Paper No. 01-3036

Transit Vehicle Arrival Prediction

Algorithm and Large-Scale Implementation

D. J. Dailey, S. D. Maclean, F. W. Cathey, and Z. R. Wall

An algorithm is presented to predict transit vehicle arrival times up to 1 h in advance. It uses the time series of data from an automated vehicle location system, consisting of time and location pairs. These data are used with historical statistics in an optimal filtering framework to predict future arrivals. The algorithm is implemented for a large transit fleet in Seattle, Washington, and the prediction results for hundreds of locations are made widely available on the Web. An evaluation of the second busiest but most complex prediction site is presented to demonstrate the value of prediction over the use of schedules alone.

Under the rubric of advanced public transportation systems, a number of projects have been implemented to improve the distribution of pertinent information (departure time, vehicle delay, vehicle position) about a mass transit system directly to riders. An algorithm is presented here to predict the arrival of transit vehicles based on a few simple assumptions about the statistics of tracking transit vehicles. The goal of the algorithm is to accurately predict transit vehicle arrival times up to an hour in advance. Beyond the primary goal, there is an additional set of constraints on the algorithm that are imposed to facilitate implementation of the algorithm in real-world systems. These additional constraints are (*a*) the uncertainty in the arrival time is quantifiable, (*b*) the output of the algorithm is synchronous for display purposes, (*c*) lost or delayed data are handled efficiently, and (*d*) the prediction must be statistically better than that with schedules alone. In this paper, the time series of time and location pairs is used with historical statistics in an optimal filtering framework to predict future arrivals. Both an algorithm and a demonstration implementation for an existing large transit fleet are presented, and the results are compared with those for schedules alone.

The prediction algorithm presented here is sufficiently general to be applied to any transit property that operates a fleet that has an automatic vehicle location (AVL) system and that has repeated, scheduled service. The demonstration implementation is for the Metro King County transit property in Seattle, Washington.

Optimal filtering techniques are used to develop the prediction algorithm. Such techniques require that the problem be framed in a specific way and that certain properties of the model and model errors be true if the techniques are to be applicable. The underlying assumptions that allow optimal filtering techniques to be used to make the predictions are examined first and then construction of the Kalman filter, which acts as the predictor, is detailed.

ASSUMPTIONS

Any algorithm has basic underlying assumptions that will color the results. The assumptions used to create this algorithm are as follows:

- The vehicle locations are available irregularly, typically on a 1- to 5-min basis.
- Each scheduled trip is a realization of the stochastic process of the vehicle traveling the route.
- The stochastic process is represented by the ensemble of realizations.
- Vehicles are modeled as if moving with constant speed over a limited distance.
- Starting and stopping motions of the vehicles are included in the variability of the process model.
- The variability of the process model is normally distributed.
- There are known errors in the measurement of the location of the vehicles.

These assumptions allow the problem to be formulated in a statistical framework and fulfill the requirements necessary to use an optimal filtering technique, the Kalman filter, to make optimal estimates of the predicted time until arrival for individual vehicles.

The first assumption suggests that the time series created by sampling the transit vehicle locations is an undersampled representation of the actual vehicle path. With such an undersampled representation, the detailed path of the vehicle cannot be exactly duplicated. This assumption leads to the assertion that a detailed vehicle model for the motion is inappropriate for this application and a simple motion model is more appropriate. This model is the basis for the Kalman filter update equations presented here. The sample rate for the AVL system used in this implementation is on the order of 1–3 min, which is much too slow to capture behaviors such as acceleration and deceleration of the vehicles needed for detailed vehicle models.

The second assumption is based on the observation that the vehicles are operated to meet a regular schedule and that deviations from that schedule represent variability in this process. The bus driver's goal is to meet arrival time constraints that are posted on the schedule; however, each day the driver faces different traffic conditions that create variability in the arrival time. This suggests that if the vehicle is repeatedly observed on a scheduled daily trip, an average arrival time, with some variability, can be estimated. This notion dovetails with the third assumption.

The third assumption is that the scheduled vehicle trips be observed on different days as independent ensembles of a process that represents the bus traversing the route. This assumption enables one to use ensemble averaging to obtain overall statistics for the process. A sanity check on the first three assumptions is possible by using a time history diagram of vehicle position along a trip. Figure 1 is such a diagram, where the position along the trip is on the vertical axis and the time of day is on the horizontal axis. The data points represent the sum of data collected over a period of 2 months for this route that has six runs at different times of the day. The second assumption is supported by the overall linear shape and even distribution of points for each of the runs, which indicates that there is a general trend in the behavior of

University of Washington, Box 352500, Seattle, WA 98195-2500.

the vehicle on these trips. The third assumption is also supported by the clear trends and lack of dispersion in the data from a series of days.

The fourth assumption is that a synthetic time-to-arrival function can be created for every destination. This time-to-arrival function $\overline{T}_a(x)$ can be evaluated at each x, so that the time remaining until arrival is

$$\overline{T}_a(x) = \frac{x}{\overline{s}(x)} \tag{1}$$

The $\overline{s}(x)$ function is analogous to the speed the vehicle would have to travel if it were to travel at a constant speed from location x to the goal. This is not to be confused with the speed of the vehicle at x, which is not used in this approach.

From Figure 1, clearly a globally constant model for $\overline{s}(x)$ is unrealistic for most destinations. Therefore, a piecewise constant model is asserted. To do this one must select a destination of interest and transform the data into pairs of distance until arrival and time until arrival. Now the trip is divided into shorter segments over which it can be reasonably assumed that the data are approximated as a piecewise constant function in space and time. From this subsampling of the trip, an average overall arrival function value is estimated. If the points over the kth interval of size Δx_k about x_k are uniformly distributed over the region, the mean of the temporal values of the points is used to compute arrival time $\overline{T}_a(x)$. $\overline{s}(x)$ is constructed as

$$\overline{s}_k = \frac{x_k}{\overline{T}_a(x_k)} \tag{2}$$

$\overline{s}(x)$ is defined over the kth interval by $\overline{s}(x) = \overline{s}_k$. This approximation is presented in Figure 2, where the observed data are represented by points and $\overline{s}_k$ is the slope of the line between the centroid of the points and the arrival time.

A constant velocity model for subsets of the travel path is assumed in which the deviations from this model are identified as part of the randomness inherent in the process (for example, stopping and starting are effectively noise). The deviation of the points from the presumed linear behavior has a probabilistic distribution. To test the properties of this distribution, a line is fit through the points and the deviations of the individual points from this line are used as realizations of this probability distribution. It is hypothesized that the distribution is normal, and a standard distribution membership test, the Kolmogorov–Smirnov (K–S) test, is used to test this hypothesis (*1*). The K–S test computes the probability that two distributions are the same. The K–S test uses a metric of maximum absolute difference between two cumulative distribution functions (D). The metric is used to compute the following sum:

$$Q_{KS}(\lambda) = 2\sum_{j=1}^{\infty}(-1)^{j-1}e^{-2j^2\lambda^2} \tag{3}$$

$$\lambda = \left(\sqrt{N} + 0.12 + 0.11/\sqrt{N}\right)D \tag{4}$$

where N is the number of points. Larger K–S test values show that the two cumulative distributions are similar. To apply the K–S test to the data, the following steps are used: (*a*) Select an appropriate range of samples. (*b*) Find the sample mean and variance of these points. (*c*) Generate a normal distribution using the calculated mean and variance. (*d*) Use the selected points to create an unbiased estimator of actual distribution. (*e*) Compare the two distributions to find the maximum absolute distance (the K–S statistic). (*f*) Use the K–S statistic to compute the probability that the selected points came from the normal distribution.

Figure 3 presents the results of this test. It indicates the K–S probability for a variety of distances into trips on both a freeway, Interstate

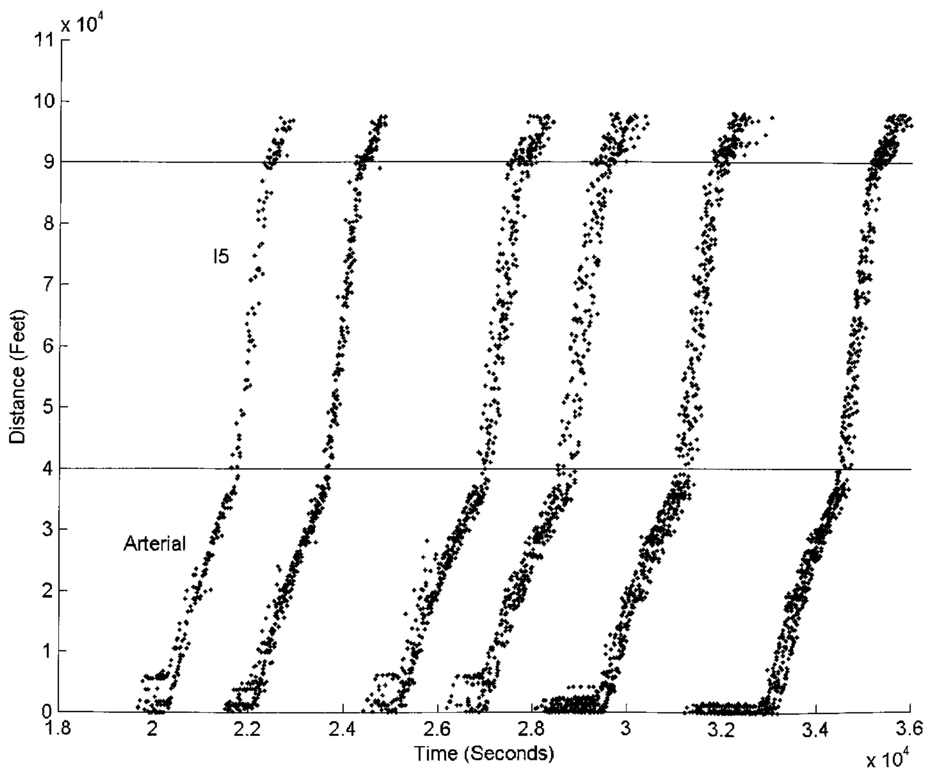

FIGURE 1 Space-time history for multiple ensembles of trips on Route 301.

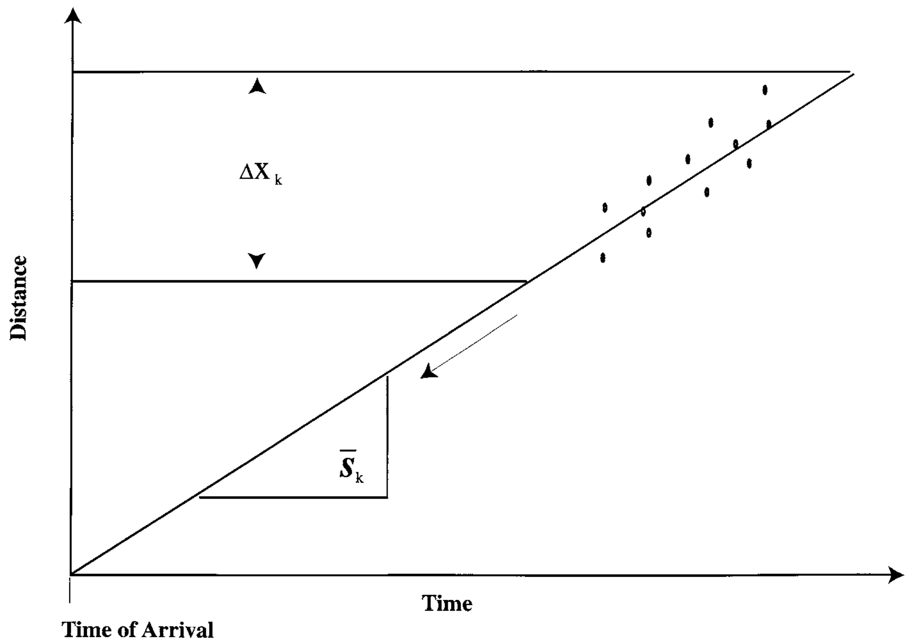

FIGURE 2 Time of arrival function $\bar{s}$.

5 (left), and an arterial, State Route 99 (right). The high probability of distribution membership justifies the fifth assumption of a normal distribution for the deviation of observed data from the linear motion model.

ALGORITHM

The prediction algorithm uses a Kalman filter whose state variables at the kth step are the vehicle location x_k, the time t_k, and the time until arrival b_k

$$X_k = \begin{bmatrix} x_k \\ t_k \\ b_k \end{bmatrix} \tag{5}$$

and the observables are the reported position z_k and the reported time of measurement τ_k,

$$Z_k = \begin{bmatrix} \tau_k \\ Z_k \end{bmatrix} \tag{6}$$

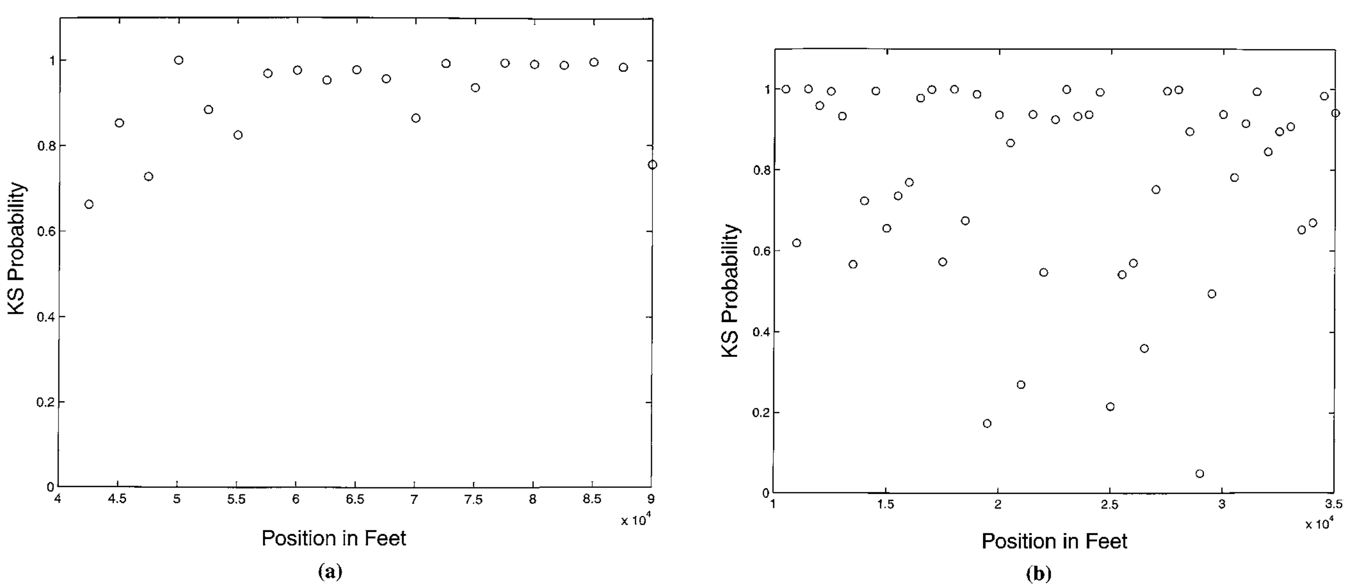

FIGURE 3 K–S test results for (*a*) Interstate 5 and (*b*) State Route 99.

The prediction of time of arrival is done synchronously at each Δt time step, and the observations of vehicle location are reported asynchronously as they are available. The model for the state transition at each time step of duration Δt_k has two parts: (*a*) a deterministic part involving the state variables, the time step (Δt), and the time-to-arrival function [$\bar{s}(x)$]; and (*b*) a stochastic part (w), assumed to be normally distributed with zero mean. The time update equations are

$$\bar{x}_{k+1} = \bar{x}_k + \bar{s}_k \Delta t_k + w_k^x \tag{7}$$

$$\hat{t}_{k+1} = \hat{t}_k + \Delta t_k + w_k^t \tag{8}$$

$$\bar{b}_{k+1} = \bar{b}_k - \Delta t_k + w_k^b \tag{9}$$

The data update for the state variables also has a deterministic and a stochastic component and takes the form of

$$\hat{x}_k = \bar{x}_k + w_k^x \tag{10}$$

$$\hat{t}_k = \frac{\hat{x}_k}{\bar{s}_k} + w_k^t \tag{11}$$

$$\hat{b}_k = \frac{\hat{x}_k}{\bar{s}_k} + w_k^b \tag{12}$$

Optimal filtering techniques often assume a relationship between the observables (z_k, τ_k), and the state vector takes the form of

$$z_k = \hat{x}_k + v_k^x \tag{13}$$

$$\tau_k = \hat{t}_k + v_k^\tau \tag{14}$$

where v is the stochastic noise in the observation process. With these variables, the problem is posed in the form of a linear Kalman filter:

$$\mathbf{X}_{k+1} = \mathbf{A}\mathbf{X}_k + \mathbf{\Gamma}_k \mathbf{u}_k + \mathbf{w}_k \tag{15}$$

$$\mathbf{Z}_k = \mathbf{H}_k \mathbf{X}_k + \mathbf{v}_k \tag{16}$$

so that the time update matrices are

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{\Gamma}_k = \Delta_t \quad \mathbf{u}_k = \begin{bmatrix} \bar{s}_k \\ 1 \\ -1 \end{bmatrix} \tag{17}$$

and the data update matrices are

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{1}{s_k} & 0 & 0 \end{bmatrix} \quad \mathbf{H} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{18}$$

The generalized solution to the Kalman filter time update (*2*) is

$$\mathbf{P}_k = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q}_{k-1} \tag{19}$$

$$\mathbf{X}_k = \mathbf{A}\mathbf{X}_{k-1}\Gamma_k \mathbf{u}_{k-1} \tag{20}$$

and the data update is

$$\mathbf{P}_{k1} = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q}_{k-1} \tag{21}$$

$$\mathbf{K}_{kg} = \mathbf{P}_{k1}\mathbf{H}^T\left[\left(\mathbf{H}\mathbf{P}_{k1}\mathbf{H}^T\right) + \mathbf{R}\right]^{-1} \tag{22}$$

$$\mathbf{P}_k = \mathbf{P}_{k1} - \mathbf{K}_{kg}\mathbf{H}\mathbf{P}_{k1} \tag{23}$$

$$\mathbf{X}_k = \mathbf{A}\mathbf{X}_{k-1} + \mathbf{K}_{kg}[\mathbf{Z}_k - (\mathbf{H}\mathbf{A}\mathbf{X}_{k-1})] \tag{24}$$

The covariance matrices for the update and measurement are

$$\mathbf{Q}_k = E\{\mathbf{w}_k \mathbf{w}_k^T\} \qquad \mathbf{R}_k = E\{\mathbf{v}_k \mathbf{v}_k^T\}$$

$$\mathbf{Q}_k = \begin{bmatrix} \sigma^2_{w_k^z} & 0 & 0 \\ 0 & \sigma^2_{w^b} & 0 \\ 0 & 0 & \sigma^2_{w^t} \end{bmatrix} \mathbf{R}_k = \begin{bmatrix} \sigma^2_{v^\tau} & 0 \\ 0 & \sigma^2_{v^z} \end{bmatrix} \tag{25}$$

This algorithm produces the optimal estimate of the arrival time given the information provided to the filter. However, predicting the future is always a challenging activity. To compare the predictions with vehicle behavior, it is necessary to estimate actual arrival time. Because the vehicle is being tracked irregularly, there is no guarantee that the location will be reported just as the vehicle arrives. To get an estimate of the real arrival, the location report is recorded just before arrival and just after arrival and the actual arrival time (T_a) is linearly interpolated. The filter continuously predicts the arrival as a function of both space and time. The statistics of the deviation of the predictions from the actual are presented as a probability surface in space and time. Figure 4*a* indicates the probability of deviation of the prediction from actual, in minutes, on the front axis (–10 to 10) and the time until arrival for which the prediction was made on the side axis (0 to 30). A similar function for the relationship between the schedule and the actual behavior is presented in Figure 4*b*. The surfaces in Figure 4 were created with the predictions made over the course of 1 day for the second busiest location in the Seattle Metro Transit's service. This location, as well as being the second busiest, is the most complex in terms of the types of trips passing this point. Comparing these two surfaces suggests that prediction with dynamic information has two to four times smaller errors than when the schedule is used alone. This indicates that there is a significant gain in information over the schedule for transit users.

SEATTLE IMPLEMENTATION OF MYBUS

A Web-based system that makes predictions for 1,200 operating vehicles at each of the 1,500 time points identified in the transit carrier's schedule database has been implemented. The implementation is presented by first describing the AVL system used by the transit carrier King County Metro and then detailing the software architecture to support the implementation.

The King County Department of Metropolitan Services (Metro) implemented an AVL system for its transit fleet in 1992. The AVL system is used by the transit dispatchers for command and control

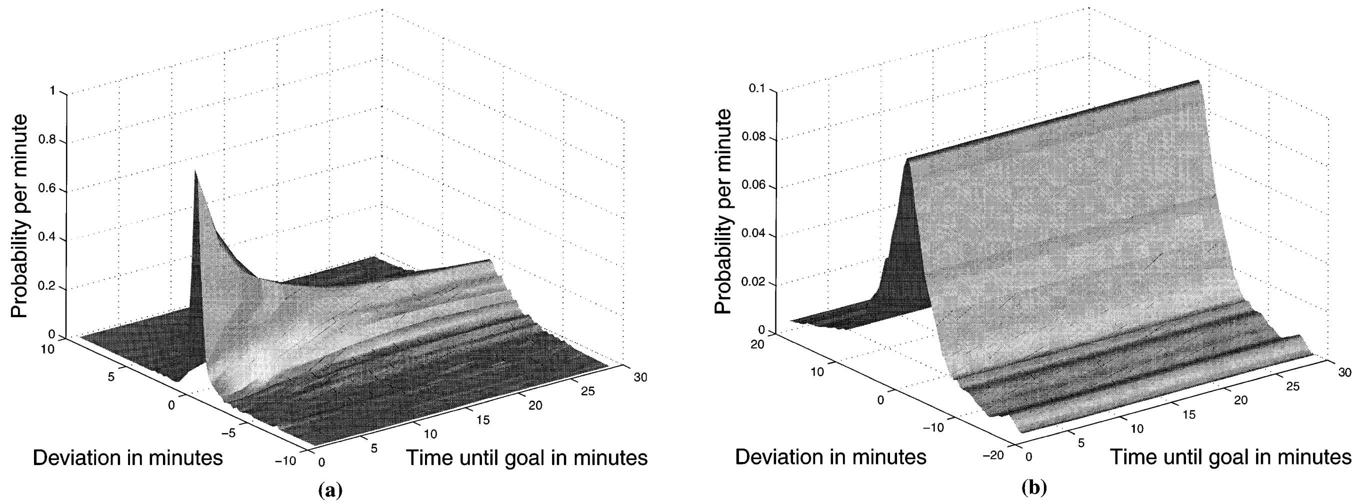

FIGURE 4 Probability of deviation from actual arrival time for (*a*) algorithm and (*b*) schedule.

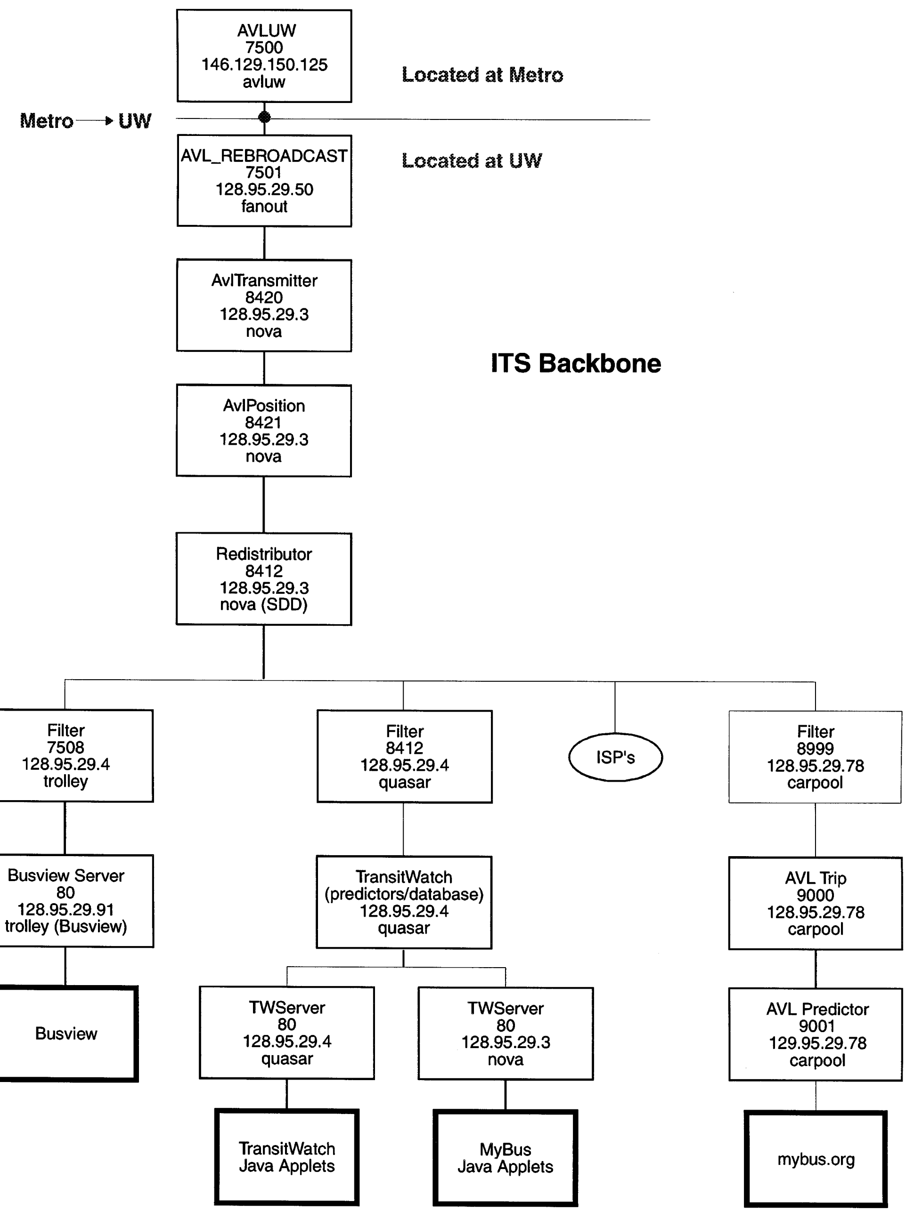

FIGURE 5 MyBus architecture (UW = University of Washington).

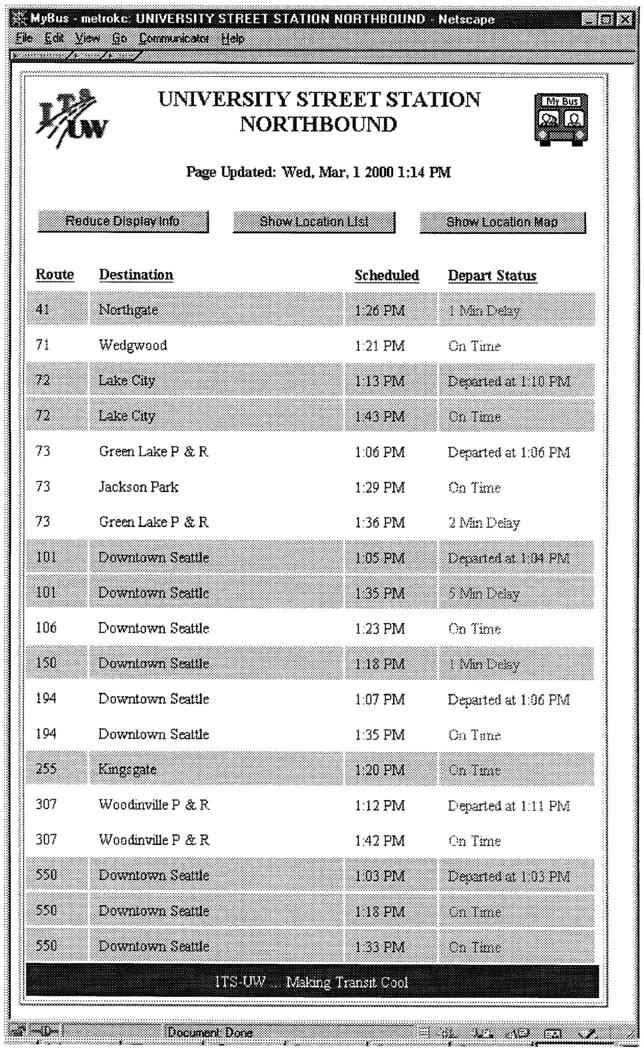

FIGURE 6 MyBus display.

purposes. There are in excess of 1,200 AVL-equipped vehicles operating on 240 bus service routes within King County's 5180-km^2 (2,000-mi^2) service area. Metro's AVL system is composed of three subsystems: geographic information system (GIS), communication, and data management (*3*). These three subsystems work in tandem, and the position information used in the work presented here is obtained by eavesdropping on the communication network connecting the communications and GIS systems. The geolocation of the vehicles is done by using odometry and digital maps. Each vehicle has a target and a sensor located on one wheel, and every time the target moves past the sensor, a click is recorded. Additionally, when a vehicle passes a signpost transmitter, the onboard receiver records the signpost identification and the time it was passed. The transit vehicles are polled irregularly, but about every minute, and the odometry data along with a unique vehicle identification and any signposts encountered are radioed back to the central management site. The distance traveled along a preplanned route and schedule information is used in the prediction algorithm as the measurement data.

The algorithm just presented is implemented as a Web application called MyBus. This application is constructed from a series of collaborating components (*4*) through which data flow. Each component performs data fusion on the data to add information to the data flow. The component labeled AVL Predictor in Figure 5 (lower right) implements the prediction algorithm described previously. The data stream resulting from this component includes predictions of departures for the vehicles found in the data stream at all the time points in the schedule database of the transit carrier.

The component labeled myus.org in Figure 5 is the Web server that creates the MyBus HTML screens. An example of this screen is presented in Figure 6. The URL http://mybus.org provides a link to the implementation for over a thousand locations in the Seattle metropolitan region. To demonstrate the portability of the concept, the information for Portland has also been implemented and future work is planned to demonstrate real-time prediction for Portland.

Not only is reliable and accurate prediction possible for a large transit fleet, but there also is significant temporal value to the transit-using public in making such predictions widely available.

REFERENCES

1. Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C*, 2nd ed. Cambridge University Press, Cambridge, U.K., 1988.
2. Anderson, B. D. O., and J. B. Moore. *Optimal Filtering.* Prentice Hall, Englewood Cliffs, N.J., 1979.
3. Dailey, D. J., M. P. Haselkorn, K. Guiberson, and P. J. Lin. *Automatic Transit Location System.* Final Technical Report WA-RD 394.1. Washington State Department of Transportation, Seattle, Feb. 1996.
4. Dailey, D. J., M. P. Haselkorn, and D. Meyers. A Structured Approach to Developing Real-Time, Distributed Network Applications for ITS Deployment. *ITS Journal*, Vol. 3, No. 3, 1996, pp. 163–180.