicle Method

The test vehicle technique has been used for TTD collection since the late 1920s. Traditionally, this technique has involved the use of a data collection vehicle within which an observer records cumulative travel time at predefined checkpoints along a travel route. This information is then converted to travel time, speed, and delay for each segment along the survey route. There are several different methods for performing this type of data collection, depending on the instrumentation used in the vehicle and the driving instructions given to the driver. Since these vehicles are instrumented and then sent into the field for TTD collection, they are sometimes referred to as "active" test vehicles. Conversely, "passive" ITS probe vehicles are vehicles that are already in the traffic stream for purposes other than data collection [20]. Historically, the manual method has been the most commonly used TTD collection technique. This method requires a driver and a passenger to be in the test vehicle. The driver operates the test vehicle while the passenger records time information at predefined checkpoints.

GPS has become the most recent technology to be used for TTD collection (e.g., [32, 33]). Using this technique, a GPS device can track test vehicle by providing useful information.

The GPS was originally developed by the Department of Defense for the tracking of military ships, aircraft, and ground vehicles. Signals, sent from the 24 satellites orbiting the earth at 20,120 km (12,500 mi), can be used to monitor location, direction, and speed anywhere in the world. An excellent book detailing the basic principle and applications of GPS can be found in [34]. A consumer market has quickly developed for many civil, commercial, and research applications of GPS technology including recreational (e.g., backpacking, boating), maritime shipping, international air traffic management, and vehicle navigation. The location and navigation advantages of GPS have found many uses in the transportation profession. Figure 3.3 illustrates the equipment needs for GPS TTD collection [20]. The test vehicle is shown at the top of the figure with the GPS and Differential GPS (DGPS) antennas mounted on the roof of the vehicle. The DGPS antenna is connected to the differential correction receiver while the GPS antenna is connected to the GPS receiver. The differential correction data is then transferred to the GPS receiver. The GPS receiver uses the differential correction data to correct incoming signals and then the corrected information is output to the in-vehicle laptop. The laptop stores the data at predefined time intervals as the vehicle travels down the roadway. When the travel time run is completed, the laptop information is downloaded to a data storage computer.

In this project, we used the GPS test vehicle methods to collect TTD on the corridor. This technique involves collecting data with the aid of instrumented vehicles capable of receiving GPS signals for position and time information. We used the GPS L2000 vehicle tracking unit, manufactured by Global Tracking Communications (GTC), to collect TTD (http://www.gpstrackmasters.net). The data includes test vehicle ID, longitude, latitude, speed, direction, time stamp, date, etc. information. The time stamp data was used to calculate section travel times. The equipment used includes the GPS antenna, the tracking unit, and a 12-V battery as shown in Fig. 3.4. The tracking unit is easy to use and it is a real-time GPS tracking system that provides vehicle tracking in a "near real-time"

environment. It is based on GPS technology to monitor a vehicle's location and travel time information. However, unlike using an in-vehicle laptop as shown in Fig. 3.3, our vehicle tracking device utilizes the T-Mobile wireless data network to transmit data to the web server maintained by GTC. The data updates every 30 seconds.

The use of Global System for Mobile communications (GSM) and Global Positioning and General Packet Radio Service (GPRS) wireless networks uniquely enables the L2000 to provide real-time, data-based services beyond traditional location. GSM is a wireless network standard adopted throughout the world and GPRS is its data enhancement providing unprecedented transmission speeds and an "always on" connection. Figure 3.5 shows our test vehicle equipped with the L2000 vehicle tracking unit.

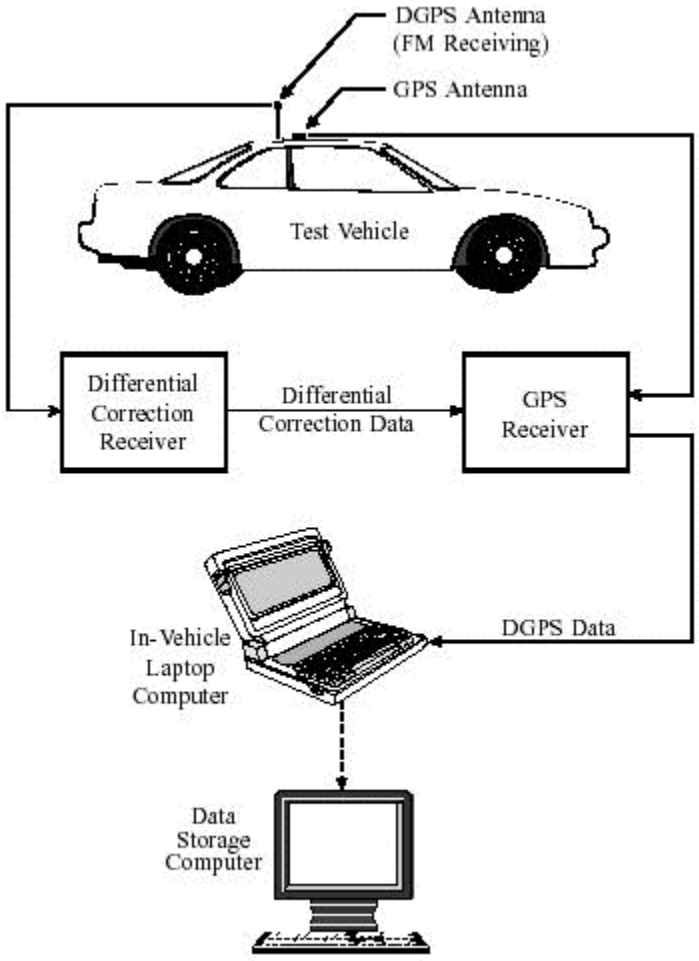

Fig. 3.3 Typical equipment setup for the GPS test vehicle method [20].

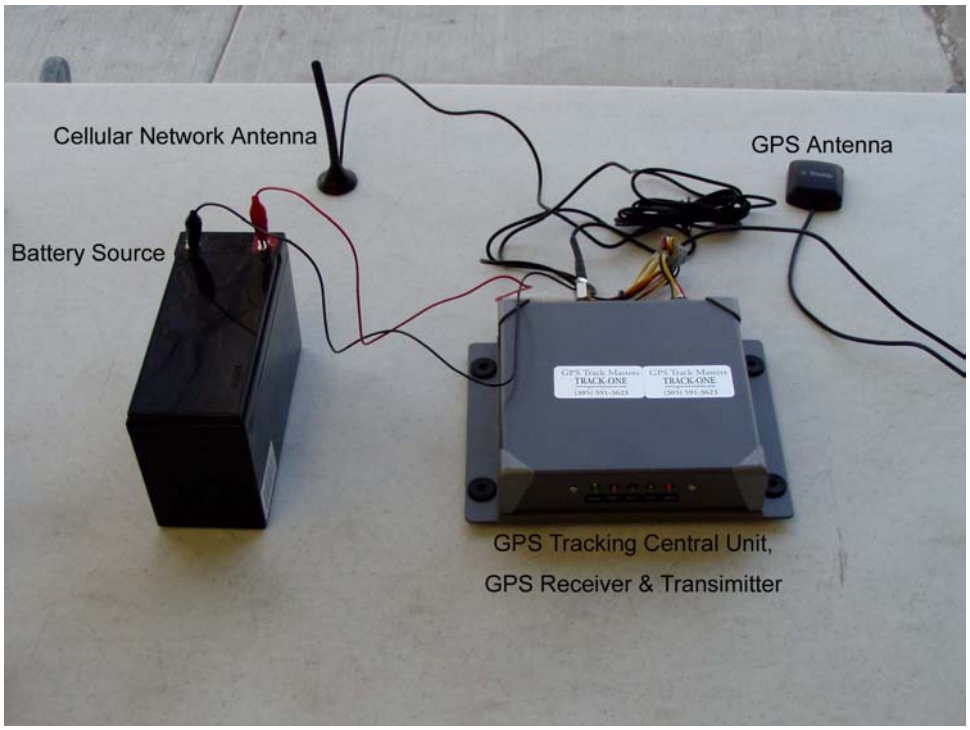

Fig. 3.4 The equipment used for data collection.

Fig. 3.5 The test vehicle equipped with the GPS L2000.

Using the signalized intersections as our checkpoints, the eight road sections shown in Fig. 3.2 are labeled In Table 3.2 and the mean values of the measured section travel times are shown in Fig. 3.6.

Section Number	Road Section
Section 1	Mesaba Avenue – Pecan Avenue
Section 2	Pecan Avenue – Arlington Avenue
Section 3	Arlington Avenue – Basswood Avenue
Section 4	Basswood Avenue – Anderson Road
Section 5	Anderson Road – Mall Entrance
Section 6	Mall Entrance – Trinity Road
Section 7	Trinity Road – Maple Grove Road
Section 8	Maple Grove Road – Haines Road

Table 3.2 Road sections along the corridor

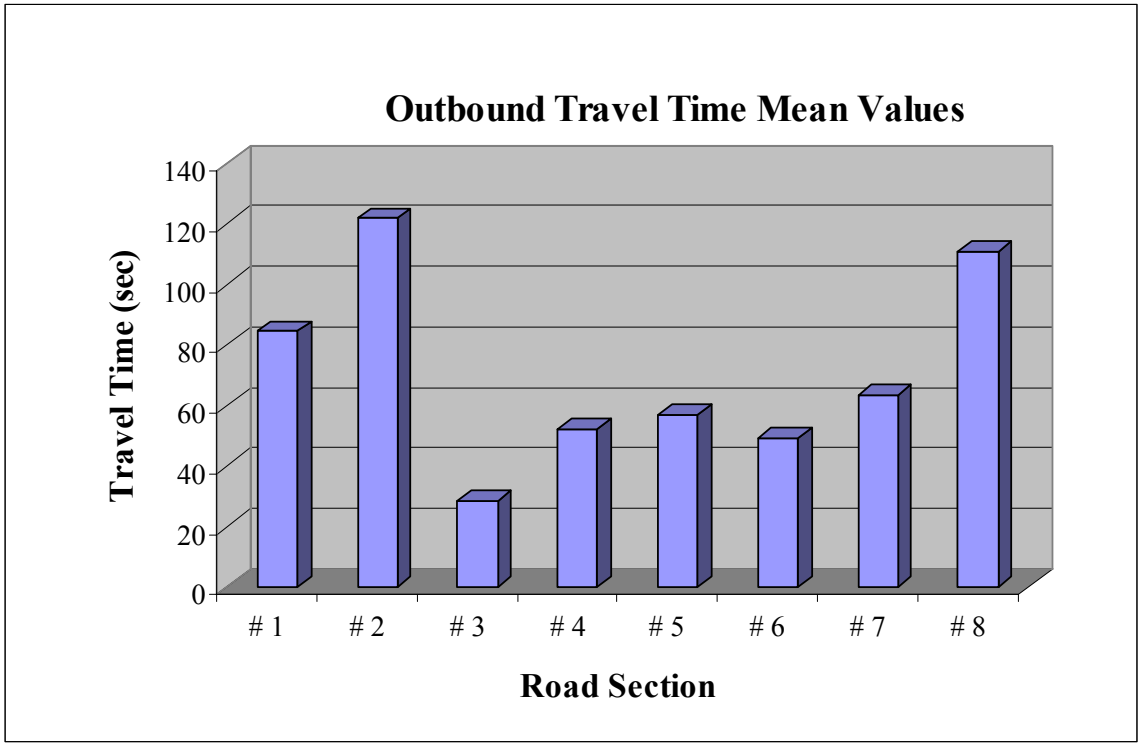

Fig.3.6 The mean values of section travel times

The GPS coordinates (i.e., latitude, longitude) at the nine intersections were measured and given in the following table.

Intersection (with HW194)	**Latitude**	**Longitude**
Mesaba Avenue	46.79578	-92.10715
Pecan Avenue	46.79944	-92.11161
Arlington Avenue	46.80040	-92.13150
Basswood Avenue	46.80048	-92.13489
Anderson Road	46.80056	-92.14289
Mall Entrance	46.80228	-92.14884
Trinity Road	46.80403	-92.15318
Maple Grove Road	46.80733	-92.16295
Haines Road	46.81466	-92.17450

Table 3.3 GPS coordinates of the intersections

The collected data can be accessed by logging on to the web server, maintained by GTC, at http://www.gpstrackmasters.net. The "Visual Address Log" and "Visual Coordinates Log" under "Reports" page can show us the vehicle ID, date, time, address, coordinates, speed, and heading, etc. information once the test vehicle completed the trip. A snapshot of the recorder data is shown in Fig. 3.7. We started the data collection on October 26, 2004 and the work continued until June 24, 2005. The accumulated section TTD was averaged and used as our historical time. Only outbound traffic was considered due to the afternoon rush hour (i.e., 3:30-5:00 pm). The modeling process was based on weekday rush hour data at the test site over this eight-month period.

For real-time data access, a communication link between the driver and the host machine is needed. Communication, synchronization, and reporting between the test vehicle and the ground station are necessary so that real-time data (longitude, latitude, speed, direction, time stamp, etc.) can be accessed and processed on line. The driver on the test vehicle needs to phone the person/operator working on the host machine when he passes the checkpoints. The operator then starts the process of getting data and extracting information from the web.

3.3 The Cohen-Sutherland Algorithm

To check whether the test vehicle enters an intersection we used the Cohen-Sutherland line-clipping algorithm [35, 36] with some modifications. We assume that the intersection is a square in space, which corresponds to the definition of window in the line-clipping algorithm. To understand the basic concept and operational principle of this algorithm, we define the following:

- Window: the space to which a line has to be clipped

- Half-space: each edge of the window divides the space into two parts, one is outside of the window and the other is inside (see Fig. 3.8)

- Out-codes: the set of binary digits that represents the position of a point with respect to window and defining the half-space that a point belongs to (see Fig. 3.9)

Visual Coordinates Log

From: 11/23/2004 1:58 AM
To: 11/23/2004 11:58 PM
Data types:

Unit: VEH001-
Id: 000945

Id	Date	Latitude / Longitude	Speed	Heading
0	11/23/2004 2:20:19 PM	46.80144/-92.15859	0	N
1	11/23/2004 2:20:21 PM	46.81497/-92.0845	15	W
2	11/23/2004 2:20:21 PM	46.81497/-92.0845	15	W
3	11/23/2004 2:20:41 PM	46.81577/-92.08569	18	NNE
4	11/23/2004 2:20:49 PM	46.81604/-92.08575	0	S
5	11/23/2004 2:21:08 PM	46.81604/-92.08579	0	N
6	11/23/2004 2:21:14 PM	46.81603/-92.08587	9	WNW
7	11/23/2004 2:21:14 PM	46.81603/-92.08587	9	WNW
8	11/23/2004 2:21:19 PM	46.81604/-92.08579	0	N

Fig. 3.7 A snapshot of the recorded data from the vendor's web server

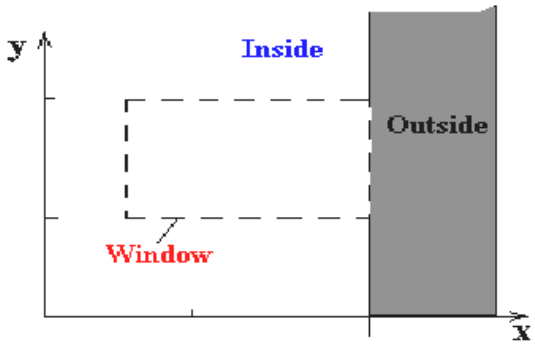

Fig. 3.8 Half-space created by an edge of the window

In Fig. 3.8, the four binary digits are defined as LRBT which stands for Left, Right, Bottom, Top. Each indicates whether a point is on the left, right, bottom and/or top of the window. For example, the out-code for a point 0110 means that a point is on the Right and Bottom of the window.

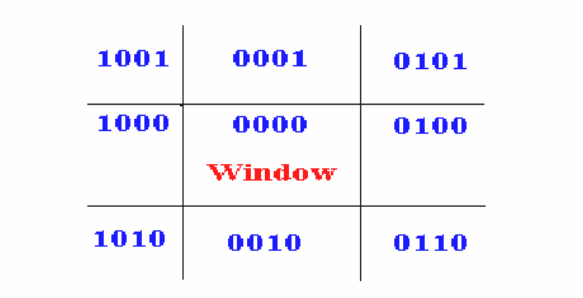

Fig. 3.9 Out codes for all the half spaces with respect to window

Based on the terms defined, the algorithm performs the following steps:

- Consider a window as the square defined surrounding the actual latitude and longitude of the intersection.
- Start with the first two consecutive points and determine their out-codes.
- If the out-code of any of the points is zero then the point is inside the space that we consider as intersection so we return that point as the intersection.
- If the bitwise AND two of the out-codes is zero, then ignore this pair as they are in the same half-space with respect to the window, and then consider the next set of consecutive points. Otherwise we assume, for simplicity, that the line connecting these two points crosses the intersection and then determine which of those points is closer to the window and return that point as the intersection.

Note that the actual Cohen-Sutherland algorithm determines the intersection of the line joining the two points with each of the windows to determine which part of that line to display. However, in our case this part is omitted because our travel space is limited to the road we travel on. We don't need to consider whether the line having zero out-code is really intersecting the window.

3.4 Real-Time Data Access

The GPS tracking unit reported its data to the web server and our task included accessing that data programmatically in real-time and detecting the intersection crossed by the test vehicle. We used a Perl module called WWW::Mechanize to do "website-scraping" [37, 38] to access real-time data. This technology simulates a browser using a program and the website responds to the program as if it is interacting with a human operator. The program is able to fill in text boxes, click buttons, choose from combo-boxes, set check-boxes and do all the things that an operator can do. The most difficult problem in this part was to simulate java-script in the program instead of the actual browser. We had to save the actual webpage and replace some of its code with our own code and then access the website through that modified code. The program first accesses the GPS TrackMasters' home page (i.e., http://www.gpstrackmasters.net) and sets the cookie. Then it clicks on the "Login" box and fills in the user name and password. After login is complete, it

navigates to the reports webpage by clicking the "Reports" button, the "Advanced Reports" button, and then selecting "Activity Report – LatLon" from the drop down box. The date entered as an argument is input to the "From" and "To" text boxes, and the fixed time is entered as the time. The program clicks on the "Generate" button, and then saves the file to the local machine. The program saves the resulting webpage, modifies its code, and accesses the live data by passing the HTML received. It keeps track of the last data seen and then accesses the website every 20 seconds as long as it keeps getting new data. If it does not receive data over a 5-minute time period then it automatically times out and exits. It also determines which intersection the vehicle is at while receiving data and calculates the time taken by the vehicle to reach that intersection.

3.5 Travel Time Calculations

In addition to automatic access to the web server while collecting data described above, several pieces of other computer programs were also developed. One of the programs extracts text from the webpage until it equals a predetermined value of the start of the data table. It then extracts the data of two nearest points at a time for comparison to an intersection. Once it locates a point before and after an intersection, it assigns the point located after the intersection as the closest point. It calculates the distance between the recorded GPS coordinates and the known intersection coordinates for error correction. The point for Haines Road is an exception because the vehicle never travels past the intersection. To assign a point to this intersection, the program calculates the distance between a given point and the intersection until the distance begins to increase, indicating the vehicle is no longer traveling towards the intersection. It assigns the first point headed away from the intersection as the point closest to the intersection. Furthermore, our computer program calculates the time difference between intersections using the time stamps from the saved file. The hour, minute, and second are extracted from the string and the total seconds from the time stamp is calculated and saved. The value of the total seconds is adjusted based on the distance of the point to the intersection and is saved in a separate variable. The time between intersections is then calculated by taking the difference of the total seconds at each intersection and the difference of adjusted total seconds of each intersection.

The formula used to adjust the time stamp of an intersection is based on (a) the distance in miles to the intersection, (b) the calculated GPS distance to the intersection, and (c) the average speed of the test vehicle. This correction is subtracted from the calculated total seconds.

The formula used for the correction is the ratio of the GPS error to the GPS "distance" between intersections multiplied by the known distance in miles, and then divided by the average speed of the vehicle. The data printed to the output shows the starting intersection, the uncorrected calculated time difference, the corrected time difference, and the range of values collected manually. Note that the data collected from the GPS L2000

vehicle tracking device is only accurate to the nearest 30 seconds, which is why the error correction is necessary and used. If the corrected calculated time is not within the range of the acceptable data, then the original uncorrected data is used instead. Note that our program listings are available upon request (Contact information at jyang@d.umn.edu).

CHAPTER 4

TRAVEL TIME DATA ANALYSIS AND MODELING

Based on the data collected in the previous chapter, we will study the travel time modeling issue via the time series analysis. The established models in the state-space form will then be incorporated with the Kalman filtering technique to predict section travel time. The model development based on the time series data analysis is first discussed. This includes determining the appropriate model type, model order selection, and model parameters estimation. Autocorrelation function of the transformed time series data is used to help determine the model type. The model order selection is based on the information criteria, while the Hannan-Rissanen (HR) algorithm [21] is used to calculate model parameter values. Finally, model validation is conducted via both the residual analysis and the PLOF test. The results derived in this chapter will be used in Chapter 5 for our TTP.

4.1 Model Development

To fit a time series model to data, we need to transform the raw data into a "well-behaved" form suitable for analyzing and modeling. In other words, the transformed data can be modeled by a zero-mean, stationary ARMA type of process described in Chapter 2. Therefore, the section TTD will first be converted to a zero-mean time series and shown in a time plot, i.e., a graph showing the observations against time. Any non-constant or variability should be removed before modeling. The time plot helps us determine if the process is stationary. If not, then the series is further processed to make it stationary. A special type of filtering, which is particularly useful for removing a trend, is simply to difference a given time series until it becomes stationary. Differencing is an effective way to remove trend and seasonal components in a time series. Note that the interpretation of the data and the model fitted are by no means unique; often there can be several equally valid interpretations consistent with the data. Experience and good judgment also play an important role in the modeling process. In this section, we will only focus on the Mesaba Avenue - Pecan Avenue section (i.e., road section # 1) and explain how a proper model is developed. This includes time series data analysis, modeling, and model validation via the residual analysis. The rest of the road sections can be similarly analyzed.

The time plot of the TTD for the Mesaba Avenue - Pecan Avenue section (i.e., road section # 1) is shown in Fig. 4.1. This plot shows the data collected over an eight-month period. The vertical axis labeled "number" represents the travel time recorded (in seconds), while the horizontal axis labeled "t" means the time duration of the 281 data points collected over time (see Table 4.2). Figure 4.2 shows the time plot with the mean value (i.e., 84.4 seconds) removed. We subtract this sample mean from the series to make it a zero-mean series. Since the time plot of the observed TTD after removing its mean value doesn't seem to be stationary, the data is differenced once as shown in Fig. 4.3.

Compared with the original plot the time plot in Fig. 4.3 looks more symmetric with respect to the vertical axis. The differenced data is then used for our modeling purpose.

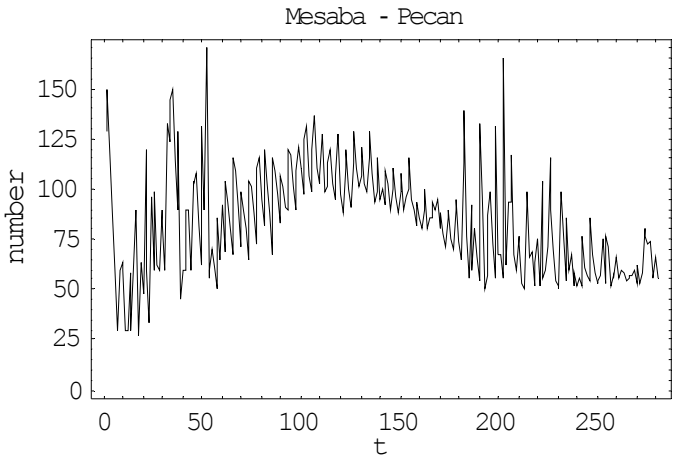

Fig. 4.1 Time plot of the Mesaba-Pecan section travel time data

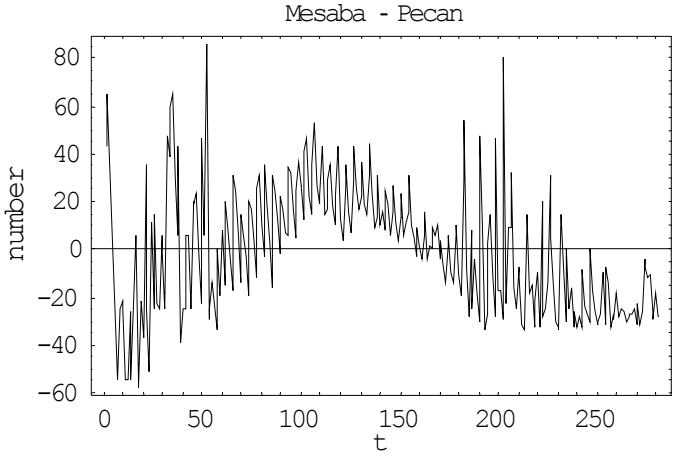

Fig. 4.2 Time plot of the section travel time data after removing its mean value

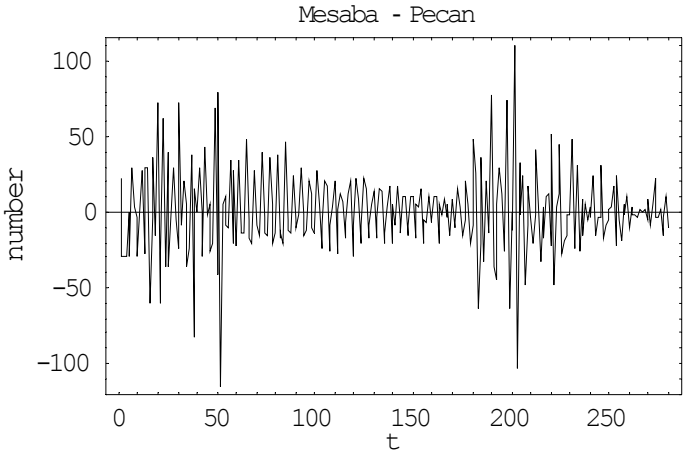

Fig. 4.3 The differenced time series data of road section # 1

The difference time series data was further analyzed by examining its autocorrelation function (ACF). ACF helps to describe the evolution of a process through time. A useful aid in interpreting a set of autocorrelation coefficients is a graph called a correlogram in which the sample autocorrelation coefficients γ_k are plotted against the lag k for k = 0, 1, …, M. Figures 4.4 and 4.5 show the correlogram and partial correlogram of the differenced data (i.e., the partial autocorrelation function (PACF) versus lag h). Note that in these two figures, the dotted lines, parallel to the x-axis, represent the error bounds for the data. The bounds are determined based on $\pm 2/\sqrt{n}$, where n represents the number of data points [21, 22]. If the value of the ACF and PACF lie within these lines, then we consider that they are not significantly different from zero. In other words, we can plot approximate 95% confidence limits at this value, and the observed ACF values which fall outside these limits are "significantly" different from zero at the 5% level. In Figs. 4.4 and 4.5, the correlograms show that ACF doesn't sharply cut-off to zero, which indicate that an AR (p) or ARMA (p, q) might be adequate.

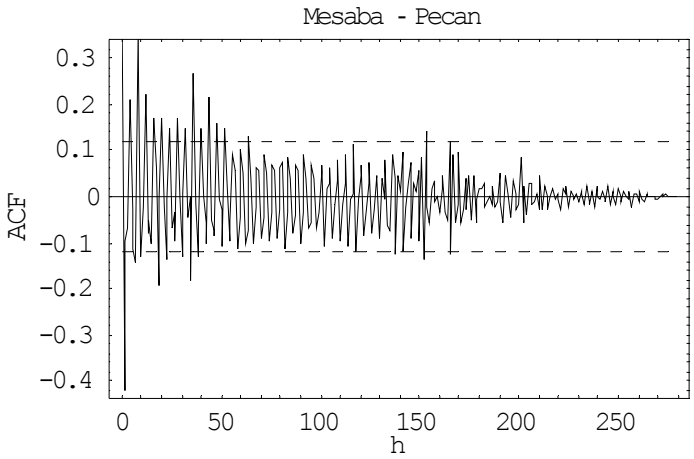

Fig. 4.4 The correlogram of the processed data

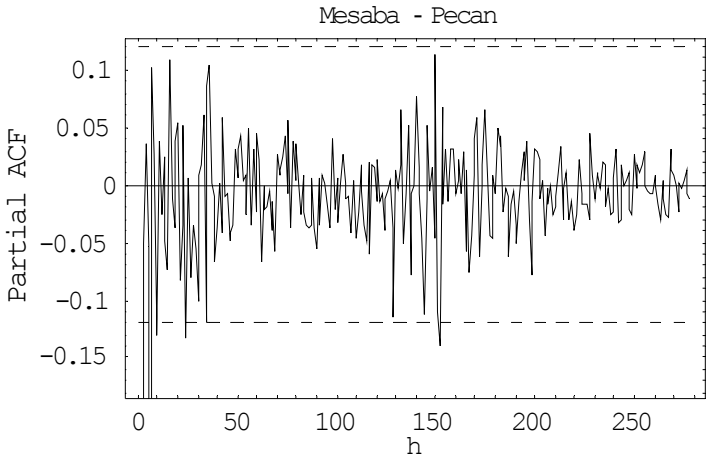

Fig. 4.5 The partial correlogram of the processed data

The algorithms used to determine the proper model order, i.e., the (p, q) values for model order selection and the model parameters estimation include: the Yule-Walker method, the Levison-Durbin algorithm, the Burg's algorithm, the Innovations algorithm, and the Hannan-Rissanen (HR) procedure [21, 24]. The Yule-Walker method and Levison-Durbin algorithm are mainly used for AR(p) model, while the Burg's algorithm works directly with the data rather than with the sample covariance function and it is asymptotically equivalent to the Yule-Walker estimates when the number of samples becomes large. The Innovations algorithm is used to obtain estimates of the MA(q) model and the HR procedure is mainly used for ARMA parameter estimates. For mixed models, i.e., those with $p > 0$ and $q > 0$, the HR algorithm is usually more successful in finding causal models. In our case, the HR method is more appropriate because our models are basically of the ARMA type. After extensive study, we found that the ARIMA (3, 1, 2) model with

$$\Phi = [-0.546239, -0.542852, -0.359531]$$

$$\theta = [-0.221893, 0.0839261]$$

seemed to generate the best results for the Mesaba-Pecan road section. In other words, the travel time model $\{Y_t\}$ for this road section can be expressed as the following time series

$$Y_t - Y_{t-1} = -0.546239\,(Y_{t-1} - Y_{t-2}) - 0.542852\,(Y_{t-2} - Y_{t-3}) - 0.359531\,(Y_{t-3} - Y_{t-4}) + Z_t - 0.221893\,Z_{t-1} + 0.0839261\,Z_{t-2}$$

The model established above was based on the results from the 3rd-order HR model parameters estimation (abbreviated as Hr3 in Table 4.3). That is, the information criteria were used to determine model orders while the HR procedure was applied to estimate the model parameter values.

4.2 Model Validation

After fitting a model to a given set of data the performance of the model needs to be examined to see if it is indeed an appropriate model. In this section, we chose two approaches, that is, the residual analysis and the portmanteau test, to verify the developed model. This approach was also used in validating other section models on the corridor.

4.2.1 Residual Analysis

If we have a "good" model, then we should expect the residuals to be "random" and "close to zero". There are several ways of checking if a model is satisfactory. One of the commonly used approaches to diagnostic checking is to examine the residues. That is, we can treat the residues as a time series and study its properties and the correlogram of the residues (i.e., the autocorrelation coefficients of the residues at different lag h). Therefore, we checked the residuals, which are the difference between the observation and fitted value, as shown in Fig. 4.6.

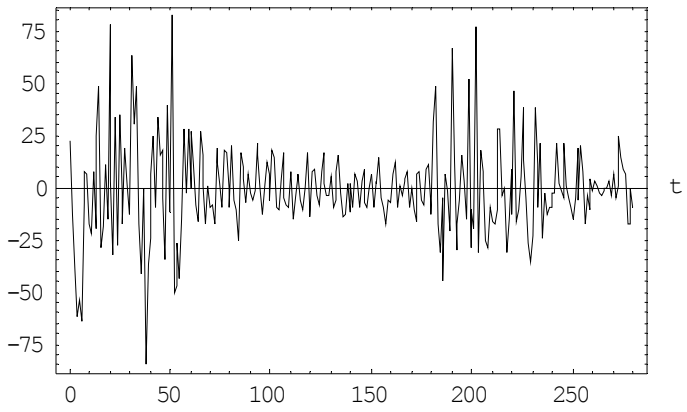

Fig. 4.6 The time plot of the residue time series.

For a good fit, the residual time series should be close to an independently, identically distributed (i.i.d.) zero-mean white noise. If the residual, say $\{y_1, y_2, \ldots, y_n\}$, is a realization of such an i.i.d. sequence, then about 95% of the sample autocorrelations should fall between the bound $\pm 2/\sqrt{n}$. A detailed analysis of residuals from ARMA processes can be found in [22]. To verify the models shown in the previous section, we conducted residual analysis. Figure 4.7 shows the correlogram of the residuals using the ARIMA (3, 1, 2) model for the Mesaba-Pecan road section. From Fig. 4.7, we see that residuals indeed lie within the bounds, indicating that the residual time series shows a zero mean white noise behavior. We see that ACF of the residue time series falls within the bounds except for a few k's and, thus, we have no reason to reject the hypothesis that the set of data constitutes a realization of a white noise process.

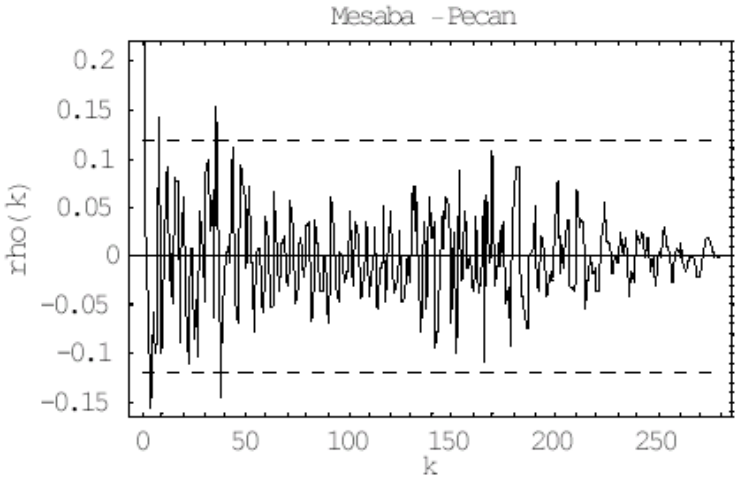

Fig. 4.7 The correlogram of the residual time series

4.2.2 Portmanteau Test

Instead of checking to see whether each sample autocorrelation coefficient falls within the bounds, it is also possible to carry out what is called a PLOF test. The portmanteau test uses a single value Q, by looking at the first k values of the residual correlogram all at once, to see if the fitted model is appropriate. The test statistic Q is defined as $Q = n\sum_{j=1}^{h} \rho^2(j)$, where n is the length of data (residues), ρ(j) is the sample autocorrelation with lag j, and h is typically chosen in the range of 15 to 30 (see [24]). Therefore, given the residual time series $\{y_1, y_2, \ldots, y_n\}$, we reject the i.i.d. hypothesis at level α if $Q > \chi^2_{1-\alpha}(h)$, where $\chi^2_{1-\alpha}(h)$ is the (1- α) quantile of the chi-squared distribution with h degrees of freedom. In other words, if the fitted model is appropriate, then Q should be approximately distributed as χ^2 with h degrees of freedom. We also conducted the portmanteau test in our study. For the ARIMA(3, 1, 2) model, we found that the corresponding portmanteau statistics Q is 430.949. Since this value Q is less than $\chi^2_{0.95}$ with the given h, we accept the fitted model as adequate.

Following the same approach, the rest of the seven road sections on the Miller Hill corridor were also studied using the time series analysis approach described above. The section travel time data (STTD) analysis for all eight sections is shown in the figures with the figure numbers summarized in the following table for easy reference purpose. Figures 4.8 - 4.56 are given at the end of this chapter (see Section 4.4 - Section Travel Time Data Analysis).

Figure	**Road Section**							
	1	2	3	4	5	6	7	8
Data # 1	4.1	4.8	4.15	4.22	4.29	4.36	4.43	4.50
Data # 2	4.2	4.9	4.16	4.23	4.30	4.37	4.44	4.51
Data # 3	4.3	4.10	4.17	4.24	4.31	4.38	4.45	4.52
Data # 4	4.4	4.11	4.18	4.25	4.32	4.39	4.46	4.53
Data # 5	4.5	4.12	4.19	4.26	4.33	4.40	4.47	4.54
Data # 6	4.6	4.13	4.20	4.27	4.34	4.41	4.48	4.55
Data # 7	4.7	4.14	4.21	4.28	4.35	4.42	4.49	4.56

Data # 1: Collected STTD

Data # 2: Zero mean STTD

Data # 3: Zero mean STTD after differencing

Data # 4: Correlogram of the differenced data

Data # 5: Partial correlogram of the differenced data

Data # 6: Residual time series data

Data # 7: Residual correlogram

Tables 4.1 - 4.3 summarize the models we suggested based on the time series data collected over an eight-month period together with the order of the HR algorithm used that generated the results. Note that the TTD collected show non-stationary property. Therefore, 1st-order differencing is used for all the road sections, that is, the ARIMA (p, 1, q) model.

Road Section/Distance	Model	AIC	BIC	Portmanteau
Mesaba Avenue – Pecan Avenue (0.5-mile)	ARIMA (3,1,2)	6.1017	6.16661	325.52
Pecan Avenue – Arlington Avenue (0.8-mile)	ARIMA(2,1,3)	6.60527	6.67018	393.691
Arlington Avenue – Basswood Avenue (0.1-mile)	ARIMA (1,1,5)	4.63175	4.71151	370.203
Basswood Avenue – Anderson Road (0.4-mile)	ARIMA (3,1,5)	4.38239	4.48624	351.305
Anderson Road – Mall Drive (0.3-mile)	ARIMA (2,1,3)	5.97521	6.04012	395.683
Mall Drive – Trinity Road (0.3-mile)	ARIMA (2,1,2)	4.51258	4.56491	304.645
Trinity Road – Maple Grove Road (0.5-mile)	ARIMA (6,1,3)	5.39922	5.51605	358.190
Maple Grove Road – Haines Road (0.8-mile)	ARIMA (2,1,1)	6.4233	6.46225	314.024

Table 4.1 Time series models and various performance measures

Road Section	No. of Data Points	Algorithm
Mesaba Avenue – Pecan Ave.	281	Hr3
Pecan Avenue – Arlington Ave.	281	Hr3
Arlington Avenue – Basswood Avenue	272	Hr4
Basswood Avenue – Anderson Road	281	Hr3
Anderson Road – Mall Drive	281	Hr4
Mall Drive – Trinity Road	278	Hr3
Trinity Road – Maple Grove Road	281	Hr3
Maple Grove Road – Haines Road	281	Hr4

Table 4.2 Number of data points and algorithm used for modeling

Road Section/Model	Model Parameters
Mesaba Avenue – Pecan Avenue ARIMA (3,1,2)	Φ = [- 0.546239, - 0.542852, - 0.359531 θ = [- 0.221893, 0.0839261] Q = 430.949
Pecan Avenue – Arlington Avenue ARIMA (2,1,3)	Φ = [0.0286419, - 0.814657] θ = [- 0.83327, 0.748828, - 0.577859] Q = 713.054
Arlington Avenue – Basswood Avenue ARIMA(1,1,5)	Φ = [- 0.584599] θ = [- 0.331988, - 0.60894, 0.0238475, - 0.0200404, 0.138766] Q = 98.2459
Basswood Avenue – Anderson Road ARIMA(3,1,5)	Φ = [- 0.357983, - 0.858853, - 0.459185] θ = [- 0.38219, 0.443095, - 0.302006, - 0.276388, -0.287582] Q = 75.5841
Anderson Road – Mall Drive ARIMA(2,1,3)	Φ = [- 0.937965, - 0.635611] θ = [0.144695, - 0.162582, - 0.537679] Q = 379.7431
Mall Drive – Trinity Road ARIMA(2,1,2)	Φ = [-0.178426, -0.217317] θ = [-0.515246, -0.169547] Q = 88.5613
Trinity Road. – Maple Grove Road ARIMA (6,1,3)	Φ = [-0.339246, -0.778076, -0.116948, -0.0513862, 0.0399999, -0.158566] θ = [-0.550511, 0.425886, -0.665323] Q = 207.459
Maple Grove Road – Haines Road ARIMA (2,1,1)	Φ = [0.119894, -0.120717] θ = [-0.7921] Q = 602.973

Table 4.3 ARIMA model parameter values

4.3 State Space Representations

State-space representations and the associated Kalman recursions have had a profound impact on time series analysis. As mentioned in Section 2.3, the Kalman recursions allow a unified approach to prediction and estimation for all processes that can be given a state-space representation. From Section 2.2, we see that the state space model for a time series $\{Y_t\}$ consists of two equations, the observation equation and the state equation. The observation equation can be expressed as

$$Y_t = H_t X_t + \nu_t$$

where ν_t is white noise with zero mean and variance R_t. The state equation determines the state X_{t+1} at time t+1 in terms of the previous state X_t and a noise term and it can be expressed as

$$X_{t+1} = \varphi_t X_t + w_t$$

where φ_t is the state transition matrices, w_t is a zero-mean white noise sequence with variance Q_t, and $(w_t\}$ and $\{\nu_t\}$ are uncorrelated. Based on this formulation and using the derivations in Section 2.2, the state-space representation of the ARIMA(3,1,2) model for the Mesaba-Pecan road section, i.e.,

$$\begin{aligned} Y_t - Y_{t-1} &= -0.546239\,(Y_{t-1} - Y_{t-2}) - 0.542852\,(Y_{t-2} - Y_{t-3}) \\ &\quad - 0.359531\,(Y_{t-3} - Y_{t-4}) + Z_t - 0.221893\,Z_{t-1} + 0.0839261\,Z_{t-2} \end{aligned}$$

can be further expressed as:

$$X_{t+1} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -0.359531 & -0.542852 & -0.546239 & 0 \\ 0.0839261 & -0.221893 & 1 & 1 \end{bmatrix} X_t + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} Z_{t+1}$$

$$Y_t = [0.0839261 \;\; -0.221893 \;\; 1 \;\; 1]\, X_t$$

$$X_t = \begin{bmatrix} U_{t-2} \\ U_{t-1} \\ U_t \\ Y_{t-1} \end{bmatrix}$$

Notice that in the above state-space representation the state vector X_t is of dimension 4 with the 4th state variable representing the travel time. The state space representations of the ARIMA models given in Table 4.3 for the rest of the sections can be similarly derived and they are given in the following pages.

Road Section # 2 (Pecan Avenue – Arlington Avenue)

Model: ARIMA (2, 1, 3)/Q = 713.054

$Y_t - Y_{t-1} = 0.0286419 (Y_{t-1} - Y_{t-2}) - 0.814657 (Y_{t-2} - Y_{t-3}) + Z_t - 0.83327Z_{t-1} + 0.748828 Z_{t-2} - 0.577859 Z_{t-3}$

State-space representation:

$$X_{t+1} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -0.814657 & 0.0286419 & 0 \\ -0.577859 & 0.748828 & -0.83327 & 1 & 1 \end{bmatrix} X_t + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} Z_{t+1}$$

$$Y_t = \begin{bmatrix} -0.577859 & 0.748828 & -0.83327 & 1 & 1 \end{bmatrix} X_t$$

$$X_t = \begin{bmatrix} U_{t-3} \\ U_{t-2} \\ U_{t-1} \\ U_t \\ Y_{t-1} \end{bmatrix}$$

Road Section # 3 (Arlington Avenue - Basswood Avenue)

Model: ARIMA (1, 1, 5)/Q = 98.2459

$Y_t - Y_{t-1} = - 0.584599 (Y_{t-1} - Y_{t-2}) + Z_t - 0.331988Z_{t-1} - 0.60894 Z_{t-2} + 0.0238475 Z_{t-3} - 0.0200404 Z_{t-4} + 0.138766 Z_{t-5}$

State-space representation:

$$X_{t+1} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.584599 & 0 \\ 0.138766 & -0.0200404 & 0.0238475 & -0.60894 & -0.331988 & 1 & 1 \end{bmatrix} X_t$$

$$+\begin{bmatrix}0\\0\\0\\0\\0\\1\\0\end{bmatrix} Z_{t+1}$$

$$Y_t = \begin{bmatrix}0.138766 & -0.0200404 & 0.0238475 & -0.60894 & -0.331988 & 1 & 1\end{bmatrix} X_t$$

$$X_t = \begin{bmatrix}U_{t-5}\\U_{t-4}\\U_{t-3}\\U_{t-2}\\U_{t-1}\\U_t\\Y_{t-1}\end{bmatrix}$$

Road Section # 4 (Basswood Avenue – Anderson Road)

Model: ARIMA (3, 1, 5)/Q = 75.5841

$$Y_t - Y_{t-1} = - 0.357983 (Y_{t-1} - Y_{t-2}) - 0.858853 (Y_{t-2} - Y_{t-3}) - 0.459185 (Y_{t-3} - Y_{t-4}) + Z_t$$
$$- 0.38219 Z_{t-1} + 0.443095 Z_{t-2} - 0.302006 Z_{t-3} - 0.276388 Z_{t-4} - 0.287582 Z_{t-5}$$

State-space representation:

$$X_{t+1} = \begin{bmatrix}0 & 1 & 0 & 0 & 0 & 0 & 0\\0 & 0 & 1 & 0 & 0 & 0 & 0\\0 & 0 & 0 & 1 & 0 & 0 & 0\\0 & 0 & 0 & 0 & 1 & 0 & 0\\0 & 0 & 0 & 0 & 0 & 1 & 0\\0 & 0 & 0 & -0.459185 & -0.858853 & -0.357983 & 0\\-0.287582 & -0.276388 & -0.302006 & 0.443095 & -0.38219 & 1 & 1\end{bmatrix} X_t$$

$$+\begin{bmatrix}0\\0\\0\\0\\0\\1\\0\end{bmatrix} Z_{t+1}$$

$$Y_t = \begin{bmatrix}-0.287582 & -0.276388 & -0.302006 & 0.443095 & -0.38219 & 1 & 1\end{bmatrix} X_t$$

$$X_t = \begin{bmatrix}U_{t-5}\\U_{t-4}\\U_{t-3}\\U_{t-2}\\U_{t-1}\\U_t\\Y_{t-1}\end{bmatrix}$$

Road Section # 5 (Anderson Road –Mall Drive)

Model: ARIMA (2, 1, 3)/Q = 379.7431

$Y_t - Y_{t-1} = - 0.937965 (Y_{t-1} - Y_{t-2}) - 0.635611 (Y_{t-2} - Y_{t-3}) + Z_t + 0.144695 Z_{t-1}$
$- 0.162582 Z_{t-2} - 0.537679 Z_{t-3}$

State-space representation:

$$X_{t+1} = \begin{bmatrix}0 & 1 & 0 & 0 & 0\\0 & 0 & 1 & 0 & 0\\0 & 0 & 0 & 1 & 0\\0 & 0 & -0.635611 & -0.937965 & 0\\-0.537679 & -0.162582 & 0.144695 & 1 & 1\end{bmatrix} X_t + \begin{bmatrix}0\\0\\0\\1\\0\end{bmatrix} Z_{t+1}$$

$$Y_t = \begin{bmatrix}-0.537679 & -0.162582 & 0.144695 & 1 & 1\end{bmatrix} X_t$$

$$X_t = \begin{bmatrix} U_{t-2} \\ U_{t-1} \\ U_t \\ Y_{t-1} \\ Y_{t-2} \end{bmatrix}$$

Road Section # 6 (Mall Drive – Trinity Road)

Model: ARIMA (2, 1, 2)/Q = 88.5613

$Y_t - Y_{t-1} = - 0.178426 (Y_{t-1} - Y_{t-2}) - 0.217317 (Y_{t-2} - Y_{t-3}) + Z_t - 0.515246 Z_{t-1} - 0.169547 Z_{t-2}$

State-space representation:

$$X_{t+1} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -0.217317 & -0.178426 & 0 \\ -0.169547 & -0.515246 & 1 & 1 \end{bmatrix} X_t + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} Z_{t+1}$$

$$Y_t = \begin{bmatrix} -0.169547 & -0.515246 & 1 & 1 \end{bmatrix} X_t$$

$$X_t = \begin{bmatrix} U_{t-2} \\ U_{t-1} \\ U_t \\ Y_{t-1} \end{bmatrix}$$

Road Section # 7 (Trinity Road – Maple Grove Road)

Model: ARIMA (6, 1, 3)/Q = 207.459

$Y_t - Y_{t-1} = - 0.339246 (Y_{t-1} - Y_{t-2}) - 0.778076 (Y_{t-2} - Y_{t-3}) - 0.116948 (Y_{t-3} - Y_{t-4})$
$- 0.0513862 (Y_{t-4} - Y_{t-5}) + 0.0399999 (Y_{t-6} - Y_{t-7}) + Z_t - 0.550511 Z_{t-1}$
$+ 0.425886 Z_{t-2} - 0.665323 Z_{t-3}$

State-space representation:

$$X_{t+1} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ -0.158566 & 0.0399999 & -0.0513862 & -0.116948 & -0.778076 & -0.339246 & 0 \\ 0 & 0 & -0.665323 & 0.425886 & -0.550511 & 1 & 1 \end{bmatrix} X_t + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} Z_{t+1}$$

$$Y_t = \begin{bmatrix} 0 & 0 & -0.665323 & 0.425886 & -0.550511 & 1 & 1 \end{bmatrix} X_t$$

$$X_t = \begin{bmatrix} U_{t-5} \\ U_{t-4} \\ U_{t-3} \\ U_{t-2} \\ U_{t-1} \\ U_t \\ Y_{t-1} \end{bmatrix}$$

Road Section # 8 (Maple Grove Road – Haines Road)

Model: ARIMA (2, 1, 1)/Q = 602.973

$Y_t - Y_{t-1} = 0.119894 (Y_{t-1} - Y_{t-2}) - 0.120717 (Y_{t-2} - Y_{t-3}) + Z_t - 0.7921 Z_{t-1}$

State-space representation:

$$X_{t+1} = \begin{bmatrix} 0 & 1 & 0 \\ -0.120717 & 0.119894 & 0 \\ -0.7921 & 1 & 1 \end{bmatrix} X_t + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} Z_{t+1}$$

$$Y_t = \begin{bmatrix} -0.7921 & 1 & 1 \end{bmatrix} X_t$$

$$X_t = \begin{bmatrix} U_{t-1} \\ U_t \\ Y_{t-1} \end{bmatrix}$$

4.4 Section Travel Time Data Analysis

The STTD for the rest of the road sections was also analyzed. The procedure includes data transformation and processing and they are shown as follows.

● Pecan Avenue – Arlington Avenue

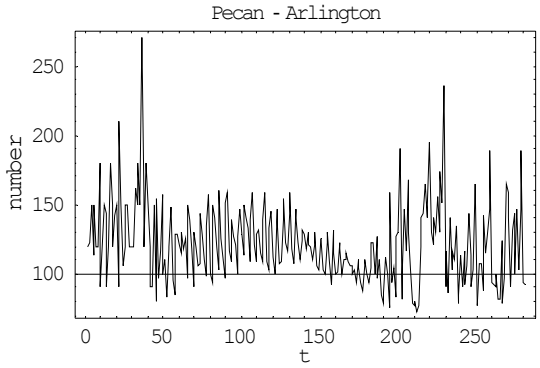

Fig. 4.8 Time plot of the Pecan-Arlington section travel time data

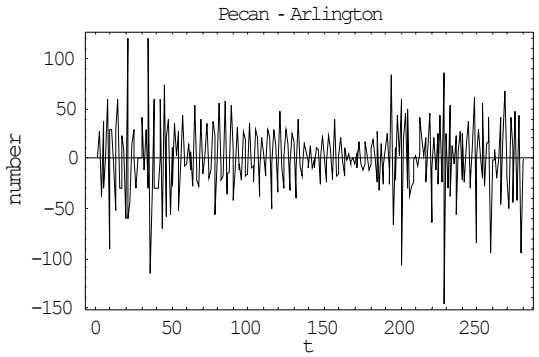

Fig. 4.10 The differenced time series data of road section # 2

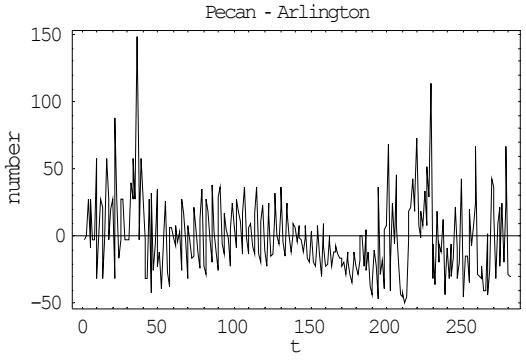

Fig. 4.9 Time plot of the section travel time data after removing its mean value

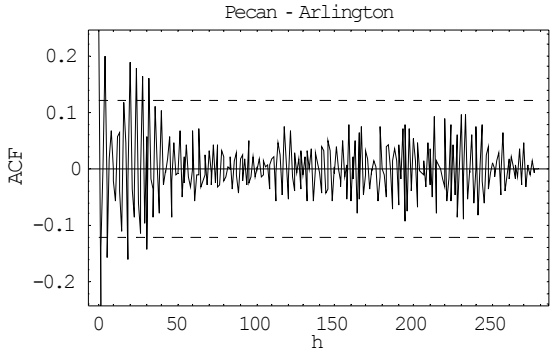

Fig. 4.11 The correlogram of the processed data

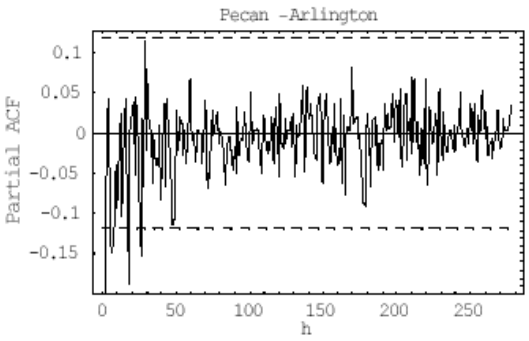

Fig. 4.12 The partial correlogram of the processed data

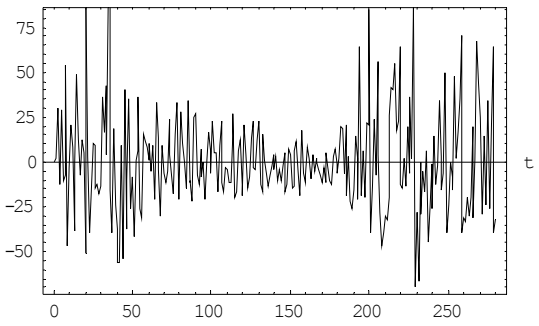

Fig. 4.13 The time plot of the residual time series

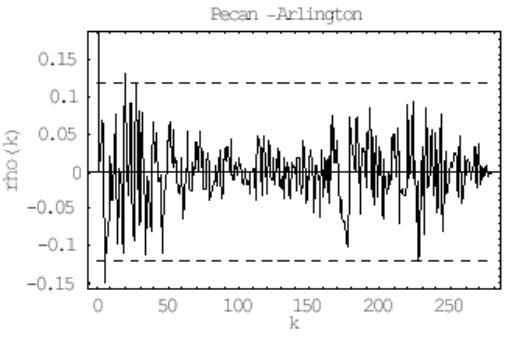

Fig. 4.14 The correlogram of the residual time series

● Arlington Avenue – Basswood Road

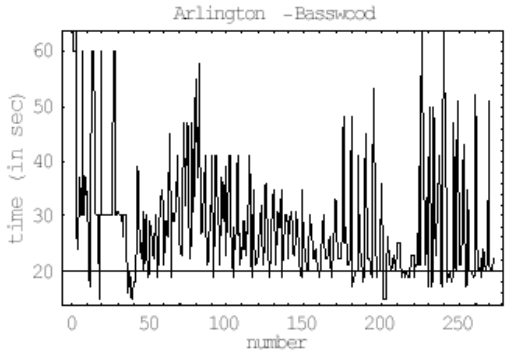

Fig. 4.15 Time plot of the Arlington-Basswood section travel time data

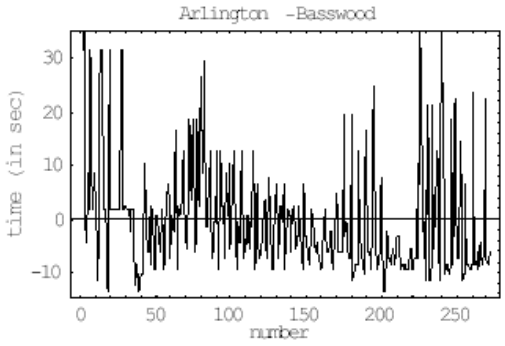

Fig. 4.16 Time plot of the travel time data after removing its mean value

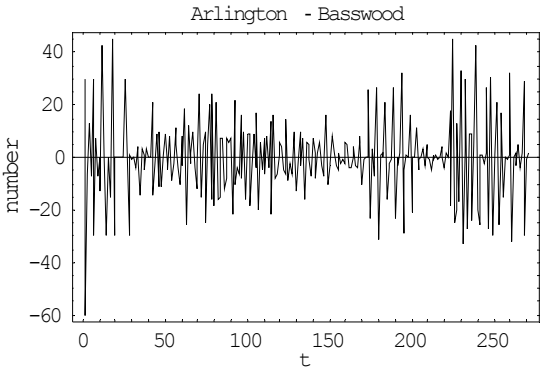

Fig. 4.17 The differenced time series data of road section # 3

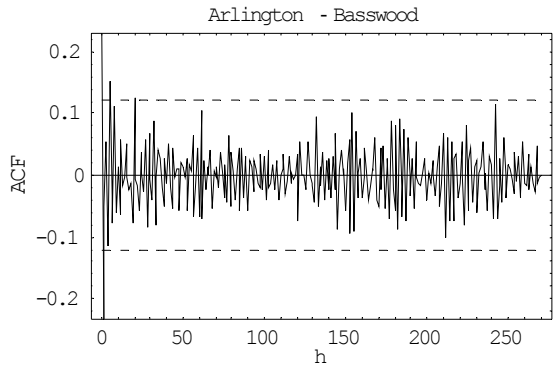

Fig. 4.18 The correlogram of the processed data

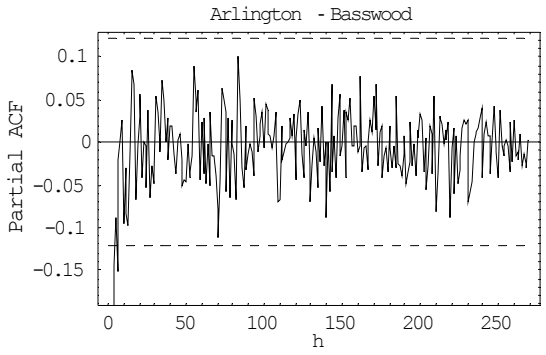

Fig. 4.19 The partial correlogram of the processed data

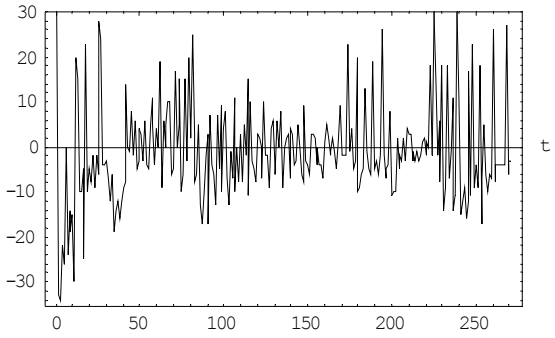

Fig. 4.20 The time plot of the residual time series

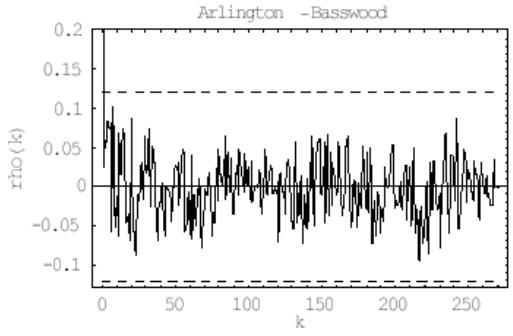

Fig. 4.21 The correlogram of the residual time series

● Basswood Road – Anderson Road

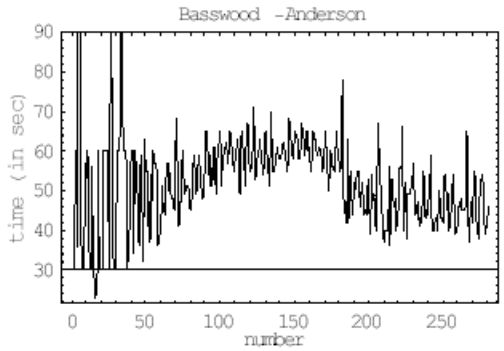

Fig. 4.22 Time plot of the Basswood-Anderson section travel time data

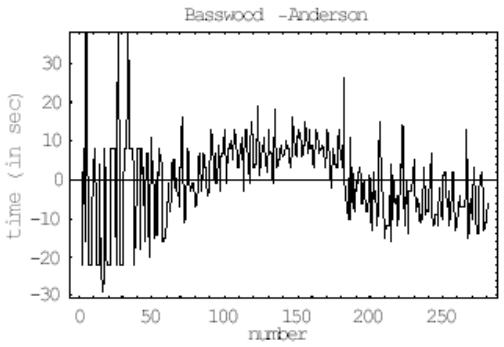

Fig. 4.23 Time plot of the section travel time data after removing its mean value

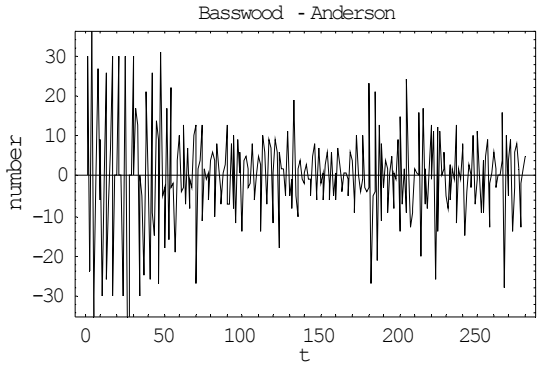

Fig. 4.24 The differenced time series data of road section # 4

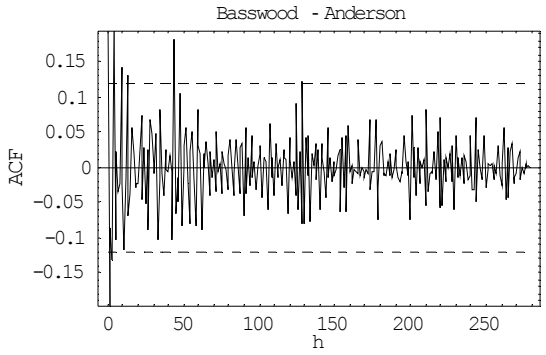

Fig. 4.25 The correlogram of the processed data

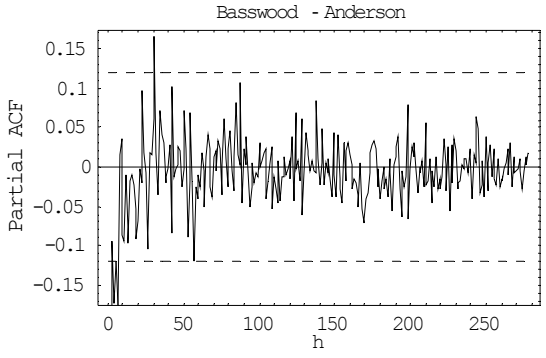

Fig. 4.26 The partial correlogram of the processed data

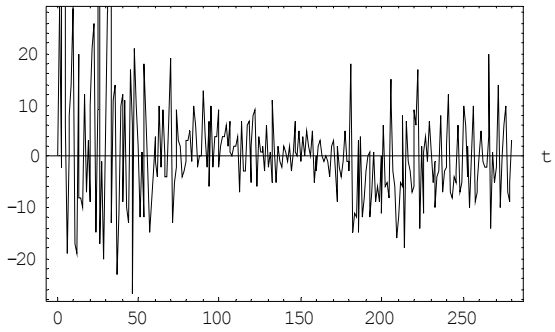

Fig. 4.27 The time plot of the residual time series

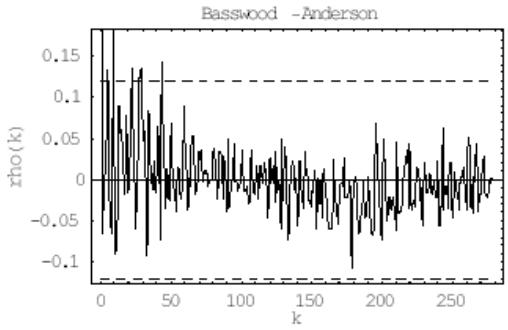

Fig. 4.28 The correlogram of the residual time series

● Anderson Road – Mall Drive

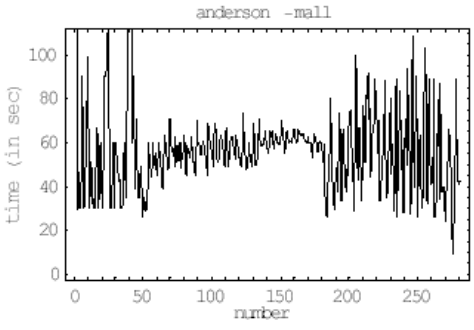

Fig. 4.29 Time plot of the Anderson-Mall section travel time data

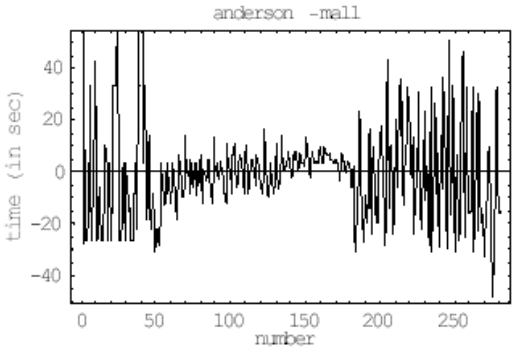

Fig. 4.30 Time plot of the section travel time data after removing its mean value

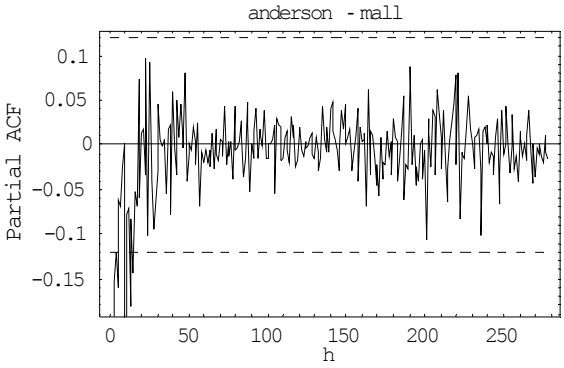

Fig. 4.33 The partial correlogram of the processed data

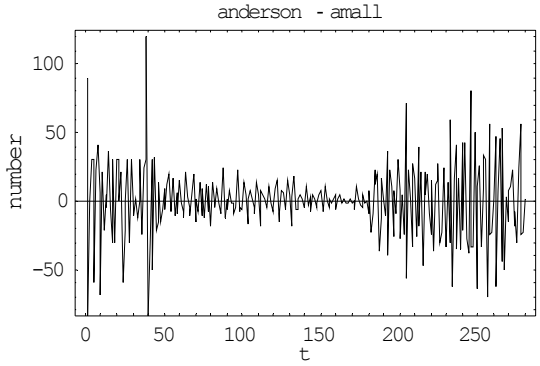

Fig. 4.31 The differenced time series data of road section # 5

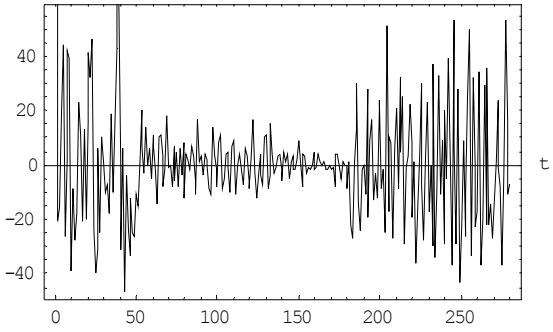

Fig. 4.34 The time plot of the residual time series

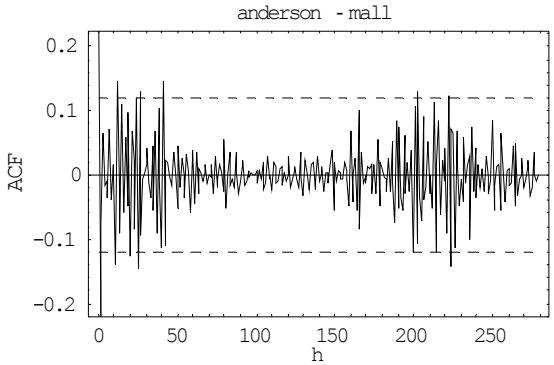

Fig. 4.32 The correlogram of the processed data

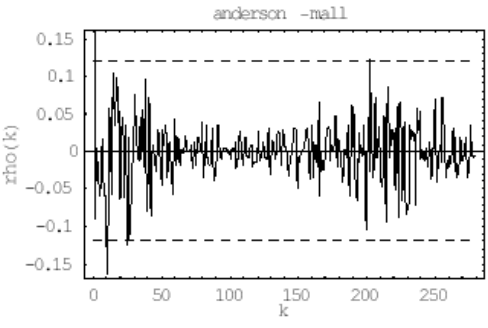

Fig. 4.35 The correlogram of the residual time series

● Mall Drive – Trinity Road

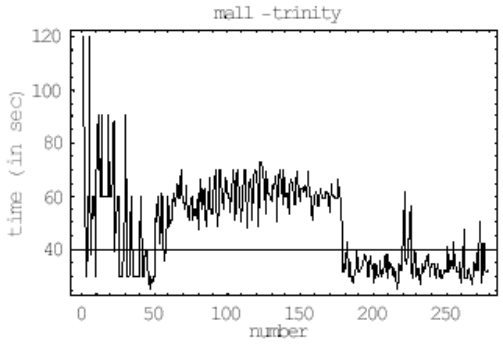

Fig. 4.36 Time plot of the Mall-Trinity section travel time data

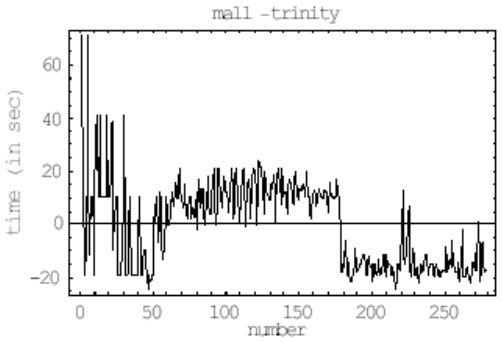

Fig. 4.37 Time plot of the section travel time data after removing its mean value

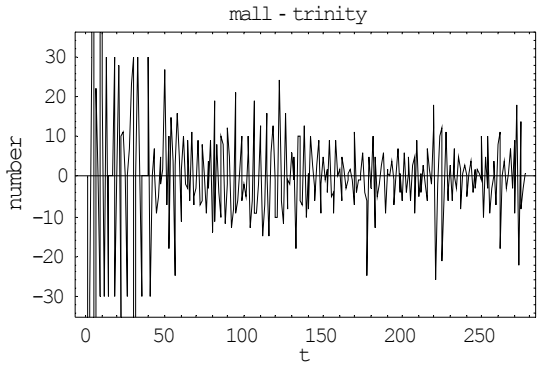

Fig. 4.38 The differenced time series data of road section # 6

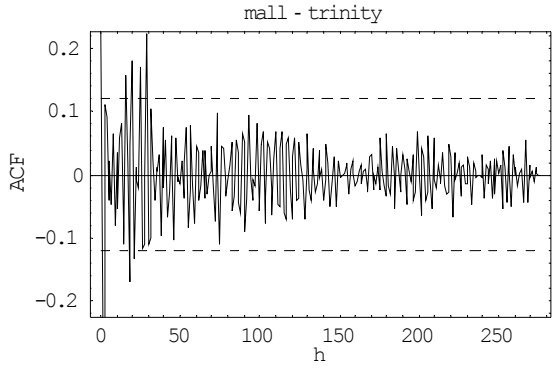

Fig. 4.39 The correlogram of the processed data

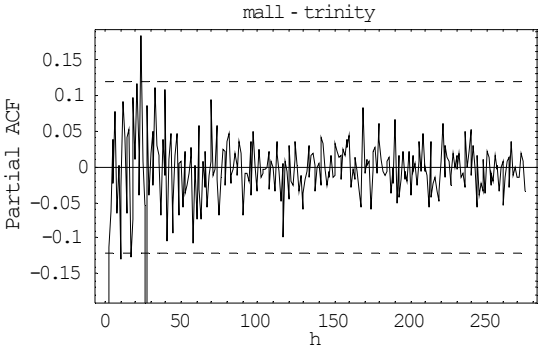

Fig. 4.40 The partial correlogram of the processed data

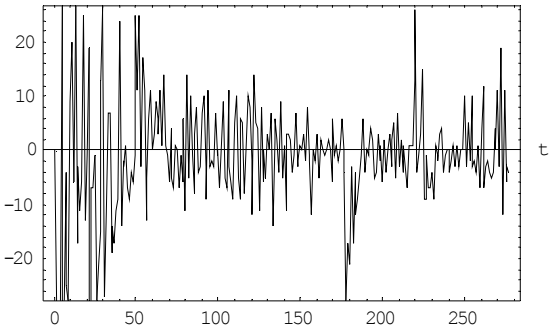

Fig. 4.41 The time plot of the residual time series

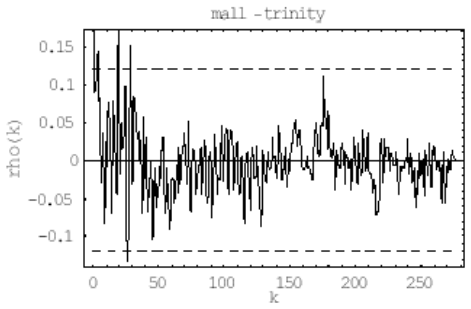

Fig. 4.42 The correlogram of the residual time series

● Trinity Road – Maple Grove Road

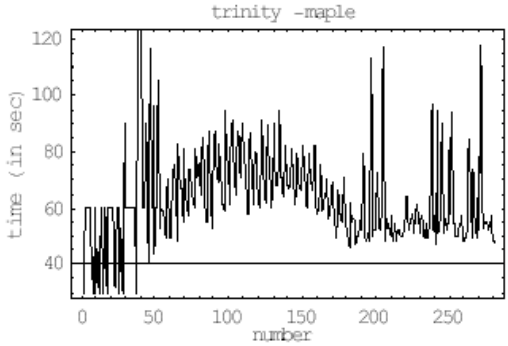

Fig. 4.43 Time plot of the Trinity-Maple Grove section travel time data

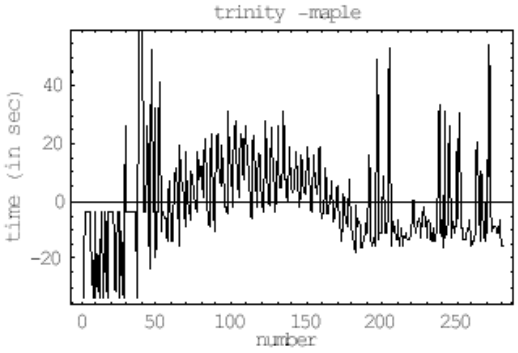

Fig. 4.44 Time plot of the section travel time data after removing its mean value

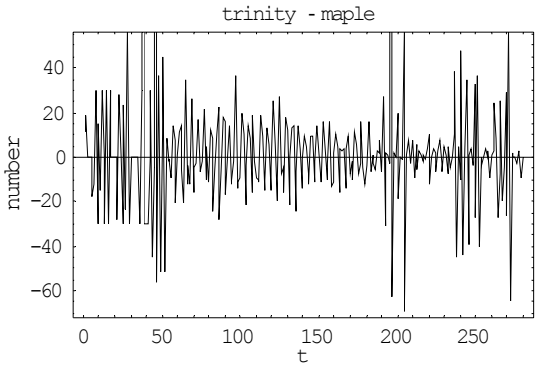

Fig. 4.45 The differenced time series data of road section # 7

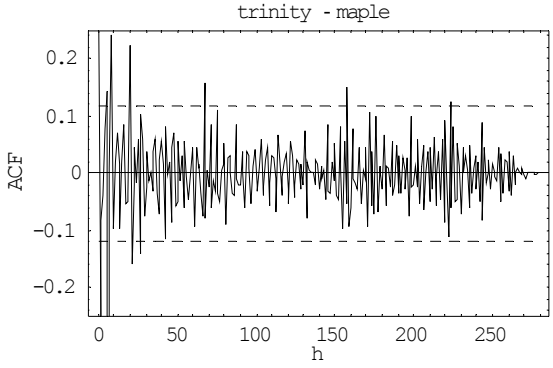

Fig. 4.46 The correlogram of the processed data

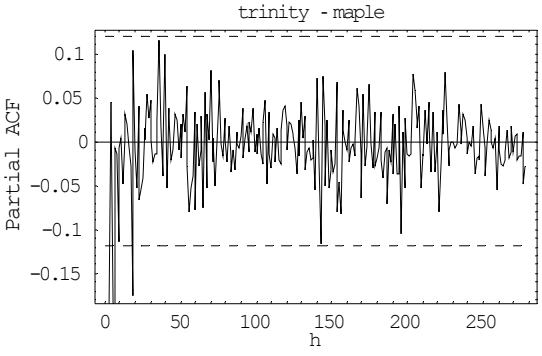

Fig. 4.47 The partial correlogram of the processed data

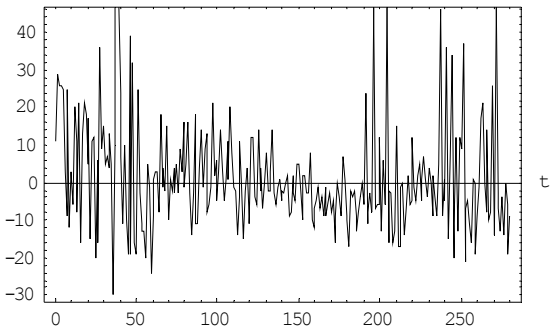

Fig. 4.48 The time plot of the residual time series

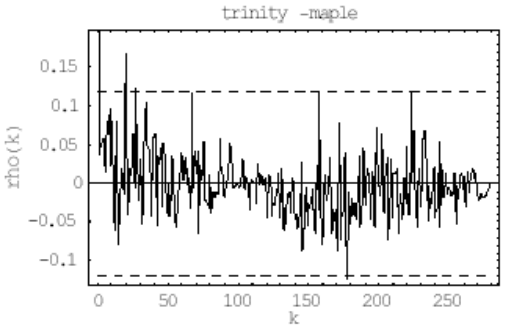

Fig. 4.49 The correlogram of the residual time series

● <u>Maple Grove Road – Haines Road</u>

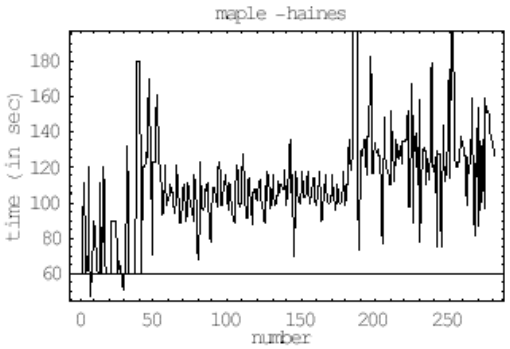

Fig. 4.50 Time plot of the Maple Grove-Haines section travel time data

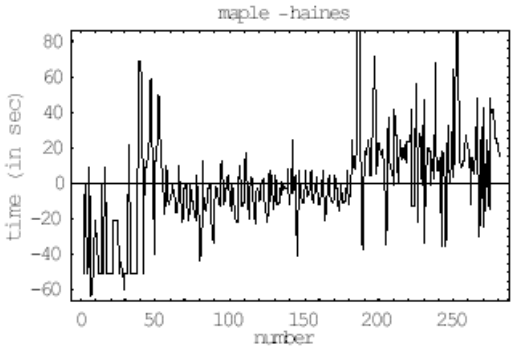

Fig. 4.51 Time plot of the section travel time data after removing its mean value

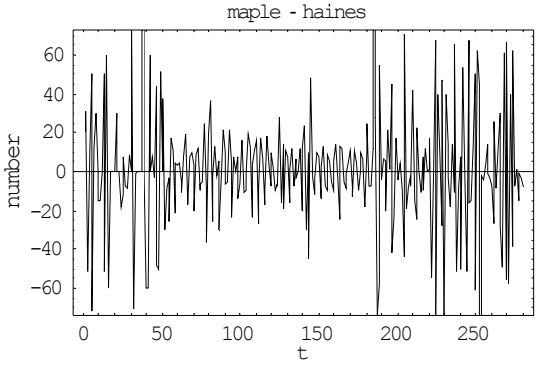

Fig. 4.52 The differenced time series data of road section # 8

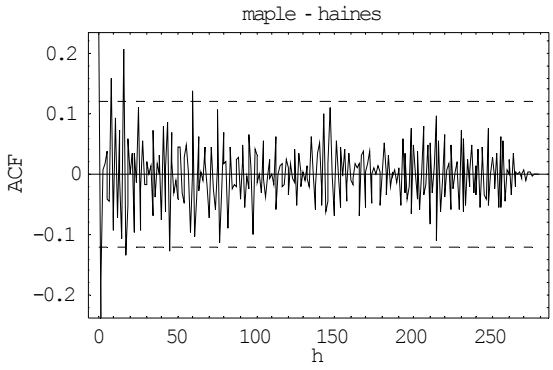

Fig. 4.53 The correlogram of the processed data

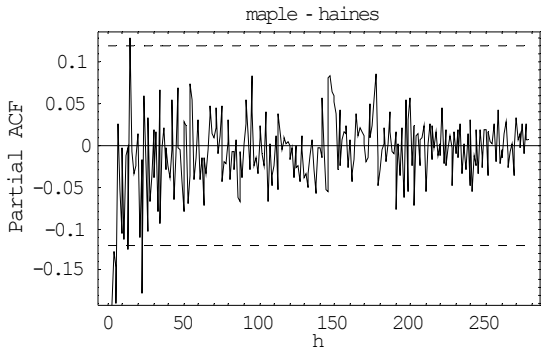

Fig. 4.54 The partial correlogram of the processed data

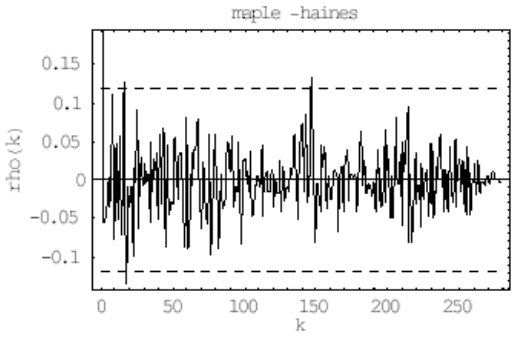

Fig. 4.56 The residual correlogram of the time series

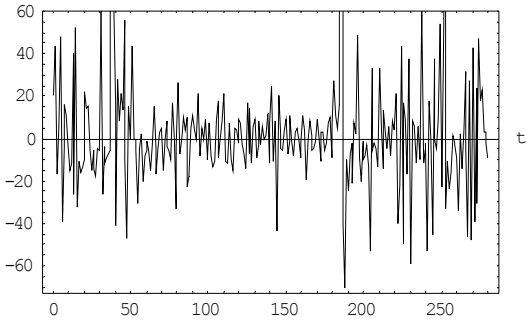

Fig. 4.55 The time plot of the residue time series

CHAPTER 5

TRAVEL TIME PREDICTION

The performance evaluation investigated in this chapter uses the developed time series models and real-time data to predict section travel time. In addition, the effect of travel time correlation data between consecutive road sections was also considered. The observed and predicted travel times on the corridor are compared and shown. The study indicates that these state-space ARIMA models with the Kalman filter can predict section travel times with reasonable accuracy. The results presented in this chapter can be easily modified and used in short-term arterial TTP for other urban areas.

5.1 Data Correlation

It is important to study the TTD between consecutive road sections to see if any close correlation between the observed data exists. If a strong correlation exists, then its effect should also be included when predicting travel time using the developed models. In other words, correlation is important in this case because the travel time of one road section may also depend on the travel time of the previous road section. The correlation of the collected time series data is studied in this section for the purpose of refining our established models. We investigate this issue by calculating the correlation coefficient via the Pearson's formula.

Correlation is a statistical technique which can show whether and how strongly pairs of variables are related. Frequently the word is used for Pearson's correlation [33] which is the covariance divided by the product of the standard deviations. The correlation coefficient provides an index of the degree to which paired measures, say {X} and {Y} data sequences, co-vary in a linear fashion. This correlation coefficient, denoted as r, is +1 or -1 when all values fall on a straight line, not parallel to either axis. If r is close to 0, it means there is no relationship between the variables. If r is positive, it means that as one variable gets larger the other also gets larger. However, a negative r simply means that as one gets larger, the other gets smaller (often called an "inverse" correlation). For the correlation calculation, we use the Pearson's formula. Let X be a sequence of values for one type, Y be a sequence of values another type and N be the number of total values, then the Pearson's formula can be represented by the correlation coefficient r as follows:

$$r = \frac{\Sigma XY - (\Sigma X \, \Sigma Y)/N}{\sqrt{[\Sigma X^2 - (\Sigma X)^2/N][\Sigma Y^2 - (\Sigma Y)^2/N]}}$$

where N is the number of data points of the sequences {X} and {Y}. Let the data sequence $\{x_k\}$ be the travel times for one road section and the sequence $\{y_k\}$ the travel times for the following road section, then using the above correlation formula, we calculated the data correlation on the corridor. Table 5.1 shows how the consecutive road

STTD is related. In this table, we set N = 92, that is, we used the data points up to March 31, 2005.

Correlation Between STTD		Correlation
Section	**Section**	**Coefficient (r)**
Section # 1	Section # 2	0.149212
Section # 2	Section # 3	0.195668
Section # 3	Section # 4	0.063996
Section # 4	Section # 5	0.020769
Section # 5	Section # 6	0.150923
Section # 6	Section # 7	- 0.082520
Section # 7	Section # 8	0.493695

Table 5.1 Correlation of measured travel time data between adjacent road sections

From Table 5.1, we found that the correlation is relatively low except for the sections between Trinity Road - Maple Grove Road and Maple Grove Road – Haines Road, where the correlation coefficient between these two streams of TTD approaches 0.5, indicating that they are weakly "linearly" correlated. In other words, on this corridor, adjacent road STTD sequences are not highly related because of their low correlation values; therefore, they can be safely treated as nearly independent. (or uncorrelated) time series data streams. Only the last two sections show a weak positive linear relationship. The relationship between these two data sequences can be roughly expressed as a linear regression equation

$$Y_k = 51.60890 + 0.68634\, X_k$$

where $\{X_k\}$ represents the TTD on the Trinity - Maple Grove section, while $\{Y_k\}$ represents the data on the Maple Grove - Haines section. The above linear regression simply suggests that once we have the value of X_k then we can use it to predict the travel time Y_k. The derivation was based on the observation of their scatter plot shown in a two-dimensional plane and calculated using the Statistical Analysis Software (SAS) [34, 35]. Since the correlation is not high, however, the use of this information should be very cautious. In the following section we demonstrate how this information is incorporated with the ARIMA state-space model given in Chapter 4.

5.2 Travel Time Prediction via ARIMA Model and Data Correlation

The travel time correlation between adjacent road sections is insignificant except for the last two road sections. We focus on the comparison of TTP on the Maple Grove - Haines section using data information from its previous road section (i.e., the linear regression in Section 5.1). In the following section we will evaluate the entire corridor using the models developed.

Since we want to use the correlation information in the prediction of travel time for the Maple Grove - Haines section, a proper weighting on the results from both the prediction model and linear regression should be determined. Nevertheless, due to the low correlation as shown in Table 5.1, the weight on this portion should be well below 50%. As a result, we have put more confidence in using the time series model to predict travel time. An experiment to predict the Maple Grove - Haines section travel time was conducted during the afternoon peak hour (i.e., 3:30 - 5:00 pm) from March 20 to March 25, 2005. We used that road section time series state-space model incorporated with its previous road section data correlation information to predict the next 24 travel-times A further study indicated that 90% weight allocated to prediction from the prediction model seemed to generate better result. The following table summarizes the total error we found from this experimental study.

Percentage Weight		**Total Error**[†]
ARIMA Model (α)	**Correlation (linear regression) (β)**	
100 %	0 %	2575
90 %	10 %	2559
80 %	20 %	2565
50 %	50 %	2708

[†] Sum of the squared errors between the observed and predicted values

Table 5.2 Error comparison under various weighting

The performance comparison with $\alpha = 100\%$ and 90% are shown respectively in Figs. 5.1 and 5.2.

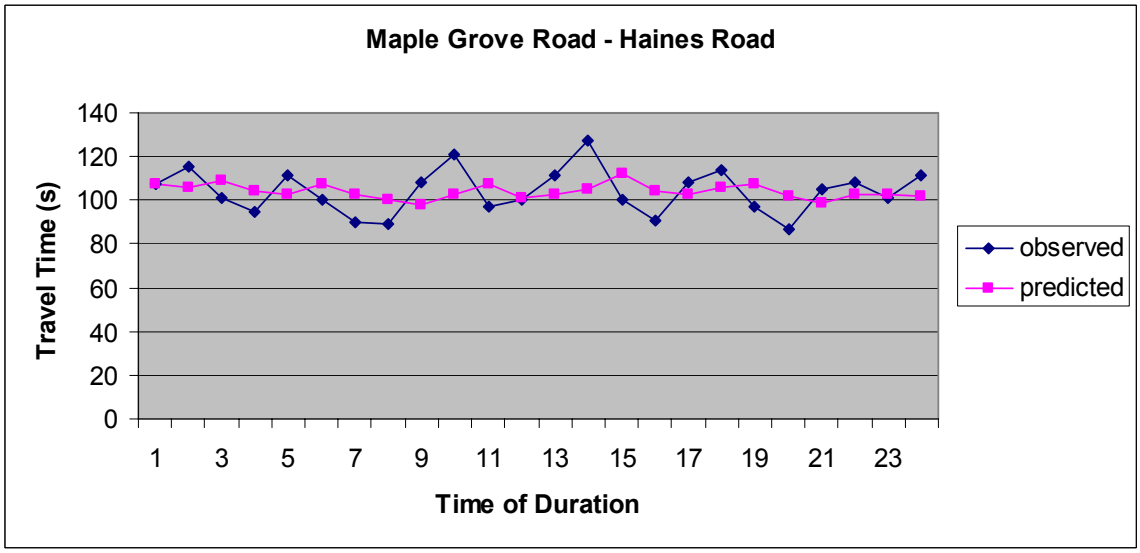

Fig. 5.1 Performance study based on the prediction model only (i.e., $\alpha = 100$ %)

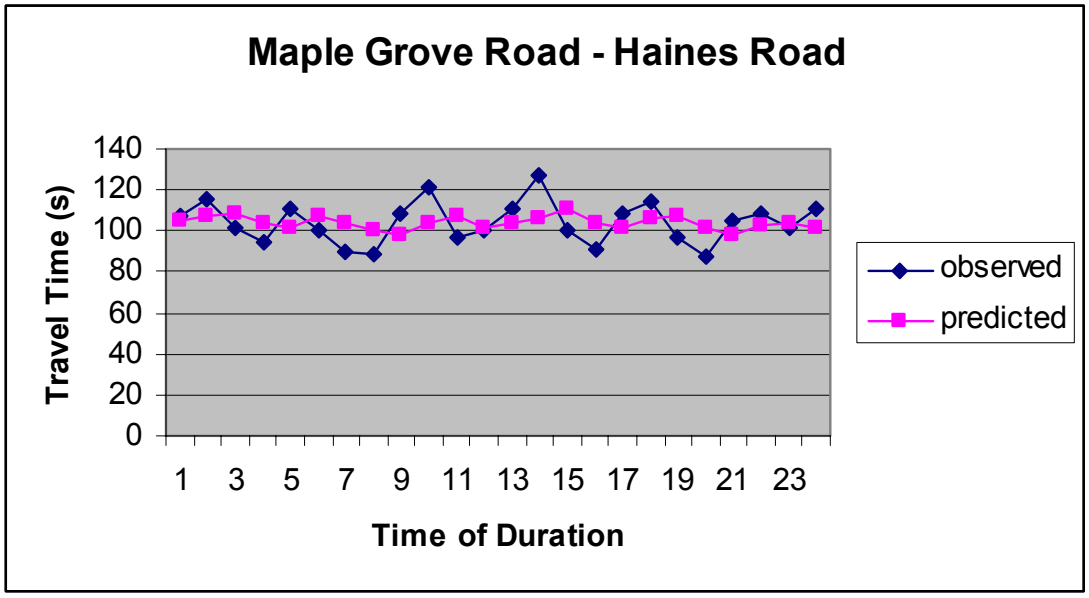

Fig. 5.2 Performance study with α chosen to be 90 %

Note that in the above figures, the curve marked with ■ represents the predicted travel times whereas the one marked with ♦ means the observed values. The x-axis denotes the number of data-points (i.e. 24 next travel times) whereas the y-axis represents the travel times in seconds.

5.3 Performance Evaluations

In this section, we use the time series models summarized in Table 4.3, their state-space representations (see Section 4.3) and the Kalman recursions described in Section 2.3 to evaluate the performance on the corridor. Note that these models are based on the measured TTD over an eight-month period and the total data point used for each road section is given in Table 4.2.

The TTP was conducted over a two-week period from July 18 to July 22 and July 25 to July 29, 2005 during the afternoon peak hour (i.e., 3:30 - 5:00 pm). Real-time data was used together with our prediction models to perform the one-step-ahead prediction. The observed and predicted values for different road sections are shown, respectively, in Figs. 5.3 - 5.10. In these figures, the curve marked with ♦ means the predicted travel times and the one marked ■ represents the observed values. The x-axis denotes the number of data-points (i.e. 40 next travel times) whereas the y-axis denotes the travel times in seconds. From Figs. 5.3-5.10, we see that these ARIMA time series models produce reasonably good TTP results for most of the road sections. The predicted values are within the range of our observed travel times. It does especially well for the road sections with higher allowed speed limit such as the Trinity – Maple Grove section and the Maple Grove – Haines section. Another observation is that the prediction error seemed relatively large on those road sections with a shorter distance. Lower speed limit and shorter link distance together with the relatively high cross-street traffic can all affect our prediction performance.

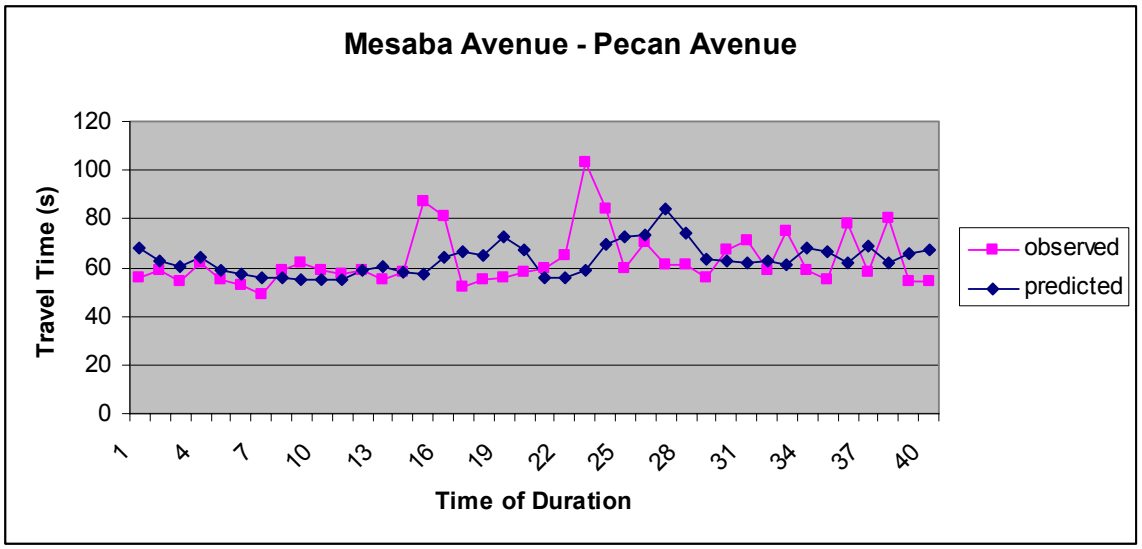

Fig. 5.3 Result comparison on the Mesaba-Pecan section.

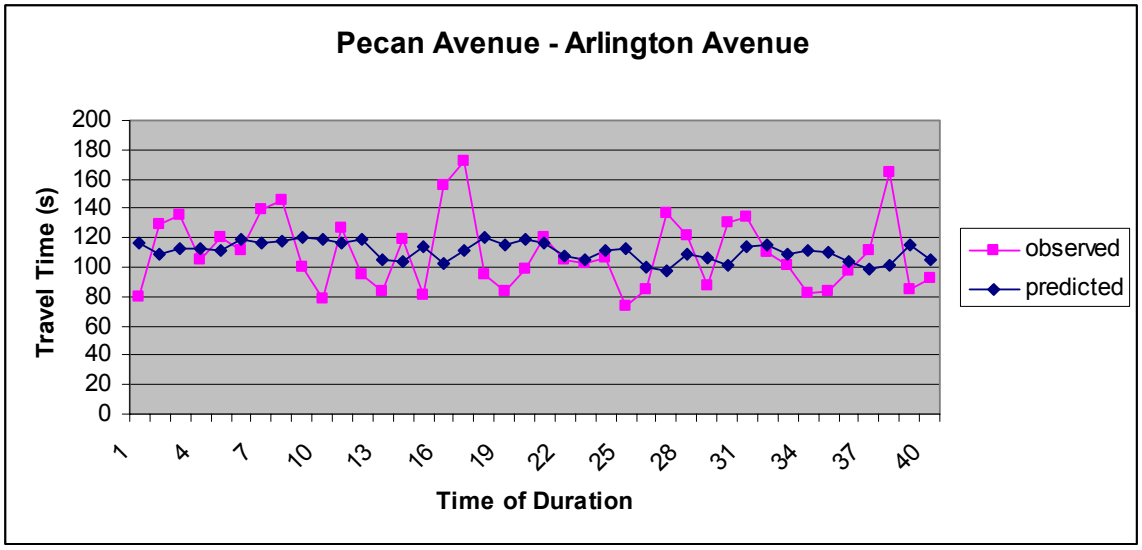

Fig. 5.4 Result comparison on the Pecan-Arlington section.

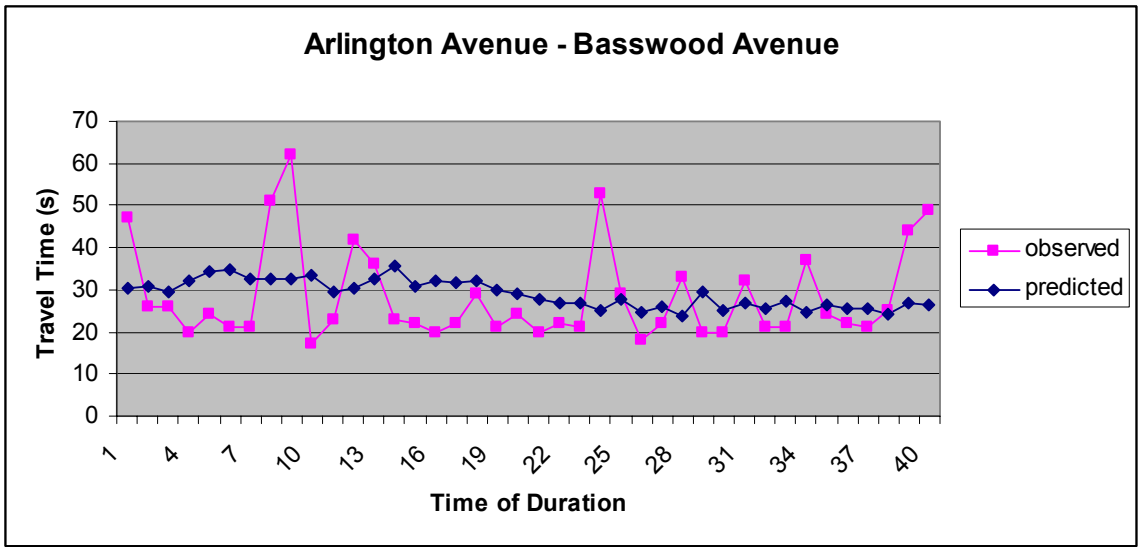

Fig. 5.5 Result comparison on the Arlington-Basswood section

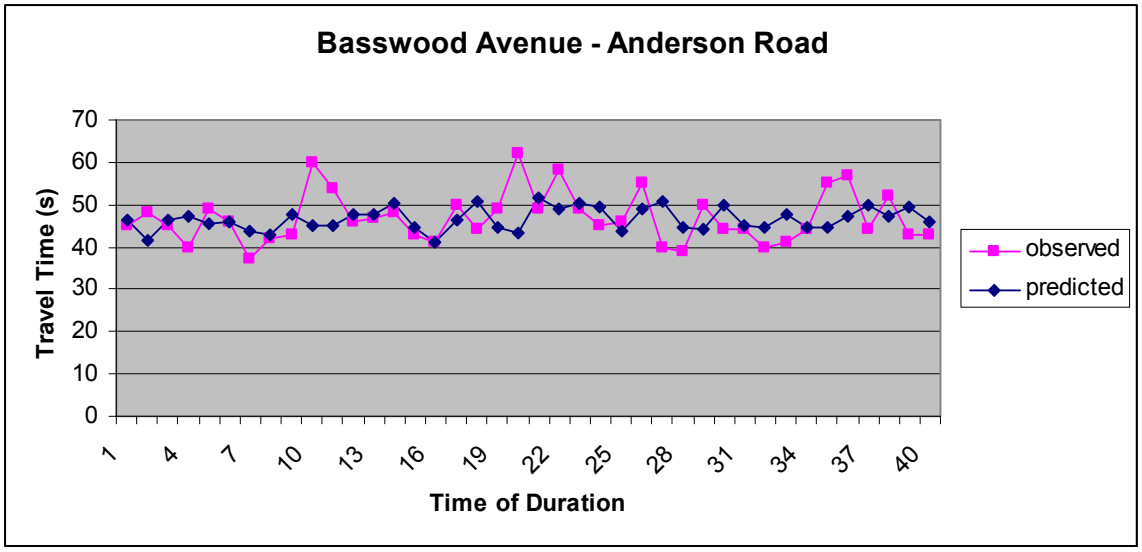

Fig. 5.6 Result comparison on the Basswood-Anderson section

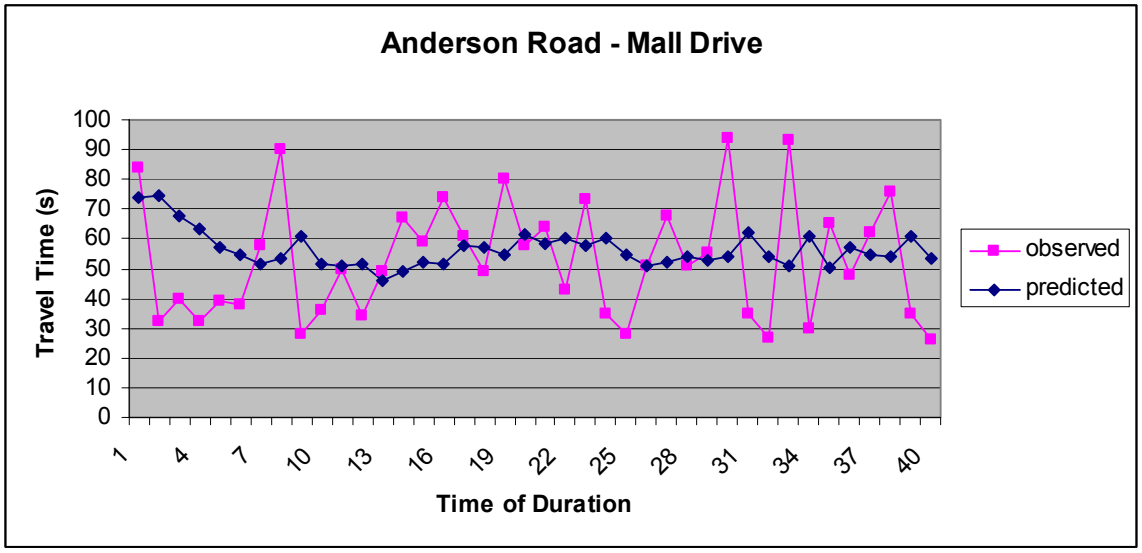

Fig. 5.7 Result comparison on the Anderson-Mall section

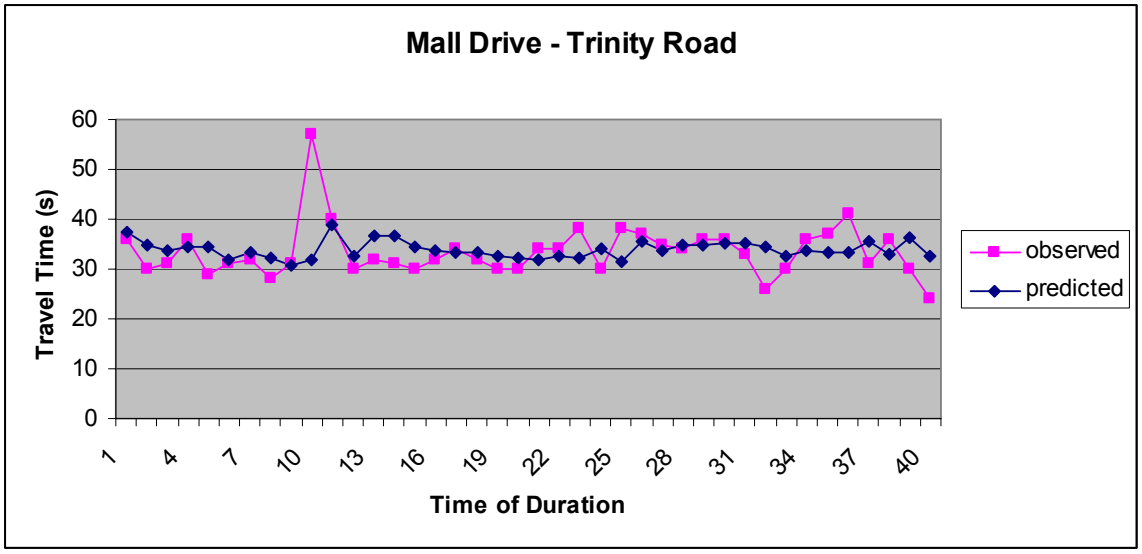

Fig. 5.8 Result comparison on the Mall-Trinity section

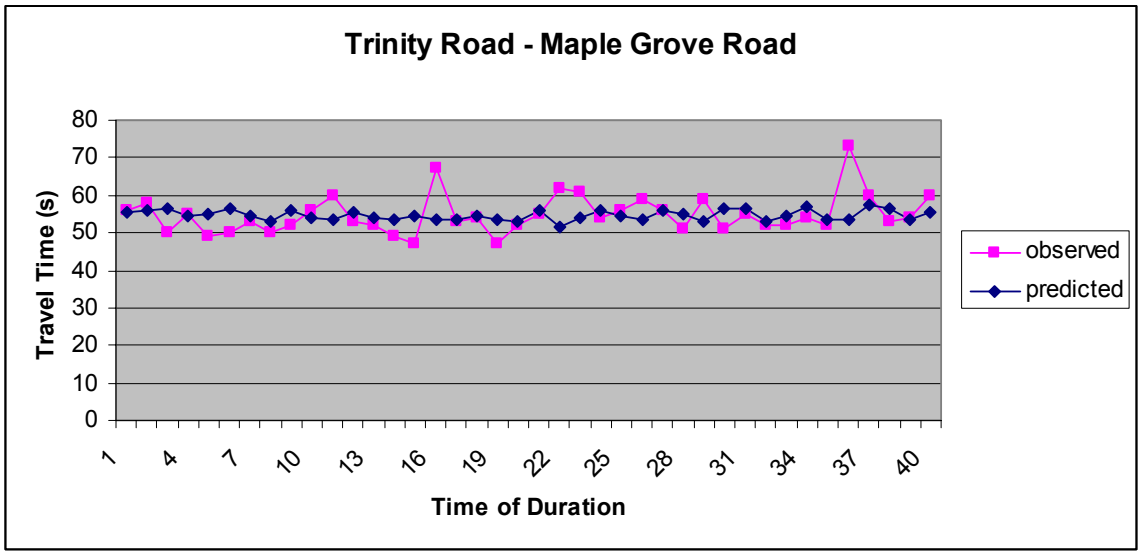

Fig. 5.9 Result comparison on the Trinity-Maple Grove section

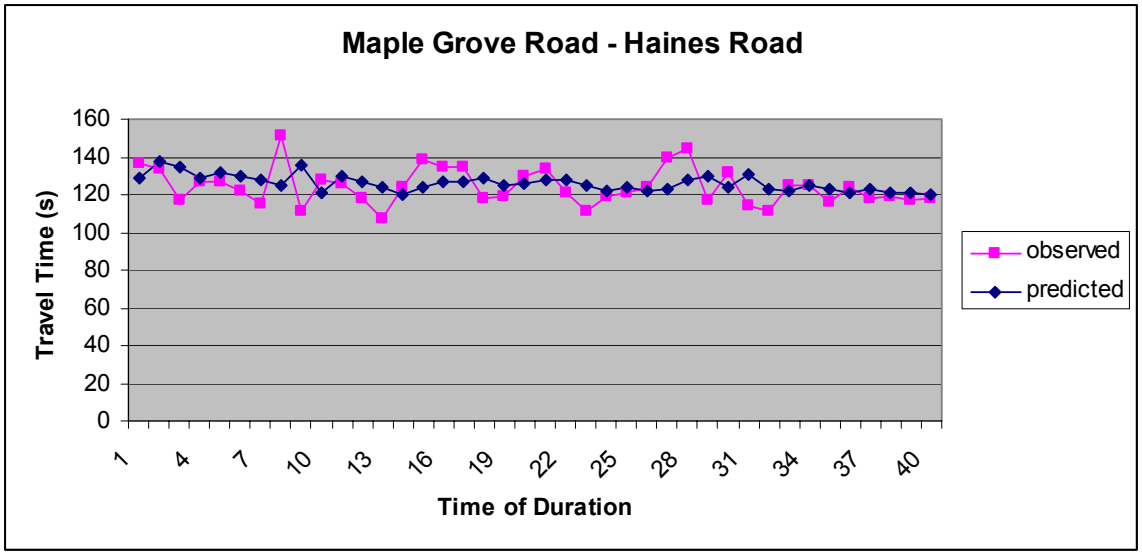

Fig. 5.10 Result comparison on the Maple Grove-Haines section

Notice that the incorporation with data correlation in TTP depends on each individual arterial under study. The road geometry, peak or non-peak hours, cross-street traffic, signal timing, etc. can all affect the effectiveness of using the data correlation information between any consecutive road sections. Of course, different traffic signal timing used on arterials during the peak- and non-peak hour can affect travel times but this will not affect our TTP results (i.e., the error performance) because the time series model order and model parameter values also need be updated under different timing plans before TTP should be proceeded.

It is known that good TTP is very challenging due to the existence of various uncertainties (e.g., weather and road conditions, traffic accidents, etc). In addition, some conditions may not be known well in advance or can even be anticipated. There are many methods/techniques proposed in the literature, but to the best of our knowledge none of these methods, both linear and nonlinear, can claim to be able to apply to all situations with high accuracy. However, based on the study we conducted, it can be seen that, in general, the auto-regressive integrated moving average approach can establish a reasonably good model for TTP purposes. This approach particularly performed well for arterials subjected to both queuing and signal delays, which sometimes cannot be handled nicely or even performed poorly by using those prediction methods developed on freeways. Our study indicates the potential and effectiveness of using the time series modeling in the prediction of arterial travel time. Furthermore, the results presented here can be easily modified and used in short-term arterial TTP for other urban areas.

CHAPTER 6

CONCLUSION

This project focuses on the modeling and prediction of arterial section travel times via the time series analysis and Kalman recursions techniques. The ARIMA model and properties are introduced and its state-space representation is also derived. The developed state-space model is then further used in the Kalman filter formulation to perform one-step-ahead TTP. The performance is conducted on a section of Minnesota State Highway 194 between Mesaba Avenue and Haines Road, one of the most heavily congested corridors in the Duluth area. We use the GPS test vehicle technique to collect STTD. The calculated STTD, based on the received GPS time stamp data, is then treated as a realization of a time series process. Therefore, time series analysis is used in the development of section travel time models. During the modeling process, the information criteria are used to select model orders, while the model parameter values are estimated via the Hannan-Rissanen algorithm. The models developed are further validated via both the residual analysis and the portmanteau test. The data correlation between adjacent road sections is also studied via the Pearson's formula. The linear regression derived from the data correlation is also incorporated with the state-space model when performing the TTP to improve the overall results. This correlation information is used only when the data correlation factor needs to be included. The performance evaluation includes the comparison of the observed and predicted values over different road sections on the corridor. We found that, in general, the ARIMA time series models produce reasonably good prediction results for most of the road sections studied. Our study shows the potential and effectiveness of using the time series modeling in the prediction of arterial travel time. Furthermore, the results presented here can be easily modified and used in short-term arterial TTP for other urban areas.

REFERENCES

[1] H. Al-Deek,, M. D'Angelo, and M. Wang, "Travel time prediction with non-linear time series," *Proceedings of the 5th ASCE International Conference on Applications of Advanced Technologies in Transportation*, 1998, 317-324.

[2] M. S. Ahmed and A. R. Cook, "Analysis of Freeway Traffic Time Series Data by Using Box-Jenkins Techniques," *Transportation Research Record* 722, 1979, 1-9.

[3] L. Rilett and D. Park, "Direct forecasting of freeway corridor travel times using spectral basis neural networks," *Transportation Research Record* 1617, 1999, 163-170.

[4] P. V. Palacharla and P. C. Nelson, "Application of fuzzy logic and neural networks for dynamic travel time estimation," *International Transactions in Operational Research*, *6*, 1999, 145-160.

[5] V. P. Sisiopiku, N. M. Rouphail, and A. Santiago, "Analysis of correlation between arterial travel time and detector data from simulation and field studies," *Transportation Research Record 1457*, 1994, 166-173.

[6] J. A. Rice and E. Zwet, "A simple and effective method for predicting travel times on freeways," http://www.state.berkeley.edu/users/rice/research.html

[7] X. Zhang and J. A. Rice, "Short-term travel time prediction using a time-varying coefficients linear model," http://www.state.berkeley.edu/users/rice/research.html

[8] X. Zhang, J. A. Rice, and P. Bickel, "Empirical comparison of travel time estimation methods," http://www.state.berkeley.edu/users/rice/research.html

[9] D. J. Dailey, "Travel-time estimation using cross-correlation techniques," *Transportation Research, Part B*, *27*, 1993, 97-107.

[10] C.-H. Wu, J.-M. Ho, and D. T. Lee, "Travel time prediction with support vector regression," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 4, 2004, 276-281.

[11] I. Okutani and Y. J. Stephanedes, "Dynamic prediction of traffic volume through Kalman filtering theory," *Transportation Research, Part B*, *18*, 1984, 1-11.

[12] B. M. Williams, P. K. Durvasula, and D. E. Brown, "Urban freeway traffic flow prediction – application of seasonal autoregressive integrated moving average and exponential smoothing models," *Transportation Research Record 1644*, 1998132-141.

[13] M. Chen and S. Chein, "Dynamic freeway travel time prediction using probe vehicle data: link-based vs. path-based," *Proceedings of Transportation Research Board 80th Annual Meeting*, 2002, Washington, D. C.

[14] I. Christiansen and L. Hauer, L, "Problem information for travel time," *Traffic Technology International*, August/September issue, 1996, 41-44.

[15] J. You and T. J. Kim, "Development and evaluation of a hybrid travel time forecasting model," *Transportation Research*, *Part C*, vol. 8, 2000, 231-256.

[16] W. H. Lin, A. Kulkarni and P. Mirchandani, "Arterial travel time estimation for advanced traveler information systems," *Proceedings of Transportation Research Board 82nd Annual Meeting*, 2003, Washington, D. C.

[17] V. P. Sisiopiku, N. M. Rouphail, and A. Santiago, "Analysis of correlation between arterial travel time and detector data from simulation and field studies," *Transportation Research Record* 1457, 1994, 166-173.

[18] V. P. Sisiopiku and N. M. Rouphail, "Toward the use of detector output for arterial link travel time estimation: a literature review," *Transportation Research Record* 1457, 1994, 158-165.

[19] I. Okutani and Y. J. Stephanedes, "Dynamic prediction of traffic volume through Kalman filtering theory," *Transportation Research, Part B*, vol. 18, 1984, 1-11.

[20] *Travel time data collection handbook* (Texas Transportation Institute & US Federal Highway Administration, FHWA-PL-98-035, March 1998).

[21] P. J. Brockwell and R. A. Davis, *Introduction to time series and forecasting* (New York, NY: Springer, 2002).

[22] C. Chatfield, *The analysis of time series: an introduction* (Boca Raton, FL: Chapman and Hall/CRC, 2003).

[23] R. H. Shumway and D. S. Stoffer, *Time series analysis and its applications* (New York, NY: Springer, 2000).

[24] G. E. Box, G. M. Jenkins, and G. C. Reinsel, *Time series analysis, forecasting and control* (Upper Saddle River, NJ: Prentice Hall, 1994).

[25] P. J. Brockwell and R. A. Davis, T*ime series: theory and methods* (New York, NY: Springer, 1991).

[26] H. W. Sorenson, *Kalman filtering: theory and applications* (New York, NY: IEEE Press, 1985).

[27] R. G. Brown and Y. C. Hwang, *Introduction to random signals and applied Kalman filtering* (New York, NY: John Wiley & Sons, 1997).

[28] P. Zarchan and H. Musoff, *Fundamentals of Kalman filtering and applications* (Reston, VA: American Institute of Aeronautics and Astronautics (AIAA), 2000).

[29] J. Chen and M. Kuchipudi., "Dynamic travel time prediction with real-time and historical data," *Transportation Research Board the 81st Annual Meeting*, Washington D.C., January 2002.

[30] S. I. Chien X. Liu, and K. Ozbay, "Predicting travel times for the South Jersey real-time motorist information system", *Journal of Transportation Research Board,* October 2002.

[31] H. Suzuki, T. Nakatsuji, Y. Tanaboriboon, and K. Takahashi, "A neural-Kalman filter for dynamic estimation of origin-destination (O-D) travel time and flow on a long freeway corridor," *Transportation Research Record* 1739, 2000, 67-75.

[32] J. Gallagher, "Travel time data collection using GPS", *National Traffic Data Acquisition Conference,* Albuquerque, New Mexico, May 1996.

[33] D. Laird, "Emerging issues in the use of GPS for travel time data collection", *National Traffic Data Acquisition Conference,* Albuquerque, New Mexico, May 1996.

[34] B. W. Parkinson and J. J. Spilker, *GPS: theory and applications,* vol. I and vol. II (Reston, VA: American Institute of Aeronautics and Astronautics (AIAA), 1996).

[35] J. Blinn, "A trip down the graphics pipeline: line clipping," *IEEE Trans. Computer Graphics and Applications*, *11*, 1999, 98-105.

[36] P. Asokarathinam, Cohen-Sutherland Line Clipping, http://www.cs.helsinki.fi/group/goa/viewing/leikkaus/lineClip.html

[37] S. M. Burke, Peral & LWP, Fetching Web Pages, Parsing HTML, Writing Spiders & More (Cambridge, MA: O'Reilly, 2002)

[38] Website Scraping, http://www.perl.com/lpt/a/2003/01/22/mechanize.html

[39] Pearson Correlation, http://davidmlane.com/hyperstat/A51911.html

[40] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to linear regression analysis* (Hoboken, NJ: John Wiley & Sons).

[41] SAS/STAT software, http://www.sas.com/technologies/analytics/statistics/stat/