A Dynamic Bus-Arrival Time Prediction Model Based on APC Data

Mei Chen*

Department of Civil Engineering, University of Kentucky, 267 Raymond Bldg., Lexington, KY 40506-0281

Xiaobo Liu

Interdisciplinary Program in Transportation, New Jersey Institute of Technology, Newark, NJ 07102

Jingxin Xia

Department of Civil Engineering, University of Kentucky, 216 Raymond Bldg., Lexington, KY 40506-0281

&

Steven I. Chien

Department of Civil & Environmental Engineering, New Jersey Institute of Technology, Newark, NJ 07102

Abstract: *Automatic passenger counter (APC) systems have been implemented in various public transit systems to obtain bus occupancy along with other information such as location, travel time, etc. Such information has great potential as input data for a variety of applications including performance evaluation, operations management, and service planning. In this study, a dynamic model for predicting bus-arrival times is developed using data collected by a real-world APC system. The model consists of two major elements: the first one is an artificial neural network model for predicting bus travel time between time points for a trip occurring at given time-of-day, day-of-week, and weather condition; the second one is a Kalman filter-based dynamic algorithm to adjust the arrival-time prediction using up-to-the-minute bus location information. Test runs show that this model is quite powerful in modeling variations in bus-arrival times along the service route.*

To whom correspondence should be addressed. E-mail: mchen@engr.uky.edu.

1 INTRODUCTION

1.1 Background

Automatic passenger counter systems have been implemented in various public transit systems. The primary benefit of this technology lies in the increased quantity and quality of information collected, including vehicle locations (newer version), speeds, travel times, occupancies, etc. Such information can enhance the capability of transit passenger information systems as well as assist in proactive transit planning and management systems, and improve the overall service quality.

Transit agencies face increased demand and the challenge to know ahead of time, whether or not their buses are running on schedule. It is necessary to know when buses will arrive at the designated stops. Bus travel times are prone to a high degree of variability mainly due to traffic congestion, ridership distribution, and weather condition. There is a need to develop a model for predicting bus-arrival times and improve the

quality of information provided to customers. Providing up-to-minute transit information can reduce the negative impact of schedule/headway irregularities on transit services. There is also a need to examine the variability of bus travel times to prepare more accurate schedules and assist transit agencies to restore service disturbances.

The travel time deviations of buses are usually caused by several stochastic factors. Transit vehicle (e.g., buses) operations are frequently disturbed by right-of-way competence with other types of vehicles, congestion on service route at different times of the day, intersection delays, variation in demands, and excessive dwell times at stops. The resulting impact of these factors on the transit system comprises bunching between pairs of operating vehicles, increasing passenger waiting times, deterioration of schedule/headway adherence, unsmooth intermodal transfers, increasing cost of operation and traffic delays, etc. All these factors will reduce the level of service of the transit agency and discourage riders to use the transit system. One way to mitigate the impacts is to provide accurate information of vehicle-arrival/departure times and expected delays at all major stops.

The deployment of travel time prediction models in advanced public transportation systems can benefit both transit agencies and riders. Providing accurate real-time transit vehicle-arrival time information to the passengers can improve transit level of service. With accurate vehicle-arrival information, transit users may efficiently schedule their departure time from work places/homes and they can catch the buses or make successful transfer at transit stations, reducing passengers' waiting times at the stops, and thus, enhance the quality of service. On the other hand, transit agencies can manage and operate their systems more effectively with real-time dispatching and scheduling. In case of service disruption, proper control actions (e.g., increase/decrease operating speed or longer dwell times at some stops) could be taken based on real-time arrival information, to restore schedule or headway and thus maintain a desirable level of service.

The public transportation systems that offer bus-arrival time information include the AC Transit in Alameda County, California (http://www.actransit.org), the City–University–Energysaver (CUE) bus system in Fairfax, Virginia (http://www.fairfaxva.gov/CUEBus/CUEBus.asp), the Vail Bus Service in Vail, Colorado (http://www.ci.vail.co.us), the Municipal Railway system in San Francisco, California (http://www.sfmuni.com), the Tri-met Transit Tracker system in Portland, Oregon (http://www.trimet.org), the King County Metro Transit in the State of Washington (http://www.its.washington.edu/transitwatch), etc.

1.2 Prediction method

Various methods have been developed for bus-arrival time prediction including time series, artificial neural network, and Kalman filtering technique. Dailey et al. (2001) developed an algorithm to predict bus-arrival time based on Kalman filter, which takes data collected by the onboard automatic vehicle location system as input. The time and distance to the destination (bus stop) is predicted once latest information on bus location and time is obtained.

Lin and Zeng (1999) developed a set of algorithms to predict bus-arrival time for a transit traveler information system in Blacksburg, Virginia. Global positioning system (GPS) data were gathered at variable time intervals, including location and time label. However, because of its inherent constraints (no fixed sample period, existence of erroneous reports), the accuracy of the prediction is compromised. Tests showed that the algorithm that uses GPS bus location data, bus timetable, delay, and dwell time at time checkpoint as input had the best performance.

The TransLink Lab at the Texas Transportation Institute developed two algorithms for predicting travel time of campus buses at the Texas A&M University (TransLink, n.d.). In the time-based algorithm, travel time estimates were obtained by counting the number of 1-minute zones (based on historical travel time data) between the current location of the bus and its destination. The distance-based algorithm modeled the impact of the distance to destination and time-of-day (reflected by running speed variations during class time and class break when a large number of pedestrians are present) on bus travel time. The latter algorithm had better performance but was more expensive to calibrate.

Chien et al. (2002) developed an enhanced artificial neural network (ANN) model to predict bus-arrival time. Specifically, an adaptive algorithm was applied to adjust the prediction based on the difference between the predicted and actual (simulated) arrival times of the bus dispatched previously. The data used in model testing was generated from simulation model.

Shalaby and Farhan (2003) presented a Kalman filter-based model to predict bus travel time. Similar to the enhancement technique introduced by Chien et al. (2002), it was assumed that all buses in the fleet are equipped with GPS and APC units. The travel time prediction relies on the experience of the previous bus. However, it would be very expensive in reality to equip all buses traveling on one route with such devices to get arrival information for each bus.

1.3 Objective

APC data from real-world transit operations provide a valuable input to the prediction of bus-arrival times. The objective of this study is to use such data in developing a methodology for bus-arrival time prediction, which could accommodate various factors that may affect bus-arrival time (such as weather) and incorporate real-time information into the prediction.

In most cases, buses share right-of-way with other vehicles in the traffic stream. The congestion (recurrent or nonrecurrent) on roadways is a major cause of delay in bus-arrival time. Recurrent congestion is periodically caused by higher traffic demand or insufficient road capacity, while the nonrecurrent congestion is caused by incidents (e.g., accident, adverse weather conditions, etc). However, among the numerous factors contributing to congestion, it is difficult to define a precise cause and effect relationship between traffic/roadway conditions and bus travel time. In other words, the impact of a particular factor (e.g., time-of-day) on bus travel time may not be quantified in an explicit format. To avoid defining an explicit function to model a complicated system, the concept of ANN was adopted in this study. Such a method is capable of modeling the complicated input/output relationship without specifying a form of function. Furthermore, it does not require independency among input variables, as required by other methods such as regression analysis.

Considering the impact of unexpected incidents during the trip, we also develop a dynamic algorithm for bus-arrival time prediction based on the Kalman filtering technique. It uses the most recent information on bus location to fine-tune the estimated bus travel time from ANN models.

This work can be viewed as the extension of what was presented by Chen et al. (2003) in which an ANN model was developed and trained using APC data. This article is organized as follows. Section 2 describes data collection and the processing procedure. Section 3 presents the bus-arrival time prediction methodology. Section 4 contains results and analyses including performance evaluation of the methodology.

2 DATA COLLECTION AND PROCESSING

2.1 APC data

The bus route 62 of NJ Transit operating in Essex county, Union county and Middlesex county in New Jersey was selected as the studied route of this study. The APC devices have been installed in some buses running along this route to monitor bus operational information at each bus stop along the trip, including main stops listed in the timetable (i.e., time points). The whole bus route 62 starts from Newark Penn Station and ends at Perth Amboy with total distance of 29.5 miles on the outbound. In total, there are 17 time points located along the bus route as shown in Figure 1, for which the NJ Transit provides the scheduled arrival times on the timetables. The OD pair on route 62 between Woodbridge Center Mall and Newark Penn Station was selected as the studied OD pair, for which bus service was provided by different patterns. A slight difference in their routing results in a different number of time points. Bus running on
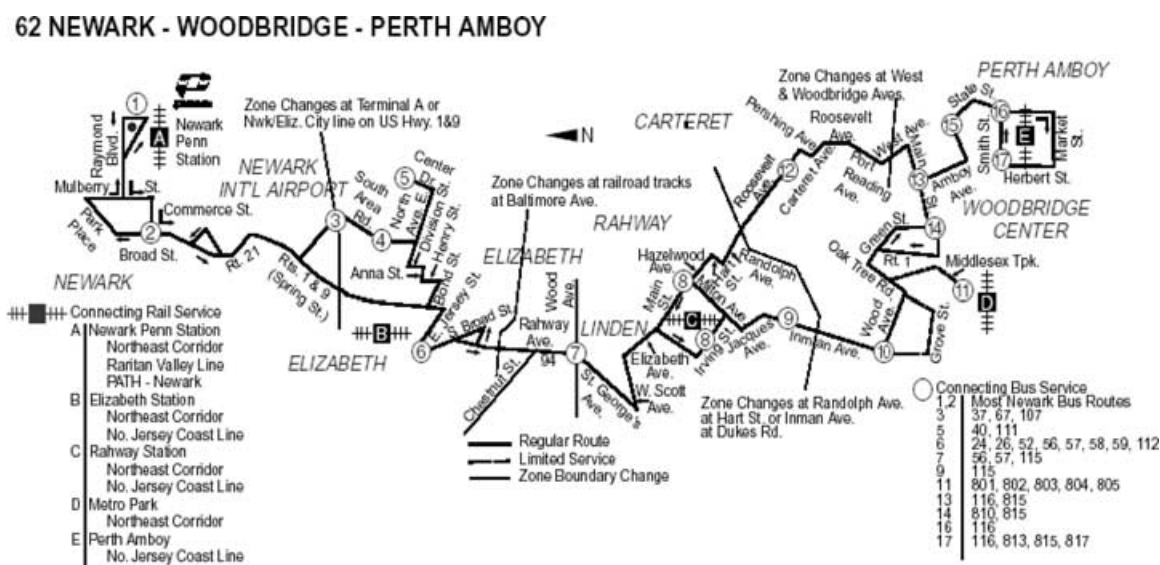

Fig. 1. Configuration of bus route 62 (source: http://www.njtransit.com/pdf/bus/T0062.pdf).

Table 1
Main index in the APC data set

Variable	Description
Sched arrival time	Scheduled arrival time at each time point
Transit day	Date of the service
Time of day	Time period of bus operation
Open time	Recorded bus door opening time
Close time	Recorded bus door closing time
Stop sequence	A unique number to all intended stops along the route.
Time point ID	Time point indicator number
Trip status	Trip status (start or end)
Lat	Latitude
Lon	Longitude
On	Number of boarding passengers at a stop
Off	Number of alighting passengers at a stop
Stop distance	Travel distance between two consecutive stops
Dwell time	The bus door open time at any stop
Leg time	Inter-stop travel time
Leave psgr load	Number of onboard passengers when the bus leaves a stop
Arrive psgr load	Number of onboard passengers when the bus arrives at a stop
Pattern ID	Unique index associated with a pattern in each pick data
Trip index	Unique index associated with a trip

different patterns will be assigned a unique pattern abbreviation. Four patterns (WM-AP, PA-WM, WMIAP, and PAIWM) are selected as the study patterns for developing prediction model, which correspond to inbound with 12 time points, outbound with 12 time points, inbound with 14 time points, and outbound with 14 time points, respectively.

The collected APC data of year 2002 consist of January Pick (from January to June, 2002), June Pick (from June to September, 2002), and September Pick (from September to December, 2002), which were retrieved from the APC database at NJ Transit. There are 447 trips made by buses equipped with APC units during the weekdays within this 1-year period, and there are 5,427 segment travel times in total. The main attributes in each APC record used for prediction development are summarized in Table 1.

2.2 Weather data

The weather information was obtained from the National Climatic Data Center (http://lwf.ncdc.noaa.gov/oa/climate/stationlocator.html). Newark International Airport Station in NJ is selected as the observation station because it is the only station that collected weather data covering the studied route. The weather information includes hourly temperature (e.g., dry bulb temperature), precipitation (e.g., rainfall and snowfall), and sky conditions (e.g., visibility, wind speed). Precipitation data is selected as a training factor in prediction model development as it is found to have the most significant impact on travel time through preliminary analysis.

2.3 Data processing

The desired data structure for observing bus operations and developing prediction models is a set of successive time point records. Thus, the actual bus travel times between time points can be compared with that posted on the timetable. However, APC can only record activity on the actually made bus stops, which led to some time point records missing when buses skipped those time points. To generate a completed set of time point information, the bus-arrival time at missing time points were interpolated based on the available information at upstream and downstream time points assuming the travel speed to be constant between two consecutive time points. After screening the outlier in original APC data set, the interpolated data are merged with the weather data as one consistent data set for developing prediction models.

3 MODEL DEVELOPMENT

The bus-arrival time prediction methodology consists of two major components, an ANN model and a Kalman filter-based dynamic algorithm. For each trip pattern, an ANN model is developed based on historical trip data collected by APC units. This model should be retrained regularly (e.g., daily or weekly—depending on the frequency of database update). The ANN output, which is the bus travel time between time points, serves as a baseline estimate of travel time. For an individual trip, the dynamic algorithm takes the most recent information on bus location and time when it arrives at that location as the input and combines them with the baseline estimate to predict bus-arrival times at downstream time points. The general flowchart of the methodology is shown in Figure 2.

3.1 Artificial neural network

Using ANN in forecasting has become more popular in transportation research. Related studies include Hua and Faghri (1994), Chin et al. (1994), Dougherty (1995), Kalaputapu and Demetsky (1995), Smith and Demetsky (1995), Zhang et al. (1997), Chang and Su (1995), Ding and Chien (2000), and McFadden et al. (2001). Designed
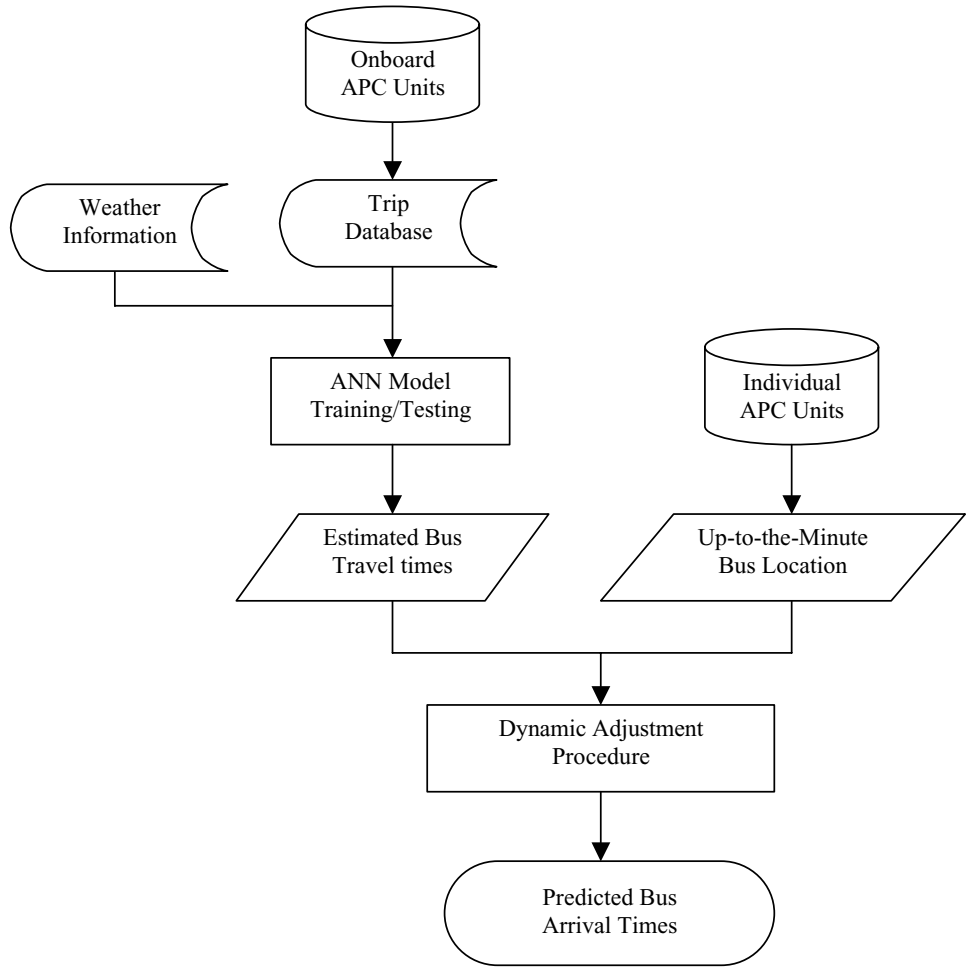
Fig. 2. Flowchart of bus-arrival time prediction methodology.

with versatile parallel distributed structures and adaptive learning processes, ANN is considered as a promising approach to describing complex systems such as transit operation that is affected by various inter-correlated and time-varied factors.

Unlike other prediction models, ANN does not require a specific form of function. This eliminates the need of function development and parameter estimation for nonlinear and time-varied systems. A well-trained ANN could capture complex relationships between the dependent variables (output such as bus-arrival time) and a set of explanatory/independent variables (input such as traffic conditions and passenger demand). Therefore, ANN technique could be very useful in prediction when it is difficult or even impossible to mathematically formulate the relationship between the input and output.

Multi-layer perceptron (MLP) type ANN architecture was chosen in this study as it is generally easy to use and can approximate almost any input/output map. It has been widely used in countless applications

including forecasting in complicated transportation systems. The proposed ANN model is in the structure shown in Figure 3. Because the APC data contain trips under different patterns, it was necessary to divide them into different data sets for model development. In addition, as the bus timetable has determined the operating schedule, the set of possible values for the time-of-day variable is fixed for each pattern. For example, trips for patterns WM-AP and PA-WM only start during MP, AP, E, and LN periods, while those for patterns WMIAP and PAIWM only start during EM, LM, MD, EA, E, and LN periods. In this study, a separate neural network model is established for each bus operational pattern to predict travel times between time points.

Among those variables that may contribute to the variation of bus-travel time, we select four input variables, day-of-week (x_1), time-of-day (x_2), weather (x_3), and segment (x_4). The day-of-week variable is self-explanatory; it could take five possible descriptive values, Monday through Friday, because only weekday trips
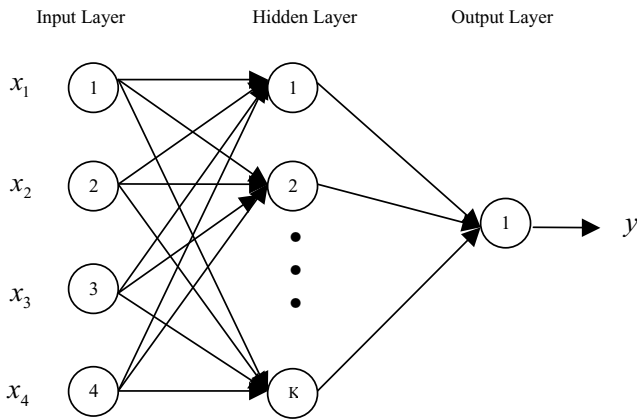
Fig. 3. Proposed ANN model.

Table 2
Structure of ANN models

Model	No. of hidden layers	No. of PEs (K)
WM-AP	1	4
PA-WM	1	5
WMIAP	1	6
PAIWM	1	5

are used for model development considering the significant difference in service between weekdays and weekend. A day is divided into eight three-hour time periods, early morning (EM), morning peak (MP), late morning (LM), mid-day (MD), early afternoon (EA), afternoon peak (AP), evening (E), and late night (LN). The time-of-day variable takes the descriptive symbol of the time period within which the trip was initiated. For the variable weather, only the precipitation factor is considered based on data availability. The variable could take two possible values, yes or no, corresponding to precipitation and nonprecipitation conditions, respectively. The segment variable identifies the section between two adjacent time points. Depending on the pattern for which the ANN model is developed, this variable could be 1 through 11 (for patterns WM-AP and PA-WM) or 1 through 13 (for patterns WMIAP and PAIWM).

The output of the model is the travel times between two adjacent time points (y). The arrival time can then be calculated based on departure time and the estimated travel time.

This model has one hidden layer and a number of processing elements (PEs, a.k.a. neurons). We went through a trial and error process to determine the appropriate numbers of hidden layers and processing elements in each layer. Usually, one or two hidden layers are sufficient to create a model that is able to predict reasonably well. It is important to note that increasing the number of hidden layers beyond two often undesirably reduces the network's ability to make better generalization or better ANN model. Through comparing the performance of ANNs with one and two hidden layers on a preliminary basis, we found that the former outperformed the latter in most cases. In addition, the more layers we have, the more training data are needed to keep the ratio between the numbers of network weights and training data records at a desirable level. Thus, only MLPs with a single hidden layer were further studied. We also tested the ANNs with various numbers of processing elements in the hidden layer (K, as shown in Figure 3) and identified the ones with the best performance. The summary of ANN structures is listed in Table 2.

The training procedure chosen is the most commonly used back-propagation algorithm, whose learning process is shown in Figure 4. The objective of the learning process is to find a weight matrix that minimizes the mean squared error (MSE), which is defined as the average squared error between the ANN
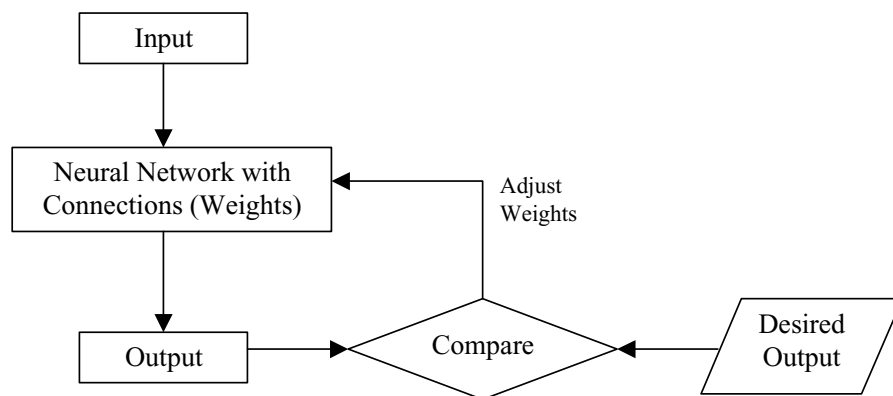

Fig. 4. BPN training procedure.

predictions and the actual values of travel times, as shown below.

$$\text{MSE} = \frac{\sum_{j=0}^{P} \sum_{i=0}^{N} (d_{ij} - y_{ij})^2}{NP}$$

where P is the number of output neurons; N is the number of samples in the data set; d_{ij} is the desired output for sample i at neuron j; y_{ij} is the network output for sample i at neuron j.

However, Vapnik (1995) indicated that an MLP network trained using a back-propagation algorithm could have the problem of overtraining beyond a critical point of learning. The symptom of overtraining is that the network performs well with data in the training set, while its performance over the test data set (those "unseen" by the network) starts to deteriorate. To solve this problem, we used an alternative stopping criterion, stopping with the cross validation, for the learning process.

The training data were divided into two sets, the training set and the cross validation set that was 20% of the total training samples. The cross validation set contains records with all possible variable values. The network performance with current weight matrix was tested against the cross validation set regularly during the learning process. When the MSE associated with the cross validation set started to increase, the learning process was terminated. This procedure prevents overtraining and ensures maximum generalization of the network (Principe et al., 2000).

3.2 Dynamic algorithm

The ANNs developed previously are based on historical data pool of bus trips. Due to the lengthy training process, it is difficult to provide online prediction in real-world applications. While new trip data are added into the database regularly, training can be conducted after each addition to ensure the ANN model is up to date. Nevertheless, the ANN models still do not have the dynamic feature of adjusting prediction using the most recent information from the trip.

A dynamic algorithm is developed based on the Kalman filtering technique. It enables online adjustment of arrival-time estimates for a particular trip based on its available travel-time information up to the moment the estimation is conducted.

Let t_k denote the travel time from time point k to the given destination (i.e., the time point for which arrival-time prediction is performed), $T_{k,k+1}$ denote the travel time from time point k to time point $k + 1$, and s_k denote the travel time from origin to time point k. Then the travel time from time point $k + 1$ to the destination t_{k+1} can be calculated as $t_{k+1} = t_k - T_{k,k+1}$, and the travel

time from origin to time point $k + 1$ can be calculated as $s_{k+1} = s_k + T_{k,k+1}$.

If z_k denotes the observed travel time from origin to time point k, then $z_k = s_k$. Let $x_k = (t_k \ s_k)^T$, the Kalman filter can be formulated as

$$x_{k+1} = \Phi_k x_k + u_k + w_k$$
$$z_k = H_k x_k + v_k$$

in which,

$$\Phi_k = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \ H_k = (0 \quad 1), \ u_k = T_{k,k+1} \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

w_k and v_k are white noises associated with the transition process and measurement, respectively. They are assumed to have zero mean and variances of Q_k and R_k, respectively.

The filtering procedure is outlined as follows.

Step 1: Initialize state variables. Set $\hat{x}_0 = (\hat{t}_0 \ \hat{s}_0)^T$, in which $\hat{t}_0$ is the estimated total travel time from the origin to the destination, and $\hat{s}_0$ is set to be 0 based on its definition.

Step 2: Initialize covariance P_0 when $k = 0$.

Step 3: State variable extrapolation.

$$\hat{x}_{k+1}^- = \Phi_{k+1} \hat{x}_k + u_k$$
$$= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} t_k \\ s_k \end{pmatrix} + T_{k,k+1} \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

in which $T_{k,k+1}$ can be obtained from estimates based on historical data.

Step 4: Covariance extrapolation.

$$P_{k+1}^- = \Phi_{k+1} P_k \Phi_{k+1}^T + Q_k$$

Step 5: Kalman gain (K) computation.

$$K_{k+1} = P_{k+1}^- H_{k+1}^T \left(H_{k+1} P_{k+1}^- H_{k+1}^T + R_{k+1} \right)^{-1}$$

Step 6: State variable update.

$$\hat{x}_{k+1} = \hat{x}_{k+1}^- + K_{k+1} \left(z_{k+1} - H_{k+1} \hat{x}_{k+1}^- \right)$$

Stop if time point $k + 1$ is the destination. Otherwise, go to step 7.

Step 7: Covariance update.

$$P_{k+1} = P_{k+1}^- - K_{k+1} H_{k+1} P_{k+1}^-$$

Go to step 3.

This algorithm is implemented in the following manner. For a particular bus trip, before the trip begins, the ANN model for this travel pattern provides a set of initial estimates of travel time from the origin to each downstream time point. These estimates are based on the values of the

input variables (day-of-week, time-of-day, weather, and segment) for this particular trip. When the bus reaches the second time point, the actual travel time between the origin and this time point (i.e., s_1) is recorded, and the estimated travel times from the current stop to all downstream time points are updated by the filter based on this information. The output at each iteration is a set of the predicted arrival times at all downstream time points. As the bus proceeds along its route, the prediction is updated whenever the most recent arrival information is obtained. The process is repeated till the bus reaches the final destination.

This procedure is designed to take into account the most recent status of an individual trip and the historical scenario in the prediction process. The ANN models provide static estimation of travel times based on historical trip data; while the dynamic algorithm enables the dynamic adjustment of the travel time prediction based on real-time trip status.

4 RESULTS ANALYSES

4.1 ANN output

The performance of a neural network model can be tested using data that were never "seen" by the network. That is, the test data cannot be used to train or validate the network. For each ANN model developed previously, 10% samples were set aside as testing data. The performance of the ANN model on the test data can be evaluated using the MSE measurement. Table 3 shows the MSEs for training and test sets for each model (pattern), in which all ANN models display reasonable performance.

Normally, the test MSE is expected to be higher than the training MSE as the test data have not been "seen" by the network. However, there are indications that the models could perform very well on "unseen" data, such as the model for pattern WMIAP.

To evaluate the performance of the prediction model, the variations between schedule and actual travel time and the variation between prediction (model output) and actual travel time are compared. The prediction accuracy

Table 3
Performance measure for ANN models

Pattern	Training MSE	Test MSE
WM-AP	0.011	0.023
PA-WM	0.016	0.016
WMIAP	0.015	0.010
PAIWM	0.008	0.009

was evaluated by computing the root mean squared error (RSME), which can be obtained from

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$$

where N is the number of test samples; y_i is the actual travel time of sample i; $\hat{y}_i$ is the ANN estimated travel time of sample i.

The RMSEs are calculated for each bus route segment (i, defined as the segment between time points i and $i + 1$). For comparison purpose, we also compute the RMSEs with respect to the scheduled travel time, with which we replace $\hat{y}_i$ in the above equation. Figure 5 shows the comparison of RMSEs of ANN output and schedule for each of the four ANN models.

It can be observed that in all cases, the ANN RMSEs for all route segments are smaller than those of the timetable. The difference could be as large as 150 seconds, approximately. This indicates that, compared to the timetable, the ANN models generally provide better indication of the bus-arrival time between two adjacent time points.

4.2 Dynamic algorithm output

The dynamic algorithm, based on the Kalman filter technique, is designed to fine-tune the arrival-time prediction based on neural network outputs (i.e., the predicted travel time between time points). Whenever the bus reaches a time point, the travel/arrival-time prediction can be adjusted by the newly obtained travel/arrival-time information.

For each trip in the database, the procedure is applied during the trip, in which the estimated travel time to each downstream time point is updated when the most recent arrival information is retrieved from the bus. To evaluate its performance, it is necessary to compare the predictions generated by the dynamic algorithm and ANN, and their deviations from the actual travel time. For example, estimated travel time from the origin (time point 1) to all downstream time points are obtained from the algorithm output. This output is then compared to the actual travel times between the same origin and destinations (downstream time points), and the RMSE is calculated for each OD pair. The RMSEs for the neural network model and the timetable could be obtained similarly. Figure 6 shows the performance comparison of the dynamic algorithm, the ANN models, and the bus timetable for trips originating from time point 1 and arriving at each downstream time point, from 2 to 12 (for patterns WM-AP and PA-WM) or 14 (for patterns WMIAP and PAIWM).

It can be observed that the dynamic algorithm, with a lower RMSE, always performs better than the
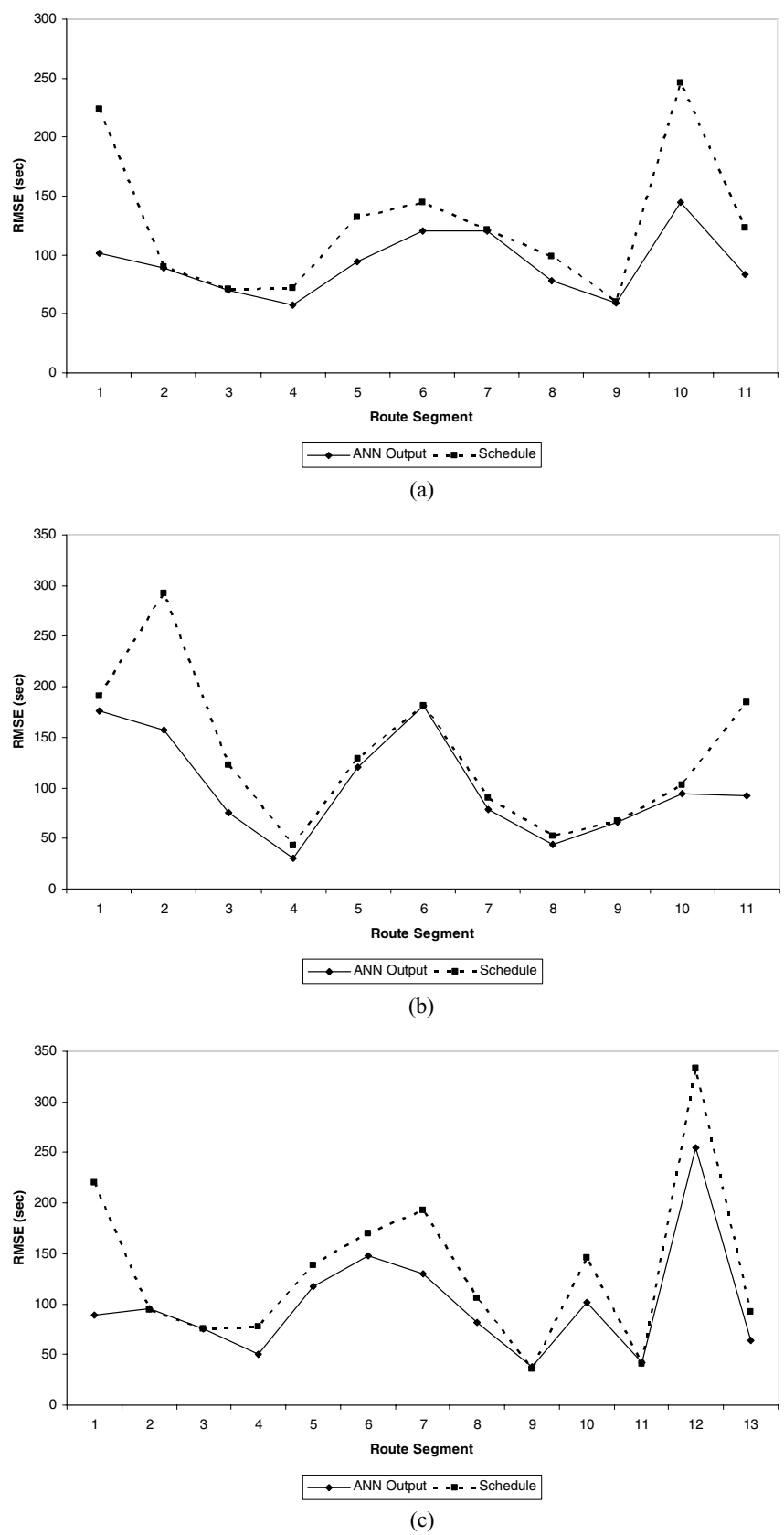
Fig. 5. Performance of ANN models compared to bus timetable (a) Pattern WM-AP (b) Pattern PA-WM (c) Pattern WMIAP (d) Pattern PAIWM.
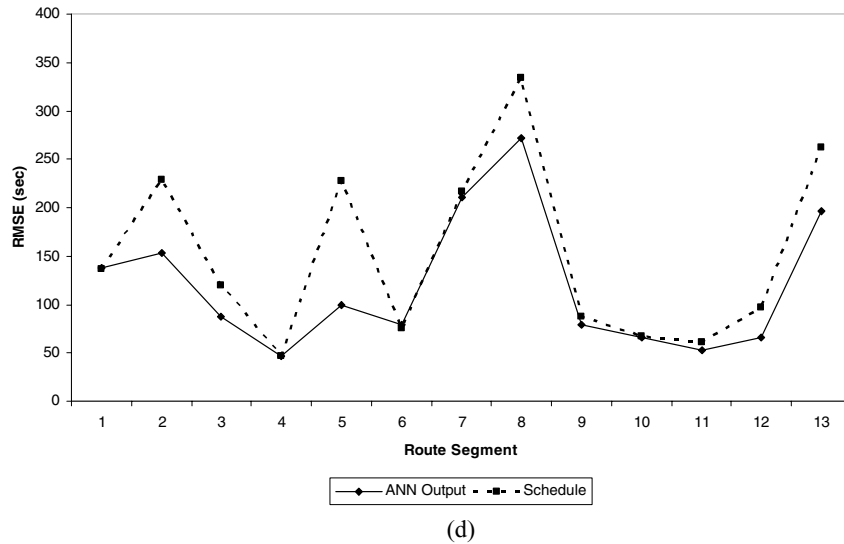
(d)

Fig. 5. *(Continued)*

corresponding ANN model. This is just as expected as it has incorporated the latest bus-arrival information into the prediction. Also, the ANN models generally give better indication of bus travel times than the timetable, except for trips under pattern PA-WM arriving at time points 7–9 from time point 1 where the differences are minimal.

One can also observe that the RMSEs of ANN model and the schedule show a trend of increase along the bus route. This can be attributed to the propagation of prediction error when the distance between the origin and destination becomes longer. Nevertheless, the dynamic

algorithm output does not show such a trend, in fact, its prediction error is generally stable with only few small-scale hikes. The incorporation of the latest bus-arrival information into the dynamic algorithm ensures the higher prediction accuracy.

Figure 7 shows the prediction errors by the dynamic algorithm, ANN model III, and schedule error for an individual trip under pattern WMIAP. This trip was made in early afternoon on a Tuesday with no precipitation. It can be observed that the dynamic algorithm provides the closest estimates of bus travel time between each OD pair (originating from time point 1) compared
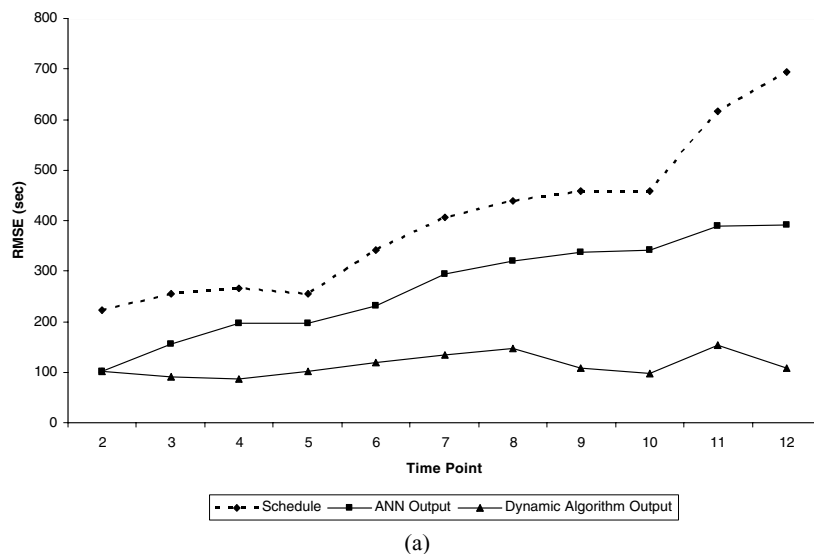

(a)

Fig. 6. Performance comparison of the dynamic algorithm, ANN model, and bus timetable (a) Pattern WM-AP (b) Pattern PA-WM (c) Pattern WMIAP (d) Pattern PAIWM.
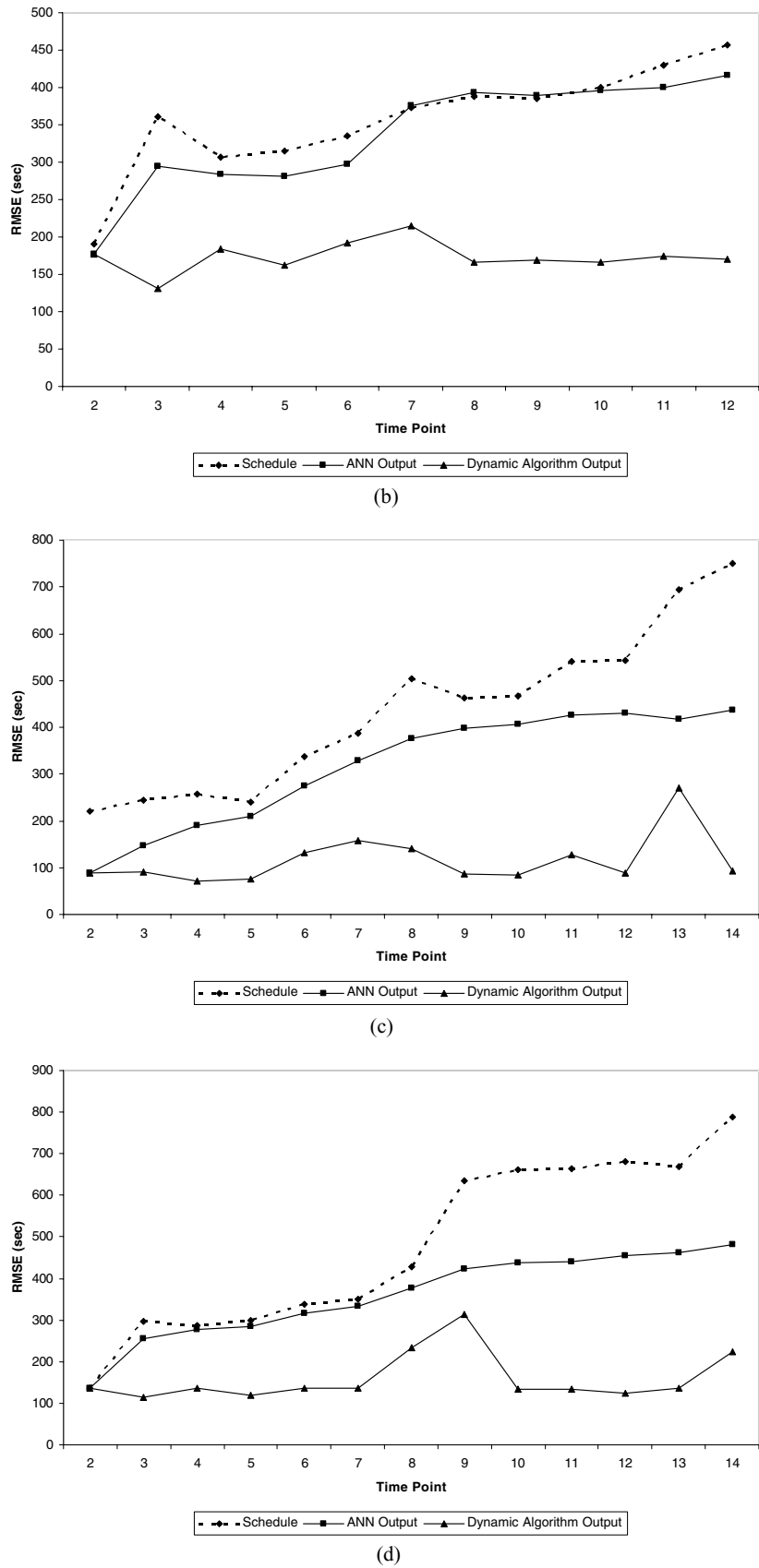
(b)

(c)

(d)

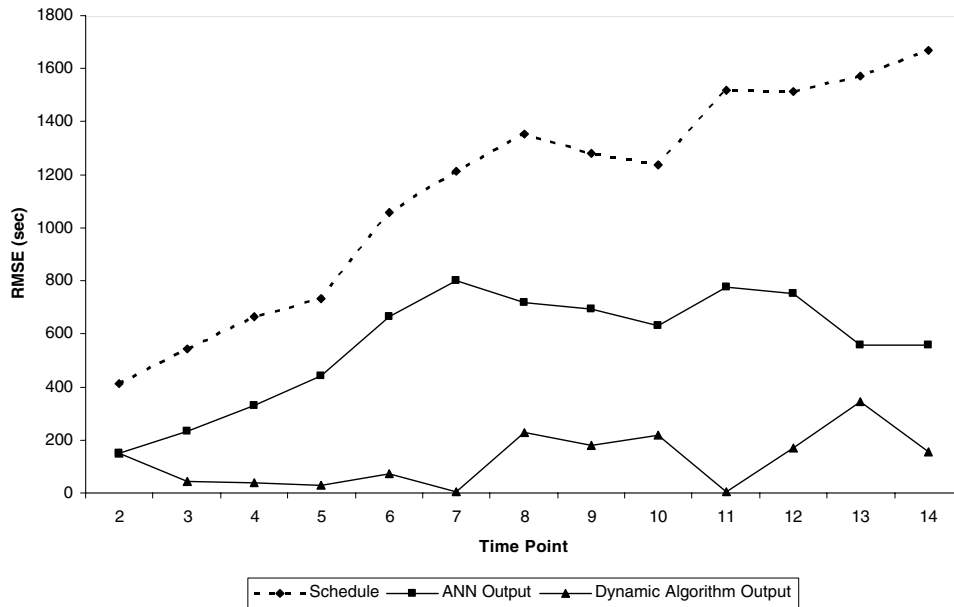**Fig. 6.** *(Continued)*

Fig. 7. Single trip prediction error comparison (schedule, ANN model, and dynamic algorithm).

to the actual travel times, and it outperforms the corresponding ANN model, which in turn provides a better indication of the travel time than the timetable.

As previously mentioned, the ANN model can be trained regularly after new data are added to the database. For the limited number of scheduled bus trips per day (or week), it is possible to make ANN simulations on a daily basis. The dynamic algorithm can then use its output as the baseline travel time estimates and make dynamic adjustment based on real-time bus location information. The calculation process of the algorithm is straightforward, and the calculation time necessary is minimal. This ensures that the algorithm can be applied in real-time.

5 CONCLUSIONS

In this study, real-world APC data are used in developing travel-time prediction methodology, which contains an artificial neural network model and a dynamic algorithm. The ANN is chosen because of its capability in mapping complicated input/output relations without requiring an explicit function form. ANN models are developed for trips under different patterns. Given trip starting time (time-of-day), day-of-week, and weather (precipitation) condition, the neural network shall generate estimated travel time on each segment along the route.

To account for the impact of unexpected delays during the trip, a dynamic algorithm is developed based on the Kalman filtering technique. It uses the most recent bus-arrival information, together with the estimated travel times generated by the ANN model, to predict arrival times at downstream time points. Tests show that ANN models generally give a better estimate of travel times than the timetable, and the dynamic algorithm always outperforms the corresponding ANN model.

With the APC data being added to the database regularly (e.g., everyday after the bus returns to the garage), the ANN model can be retrained at regular schedules to increase its reliability in performance. The dynamic algorithm can be applied online with the bus trip in progress because of its simplicity in calculation. Therefore, the methodology developed in this study can potentially be used for providing real-time bus-arrival time prediction for each time point along the route.

REFERENCES

AC Transit. Rider Information = Next Bus. Retrieved from: http://www.actransit.org/riderinfo/nextbus.wu, April 21, 2004.

Chang, G. & Su, C. (1995), Predicting intersection queue with neural network models, *Transportation Research*, **3C**(3), 175–91.

Chen, M., Yaw, J., Chien, S. & Liu, X. (2003), Using Automatic Passenger Counter Data in Bus Arrival Time Prediction, *Presented in the 82nd Annual Meeting of the Transportation Research Board*, Washington, DC.

Chien, I-Jy., Ding, Y. & Wei, C. (2002), Dynamic bus arrival time prediction with artificial neural networks, *Journal of Transportation Engineering, ASCE*, **128**(5), 429–38.

Chin, S., Hwang, H. & Pei, T. (1994), Using neural networks to synthesize origin-destination flows in a traffic circle, *Transportation Research Record*, **1457**, 134–42.

City of Fairfax. CUE Bus. Retrieved from: http://www.fairfaxva.gov/CUEBus/CUEBus.asp, April 21, 2004.

Dailey, D., Maclean, S., Cathey, F. & Wall, Z. (2001), Transit vehicle arrival prediction: Algorithm and large-scale implementation, *Journal of the Transportation Research Board*, **1771**, 46–51.

Ding, Y. & Chien, S. (2000), The prediction of bus arrival times with link-based artificial neural networks, in *Proceedings of the International Conference on Computational Intelligence & Neurosciences (CI&N)—Intelligent Transportation Systems*, 730–3, Atlantic City, New Jersey.

Dougherty, M. (1995), A review of neural networks applied to transport, *Transportation Research*, **3C**(4), 247–60.

Hua, J., & Faghri, A. (1994), Application of ANNs to IVHS, *Transportation Research Record*, **1453**, 83–90.

Kalaputapu, R. & Demetsky, M. (1995), Application of artificial neural networks and automatic vehicle location data for bus transit schedule behavior modeling, *Transportation Research Record*, **1497**, 44–52.

Lin, W. H. & Zeng, J. (1999), Experimental study of real-time bus arrival time prediction with GPS data, *Transportation Research Record*, **1666**, 101–9.

McFadden, J., Yang, W. T. & Durrans, S. R. (2001), Application of artificial neural networks to predict speeds on two-lane rural highways, *Transportation Research Record*, **1751**, 9–17.

NCDC. Retrieved from: http://lwf.ncdc.noaa.gov/oa/climate/stationlocator.html, January 10, 2003.

NJ Transit. Bus Route 62 Timetable. Retrieved from: http://www.njtransit.com/pdf/bus/T0062.pdf, January 6, 2004.

Principe, J., Euliano, N. & Lefebvre, W. (2000), *Neural and Adaptive Systems: Fundamentals through Simulations*, Wiley, NY.

San Francisco Muni. Routes and Schedules. Retrieved from: http://www.sfmuni.com/routes/indxrout.htm, April 21, 2004.

Shalaby, A. & Farhan, A. (2003), Bus travel time prediction model for dynamic operations control and passenger information systems, CD-ROM, *The 82nd Annual Meeting of the Transportation Research Board*, Washington, DC.

Smith, B. L. & Demetsky, M. J. (1995), Short-term traffic flow prediction: Neural network approach, *Transportation Research Record*, **1453**, 98–104.

Town of Vail. Transportation and Parking. Retrieved from: http://www.ci.vail.co.us/subpage.asp?page_id=162, April 21, 2004.

TransLink Research Center. Journey time prediction algorithms for transit vehicles. Retrieved from: http://translink.tamu.edu/docs/Research/BusArrivalPrediction/PredictionAlgorithms.cfm, January 6, 2004.

TriMet. Next arrivals. Retrieved from: http://www.trimet.org, January 6, 2004.

University of Washington. Smart Trek: Transit Watch. Retrieved from: http://www.its.washington.edu/transitwatch, January 6, 2004.

Vapnik, V. (1995), *The Nature of Statistical Learning Theory*, Springer Verlag, NY.

Zhang, H., Ritchie, S. & Lo, Z. (1997), Macroscopic modeling of freeway traffic using an artificial neural network, *Transportation Research Record*, **1588**, 110–19.