DEVELOPMENT OF A REAL TIME BUS ARRIVAL TIME PREDICTION SYSTEM UNDER INDIAN TRAFFIC CONDITIONS

A sub-project from

“Center of Excellence in Urban Transport”

Sponsored by

The Ministry of urban Development, Government of India.

June 2016

Sub-project team members
IIT Madras

An Interim Report on

Development of a Real Time Bus Arrival Time Prediction System under Indian Traffic Conditions

A sub-project from the "Center of Excellence in Urban Transport"
sponsored by
The Ministry of urban Development, Government of India.

Sub-project team members

IIT Madras

June 2016

This work was carried out as part of the activities in the Centre of Excellence in Urban Transport at IIT Madras sponsored by the Ministry of Urban Development.

An Interim Report on

Development of a Real Time Bus Arrival Time Prediction System under Indian Traffic Conditions

Dr. Lelitha Vanajakshi
Associate Professor
Department of Civil Engineering
Indian Institute of Technology Madras
Chennai 600 036
INDIA
Ph : 91 44 2257 4291, Fax: 91 44 2257 4252
E-mail: lelitha@iitm.ac.in

Dr. Shankar C. Subramanian
Associate Professor
Department of Engineering Design
Indian Institute of Technology Madras
Chennai 600 036
INDIA

Akhilesh Koppineni, Krishna Chaitanya, Siddharth K., Rakesh Behera, Padmanabhan, R. P, S., Vasantha Kumar, Anil Kumar
Center of Excellence in Urban Transport
Department of Civil Engineering
Indian Institute of Technology Madras
Chennai 600 036

Executive Summary

Intelligent transportation systems (ITS) are transportation services and technologies aimed at enhancing the efficiency, safety, reliability and eco-sustenance of transportation systems, without constructing new infrastructure. An important aspect of ITS is to advance public transportation to make it more attractive than private transport.

India has traditionally boasted an extensive public transportation system, being the second largest producer of buses, accounting for 16 percent of world's total bus production. However, the share of public transportation in Indian cities has been on a steady decline over the last few decades due to, among other reasons, poor management of services. One of the major factors responsible for the success or failure of any public transport service is its reliability. One way of improving the reliability is to provide the passengers with accurate and reliable information regarding the service. This report describes a study performed on this important aspect, viz. bus arrival time prediction, a key field covered under the wide umbrella of ITS. This study is a part of a larger investigation and this report presents preliminary observations.

During this phase of research, analyses were first performed using the GPS data collected manually from buses running in route numbers 21L and 21G in Chennai, India. Once the performance was found to be acceptable, online data integration was carried out for route numbers 5C and 19B. In order to have a real time automated application, it was required to develop an automated data collection filtration and analysis system without manual intervention. Numerous issues were identified and addressed during real-time implementation of the model. Issues such as effect of traffic jam, overtaking among the buses on the same route, bus breakdown, abrupt changes in bus routes, etc. which are specific to Indian conditions were taken into consideration.

A model based prediction scheme was used and the Kalman Filtering Technique was adapted to estimate and predict the travel time of buses. Different modifications to the system were attempted for improving the performance and those that were improving the performance were incorporated into the model. The study also developed a prototype of the complete system integrating with the information dissemination units. Prototypes for three different information dissemination modes were included, viz. bus stop VMS display, Kiosk display at bus stops and web based application. An advanced kiosk display system was developed considering specific requirements of users. Such a system has advantages of cost efficiency and provides greater information in a cognitively ergonomic format to aid the commuter's decision making skills.

Although the application developed covers all aspects of real-time implementation, there is still scope for improvement. Search for better algorithms for more accurate prediction is an open ended problem. In the present study due to limitations of the KFT based models, the data from the test vehicle was not used in the algorithm and was kept for validation purpose alone. New algorithms that would use real-time data from the test vehicle are being explored. Modification of prediction algorithm will be an ongoing process along with evaluations. Transferability and scalability of the system need to be taken into account.

Acknowledgements

This report was developed as part of the activities at the Centre of Excellence in Urban Transport (CoE-UT), IIT Madras, sponsored by the Ministry of Urban Development, Government of India. We thank the Ministry of Urban Development for sponsoring the CoE-UT at IIT Madras. We also thank the Director, Dean (Industrial Consultancy & Sponsored Research), the Head of the Department, Department of Civil Engineering for their support and guidance to the Centre. We thank the Centre coordinators for providing us the opportunity to work on this report. Special thanks to Valardocs for the prompt and professional technical editing support.

Development of a Real Time Bus Arrival Time Prediction System under Indian Traffic Conditions

I. INTRODUCTION

1.1. OVERVIEW

India's transport sector is large and diverse; it caters to the needs of 1.1 billion people. According to a World Bank report (2007), the transport sector contributed about 5.5 percent to India's GDP, with road transportation contributing the lion's share. Since the early 1990s, India's growing economy has witnessed a rise in demand for transport infrastructure and services. However, the sector has not been able to meet this growing demand. There is therefore, a need for alternative solutions to manage this demand. A promising approach is the adoption of Intelligent Transportation Systems (ITS).

Intelligent transportation systems (ITS) are transportation services and technologies aimed at enhancing the efficiency, safety, reliability and eco-sustenance of transportation systems, without constructing new infrastructure. ITS has many sub systems such as Advanced Traffic Management Systems (ATMS), Advanced Public Transit Systems (APTS), Advanced Traveler Information Systems (ATIS), Advanced Vehicle Control Systems (AVCS), and Commercial Vehicle Operations (CVO).

An important aspect of ITS is to advance public transportation to make it more attractive than private transport. India has traditionally boasted an extensive public transportation system, being the second largest producer of buses, accounting for 16 percent of world's total bus production[1]. However, the share of public transportation in Indian cities has been on a steady decline over the last few decades due to, among other reasons, poor management of services.

Advanced Public Transportation Systems (APTS) applies state-of-art transportation management and information technologies to public transit systems to enhance efficiency of operation and improve safety. It includes real-time passenger information systems, automatic vehicle location systems, bus arrival notification systems, and systems providing priority of passage to buses at signalized intersections (transit signal priority).

This report describes a study performed on one aspect of APTS, viz. bus arrival time prediction, a key field covered under the wide umbrella of ITS. This study is a part of a larger investigation and this report presents preliminary observations.

[1] http://www.siamindia.com/UpLoad/Coverage/3192/Newsletter3.pdf

1.2. MOTIVATION

Travel duration in public transportation systems is a direct measure of their efficiency and usefulness. Travel time information is also important in planning operations, signal timing coordination and route assignments. Design and implementation of ITS tools depend on accurate predictions of travel durations by extrapolation of existing travel time data.

There have been many reports of short-term travel time prediction in the past. These investigations have been performed under conditions of homogeneous traffic. Such models and algorithms may not be directly applicable to Indian traffic conditions.

The traffic in India is heterogenic in composition. A variety of vehicles – two, three and four wheelers, in addition to a large pedestrian population, share the Indian urban road. This heterogeneity, coupled with poor lane discipline makes travel time prediction more challenging than can be handled by conventional methods. There have been only a few earlier studies on understanding traffic behavior under Indian road conditions (Vanajakshi et al., 2009; Ramakrishna et al., 2006). While these studies provide insights into the nature of traffic, they have not considered real time field implementation of their models.

Figure 1: Common causes of bus delays in India (a) overcrowding (b) Unscheduled strikes, political/religious conventions (c) Accidents (d) Poor lane discipline

1.3. OBJECTIVES

The objective of the project is to develop a bus passenger information system to efficiently provide real time bus arrival time information under Indian traffic conditions. This is a preliminary step towards field implementation of reliable APTS for Indian Public Transportation Systems.

The technology uses GPS to gather data, factoring conditions such as traffic jams, overtaking, breakdowns and unscheduled changes in bus routes that are specific to Indian conditions. Prototypes were developed for selected routes in Chennai, India. This can be extendable to other places if the buses plying these routes are equipped with GPS/GPRS units. Bus travel information can eventually disseminated to the public through various media such as display boards and kiosks at bus stops, display boards in bus, through the internet and by SMS. Prototypes for each of these dissemination systems are also developed.

The project comprises the following tasks:

1. Review on state-of-art of bus arrival prediction algorithms and systems
2. Identification and Procurement of hardware – GPS units, and servers
3. Obtaining permission for permanently fixing the units in MTC buses and installation
4. Real time data collection and Data quality control
5. Database development
6. Travel time and arrival time prediction algorithm development
7. Corroboration of the algorithm
8. Integration with real time data
9. Prototype development
10. Performance evaluation
11. Scalability, Transferability issues and
12. Field implementation

Of the twelve tasks, the first ten have been performed to various degrees of completion and are described and analysed in the following chapters.

1.4. ORGANIZATION OF THIS REPORT

This report first discusses existing knowledge and describes field implementations of bus-travel time prediction methods used worldwide. This is followed by a description of the tasks conducted so far and a discussion of results observed in these studies. The report concludes with a plan for future work (tasks 10, 11 and 12 above) in order to meet the objectives described in the previous section.

II. REVIEW OF EXISTING KNOWLEDGE AND TECHNOLOGY

2.1. LITERATURE SURVEY

An important reason for the non-preference of public transportation over private transport is the lack of information on actual bus arrival times. The stochastic nature of public transit attributes such as travel time, dwell time, demand, etc., often results in unpredictable waiting and ride times. Reports show that the passengers value information about the next bus arrival time as the highest (around 79.4%) followed by the schedule adherence (around 72%) (Peng et al., 2001).

Prediction of travel times is an important route to improve the reliability and utility of public transit information. Travel time prediction involves development and use of technologies such as automatic vehicle location (AVL), special prediction algorithms and tools of communication.

Automatic vehicle location enables to remotely track the location of a vehicle with the use of mobile radio receiver, GPS receiver, GPS modem, GPS antenna, GIS (Geographic Information Systems) etc.

There have been many algorithms and methods reported in literature for prediction of travel times. Some of these include machine learning techniques (Artificial Neural Networks, Support Vector Machines), model based approaches (Kalman filtering) and statistical methods (regression analysis, time-series analysis) for the prediction of travel time. This section briefly describes existing research in the area of travel time prediction, with special reference to public transportation.

Historical data based models predicts the travel time at a particular time as the average travel time for the same period historically (Li and McDonald, 2002; Chen et al., 2003). It is based on the assumption that a historical profile can be developed for travel time that can represent the average traffic characteristics over days that have similar profile. These models assume that the travel time trend is similar to the previous days (i.e., travel time in the selected link at 9.00 am is the same every day).
These models perform well under normal conditions. However, under stochastic conditions prediction accuracy is greatly reduced.

Real-time approach predicts the travel time at the next time interval x(t+1) to be the same as the present time interval, x(t). Essentially it assumes that the future travel time to be the same as the present one. This method is inefficient whenever there is unavailability of data (i.e., loss in reception, equipment failure) (Karbassi and Barth, 2003; Manolis and Kwstis, 2004).

Statistical models are commonly used for the prediction of travel time and include models such as time-series and regression models. Time-series models assume that the historical patterns will remain in the future too (D'Angelo et al., 1999; Ishak and Al-Deek, 2002). Regression models are conventional approaches for predicting travel time and predict a dependent variable based on a function formed by a set of independent variables (Rice and van Zwet, 2001; Frechette and Khan, 1998). Travel time along a particular link is influenced by a number of factors such as driver behaviour, composition of vehicles, carriageway width, intersections, signals, etc., and all or some of these factors are used as independent variable in many studies. The accuracy of these models depends on whether all the dependant variables are identified and incorporated in the model, which is a difficult task.

Machine learning techniques such as Artificial Neural Networks (ANN) are one of the most commonly reported techniques for traffic prediction mainly because of its ability to solve complex non-linear relationships (Clark et al., 1993; Dougherty and Cobbett, 1997; Kirby et al., 1997; Dia, 2001, vanajakshi et al., 2004, 2007). ANN models are developed emulating the human brain and trained to learn from example (Stergiou and Siganos, 2009). A series of interconnected elements (neurons) forms the basis of these models and can be best used for pattern matching, prediction, etc. However, these are data driven techniques and require large set of data for better learning. Also, they are problem specific models and whenever the input variables change, the whole model has to be re-structured.

Model-based approach develops models that can capture the dynamics of the system. While the statistical models and machine learning algorithms are data driven techniques (i.e., relationship between the variables is established from the data collected) and location specific, a model based approach establishes relationship between the variables based on theory and then validate using field observation (Liu et al., 2005). These models are more robust and generic and not usually data specific or site specific. Many of the model based studies use Kalman filtering technique for the estimation and prediction of the parameters.

Travel time prediction methods applied specifically to public transportation services also falls in the above categories. For example, Manolis and Kwstis (2004) developed a bus arrival prediction model for the city of Heraklion in Greece using historical method. GPS-GPRS technology was used to collect the real-time data. Each bus reported its location at every 150m to a central computer and the mean speed for every 500m was then calculated. All the bus stops were geo-coded into the GIS supporting system and the travel times near the bus stops were simply subtracted to get 'Bus stop waiting time'. Mean speed of the vehicles along the main corridors were obtained from central computer and travel time was calculated. The prediction accuracy was in the range of 1 - 2 minutes.

Patnaik et al. (2004) developed regression models using APC units installed on buses. They developed the models using path based data. The variables selected were distance travelled, demand characteristics and time of day. The study results were not validated using field data.

Lin and Zeng (1999) developed a real-time bus arrival information system for a rural area using a set of mathematical algorithms. They used bus location data, schedule information, difference between scheduled and actual arrival times, and the waiting time at time-check stops. The trajectory of the vehicle was constructed based on the time-distance diagram using the AVL data from buses. The performance of the algorithm was tested using various levels of information for three criteria namely overall precision, robustness and stability and they reported that dwell time at the time-check stops is most relevant to the performance. However, the GPS units provided the location detail at every 45 seconds only. This long gap between location data may lead to missing of dwell times at bus stops.

Chien et al. (2002) developed link-based and path-based ANN models to predict bus arrival time dynamically. They used simulated data on volume and passenger demand using CORSIM simulation software. They developed an adaptive algorithm for bus travel time prediction. They stated that conventionally used back-propagation algorithms are difficult to implement for real time applications due to its lengthy learning process. They reported that ANN outperformed the models without integration of adaptive algorithm.

Jeong and Rilett (2004) evaluated the performance of historical data based model, regression model and ANN model for the short term bus arrival time prediction. They used AVL data from buses for a period of six months. The variables considered were arrival time, dwell time and schedule adherence at each bus stops and reported that the ANN model outperformed the other models.

Cathey and Dailey (2003) and Dailey et al. (2001) developed an algorithm to predict the bus travel time for Seattle, Washington. They used a combination of historical and AVL data. Three components were used in their algorithm namely tracker, filter and predictor. The Kalman filter was used in the filter component. They reported that their algorithm could predict the bus arrival time at less than 12% error.

Liu et al. (2006) used a hybrid model (SSNNEKF) based on State Space Neural Networks (SSNN) and the Extended Kalman Filter (EKF) for the prediction of bus travel time. The SSNN model require large data set for offline training. Hence, they developed an EKF model to train the SSNN. The SSNNEKF performance was reported to be superior over the individual models.

One of the main differences between bus travel time and other vehicle's travel time is the frequent stoppage of bus at bus stops and the associated delays. Thus, the performance of bus arrival time prediction methods depend on their ability to incorporate the dwell times and other common delays such as delay due to signals, intersections, congestion etc. Thus, the dwell time delay associated to public transport is the major component of its total travel time and need to be considered explicitly. However, there are only limited studies reported which considered this important factor. Some of the studies which considered dwell times explicitly are discussed below.

Shalaby and Farhan (2003) used data collected with AVL and Automatic Passenger Counters (APC) for the prediction of bus arrival time. They developed a Kalman filtering technique (KFT) based approach for bus arrival prediction and compared the results with a historical model, regression model and time lag recurrent neural network model. The dwell time was calculated as a function of the number of passengers alighting and boarding at bus stops. The KFT based approach was reported to outperform the other models.

Liu et al. (2005) developed a macroscopic model for predicting the link travel time in urban corridors incorporating intersection delays. The travel time is divided into link cruising times and intersection delays. They tested their model for 3 different traffic demand patterns for a sample link setup in VISSIM. Lin and Zeng (1999) and Jeong and Rilett (2004) considered dwell time as one of the independent variables in their travel time prediction models.

The literature survey shows that most of the reported studies on bus travel time/arrival time prediction have been developed for homogeneous traffic conditions only. This is because heterogeneous traffic condition is very complex and analysing it may be more challenging.

The algorithm presented in this study uses the data coming from the previous vehicles to predict the travel time of the test vehicle. The prediction accuracy depends on the data coming from the vehicles under consideration. If they are travelling under similar traffic conditions as the test vehicle (which will be true in most of the cases), the prediction will be reasonably accurate. Thus, it can be seen that the algorithm can be used for both heterogeneous and homogeneous traffic conditions. This becomes more useful under heterogeneous traffic conditions since the factors which need to be explicitly considered for an accurate travel time prediction under heterogeneous traffic conditions using most of the other methods are much more than that under homogeneous traffic conditions. Also, many of those factors may not be available in real time due to lack of automated data collection systems. Under such conditions the prediction algorithm should be requiring less data, and less number of parameters, which makes the proposed algorithm much more attractive. As explained already, the algorithm is not explicitly taking into account the heterogeneity and lack of lane discipline. However, they are implicitly taken in to account since the real time data which is coming in captures the characteristics due to these features. This is the main reason for the use of an algorithm which uses the data from the immediate previous buses which will help to capture all the variations that happen due to these traffic characteristics.

Bus arrival prediction systems are implemented in many Western countries. However, implementation of such systems is challenging in countries like India where automated traffic data collection is still in its nascent stage. Most of the existing systems reported from other countries are based on historic data base and are not suitable to the Indian scenario where the data base is just being developed. Also, any methodology based on data pattern may not be the best solution due to the highly stochastic and heterogenic nature of Indian traffic. Other popularly adopted field approaches such as estimating travel time based on the real time distance to the next bus stop and average speed maintained by the vehicle may not work due to the wide variation in driving conditions even within small distances. Thus, the development of a bus arrival time prediction system under Indian conditions is very challenging. The present study addresses many of these problems and develops a prototype of a real time bus arrival prediction system under Indian traffic conditions.

ITS applications are just gaining momentum in India and hence there have been only a few indigenous studies in this area. Ramakrishna et al. (2006) evaluated the performance of a Multiple Linear Regression Model (MLR) and an Artificial Neural Network (ANN) model for bus travel time prediction under heterogeneous Indian traffic conditions. They used GPS data from three consecutive public transport buses from Chennai for the analysis. The results indicated better performance of the ANN model compared to the MLR model. Vanajakshi et al. (2009) predicted the bus arrival time under Indian traffic conditions using GPS data from buses. Their analysis was using a model based algorithm and used Kalman filtering method for the prediction of travel time.

In this study a model based prediction algorithm based on the Kalman filtering technique has been developed for predicting the bus arrival time under heterogeneous traffic conditions. The algorithm was integrated with real time data and a complete system was developed. The display media such as display boards, web sites, kiosks etc were also integrated to the system and the complete prototype is being evaluated.

Since the present study concentrates on the real time field implementation of the system, such existing systems are also reviewed. U.S.A has the most extensive implementation of predictions systems as seen by the number of applications being used. A few popular among them are mentioned below:

2.2. Field Implementations of Bus Arrival Prediction Systems

(a) BusTime™

BusTime™ is a transit technology solution developed by Clever Devices, U.S.A to provide convenient access to real-time bus arrival times and tracking information. It uses GPS technology and sophisticated software to track buses along their route and calculate their arrival time for specific stops and disseminate the information through dynamic message signs, internet, cell phones, PDAs, personal computers and electronic signs. This system requires tools of Intelligent Vehicle Network (IVN), Transit Control Head (TCH) Real-Time Communications, BusTools, BusLink Wireless Communications and Digital Communications Controller for operation and was implemented in Chicago in 2007

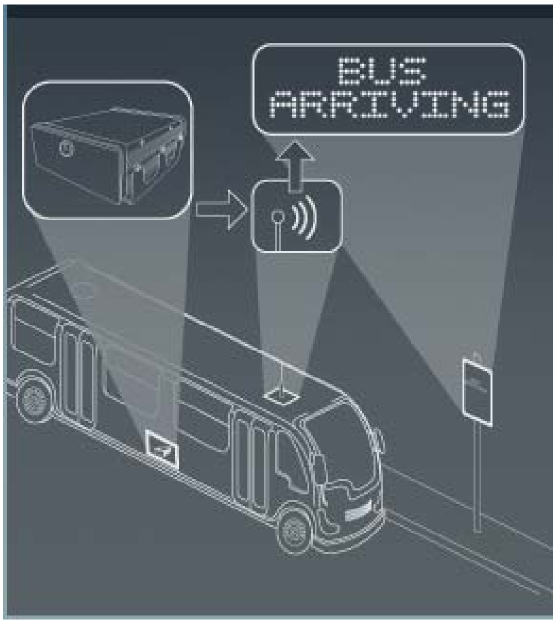


Figure 2: Principle of BusTime Passenger Information System

(b) Estimated Time of Arrival Predictions

Telargo, U.S.A uses ITS principles to provide real-time Estimated Time of Arrivals (ETA) predictions to passengers on bus stops and those traveling in the bus. The ETA is calculated from various parameters such as the current vehicle position, statistical history of trips and the average speed on the assigned route. A"Smart Bus Stop" solution has been implemented by Telargo where buses communicate directly with the bus stops via wireless ZigBee communication for accurate ETA information.

Figure 3: Zigbee Smart Bus Stop solution scheme

The ETA information is shared with the passenger through mobile internet, text messages (SMS), public display boards and voice announcements on bus and/or on stops, passenger information kiosks on stops and web.

c. NextBus

NextBus, California uses GPS data to provide vehicle arrival/departure information and real-time maps to passengers and managers of public transit, shuttles, and trains.

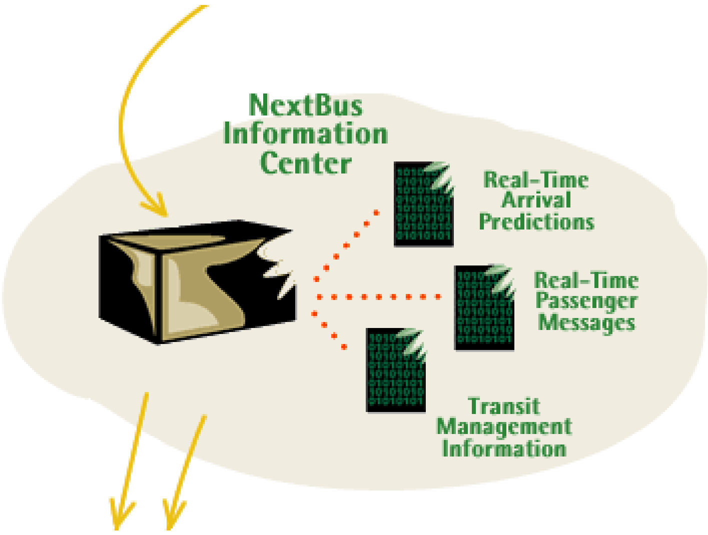


Figure 4: Working Principle of NextBus

The information is disseminated through the internet, cell phones, message signs at bus stops, Palm Pilots, and other Personal Digital Assistants (PDAs). The NextBus real-time passenger information system also provides transit authorities with numerous fleet management tools designed to improve operating efficiency, enhance service, and meet government reporting requirements. This system is widely used and is implemented in the following States in U.S.A.

Arizona	Maryland.	Ontario.
California	Massachusetts.	Oregon.
Colorado	New jersey.	Pennsylvvania.
Columbia.	New mexico.	South Carolina.
Florida	New York.	Texas
Goergia.	North Carolina.	Virginia.
Iowa.	Ohio.	Washington.
Kentucky.	Oklahoma.	Winconsin.

d. Synchromatics

Synchromatics passenger information systems display arrival predictions and real-time bus loads (based on automated passenger counting). Syncromatics also provides interactive maps available through easy-to-navigate web portals. Traverlers can access this information via public portals, mobile phones, shelter signs and call-in phone systems.

e. King County Metro - Seattle

Transit riders in the Seattle area are offered a wide range of traveler information services.:

- MyBus – shows predicted bus arrival times at various time points
- BusView – provides an on-line map of bus locations
- Transit Alters – provides e-mail notifications

King County Metro Internet and wireless application provides details on the real-time bus arrival of Metro buses. The prediction algorithm for this application uses Historical operational data in addition to real-time bus location data to predict the arrival times.

f. OneBusAway (Seattle-Washington)

OneBusAway is a set of transit traveler information tools designed to providing real-time arrival information for Seattle-area bus riders. This tool uses a software system designed by the Intelligent Transportation Systems Research group at the University of Washington (ITS/UW) for predicting bus travel times, based upon access to transit agency schedule and real-time vehicle location data. Estimated departure times are derived from the corresponding travel time estimations. Departure time estimates are made available to the traveling public through web pages as well as through hand-held devices. The new set of tools provided by OneBusAway provided mapping between stop id and real-time arrival was constructed so that users could quickly access information using a stop's posted id. Multiple interfaces are developed to promote greater access to information. Furthermore, an Interactive-Voice-Response (IVR) phone interface, an SMS interface, an iPhone-optimized web interface, and a basic text-only Web interface are also provided to the user to access information using a variety of devices. The standard Web interface allows a user to search for stops by route, street address or map area. It also facilitates the user to enter a stop id to quickly receive arrival information, or to search for a stop using a search tree that narrows results based on the route, destination of travel, and street location of the target stop, allowing stop lookup when the posted stop id is missing or the user is not physically at the stop. Real-time arrival information provided by OneBusAway includes details about the route, destination, and time remaining until departure. In addition to it, a full schedule is also provided for each stop.

g. BusTime (Chicago)

BusTime is Clever Devices' popular transit technology solution that uses GPS technology and sophisticated software to track the buses along their route and calculate their arrival time for specific stops. BusTime allows transit agencies to link with the latest information technologies to communicate spot-on arrival predictions to passengers through the Internet, phones, email and via Dynamic Message Signs (DMS) at bus stops and shelters. Same can be easily accessible for the bus arrival time information by the passengers for their better trip planning in order to avoid the waiting at the bus stops. The real-time bus arrival information helps transit agencies run more efficient operations while increasing information and convenience to passengers and also improving the overall ridership experience. The Chicago Transit Authority (CTA) has seen a measurable increase in ridership since launching the BusTime bus tracker in 2007.

h. TriMet – Public Transportation for the Portland, Oregon, Metro Area

Tri-Met's Transit Tracker is a traveler information system that provides real-time transit information via the Internet and through LED DMDSs at several bus stops. Transit Tracker uses a GPS-based AVL system to determine bus locations and loop sensors. The signs at bus shelters provide real-time arrival information. Bus stops that server more than one route are outfitted with a multiline LED display to list arrival information for three or four buses.

i. Bus Locator - RTD (Denver)

Denver's Regional Transportation District's (RTD's) Bus Locator is an Internet application that provides ETA information for the next two to three bus arrival times based on the route and direction selected. When real-time data is not available, the Internet application displays scheduled arrival times instead. The basic technology used for this system is a text-to-speech system, in which schedules in Extensible

Markup Language (XML) format are translated into voice schedules. The real-time information is taken from the same server that is used to provide arrival information for the Internet application. Using the same real-time data that is provided via Bus Locator and Talk-n-Ride, Denver RTD implemented an application to provide real-time next arrival information to wireless devices, including PDAs and WAP phones. This application is called Mobile-n-Ride. Once the user connects to the Internet web page via mobile device, traveler enters a route number, a direction, and a stop. After all the information is provided, a message containing the ETA for the next two or three vehicles is sent to the mobile device using the same operating system as the device that requested the information.

j. California

This system utilizes the loop-transponder technology as an Automatic Vehicle Locator (AVL) to detect and identify bus locations. It has been deployed in the Wishire and Ventura Boulevards in California, where more than 150 Metro Rapid buses have been equipped with system transponders and can be detected when passing through any of the 331 loop detectors throughout the two corridors. In this system, the TPM [Transit Priority Manager] first tracks every data that is generated when a bus traverses through a detector in the system. This consists of two real-time lists—the Hot-List (HL) and the Run-List (RL) objects. The HL tracks movement of every bus operating along a TPM corridor, which contains the bus attributes, position, and running status. The RL stores the detail time-point table and detector attributes, including bus scheduled arrival time-points, and actual arrival time-points. Bus travel time is a function of distance and prevailing bus speed.

The TPM employs a Dynamic Bus Schedule Table technique (DBST) using an innovative algorithm approach called the Time Point Propagation (TPP) method, which dynamically builds the Schedule Arrival Time Point table with a runtime information from the prior bus arrival time for the same locations plus the active headway value of the current bus.The actual arrival time-point is also used for the prediction of Estimated Time of Arrival (ETA) of the next bus. ETA is calculated based on the previous bus travel time under the assumption that the current bus would experience the same or similar traffic conditions in the same segment of the corridor.

The predicted bus arrival information is then transmitted through Cellular Digital Packet Data (CDPD) services to LED display signs at major bus stations. According to a field survey, the accuracy of the bus arrival information is relatively high.

k. New Jersey Transit Corporation

The New Jersey Transit Corporation (NJT) offers a variety of Web-based features, including automated itinerary trip planning and electronic notifications accessible via their Web site. MyTransit, the agency's advanced registration feature and a form of CRM, allows customers to develop their own customized transit schedule for all NJ TRANSIT Rail, Bus and Light Rail services. Thus, customers can receive free instant transit service alerts about delays of 30 minutes or more. In addition, they can quickly view and print schedules and also participate in quick surveys which on the other hand help NJ TRANSIT address customers travel needs.

Brooklin MTA Bus Sevices, ARLINGTON transit system, Santa Clara Bus, WAHOO BUS (Virginia) and Chicago CTA bus tracker are other applications where APTS has been implemented.

AVL technology has been used in other Western countries as well for APTS implementations. For example, the iBus system in London is based on an AVL package, which has been scaled, extended and customised to meet TfL's strategic and operational business needs. The System is complex technically and widely distributed, with software and computer equipment residing in each vehicle, as well as in operators' bus garages, and at TfL's two Bus Network Control Centres in Central and North-East London. The System is essentially made up of four inter-connected components (Figure 6):

- An "on-board" unit mounted in each bus (Item L in Figure 6);
- A "data server" (or large personal computer) at individual bus garages (Item O);
- a central "system server" (or powerful computer) located remotely (Item K), which holds the master records of bus routes, their timing points, operating frequencies, as well as, for example, the locations of detectors for giving bus priority at signals; and
- Local and user "databases" (Items A and B), together with a "core system", which are used for bus network management and management reporting.

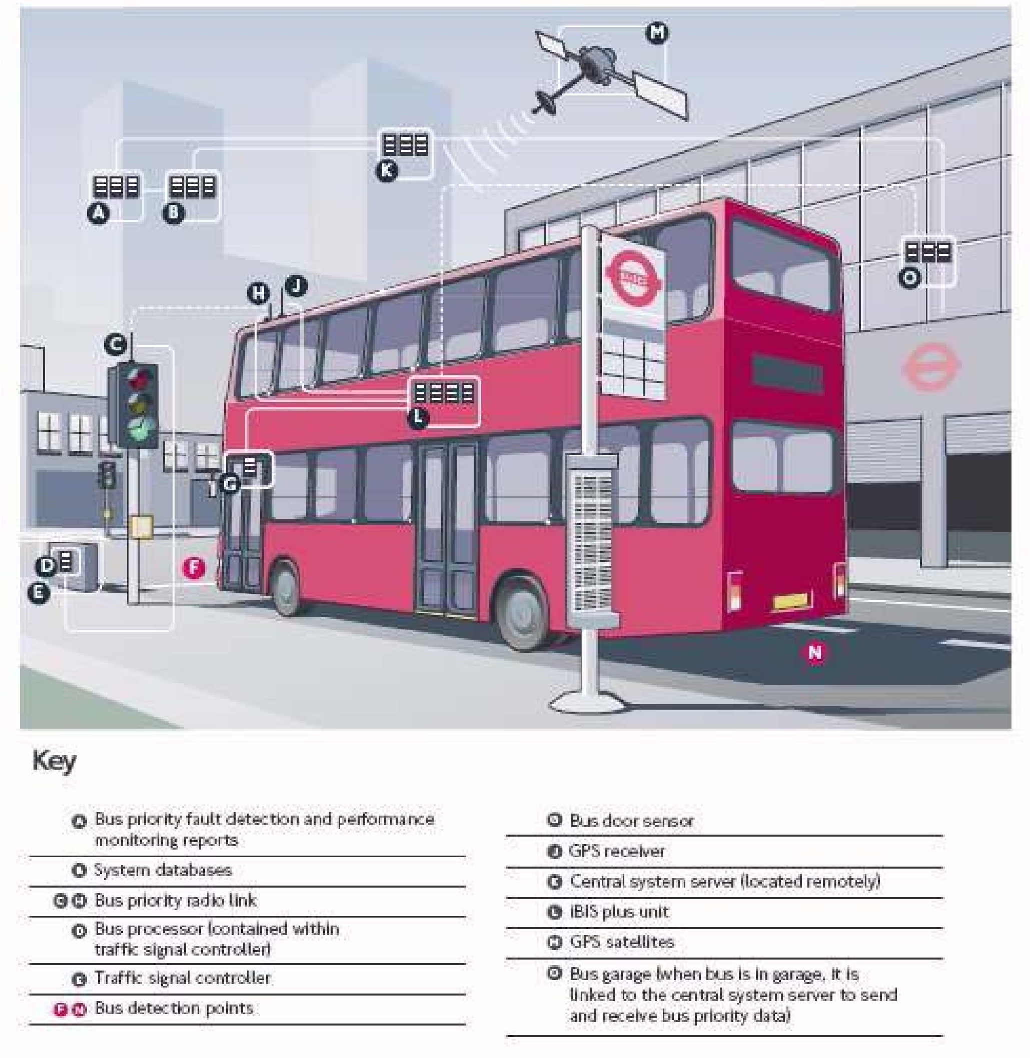


Figure 5: The iBus system in London

iBus provides real-time route, destination and next-stop signage (and associated voice announcements) for passengers on board. It is also being integrated with the "Countdown" information displays at bus stops (Transport for London, 2009), to provide improved real-time predictions of vehicle arrivals (see Figure 5a), and (from 2010) passengers can obtain details of upcoming bus arrivals at a bus stop by text message through their mobile phones

In Alberta, Canada, nextbus system is used for bus tracking. The service is called roam bus service system. Each vehicle is fitted with GPS receiver which transmits the required data to a central location, where a computer running proprietary software calculates the projected arrival times. This information is made available on nextbus website and through electronic signs at select bus stops. In Alberta the information is displayed at 10 selects stations. The bus services are provided at each bus stop for every 40 minutes.

A few transportation agencies in India have already tried to implement the bus arrival prediction systems in the country. Details of those that are currently in operation are mentioned below:

a. DIMTS, Delhi – Delhi Integrated Multi-modal Transit System

This project was introduced in Delhi to improve the efficiency, reliability and punctuality of the bus operations in Delhi and ensure optimal deployment of the fleet which would eventually lead to higher commuter satisfaction and improved level of confidence in bus services. This was implemented by the Transport Department, Government of National Capital Territory of Delhi (GNCTD) and used the Global Positioning System (GPS) enabled Automatic Vehicle Location (AVL) for this project.

The GPS enabled AVL allows real-time tracking of bus movement and provides information on its location. This information is then used along with other details such as the average speed of the bus, the route followed etc to provide the passengers waiting at the bus stops with the expected arrival time of their bus. This information is displayed on LED boards installed at the bus stops as well as inside the buses.

b. MTC, Chennai – Metropolitan Transport Corporation, Chennai

Chennai was one of the first cities to implement a bus passenger information system, in the country. The Metropolitan Transportation Corporation (MTC) introduced GPS tracking of its city buses in 2008. Initially a pilot project on 55 buses was launched. Later this was followed up by expanding the service to over 500 buses in the city. According to MTC, the state run passenger transport buses in the city are fitted with GPS/GPRS units that enable passengers to track buses on their mobile phones, bus stops and on the internet.

The World Bank provided a grant of Rs 3 crore to Pallavan Transportation Consultancy Services (PTCS) for this initiative. PTCS is the nodal agency for monitoring the project. A consortium of IIIT Bangalore, Ashok Leyland, Lattice Bridge, Siemens Information Systems, and Pallavan Transport Consultancy Services was formed to implement this system for the city buses in Chennai.

c. ICTSL, Indore – Indore City Transport Service Limited

Indore City Transport Service (ICTSL) introduced the GPS based Online Bus Tracking System (OBTS) and LED system for display of information to offer better facilities to commuters in the city in 2007. ICTSL has a control room for OBTS where every bus was fitted with a GPS-based tracking device with online data transfer facility. With this ICTSL plans to flash the estimated time of arrival on display screens at 50 bus stops to help the passengers waiting for the buses the arrival timings and other information related to the buses.

The main objective of ICTSL was to create specialized and effective regulatory agency to monitor cost effective and good public transport services. This will also help in knowing schedule and itinerary adherence, exact kilometer travelled by bus, punctuality and improvement on driving pattern, unauthorized and unscheduled stoppages.

<u>d. BMTC, Bangalore – Bangalore Métropolitain Transport Corporation</u>

The BMTC implemented the off-line vehicle tracking system using GPS in March 1999 in technical collaboration with Bharat Electronics Limited (BEL) to monitor trip operations of buses hired from private operators and automatically calculating hire charges according to the kilometer covered on a daily basis. GPS units were installed on 200 buses. However, data collection turned out to be a cumbersome process, as it required personnel to physically go to each bus and download the data. Subsequently, they switched over to on-line vehicle tracking system designed to track the movement of a bus via satellite through the radio frequency signals from the on-bus transmitter unit. The captured data was to be sent to the control centre using GSM technology at an interval of 10 seconds.

However, BMTC reported that the GPS tracking facility facing limitations such as loss of connectivity due to selective availability of satellite constellation; buses not being in the direct and unobstructed line of sight with the satellite as large parts of the travel paths had dense tree covers, flyovers and bridges; between a cluster of high-rise buildings or while buses were parked under a shelter. Moreover, the tracked data were not linked to schedules and hence monitoring of deviations in operations could not be done on a real-time basis.

All of the other implementations reported limitations similar to the ones reported by BMTC, Bangalore. Moreover, all of them have concentrated on demonstration of feasibility rather than optimizing the quality of the service being provided. The accuracy of arrival time prediction is poor, perhaps because of deficiencies in algorithms used for prediction. In most cases, the details of the prediction algorithms are not available since they were implemented by an external agency. Many of these implementations used methodologies developed for western countries that have homogenous traffic, which may not be suitable in the heterogeneous Indian traffic conditions.

2.5 Summary

The above literature review of the models and algorithms for bus arrival time prediction shows that many models are based on historical arrival patterns and/or other variables correlated with the arrival time. The variables used include historical arrival time (or travel time), schedule observance, time-of-day, day-of-week, dwell time, number of stops along the route, distance between adjacent stops and the road–network condition. The methodology used for collection and transmission of such data on such variables has been to use the emerging technologies of wireless communication, AVL (e.g. GPS), APC, and other sensing technologies.

Historical based models assume that the conditions of traffic do not change much, which may not be true when the there is a switch from off-peak to peak and vice-versa. These were mainly used in areas where congestion is a minimum because the model assumed traffic patterns are similar. However, it could be argued that it is also be possible to observe such patterns in areas where congestion is severe. This can be found out from extensive historical data analysis by looking into the distribution of travel time over time of day or day of week and so on. In areas where there are stable demand and similar traffic pattern, historical based models are able to give satisfactory bus arrival time information, so there is no need to go for complex prediction models. Machine learning techniques have been shown to outperform other methods in cases where enough data base is available. KF models can be applied on-line while the bus trip is in progress due to its simplicity in calculation and is one of the most popularly recommended for real time applications.

It can also be seen that most of the above mentioned studies were performed in homogenous and lane-disciplined traffic conditions. The heterogenic traffic pattern and wide variations in driving conditions make existing Western methodology unsuitable for Indian conditions. A variety of vehicles – two, three and four wheelers, in addition to a large pedestrian population, share the Indian urban road. This heterogeneity, coupled with poor lane discipline makes travel time prediction more challenging than can be handled by the reported methods. There is, therefore, a need for models that can capture the stochastic behavior of Indian traffic with little data requirement.

There have been few reports on bus arrival prediction under heterogeneous conditions (Patnaik *et al.* 2004, Ramakrishna *et al.* 2006, Vanajakshi *et al.* 2008) and practically none on field implementation. The review of implementations shows that very few are available under Indian scenario and none of them are carried out in a systematic way and lacks theoretical support for the prediction algorithms. For example: MTC, Chennai uses a packaged prediction system developed elsewhere without the flexibility of making changes to suit the Indian conditions. Most of these implementations use methodology based on simple assumption of speed being constant and calculates travel time based on that. Mostly offline information based on historical average or average performance of the present vehicle is used. However, the Indian traffic scenario is very dynamic and use of offline data may not be able to capture the real time variations in the data.

The present study will be an attempt in this direction to develop a real time bus arrival time prediction field implementable system under Indian traffic conditions based on the work reported by Vanajakshi et al. (2008). This project aims at testing the algorithm and modifying and extending it and to develop a prototype for real time bus arrival time information system.

III. STUDY ROUTE, DATA COLLECTION AND QUALITY CONTROL

The present study is aimed at developing a real-time bus arrival time prediction field implementable system under Indian traffic conditions. This section describes the selected study route, techniques used for data collection, data collection procedure, data storage, data filtering technique and the data analysis carried out.

3.1 Data Collection

3.1.1. Global Positioning System

Data for this project is collected using GPS units installed in the buses. The Global Positioning System (GPS) is a satellite-based navigation system consisting of a network of 24 satellites placed into orbit by the U.S. Department of Defense. GPS satellites circle the earth twice a day in a very precise orbit and transmit signal information to earth. GPS receivers take this information and use triangulation to calculate the receiver's exact location. The GPS receiver essentially compares the time a signal was transmitted by a satellite with the time it was received. The time difference tells the GPS receiver how far away the satellite is. With distance measurements from 3 or more satellites, the receiver can determine the vehicle's position. There are no subscription fees or setup charges to use GPS.

3.1.2 General Packet Radio Service

General packet radio service (GPRS) is a packet oriented mobile data service on the 2G cellular communication systems of Global System for Mobile communications (GSM). GPRS is a method of enhancing 2G mobile communication devices to enable them to send and receive data more rapidly. With a GPRS connection, the device is "always on" and can transfer data immediately, and at higher speeds: typically 32 - 48 kbps.

In the present study, the selected buses are equipped with GPS/GPRS modules. These devices are powered by the bus' power supply unit and have 2 main functional parts, GPS receiver and GPRS module. The GPS receiver receives signals from the GPS satellite and dynamically calculates the bus' current location. This data is sent to the central server by the GPRS module. Each module has a GPRS activated Systematic Identification Module (SIM) card installed.

Data from these GPS/GPRS units installed in the MTC buses are sent to a dedicated server in IIT with a fixed global IP once every 5 sec. This data packet when received by the server is added to an SQL database as well as appended to a .csv (comma separated value) file. Data from each bus for the day is stored in a different file. The file names are of the format 'DDMON-BusID' eg: '10Nov-iit07', and are stored in a separate folder for every year. Files are archived daily and new files are created to store the incoming data everyday at 00:00:00. The real-time data received in the server is immediately used in order to make a real-time prediction of the arrival time of the bus.

3.2 Study Route

The 5C and 19B bus routes –with the following characteristics were selected for the final implementation of the project. Figure 6 below shows these two routes.

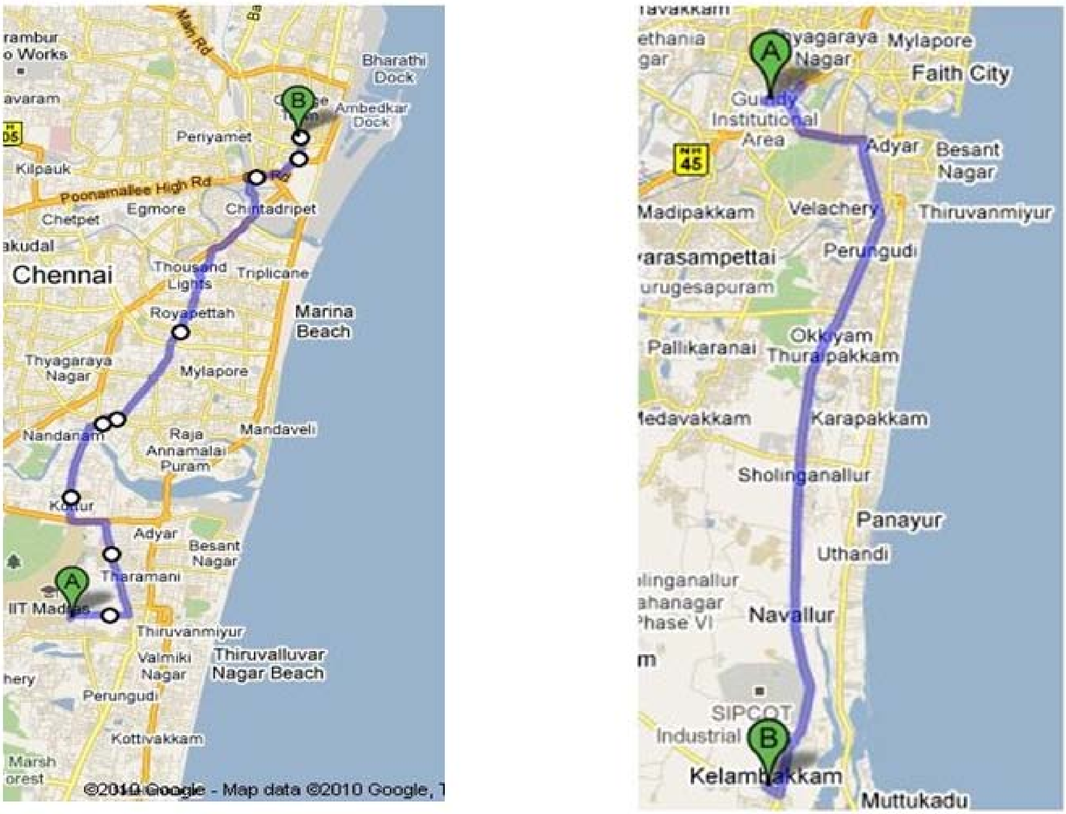


Figure 6: (a) 5C Bus route (b) 19B Bus Route

The characteristics of these selected routes are:

1) The 5C route is between Taramani and Parrys and is about 16.8 kilometers long; it takes about 60 to 70 minutes for the bus to travel from one terminus to the other in each direction.
2) 19B route is between Kelambakkam and Broadway and is about 31.5 kilometers long and it takes about 90-100 minutes to travel from one terminus to the other.
3) There are 7 buses in 5C route with headway of around 30 minutes and there are 17 buses in the 19B route with headway of around 15 minutes.
4) There are 10 bus-stops along 5C route and 17 bus-stops along 19B route.

Data from route number 21 L and 21 G were used for offline testing of the algorithm by manually collecting data.

3.3. Data Collection and Storage

The raw data from the GPS units are obtained approximately every 10 seconds. The raw data contains information on 'deviceID', 'Clock time', 'latitude' and 'longitude' in the following format as shown in Table 1.

Table 1: Sample of raw data received at the server on 1st Feb, 2011.

Device ID	Latitude	Longitude	Clock Time
iit04	13.029	80.24557	10:15:38
iit04	13.029	80.24557	10:15:43
iit04	13.029	80.24557	10:15:48
iit04	13.029	80.24557	10:15:53
iit04	13.029	80.24557	10:15:58
iit04	13.029	80.24557	10:16:03
iit04	13.02906	80.2456	10:16:08
iit04	13.02936	80.2456	10:16:13
iit04	13.02958	80.24585	10:16:18
iit04	13.02952	80.24623	10:16:23
iit04	13.02948	80.24651	10:16:28
iit04	13.02943	80.24675	10:16:33
iit04	13.02938	80.24704	10:16:38
iit04	13.02935	80.24723	10:16:43
iit04	13.0293	80.24752	10:16:48
iit04	13.02924	80.24794	10:16:53
iit04	13.02921	80.24824	10:16:58
iit04	13.02916	80.24867	10:17:03
iit04	13.02902	80.24921	10:17:08

This data received from each device is stored in the server under a different file name of the format 'DeviceID_Date'. All the data files created on the same day are stored under the file name 'Date' for proper organization of the data. A backup of these files is created every day. From this data, the information of interest in the present study namely travel time can be easily estimated as discussed in the section below.

3.4. Data Filtration

All the data received at the server are not necessary for the bus Arrival Time Prediction (ATP). Hence, this raw data must be filtered in order to input only the required data into the algorithms. The unfiltered data consists of location details of the bus for the entire duration for which it is being monitored. For example, there is no need to track data sent by the bus while it is parked in the bus depot at night, or while it is kept waiting at the terminus during lunch etc. Such data must be filtered out and only the essential data has to be fed into the algorithm in order to get the required results. A program is developed for automating this filtering process. The filtered data can be used for the application. In the present study, the selected approach is a model based prediction algorithm that uses Kalman filtering technique. In this algorithm, data from two previous buses, PV1 and PV2 are used as input to predict the arrival time of the current bus, which is called test vehicle, TV. Thus, the real time data received from three consecutive buses are used as detailed below.

The first step in this process is assigning unique identification numbers to buses and bus-stops. A route ID is assigned to the buses to identify the route in which the bus is plying. For every route there are two route IDs, one for each direction. For assigning the route IDs, the current latitude, and longitude of the bus are compared to the latitude and longitude of the terminus. In order to accomplish this, latitude and longitude details of all termini are collected and stored in the data base. Once the bus is within a 50 m radius of the terminus, it is assigned a 'routeID'. A trip is assumed to start when the bus enteres this region and its velocity is greater than 5 kmph. The velocity is then calculated from the known latitude and longitude using Haversine's Formula. Since buses make multiple trips within a day, a new trip is created every time it departes from the terminus in that specific route and is assigned a new 'tripID'. There is a possibility that a 'routeID' is assigned to the bus during its return to the terminus during the previous trip in the reverse direction. This possibility is taken into account and data from the bus is discarded as long as the bus is waiting at the terminus. New 'routeID' and 'tripID' are assigned to the bus only when it starts on a new trip. From this point onwards, the incoming data from the bus is stored in the file 'tripID_routeID' until it reaches its destination terminus, where the 'routeID' and 'tripID' changes. This process is carried out throughout the day and the data is segregated into different trips during that day. These trip data files are stored under the folder name 'routeID_date'. For every route there are two routeIDs, one for each direction. For example, for 5C route, '333' denotes that the bus is going from Broadway to Taramani. Table 2 shows a sample of the data filtered out from the above raw data.

Table 2 : Sample data after filtering

Route ID	Device ID	Latitude	Longitude	Clock Time
333	iit04	13.029	80.24557	10:15:38
333	iit04	13.029	80.24557	10:15:43
333	iit04	13.029	80.24557	10:15:48
333	iit04	13.029	80.24557	10:15:53
333	iit04	13.029	80.24557	10:15:58
333	iit04	13.029	80.24557	10:16:03
333	iit04	13.02906	80.2456	10:16:08
333	iit04	13.02936	80.2456	10:16:13
333	iit04	13.02958	80.24585	10:16:18
333	iit04	13.02952	80.24623	10:16:23
333	iit04	13.02948	80.24651	10:16:28
333	iit04	13.02943	80.24675	10:16:33
333	iit04	13.02938	80.24704	10:16:38
333	iit04	13.02935	80.24723	10:16:43
333	iit04	13.0293	80.24752	10:16:48
333	iit04	13.02924	80.24794	10:16:53
333	iit04	13.02921	80.24824	10:16:58
333	iit04	13.02916	80.24867	10:17:03
333	iit04	13.02902	80.24921	10:17:08
333	iit04	13.02896	80.24951	10:17:13
333	iit04	13.02894	80.24963	10:17:18
333	iit04	13.0291	80.24992	10:17:23
333	iit04	13.0295	80.25025	10:17:28

Since the bus arrival predictions have to be implemented in real-time, the above data filtration has to be an automated process. In all the earlier studies this was carried out manually, and hence the process used offline data. Thus, the real-time automation of the data filtration technique is a key deliverable of this study.

3.5. Data Analysis

Further analysis of the data for bus arrival time prediction uses the above filtered dataset stored in the database with the following characteristics using which it can be uniquely identified:

1. tripID
2. routeID
3. date

Each dataset is stored under the file name 'tripID_routeID'. These files are stored in folders named 'routeId_date' under the 'date' folder. Everyday a new 'date' folder is created at midnight. As mentioned earlier, the prediction algorithm used in this study need 3 input files to process the data namely 'PV1' dataset, 'PV2' dataset and the 'TV' dataset. So, once a bus (TV) starts from a terminus, it is assigned a route and its arrival prediction is started, provided two previous buses are already in the route or have covered the route. Thus, the corresponding 'PV1' and 'PV2' datasets for the current TV trip must be identified as discussed below.

PV1 and PV2 for a selected TV should be from the same date folder, and have the same routeID. Thus, if the tripID of TV is '*t*' then PV1 and PV2 are selected such that their tripIDs are '*t-1*' and '*t-2*' respectively. Since the arrival of the TV at the following bus stop must be predicted, it is necessary that both PV1 and PV2 had already crossed this bus-stop. Once the three datasets PV1, PV2 and TV are identified, they are fed into the algorithm.

There may be many buses plying along a particular route at a given point of time. Of these, the arrival time of the one expected to arrive first at the following bus-stop must be predicted. This choice is made by considering the index number '*i*' of the bus. Index numbers are provided for every bus-stop in a selected route and for every bus plying along that route. The 'Index Number *I*' of a bus-stop is defined as the position number of the bus-stop along that route. Thus, it is possible for a single bus-stop to have different values of *I* along different routes. The 'Index Number *i*' of a bus is defined as the number of bus-stops that the bus has crossed in that route. For a bus plying between n^{th} bus-stop and $(n+1)^{th}$ bus-stop along a particular route, its *i* value is equal to n. The bus for which bus arrival prediction must be performed along this route is identified by comparing the 'Index Number *i*' of different buses plying along that route. The bus with the highest current 'Index Number *i*' ($< I$) is chosen and a prediction is made for this particular bus. At any instant there could be multiple buses with the same index number '*i*'. In such cases the bus whose index number changes first is used, as it is expected to arrive earlier.

IV. PREDICTION METHODLOGY

The algorithms used for travel time prediction in this study are based on dynamic traffic models. To begin with, a prediction model was developed characterizing the evolution of travel time over space. In this model, the travel time in a particular subsection was assumed to be affected by the travel time in the previous subsection.

In this approach, the entire route was discretized into smaller subsections, and the test vehicle travel time in the current subsection was obtained by using the travel time data of the two previous vehicles (denoted as PV1 and PV2). Here, the inputs were selected based on the availability rather than suitability. This was mainly due to lack of availability of historic data. The details of the spatially discretized model are presented in section 4.1, 4.2, 4.3 and 4.4.

The next stage of the study addressed this limitation by using a database of travel time and developing a mathematical model for characterizing the temporal evolution of travel time by selecting significant inputs for the model based on statistical analysis. The discretization over time is expected to reflect the effect of roadway characteristics such as carriageway width, signalized intersections, etc., in a given subsection to capture the uncertainties and variability in travel time. The details about input selection and the time discretized model are presented in Section 4.5.

4.1 Space Discretisation Prediction Method

The base algorithm (Vanajakshi et al. 2008) predicts the arrival time/travel time of the vehicle of interest, called as Test Vehicle (TV), using data from the two previous buses traveling in the same route, which are called as Probe Vehicle1 (PV1), and Probe Vehicle 2 (PV2). To start with, the route under consideration is divided into *N* subsections and it is assumed that the evolution of travel time of two consecutive subsections is related as

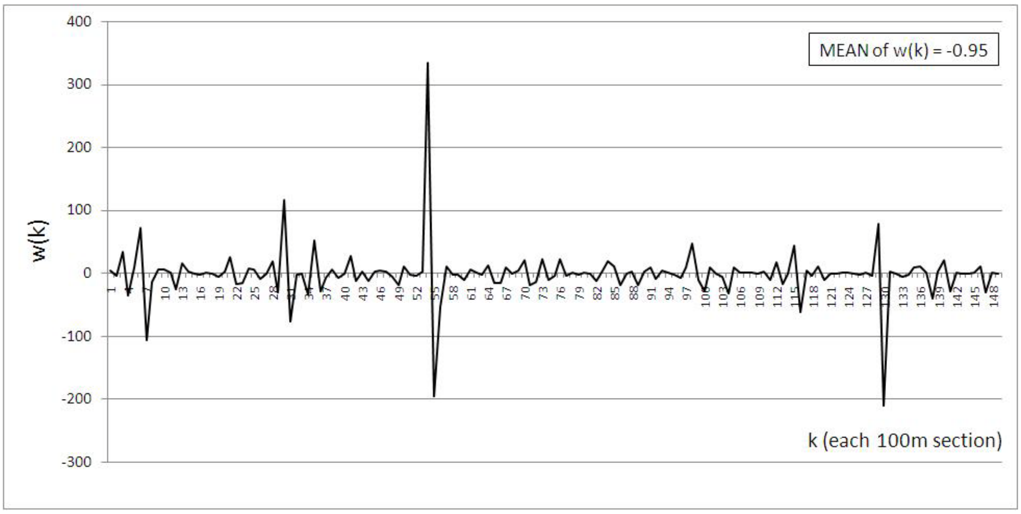
$$x(k+1) = a(k)x(k) + w(k), \qquad (4.1)$$

where, $x(k)$ is the travel time for covering the k^{th} subsection, $a(k)$ is a parameter which relates the travel time taken in the $(k+1)^{th}$ subsection to the travel time taken in the k^{th} subsection and $w(k)$ is the process disturbance associated with the k^{th} subsection. The measurement process is assumes to be given by

$$z(k) = x(k) + v(k), \qquad (4.2)$$

where, $z(k)$ is the measured travel time of the k^{th} subsection and $v(k)$ is the measurement noise. It is assumed that both $w(k)$ and $v(k)$ are zero mean Gaussian white noise signals. This assumption of w(k) and v(k) as zero mean W-N Gaussian signal was checked using real world data. The equations for calculating w(k) and v(k) can be obtained by rearranging state and measurement equations. The $w(k)$ and $v(k)$ were tested for zero mean W-N Gaussian using 105 actual bus trips data spanned over 5 days. In each trip, $w(k)$ and $v(k)$ were calculated for each of the 100m sections of the 15 km stretch. In order to check whether it is W-N, i.e., the value of w or v at any k is independent of the value of w or v at any other k, the concept of autocorrelation is used. For a W-N process, with mean zero and standard error $\frac{1}{\sqrt{n}}$, if approximately 95% of sample autocorrelations fall between the bounds $\pm\frac{1.96}{\sqrt{n}}$, where '*n*' is the sample size, then the series is considered as a stationary process, which is uncorrelated, i.e. the series is white-noise (W-N) (*1, 2*). A sample plot for w(k) and v(k) is shown in Figure 7a and 7b respectively. It is clearly visible from Figure 7 that, the mean value is close to zero for both w(k) and v(k). For w(k), the mean is found to be -0.95 and for v(k), it is -0.01. The Auto Correlation Function (ACF) calculated using w(k) and v(k) is shown in Figures 8 a and b respectively. The lag is taken as 150 while calculating ACF using "R" software. From Figure 8, it can be seen that the ACF at lag '0' is 1, because the correlation of sample on itself will be maximum. For a white-noise (stationary process), with mean zero and standard error $\frac{1}{\sqrt{n}}$, we would expect approximately 95% of sample autocorrelations should fall between the bounds $\pm\frac{1.96}{\sqrt{n}}$, where 'n' is the sample size. For w(k), '*n*' is 149 and so the bounds are ± 0.1606 as shown in Figure 2(a) in dotted lines. We can say that, roughly 5% of 150 (lag) that is, a maximum of 7 ACF's can fall outside the bounds of ± 0.1606, to confirm that the series is W-N. As we can see from Figure 8 (a) that, only 5 ACF's were falling outside the bounds, i.e., 95% of sample autocorrelations were falling within the bounds and so we can confirm that, the series is W-N. Similarly, for v(k) also, only 2 ACF's were falling outside the bounds thus confirming that the series is W-N.

For checking the normal distribution of $w(k)$ and $v(k)$, the statistical measure 'skewness' has been employed. According to (3), if the skewness lies between -1 and +1, the sample data follows a normal distribution. Hence, the skewness calculated using $w(k)$ and $v(k)$ has been checked for all the 105 trips considered. The skewness values for the w(k) and v(k) are found to be -0.058 and 0.063 respectively and are within the acceptable range of -1 to +1 for Gaussian or normal distribution assumption.


MEAN of w(k) = -0.95
w(k)
k (each 100m section)

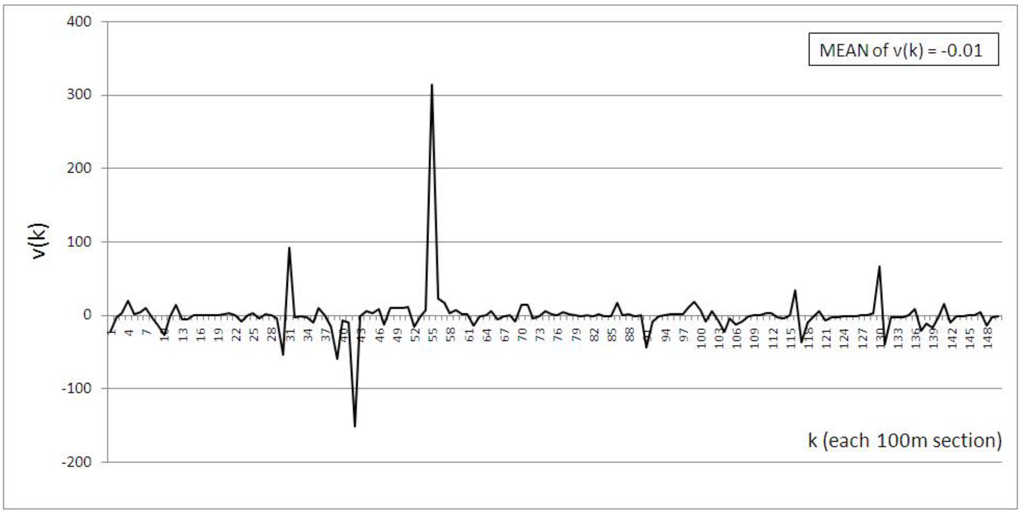
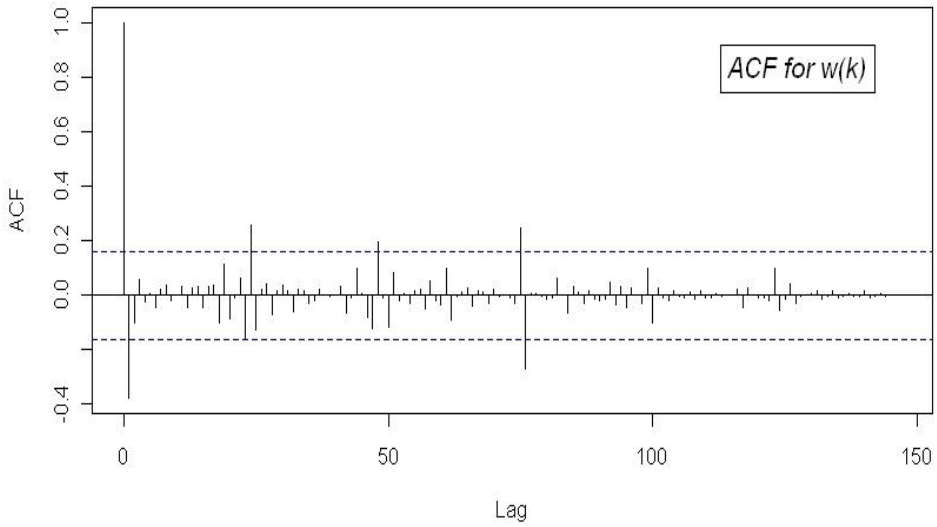

MEAN of v(k) = -0.01
v(k)
k (each 100m section)

Figure 7 (a) & (b). Sample plot of w(k) and v(k)



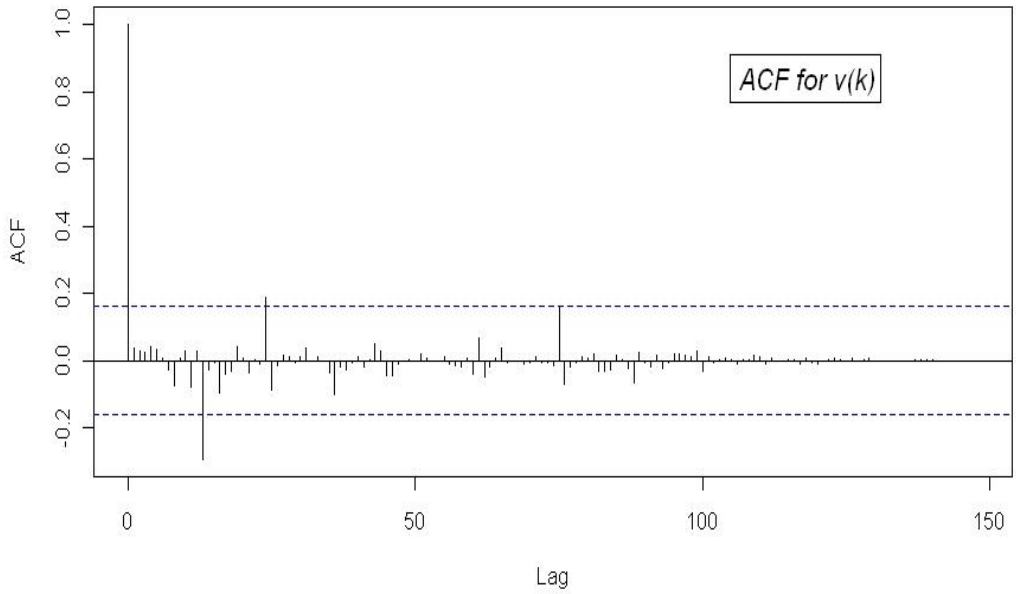


Figure 8 (a) & (b). ACF plot for w(k) and v(k)

The prediction scheme consists of the following steps:

1) The parameter $a(k)$ is calculated for each subsection using the travel time data for PV1 through

$$a(k) = x_{pv1}(k+1)/x_{pv1}(k), \tag{4.3}$$

where, $x_{pv1}(k)$ is the travel time taken by PV1 to cover the k^{th} subsection.

2) The Kalman filter algorithm is then used to predict the travel time of the test vehicle.

Kalman filter (Kalman, 1960) is a popular tool for recursive estimation of variables that characterise a system (Chien and Kuchipudi, 2003; Yang, 2005; Chu et al., 2005).

It is a model based estimation scheme that takes into account the stochastic properties of the process disturbance and the measurement noise. The Kalman filter not only works well in practice, but is theoretically attractive because it can be shown that of all possible filters, it is the one that minimises the variance of the estimation error (Simon, 2001). It is a recursive algorithm that first predicts the state of the system and then corrects the same using measurements. The set of mathematical equations which carry out the above two steps can be slotted into two groups.

- Time-update equations (predictor)
- Measurement-update equations (corrector)

In the time-update equations the *apriori* estimate is calculated using estimates of the current state and error co-variances to project forward in time. In the measurement-update equations an improved aposteriori estimate is obtained by incorporating new measurements from the system. Thus, in Kalman filter initial parameters are predicted and they are adjusted with every new measurements, thus making it appropriate for problems such as travel time prediction. Figure 9 shows the schematics of the Kalman filtering algorithm.

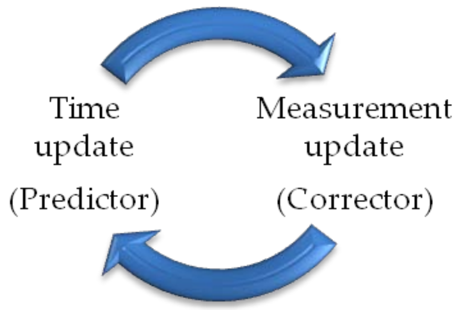


Figure 9: Kalman Filter Cycle

The Kalman filter, being recursive, does not require all previous data to be stored and reprocessed every time a new measurement is taken. This is important for quick predictions of parameters. The Kalman filter is used for estimation and prediction when the governing equations of the system are linear. Let us assume that the linear governing equations of the system can be written as

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k, \quad (4.3a)$$

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k, \quad (4.3b)$$

where $\mathbf{x}_k$, $\mathbf{u}_k$ and $\mathbf{z}_k$ denote respectively the state variables, the input and the output at the k^{th} instant of time and $\mathbf{w}_k$ and $\mathbf{v}_k$ denote respectively the process disturbance and the measurement noise. The process disturbance covariance is denoted by **Q** and the measurement noise covariance by **R**. The matrix **A** relates the state at time step k

to the state at $k+1$, the matrix $\mathbf{B}$ relates the input $\mathbf{u}$ to the state $\mathbf{x}$ and the matrix $\mathbf{H}$ relates the state to the measurement $\mathbf{z}$. Let $\hat{\mathbf{x}}_k^-$ and $\hat{\mathbf{x}}_k^+$ denote respectively the a priori estimate and the a posteriori estimate of the state variables at the k^{th} instant of time. Similarly let $\mathbf{P}_k^-$ and $\mathbf{P}_k^+$ denote respectively the a priori and the a posteriori error covariance at the k^{th} instant of time. The following recursive algorithm is then used to obtain the estimate of the state variables:

1. The a priori estimate in the $(k+1)^{th}$ interval of time is obtained through
$$\hat{\mathbf{x}}_{k+1}^- = \mathbf{A}\hat{\mathbf{x}}_k^+ + \mathbf{B}\mathbf{u}_k. \tag{4.3c}$$
2. The a priori error covariance in the $(k+1)^{th}$ interval of time is obtained through
$$\mathbf{P}_{k+1}^- = \mathbf{A}\mathbf{P}_k^+\mathbf{A}^T + \mathbf{Q}. \tag{4.3d}$$
3. The Kalman gain $\mathbf{K}_{k+1}$ is calculated through
$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1}^-\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_{k+1}^-\mathbf{H}^T + \mathbf{R}\right)^{-1}. \tag{4.3e}$$
4. Then the a posteriori state estimate is calculated through
$$\mathbf{x}_{k+1}^+ = \mathbf{x}_{k+1}^- + \mathbf{K}_{k+1}\left(\mathbf{z}_{k+1} - \mathbf{H}\mathbf{x}_{k+1}^-\right). \tag{4.3f}$$
5. Finally, the a posteriori error covariance is obtained through
$$\mathbf{P}_{k+1}^+ = \left(1 - \mathbf{K}_{k+1}\mathbf{H}\right)\mathbf{P}_{k+1}^-. \tag{4.3g}$$

The five steps are repeated for each time interval to obtain the estimate of the state variables when the governing equations are linear.

This base method assumes the process disturbance co-variance Q and measurement co-variance *R* to be constant throughout the simulation; however the values of *Q* and *R* can be made to vary to reflect the accuracy and reliability of data obtained from PV1 and PV2. The simple Adaptive Prediction Method presented in the next section is a method to take into account this variation in *Q* and *R.*

4.2. Adaptive Prediction Method

The simple Adaptive Prediction Method presented in this subsection computes the value of Q and *R* at every step by utilizing the travel time information from previous subsections. For this, equation (4.1) and (4.2) can be rearranged as

$$w(k) = x_{tv}(k+1) - a(k)x_{tv}(k), \tag{4.4}$$

$$v(k) = x_{tv}(k) - x_{pv2}(k), \tag{4.5}$$

where, $x_{tv}(k)$ is the travel time for the test vehicle in the k^{th} subsection

Using Equations (4.4) and (4.5) one may compute *Q(k)* and *R(k)* as:

$$Q(k)=\frac{1}{S}\sum_{i=k=S}^{k}\left(w(i)-\overline{w}(k)\right)^{2}, \tag{4.6}$$

where,

$$\overline{w}(k)=\frac{\sum_{i=k-S}^{S}w(i)}{S}, \tag{4.7}$$

and S is a predetermined constant number.

Similarly, *R(k)* can be estimated using

$$R(k)=\frac{1}{S}\sum_{i=k=S}^{k}\left(v(i)-\overline{v}(k)\right)^{2}, \tag{4.8}$$

4.3. Enhanced Prediction Method

Another modification that was explored in the present study for better prediction performance was the use of weights for travel time values. This is called as Enhanced Base Prediction Method in this study. The method takes a weighted average of the travel times of PV1 and PV2 as the measured travel time for that subsection. The weights are estimated based on the relative travel time values of PV1 and PV2 as detailed below.

We define $v(k)$ as,

$$v(k)=x_{pv1}(k)-x_{pv2}(k), \tag{4.9}$$

If $v > d$, then subsection k belongs to Group 1
If $-d \leq v \leq d$, then subsection k belongs to Group 2
If $v < -d$, then subsection k belongs to Group 3
where, d is a predetermined threshold value and Group1, Group2 and Group 3.

By studying the prediction method for Group 1 and Group 3 one may conclude that in case of the difference between the travel times of PV1 and PV2 exceeding d in a particular subsection the prediction is done based solely on the data provided by the vehicle having the lower travel time of the two.

4.4. Dwell Time Incorporation

In the earlier attempts the dwell time, the time spent at bus stops, were not considered separately. In this modification, the dwell times were separated and predicted separately to check for improved performance.

The locations of the bus stops were identified using the latitude and longitude. The location information was matched with the raw data file containing the travel time and the 100m subsections containing the bus stops were identified. It was assumed that the dwell time and the running time of a vehicle in the kth subsection are related through,

$$d_{(k)} = c_{(k)}x_{(k)} + w_{1(k)} \tag{4.10}$$

where, $d_{(k)}$ is the dwell time at a bus stop in the kth subsection and $c_{(k)}$ is the parameter that relates the dwell time of the vehicle in the kth subsection to the running time of the vehicle in the same subsection, $w_{1(k)}$ is the process disturbance and $Q_{1(k)}$ and $R_{1(k)}$ are the variances of the process disturbance and measurement noise associated with the prediction of the dwell time respectively.

The parameter $c_{(k)}$ was assumed to be dependent on the dwell time and running time of the vehicle in the subsection with bus stop. It was calculated using data from V eh1 as:

$$c_{(k)} = \frac{d_{veh1(k)}}{x_{veh1(k)}} \tag{4.11}$$

where, $d_{veh1(k)}$ is the dwell time of Veh1 at a bus stop in the $x_{(k)}$ subsection

The following steps were carried out to predict the dwell time of the test vehicle in the next subsection with a bus stop:
1. The apriori estimate of the dwell time was calculated using,

$$\hat{d}_{(k)}^{-} = c_{(k)}\hat{x}_{(k)}^{-} \tag{4.12}$$

c(k) was calculated using Eq. 4.11.

2. The apriori error variance associated with the dwell time (denoted by P_1^-) was calculated using,

$$P_{1(k)}^{-} = c_{(k)}P_{1(k)}^{-}c_{(k)} + Q_{1(k)} \tag{4.13}$$

3. To calculate the aposteriori dwell time estimate, a gain factor (denoted by K1)) was calculated using,

$$K_{1k} = P_{1(k)}^{-}\left[P_{1(k)}^{-} + R_{1(k)}\right]^{-1} \tag{4.14}$$

4. The aposteriori dwell time estimate and error variance were calculated using,

$$\hat{d}_k^{+} = \hat{d}_k^{-} + K_{1(k)}\left[z_{1(k)} - \hat{d}_{(k)}^{-}\right] \tag{4.15}$$

And

$$P^{+}_{1(k)} = \left[1 - K_{1(k)}\right] P^{-}_{1(k)} \tag{4.16}$$

where, z_1 is the dwell time corresponding to Veh2.

Finally, the estimate of total travel time in the k[th] subsection (denoted by $\hat{t}_k$) is calculated using,

$$\hat{t}_{(k)} = \hat{x}^{+}_{k} + \hat{d}^{+}_{k} \tag{4.17}$$

The scheme was implemented and compared with the base method as well as adaptive scheme and the results are shown in the next chapter.

4.5. Time Discretized Prediction method

The evolution of travel time between various time intervals in a given subsection was assumed to be

$$x(t+1) = a(t)x(t) + w(t), \tag{2}$$

where *a(t)* is a model parameter that relates the bus travel time taken in the t^{th} trip and the $(t+1)^{th}$ trip in a particular subsection, *x(t)* is the travel time taken for covering the given subsection in the t^{th} trip and *w(t)* is the associated process disturbance. The measurement process was assumed to be governed by

$$z(t) = x(t) + v(t), \tag{3}$$

where *z(t)* is the measured travel time in a given subsection for a trip *t* and *v(t)* is the measurement noise. It was further assumed that *w(t)* and *v(t)* are zero mean white Gaussian noise signals with *Q(t)* and *R(t)* being their corresponding variances. Thus, two sets of data were required - one set of data to calculate the model parameter '*a(t)*' and another data set to calculate the *a posteriori* travel time estimate. These inputs were identified based on pattern analysis as discussed below.

4.5.1. Travel Time Pattern Analysis

Traffic patterns can be typically classified as yearly, monthly, weekly, daily and hourly. Yearly pattern analysis checks whether the travel time data of same-day/same-time trip of the previous year(s) have a similar pattern as that of the current trip. Similarly, monthly, weekly and daily patterns are compared with the corresponding month's, week's and day's trips respectively. Trip-wise pattern analysis checks whether the current trip has a similar pattern as that of the previous trips on the same day. This may help to capture the traffic conditions on that particular day and incorporate events such as accidents and route diversions.

To analyze the travel time patterns, the Z-test for the mean of a population of differences for paired samples was conducted for the hypothesis testing at 5% level of significance. The test compared each 100m subsections' travel time of the output trip to the input trip to check whether the difference in the mean of the pair is zero or not. The check for daily pattern analyzes the significance of trips that happened on the same time period of the previous days to that of the current trip. A basic assumption of the Z-test for the mean of a population of differences for paired samples data is that the differences of 100 m subsection travel time of the output trip and the input trip follow a normal distribution. Tests were carried out to find whether this assumption is true by using a statistical measure "skewness" given as

$$Skewness\ S = \frac{[3\times(\bar{x}-M)]}{s}, \quad (4.18)$$

where $\bar{x}$ is the sample mean, M is the sample median and s is the standard deviation of the sample. According to Rees *(26)*, if the skewness value is greater than +1, the distribution has positive skew and if the skewness value is less than -1, the distribution has negative skew. If the skewness value lies in between -1 and +1, the distribution is roughly symmetrical, i.e., it follows a normal distribution. In the present study, the skewness is calculated for the differences of 100 m subsection travel times between the output trip and the input trip. The results of skewness calculated for various trips on a sample day, 29th January 2013, are shown in Figure 10. From the figure it can be observed that the calculated skewness values lie within the range of -1 and +1. Since none of the values are outside the range of -1 and +1, it can be concluded that differences of 100 m subsection travel time of the output trip and the input trip follow a normal distribution and hence the Z-test can be adopted for hypothesis testing. The Z-test is used to test the hypothesis, and is given by

$$\overline{Z} = \frac{\bar{d}}{\frac{s_d}{\sqrt{n}}}, \quad (4.19)$$

where $\bar{d}$ is the mean of differences of 100 m section travel time of the output trip and the input trip, s_d is the standard deviation of the sample and n is the sample size. In the present study, the test was conducted at 5% level of significance. So, if the calculated Z-value lies in between -1.96 and +1.96, then we can say that the null hypothesis is accepted, which means that the mean of differences is zero.

Travel time patterns for peak (both morning peak, 6 AM-8 AM and evening peak, 3 PM-8PM together) and off-peak (both morning peak, 8 AM-10 AM and evening off-peak, 10 AM-3 PM together) trips were analyzed separately for each day of the week. Each peak and off-peak output trip was compared with the 28 previous days' corresponding input trips with the same starting time as that of the output trip. To analyse the daily pattern, 5880 (15 days × 14 trips/day × 28 preceding days' same time trips) 'Z' values were calculated. Then, a ratio has been calculated between the number of times the null hypothesis was accepted to the total number of times the hypothesis was tested for each case. If the ratio is high, we can conclude that the target trip is significant in predicting the current trip (output trip). The results obtained from the statistical analysis for peak zones are shown in Figure 11 and Figure 12, and the detailed results are given in Table 3. Table 3 illustrates the trend in the rankings followed by daily patterns. For example, a trip that happened on a Sunday off-peak zone shows a strong correlation with the previous week's same day/same time trip corresponding to it.

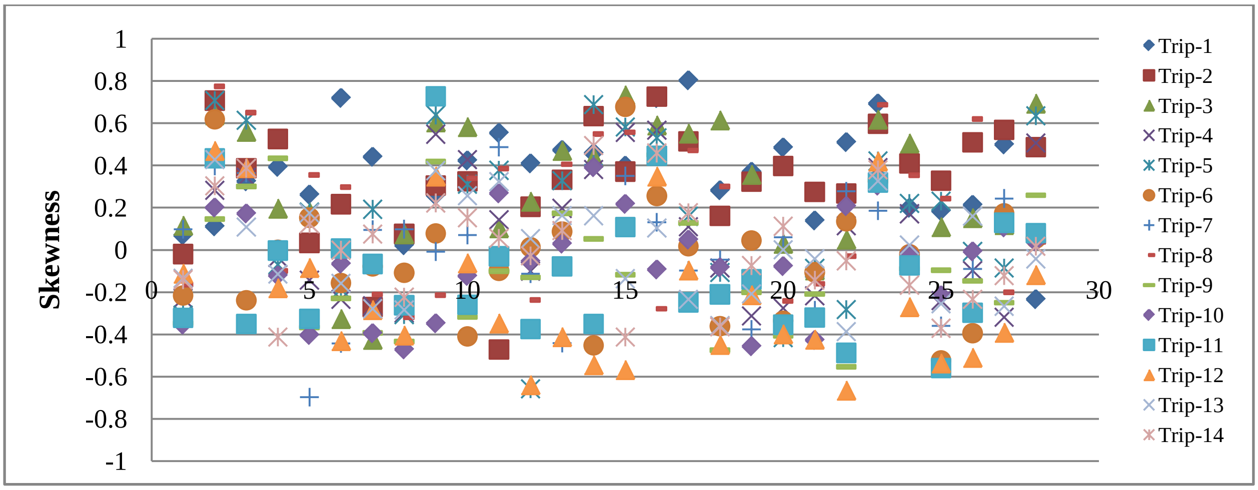


FIGURE 10 Skewness Calculated For Various Trips For A Sample Day.

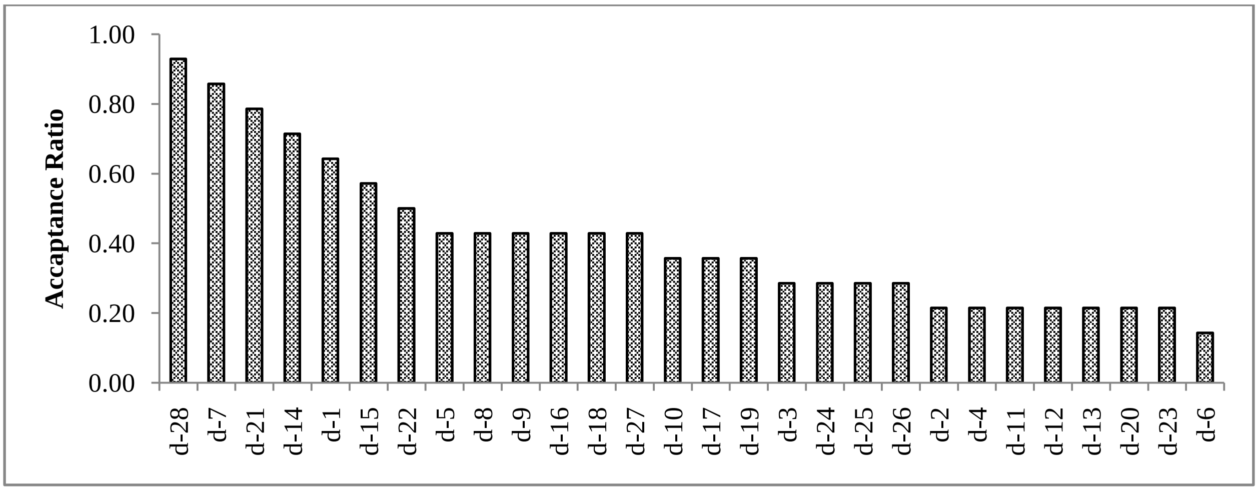


FIGURE 11 Travel Time Patterns Observed For Sunday.

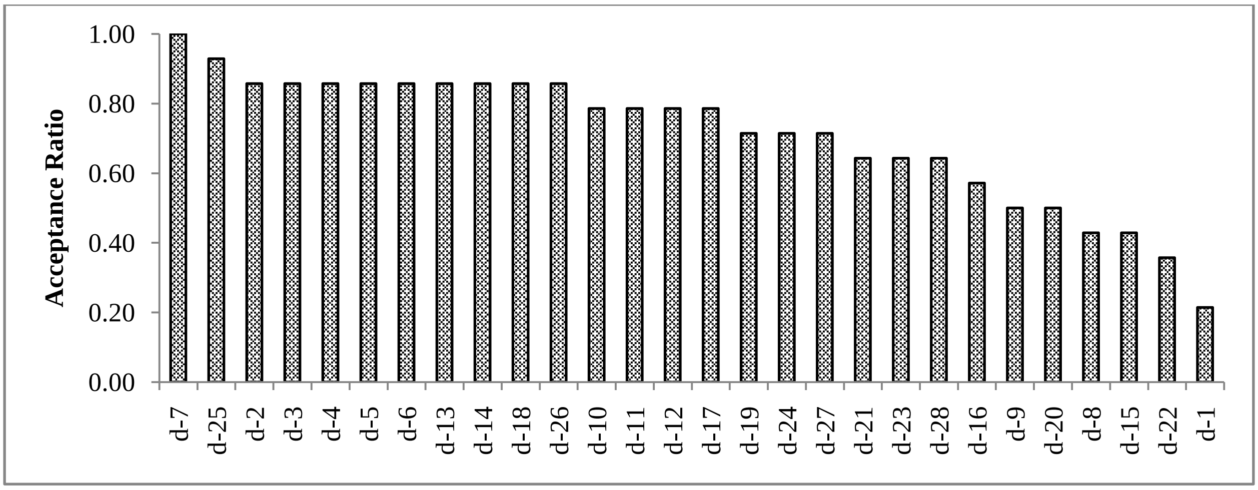


FIGURE 12 Travel Time Patterns Observed For Monday Peak Period.

TABLE 3 Pattern analysis results

Rank	Sunday		Monday		Tuesday		Wednesday		Thursday		Friday		Saturday	
	Off-Peak	Peak	Off-Peak	Peak	Off-Peak	Peak	Off-Peak	Peak	Off-Peak	Peak	Off-Peak	Peak	Off-Peak	Peak
1	d-7	d-28	d-5	d-7	d-3	d-5	d-2	d-9	d-3	d-2	d-1	d-1	d-3	d-4
2	d-14	d-7	d-2	d-25	d-5	d-7	d-9	d-14	d-8	d-6	d-4	d-11	d-28	d-8
3	d-15	d-21	d-7	d-2	d-7	d-11	d-26	d-26	d-28	d-8	d-9	d-16	d-5	d-15
4	d-21	d-14	d-11	d-3	d-1	d-12	d-1	d-28	d-2	d-13	d-20	d-2	d-21	d-21
5	d-28	d-1	d-14	d-4	d-4	d-25	d-5	d-2	d-7	d-22	d-2	d-3	d-1	d-22
6	d-22	d-15	d-4	d-5	d-6	d-6	d-6	d-7	d-13	d-1	d-3	d-4	d-2	d-28
7	d-1	d-22	d-12	d-6	d-8	d-15	d-7	d-13	d-19	d-3	d-7	d-7	d-4	d-17
8	d-8	d-5	d-18	d-13	d-24	d-20	d-19	d-1	d-1	d-7	d-10	d-22	d-8	d-24
9	d-10	d-8	d-24	d-14	d-10	d-27	d-20	d-6	d-5	d-17	d-11	d-8	d-9	d-25
10	d-18	d-9	d-6	d-18	d-11	d-4	d-21	d-15	d-10	d-20	d-16	d-14	d-7	d-5
11	d-26	d-16	d-10	d-26	d-12	d-8	d-27	d-19	d-21	d-27	d-22	d-28	d-12	d-7
12	d-2	d-18	d-13	d-10	d-14	d-10	d-11	d-27	d-26	d-10	d-6	d-15	d-15	d-16
13	d-3	d-27	d-17	d-11	d-15	d-13	d-13	d-5	d-27	d-14	d-14	d-18	d-22	d-1
14	d-4	d-10	d-23	d-12	d-13	d-14	d-18	d-16	d-6	d-15	d-15	d-9	d-24	d-6
15	d-9	d-17	d-25	d-17	d-17	d-18	d-25	d-20	d-12	d-21	d-27	d-25	d-10	d-10
16	d-16	d-19	d-26	d-19	d-18	d-19	d-8	d-21	d-15	d-24	d-28	d-27	d-14	d-11
17	d-19	d-3	d-3	d-24	d-19	d-26	d-4	d-22	d-9	d-26	d-8	d-20	d-17	d-14
18	d-23	d-24	d-27	d-27	d-21	d-1	d-12	d-8	d-14	d-5	d-13	d-23	d-23	d-23
19	d-25	d-25	d-9	d-21	d-25	d-3	d-14	d-12	d-16	d-9	d-18	d-6	d-25	d-26
20	d-27	d-26	d-16	d-23	d-26	d-17	d-22	d-18	d-17	d-23	d-12	d-17	d-26	d-2
21	d-5	d-2	d-19	d-28	d-27	d-21	d-28	d-23	d-18	d-28	d-21	d-21	d-13	d-3
22	d-12	d-4	d-20	d-16	d-16	d-22	d-15	d-25	d-20	d-16	d-24	d-24	d-16	d-9
23	d-17	d-11	d-21	d-9	d-28	d-24	d-16	d-4	d-22	d-19	d-19	d-10	d-6	d-12

24	d-20	d-12	d-28	d-20	d-2	d-28	d-23	d-10	d-24	d-12	d-23	d-13	d-11	d-13
25	d-24	d-13	d-15	d-8	d-20	d-9	d-17	d-11	d-23	d-4	d-25	d-5	d-18	d-19
26	d-6	d-20	d-22	d-15	d-23	d-2	d-10	d-3	d-11	d-11	d-17	d-19	d-19	d-18
27	d-11	d-23	d-1	d-22	d-22	d-16	d-3	d-17	d-4	d-18	d-5	d-12	d-20	d-20
28	d-13	d-6	d-8	d-1	d-9	d-23	d-24	d-24	d-25	d-25	d-26	d-26	d-27	d-27

* (d-n) represents previous n^{th} day same time trip

The above ranking was used while selecting the input data for the prediction method. In order to take into account the present day traffic conditions, data from previous two buses (PV1 and PV2) were also taken into consideration. This data will reflect the effect of events that have taken place in that subsection on that day. The pattern analysis results were arranged as two sets of data in the order of preference from lower to higher for both peak and off-peak time zones on all days in Table 4. The data presented in column 1 were used to obtain the value of *a(t)* during each time interval and the data from column 2 were used to obtain the *a posteriori* travel time estimate.

Table 4 Data set used for KF technique

Sunday		**Monday**		**Tuesday**		**Wednesday**		**Thursday**		**Friday**		**Saturday**	
d-13	d-11	d-8	d-1	d-9	d-22	d-24	d-3	d-25	d-4	d-26	d-5	d-27	d-20
d-6	d-24	d-22	d-15	d-23	d-20	d-10	d-17	d-11	d-23	d-17	d-25	d-19	d-18
d-20	d-17	d-28	d-21	d-2	d-28	d-23	d-16	d-24	d-22	d-23	d-19	d-11	d-6
d-12	d-5	d-20	d-19	d-16	d-17	d-15	d-28	d-20	d-18	d-24	d-21	d-16	d-13
d-27	d-25	d-16	d-9	d-26	d-25	d-22	d-14	d-17	d-16	d-12	d-18	d-26	d-25
d-23	d-19	d-27	d-3	d-21	d-19	d-12	d-4	d-14	d-9	d-13	d-8	d-23	d-17
d-16	d-9	d-26	d-25	d-18	d-17	d-8	d-25	d-15	d-12	d-28	d-27	d-14	d-10
d-4	d-3	d-23	d-17	d-13	d-15	d-18	d-13	d-6	d-27	d-15	d-14	d-24	d-22
d-2	d-26	d-13	d-10	d-14	d-12	d-11	d-27	d-26	d-21	d-6	d-22	d-15	d-12
d-18	d-10	d-6	d-24	d-11	d-10	d-21	d-20	d-10	d-5	d-16	d-11	d-7	d-9
d-8	d-1	d-18	d-12	d-24	d-8	d-19	d-7	d-1	d-19	d-10	d-7	d-8	d-4
d-22	d-28	d-4	d-14	d-6	d-4	d-6	d-5	d-13	d-7	d-3	d-2	d-2	d-1
d-21	d-15	d-11	d-7	d-1	d-7	d-1	d-26	d-2	d-28	d-20	d-9	d-21	d-5
d-14	d-7	d-2	d-5	d-5	d-3	d-9	d-2	d-8	d-3	d-4	d-1	d-28	d-3
PV1	PV2	PV1	PV2	PV1	PV2	PV1	PV2	PV1	PV2	PV1	PV2	PV1	PV2
TV		TV		TV		TV		TV		TV		TV	

The steps in the algorithm were as follows:

1. The entire section of travel between origin and destination was divided into *N* subsections of equal length (100 m).
2. Let the length of the column 1 in Table 4 be *'g'*. The travel time data from column 1 were used to obtain the value of A through

$$a(t) = \frac{x(t+1)}{x(t)}, t = 1,2,3,...,(g-1). \quad (4)$$

3. Let $x_{TV}(t)$ denote the travel time taken by the test vehicle (which is the vehicle for which the travel time needs to be predicted) to cover a given subsection. It was assumed that
$$E[x_{TV}(1)] = \hat{x}(1), \tag{5}$$
$$E[x_{TV}(1) - \hat{x}(1)^2] = P(1) \quad , \tag{6}$$
where $\hat{x}(t)$ is the estimate of travel time of the TV in the t^{th} time interval.
4. For $t = 2,3,4,\ldots,(g\text{-}1)$, the following steps were performed:
   a. The *a priori* estimate of the travel time was calculated by using $\hat{x}^-(t+1) = a(t)\hat{x}^+(t)$, where the superscript '–' denotes the *a priori* estimate and the superscript '+' denotes the *a posteriori* estimate.
   b. The *a priori* error variance (denoted by P^-) was calculated using
   $$P^-(t+1) = a(t)P^+(t)a(t) + Q(t), \tag{7}$$
   c. The Kalman gain (denoted by K) was calculated by using
   $$K(t+1) = P^-(t+1)[P^-(t+1) + R(t+1)]^{-1}, \tag{8}$$
   d. The *a posteriori* travel time estimate and error variance were calculated using, respectively,
   $$\hat{x}^+(t+1) = \hat{x}^-(t+1) + K(t+1)[z(t+1) - \hat{x}^-(t+1)], \tag{9}$$
   $$P^+(t+1) = [1 - K(t+1)](t)P^-(t). \tag{10}$$

The above steps 2, 3 and 4 mentioned in the algorithm will be repeated for all *N* subsections to predict travel time. Thus, the objective here was to predict the travel time of the TV using the travel time obtained from previous all vehicles including previous two vehicles (PV1 and PV2) in a given subsection.

V. IMPLEMENTATION AND RESULTS

This chapter details the validation procedures adopted to evaluate the performance of the algorithm and describes implementation efforts and results obtained therein. The performance of the various procedures was evaluated to select one that can be used for the real time field implementation. Following this, the selected algorithm was used for the development of the complete application that can be used for automated real time field implementation.

5.1. Performance evaluation

The prediction accuracy is measured in a variety of ways. Some agencies monitor both the AVL and real-time arrival information systems directly (usually from a central location) to determine the accuracy of the predictions. This monitoring can take place either in real time or historically, using data logs from the signs and/or central system. Other agencies conducted field visits to check the accuracy.

Preliminary validation studies were performed on route 21 G and 21 L before the automated data collection system was in place by collecting GPS data manually. This was carried out offline, i.e. the buses were not fitted with GPS but the data were collected by carrying the units and travelling in the bus. The preliminary validation was followed by the real time processing of the data from Route 5C and the algorithm and model were validated and tested. The real time validation are described in section 5.2.

The Mean Absolute Percentage Error (MAPE) was considered as the measure of effectiveness (Eq. 5.1). It is the mean of absolute percentage differences between the predicted and the actual travel times and is calculated as:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\hat{x}_{p_i} - x_{m_i}}{x_{m_i}}\right| \times 100$$

where,
$\hat{x}_{p_i}$ = Predicted travel time of TV for ith subsection,
$\hat{x}_{m_i}$ = Measured travel time of TV for ith subsection

5.1.1. Offline Model Corroboration

Offline corroboration of the model was carried out on Route 21 G and 21L. 21L is approximately 20 km long with a travel time of about 60 - 75 minutes and connects Parry's corner to Velachery. There are 21 bus stops (including the terminal stops in this route). 21G is approximately 28 km long. The travel time in this route is approximately 2 hours with 19 bus stops and 29 signalized intersections. Data were collected manually for 10 days each from both these routes.

Corroboration of the space discretised model

The short term travel time prediction was carried out using the base algorithm and the results were compared with the measured travel time data. The entire section was broken down into *N* consecutive subsections of 100 m length. The length of the entire section under study is approximately 28 km. The spatial discretization of 100 m was chosen in order to have enough data points to run the algorithm effectively.

Figure 13 presents a sample comparison of the predicted travel times and the measured values over 500 m subsections of the entire section. Here, the predicted travel time to cover the first 500 m subsection was found by adding the predicted values provided by the algorithm over each of the first five 100 m subsections. This procedure was repeated for all subsequent 500 m subsections. It can be observed that the prediction results agree reasonably well with the measured data.

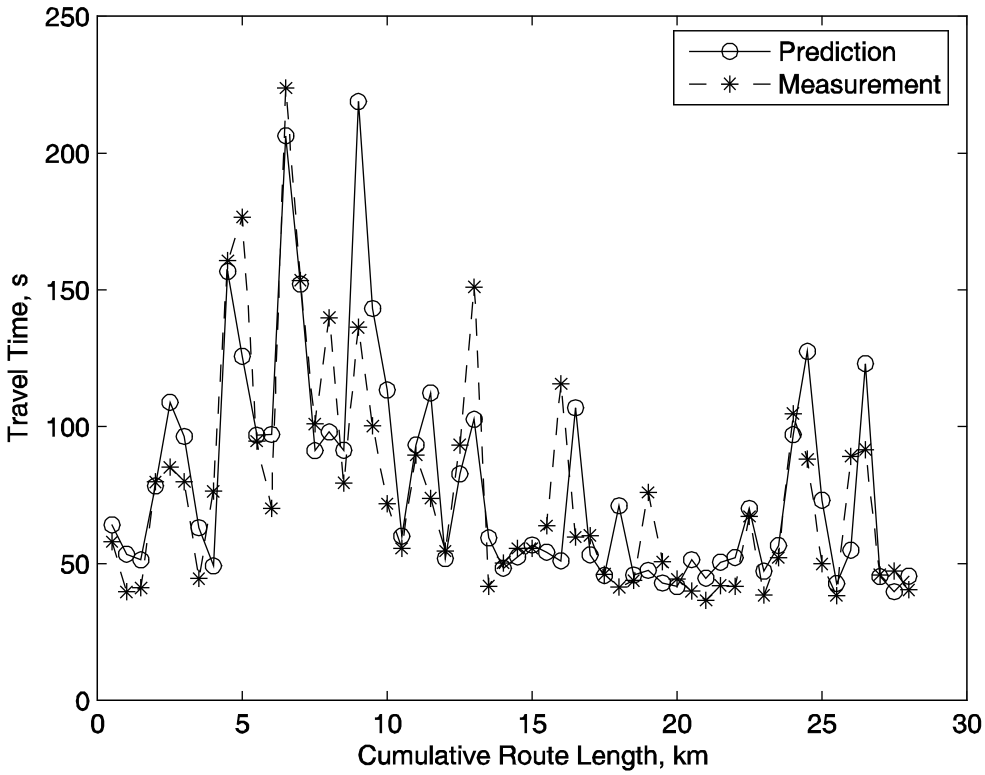


Figure 13: Comparison of the predicted and measured travel time for the test vehicle.

However, for an APTS application, one of the requirements is to predict travel time between bus stops to provide information about the next bus arrival. Keeping this in mind, two main subsections were selected within the test section and the corresponding travel times were predicted using the proposed algorithm. The first subsection (referred to as "Subsection 1" in Figure 14) connects the Guindy bus stop to the Chennai Airport bus stop, which is approximately 6.5 km long. The second subsection (referred to as "Subsection 2" in Figure 11) connects the R. A. Puram bus stop to the Guindy bus stop and is approximately 6.8 km long. These two subsections were selected since they are important in the bus route under study, the first subsection connecting the city's airport (which is located in the outskirts of the city) to the Central Business District (CBD) and the second subsection as a representative stretch within the city limits with many signalized intersections. When compared with the first subsection, the traffic in the second subsection is more congested and is more constrained with bottlenecks. Thus, these two subsections represent reasonably contrasting traffic scenarios in which the performance of the proposed algorithm is evaluated. The measured travel times of the TV on the days of study were between 643 seconds and 754 seconds for subsection 1 and between 971 seconds and 1284 seconds for subsection 2. Thus, it can be observed that the travel times in the two subsections are appreciably different (although they are of almost the same length), showing that the traffic scenarios in these two subsections are contrasting.

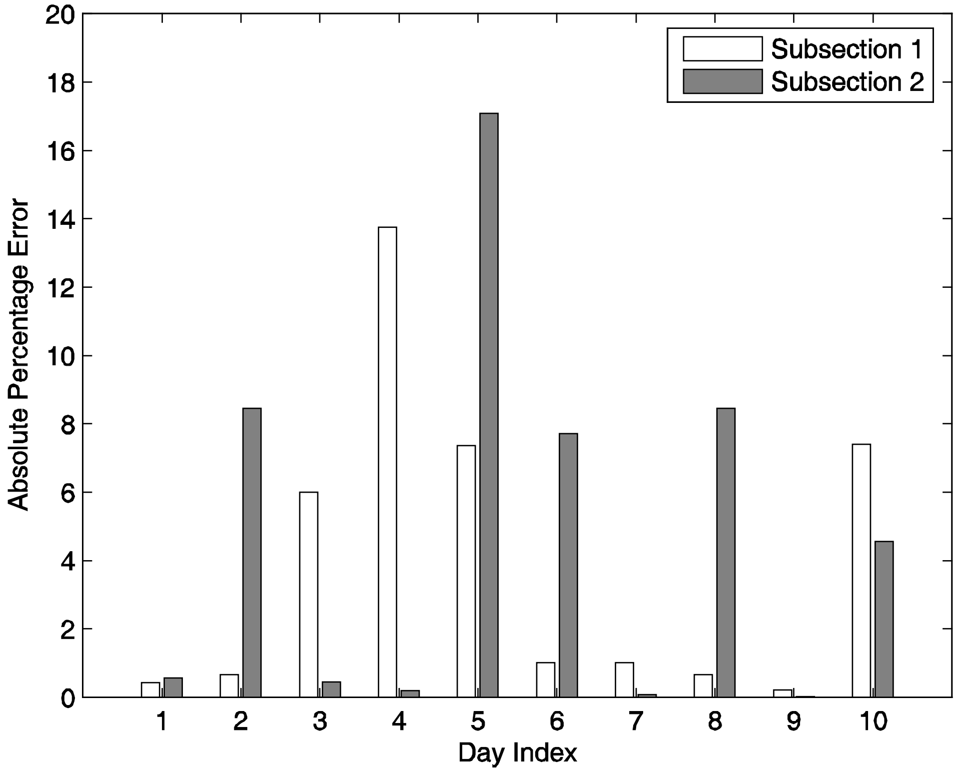


Figure 14: Prediction error for both the subsections under study.

The algorithm was used to predict the travel time over the two subsections for all the 10 days. Absolute Percentage Error (APE) was used as a measure of prediction accuracy and was calculated using

$$\mathrm{APE} = \frac{\left|\hat{x}_S - x_{TVM}\right|}{x_{TVM}} * 100, \tag{3}$$

where $\hat{x}_S$ is the predicted travel time of the test vehicle to cover this subsection and x_{TVM} is the corresponding travel time measured from field. The results for subsections 1 and 2 are plotted in Figure 14. It can be observed that the travel time in the first subsection is predicted reasonably well with the APE varying from 0.2191 % to 13.7502 %. The travel time in the second subsection is also predicted reasonably well with the corresponding APE varying from 0.0189 % (on Day 9) to 17.086 %.

Now, as a benchmark, we have compared the prediction of the proposed algorithm with an alternate technique (referred to as the average approach from here on) where the predicted travel time of TV in a given subsection is taken to be the average of the travel times of PV1 and PV2 in that subsection. The corresponding results are shown in Figures 13 and 14.

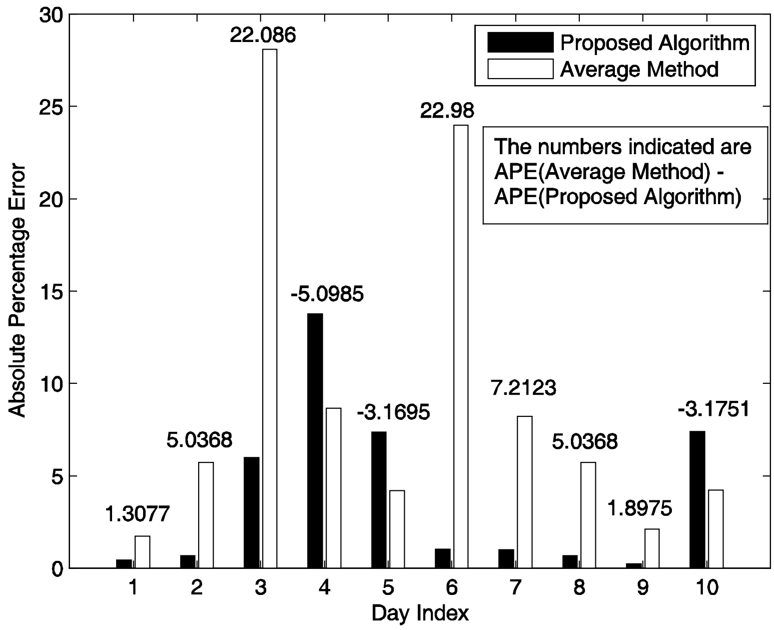


Figure 15: Comparison between the Kalman Filter (KF) algorithm and the averaged values: Prediction error for subsection 1.

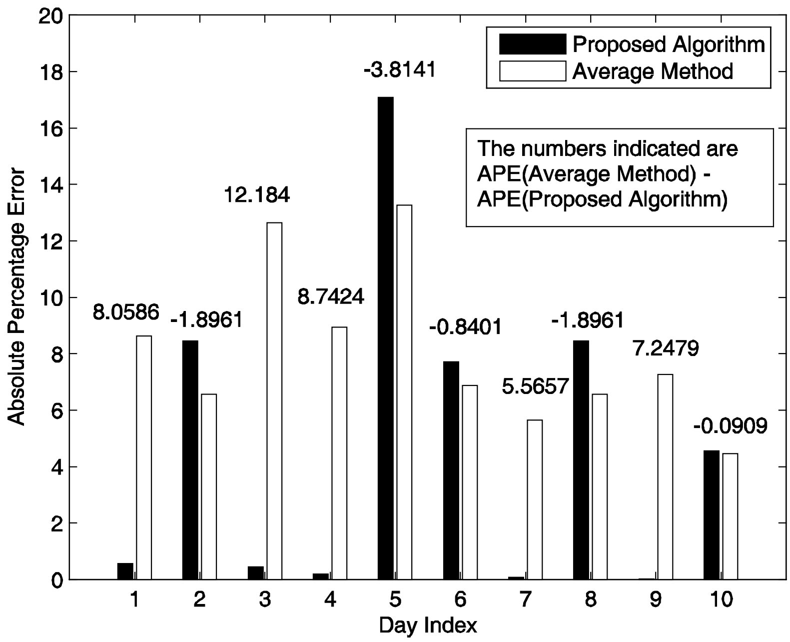


Figure 16: Comparison between the Kalman Filter (KF) algorithm and the averaged values: Prediction error for subsection 2.

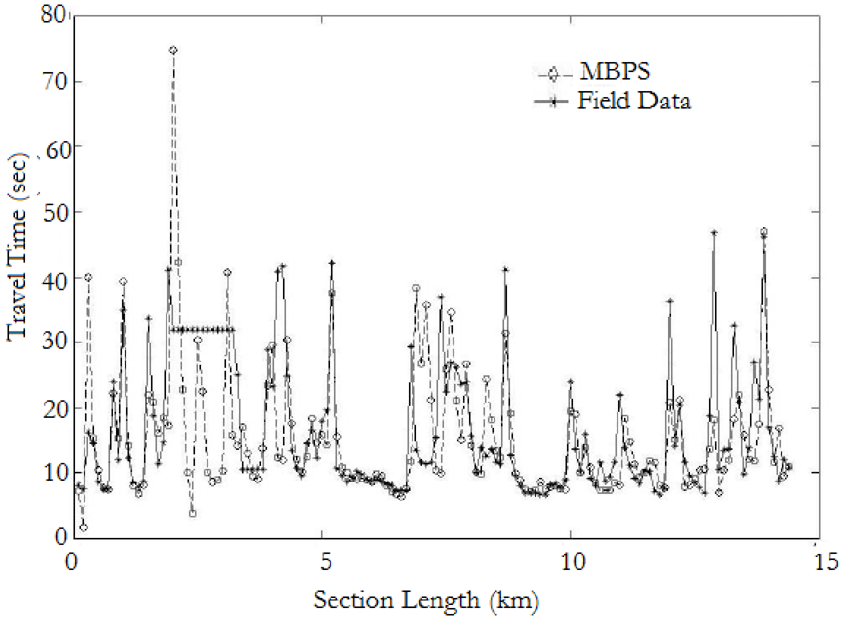
Figure 16 shows the APE values obtained using the proposed algorithm as well as those obtained using the average approach for the first subsection under study. It can be seen that the proposed algorithm gives better prediction than the average approach on seven out of ten days in this case. Also, it can be noted that on the three days when the performance of the average approach is better than the proposed algorithm, the relative advantage is marginal. However, when the proposed algorithm outperforms the average approach, the relative advantage is much higher. The relative advantage is quantified using the difference between the absolute percentage error obtained from the average approach and the corresponding value obtained from the proposed algorithm. These values are indicated above each set of bars in the figure. Overall, the prediction of the proposed algorithm outperformed the average approach by an average value of 9.3653 % over the seven days whereas the average approach outperformed the proposed algorithm over the remaining days by an average value of 3.8143 %. These numbers are obtained by taking the average of the corresponding values shown above the bars in Figure 13.

A similar comparison is made for the second subsection between R. A. Puram and Guindy and the results are shown in Figure 14. In this case, the proposed algorithm outperformed the average approach on five days by an average value of 8.3597 %. The average approach gave a better result on the remaining days, but the overall advantage was only 1.7074 %. Here also, we can observe the relative advantage of the proposed algorithm over the average approach. Overall, it can be seen that the proposed algorithm performs better than the average approach.

As mentioned in section 2, a comparative study was also carried out to check the relative performance of the model using data from three consecutive days and three consecutive vehicles on the same day. One sample result of the same is discussed below. Here data from three consecutive days (Days 5, 6, and 7) were selected because on each of these days, the Test Vehicle (TV) started from its origin at approximately the same time. The proposed algorithm was used to calculate the travel time of TV on Day 7 using the corresponding data from Days 5 and 6. The APE thus obtained for subsection 1 and subsection 2 were 10.1554 % and 2.1593 % respectively. The corresponding APE values for the two subsections on Day 7 using data from the same day were 1.007 % and 0.07725 % respectively. Thus, it can be seen that using the data from the same day is able to give better results for this particular set. This may be due to the fact that the data from the same day are able to reflect the effects of events such as accident, break downs, etc., that have taken place in the route just before the TV has traveled on the route. Based on these results as well as on practical difficulties discussed in section 2, the present study used data from three consecutive vehicles traveling on the same day.

Corroboration of dwell time incorporated model

The corroboration of the dwell time incorporated travel time prediction model was carried out using data from 21L route. The predicted travel time of TV using the dwell time incorporated model based prediction scheme (MBPS) was compared with the actual travel time measured from field for every 100m subsections. Figures 17 (a) and (b) shows the comparison between the predicted and measured travel times for every 100m subsections for two sample days. The predicted and measured times compare well.



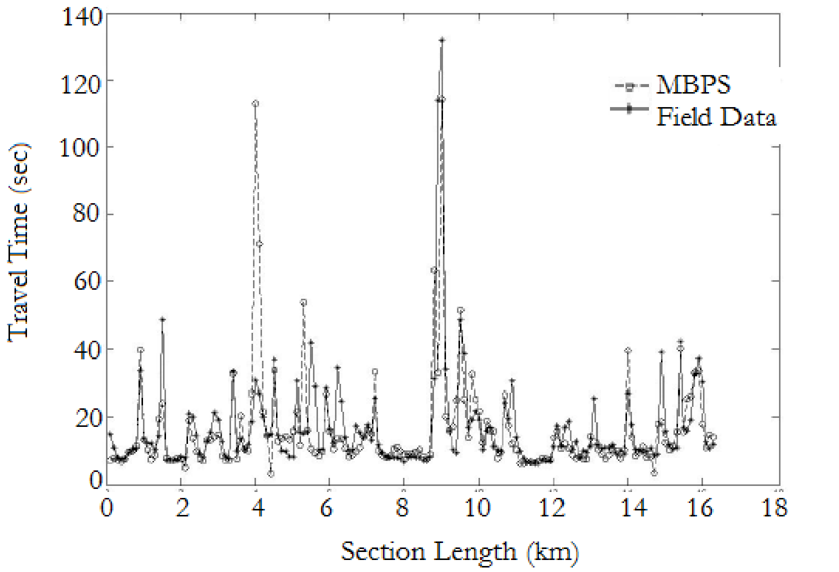
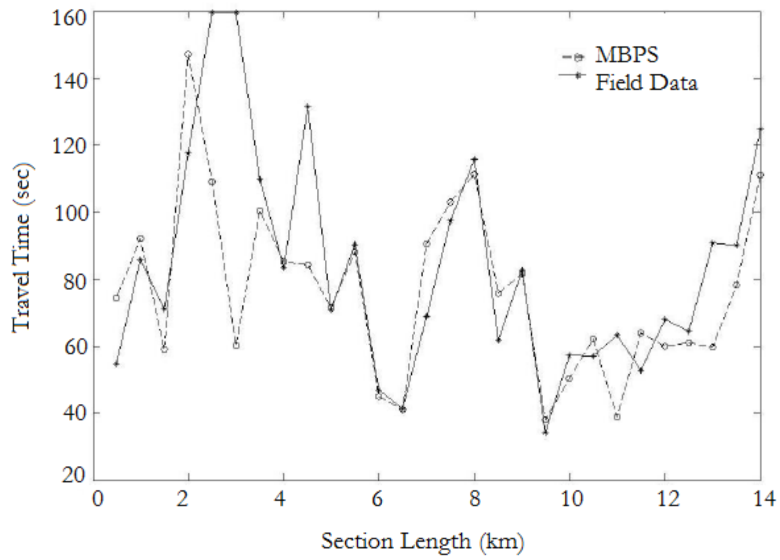
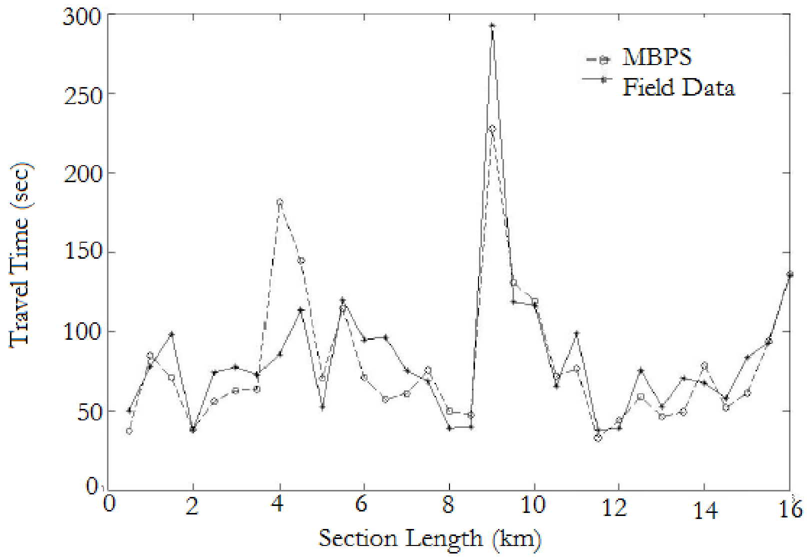


Figure 17: (a) Comparison between MBPS and Field data for 100m subsections (b) Comparison between MBPS and Field data for 100m subsections

In reality the interest focuses on travel time between bus stops which are spaced at least 500 m apart. Hence, prediction was carried out for 50 m sub sections and prediction accuracy was checked. Sample plots are shown below (Figures 18 (a) and (b)). The MAPE between the predicted travel time and the measured travel time of the TV for the 10 days were calculated and given in Table 5. The values of the MAPE for these ten days varied from 16.65% to 45.27% with an average value of 26.63%.



(a)



(b)

Figure 18: (a) Comparison between MBPS and Field data for 500m subsections (b) Comparison between MBPS and Field data for 500m subsections

Table 5: MAPE for 500 m subsections

S No.	MAPE
1	16.65
2	22.18
3	20.39
4	45.27
5	33.47
6	18.00

7	22.58
8	29.82
9	20.25
10	37.73

It can be seen from Table 5 that the results are promising with an average MAPE of 26.63% for 10 days. In general a MAPE of 10% is considered very good, a MAPE in the range 20 % - 30 % or even higher is quite common. According to Brooks *et al*. [Brooks, 2006], if the MAPE is less than 40 %, the forecast is reasonably reliable. However, in reality, the algorithm must predict the arrival time of buses at bus stops and hence the prediction must be performed between bus stops. Hence, predictions were also carried out between consecutive bus stops. For corroborating the prediction scheme, only 13 bus stops were considered due to inconsistency of the initial data obtained from the GPS instruments. On some days, the data logging from the GPS instrument started only after the bus travelled over some distance. It was found that this distance was limited to the 4th bus stop in all cases. Hence the first 4 bus stops were not considered in this analysis and the 5th bus stop (namely Queen Mary's College stop) was taken as the starting bus stop. The travel time was predicted from this stop for successive pairs of bus stops and the corresponding results are presented below. Figure 11 shows the comparison between the predicted travel time and the observed travel time for successive pairs of bus stops for all days.

The MAPE between the predicted travel time and the measured travel time of the test vehicle for successive bus stops are presented in Table 6. The values of the MAPE for these 10 days varied from 14.24% to 47.11%.

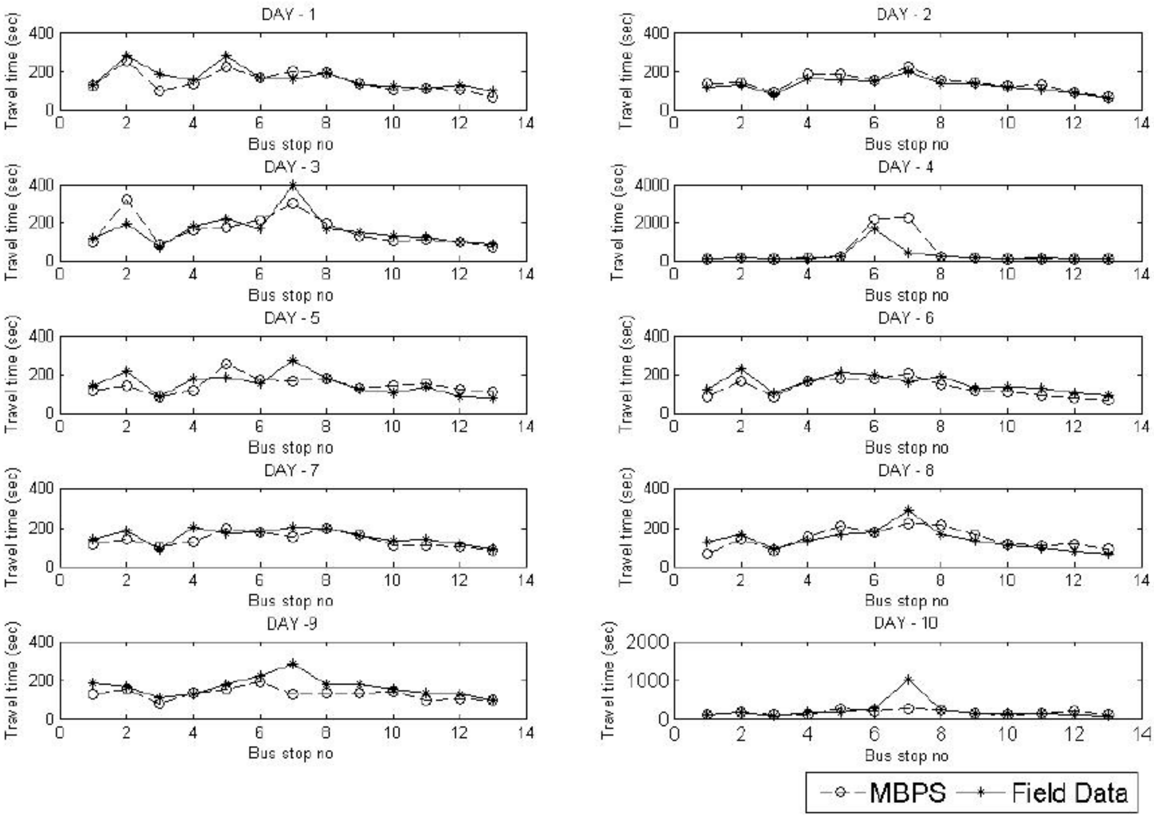


Figure 16: Comparison of travel time between MBPS and Field data for successive bus stops

Table 6: MAPE values for successive pairs of bus stops – MBPS

S.NO.	MAPE
1	14.24
2	37.00
3	20.46
4	47.11
5	24.72
6	19.32
7	14.67
8	22.45
9	19.13
10	30.75

d. Comparison with Field Approach

The performance of the prediction scheme was compared with a popularly adopted field method namely the "average speed method". In this method, the travel time of a vehicle is predicted using the current location, the average speed of the vehicle in the previous subsection and the distance to be covered in the current subsection. The results obtained for successive bus stops using this approach are given in Table 7. The MAPE values are shown in Figure 17.

Table 7: MAPE values for successive pairs of bus stops - Field approach

S.NO.	MAPE
1	32.34
2	29.54
3	36.57
4	64.38
5	57.90
6	37.02
7	27.39
8	49.45
9	26.14
10	88.10

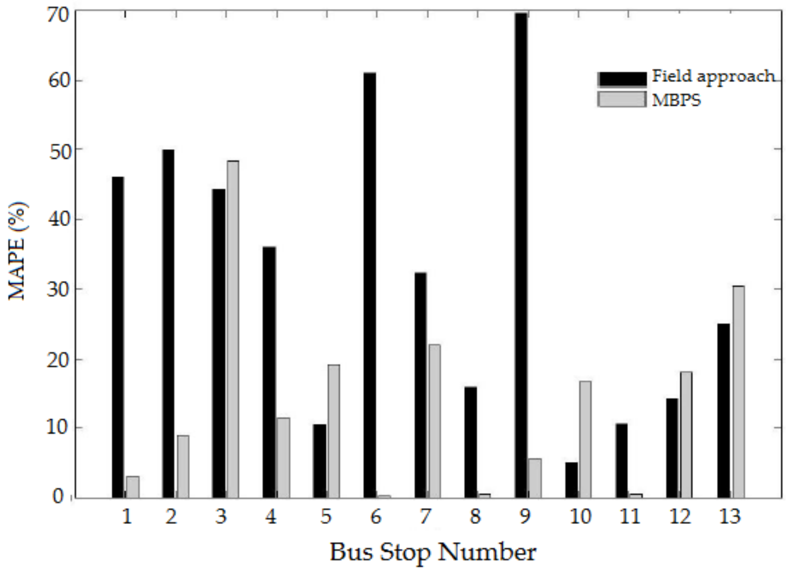


Figure 17: MAPE comparison between Field approach and MBPS for successive bus stops

A comparison of MAPE values of Field approach and MBPS (Table 8) shows that MBPS outperforms field approach during most days. The average MAPE for the Field approach was 44.88% whereas the average MAPE for the MBPS was only 24.99%. Thus, the proposed algorithm was able to improve the prediction accuracy compared to the Field approach.

.

Table 8: Comparison of MAPE between Field approach and MBPS for successive bus stops

S. No	Field approach	MBPS
1	32.34	14.24
2	29.54	37.00
3	36.57	20.46
4	64.38	47.11
5	57.90	24.72
6	37.02	19.32
7	27.39	14.67
8	49.45	22.45
9	26.14	19.13
10	88.10	30.75

Corroboration of time discretization method

The results obtained from the implementation of the prediction algorithm presented in the section 4.5.2, which will be referred to as the 'time discretization method', are discussed in this section. Since the proposed algorithm uses the identified significant historic data as inputs to predict bus travel time in each subsection, a comparison was carried out with a scheme that does not use the pattern based historic data (referred to as the 'space discretization method') (Vanajakshi et al. 2008). The scheme presented in section 4.5.2 was used to predict travel time for each 100 m subsection. However, from the data it can be observed that is a distance of at least 500 m between bus stops. Hence, the final comparison was made between predicted and measured travel times for every 500 m subsection. The predicted travel time to cover a 500 m subsection was found out by adding the predicted travel time from the corresponding five 100 m travel time values obtained from the prediction algorithm. Figure 18 (a) shows the comparison between the predicted and measured travel times for every 500m subsections for sample trip. From Figure 18 (a), it can be observed that the predicted values are closely matching with the measured data.

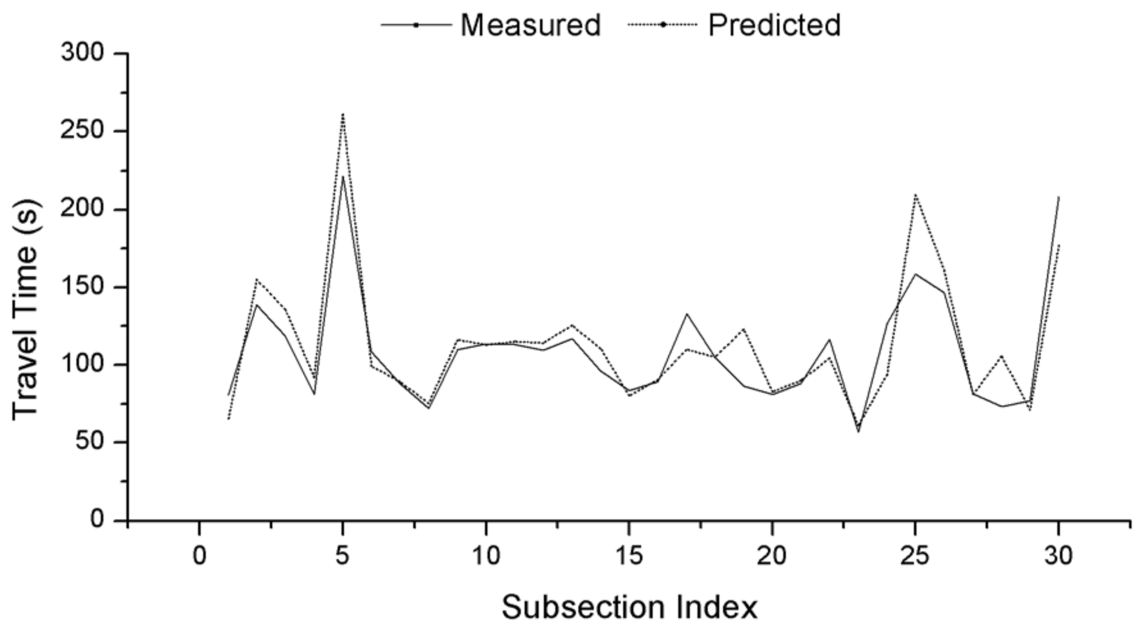


Figure 18 (a): Predicted and measured travel times for a peak period trip on a sample day in 5C route.

Figure 18(b) present the results obtained for all trips that happened on a sample day. From Figure 18 (b), it can be observed that the proposed time discretization method is performing better than the space discretization method.

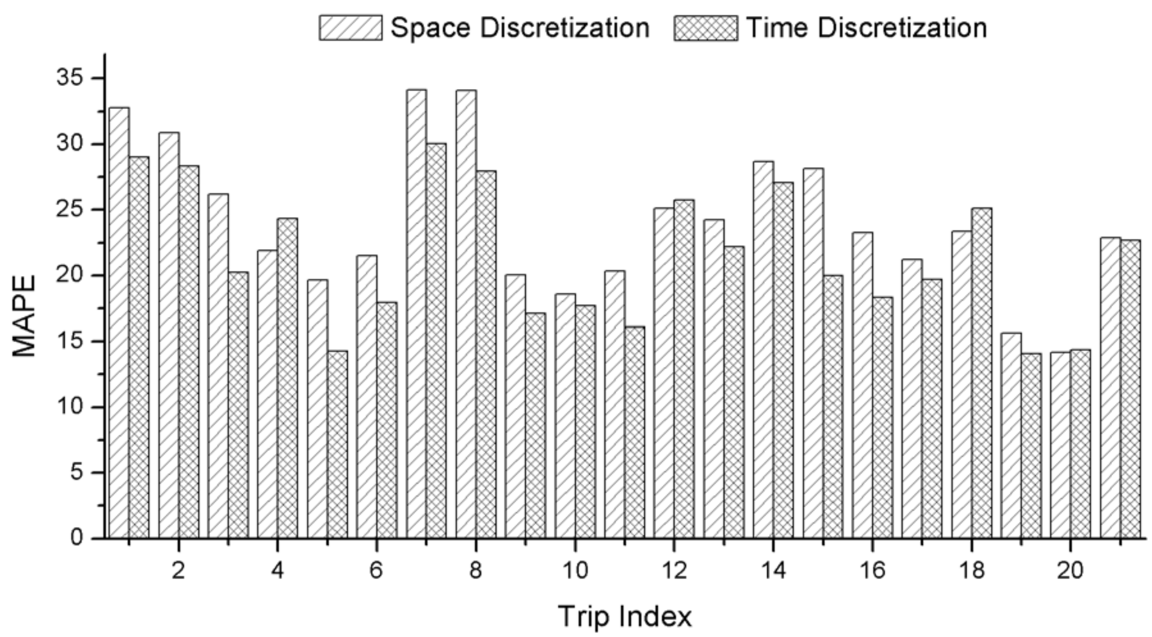


Figure 18 (b): MAPE values obtained for all trips on a sample day (5C route).

A similar analysis was carried out for all days and average results are given in Table 9. It can be observed that time discretization is performing better than space discretization for all days.

Table 9: MAPE Comparison between Time and Space Discretization Methods

Day Index	Time Discretisation	Space Discretisation
Day 1	25.88	29.43
Day 2	29.16	33.57
Day 3	29.95	33.56
Day 4	29.88	32.53
Day 5	30.32	31.95
Day 6	20.55	22.10
Day 7	27.55	29.16

5.2 Online Implementation

The offline testing of the algorithm is easy to manage since the input are static and kept ready to be read by the algorithm. However, for online implementation, all the steps need to be automated starting from identifying the file to be read, data filtering, to dynamic running of the prediction algorithm as real time data gets appended to the input file. While making the algorithm online, the corroboration was carried out using data from 5C buses Also, two modifications from base method was also tested for possibility of improved performance. The evaluation of these methodologies is discussed first followed by the real time implementation.

The base method of bus arrival time prediction discussed in the earlier chapter using real time data from two previous vehicles is first shown followed by the two modifications. A simple Kalman Filtering Technique was adopted to predict the bus arrival time.

5.2.1 Base Prediction Method

The model discussed in the earlier chapter was implemented and the estimation was carried out using Kalman filtering technique. The general steps of KFT discussed in section 4.1 were implemented for the specific problem under consideration as follows:

If $x_{tv}(k)$ is the travel time for the test vehicle to cover the k^{th} subsection, then it is assumed that $E[x_{tv}(1)] = \hat{x}(1)$, and $E[(xtv(1) - \hat{x}(1))^2] = P(1)$, where $\hat{x}(k)$ is the estimate of the travel time for the test vehicle in the k^{th} subsection. The following steps are then applied recursively for subsections k=2 to N-1:

i. The a priori travel time estimate is calculated using
$\hat{x}^{i}(k+1) = a(k)\hat{x}^{ii}(k)$, (5.1)
Where, the superscript '*i*' denotes the priori estimate and the superscript '*ii*' denotes the posteriori estimate.

ii. The a priori error variance is given by
$P^{i}(k+1) = a(k)P^{ii}(k)a(k) + Q(k)$, (5.2)
where, $Q(k)$ is the variance of the process distribution $w(k)$.

iii. The Kalman gain '*K*' is given by
$K(k+1) = P^{i}(k+1)\left[P^{i}(k+1) + R(k+1)\right]^{-1}$, (5.3)

iv. The a posteriori estimate of the travel time I given by,
$\hat{x}^{ii}(k+1) = \hat{x}^{i}(k+1) + K(k+1)\left[z(k+1) - \hat{x}^{i}(k+1)\right]$, (5.4)
where, $z(k+1)$ is given by the measured travel time of PV2 to cover the $(k+1)^{th}$ subsection.

v. The a posteriori error variance is calculated using
$P^{ii}(k+1)=[I-K(k+1)]P^{i}(k+1)$, (5.5)

The a posteriori travel time estimate is taken as the final travel time estimate for every subsection.

5.2.2 Simple Adaptive Prediction Method

In the Base method, the values of *Q* (process disturbance co-variance) and *R* (measurement co-variance) are kept constant throughout. In the Simple Adaptive method, the values of Q and *R* are computed at every step by utilizing the travel time information from previous subsections. The Simple Adaptive Prediction Method consists of the following steps:

1) The parameter *a(k)* is calculated using equation (4.3).

2) The following steps are then applied recursively for subsection k =2 to *S* (taken to be 20 in this study), where the values of *Q(k)* and *R(k)* are treated as constants during the first *S* iterations:

i. The a priori travel time estimate is calculated using equation
ii. The a priori error variance is given by equation (5.2).
iii. The Kalman gain '*K*' is given by equation (5.3).
iv. The a posteriori estimate of the travel time is given by equation (5.4).
v. The a posteriori error variance is calculated using equation (5.5).

3) The following steps are then applied recursively for subsections *k*=S to N-1;

i. The a priori travel time estimate is calculated using equation (5.1).
ii. The a priori error variance is given by equation (5.2) with *Q(k)* being calculated using equation (5.3).
iii. The Kalman gain is given by equation (5.3) with *R(k+1)* being calculated using equation (5.5).
iv. The a posteriori estimate of the travel time is given by equation(5.4)
v. The a posteriori error variance is calculated using equation (5.5).

5.2.3 Enhanced Base Prediction Method

In the Enhanced Base method a weighted average of the travel times of PV1 and PV2 is taken as the measured travel time for that subsection. These weights are estimated based on the relative travel time values of PV1 and PV2 as discussed in section 4.3. In the present study, the value of d was taken as

$$d = \frac{\sum_{i=1}^{N} x_{pv1}(i)}{N} + \frac{\sum_{i=1}^{N} x_{pv2}(i)}{N}, \tag{5.6}$$

which is the sum of the means of the travel times of PV1 and PV2.

Once the initial conditions have been assumed, the following steps are applied recursively for subsections $k=2$ *to* N:

b) If the *kth* subsection belongs to Group 1, then
1) The parameter a(k) can be calculated as :

$$a(k-1) = \alpha \times \frac{x_{pv1}(k)}{x_{pv1}(k-1)} + (1-\alpha) \times \frac{x_{pv2}(k)}{x_{pv2}(k-1)}, \tag{5.7}$$

i. A priori estimate is calculated using equation (5.1).
ii. A priori error variance is calculated using equation (5.2).
iii. Kalman gain is given by equation (5.3).
iv. A posteriori estimate:

$$\hat{x}^{ii}(k) = \hat{x}^{i}(k) + K(k)\lfloor z(k) - \hat{x}^{i}(k) \rfloor, \tag{5.8}$$

where,

$$z(k) = \alpha \times x_{pv1}(k) + (1-\alpha) \times x_{pv2}(k), \tag{5.9}$$

v. A posteriori error variance is given by equation (5.5).

If the k^{th} subsection belongs to Group 2 then:

$$a(k-1) = x_{pv2}(k)/x_{pv2}(k-1)\ , \tag{5.10}$$

i. A priori estimate is given by equation (5.1).
ii. A priori error variance is calculated using equation (5.2).
iii. Kalman gain is given by equation (5.3).
iv. A posteriori estimate is given by equation (5.8).
v. A posteriori error variance is given by equation (5.5).

If the k^{th} subsection belongs to Group 3 then:

$$a(k-1) = x_{pv1}(k)/x_{pv1}(k-1)\ , \tag{5.11}$$

i. A priori estimate is given by equation (5.1).
ii. A priori error variance is calculated using equation (5.2).
iii. Kalman gain is given by equation (5.3).
iv. A posteriori estimate is given by equation
v. A posteriori error variance is given by equation (5.5).

The value of α for these three different cases is as follows:

Group 1: α=0,
Group 2: α=0.5,
Group 3: α=1.

In the prediction method of Group 2 the value of α = 0.5 is taken for simplicity.

5.3. Results and Performance Comparison

As discussed in the previous section, the performances of three algorithms were compared to select the most suitable algorithm for real time bus arrival time prediction. Accuracies of these methods were compared during different times of the day namely; the morning peak period, afternoon off-peak period and evening peak period. One week's data from 6th March to 12th March, 2011 from the 5C route was used for comparison. The graph in Figure 19 shows a typical travel time vs. time of the day plot for one day of the week under consideration. Similar plots for other days are provided in Appendix.

It can be seen that there are two distinct peak periods during the day, one in the morning from around 7:30 AM - 10:30 AM and another in the evening between 4:30 PM – 9:00 PM. It can also be observed that during afternoons the travel times are relatively low and also consistent. Hence, the time period between 10:30 AM – 4:30 PM is considered off-peak period for this study. A similar pattern was seen for all weekdays of the week under consideration. The algorithm requires that at least two buses (for PV1 and PV2) have traversed the same route before TV. The first TV is usually seen after 7:30 AM. Also the last trip of the day is from one of the terminus to the bus shelter (usually not a terminus). Last valid trips for the day are usually seen to start before 9:30 PM. Hence, the buses plying between 9 PM and 7:30 AM are not considered for the study.

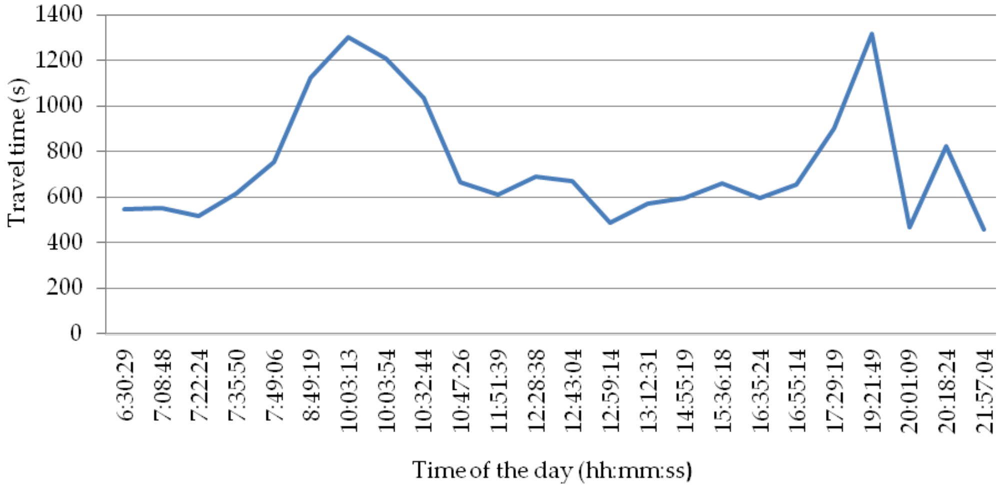


Figure 19: Travel time vs. Time of the day for 10th March, 2011

From this the time periods given in Table 10 were selected as peak and off-peak period.

Table 10: Distribution of peak and off-peak periods during the day

Period	**Time of the day**
Morning Peak	7:30 AM - 10:30 AM
Off-peak	10:30 AM - 4:30 PM
Evening Peak	4:30 PM - 9 PM

The accuracy of the prediction of the algorithms was separately analyzed for the above mentioned three time periods. This would help in identifying the better predictor during each of these time periods.

5.3.1 Accuracy of Prediction and Comparisons

The accuracy of the prediction schemes are quantified using the Mean Absolute Percentage Error (MAPE) given by

$$MAPE = \frac{\sum_{1}^{N} \frac{\left| x_{tvm}(k) - \hat{x}^{ii}(k) \right|}{x_{tvm}(k)} * 100}{N}, \quad (5.12)$$

where, $x_{tvm}(k)$ is the measured travel time of the test vehicle in the k^{th} subsection, N is the total number of subsections. The implementation was carried out by dividing the entire route into subsections of 200m each and MAPE in predicted travel times was calculated for each of this sub sections. Results obtained using all three prediction methods are discussed below. The MAPE values for 200 m subsections for the Base and Simple Adaptive methods are given in Table 9 under varying traffic flow conditions.

Table 9: MAPE for Base and Simple Adaptive methods for 200 m subsections

During morning peak period		
Date	**Base Method (%)**	**Simple Adaptive Method (%)**
6th Mar, 2011	32.00	31.83
7th Mar, 2011	23.71	23.15
8th Mar, 2011	25.56	25.47
9th Mar, 2011	31.06	28.65
10th Mar, 2011	27.56	27.70
11th Mar, 2011	34.62	35.78
12th Mar, 2011	26.68	26.90
During Afternoon Off-peak Period		
6th Mar, 2011	37.98	40.71
7th Mar, 2011	25.31	26.52
8th Mar, 2011	35.04	35.33
9th Mar, 2011	32.09	34.52
10th Mar, 2011	32.39	28.31
11th Mar, 2011	35.01	30.64
12th Mar, 2011	30.80	29.55
During Evening Peak Period		
6th Mar, 2011	25.90	25.64
7th Mar, 2011	27.10	28.38
8th Mar, 2011	28.92	28.69
9th Mar, 2011	35.22	33.12
10th Mar, 2011	39.24	39.83
11th Mar, 2011	36.19	38.95
12th Mar, 2011	31.66	33.43

As can be seen from the Table 9, no significant reduction in the MAPE values was observed on using the Simple Adaptive method. A similar comparison was carried out using the Enhanced Base Prediction method and the results are presented below.The MAPE values for 200 m subsections for the Base and Enhanced Base method are given in Table 11.

Table 11: MAPE for Base and Enhanced Base methods for 200 m subsections

During Morning Peak Period		
Date	**Base Method (%)**	**Enhanced Base Method (%)**
6th Mar, 2011	32.00	32.12
7th Mar, 2011	23.71	24.43
8th Mar, 2011	25.57	23.96
9th Mar, 2011	31.07	30.06
10th Mar, 2011	27.57	27.24
11th Mar, 2011	34.62	34.66
12th Mar, 2011	26.69	26.95
During Afternoon Off-peak Period		
Date	**Base Method (%)**	**Enhanced Base Method (%)**
6th Mar, 2011	37.98	40.51
7th Mar, 2011	25.31	27.022
8th Mar, 2011	35.04	35.26
9th Mar, 2011	32.09	35.57
10th Mar, 2011	32.39	30.81
11th Mar, 2011	35.01	30.46
12th Mar, 2011	30.80	29.05
During Evening Peak Period		
Date	**Base Method (%)**	**Enhanced Base Method (%)**
6th Mar, 2011	25.90	24.88
7th Mar, 2011	27.10	25.81
8th Mar, 2011	28.92	30.34
9th Mar, 2011	35.22	33.17
10th Mar, 2011	39.24	38.20
11th Mar, 2011	36.19	38.22
12th Mar, 2011	31.66	32.08

It can be seen that even in case of Enhanced Base Prediction method, the reduction in the MAPE value is not very significant. In fact, in some cases, the Enhanced Base Method yielded higher MAPEs than both Base and Simple Adaptive methods. One possible explanation could be that the choice of α = 0.5 was not be suitable for this study. It is possible that the value of α is different for each route, depending on the characteristics of the route. A route-specific optimized α value could enhance performance. Further studies are required to understand route characteristics that can affect the optimum value of α.

VI. APPLICATION DEVELOPMENT

The base method was selected for real-time implementation due to its simplicity and relatively comparable performance with the modified versions. This section lists the various steps that were followed for the development of the real time application.

6.1 Identified Issues

An application for real time bus arrival time prediction requires complete automation not in terms of data collection, quality control and running the algorithm alone, but should be capable to handle real field issues such as bus overtaking, rerouting enroute, break down etc.

Some of the specific field issues that were identified as leading to erroneous predictions or no prediction are:

- Choice of the test vehicle, for which the arrival time must be displayed
- Identification of buses plying along different routes
- Separation of vehicles plying in one direction vehicles from those plying the opposite direction
- Recognition of missing bus stop locations between two data points
- Calculation of difference in total distance travelled
- Identifying possible detours
- Detection of bus break-downs and traffic jams enroute
- Factoring overtake of the test vehicle by another bus along the same route

Such issues must be addressed in real-time during the daily functioning of the bus service. An application was developed to support real-time. This application addressed all the above mentioned issues and enabled use of the algorithm to obtain useful ETAs of buses at bus-stops as discussed below.

6.2 Application Development

The prediction of arrival for the test vehicle is based on the travel time of two previous vehicles that traveled in the selected route. Thus, the prediction was possible from the third vehicle onwards in a day. All the buses and bus-stops were identified with unique identification numbers. Each route thus had 2 unique routeIDs for the two directions of travel along that route.

Defined below are the attributes assigned to the buses. These attributes form a mutually exclusive and exhaustive set of states in which a bus can possibly be at any given point of time.

IDLE- The bus has not been assigned a route and its speed>0.
SLEEP- The bus has not been assigned a route and its speed=0.
JAM- The bus has been assigned a route and its speed=0.
CONTINUE- The bus has been assigned a route and its speed>0.

Each of these attributes was associated with a counter. If any of these counter limits for a bus was exceeded, then that bus was not considered for prediction.

Throughout the runtime of the application, a speed check was continuously performed for every bus. The first check ascertained that the bus has been assigned a route. If it had not been assigned a route then it was assumed to be either in 'IDLE' state or in 'SLEEP' state. If a bus was in 'IDLE' state, it meant that the bus had not been assigned a route but was still moving. In this case the bus was either moving to the terminus before the day's first trip or was moving after it was discarded from a route for some reason. A bus was assumed to be in 'SLEEP' state when it had been discarded after being assigned a route. This usually happens when the bus is kept at the bus shelter during night time.

Once the bus had been assigned a route, ideally the bus would continue moving and stop only at scheduled bus-stops. However, this was not the scenario in real field. The bus could take a detour, breaks down, is struck in a traffic jam, etc. To identify any of these deviations from ideal behavior, once the bus was assigned a route, continuous check was carried out to see if the bus was moving and was in the correct route. When speed became zero at any point of time, a counter ('JAM') was initiated, to record the stoppage time. When counter exceeded a predefined upper limit, it was assumed that the bus stopped functioning and the bus was discarded. As the arrival time for this bus must be kept displayed as long as the route remains assigned, it was important to choose the 'JAM' counter appropriately. As long as the bus was in route and its speed was non-zero a check was performed to see if the next stop has arrived. If this returned a false value, another counter called 'CONTINUE' was initiated. If 'CONTINUE' exceeded a predefined upper limit then the bus was discarded, as there is a possibility that the bus may have taken a detour. But if a stop was found, this information was updated and the corresponding prediction at that bus-stop for that particular bus ended.

For a bus that is plying along a route, it's 'Index Number *i*' is defined as the number of bus-stops the bus crossed in that route. For a bus plying between nth bus-stop and n+1 bus-stop along that particular route the Index number i = n. The 'Index Number *I*' of a bus-stop along a particular route is defined as the position number of the bus-stop along that route. So, it is possible for the same bus-stop to have different values of *I* along different routes. Following this, prediction for the next bus along this route was initated by checking the 'Index Number *i*' of different buses plying along that route. The bus with the highest current 'Index Number *i*' (< *I*) was chosen and prediction is carried out for that particular bus.

As and when the travel time data from the buses were received at the server, it was subjected to the above mentioned procedure. Many new attributes were added to the dataset before it was sent to the ATP algorithm. Table 12 below shows a sample output dataset after passing through the application. The input file for this application is the raw dataset seen in Chapter 3. The attribute 'Index Number *i*' of a bus was used to decide at which bus-stop the required information has to be shown.

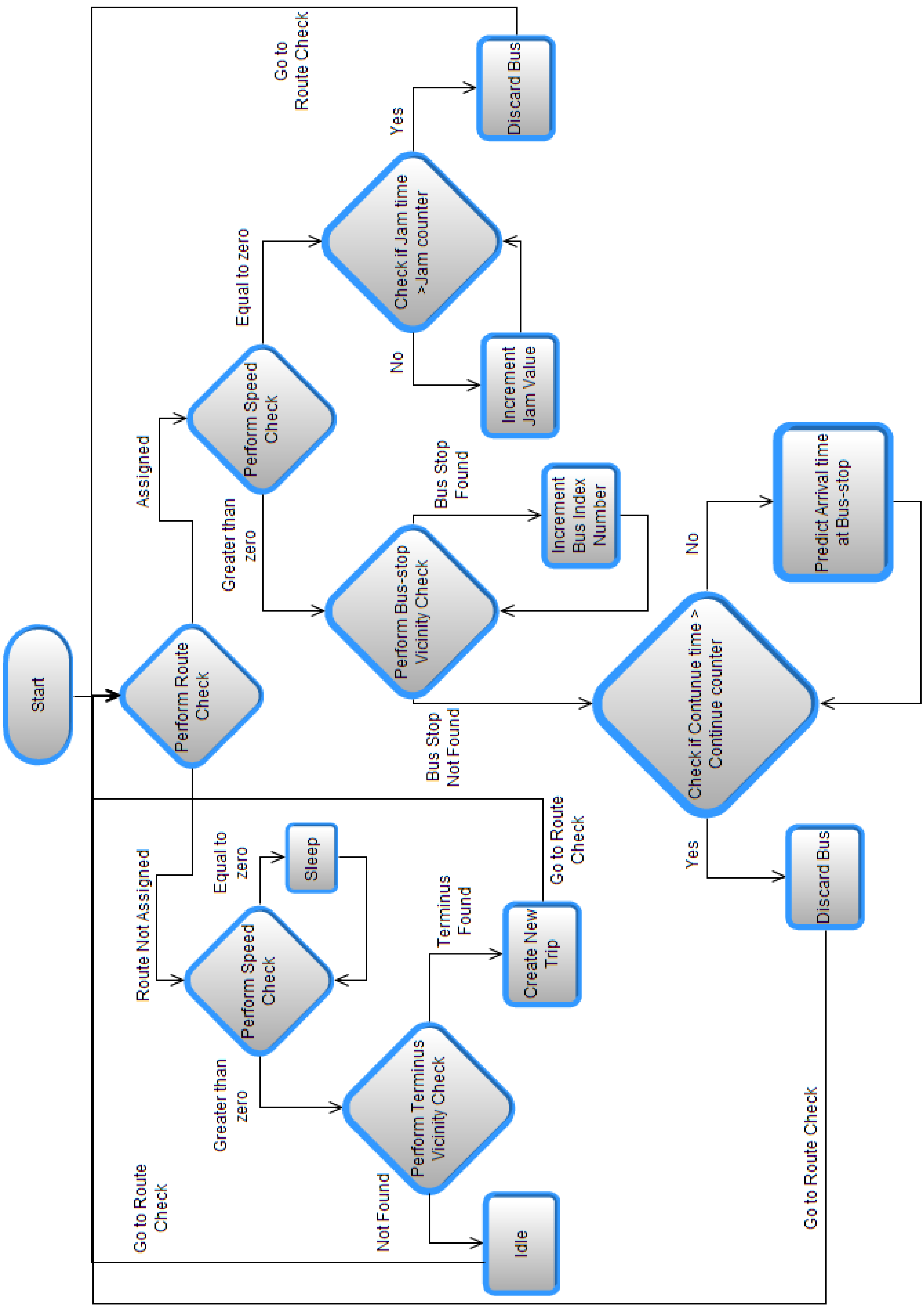


Figure 20: Flowchart showing the detailed schematic of the application

Route ID	Device ID	Latitude	Longitude	Clock Time	Index Number (i)	Idle	Continue	Jam	Sleep
333	iit04	13.029	80.2456	10:15:38	4	234	45	22	742
333	iit04	13.029	80.2456	10:15:43	4	234	45	23	742
333	iit04	13.029	80.2456	10:15:48	4	234	45	24	742
333	iit04	13.029	80.2456	10:15:53	4	234	45	25	742
333	iit04	13.029	80.2456	10:15:58	4	234	45	26	742
333	iit04	13.029	80.2456	10:16:03	4	234	45	27	742
333	iit04	13.0291	80.2456	10:16:08	4	234	45	28	742
333	iit04	13.0294	80.2456	10:16:13	4	234	45	29	742
333	iit04	13.0296	80.2459	10:16:18	4	234	46	29	742
333	iit04	13.0295	80.2462	10:16:23	4	234	47	29	742
333	iit04	13.0295	80.2465	10:16:28	4	234	48	29	742
333	iit04	13.0294	80.2468	10:16:33	4	234	49	29	742
333	iit04	13.0294	80.247	10:16:38	4	234	50	29	742
333	iit04	13.0294	80.2472	10:16:43	4	234	51	29	742
333	iit04	13.0293	80.2475	10:16:48	4	234	52	29	742
333	iit04	13.0292	80.2479	10:16:53	4	234	53	29	742
333	iit04	13.0292	80.2482	10:16:58	4	234	54	29	742
333	iit04	13.0292	80.2487	10:17:03	4	234	55	29	742
333	iit04	13.029	80.2492	10:17:08	4	234	56	29	742
333	iit04	13.029	80.2495	10:17:13	4	234	57	29	742
333	iit04	13.0289	80.2496	10:17:18	4	234	58	29	742
333	iit04	13.0291	80.2499	10:17:23	4	234	59	29	742
333	iit04	13.0295	80.2503	10:17:28	4	234	60	29	742

Table 12: Processed data after addition of new attributes

6.3 Issues Addressed

The methodologies followed to address the issues identified during the development of the application are described below.

6.3.1 Choice of Test vehicle

There may be many buses plying along a particular route at a given time. Of these, only the one that is expected to arrive earliest at a bus-stop has to be predicted at the bus-stop. This choice was made by taking into consideration, the index number 'i' of the bus. The bus with the highest 'i' ($< I$) value was chosen for prediction. When there were multiple buses with same index number 'i', only the bus whose index number changed first was used, as it was expected to arrive earlier.

6.3.2 Bus-Stop Identification

The latitude and longitude of each bus stop was accurately identified manually by physically recording them in a GPS unit. Bus stops were identified using markers within 50 m diameter of each bus stop. This was to ensure that a bus stop was not missed in case a bus passed the bus stop without stopping there. In such cases, at least one data point was to be obtained from around the bus stop (which was ensured by the 50 m diameter area around the bus stop) to update the 'Index Number i' of the bus. In the absence of this provision, the system would continue to predict time for bus arrival at the bus stop even after the bus passed that stop, instead of predicting the ETA of the next bus.

6.3.3 Identifying Direction

All routes were bi-directional i.e. buses plied in both directions along a particular route. Each direction was identified with unique 'routeID'. The bus was assigned the 'routeID' once it was detected at the bus terminus. It was assumed that once the bus started from a bus terminus it would proceed to the other terminus and its 'routeID' remained invariant until it reached the destination terminus. On arrival at the destination terminus, an opposite 'routeID' was assigned to the bus for the return journey.

6.3.4 Route Change and Missed Bus-Stop Location

One unique feature of the reported algorithm is that the discretisation was performed over space rather than over time unlike the previously mentioned KF models. Thus, the ETAs displayed at a bus-stop change according to the distance travelled by the bus. Whenever the bus started from a terminus along a particular route, a separate counter was initiated to measure the time taken for the bus to reach the next bus stop. This counter is re-set to zero once a bus-stop was encountered. Whenever the counter value exceeded a certain predetermined value (arrived from historic data) the bus along the route was discarded assuming that it had taken a detour/ or missed the bus-stop. The counter upper-limit was carefully chosen; if it was too big, inaccurate ETA would be displayed at the bus-stop for a longer time when the bus took a detour, and if it was too small then the bus could be discarded even while it was on route. The counter upper limit was chosen from historic data.

6.3.5 Bus Breakdown/ Traffic Jam

Whenever the bus stopped enroute (other than at bus-stops) for duration longer than usual, a JAM counter was initiated. After waiting for a specified time (calculate from jam durations collected from the historical GPS data) that particular bus was rejected for that trip in the route assuming that the bus may have broken-down or encountered a traffic jam.

6.3.6 Overtaking

When a bus (either PV or TV) was overtaken by another bus in the same route, appropriate corrections were incorporated by taking into consideration the 'Index number *i*' of the bus. Since any two buses traveling between bus stops n and n+1 would have the same *i*=n, the effect of correction could only be seen from n+2 bus-stop onwards. This is one of the limitations of the application.

6.3.7 Correction for Difference in Distance Travelled

The reported algorithms estimate and predict the travel time of *TV* using travel time data collected from *PV1* and *PV2*. The number of data points collected between two terminals of the route from two different buses are usually not the same due to reasons such as unequal length of path traversed, difference in time taken to cover the entire route etc. The original study by Vanajakshi et al. (2008) truncated the last few data points of the longer trip to spatially equalize the trip lengths. This can lead to inaccuracy in the predicted time. Additionally, due to the (marginally) unequal distance traversed between two bus stops there are errors like rapid decrease in the arrival time prediction at the intermediate bus-stops especially towards the end of the route. This happens due to the misalignment of the bus stops along the route.

The methodology of data input into the algorithm was modified to circumvent the errors arising from misalignments. Data sets from *PV1* and *PV2* were fed into the Kalman filter, which plotted and curve-fit these values. According to the modified methodology, data sets were separately collected for different stretches along the route and the distance values were scaled between consecutive bus stops with respect to the smaller of the two data sets collected between these two stops. These scaled values were combined to reconstruct the full routes for both *PV1* and *PV2* trips, which were then fed into the *KFT* algorithm. This obviated the need to truncate any data set, and led to a more realistic travel time profile of the bus along the route. This also reduced sudden jumps in the predicted arrival times as the buses have similar travel time profiles between two stops.

6.4 Evaluation of Performance of the Application

The accuracy of prediction of the real-time application was tested for buses traveling along the 5C route during different times of the day. Trips made during the morning peak, evening peak and off-peak period were separately analyzed. ETA for every bus was updated every 10 seconds. The predictions were implemented as intervals. The different intervals into which the present status of the buses plying along a particular route (5C here) were:

- If the ETA was greater than 15 minutes then the message displayed was
  'Route: 5C Estimated Arrival Time: Greater than 15 mins'

- If the ETA was less than 15 minutes then the message displayed was
  'Route: 5C Estimated Arrival Time: Within 15 mins'

- If the ETA was less than 10 minutes then the message displayed was
  'Route: 5C Estimated Arrival Time: Within 10 mins'

- If the ETA was less than 5 minutes then the message displayed was
  'Route: 5C Estimated Arrival Time: Within 5 mins'

- If the ETA was less than 3 minutes then the message displayed was
  'Route: 5C Estimated Arrival Time: Within 3 mins'

- If the ETA was less than 1 minute then the message displayed was
  'Route: 5C Estimated Arrival Time: Within 1 min'

- If insufficient information was available, then the message displayed was
  'Route: 5C Insufficient Information, Waiting...'

This information (range/time interval of arrival) is more useful to daily commuters like professionals and students who are on a tight schedule. If the bus is expected to arrive in more than 15 – 20 minutes the commuter may choose to take alternate means of transport, especially in a busy city like Chennai. As the time of arrival of the bus approaches, the window of prediction also decreases for more accurate information.

The accuracy of these predictions was tested in real time as follows.

If the information displayed is

'Route: 5C Estimated Arrival Time: Within 15 mins',

the information is correct as long as the bus take less than 15 minutes and more than 10 minutes to arrive at the bus-stop. Once the bus goes to below 10 minutes zone the display must change to:

'Route: 5C Estimated Arrival Time: Within 10 mins'

If the display shows 'Within 15 mins', when it takes less than 10 minutes or more than 15 minutes to arrive, inaccurate information has been conveyed. Thus, accuracy of the application can be calculated as:

$$\text{Prediction Range Accuracy}\,(\%) = \frac{\text{No. of times correctly predicted}}{\text{Total No. of Predictions made in the range}} \times 100 \qquad (6.1)$$

Using this measure, the performance of the Base Prediction algorithm and Historical Averaging method were compared. Table 6.2 shows a comparison of the accuracy of prediction by the above methods under varying traffic conditions. The number in brackets denotes the average error (in seconds) of the predicted value.

It can be seen from the results that the Base Prediction Model is performing better than the Historical Averaging Model in terms of real-time accuracy. Remaining errors can be mainly attributed to communication delay between the GPS/GPRS unit installed in the bus and the server. This delay occurs due to low GSM connectivity in certain areas.

The method used for the prediction of arrival times is different from the existing method of prediction being used by MTC, Chennai and other such corporations. Performance evaluation of the existing bus arrival prediction in bus stop was also carried out and is described in the following section.

Table 13: Comparison of the real-time performance of the application

During Morning Peak Period

Range of Prediction (mins)	Accuracy of Base Prediction Model	Accuracy of Historical Averaging Model
10 < ETA < 15	85.71 % (30.94)	85.71 % (20.40)
5 <ETA < 10	88.46 % (15.73)	85.18 % (21.92)
3 < ETA < 5	57.14 % (44.91)	46.66 % (57.06)
1 < ETA < 3	85.71 % (57.50)	66.66 % (43.50)
ETA <1	100 % (14.40)	100 % (16.25)

During Afternoon Off-peak Period

10 < ETA < 15	88.88 % (36.91)	82.75 % (36.22)
5 <ETA < 10	100 % (31.61)	100 % (24.76)
3 < ETA < 5	84.61 % (14.75)	70.58 % (20.25)
1 < ETA < 3	100 % (10.45)	87.5 % (12.92)
ETA <1	100 % (8.28)	100 % (10.67)

During Evening Peak Period

10 < ETA < 15	100 % (44.74)	100 % (55.66)
5 <ETA < 10	87.88 % (23.04)	85.29 % (34.61)
3 < ETA < 5	87.5 % (17.36)	87.5 % (27.31)
1 < ETA < 3	100 % (8.45)	100 % (28.50)
ETA <1	85.7 % (5.67)	75 % (6.00)

6.6 Summary

This section discussed the real time implementation and the issues faced during this stage. It can be seen the raw data which is received at the server must be processed for various scenarios before it can be sent to the algorithm for real time prediction. The new attributes added to the algorithm determine the status of the bus and are used to decide whether data being transmitted by a particular bus has to be sent to the prediction algorithm or not. If at any stage during the trip any of the counter's limits is crossed, the prediction for that particular bus is stopped, and the next active bus along the route is sought. This process of receiving data from buses, assigning attributes, analyzing them and predicting their ETAs at selected bus-stop is performed throughout the day. One limitation of this process is that it cannot provide ETAs for the first two buses of the day, as it needs data from two previous trips along the same route on the same day to start the prediction. Once such a complete system is ready, the performance evaluation was carried out and attempts to improve the prediction accuracy are ongoing. Any algorithm which is found to perform better than what is used will be replaced in the back end of the application. Two such modifications are identifying and incorporating suitable input parameters and exploring other prediction methods and are discussed in the next two sections.

VII. PROTOTYPE DEVELOPMENT

Various information dissemination options were evaluated. Three different methods were found to be suitable for the present application, namely Bus-Stop Variable Message Signs (VMS), Kiosk at bus stops and Web Based information system. The prototypes for each were developed/tested as described in subsequent sections.

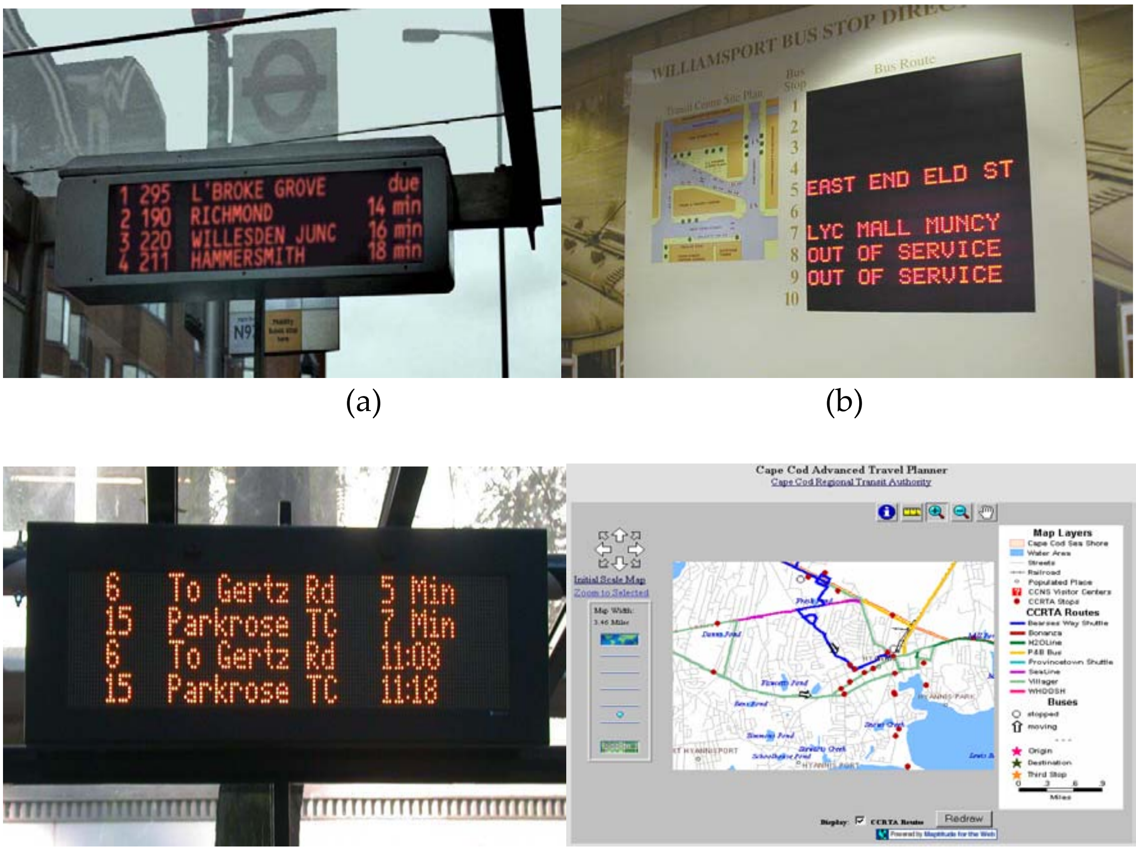
7.1 Popular Bus Arrival Information Dissemination Methods

A brief discussion on the various methods of dissemination of information is presented in this section.

2.3.1. Message Boards:

The most common medium used for the dissemination of real-time bus arrival information is the electronic sign also known as dynamic message sign(DMS), or variable message signs (VMS) located at the bus stops. Of all the electronic signs available the most prevalent are light emitting diodes(LED) and liquid crystal dispaly (LCD)signs. These displays are used to provide countdown information [Figure 21a], real-time location [Figure 21b], real-time bus arrival times [Figure 21c] and location of bus on a map [Figure 21d].

The communications technologies that are used most often to transmit information to electronic signs at bus stops are cellular communications [mostly cellular digital packet data (CDPD)] and conventional telephone lines.



(a) (b)

(c) (d)

Figure 21: Visual Displays to provide information

These display units are available in various sizes and are flexible in terms of display messages since they can be controlled remotely through GPRS. The display unit used in this study has an in-built GPRS module, which can receive information from the server in real time. Information is transmitted from the server at regular intervals (usually in less than a minute) and this information can be displayed at the bus-stop. Usually the LED matrix only displays numerical and text messages. The display can be made in any language and usually the display used in India show the display both in English and local language.

Most metropolitan transport corporations that are attempting to incorporate bus arrival information systems use such display units. For example, the display units used by MTC shows the current time and the time at which the next bus is expected to arrive. Figure 22 shows MTC's display format with its various information fields: the route number of the bus, type of the bus (deluxe in this case) and the final destination of the bus route along with the expected arrival time of the next bus.

Figure 22: MTC's display format

The format of display suggested in the present study is different from MTC's format to simplify it and make it more informative. In order to keep the information provided short and simple, the modified format is used and is shown in Figure 23.

Route Number	Expected Time of Arrival	Time Now
5C	Within 15 mins	04:42 PM

Figure 23: Modified display format

7.1.2 Kiosk at Bus Stops

A computer kiosk houses a computer terminal that often employs custom software designed to function flawlessly while preventing users from accessing system functions. Computerized kiosks may store data locally, or retrieve it from a computer network. Some computer kiosks provide a free, informational public service, while others serve a commercial purpose (mall kiosk). Touchscreens, trackballs, computer keyboards, and pushbuttons are all typical input devices for interactive computer kiosk.

Interactive kiosks can be used at bus-stops to provide commuters with useful information. The user enters the particulars of the information he/she is seeking. For example, a commuter can know the arrival time of a particular bus number at the bus-stop by entering the route number and the bus-stop at which the arrival time is required. The computer fetches data from the server in real-time and thus each query gives the latest available information.

Kiosks are different from display units as they can also be used to advertise. They are interactive and more personalised compared to the bus-stop display units. Figure 24 (a) shows a small sample kiosk at a bus-stop, showing waiting time at the bus-stop for different routes passing through that particular bus-stop. Another option for Kiosk display, where map application can be run, is to display the location of bus on a map. Figure 24 (b) shows a sample display for this.

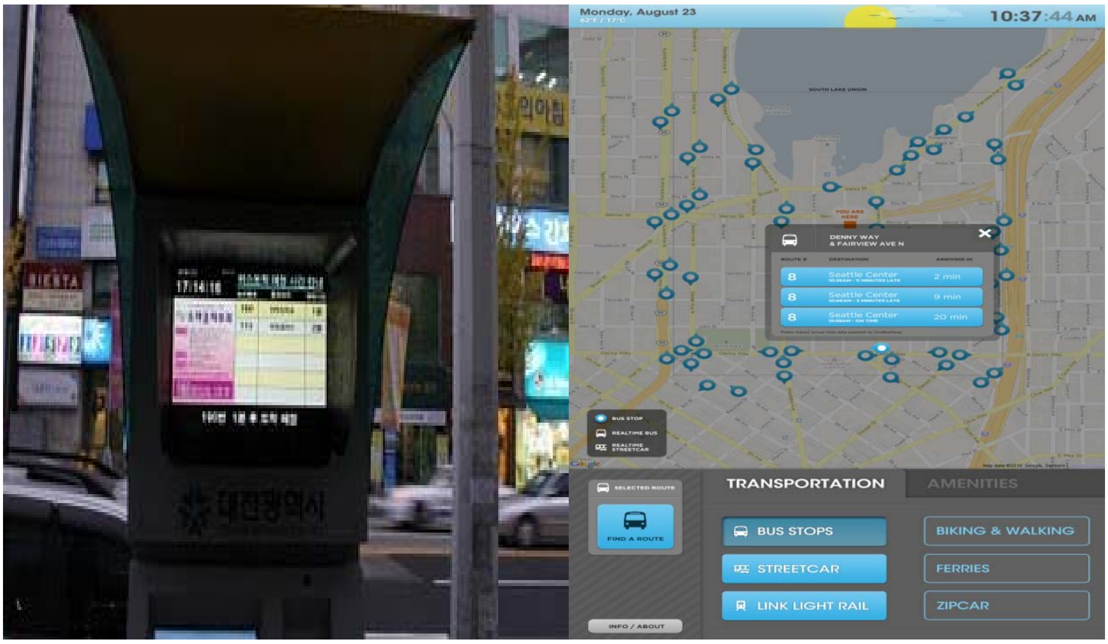


Figure 24 (a): Kiosk showing waiting times (b): Kiosk showing bus location on map

As a part of the present project a similar Information Dissemination Unit (IDU) was developed. The IDU is a suitably customized simple computer unit which can either be used as a display board at the bus-stops or can be used as an interactive kiosk unit. Each IDU consists of a simple low-end computer with a GPRS module attached to it for data transfer. The GPRS module is the client to the central server where all the data is processed and sent to the IDU. The application running on the IDU runs a separate instance for each bus-stop and queries the central server for updated bus-stop data, every 10 seconds. These IDUs can also be used to display advertisements with alternating bus arrival information.

The variable names 'busstopid', 'routeid', 'bus_indexno', 'bus_lat', 'bus_lon', 'arrival_time' represent the bus-stop number, route number, bus index number '*i*', bus' current latitude, bus' current longitude, arrival time of the next bus respectively. Figure 25 shows a screenshot of the Bus-Stop IDU schema. Along with the arrival time prediction of the bus, it also shows the current location of the bus between two bus-stops. The blue triangle points to the bus-stop at which the screen is installed indicating the commuter's location. The bus symbol indicates where the bus is right now. For multiple routes, the display is refreshed for every 10 seconds to show the latest available information along each route. Each data packet sent to the Bus Stop Display IDU is in the following format: 'busstopid, routeid, bus_indexno, arrival_time'

Figure 25: Bus-Stop Display IDU application screenshot

The Map Display Bus-Stop IDU application is shown in Figure 26. This application shows the current location of different buses in the vicinity of the bus-stop and their arrival times at the bus-stop. This information is updated every 10 seconds. Each data packet sent to the Bus Stop Display IDU with map display is in the following format: 'busstopid, routeid, bus_lat, bus_lon, arrival_time'

Figure 26: Bus-Stop Display IDU application with map display

Another option for IDU is to make it interactive and dynamic, where map applications can be run. The main idea is to display the dynamic location of the bus on a map. Figure 27 shows an IDU screenshot where the user can choose route and then a map with arrival time and current locations is displayed. Such an application can be used at a bus-stop where the commuter can choose a bus route of interest. The map application then queries the central server every 10 seconds, for the details of all the buses currently moving on that route. Each data packet with details of the next arriving bus, sent to the Bus Stop Kiosk IDU is in the following format: 'busstopid, routeid, bus_lat, bus_lon, arrival_time'

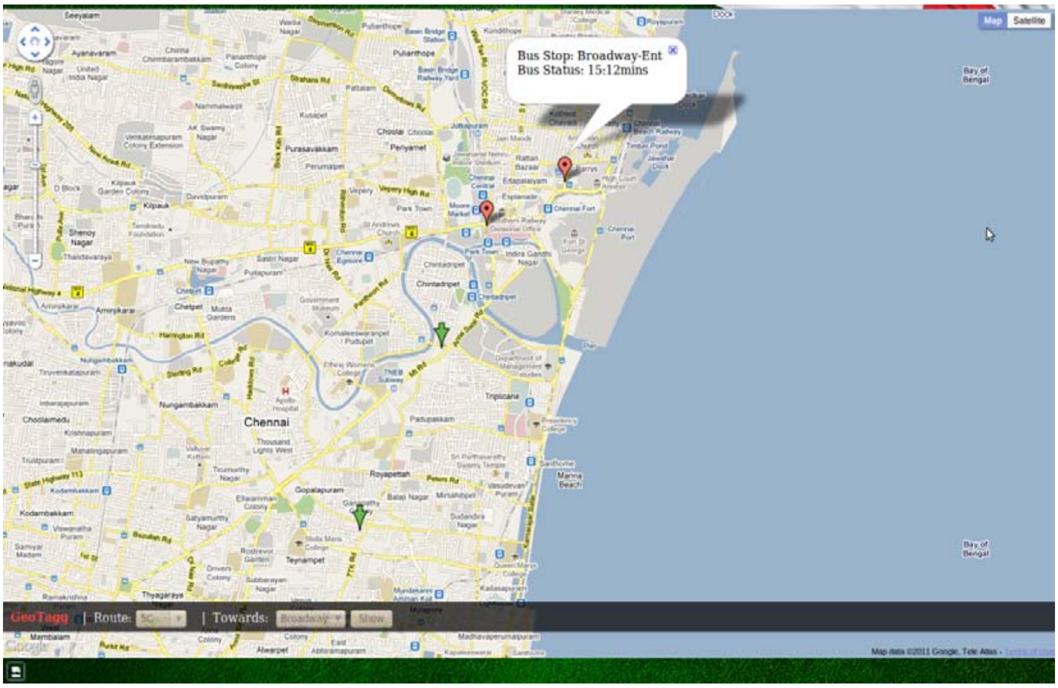


Figure 27: Kiosk IDU application screenshot with route selection option

For each query made by the Kiosk IDU multiple data packets are sent back by the central server to show the current location of all the buses along the route, yet to reach that bus-stop. Each of these data packets sent to the Bus Stop Kiosk IDU is in the following format: 'busstopid, routeid, bus_lat, bus_lon'

Bus-stops are indicated by red markers and buses by green arrows. When the user clicks on a bus-stop then the ETA of the next buses arriving at that bus-stop are displayed.

The IDU developed provides greater information in a cognitively ergonomic format to aid the commuter's decision making skills. The IDU can be suitably customized and either be used as a display board at the bus-stops or as an interactive kiosk unit as per the requirement. Hence the objective of developing a reliable and robust system for the implementation of the algorithm was achieved.

7.1.3 Web Based Information Service

Web-based bus arrival information dissemnation is also a popular route. Usually an interactive website helps users select the desired route and bus stop to get the real-time bus arrival and location details. Website information dissemination allows for travel planning from the comfort of one's office or home and saves time. A webbased information system used in Chicago is shown below.

Figure 28 shows one such example of a web service.

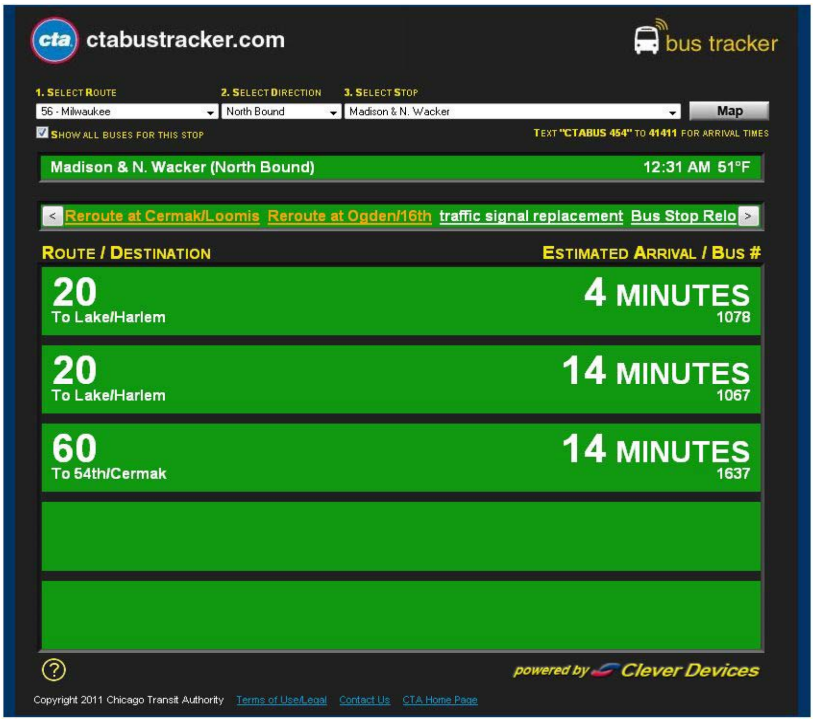


Figure 28: Chicago Transit Authority website providing real-time arrival details

A similar web-application was developed in this study to inform commuters of their expected wait time at the bus stops. Providing such information before they start the trip can help avoid the wait time at bus stops. The web based application developed for this purpose can be interactively used by commuters to check the expected arrival time of their desired bus at the required bus stop. The web site provides the users with information on the current location of the bus and it's ETA at any chosen bus-stop. This website was based on Google Maps. Figure 29 shows a screenshot of the site. The website provides the flexibility of choosing the bus-stop at which the user wishes to board the bus. Unlike kiosk display, the website provides the estimated arrival time at the bus-stop even when the ETA is greater than 15 minutes. This change was incorporated to help the user decide when she/he needs to go to the bus-stop to board the desired bus.

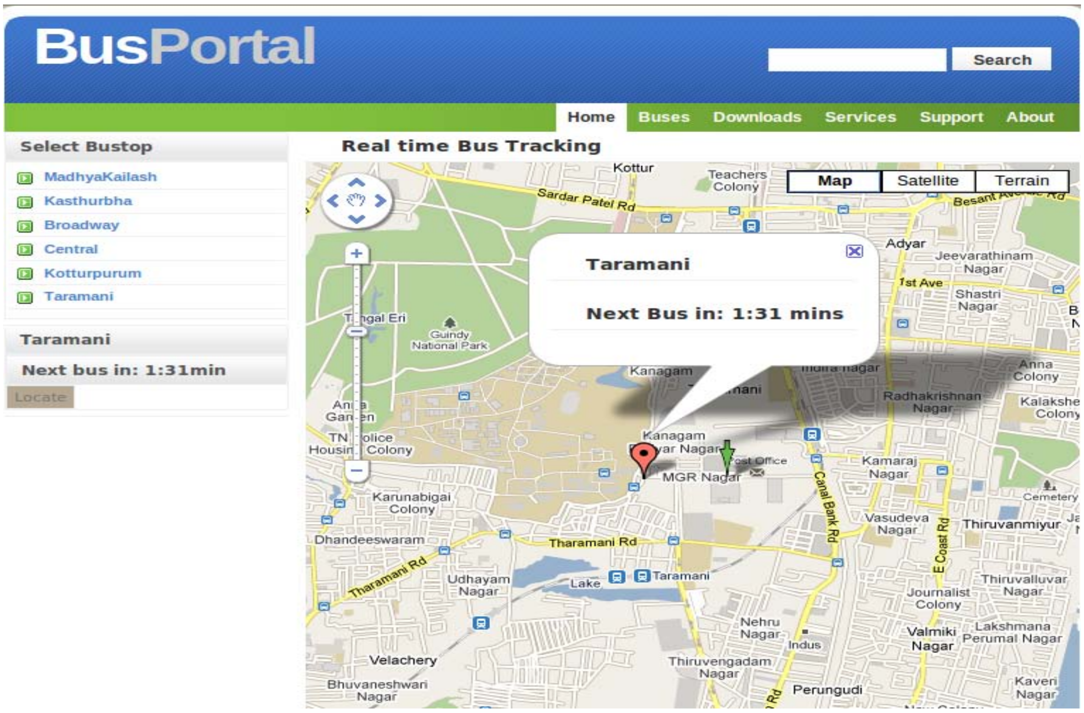


Figure 29: Google Maps-based website showing the bus location and ETA

The expected arrival time of the selected route for the selected bus stop for chosen time of the day can also be provided to users through SMS. For this kind of a service users are required to subscribe beforehand. Once registered, users get regular SMS during different times of the day as per their choice.

These integrated prototypes are being evaluated and once the performance is assured, they can be implemented in field.

VIII. CONCLUSION

8.1 Concluding Remarks

One of the major factors responsible for the success or failure of any public transport service is its reliability. One way of improving the reliability is to provide the passengers with accurate and reliable information regarding the service.

For public bus services, dissemination of accurate information regarding current bus location, arrival time at bus stops, etc. to the traveler will help to reduce the uncertainty and waiting time involved in the use of bus service system and enhance efficiency. Improvements in service efficiency and reduction of waiting time will attract more passengers to this means of public transport, thus easing the traffic burden on the roads. Such implementations will also benefit public transport authorities in helping them plan routing and scheduling activities. Thus, indigenous techniques are required for accurate forecasting of vehicle travel times.

Most studies in the area of bus travel time/arrival time prediction are based on homogeneous traffic conditions. The traffic conditions in countries like India are different with its heterogeneity and lack of lane discipline and hence predicting the bus arrival time is not simple. Only a limited number of studies have been reported for such heterogeneous disorganized traffic conditions. The present study is an attempt to develop a real time bus arrival prediction system specifically suited to Indian traffic conditions.

Towards this development, analyses were first performed using the GPS data collected offline from buses running in route numbers 21L and 21G collected manually. Once the performance was found to be acceptable, online data integration was carried out for route numbers 5C and 19B in Chennai, India. In order to have a real time automated application, it was required to develop an automated data collection filtration and analysis system without manual intervention. Numerous issues were identified and addressed during real-time implementation of the model. Issues such as effect of traffic jam, overtaking among the buses on the same route, bus breakdown, abrupt changes in bus routes, etc. which are specific to Indian conditions were taken into consideration.

A model based prediction scheme was used and the Kalman Filtering Technique was adapted to estimate and predict the travel time of buses. Different modifications to the system were attempted for improving the performance and those that were improving the performance were incorporated into the model. The study also developed prototype systems for information dissemination which can be readily used for field implementation. Prototypes for three different information dissemination modes were developed, viz. bus stop VMS display, Kiosk display at bus stops and web based application. An advanced kiosk display system was developed considering specific requirements of users. Such a system has advantages of cost efficiency and provides greater information in a cognitively ergonomic format to aid the commuter’s decision making skills.

This project entailed cooperative effort by experts from various organizations. Project management was a challenging task given the variety in expertise of participants. The main technical obstacle was data reception quality from the

GPS/GPRS unit to the server. The quality of data transmitted to the server was also dependent on the vendor providing this service. Real-time data loss occurred whenever there was low GPRS connectivity. Due to dense tree coverage on the campus roads, especially on Bonn Avenue, GSM connectivity was a major interruption to the proper functioning of this system on campus. Also, the service was heavily vendor dependent since most of the applications are proprietary leading to less control on the hardware and software that captures data from GPS units.

8.2 Further Work

Though the application developed covers all aspects of real-time implementation, there is still scope for improvement. Search for better algorithms for more accurate prediction is an open ended problem. In the present study due to limitations of the KFT based models, the data from TV was not used in the algorithm and was kept for validation purpose alone. New algorithms that would use real-time data from the test vehicle must be developed. This will clearly improve the prediction accuracy as the input variables then will represent a scenario much closer to real-time occurrences. This shall be useful when the time headway between two buses is high. In such a case the two consecutive buses can experience totally different traffic conditions. Modification of prediction algorithm will be an ongoing process. Evaluation of the developed system is also ongoing. Transferability and scalability of the system need to be taken into account. Once these are carried out, the system will be ready for field implementation. Field implementation and evaluation in the real world scenario will also be carried out.

Publications

Journal Publications

1. Padmanaban, R.P.S., Divakar, Lelitha Vanajakshi And Subramanian, S.C., AN Automated Real Time Bus Arrival Time Prediction System Incorporating Delays, *IET Journal on Intelligent Transportation Systems*, Vol. 4, No. 3, pp. 189–200, 2010.
2. Vasantha Kumar, Lelitha Vanajakshi, Bus Travel Time Pattern Identification and Prediction, Transport, (Institution of Civil Engineers UK), Issue TR1, Pages 1–10 http://dx.doi.org/10.1680/tran.12.00001, 2012.
3. Vasantha Kumar, Lelitha Vanajakshi, Urban Arterial Travel time Estimation using Buses as Probes, The Arabian Journal for Science and Engineering, Volume 39, Issue 11, pp 7555-7567, 2014.
4. Vasantha Kumar, Krishna C. Dogiparthi, Lelitha Vanajakshi and Shankar C. Subramanian, Integration Of Exponential Smoothing With State Space Formulation For Bus Travel Time And Arrival Time Prediction, Transport (Taylor and Francis), 2015, DOI: 10.3846/16484142.2015.1100676.
5. Anil Kumar, Snigdha Mothukuri, Lelitha Vanajakshi and Shankar C. Subramanian, An Analytical Approach to Identify the Optimum Inputs for a Bus Travel Time Prediction Method, Transportation Research Record: Journal of the Transportation Research Board, No. 2535 on pages 25–34, 2015. DOI 10.3141/2535-03.

Conferences

1. Padmanabhan, R.P.S., Lelitha Vanajakshi and Subramanian, S.C., 'Automated Delay Identification for Bus Travel Time Prediction towards APTS Applications', *IEEE* sponsored International Conference on Emerging Trends in Engineering and Technology, India, *2009*.
2. Padmanabhan, R.P.S., Lelitha Vanajakshi and Subramanian, S.C., 'Dwell Time Incorporated Bus Travel Time Prediction Model For APTS Applications', *IEEE Intelligent vehicles symposium, China, 2009*.
3. Akhilesh Koppineni, Krishna Chaitanya, Sidharth, Lelitha Vanajakshi, Development of a Bus Arrival Time Prediction System for Indian Traffic Conditions, *National Conference in Recent Advances in Civil Engineering,* Varanasi, India, 2011.
4. Vasantha Kumar, And Lelitha Vanajakshi, Application of Multiplicative Decomposition and Exponential Smoothing Techniques for Bus Arrival Time Prediction, *Transportation Research Board,* National Research Council, Washington, D. C., 2012.
5. Akhilesh Koppineni, Krishna Chaitanya, Sidharth, Lelitha Vanajakshi, Development of an Automated Bus Arrival Time Prediction System under Indian Traffic Conditions, *Transportation Research Board,* National Research Council, Washington, D. C., 2012.
6. Astitva Tripathi, Shankar C. Subramanian, Lelitha Vanajakshi, An Enhanced Bus Travel Time Prediction Method For Heterogeneous Traffic, 9th European ITS Congress, Dublin, June 2013.
7. Anil Kumar, Lelitha Vanajakshi and Shankar C. Subramanian, Day-Wise Travel Time Pattern Analysis under Heterogeneous Traffic Conditions, 2nd Conference of Transportation Research Group of India (CTRG), Procedia - Social and Behavioral Sciences, Volume 104, 2 December 2013.
8. Anil Kumar, Lelitha Vanajakshi and Shankar C. Subramanian, Pattern-Based Bus Arrival Time Prediction Under Heterogeneous Traffic Conditions, *Transportation Research Board Annual Conference,* National Research Council, Washington, D. C., 2014.
9. Bachu anil Kumar, Lelitha Vanajakshi, Use of Data Mining for Improved Bus Travel Time Prediction, 4th International Chinese European Workshop (CEW), 2016, Delft, Netherlands, 29th June to 1st July.

References

[Ahmed and Cook 1979] AHMED, M. S., and COOK, A. R. (1979): "*Analysis of Freeway Traffic Time-Series Data by Using Box-Jenkins Techniques*", Transportation Research Record 722, Transportation Research Board, 1979.

[Al-Deek *et al.* 1998] AL-DEEK, H., M. D'ANGELO and M. WANG (1998): "*Travel Time Prediction with Non-Linear Time Series*". Proceedings of the ASCE 1998 5th International Conference on Applications of Advanced Technologies in Transportation, Newport Beach, CA, pp.317-324.

[Bin *et al.* 2006] BIN, YU, ZHONGZHEN, Y., and BAOZHEN, Y. (2006): "*Bus arrival time prediction using support vector machines*". Journal of Intelligent Transportation Systems, 10(4), 151–158.

[Brooks et al. 2006] Brooks, P. Metrics for IT Service Management. Van Haren Publishing, Netherlands, 2006.

[Chen *et al.* 2004] CHEN M., LIU X.B. and XIA J.X. (2004): "*A dynamic bus arrival time prediction model based on APC data*". Computer Aided Civil and Infrastructure Engineering, 19:364 - 376.

[Chien *et al.* 2002] CHIEN, S.I.J., DING, Y. and WEI, C. (2002): "*Dynamic Bus Arrival Time Prediction with Artificial Neural Networks*" Journal of Transportation Engineering, Volume 128, Number 5, pp. 429-438.

[Chien and Kuchipudi 2003] CHIEN, S.I.J. and KUCHIPUDI, C.M. (2003): "*Dynamic Travel Time Prediction with Real-Time and Historic Data*" Journal of Transportation Engineering, Volume 129, Number 6, pp. 608-616.

[Hoogendoorn and Van Lint 2008] HOOGENDOORN, S. and VAN LINTS, H (2008): "*Reistijdvoorspellingen en reistijdbetrouwbaarheid*". NM Magazine, 2008, 3.

[Jeong 2004] JEONG R.H. (2004): "*The Prediction of Bus Arrival time Using Automatic Vehicle Location Systems Data*", A Ph.D. Dissertation at Texas A&M University

[Kalman 1960] KALMAN R.E. (1960): "*A new approach to linear filtering and prediction problems*", Transactions of the ASME–Journal of Basic Engineering, 82 (Series D), 1960, pp.35–45.

[Kirby *et al*. 1997] KIRBY, H., DOUGHERTY, M., and WATSON, S. (1997): ''*Should we use neural networks or statistical models for short term motorway traffic forecasting?*'' Int. J. Forecasting, 13(1), 43–50.

[Lee and Fambro 1999]LEE, S., and FAMBRO, D.B. (1999): "*Application of Subset Autoregressive Integrated Moving Average Model for Short-Term Freeway Traffic Volume Forecasting*", Transportation Research Record 1678, Transportation Research Board, 1998.

[Lin and Zeng 1999] LIN, W.H. and ZENG, J. (1999): "*Experimental Study of Real-Time Bus Arrival Time Prediction with GPS Data*" Transportation Research Record, 1666, pp. 101-109.

[Patnaik *et al.* 2004] PATNAIK, J., S.CHEIN, and A.BLADIHAS (2004): "*Estimation of Bus Arrival Time, Using APC Data*". Journal of Public Transportation, Vol. 7, No. 1, pp. 1–20. 7

[Ramakrishna *et al.* 2006] RAMAKRISHNA, Y., RAMAKRISHNA P, SIVANANDAN R (2006): "*Bus Travel Time Prediction Using GPS Data*",proceedings, Map India 2006

[Ricknagel 2001] RECKNAGEL, F. (2001): "*Applications of machine learning to ecological modeling*". Ecol. Model. 146, 303–310.

[Shalaby and Farhan 2004] SHALABY, A., and A. FARHAN (2004): "*Bus Travel Time Prediction for Dynamic Operations Control and Passenger Information Systems*". CD-ROM. 82nd Annual Meeting of the Transportation Research Board, National Research Council, Washington D.C.

[Thomas *et al.* 2010] T. THOMAS, W.A.M. WEIJERMARS and E.C. VAN BERKUM (2010): "*Predictions of Urban Volumes in Single Time Series*", IEEE Transactiuons on intelligent transportation systems,vol. 11, no1, pp. 71-80

[US Department of Defense 1983]
http://en.wikipedia.org/wiki/Global_Positioning_System

[Vanajakshi *et al.* 2008] VANAJAKSHI L., SUBRAMANIAN S.C., SIVANANDAN R. (2008): "*Travel time prediction under heterogeneous traffic conditions using global positioning system data from buses*". IET Intelligent Transport Systems, 2008, Vol. 3, No. 1, pp. 1–9.

[Wall and Dailey 1999] WALL, Z., and D. J. DAILEY (1999): "*An Algorithm for Predicting the Arrival Time of Mass Transit Vehicles Using Automatic Vehicle Location*

Data". CD-ROM. 78th Annual Meeting of the Transportation Research Board, National Research Council, Washington D.C

[William *et al.* 1998] WILLIAM, B. M., DURVASULA, P. K., and BROWN, D. E. (1998): "*Urban Freeway Traffic Flow Prediction: Application of Seasonal Autoregressive Integrated Moving Average and Exponential Smoothing Models*", Transportation Research Record 1644, Transportation Research Board, 1998.

[Williams and Hoel 2003] WILLIAMS, B. and HOEL, L. (2003): "*Modeling and forescating vehicle traffic flow as a seasonal arima process: Theoretical basis and empirical results*". Journal of Transportation Engineering, 129(6):664–672.

[WORLD BANK GROUP 2007]
http://web.worldbank.org/WBSITE/EXTERNAL/COUNTRIES/SOUTHASIAEXT/EXTSARREGTOPTRANSPORT/0,,contentMDK:20703625~menuPK:868822~pagePK:34004173~piPK:34003707~theSitePK:579598,00.html

[Yang 2005] YANG, J.S. (2005): "*Travel time prediction using the GPS test vehicle and Kalman filtering techniques*". In American Control Conference, Portland, Oregon, USA.

[Yasdi 1999] YASDI, R., (1999): "*Prediction of road traffic using a neural network approach*". Neural Computing and Applications 8, pp. 135–142

[Zheng *et al*. 2006] ZHENG, W., LEE, D.-H., and SHI, Q. (2006): "*Short-Term Freeway Traffic Flow Prediction: Bayesian Combined Neural Network Approach*", Journal of Transportation Engineering, 132(2), 114–121.

[Vanajakshi et al. 2004] Vanajakshi, L., and L. R. Rilett, 'A Comparison of the Performance of Artificial Neural Networks and Support Vector Machines for the Prediction of Vehicle Speed', *IEEE Intelligent Vehicles Symposium*, 2004, pp. 194-199.

[Vanajakshi et al. 2007] Lelitha Vanajakshi and Rilett, L. R. Support Vector Machine Technique for the short term prediction of Travel Time, *IEEE IntelligentVehicles Symposium*, pp. 600 – 605, Istanbul, Turkey, 2007.