

Modeling Schedule Recovery Processes in Transit

Operations for Bus Arrival Time Prediction

Wei-Hua Lin
Department of Systems and Industrial Engineering
The University of Arizona,
Tucson, AZ 85721-0020, U.S.A.
Phone: 520-621-6553
Fax: 520-621-6555
Email: weilin@sie.arizona.edu

Robert L. Bertini
Department of Civil & Environmental Engineering
Portland State University
Portland, OR 97207-0751, U.S.A.
Phone: 503-725-4249
Fax: 503-725-5950
Email: bertini@pdx.edu

August 2004

ABSTRACT

Many existing algorithms for bus arrival time prediction assume that buses travel at free-flow speed in the absence of congestion. As a result, delay incurred at one stop would propagate to downstream stops at the same magnitude. In reality, skilled bus operators often constantly adjust their speeds to keep their bus on schedule. This paper formulates a Markov chain model for bus arrival time prediction that explicitly captures the behavior of bus operators in actively pursuing schedule recovery. The model exhibits some desirable properties in capturing the schedule recovery process. It guarantees provision of the schedule information if the probability of recovering from the current schedule deviation is sufficiently high. The proposed model can be embedded into a transit arrival time estimation model for transit information systems that use both real-time and schedule information. It also has the potential to be used as a decision support tool to determine when dynamic or static information should be used.

INTRODUCTION

Automatic vehicle location (AVL) systems based on global positioning systems (GPS) have been widely adopted by many transit systems to monitor the movements of buses on a real time basis. The AVL systems are not only useful for fleet management but also important for providing transit signal priority treatment, developing demand responsive systems and dynamic transit information systems, and scheduling the coordinated timed transfer among buses or between transit and other transportation modes [Casey, et al., 2000]. One core component in many of these systems is the prediction of bus travel time between two given points. The focus of this paper is on the prediction of the arrival time of the next bus.

For the past few years, many efforts have been devoted toward developing algorithms and procedures for bus arrival time prediction (for a review, see Mishalani, et al., 2000). Existing algorithms often assume that buses travel at free-flow speed in the absence of congestion. As a result, when predicting bus arrival times at subsequent bus stops, the delay incurred at one stop will be carried forward in the downstream direction. In reality, skilled bus operators constantly adjust their speeds in order to keep their buses on schedule. In fact, some transit systems provide real-time schedule adherence status to their operators on an in-vehicle display terminal [Strathman, et al., 2001]. This paper formulates a Markov chain model for bus arrival time prediction that explicitly captures the behavior of bus operators in actively pursuing schedule recovery when delay occurs.

Prediction accuracy for an advanced traveler information system is an issue that should be considered in the context of specific requirements from the perspectives of system users and system suppliers. On the supply side, i.e. for transit agencies, the degree of accuracy is often pursued on the order of minutes. Note that many transit agencies define delay as “behind schedule for more than five minutes.” Considering accuracy on the order of seconds may be perceived as overkill even though the information is occasionally needed for display purposes when the bus is just a few seconds away from the bus stop. In this situation it is odd to see a message board showing that “the bus is one minute away” or “the bus is zero minutes away” while one can actually see the incoming bus. This problem, however, can be easily resolved by switching information from quantitative to qualitative, such as displaying the message “the bus is due” or “the bus is approaching.” Besides, prediction accuracy is also tied to the accuracy of the data sources and the reliability of the communication and geolocation systems. In many places the polling of AVL data occurs at least once every thirty seconds, making it unnecessary to

pursue accuracy at a degree much higher than half a minute. On the demand side, transit riders are primarily interested in dynamic transit information if bus headways are relatively large. For small headways such as ten to fifteen minutes, the average wait is only five to seven minutes, and thus, the benefits of real time transit information are minimal. We believe that an accuracy level on the order of one minute would suffice for most applications.

A typical bus arrival time prediction algorithm uses both “static information” and “dynamic information.” “Static information” refers to bus schedule information, historical information about recurrent traffic conditions, and average dwell time at bus stops. “Dynamic information” includes: real-time bus location data, delay at bus stops, weather and current traffic conditions. For prediction purposes, it is also important to incorporate the information of how delay propagates into a prediction algorithm. Under normal traffic conditions, it seems reasonable that delays at two nearby stops are strongly correlated. Delays for two stops far from one another are usually weakly correlated. This is revealed by field data obtained from Blacksburg, VA, as shown in Figure 1. The degree of correlation is, of course, related to the amount of built-in slack time embedded in the bus schedule and the distance between the two stops. The plots shown in Figure 1 also reveal the degree of schedule recovery, represented by the magnitude of the slopes of the dashed lines fitted to the scattered data points. The smaller slope shown in the second plot is a sign of a larger schedule recovery. Even with the bunching effect and/or congestion, one would anticipate that responsible operators should adjust their speeds in accordance with their on-time status. As shown in the plot, schedule recovery is clearly a function of the distance between stops. The question arising here is how to quantify the recovery process so that this information can be incorporated into our prediction algorithm.

The paper is organized as follows. In the next section, we describe briefly the process of converting a two-dimensional bus route map into a one-dimensional bus path with link-node representations, since the model proposed in this paper is based on a linear bus path. The following section discusses the desired properties for a robust arrival time prediction algorithm, which in turn, determines the required properties of our formulation. The next section provides the assumptions and model formulation. This is followed by a numerical example and a summary of results.

CONVERSION OF TWO-DIMENSIONAL BUS ROUTE MAP TO A ONE-DIMENSIONAL BUS ROUTE PATH

Bus routes can exhibit several different forms. A typical bus route is a closed path on which a bus departs and returns to the same location in a single trip. A route can be circular or non-circular. 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.

The geographic information system (GIS) or GPS tools used in a transit system for monitoring the movement of buses can also be used to create a digital route map. The bus route map shown on a conventional map, however, cannot be used directly for the purposes of constructing bus trajectories. It was shown by Lin and Zeng [1999] that the actual location of a bus with respect to the path could sometimes be ambiguous when the bus path is represented on a regular two-dimensional map. Complications can arise in a number of scenarios. One possible complication is that a bus may travel on the same segment of a street in the same direction more than once during a trip as shown in Figure 2. As a result, the distance from a point on the segment (say,

point *a*) to another point downstream (say, point *b*) on a two-dimensional map as shown in Figure 2 may have two different lengths, depending on whether it is the first time or the second time the bus is traveling on that repeated segment. To avoid ambiguities like this, it is advantageous to convert the two-dimensional map into a linear map with link-node representations. With a linear map, the road segment that has been traversed by a bus more than once is represented by two different links. Moreover, all links are directional and ordered. This representation then obviates the need for nodes, making data storage even more efficient.

The GPS-based point information collected on a real-time basis can be projected onto a unique link based on a minimum distance rule [Lin and Zeng, 1999]. The trajectory of a bus can then be reconstructed easily on a time-space diagram as shown in Figure 3. With this knowledge of the spatial and temporal description of a moving bus coupled with schedule data, one is able to evaluate all attributes pertaining to its performance, such as dwell time at bus stops, delay at intersections, average speed, schedule adherence, and so on.

The construction of such a linear bus path with link representations can be accomplished without resorting to GIS tools. Lin and Padmanabhan [2002] developed a simple procedure for constructing a one-dimensional map from available GPS bus trajectory data. Tests showed that the resulting digitized map was sufficiently accurate for the application of bus arrival time estimation. The bus path information can be conveniently processed either as a sequence of ordered links with equal length or as a sequence of ordered links with equal travel time. Moreover, the digitized map produced is easy to modify for special operations, such as temporary bus detours for special events or work zones.

How quickly an operator can bring the bus back on schedule depends on the magnitude of the deviation from the schedule and the length of the remaining trip. It is intuitive that full schedule

recovery can be realized only if the remaining bus trip is sufficiently long. Under the assumption that links have equal travel times, the degree of schedule recovery between any two neighboring links is identical as well. This assumption is the basis for the development of our model that will be discussed in detail later. We associate the real time delay information with discrete states representing the magnitude of the delay. With equal travel time links, it follows that the probabilities for the delay status of a bus to change from one state into another (the change in delay) as it moves from one link to another are also identical.

DESIRED PROPERTIES FOR A PREDICTION ALGORITHM

Let $E_{k,i}^{(t)}$ be the predicted delay for i th bus trip at bus stop k obtained at time t , $k \in K$ and $i \in I$. K and I are sets for bus stops and bus trips, respectively. Let the actual delay for the i th bus trip at bus stop k be $D_{k,i}$. Also, let $A_{k,i}$, $S_{k,i}$ be the actual arrival time and scheduled arrival time for the i th bus trip at bus stop k , respectively. $D_{k,i} = A_{k,i} - S_{k,i}$. The value of $D_{k,i}$ is positive (negative) for situations when a bus is running behind (ahead of) schedule. $D_{k,i}$ is available only at $t \geq A_{k,i}$. Let $\varepsilon_{k,i}^{(t)}$ be the absolute prediction error for prediction made at time point $t < A_{k,i}$, i.e. $\varepsilon_{k,i}^{(t)} = |E_{k,i}^{(t)} - D_{k,i}|$.

The effectiveness of a prediction algorithm can be considered by the following three measures: overall performance, robustness, and stability.

Overall Performance

A desirable arrival time prediction algorithm should minimize its overall deviation from the actual arrival times. The measure of effectiveness (MOE) for overall performance is the total prediction error normalized by the total number of predictions made, N ,

i.e., $MOE_1 = \frac{1}{N} \sum_{k \in K} \sum_{i \in I} \sum_{t < A_{k,i}} \varepsilon_{k,i}^{(t)}$. One can also use the standard least squares method, i.e.,

$MOE_1 = \frac{1}{N} \sum_{k \in K} \sum_{i \in I} \sum_{t < A_{k,i}} \varepsilon_{k,i}^{(t)2}$. An algorithm with a small value of MOE_1 is desirable.

Robustness

While keeping the overall prediction errors as small as possible, a robust algorithm should have a small maximum prediction error. An algorithm with a small value of MOE_1 may occasionally yield large prediction errors. This is undesirable since it may divert some transit users away from the bus stop and consequently cause them to miss the bus. Even if this happens to a small number of users, it would undermine the confidence of bus riders in the credibility of the overall transit information system. Therefore, the second measure is to minimize the occurrence of large deviations. A robust algorithm should control its maximum prediction error within a tolerable range. The mathematical expression for this measure is $MOE_2 = \max_{k,i,t} \{|\varepsilon_{k,i}^{(t)}|\}$.

Stability

Finally, a desirable algorithm should also be stable, such that the prediction of arrival time does not fluctuate wildly from time to time. The mathematical expression for this measure is

$MOE_3 = \sum_{k \in K} \sum_{i \in I} \sum_{t < A_{k,i}} |E_{k,i}^{(t)} - E_{k,i}^{(t-1)}|$. Note that this measure can only be considered in conjunction with

other measures. Otherwise, the bus schedule would become the best predictor in terms of stability. Since the bus schedule information is static, MOE_3 is always zero, regardless of how far the actual arrival time of a bus deviates from the predicted or scheduled arrival time.

Figure 4 is a plot that demonstrates how bus arrival time information is updated at a specific bus stop. The plot shows a bus trajectory on a time-space diagram and arrival time predictions made for a specific stop with its scheduled bus arrival time at 9:45. The solid vertical line corresponds

to the scheduled arrival time and the dashed vertical line corresponds to the actual arrival time. The small circles are the arrival time predictions made at various time points and locations. One can easily identify the exact time and location where a specific prediction was made by drawing a horizontal line that joins the circle and the bus trajectory curve. The coordinate of the intercepting point on the bus trajectory curve yields the time and location at which the prediction was made. As shown on the plot, the predictions largely center on the scheduled arrival time as the bus moves farther away. The bus was actually about five minutes ahead of schedule.

Ideally, a prediction algorithm would yield predictions that center on the dashed curve instead and have small fluctuations horizontally (the properties of overall performance and robustness). The prediction points that are adjacent vertically should not be far apart (the property of stability).

MODEL ASSUMPTIONS AND FORMULATION

The model formulation discussed below is based on a “good operator” assumption, which is, if a bus in operation deviates from its schedule, the bus operator will adjust the bus speed, by either speeding up or slowing down, to ensure schedule adherence. The recovery process, however, is bounded by other constraints, such as distances and the bus speed. For simplicity in notation, the discussion given below is in the context of assuming uniformly spaced bus stops. A discussion of removing this assumption is given towards the end of this section. The discussion below is based on nominal traffic conditions. The model, with some adjustment, also applies to situations with incident- or demand-related congestion.

Let bus stops be ordered with indices $k = 0, 1, 2, 3, \dots$ in the downstream direction. Suppose that the bus is currently at stop k_0 . The delay information at bus stop k_0 is thus available, which

can be used to update the prediction of delays for all downstream stops. It is desirable that the relationship between the actual delay at stop k_0 , $D_{k_0,i}$, and the predicted delay $E_{k,i}^{(t)}$ for all downstream stops has the following properties:

(1) If $D_{k_0,i} = 0$, then $E_{k,i}^{(t)} = 0$, $\forall k > k_0$.

(2) If $D_{k_0,i} \neq 0$, then for $\forall k > k_0$, the predicted delay obtained at time t should be:

a) $|E_{k,i}^{(t)}| > |E_{k+1,i}^{(t)}|$;

b) $|E_{k,i}^{(t)}| \approx |E_{k+1,i}^{(t)}|$; and

c) $\lim_{k \rightarrow \infty} E_{k,i}^{(t)} = 0$.

Property (1) states that if the bus is currently on schedule, then the best prediction for the arrival time of the bus at subsequent bus stops should simply be the scheduled arrival time. Property (2) gives a list of desired properties for the prediction of the arrival time at downstream stops ($k > k_0$) if the bus is not on time at stop k_0 . Property (2a) states that the degree of schedule recovery increases with distance. Clearly, bus operators need time and distance to bring their buses back on schedule. The longer the distance, the more likely it is that the bus will return to its schedule. Property (2b) implies that there is a strong similarity between delays occurring at two neighboring stops. This property will make the recovery curve smooth throughout its possible range. Property (2c) states that a full recovery of the bus schedule will be accomplished eventually given that the remaining trip length is sufficiently long. The predicted delay decays in distance and approaches zero, suggesting a final return to the original schedule. These two properties are consistent with the three performance measures discussed in the previous section.

The above characteristics in the process of schedule recovery can be naturally captured by a one-step discrete Markov chain model. Suppose we consider discrete states for delay and introduce a

vector \mathbf{d} , with $\mathbf{d} = [d_1 \ d_2 \ d_3 \ \dots \ d_M]$, representing a broad range of delay in some predefined time units (one minute, two minutes, etc.). Vector \mathbf{d} can be symmetric, such as $[-10 \ -9 \ \dots \ 0 \ \dots \ 9 \ 10]$ or asymmetric such as $[-10 \ -9 \ \dots \ 0 \ \dots \ 7 \ 8]$. The choice of M for a specific route depends on the past delay record for the route. The $M \times M$ one step transition matrix, P , for any two neighboring stops is of the following form:

$$P = \{p_{ij}\} = \begin{pmatrix} p_{11} & p_{12} & \cdots & \cdots & p_{1M} \\ \vdots & & & & \\ & & & & \\ & & & & \\ p_{M1} & p_{M2} & \cdots & \cdots & p_{MM} \end{pmatrix}$$

Each entry in the transition matrix $\{p_{ij}\}$ represents the conditional probability that given a bus is delayed d_i units of time at stop k the bus is delayed d_j units of time at an immediate downstream stop $k+1$. Let $p_j(k)$ be the probability that delay at bus stop k is d_j . The state probability for bus stop $k+1$ can then be written as a linear equation system:

$$\begin{aligned} p_1(k+1) &= p_1(k)p_{11} + p_2(k)p_{21} + \dots + p_M(k)p_{M1} \\ p_2(k+1) &= p_1(k)p_{12} + p_2(k)p_{22} + \dots + p_M(k)p_{M2} \\ \dots & \quad \dots \quad \dots \quad \dots \quad \dots \\ p_M(k+1) &= p_1(k)p_{1M} + p_2(k)p_{2M} + \dots + p_M(k)p_{MM} \end{aligned}$$

The above linear equation system can be conveniently written in the matrix form $\mathbf{p}(k+1) = \mathbf{p}(k)P$, where vector $\mathbf{p}(k) = [p_1(k) \ p_2(k) \ \dots \ p_M(k)]$. For a bus currently at stop k , if the bus is delayed d_j time units, then the j th element in vector $\mathbf{p}(k)$ is 1 and all other elements are zero. By substituting repeatedly, knowledge of the delay at the present bus stop (say stop 0) is sufficient to describe fully the future probabilistic behavior of delay at all downstream stops ($k > 0$). With the one-step transition matrix, the probability vector for $k > 0$ can be computed with the Markov chain model: $\mathbf{p}(1) = \mathbf{p}(0)P$, $\mathbf{p}(2) = \mathbf{p}(0)P^2$, ..., and $\mathbf{p}(k) = \mathbf{p}(0)P^k$ for stops 1, 2, ..., k . Given the

delay status at stop 0 at time t , the predicted delay at stop 1 obtained at time t is $E_{1,i}^{(t)} = \mathbf{p}(1) \mathbf{d}^T = \mathbf{p}(0) P \mathbf{d}^T$. The predicted delay at any downstream stop k is $E_{k,i}^{(t)} = \mathbf{p}(0) P^k \mathbf{d}^T$.

The transition matrix can be estimated with the following mathematical programming problem:

$$\min \sum_i \sum_k \sum_{t < A_{k,i}} (D_{k,i} - E_{k,i}^{(t)})^2$$

s.t.

$$E_{k,i}^{(t)} = \mathbf{p}(0) P^k \mathbf{d}^T$$

$$0 \leq p_{ij} \leq 1.$$

The decision variables p_{ij} can then be determined by solving the above mathematical programming problem. Note that the above formulation is based on the minimization of the overall prediction errors, which is consistent with MOE_i discussed earlier.

Figure 5 shows a class of curves generated from the proposed model. Qualitatively, one can see that they exhibit the two properties discussed at the beginning of this section. The decreasing curve shown in Figure 5 indicates that delay will return to zero eventually. This result departs from many existing bus arrival time estimation algorithms in that it does not assume that delay propagates in the downstream direction with the same magnitude in the absence of built-in slack time. It also shows that the proposed model will eventually generate the schedule information, independent of the initial delay status. The curves shown in Figure 5 are symmetric for the cases of ahead of schedule and behind schedule. Transit agencies usually strongly discourage operators from being early. This can be easily captured with the proposed model by modifying the assigned probabilities in the transition matrix. The result is shown in Figure 6, which shows a much shorter recovery process for being early than being late.

For simplicity in notation and in exposition, the discussion given above was based on the assumption that bus stops are uniformly spaced. This restriction is not realistic but can be relaxed very easily with few changes to the model described above. Suppose that the bus stops spaced at equal distances along a bus path are virtual bus stops. In actuality, only a subset of these virtual bus stops represents real bus stops. With this representation, we only need to make sure that the locations of a subset of these virtual bus stops coincide with the locations of actual bus stops. Consequently, the predicted delays are preserved only for those virtual bus stops that correspond to the real bus stops. The use of the virtual bus stop concept would require the calibration process of the transition matrix to be modified accordingly. The transition matrix was developed based on the relationship between the delay statuses of two neighboring bus stops. Unfortunately, delay information for virtual stops is not readily available if a virtual stop does not coincide with a real bus stop. In this case, interpolation should be considered in estimating the probabilities in the transition matrix since schedule recovery is a function of distance.

NUMERICAL EXAMPLES

Consider a case in which the mapping of virtual bus stops to actual bus stops is given in Table 1. Suppose we consider only five possible delay states. Let $\mathbf{d} = [-4 \ -2 \ 0 \ 2 \ 4]$. The transition matrix is given by:

$$P = \begin{pmatrix} 0.93 & 0.05 & 0.02 & 0.00 & 0.00 \\ 0.00 & 0.95 & 0.04 & 0.01 & 0.00 \\ 0.00 & 0.01 & 0.98 & 0.01 & 0.00 \\ 0.00 & 0.01 & 0.04 & 0.95 & 0.00 \\ 0.00 & 0.00 & 0.02 & 0.05 & 0.93 \end{pmatrix}$$

Let the probability vector for delay at bus stop 0 be $\mathbf{p}(0) = [0 \ 1 \ 0 \ 0 \ 0]$, indicating a delay of -2 minutes at the current stop. In other words, the bus is 2 minutes ahead of schedule. The

probability vectors for bus stops 1 (corresponding to virtual bus stop 10) and 2 (corresponding to virtual bus stop 24) can be calculated by $\mathbf{p}(10) = \mathbf{p}(0)P^{10}$ and $\mathbf{p}(24) = \mathbf{p}(0)P^{24}$, respectively.

Since $E_{k,i}^{(t)} = \sum_{j=1}^5 p_j(k)d_j$, the predicted delay at bus stops 1 and 2 are -1.08 minutes and -0.43 minutes, respectively.

The following transition matrix depicts a faster recovery process:

$$P = \begin{pmatrix} 0.91 & 0.07 & 0.02 & 0.00 & 0.00 \\ 0.00 & 0.91 & 0.08 & 0.01 & 0.00 \\ 0.00 & 0.01 & 0.98 & 0.01 & 0.00 \\ 0.00 & 0.01 & 0.07 & 0.92 & 0.00 \\ 0.00 & 0.00 & 0.02 & 0.05 & 0.93 \end{pmatrix}$$

The predicted delays at bus stops 1 and 2 are -0.34 minutes and -0.06 minutes, respectively, a much faster recovery process than that in the previous case. In the above two examples, we assume that the probability transition matrix is homogenous. This does not have to be the case. In fact, computation of schedule recovery is done segment by segment. It is separable but additive. In case there is an incident downstream, the schedule recovery computation can be reset at the point of the incident with a new probability transition matrix. The effect of the incident will thus be reflected in the bus arrival time estimation for all the stops downstream of the incident location.

CONCLUSION AND FUTURE RESEARCH

This paper proposes a Markov chain model that explicitly captures the behavior of bus operators in pursuing schedule recovery when a bus is delayed. The model discussed in this paper is not a complete description of a bus arrival time estimation algorithm. Rather, the proposed model can serve as a component in an arrival time prediction algorithm. The component described here can

be used to replace the assumption of free-flow travel time (or zero delay) in the absence of congestion used in many algorithms. In the proposed model, delay incurred at one stop will propagate in the downstream direction (similar to the result obtained with the free flow travel time assumption) but will decay over time and space. This is consistent with the fact that in reality good operators often constantly adjust their speeds to keep their buses on schedule. Some desirable properties of the proposed model were discussed. Numerical results indicate that the proposed model is able to return reasonable results.

The assumptions made in the paper were based on our observations from limited data sets. Studies are ongoing to test the effectiveness of the proposed model with field data from several transit agencies. We are also exploring efficient ways of constructing the probability transition matrix. Ideally, the matrix should be calibrated with data from actual transit operations. This is, however, a tedious process but can be automated. It would also be interesting for a future study to examine if the probability transition matrix produced without detailed calibration is robust enough for a transit information system.

REFERENCES

- Casey, R.F., L. N. Labell, L. Moniz, J. W. Royal, M. Sheehan, T. Sheehan, A. Brown, M. Foy, M. Zirker, C. L. Schweiger, B. Marks, B. Kaplan, and D. Parker (2000). Advanced Public Transportation Systems: The State of the Art Update 2000, FTA-MA-26-7007-00-1.
- Lin, W.H. and V. Padmanabhan (2002) A simple procedure for creating digitized bus route information for Intelligent Transportation System applications. IN: *Transportation Research Record*, National Research Council. (*In press*)

Lin, W.H. and J. Zeng (1999). An experimental study on real time next bus arrival time prediction with GPS data, IN: *Transportation Research Record* no. 1666, p101-109, National Research Council.

Mishalani, R. G., Lee, S., and M. R. McCord (2000) Evaluating real-time bus arrival information systems, IN: *Transportation Research Record*. no. 1731, National Research Council.

Strathman, J.G., Kimpel, T.J., Dueker, K.J., Gerhart, R.L., Turner, K., Griffin, D., and Callas, S. (2001). "Bus transit operation control: review and an experiment involving Tri-Met's automated bus dispatching system." *Journal of Public Trans.*, 4 (1), 1-26.

Table 1. Mapping of the actual bus stop index to its corresponding virtual stop index

Virtual Stop Index	Actual Stop Index
0	0
1	-
2	-
3	-
4	-
5	-
6	-
7	-
8	-
9	-
10	1
11	-
12	-
13	-
14	-
15	-
16	-
17	-
18	-
19	-
20	-
21	-
22	-
23	-
24	2
25	-
.....	

Figure 1. Plots of delays at two stops (a) when two stops are close to each other (stops A and B); (b) when two stops are far apart (stops A and C).

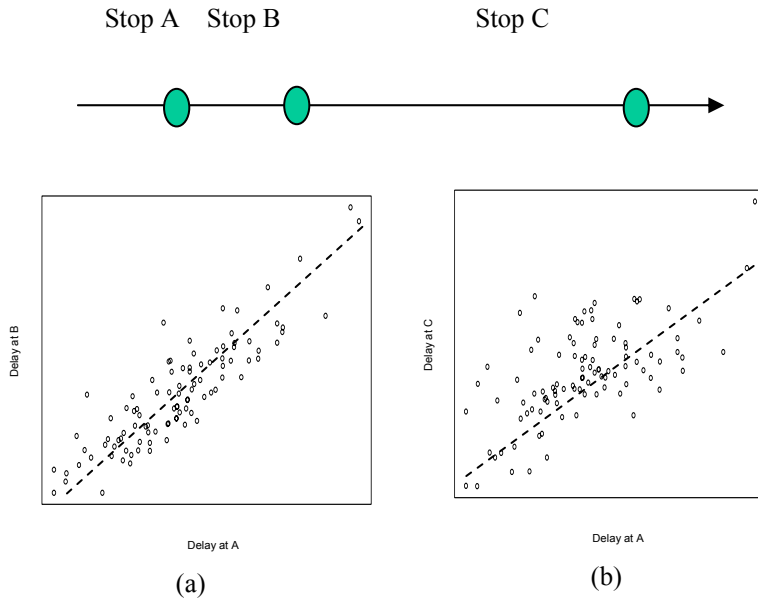


Figure 2. The segment that is traversed more than once during a bus trip.

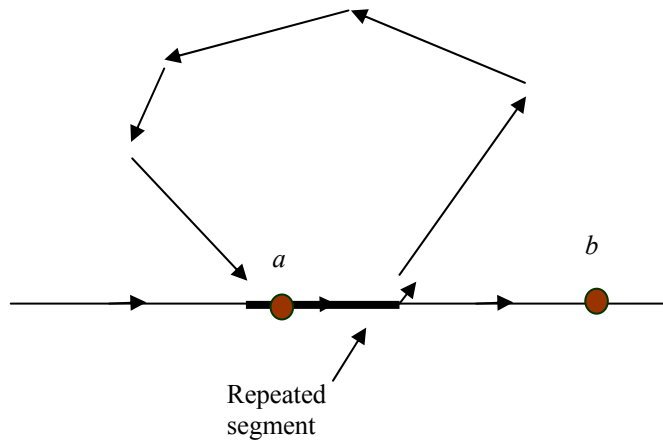


Figure 3. A time space diagram for bus trajectories (ticks on the y-axis represent the location of bus stops).

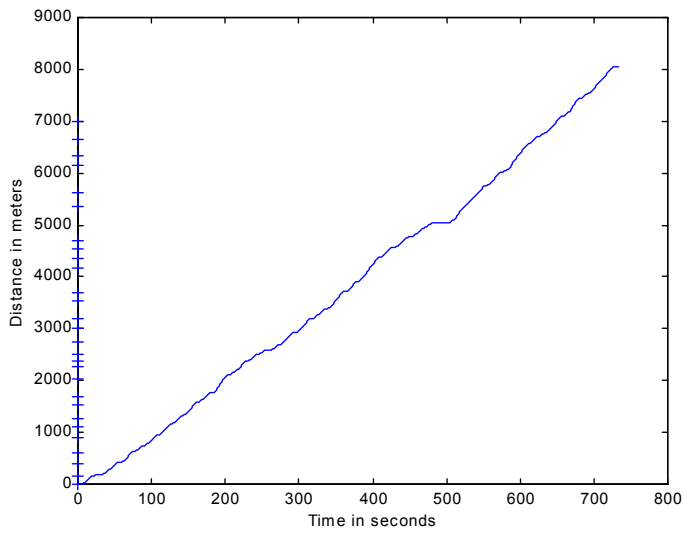


Figure 4. The information updating process in a bus arrival time information system (circles are the predicted arrival times viewed at the bus stop at different times).

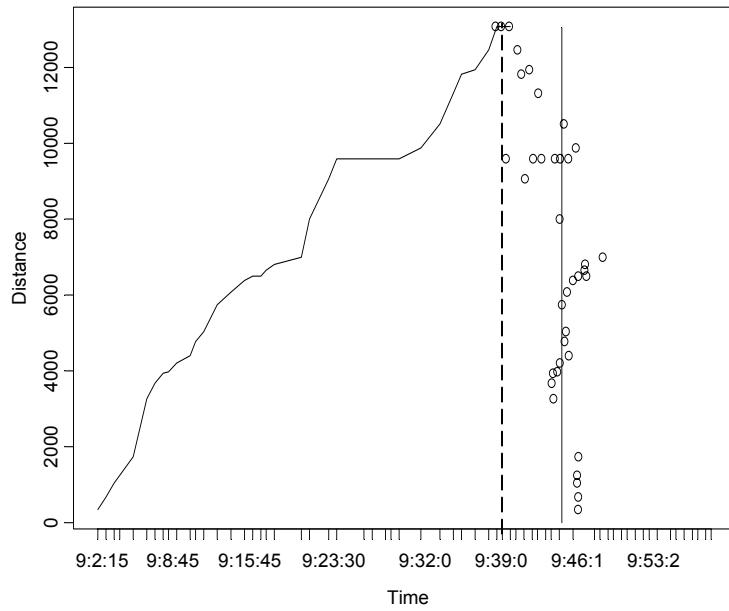


Figure 5. The decay characteristics of the predicted delay (symmetry in the schedule recovery process for ahead of schedule and behind schedule).

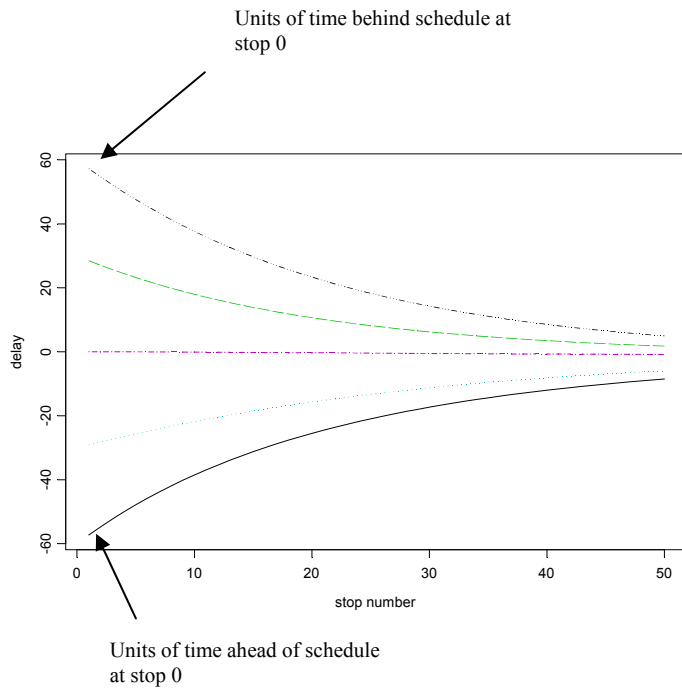


Figure 6. The decay characteristics of the predicted delay (asymmetry in the schedule recovery process for ahead of schedule and behind schedule).

