

# Hybrid model for prediction of bus arrival times at next station

Bin Yu, Zhong-Zhen Yang\*, Kang Chen and Bo Yu

*Transportation Management College, Dalian Maritime University, Dalian, 116026, China*

## SUMMARY

Effective prediction of bus arrival times is important to advanced traveler information systems (ATIS). Here a hybrid model, based on support vector machine (SVM) and Kalman filtering technique, is presented to predict bus arrival times. In the model, the SVM model predicts the baseline travel times on the basis of historical trips occurring data at given time-of-day, weather conditions, route segment, the travel times on the current segment, and the latest travel times on the predicted segment; the Kalman filtering-based dynamic algorithm uses the latest bus arrival information, together with estimated baseline travel times, to predict arrival times at the next point. The predicted bus arrival times are examined by data of bus no. 7 in a satellite town of Dalian in China. Results show that the hybrid model proposed in this paper is feasible and applicable in bus arrival time forecasting area, and generally provides better performance than artificial neural network (ANN)-based methods. Copyright © 2010 John Wiley & Sons, Ltd.

KEY WORDS: transportation; arrival time; hybrid model; prediction

## 1. INTRODUCTION

There is a potential need for advanced public transportation systems (APTS) and advanced traveler information systems (ATIS) which can provide accurate bus arrival information to transit travelers and transit agencies with the rapid development of new information technology such as the use of automatic vehicle location (AVL), identification (AVI) systems, and automatic passenger counters (APC), etc. Generally, transit vehicle arrives stochastically at stations/stops in urban networks, because of the variable travel times on links and the variable dwell times at stops, etc. Thus, the deployment of bus arrival time prediction model is a challenging task. The travel time prediction in traffic field has been studied extensively in the past decade. Smith and Demetsky [1] proposed a neural network-based model to predict traffic volumes in intelligent vehicle highway systems. Their results showed that the back-propagation model was superior to the traditional approaches based on an historical, data-based algorithm and a time-series model. Hellinga and Fu [2] presented a method based on sampling techniques to reduce the effect of this bias. Their results showed that the method could reduce the mean travel time error. Chien *et al.* [3] presented a link-based artificial neural network (ANN) and a stop-based ANN to predict bus arrival times in real-time. Their results showed that the link-based ANN model was suitable for the routes with fewer intersections between stops. Furthermore, they suggested that stop-based ANN model might be suitable for the routes with fewer intersections between stops. Chen *et al.* [4] presented a dynamic model, which consisted of an ANN model and a Kalman filter technique, to predict bus arrival times. Their results showed the method was suitable for bus arrival time prediction. Cathey and Dailey [5] presented a general prescription on the prediction of transit vehicle arrival/departure, which can describe the steps in any prediction scheme. The prescription consisted of a tracker, a filter and a predictor, and corresponding algorithms were also proposed. Based on the method proposed by Cathey and Dailey [5], Dailey and Cathey [6] predicted vehicle arrival/departure for the Portland Metro area. In their approach, the difference between the report times and the interpolated schedule times was estimated. The classic paradigm of “generate and test”, which is a

\*Correspondence to: Zhong-Zhen Yang, Transportation Management College, Dalian Maritime University, Dalian, 116026, China. E-mail: Yangzhongzhen@263.net

simple method that repeats until it gets a complete solution, was also used to determine trip and the distance along the underlying route. They argued that the algorithm could improve tracking efficiency, eliminate ambiguities, and increase probability of update correctness. Li and McDonald [7] presented a method based on probe vehicles to estimate travel times. They compared the estimated travel times with actual mean travel times. The results showed that the proposed method was not suitable for extremely slow vehicle conditions. Dailey *et al.* [8] presented an algorithm to predict bus arrival times based on times and location pairs from an automated vehicle location system. Lin and Zeng [9] discussed the link-node representation based on global positioning system (GPS) data and analyzed GPS data. A set of algorithms were discussed and compared. Lin and Bertini [10] presented a component model for bus arrival time prediction based on Markov chain, which can reflect bus operators to adjust the bus speeds to pursue schedule recovery. Chien *et al.* [11] developed a probabilistic model for bus arrival times aiming to minimize total wait times incurred by pre-trip passengers. They suggested, the probabilistic model could be applied to *any arrival distributions of buses and passengers*.

Existing studies for bus arrival (travel) time prediction have focused on time series, Kalman filtering technique and ANN. The time series methods are difficult to model nonlinear relations and usually have a short time lag while applied in real time system. On the other hand, they are usually easy and fast in use. Kalman filter technique is an effective online prediction tool, while it is not able to take historical data into calculation or consideration. Generally, in transit service, each bus serves along fixed-route and fixed-stop day by day. Thus, there are some potential rules in history trip data. The potential rules might be useful for transit arrival time estimation. Since ANN or support vector machine (SVM) can be adjusted to map the input–output relationship for the nonlinear system, the methods are suitable for inducting some potential rules from time series data.[12].

SVM [13,14]) can utilize linear model to implement nonlinear class boundaries through some nonlinear mapping from one-dimensional input vectors into high-dimensional feature space that is disposed using a multidimensional data structure. Since training SVM is equivalent to solving a linearly constrained quadratic programming problem, SVM shows the strong resistance to the over-fitting problem and the high generalization performance compared with other networks' training. It has successfully been applied as a solution to some classic problems, such as incident detection [15], traffic-pattern recognition [16], head recognition [17], and travel time prediction [18–20]). Furthermore, the study presented by Yu *et al.* [18] suggested that SVM model is suitable for bus arrival time prediction. These successful results of time-varying applications with SVM prediction motivate our research to use SVM for modeling bus arrival times. However, Chen *et al.* [4] pointed out that the history data-based models had difficulty in dealing with dynamic traffic conditions. Referring to the dynamic algorithm proposed by Chen *et al.* [4], a Kalman filter-based method is applied to adjust the outputs from SVM model. Thus, a hybrid prediction model is presented in this paper, which consists of a SVM model and a Kalman filter-based method.

The structure of this paper is organized as follows: Section 2 provides the structure of the hybrid model for predicting bus arrival time; Section 3 contains results and analysis including performance evaluation of the methodology; and lastly, the conclusions are presented in Section 4.

## 2. MODEL DEVELOPMENTS

The hybrid prediction model consists of two major steps: one is the SVM model for predicting baseline travel times between points; the other is the Kalman filtering-based dynamic algorithm to adjust the predicted baseline times with the latest arrival information. Figure 1 depicts the framework of the hybrid model. The two components of this hybrid model, the SVM model, and the dynamic algorithm, are discussed, respectively.

### 2.1. Support vector machines for regression estimation

Given a set of data, points,  $(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)$  ( $x_i$  is the input vector,  $x_i \in X \subseteq R^n$ ;  $y_i$  is the desired value,  $y_i \in Y \subseteq R$ ,  $l$  is the number of training samples) are randomly and independently

generated from an unknown function. SVM approximates the function using the following form [21]:

$$f(x) = \omega \bullet \phi(x) + b \quad (1)$$

where  $\phi(x)$  represents the high-dimensional feature spaces which are nonlinearly mapped from the input space  $x$ . The coefficients  $\omega$  and  $b$  are estimated by minimizing the regularized risk function:

$$\frac{1}{2} \|\omega\|^2 + C \frac{1}{l} \sum_{i=1}^l L_{\varepsilon}(y_i, f(x_i)) \quad (2)$$

The first term  $\|\omega\|^2$  is called the regularized term. Minimizing  $\|\omega\|^2$  will make a function as flat as possible, which may play an important role in controlling the function capacity. The second term  $\frac{1}{l} \sum_{i=1}^l L_{\varepsilon}(y_i, f(x_i))$  is the empirical error measured by the  $\varepsilon$ -insensitive loss function, which is defined as below [14]:

$$L_{\varepsilon}(y_i, f(x_i)) = \begin{cases} |y_i - f(x_i)| - \varepsilon, & |y_i - f(x_i)| \geq \varepsilon \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

This defines  $\varepsilon$  tube (Figure 2) so that if the predicted value is within the tube the loss is zero, while the predicted point exceeds the tube, the loss is magnitude of the difference between the predicted value and radius  $\varepsilon$  of the tube.  $C$  is called the regularization constant. Increasing the value of  $C$  will result in the relative importance of the empirical risk with respect to the regularization term to grow.  $\varepsilon$  is called the tube size and it is equivalent to the approximation accuracy placed on the training data points. Both  $C$  and  $\varepsilon$  are user-prescribed parameters. In the case of infeasibility, one can introduce slack variables  $\xi_i, \xi_i^*$  to cope with infeasible constraints of the optimization problem.

The minimization of Equation (2) is a standard problem in optimization theory. This can be solved by applying Lagrangian theory and the weight vector ( $\omega$ ) equals the linear combination of the training data:

$$\omega - \sum_{i=1}^l (a_i - a_i^*) x_i = 0 \quad (4)$$

thus

$$f(x) = \sum_{i=1}^l (a_i - a_i^*) \phi(x_i) \bullet \phi(x) + b \quad (5)$$

By introducing kernel function  $K(x_i, x)$  the Equation (5) can be rewritten as follows:

$$f(x) = \sum_{i=1}^l (a_i - a_i^*) K(x_i, x) + b \quad (6)$$

The value of  $K(x_i, x_j)$  is equal to the inner product of two vectors,  $x_i$  and  $x_j$  in the feature space  $\phi(x_i)$  and  $\phi(x_j)$ , that is,  $K(x_i, x_j) = \phi(x_i) \bullet \phi(x_j)$ . By use of kernels, all necessary computations can be performed directly in input space, without having to compute the map  $\phi(x)$ . There are some popular

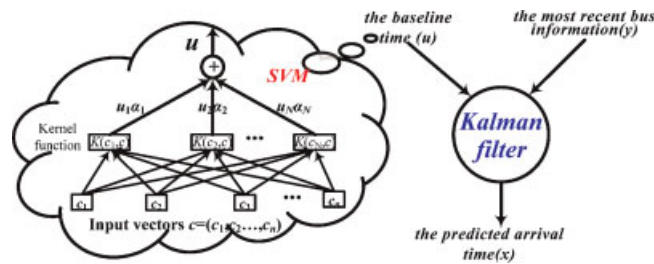


Figure 1. The framework of the hybrid prediction model.

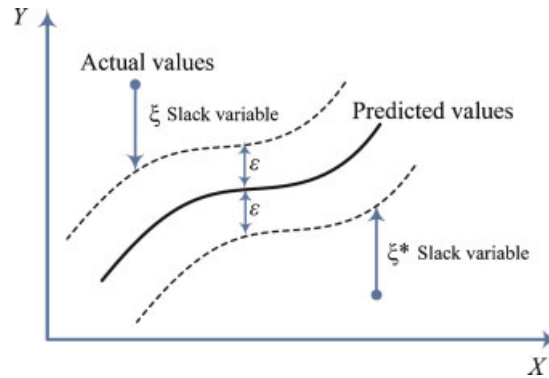


Figure 2. The parameters for the support vector regression.

kernel functions, the linear kernel  $K(x_i, x_j) = x_i \bullet x_j$ , polynomial kernel  $K(x_i, x_j) = (x_i \bullet x_j + 1)^d$ , and the radial-basis function (RBF) kernel  $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$ , etc, where  $d$  and  $\gamma$  are the kernel parameters. Using different kernel functions, one can construct different learning machines with arbitrary types of decision surfaces. For more details see [13,14,21].

Among the variables that may contribute to the variation of bus running, five input variables and an output variable are used in the SVM model. Let  $\mathbf{c}$  denote the input vector, which consists of the five variables, time-of-day ( $c_1$ ), weather condition ( $c_2$ ), predicted segment ( $c_3$ ), the travel times of current segment ( $c_4$ ), and the latest travel times of next segment ( $c_5$ ) [18] and  $\mathbf{u}$  denote output vector, the bus travel times on the route segment between two adjacent points. The time-of-day variable is self-explanatory, which can take two possible descriptive values, peak time (6:30–7:30 AM) or off-peak time (10:00–11:00 AM). For weather conditions, only sunny days and rainy days are considered. The route segment variable identifies the section between the current stop and the next stop at which the arrival times are to be predicted. The  $c_4$  denotes the travel times of the current bus on the current route segment where the current bus has just finished running. The variable is used to show the current operating conditions of the bus. The  $c_5$  denotes the travel times on the predicted route segment of the preceding bus that has last finished its running on the predicted segment. The variable is expected to estimate the traffic conditions of the next segment. The latest travel times on the predicted segment will be updated after a bus finishes its running on the predicted segment. Assume that there are the stops numbered from the origin terminal through the destination terminal, as shown in Figure 3. While a vehicle reaches the stop  $k$ , the input variables of the SVM model for the arrival times at the stop  $k+1$  is shown in Figure 3.

## 2.2. Kalman filtering-based dynamic algorithm

The SVM model discussed previously is based on historical data since a large amount of times for online training will be involved. Even if the SVM model is to be retrained regularly, e.g. daily or weekly, the SVM model still does not dynamically adjust the prediction output. Thus, the travel times predicted by the SVM model are served as a baseline of the travel times on the predicted segment. The Kalman filter is a minimum mean-square error estimator. It only requires the estimated state from the previous time step and the current measurement to compute the estimate for the current state. This means that no history of observations and/or estimates is required. Thus, a Kalman filtering technique-based method is developed to implement the online adjustment of the predicted arrival times based on the latest information.

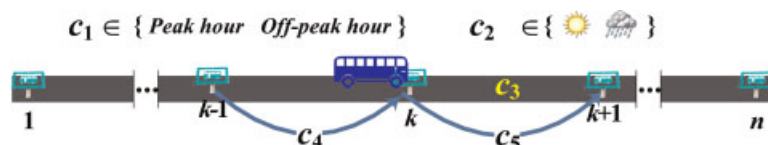


Figure 3. Five input variables in the proposed SVM model.

Let  $x_k$  denote the state at the current time step ( $k$ ) and  $y_k$  denote the measured state at the current time step ( $k$ ). The Kalman filter technique firstly uses the state estimate from the previous time step ( $k-1$ ) to produce an estimate of the state at the current time step ( $k$ ). Then, the measurement information at the current time step ( $k$ ) is used to refine this prediction to arrive at a new and more accurate state estimate. The state equation and the measurement equation are as follows:

$$x_k = \Phi_{k-1}x_{k-1} + B_{k-1}u_{k-1} + w_{k-1} \quad (7)$$

$$y_k = H_k x_k + v_k \quad (8)$$

Where,  $u_{k-1}$  is system input;  $\Phi_{k-1}$  relates the state at the previous time step ( $k-2$ ) to the state at the current step ( $k-1$ ).  $B_{k-1}$  relates the control input  $u_{k-1}$  to the state  $x_{k-1}$ ;  $H_k$  relates the state to the measurement  $y_k$ . The random variables  $w_{k-1}$  and  $v_k$  represent the process and measurement noise, respectively, which are assumed to be independent of each other, and with normal probability distributions.

We define  $\hat{x}_k^-$  to be a prior state estimate at the time step  $k$ , given knowledge of the process prior to the time step  $k$ , and  $\hat{x}_k$  to be a posterior state estimate at the time step  $k$ , given measurement  $y_k$ .

$$\hat{x}_k^- = \Phi_{k-1}\hat{x}_{k-1} + B_{k-1}u_{k-1} \quad (9)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - H_k\hat{x}_k^-) \quad (10)$$

The Kalman gain  $K_k$  is to reflect the stochastic nature of the process and measurement, which is computed by an optimal linear estimator that minimizes the squared error on the expected value of the state estimate  $\hat{x}_k$ . The difference  $(y_k - H_k\hat{x}_k^-)$  is called the measurement innovation or the residual. The residual reflects the discrepancy between the predicted measurement  $H_k\hat{x}_k^-$  and the actual measurement  $y_k$ .

Then, a prior and a posterior estimate errors can be defined as

$$e_k^- = x_k - \hat{x}_k^- \quad (11)$$

and

$$e_k = x_k - \hat{x}_k \quad (12)$$

Correspondingly, the prior estimate error covariance is then

$$P_k^- = E[e_k^- e_k^{-T}] = \Phi_{k-1}P_{k-1}\Phi_{k-1}^T + Q_{k-1} \quad (13)$$

and the posterior estimate error covariance is

$$P_k = E[e_k e_k^T] = P_k^- - P_k^- H_k^T K_k^T - K_k H_k P_k^- + K_k (H_k P_k^- H_k^T + R_k) K_k^T \quad (14)$$

where,  $Q_k$  and  $R_k$  are covariance matrices of process noise  $w_k$  and measurement noise  $v_k$ .

The gain matrix  $K_k$  is chosen to minimize the posterior error covariance. One form of the resulting  $K$  that minimizes Equation (14) is given by

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \quad (15)$$

From Equation (15), as the measurement error covariance  $R_k$  approaches zero, the gain  $K$  weights the residual more heavily. On the other hand, as the prior estimate error covariance  $P_k^-$  approaches zero, the gain  $K$  weights the residual less heavily. Another way of thinking about the weighting by  $K$  is that as the measurement error covariance  $R_k$  approaches zero, the actual measurement  $y_k$  is “trusted” more and more, while the predicted measurement  $H_k\hat{x}_k^-$  is trusted less and less. On the other hand, as a prior estimate error covariance  $P_k^-$  approaches zero the actual measurement  $y_k$  is trusted less and less, while the predicted measurement  $H_k\hat{x}_k^-$  is trusted more and more. For more details see Reference [22].

Let  $x_k$ ,  $y_k$  denote the estimated travel times and the observed travel times from origin to point  $k$ , respectively. Let  $u_k$  denote the baseline travel times on the  $k$ th segment which are predicted by the

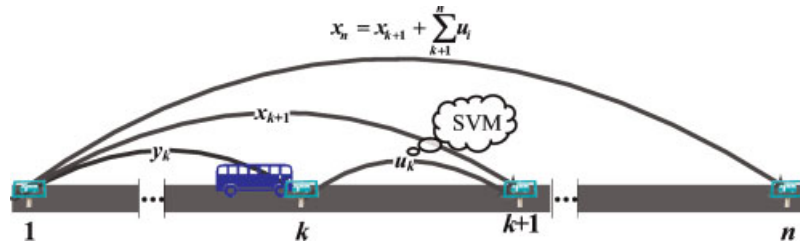


Figure 4. The process of the proposed prediction model.

above SVM. Since  $x_k$ ,  $u_k$ , and  $y_k$  denote travel times and one-dimensional,  $\Phi_{k-1} = (1)$ ,  $B_{k-1} = (1)$ , and  $H_k = (1)$ . The Kalman filtering addresses the bus arrival time prediction by estimating the state  $x_k$  of a discrete time controlled process that is governed by the following linear stochastic difference equation:

$$x_k = x_{k-1} + u_{k-1} + w_{k-1} \quad (16)$$

with a measurement  $y_k$  that is

$$y_k = x_k + v_k \quad (17)$$

### 2.3. Hybrid model for bus arrival time prediction

Figure 4 depicts the process of the hybrid model, which is described below. Before application, the SVM model has to choose support vectors from historical data and the well-trained SVM model can estimate the baseline travel times on each segment  $u_k$ . For a prediction,  $x_1$  is set to 0 based on its definition as a bus is at the origin point and the arrival times  $\hat{x}_2^-$  on the first route segment are predicted according to the baseline times  $u_1$ . When the bus reaches the second point, the actual travel times  $y_2$  are recorded, then  $K_2$  is calculated and the travel times  $\hat{x}_3^-$  on the second route segment is predicted with the baseline times  $u_2$ , and posterior state  $\hat{x}_2$  is also estimated. Then the estimated travel times from the current point to all downstream points are updated by the filter based on the information. The output at each iteration is a set of the predicted arrival times at all downstream points. As the bus proceeds along its route, the prediction will be updated whenever the latest arrival information is obtained. The process repeats till the bus reaches the terminal.

## 3. NUMERICAL TEST

### 3.1. Data collection and processing

The hybrid model for bus arrival time prediction has been tested by the data of bus no. 7 in the satellite town of Dalian, China. The bus no. 7 goes from the center to the suburb in the city, which is 14.5 km and consists of total of 20 stops per direction. The bus no. 7 is highly congested in the morning and afternoon peak periods and only the eastbound direction is studied. The travel times are about 1 hour during peak periods and are about 45 minutes during off-peak periods. Totally, there are 12 points located along the bus route as shown in Figure 5.

#### 3.1.1. Data collection

Bus travel times are variational in different conditions (time-of-day, route segment or weather). The time-of-day variable can take two values, peak period (P) or off-peak period (O). The time-of-day variable takes the descriptive symbol of the time period within which the trip is initiated. For weather conditions, the variable can take two possible values, 1 or 0, corresponding to *sunny day* and *rainy day*, respectively. The route segment variable can be 1 through 11.

To gain the data of travel times and weather, we conducted on-board survey of all trips on the bus no. 7 from September to October, 2005. The collected data consists of the weather and the arrival times at the 12 time points in each individual trip at morning peak period and morning off-peak period during weekdays. According as time-of-day and weather, the data are divided into four patterns, which correspond to peak period and sunny day (SP), off-peak period and sunny day (SO), peak period and



rainy day (RP), off-peak period and rainy day (RO), respectively. There are 290 valid trips within 1 month, among which 180 trips are of SP, 70 trips are of SO, 22 trips are of RP and 18 trips are of RO. Since the bus route is divided into 11 segments by the 12 points, there are 3190 travel times in total. To avoid attributes in greater numeric ranges dominating those in smaller numeric ranges in training process, the data should be scaled [23]. In the modeling process, the data sets are scaled to the range between 0 and 1 as follows:

$$x'_i = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)} \quad (18)$$

$$y'_i = \frac{y_i - \min(y_i)}{\max(y_i) - \min(y_i)} \quad (19)$$

where,  $x'_i, y'_i$  are the scaled values.

### 3.1.2. Model identification

Using the kernel function, the data can be mapped implicitly into a feature space. RBF kernel is selected in this study. There are three parameters in SVM model while using RBF kernels:  $C$ ,  $\varepsilon$ , and  $\gamma$ . There are several methods developed to identify the best  $C$ ,  $\varepsilon$ , and  $\gamma$ , among which grid-search is frequently used and most reliable but some complex ones [24,25]. Here, the three parameters were selected as  $(2^{-2}, 2^{-5}, 0.52)$ .

## 4. RESULTS

The basic strategy used for the training process consists in utilizing three data sub-sets, respectively for training, cross-validation and testing. At first, about 10% samples of the data are set aside as testing data. The remaining data, which consists of 162 trips of SP, 63 trips of SO, 20 trips of RP and 16 trips of RO, are randomly assigned into two groups, the first with 120 trips of SP, 45 trips of SO, 14 trips of RP, and 10 trips of RO for training and the second with the other data for cross validation.

Then, the variations between the SVM output and actual travel times, and the variations between the schedule and actual travel times are compared. Accuracy is evaluated by computing the root mean squared error (RMSE) of the predicted arrival time on each route segment. Figure 6 shows the comparison of RMSEs of the SVM output and the scheduled travel times for the four types. It can be observed that in all cases, the SVM RMSEs for all route segments are smaller than those of the scheduled travel times. This indicates that, compared to the timetable, the SVM model can provide better results.

Then, to evaluate the performance of the proposed hybrid model, an ANN and Kalman filtering-based model, named ANN + Kalman model in this paper, is constructed using the same data sets to the proposed hybrid model. In the ANN + Kalman model, a standard three-layer ANN is constructed. The

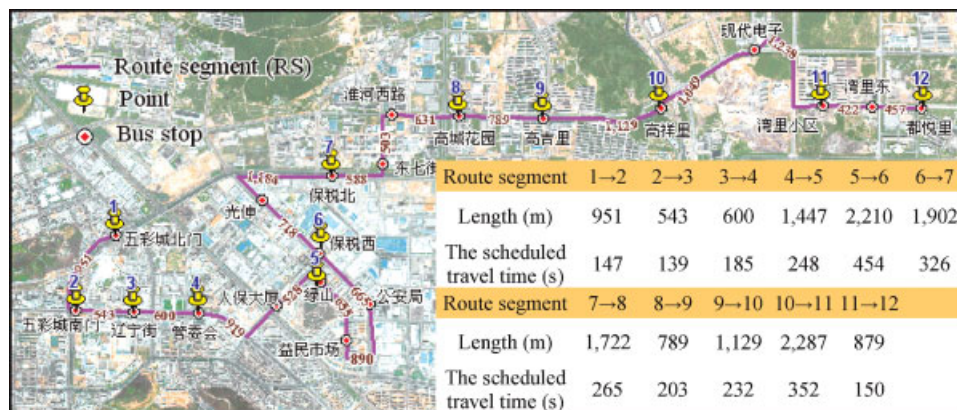


Figure 5. Configuration of transit route number 7.

scaled conjugate gradient algorithm [26] is employed for training, and the hidden neurons are optimized by trials and errors. The final ANN architecture consists of five same input features, three hidden neurons, and an output neuron. Sigmoid transfer function is used in the hidden and output neurons. For the ANN presented in this paper, training is performed for about 450 epochs, until the mean-squared error falls below 0.05. Also, the Kalman filtering in ANN + Kalman models is as the one in the proposed hybrid model. In order to make the same basis of comparison, the same training and verification sets are used for both models. The performances of the hybrid model (SVM + Kalman), ANN + Kalman, and SVM on the four patterns are presented in Figure 7.

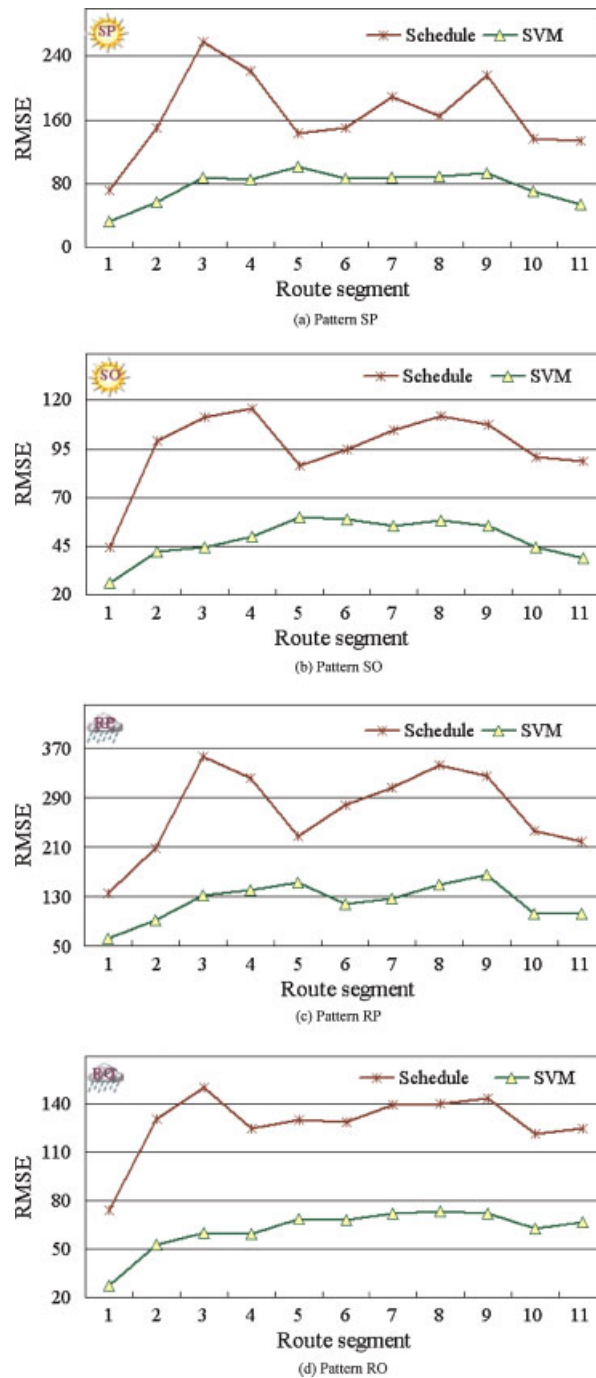


Figure 6. Comparison of prediction errors for SVM model and schedule.



It can be observed that the SVM + Kalman and ANN + Kalman models, with lower RMSEs, always perform better than the corresponding SVM models. This is just as expected as the incorporation of the latest bus arrival information into the dynamic algorithm can ensure the higher prediction accuracy. Furthermore, it is demonstrated that the SVM + Kalman models exhibit some advantages over the ANN + Kalman model. On the four patterns, the SVM + Kalman models outperform the ANN + Kalman models by 10.04, 7.45, 11.37, and 11.10% for the verification data, respectively. This can be attributed to the fact that SVM implements the structural risk minimization (SRM) principle, which seeks to minimize an upper bound of the generalization error. SRM has been shown to be superior to

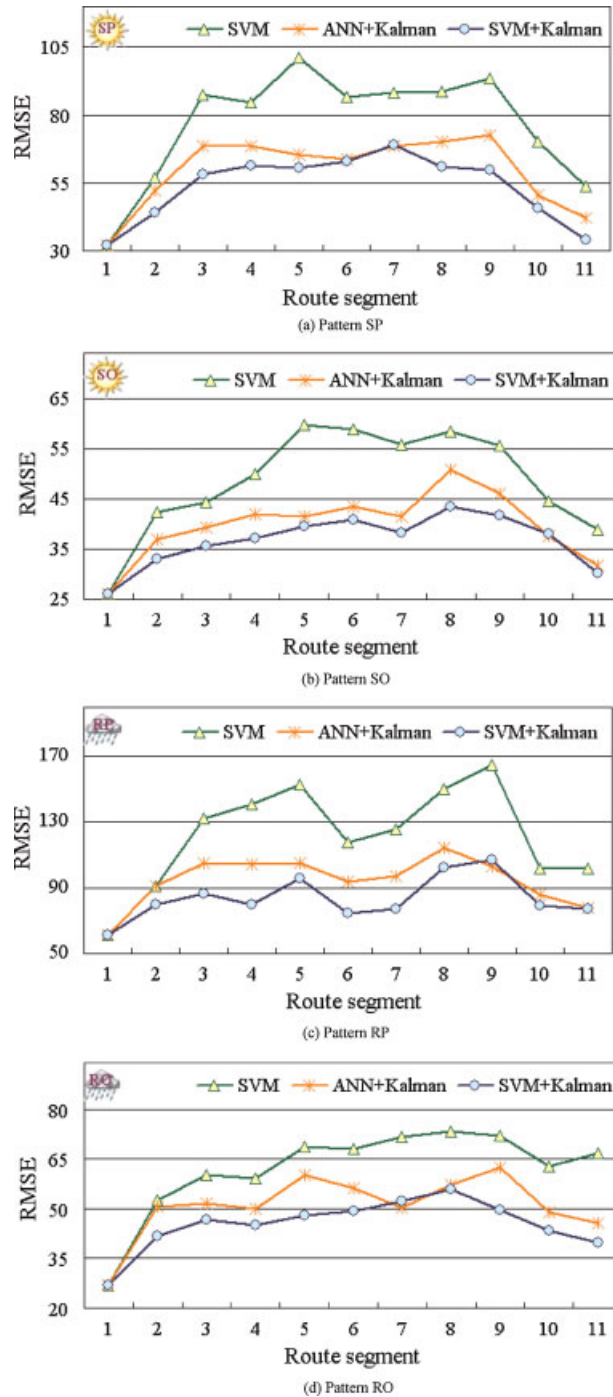


Figure 7. Comparison of prediction errors for three methods.

traditional empirical risk minimization (ERM) principle implemented by conventional neural networks, which seeks to minimize the training error. The solution of SVM may be globally optimal, while ANN may tend to fall into a local optimal solution. At the same time over-fitting is unlikely to occur with the SVM, if the parameters are properly selected. So SVM seems to be a powerful alternative for bus arrival time prediction.

In addition, we can see that the RMSEs of the prediction results for the 5th route segment are highest. This is because that the route segment is located in the central business district (CBD) of Dalian satellite town. CBD traffic slows to a near-standstill in peak hours, especially on rainy days, which causes transit vehicle arrivals to deviate from the schedule and the vehicle arrival times not to be accurately predicted. In addition, except for the 5th route segment, it can be observed that the RMSEs of the prediction results for the 8th and 9th route segments are also large. It is mainly attributed that the passenger flows are large and the roads are narrow in the area. This may cause a variety of external influences, which affects travel times of transit vehicle on the links or dwell times at the stops and increases the RMSEs of the prediction results for the route segments.

## 5. CONCLUSIONS

One of the major stochastic characteristics in transit operations is that bus arrivals always deviate from the schedule, which could discourage passengers from using the transit system. This study has developed a hybrid model which can dynamically predict bus arrival times while considering the stochastic traffic. The hybrid model consists of two major elements: an SVM model and a Kalman filtering-based dynamic algorithm. The SVM can map complicated input/output relations without requiring an explicit functional form and also provide the baseline times on segment based on historical trip data. To account for the impact of unexpected delays during running, the dynamic algorithm based on Kalman filtering uses the latest bus arrival information, together with the estimated baseline travel times generated by the SVM model, to predict arrival times at downstream points. The tests show that the SVM-based model generally outperforms the corresponding ANN-based model.

Furthermore, GPS, which is a part of Intelligent Transportation Systems (ITS), will be adopted by many transit agencies. This can allow transit agencies to track their transit vehicles in real-time [27,28]. Moreover, real-time data will be available and the accuracy of bus arrival time prediction can be further improved. The dynamic algorithm based on Kalman filter can be applied online with the bus trip in progress because of its simplicity in calculation. The future focus of this research would be to develop a new training method for online SVM processing large-size problems, and to extend the dynamic algorithm to a multi-dimensional Kalman filter.

## 6. ABBREVIATIONS

The following symbols are used in this paper:

<i>ATIS</i>	= advanced traveler information systems;
<i>SVM</i>	= support vector machine;
<i>APTS</i>	= advanced public transportation systems;
<i>AVL</i>	= automatic vehicle location;
<i>AVI</i>	= identification systems;
<i>APC</i>	= automatic passenger counters;
<i>ANN</i>	= artificial neural network;
<i>GPS</i>	= global positioning system;
<i>RBF</i>	= the radial-basis function;
<i>SP</i>	= peak period and sunny day;
<i>SO</i>	= off-peak period and sunny day;
<i>RP</i>	= peak period and rainy day;
<i>RO</i>	= off-peak period and rainy day;
<i>RMSE</i>	= root mean squared error;
<i>SRM</i>	= structural risk minimization;

ERM = empirical risk minimization;  
 CBD = central business district;  
 ITS = Intelligent Transportation Systems;

## 7. SYMBOLS

$C$  = the regularization constant;  
 $\varepsilon$  = the tube size;  
 $\gamma$  = the kernel parameter;  
 $c_1$  = time-of-day;  
 $c_2$  = weather condition;  
 $c_3$  = predicted segment;  
 $c_4$  = the travel times of the current bus on the current route segment;  
 $c_5$  = the travel times on the predicted route segment of the preceding bus;  
 $u_{k-1}$  = system input in Kalman filter.  
 $K_k$  = Kalman gain.

## ACKNOWLEDGEMENTS

This research is financed by the National Science Foundation for Post-doctoral Scientists of China 20080440168, the Doctoral Program Foundation for Young Scholar of Institutions of Higher Education of China through project 20070151013 and the Special Fund for Basic Scientific Research of Central Colleges, Dalian maritime university (2009QN094).

## REFERENCES

1. Smith BL, Demetsky MJ. Short-term traffic flow prediction: neural network approach. *Transportation Research Record* 1995; **1453**:98–104.
2. Hellinga BR, Fu LP. Reducing bias in probe-based arterial link travel time estimates. *Transportation Research Part C* 2002; **10**(4):257–273.
3. Chien SIJ, Ding Y, Wei C. Dynamic bus arrival time prediction with artificial neural networks. *Journal of Transportation Engineering, ASCE*, 2002; **128**(5):429–438.
4. Chen M, Liu XB, Xia JX, Chien SI. A dynamic bus-arrival time prediction model based on APC data. *Computer-Aided Civil and Infrastructure Engineering* 2004; **19**(5):364–376.
5. Cathey FW, Dailey DJ. A prescription for transit arrival/departure prediction using automatic vehicle location data. *Transportation Research Part C* 2003; **11**(3–4):241–264.
6. Dailey DJ, Cathey FW. Arrival/departure prediction under adverse conditions using the tri-met AVL system, Volume II, *Transportation Northwest Regional Center–TransNow(USDOT), Final Technical Report, TNW 2001-10.2*, 2002.
7. Li Y, McDonald M. Link travel time estimation using single GPS equipped probe vehicle. *Proceedings of The IEEE 5th International Conference on Intelligent Transportation Systems*, 2002; 932–937.
8. Dailey DJ, Wall ZR, Maclean SD, Cathey FW. An algorithm and implementation to predict the arrival of transit vehicles, *Proceedings of 3rd IEEE Intelligent Transportation Systems Conference (ITSC-2000) DEARBORN, MI, OCT 01-03, 2000*; 161–166.
9. Lin WH, Zeng J. An experimental study on real-time bus arrival time prediction with GPS data. *Transportation Research Record* 1999; **1667**:101–109.
10. Lin WH, Bertini RL. Modeling schedule recovery processes in transit operations for bus arrival time prediction. *Journal of Advanced Transportation* 2004; **38**(3):347–365.
11. Chien SIJ, Daripally SK, Kim K. Development of a probabilistic model to optimize disseminated real-time bus arrival information for pre-trip passengers. *Journal of Advanced Transportation* 2007; **41**(2):195–215.
12. Thissen U, Brakel RV, Weijer APd, Melssen WJ, Buydens LMC. Using support vector machines for time series prediction. *Chemometrics and Intelligent Laboratory Systems* 2003; **69**(1–2):35–49.
13. Vapnik VN. An overview of statistical learning theory. *IEEE Transactions on Neural Networks* 1999; **10**(5):988–999.
14. Vapnik VN. *The Nature of Statistical Learning Theory*, Springer: New York, 2000.
15. Yuan F, Cheu RL. Incident detection using support vector machines. *Transportation Research Part C* 2003; **11**(3–4):309–328.
16. Ren JT, Ou XL, Zhang Y, Hu DC. Research on network level traffic pattern recognition. *Proceedings of The IEEE 5th International Conference on Intelligent Transportation Systems*, 2002; 500–504.

17. Reyna R, Giralt A, Esteve D. Head detection inside vehicles with a modified svm for safer airbags. *Proceedings of The IEEE 4th International Conference on Intelligent Transportation Systems*, 2001; 500–504.
18. Yu B, Yang ZZ, Yao BZ. Bus arrival time prediction using support vector machines. *Journal of Intelligent Transportation Systems* 2006; **10**(4):151–158.
19. Wu CH, Ho JM, Lee DT. Travel-time prediction with support vector regression. *IEEE Transactions on Intelligent Transportation Systems* 2004; **5**(4):276–281.
20. Yu B, Yang Z-Z. A dynamic holding strategy in public transit systems with real-time information. *Applied Intelligence* 2009; **31**(1):69–80.
21. Cao LJ, Tay FEH. Support vector machine with adaptive parameters in financial time series forecasting. *IEEE Transactions on Neural Networks* 2003; **14**(6):1506–1518.
22. Grewal MS, Andrews AP. *Kalman Filtering: Theory and Practice Using MATLAB*; 2nd edn, John Wiley & Sons: New York, 2001.
23. Sarle WS. Neural Network FAQ. Periodic Posting to the Usenet News Group Comp.ai.neural-nets. Available at: <ftp://ftp.sas.com/pub/neural/FAQ.html> 1997.
24. Hsu CW, Chang CC, Lin CJ. A practical guide to support vector classification, *Technical Report*, Department of Computer Science and Information Engineering, National Taiwan University, 2003.
25. Dong B, Cao C, Lee SE. Applying support vector machines to predict building energy consumption in tropical region. *Energy and Buildings* 2005; **37**(5):545–553.
26. Moller MF. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks* 1993; **6**(4):523–533.
27. Tam ML, Lam WHK. Using automatic vehicle identification data for travel time estimation in Hong Kong. *Transportmetrica* 2008; **4**(3):179–194.
28. Chen M, Yaw J, Chien SI, Liu XB. Using automatic passenger counter data in bus arrival time prediction. *Journal Of Advanced Transportation* 2007; **41**(3):267–283.