Open access • Journal Article • DOI:10.3141/1666-12

Experimental Study of Real-Time Bus Arrival Time Prediction with GPS Data:

Wei-Hua Lin, Jian Zeng

Institutions: Virginia Tech

Published on: 01 Jan 1999 - Transportation Research Record (Transportation Research Board of the National Academies)

Topics: Automatic vehicle location, Global Positioning System and Headway

AN EXPERIMENTAL STUDY ON
REAL TIME BUS ARRIVAL TIME PREDICTION WITH GPS DATA

Wei-Hua Lin and Jian Zeng

Center for Transportation Research
and
Department of Civil and Environmental Engineering
200 Patton Hall
Virginia Polytechnic Institute & State University
Blacksburg, VA 24061
Fax: (540) 231-7532
Voice: (540) 231-5476
Email: whlin@vt.edu

Abstract

Bus headway in a rural area is usually much larger than that in an urban area. Providing real-time bus arrival information could make the public transit system more user-friendly and thus enhance its competitiveness among various transportation modes. As part of an operational test for rural traveler information systems currently ongoing in Blacksburg, Virginia, an experimental study has been conducted on forecasting the arrival time of the next bus with AVL techniques. This paper discusses the process of developing arrival time estimation algorithms, including route representation, GPS data screening for identifying data quality and delay patterns, algorithm formulation, and the development of measures of performance. Whereas GPS-based bus location data are adopted in all four algorithms presented in the paper, the extent to which other information is used in these algorithms varies. In addition to bus location data, information relevant to the performance of an algorithm also includes scheduled arrival time, delay correlation, and waiting time at time-check stops. The performance of an algorithm using different levels of information is compared against three criteria: overall precision, robustness, and stability. Our results show that at the site where the study is being conducted, the dwell time at time-check stops is most relevant to the performance of an algorithm.

Key words: AVL techniques, Traveler Information Systems, Advanced Public Transportation Systems.

INTRODUCTION

Bus headway in a rural area is usually much larger than that in an urban area. In many rural areas, the bus headway could be as long as an hour during the daytime. A variable message sign showing the bus arrival time at bus stops could reduce the anxiety of passengers waiting for the bus. Furthermore, disseminating arrival time information through other interfaces such as internet, cable TV, and audiotex could make the public transit system more user-friendly and thus increase its competitiveness among various transportation modes. In the town of Blacksburg, Virginia, an operational test has been conducted to develop a rural traveler information system with AVL techniques. The central component of such a system is predicting the arrival time of the next bus. A set of algorithms has been formulated for bus arrival time estimation specifically for a rural environment. The algorithms are currently being tested in the laboratory setting. A glimpse of the system is given in Figure 1 in which a computer visualization was created for a single bus stop on a single bus route based on the real GPS data. This paper discusses the findings and lessons learned in the process of the algorithm development.

AVL techniques have widely been perceived as useful technologies in many APTS areas, such as automatic vehicle location, vehicle component monitoring, automatic passenger counters, electronic fare boxes, timed transfers, traffic signal preemption (TSP), and Advanced Traveler Information Systems (ATIS) (1-4). Literature on the detailed technology development in all these areas, however, is quite limited. In the area of traveler's information systems, Hall *et al.* (5) evaluated the application of ITS technology for bus timed transfers. They showed that modest benefits in terms of reduction in passenger delay can be obtained with ITS technology. To the authors' knowledge, the research in arrival time estimation at bus stops based on real-time bus location data has not been reported, although in various places, tests have been ongoing to provide real-time bus arrival information.

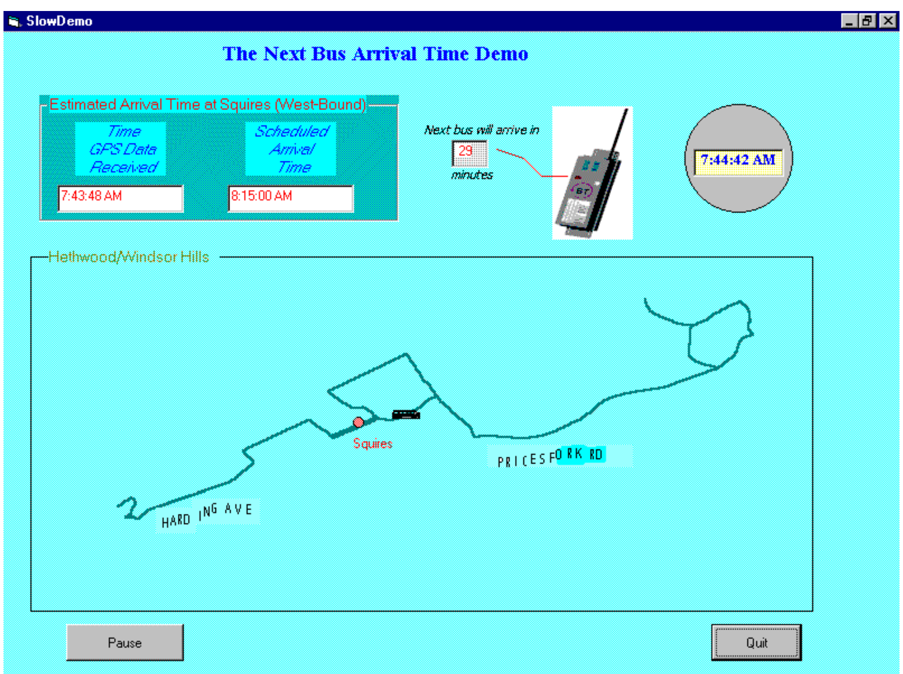


FIGURE 1 Computer visualization of the arrival time of the next bus.

Arrival time estimation is not a new concept. Systems that provide real-time arrival information of the next train have been in place for rail transit operations for a long time. For example, in the San Francisco Bay Area, every BART (Bay Area Rapid Transit) station has an overhead display panel showing the arrival time of an incoming train. Although identical in objectives, the technology used for arrival time estimation in a rail transit system is not directly transferable to a fixed route bus system due to two distinct differences between the operations of these two modes. First, relatively speaking, rail transit is a well-controlled mode; its operation is confined to tracks usually with little external interference. The location of a train at every moment, therefore, can be determined internally based on the speed profile of a train. Real time bus locations, however, cannot be tracked easily without external sensors. Bus maneuvering is often constrained by other vehicles or pedestrians on the road. Second, with the advancement in technologies, train operations today can be preprogrammed into a computer system and thus require less direct control from the train operator. Travel time between two stations can be predicted with a high degree of certainty. The operation of buses, on the other hand, remains predominantly at the control of human beings. The driver's behavior plays an important role in on-time performance of the buses. The uncertainty is unlikely to be reduced significantly with the presence of traffic signal preemption at intersections or through the use of separate HOV lanes for buses. The arrival time of the next bus could fluctuate in accordance with changes in traffic. How to deal with these two barriers is yet to be explored.

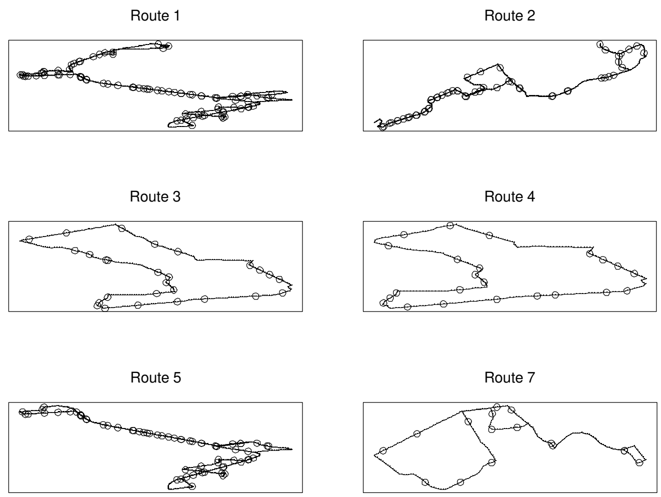
This paper starts with a discussion of the link-node representation of a bus route in which the GPS-based bus location data are mapped onto a bus path, and bus trajectories are constructed on a time-distance diagram. Following that, the results from GPS data analysis for bus schedule adherence, delay patterns, and data quality are presented. Based on the available information and the identified delay pattern, a set of algorithms for arrival time estimation is proposed and the measures for comparing these algorithms are given. Findings and lessons learned are summarized in the last section.

GPS DATA PREPARATION AND ROUTE REPRESENTATION

Currently, Blacksburg Transit (BT) operates on seven routes (see Figure 2 for selected routes). As part of the operational test, the entire vehicle fleet of BT (a total of 30 buses) has been installed with GPS units. The AVL system is Hyperdyne's AVTraxTM. The system tracks vehicles using an on-vehicle GPS-based vehicle computer which transmits vehicle position data to a central station where data are processed and made available to traveler information systems. Data are stored in the Microsoft Access database. Each record consists of four fields: the GPS ID, a time label, and the longitude and latitude of the bus location.

GPS data are point data available only at given time intervals. The knowledge of the time that a bus is at A on a route map is necessary but not sufficient to determine when the bus will reach B. In addition to the bus location data, information essential to estimating when the bus would arrive at B must also include the direction a bus moves and the actual bus travel distance, instead of the Euclidean distance, between A and B. To capture the spatial and temporal content of a moving bus, the point information from the GPS data needs to be projected onto a time-distance diagram so that the trajectory of the bus *over time* can be reconstructed. This process is done in three steps:

1) digitize the bus route into a set of nodes on a regular map spaced at about 20 meters apart;
2) convert a two-dimensional bus route into a one-dimensional bus path with link-node representation;
3) construct bus trajectories on a time-distance diagram.

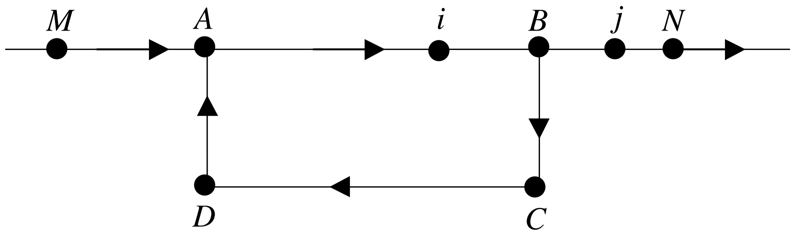


*Stop locations are indicated by circles.

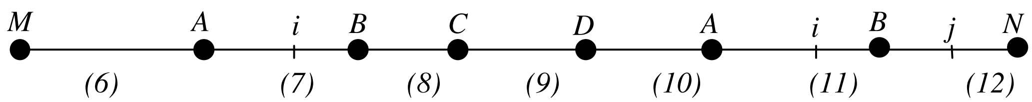
FIGURE 2 Selected bus routes in the town of Blacksburg.

A typical route is a closed path, as shown in Figure 2, in which a bus will depart and return to the same location in a single trip. A route could be circular (e.g. routes 3 and 4) or non-circular (e.g. routes 1, 2, 5 and 7). In the former case, a bus will not travel on the same segment twice for a single trip. In the latter case, a bus could travel on the same segment more than once for a single trip, but in the opposite direction. As the first step, MapixTM, a GIS software package also provided by Hyperdyne, has been utilized to convert each route into a group of nodes in longitudinal and latitudinal coordinates. Nodes are not unique to a route since two different routes may overlap a significant portion of the road. Nodes, however, are the most basic element of a route map.

The second step is to specify how nodes for a specific route should be connected to form a bus path. Note that the length of an actual bus path could be different from the length of a bus route since a bus may travel on a specific segment of the route more than once during a single trip in the same direction. For example, suppose the bus path from *M* to *N* is *M-A-B-C-D-A-B-N* as shown in Figure 3(a). The path from *i* to *j* could be either *i-B-j* or *i-B-C-D-A-B-j*. Segment *A-B* will be traversed more than once during a trip. This multiple path problem can be resolved by representing node pairs with unique directional links labeled in ascending order along the path on which a bus is traveling. If, during a single trip, a bus travels on the same segment twice, the segment will be represented by two different links. In our example, the actual bus path from *M* to *N* in link representation is given by 6-7-8-9-10-11-12, as shown in Figure 3(b). Links 7 and 11 represent the same segment *A-B*. Their downstream links, however, are different. In our study, the link length is roughly uniform, about 20 meters. This resolution is sufficient to capture the continuous movement of a bus. Figure 4 shows a segment of route 2 represented by links and nodes. The entire twelve-mile-long route is converted to a closed path with 1028 links. Given that the length of a link is very short, the knowledge of the actual location of a bus within a link becomes insignificant to the prediction accuracy. Besides, it is unlikely that two adjacent bus stops would ever reside in the same link. Therefore, our algorithm development and data analysis are based on ordered links instead of nodes. Nodes, however, are used to determine the physical location of a link on a route map so that each GPS bus location data point can be mapped on a bus path.



(a) A sample segment of a bus route.



(b) Link-node representation of the segment.

FIGURE 3 Link and node representation of a bus route segment.

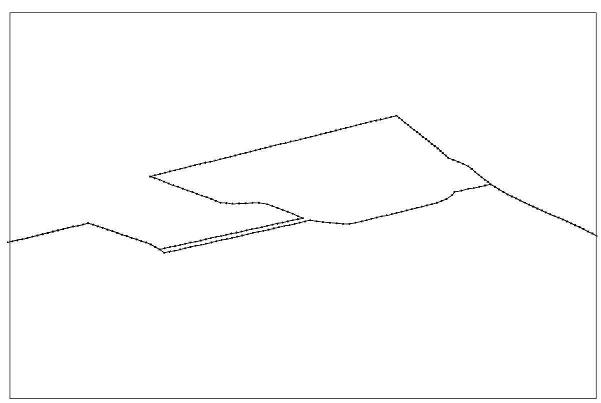


FIGURE 4 Link and node representation of a bus route segment in our study site.

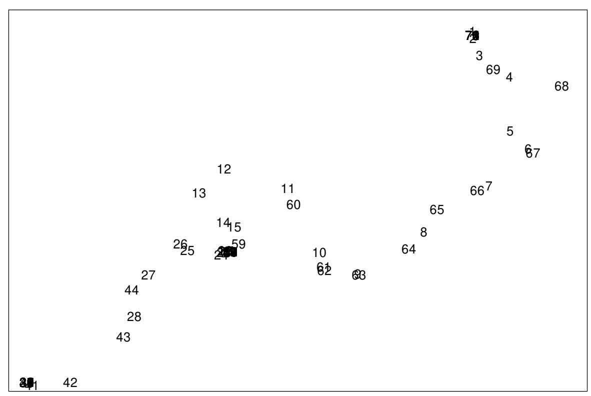
The ordered links essentially convert a two-dimensional route into a one-dimensional path. With the link and node representation of a path, bus trajectories can be easily constructed on a time-distance diagram. As an illustration, Figure 5 shows the conversion process. The raw GPS bus location data are shown in Figure 5(a). Each data point in the figure is labeled by a number, indicating the sequence in which the data are received. One can follow the data sequence to trace the bus trajectory. The same data set can now be converted into bus trajectories on a time-distance diagram as shown in Figure 5(b). This representation is very useful. Information needed to analyze the bus operation is covered in this graphical representation, including schedule adherence, bus speeds at every moment, and delay at bus stops. The wait at time-check stops, represented by the segment of the curve horizontal to the time axis, can easily be visualized. This plot provides a useful tool for screening the GPS data.

GPS DATA SCREENING

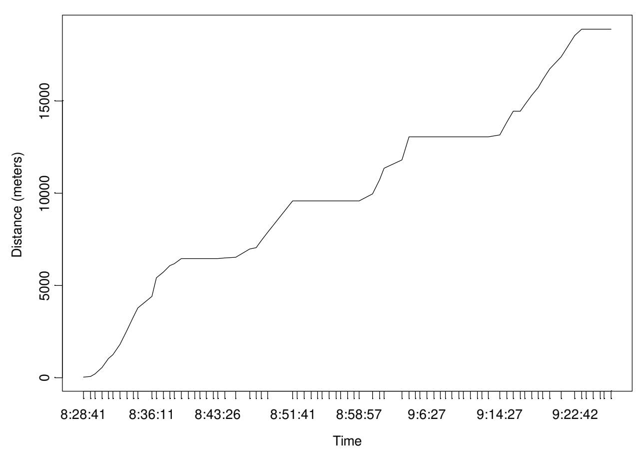
Data screening is performed to assess the quality of GPS data and how data can be effectively used to forecast the arrival time of the next bus. The screening process includes three parts: GPS data accuracy assessment, on-time performance assessment, and delay correlation assessment for multiple stops.

Accuracy range of GPS data

The accuracy level needed in predicting the arrival time of the next bus should be tied with the resources available and with the consideration of public interests. The bus display board shown in Figure 1 can only display arrival information in minutes. This limitation alone would rule out the need to pursue an accuracy level to anywhere under a minute. The accuracy of the prediction is further constrained by the quality of the input data. For bus arrival time estimation, GPS bus location data, which are the primary input to the algorithm, have two inherent constraints that could affect the accuracy of the prediction.



(a) GPS data on a two-dimensional plane.



(b) Bus trajectories on a time-distance diagram.
FIGURE 5 Conversion of raw GPS data to bus trajectories on a time-space diagram.

First, GPS bus location data do not have a fixed sample period. For a total of over 10,000 records archived in our database, the minimum sample interval is 30 seconds. The average sample interval is 46 seconds. About 15% of the data have sample intervals over one minute. The exact location of the bus is only known at the time a data point is received. If we assume that GPS data are received precisely every 30 seconds and buses are traveling at an average 25 km/hr, we won't be able to determine the exact location of a bus on a segment that could be as long as 200 meters. Most of the time, the bus location is obtained by interpolation. Occasionally, no data are received for some time. In our case, the longest time interval during which no data are received is almost 7 minutes.

Second, GPS data are not error-free. In addition to the reported accuracy range specified by the manufacturer, there is a data consistency issue when it comes to determining the bus location on a route map in longitudinal and latitudinal coordinates. Theoretically, the bus location should always fall on a link of a route if the bus is in service for that route. Suppose (x_b, y_b) is the location of a bus. To determine whether the bus is on link *i*, we use the following expression:

$$D_i = \sqrt{(x_{ui} - x_b)^2 + (y_{ui} - y_b)^2} + \sqrt{(x_{di} - x_b)^2 + (y_{di} - y_b)^2} - \sqrt{(x_{ui} - x_{di})^2 + (y_{ui} - y_{di})^2} ,$$

where (x_{ui}, y_{ui}) and (x_{di}, y_{di}) represent the location of the upstream and downstream nodes of link *i*, respectively. Note that the above expression is nonnegative. When data are perfect and the bus is indeed on link *i*, the value of D_i should be zero. Based on this logic, the link that a bus is currently on is identified by finding link *i* such that D_i is the smallest. This measure also quantifies the level of consistency in the data. For the data set we work with, 90% of the GPS bus location data give the minimum D_i value within 82.53 meters. Data with a minimum value of D_i greater than 100 meters are treated as outliers and are discarded.

These two constraints can be eliminated by the employment of more advanced technologies. Differential GPS (DGPS) could be as accurate as 3.3 feet (4), which can be used to discern vehicle location at a finer resolution. One-second data could be obtained with more communication units. Likewise, the hardware equipment, such as the bus arrival time display board, could be made to display more digits. Whether these improvements are necessary to meet the interest of the general public is another research issue that warrants further study. These issues are beyond the scope of this paper. However, the algorithm development should consider data constraints. Although the system we are developing is compatible with both GPS and DGPS, we have only used it for regular GPS units thus far.

Delay at a single stop

Bus delay at a single bus stop is the difference between the scheduled arrival time and the actual arrival time. A negative delay indicates that the bus is ahead of the schedule. There is a limitation to the accuracy of the delay since data are collected at discrete time intervals. We are more interested in the variance in delay than in the average delay. The plot in Figure 6 gives a clear indication of the variance in delay.

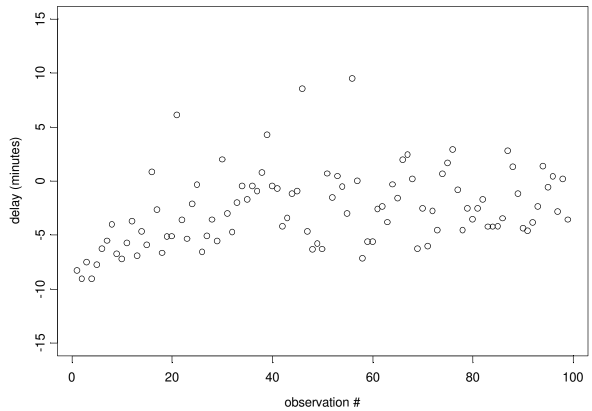


FIGURE 6 Delays at a single bus stop.

On average, buses arrive at this stop about 2.6 minutes ahead of schedule. The standard deviation is about 3.5 minutes. Similar analysis has been used for other stops. It should be noted that a large standard deviation of the actual arrival time would make prediction of the arrival time more attractive to the passengers if the error range of the prediction is sufficiently small.

Correlation of delay at multiple stops

Next, we select a pair of stops and examine the delay correlation of a bus at these two stops. Delays at the two stops are plotted for 100 bus trips in Figure 7, in which each point represents a record for a single bus. The horizontal axis represents its delay at the first stop, and the vertical axis represents its delay at the second stop. It is interesting to note that drivers tend to leave the first stop at a time later than the scheduled departure time, but are able to pick up some time on their way to the second stop so that they arrive at the second stop ahead of schedule. This may be attributed to a large built-in slack time embedded in the bus schedule.

In order to determine the correlation level, a regression analysis is performed on the data shown in Figure 7. The following are the summary statistics for fitting a linear curve:

Residual Standard Error = 99.0933, Multiple R-Square = 0.779
N = 99, F-statistic = 341.985 on 1 and 97 df, p-value = 0

	Coef	std.err	t.stat
Intercept	-378.6308	15.5327	-24.3763
X	1.1960	0.0647	18.4928

The t-statistics indicate that a linear model with intercept is appropriate to capture the delay correlation between two stops. This result is also consistent with our visual inspection.

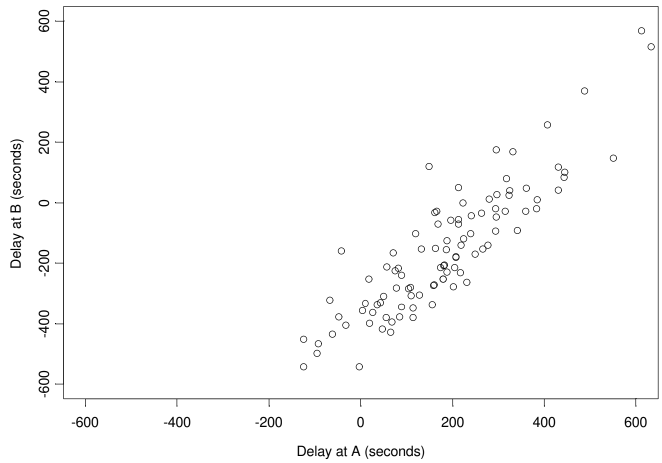


FIGURE 7 Correlation between departure and arrival times at two neighboring stops.

ALGORITHM DEVELOPMENT

Notation

$t_k(i)$: Time *k*th bus is at link (or stop) *i*.
$\tau(i, j)$: Estimated travel time between links (or stops) *i* and *j*.
$t_k(j \mid i)$: Predicted arrival time at link *j* when bus *k* is at link (or stop) *i*.
$t_d(i)$: Scheduled bus departure time based on the bus schedule at link (or stop) *i*.
$w(i)$: Average waiting time at time check link (or stop) *i*.

Four algorithms are formulated in this section. Information used as input to each algorithm is summarized in Table 1. Since the algorithm is primarily formulated for a rural environment in which traffic surveillance systems are not as accessible as those in an urban area and traffic congestion is not a major issue, traffic information is not included in our formulation. Our experience shows that this may not be a serious omission for our study site.

Baseline information: Bus schedule table only

The simplest bus arrival time algorithm is to provide the bus schedule information at bus stops. Currently, only some major stops with lots of timed transfers provide this type of information on the display board. To run such an algorithm, one only needs to insure that the clock used for updating the information is fully synchronized with the real time clock. The display is updated promptly whenever the bus schedule indicates the arrival of a bus, regardless of the actual arrival of the bus. The following discussion is trip specific. For simplification, the notation to distinguish a specific trip is omitted.

TABLE 1 Algorithms and the input information used.

Algorithm	Bus Location	Bus Schedule Table	Delay	Time Check
1	√			
2	√	√		
3	√	√	√	
4	√	√	√	√

Algorithm 1: GPS bus location data

The input to this algorithm is the real-time GPS bus location data coupled with a predefined matrix $\{\tau(i, j)\}$, representing estimated travel time between i and j. The time that the kth bus is at link i is denoted as $t_k(i)$, and the predicted arrival time at link j when the bus is currently at link i as $t(j \mid i)$. The algorithm implicitly assumes that the dwell time at bus stops is instantaneous and that delay propagates downstream in the same magnitude. The predicted arrival time of the kth bus at location j is then $t_k(j \mid i) = t_k(i) + \tau(i, j)$.

The advantage of this algorithm is that we do not have to keep track of the arrival time at stop j for a specific trip from the bus schedule table. The preparation of the travel time matrix $\{\tau(i, j)\}$ can be done off-line. To estimate the travel time matrix, one can simply read from the bus schedule table the travel time between two major stops and derive travel time between any link pair between these two stops by interpolation. For example, suppose links i and j are on a segment bounded by two major stops m and n; $t_d(m)$ and $t_d(n)$ are the scheduled arrival times given by the bus schedule table for these two stops. The travel time between links i and j is:

$$\tau(i, j) = \frac{d_{ij}}{d_{mn}}\left(t_d(n) - t_d(m)\right),$$

where d represents the distance between a link pair. For multiple buses serving the same path, the arrival time of the next bus at link j is $\min_k\{t_k(j)\}$.

Algorithm 2: GPS bus location data + bus schedule table

In this algorithm, arrival time at a bus stop based on the bus schedule for a specific trip is required as an additional input. The basic underlying assumption for this algorithm is that when a bus is far away from stop j, the correlation between the delay at the current stop and the potential delay at the destination stop is so weak that it can be omitted. The correlation will become significant only when the bus is approaching link j. The algorithm can be formulated as follows:

$$t_k(j \mid i) = \begin{cases} t_s(j) & \text{if } i < \mu < j \\ t_k(i) + \tau(i, j) & \text{otherwise} \end{cases}$$

μ is a threshold which can be determined based on the result of the data analysis. Our experience shows that choosing μ to be the index of the time check link immediately upstream of the destination link *j* yields the best result.

Algorithm 3: GPS bus location data + bus schedule table + delay

The previous two algorithms assume that a bus operates at a constant speed regardless of delay. This algorithm is based on the notion that drivers are aware of the delay incurred and will adjust their speeds accordingly to arrive at stops on time. The algorithm needs to keep track of the delay at both the current and the destination stop. We also assume that the further a bus is away from the destination link, the more likely the bus driver is able to recover from an early delay. For scheduled arrival time at the current location, $t_d(i)$, the delay at the current location is $t_k(i) - t_d(i)$. A negative delay indicates that the driver arrives early. For every link pair on the path, we calculate the expected arrival time at *j*, $t(j \mid i)$, conditioned on the bus location *i* at *t(i)*. The arrival time is estimated by

$$t_k(j \mid i) = t_k(i) + \lambda\big(t_d(i) - t_k(i)\big) + \tau(i, j) = (1-\lambda)t_k(i) + \lambda t_d(i) + \tau(i, j),$$

where $\lambda \in [0,1]$. This algorithm is identical to Algorithm 1 if λ is set to 0. A reasonable choice of λ is that λ is a decreasing function of the distance between *i* and *j,* indicating that the closer the bus is from the destination link, the more likely that the current delay will affect the projected arrival time. An appropriate function would be $\lambda = d_{ij} / d$, where *d* is the entire length of a bus trip.

Algorithm 4: GPS bus location data + bus schedule table + delay + time check point

It is observed that the dwell time of a bus at a time check link is usually much longer than that at regular stops in Blacksburg. Waiting time at a time check stop is the major contributor to schedule adherence. The assumption that the scheduled arrival time is the same as the scheduled departure time becomes weak for time check links. Consequently, travel time between a link pair that covers at least one time check stop is a function of the dwell time at time check links. Delay incurred upstream can diminish at time check stops. Without loss of generality, suppose the total number of links for a single trip is $n = 1, 2, \ldots, N$. Define $c(m)$, $m = 1, 2, \ldots, M$, as the link number for the *m*th check point stop on a route. Travel time matrix $\{\tau(i, j)\}$ defines the travel time between two links that do not have any time check stops in between, although either *i* or *j* or both could be links with time check stops. $\tau\big(c(m), c(m+1)\big)$ is the travel time between two consecutive time check stops. For a link pair that straddles at least one time check stop, travel time between the two links depends on the dwell time at time check links. For link *i,* suppose its closest downstream and upstream time check links are denoted as $c_d(i)$ and $c_u(i)$, respectively. The general form for the arrival time at link *j* predicted at link *i* for the *k*th bus can be expressed as follows:

$$t_k(j \mid i) = \begin{cases} t_k(i) + \tau(i, j); & \text{if } c_d(i) \geq j \\ \max\left\{ t_k(i) + \tau(i, c_d(i)) + \sum_{a=\{m \mid c(m)=c_d(i)\}}^{\{n-1 \mid c(n)=c_u(j)\}} \tau\big(c(a), c(a+1)\big),\ \ t_d(c_u(j)) \right\} + \tau(c_u(j), j); & \text{if } c_d(i) < j \end{cases}$$

where *m* is the closest time check stop downstream of link *i* ($c(m) = c_d(i)$) and *n* is the closest time check stop upstream of link *j* ($c(n) = c_u(j)$). Here, the travel time between two time check links *m* and *n* can be calculated by:

$$\tau(m,n) = t_d(n) - t_d(m) - w(n),$$

where $w(n)$ can be estimated by the average waiting time at time check link *n*. Travel time between links bounded by time check links *m* and *n* could be obtained by interpolation as discussed in Algorithm 1.

MEASURES OF PERFORMANCE

In this section, three measures have been developed to determine the overall precision, the robustness, and the stability of an algorithm. The measures developed below all make use of the difference between the actual arrival time and the predicted arrival time at link *j* for bus *k*, i.e., $\varepsilon_k = t_k(j \mid i) - t_k(j)$. We use $\varepsilon_{k,m}$ to denote the prediction error for the *m*th sample.

A good algorithm should be such that its overall deviation between the predicted and actual arrival times is low. The standard least square method is used for this testing purpose, i.e. $M_1 = \sqrt{\sum\sum \varepsilon_{k,m}^2}$. A low M_1 value is desirable for an algorithm. However, an algorithm with a low value of M_1 may occasionally yield a prediction with a large deviation. This is undesirable since it may divert passengers away from the bus stop and eventually cause them to miss the bus. The second measure is designed to detect this behavior. It examines the robustness of an algorithm such that its maximum deviation is within a certain range. The mathematical expression for this is $M_2 = \max_{k,m}\{|\varepsilon_{k,m}|\}$. Finally, we need to measure the stability of the algorithm. It is not appropriate for the prediction of arrival time to fluctuate drastically within a short period of time. The mathematical expression is given by $M_3 = \sum\sum \left|\varepsilon_{k,m} - \varepsilon_{k,m-1}\right|$. Note that this measure should only be considered in conjunction with other measures. Otherwise, the bus schedule would be the best algorithm, since its value of M_3 is zero.

These three measures have been applied to the four algorithms described in the previous section and the baseline case in which the only information available is the bus schedule table. The results are given in Table 2. It appears that the performance of Algorithm 4 is better than all other algorithms.

CONCLUDING REMARKS

For public transit systems, arrival time estimation is an important component in a traveler information system, especially in a rural environment where the bus headway is large. Although bus schedules are often available through various sources, providing real-time bus arrival information at the bus stop or other interfaces would make the public transit system more user-friendly, and therefore more attractive to riders.

Four GPS data-based arrival time estimation algorithms were discussed in this paper. In addition to GPS-based bus location data, other information has been used as input data, including bus schedule information, bus delay patterns, and bus stop type information (a time-check stop vs. a regular stop). Whereas all algorithms employ GPS-based bus location data, the level of other information used in each algorithm varies. It was shown in the paper that data screening is necessary to determine the type of information that should be included in the algorithm. In our study site, we found that treating time-check stops and regular stops differently would make a big difference in the performance of the algorithm. For regular bus stops, since the arrival and departure time of a bus happens within a small time window relative to the precision required (over a minute), we simply assume that the arrival time is equal to the departure time. The same assumption is not valid for time-check stops, where a bus may stop for over five minutes.

TABLE 2 A comparison of four arrival time estimation algorithms.

Algorithm	Overall Deviation (M_1)	Max. Deviation (M_2)	Stability (M_3)
Base case	3.6	4.7	0.0
Algorithm 1	2.8	7.7	1.0
Algorithm 2	4.9	12.0	0.7
Algorithm 3	2.9	7.0	0.9
Algorithm 4	2.0	4.5	0.2

The performance of these algorithms was compared under three criteria: overall precision, robustness, and stability. Mathematical expressions for these criteria were developed in the paper. The overall precision measure determines the average deviation of the predicted arrival from the actual arrival time. The robustness measure determines if an algorithm will occasionally give a prediction that is far off the actual arrival time. The stability measure checks if the prediction given by an algorithm fluctuates from time to time. Our results show that the algorithm considering time-check information has the best performance.

Currently, GPS data are available to us approximately every 45 seconds. We will further examine if the performance of an algorithm will degrade significantly when the sample period increases. Significant cost reduction could be realized by using data with longer sample periods. Future research can also include a study on the optimal link length. The link and node representation is currently designed for both GPS and DGPS data. The average length of a link is currently 20 meters. Links with longer length could reduce the total number of links needed to represent a route and thus reduce the overall computational time.

ACKNOWLEDGMENTS

Research supported by the project “Rural Traveler Information System Operational Test” sponsored by Federal Transit Administration. We wish to thank Mike Connelly and Kevan Danker of Blacksburg Transit for their support during the project.

REFERENCES

1. Casey, R. F., L. N. Labell, J. A. LoVecchio, R. S. Ow, J. W. Royal, J. C. Schwenk, L. Moniz, E. S. Carpenter, C. L. Schweigher, and B. Marks (1998) *Advanced Public Transportation Systems: The State of the Art. Update ’98.* DOT-VNTSC-FTA-97-9, U.S. Department of Transportation.

2. Gomez, A., F. Zhao, and L. D. Shen (1998) *Benefits of Transit AVL and Transit AVL Implementation in the U.S.*, Preprint, Annual Transportation Board Meeting, 1998, Transportation Research Board, Washington, D.C.

3. Hardin, J. A., R. G. Mathias, and M. C. Pietrzyk (1996) *Automatic Vehicle Location and Paratransit Productivity*. National Urban Transit Institute, Center for Urban Transportation Research.

4. Hickman, M. and T. Day (1996) *An Information and Institutional Inventory of California Transit Agencies,* California PATH Research Report, UCB-ITS-PRR-96-12.

5. Hall, R., M. Dessouky, A. Nowroozi, and A. Singh (1997) *Evaluation of ITS Technology for Bus Timed Transfers*, California PATH Research Report, UCB-ITS-PRR-97-37.