Predicting the travel time of arterial traffic using particle filter with speed matrix

Qianguong Yang*, and Qi Peng
Fifth research Institute of MIIT, China

Abstract. Travel time prediction is an essential part of intelligent transportation system applications. However, the existing travel time prediction methods mainly focus on the freeway due to its simplicity and the high coverage of sensors and few researches have been conducted for the urban arterial road. Consequently, a travel time prediction algorithm based on particle filter is proposed in this paper to predict short-term travel time of the arterial traffic with historical floating car data and the concept of speed matrix is developed to illustrate the spatiotemporal properties of the arterial traffic. Unlike previous travel time prediction methods, the proposed algorithm uses particles with corresponding weights to model the traffic trend in the historical data instead of state-transition function and the weight for each particle are calculated with similarities between the speed matrix of the particle and the current traffic pattern. Moreover, a resampling process is developed to solve the degeneracy problem of the particles by replacing the low-weight particles with historical data. A real floating car dataset of 10357 taxis over a period of 3 months within Beijing is utilized to evaluate the performances of the algorithms. The proposed algorithm has the least errors by comparing with other three algorithms.

1 Introduction

As one of the best indicators for evaluation of the performances of the transportation system, accurate travel time data are crucial for efficient traffic management and transport planning[1,2]. All traffic manage centers are encouraged to provide travel time and incident information, which give useful information to travelers and help them to make smart route decisions [3-5]. Such information can help drivers choose the route to detour from the congested highways and provide additional capacity, which will reduce the burden of the traffic and help relieve the congestions. Moreover, reliable travel time prediction results enable establishing dynamic control strategies. As performance indicators, travel times data can also serve as a tool for comparing various traffic management strategies quantitatively. Accurate travel time can enhance traffic management systems by giving opportunities to react to the traffic changes proactively rather than passively.

In this paper, an algorithm is proposed to use the particle with corresponding weight to model the historical trend hidden in the historical data instead of building up a state-transition function with macroscopic physical model. After a new measurement arrives, the similarity between the historical traffic patterns represented by each particle and

* Corresponding author: yangqiangrong@ceprei.biz

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (http://creativecommons.org/licenses/by/4.0/).
the current traffic pattern is calculated and the particles with low similarities are removed and resampled with SIR (sampling importance resampling). The concept speed matrix are developed to represent the spatial-temporal properties of the target segments and the similarities are calculated with speed matrix instead of a time sequence of speed which is used in most of the previous particle filter approaches. A floating car dataset from 10357 taxis over a period of 3 months within Beijing are utilized to evaluate the proposed approach. The results show that the proposed approach performs better than the nearest neighbor method and two kalman filter methods.

2 Methodology

2.1 Generation of the speed matrix

\[
\begin{pmatrix}
v_{11} & v_{12} & \cdots & v_{1m} \\
v_{21} & v_{22} & \cdots & v_{2m} \\
\vdots & \vdots & \ddots & \vdots \\
v_{n1} & v_{n2} & \cdots & v_{nm}
\end{pmatrix}
\]

Speed matrix is a concept used in many transportation applications to represent the traffic condition of all the roads in all the time slots under consideration and equation (1) shows the speed matrix for road segment set \( R = \{r_1, r_2, \ldots, r_m\} \) and time slot set \( T = \{t_1, t_2, \ldots, t_n\} \). In this paper, to demonstrate the spatiotemporal properties of the traffic condition of the target road \( r \), the speed matrix is generated for all the segments whose distance to \( r \) is less than 3 (which are the target road \( r \), the neighboring segments of \( r \) and the neighboring segments of the neighboring segments of \( r \)). The time slots under consideration are from short past time \( t-p+1 \) to the current time slot \( t \).

2.2 Particle filter approach for travel time prediction

After generating the speed matrix, the travel time can be predicted with the particle filter approach. The input for the particle filter approach is the current speed matrix generated from the received new floating car data for each time interval and the update of the measurement from \( Y_{t-1} \) to \( Y_t \) is conducted by shifting the measurement time window one step forward. Each particle can be recognized as a speed matrix which represents the spatiotemporal traffic condition of the target road. The time update of each particle from \( x_{t-1}^i \) to \( x_t^i \) is conducted by shifting the data one step ahead in the historical data. At the same time, the corresponding weight can be calculated as the similarities between the real-time traffic pattern and the historical traffic pattern. Consequently, the distribution of the travel time on the next time interval \( t+1 \) can be predicted as \( \sum_{i=1}^{N} w_i^i x_{t+1}^i \). For multi-step prediction with prediction horizon \( p \), the speed matrix is generated by moving forward \( p \) steps iteratively, but the weights for each particle are kept the same with the weight updated by the current measurement. So the travel time on \( t+p \) can be predicted as \( \sum_{i=1}^{N} w_i^i x_{t+p}^i \).
Then the rest of this section will present the detailed steps of the particle filter approach.

**Initialization:** Since there is no priori-knowledge about the initial states of the particles, the initial speed matrix should be assigned to all particles. For each particle $i$, the initialization process is conducted by randomly choosing the day index $d^i_0$ and the time index $j^i_0$ in that day. Then the initial speed matrix for that particle is constructed by selecting the corresponding historical data. Assume there are $n$ segments whose distance to the target road $r$ is less than 3 and the time slot index for day $d^i_0$ and time $j^i_0$ is $t^i_0$, then the initial speed matrix is shown in equation (2).

$$
\begin{pmatrix}
  v_{1, (j^i_0 - p + 1)} & v_{1, (j^i_0 - p + 2)} & \ldots & v_{1, j^i_0} \\
  v_{2, (j^i_0 - p + 1)} & v_{2, (j^i_0 - p + 2)} & \ldots & v_{2, j^i_0} \\
  \vdots & \vdots & \ddots & \vdots \\
  v_{n, (j^i_0 - p + 1)} & v_{n, (j^i_0 - p + 2)} & \ldots & v_{n, j^i_0}
\end{pmatrix}
$$

(2)

where $v_{ij}$ represents the speed of road $i$ at time slot $j$.

**Time update process:** Since there is no state-transition function in the proposed approach, the time update process is conducted by propagating the states of each particle along the historical data sequence. For a particle $X^i_{t-1}$ at time $t-1$, the time update process is conducted by shifting the time index one step further and keeping the same day index, which are shown in equation (3)(4). The next process is trying to find out the particles that are not able to predict the travel time $p$ steps ahead. These invalid particles are selected by moving the time index $p$ steps ahead and check if there is available historical data on that time index. All the selected invalid particles are removed and will be resampled in the following steps.

$$
d^i_t = d^i_{t-1}, i \in [1, N]
$$

(3)

$$
f^i_t = j^i_{t-1} + 1, i \in [1, N]
$$

(4)

**Measurement update process:** In this paper, the similarity function are calculated with the speed matrix in the following way: First, assume the speed matrix for particle $i$ and the current traffic pattern in time index $t$ are $M^i_t$ and $M^c_t$. And the number of the segment whose distance to the target segments $r$ is 0, 1, 2 are $1, n_1, n_2$ respectively ($1+n_1+n_2=n$). Moreover, it’s reasonable to assume the data sequence in the speed matrix $M$ for each kind of segments are respectively $\{m_1\}, \{m_2, m_3, ..., m_{n_1+1}\}, \{m_{n_1+1}, m_{n_1+1}, ..., m_n\}$ ($m_i$ is the $i$th row of speed matrix $M$) because the rows are easy to change order without impacting the speed matrix. Then the linear correlation coefficients are calculated for between each data sequence $m^i_j$ and $m^c_j$ ($j=1, 2... n$) in the speed matrix for particle $i$ and current traffic.

However, the similarities for different kinds of segments will have different impact on the target segment (the similarity between the traffic patterns of the target roads are more meaningful than the similarities between the traffic patterns of the neighboring segments). For the above reason, different weights are assigned to the similarities and the similarities are calculated as below:
\[ p_{\epsilon}(y_{t} - h(x_{t})) = \rho(m_{t}^{i}, m_{t}^{c}) + \lambda_{1} \sum_{j=1}^{n} \rho(m_{t_{i},t_{j}}^{i}, m_{t_{i},t_{j}}^{c}) + \lambda_{2} \sum_{j=1}^{n} \rho(m_{t_{i},t_{j}}^{i}, m_{t_{i},t_{j}}^{c}) \]  

(5)

In this paper, \( \lambda_{1}, \lambda_{2} \) are set to equal 0.7, 0.5 respectively. The reason for choosing the weight as above is that the roads whose distance is larger than 2 will have little spatial and temporal effect on the target road according to the similarity analysis. Moreover, obviously the more neighboring road will have larger effect on the target and the weight of the road are chosen according to experiment results.

Thereafter, all the valid particles are sorted according to the weight in descending order and the top \( \text{N}_{\text{val}} \) particles are maintained and the rest particles will be resampled in the next step. In this way, the invalid particles will be re-selected so that new particles with similar traffic pattern with the current traffic pattern can be selected.

**Resampling:** In the Measurement updating process mentioned in the last step, there is a common problem named degeneracy problem. The degeneracy problem is a problem that the variance of the weight can only increase in time, which results in that only one particle have a non-negligible weight after several iterations. Consequently, the particles need to be resampled for the reason that the concept of particle filter is to use a large number of particles to approximate the posterior function accurately. The basic idea of resampling is removing the particles with small weight and concentrating on the particles with large weight. A partial sampling importance resampling (SIR) is proposed instead of the traditional threshold-based resampling method, which will be discussed in detail below.

SIR is derived from the sequential importance sampling by sampling in different pre-chosen intervals instead of sampling uniformly in the whole data. The SIR approach can be easily updated and the importance is easily calculated. In the particle filter approach, the remaining \( \text{N}_{\text{val}} \) invalid particles in the last step will be resampled in this step. First, the similarities \( \lambda^{i} \) between the traffic pattern of each day \( Y^{i} \) in the historical data (Assume there are D days of valid data in the historical data) and the current day \( Y^{c} \) are calculated. Next, the cumulative density intervals \( C \) are constructed based on the similarities \( \lambda^{i} \) with equation (6). Finally, for each particle in the remaining \( \text{N}_{\text{val}} \) invalid particles, a random value \( \sigma^{i} \) is generated between \([0, \sigma^{D}]\) uniformly and determine which interval covers \( \sigma^{i} \). If \( \sigma^{i} \) is in the jth interval, the day index of this particle is set to be \( j \) and the time index is randomly selected. Moreover, the weight of this particle is set to be \( \lambda^{j} \). In this way, the day with closer similarity to the current traffic pattern has a larger probability to be selected in the resampling process.

\[ c^{i} = \lambda^{i}, c^{j} = c^{j-1} + \lambda^{i}, i \in [2, D] \]  

(6)

**Travel time prediction:** In the proposed particle filter approach, the p-step ahead travel time prediction is estimated by shifting the time step of each particle along the historical data and keeping the same weight until a new measurement arrives. The average prediction results are calculated as the weighted average travel time of each particle as equation (7).
are chosen according to experiment results.

A partial sampling importance resampling (SIR) is proposed instead of the

the more neighboring road will have larger effect on the target and the weight of the road

data and keeping the same weight until a new measurement arrives. The average prediction

the variance of the weight can only increase in time, which results

removing the particles with small weight and concentrating on the particles with large

particles to approximate the posterior function accurately. The basic idea of resampling is

the resampling process.

In this paper,

\[
\sum_{i=1}^{N} w_i = \sum_{i=1}^{N} w_i x_{i, p} x_{i, p}
\]

\[
(7)
\]

3 Evaluation

3.1 Data and environment set up

The evaluation of the proposed algorithm is conducted based on a real probe vehicle

trajectory dataset[6,7], which are collected by the Microsoft research Center for the project

T-drive. T-drive is a smart driving direction services based on GPS trajectories of a large

number of taxis. It helps user find out the practically fastest path to a destination at a given

departure time. The collected dataset contain the GPS data from 10357 taxis over a period

of 3 months (from 2013.09 to 2013.12) within Beijing. The total number of GPS points

reaches 15 million and the total distance of the trajectory reaches 9 million kilometers. The

average sampling interval is about 177 seconds with a distance about 623 meters. Each data

sequence consists of the taxi ID, data time, longitude, latitude.

3.2 Evaluation of the algorithms

Two Kalman filters[8,9] and KNN are used in the evaluation of the method as comparative

method. The MAE and MAPE of the four travel prediction algorithms are represented in

Table 1. As shown in the table, the proposed particle filter approach produces a better

performance over other methods in all predictions horizons. Among all the prediction

methods, the first kalman filter with the state-transition function which models the ratio

between the measurement of time t-1 and t-2 produces the worst performances. The results

of the first kalman filter show a degeneracy problem, which means that the errors produced

by this method increase significantly as the prediction horizon increases. The reason for this
degeneracy problem lies in the assumption of the method that the short-term traffic pattern

will propagate in the near future. However, when the prediction horizon is large, this

assumption may not be true. The KNN method outperforms both kalman filters in all

prediction horizon but it’s not as good as the proposed algorithm. Moreover, both the KNN

and the proposed algorithms produce consistent results over a long period prediction time

(1 hour). The relative MAPE increase from 10 min to 1 hour for the KNN method and the

proposed method are 41.2% and 37.2% respectively, which is much smaller than both

kalman filters (65.9% and 68.8% respectively).

Table 1. Predictions result of all algorithms.

| Method | Indices | 10   | 20   | 30   | 40   | 50   | 60   |
|--------|---------|------|------|------|------|------|------|
| KF1    | MAE(min)| 4.42 | 4.81 | 5.29 | 5.97 | 6.41 | 6.95 |
|        | MAPE (%)| 21.19| 24.09| 26.14| 30.57| 33.48| 35.78|
|        | RMSE(min)| 6.69 | 7.21 | 7.76 | 8.95 | 9.74 | 10.52|
| KF2    | MAE(min)| 4.03 | 4.62 | 5.02 | 5.46 | 6.11 | 6.60 |
|        | MAPE (%)| 20.36| 23.67| 25.33| 29.13| 31.72| 33.73|
|        | RMSE(min)| 6.21 | 6.87 | 7.53 | 8.09 | 9.16 | 9.97 |
### 3.3 Parameter determination

A sensitivity test is conducted to demonstrate the impact of three parameters: the length of the data sequence \( p \), the number of particles \( N \) and the resampling ratio \( 1 - \frac{N_{val}}{N} \). The optimal value for these parameters are also determined in the sensitivity test. First, the value ranging from 1 to 10 for the length of the data sequence \( p \) are used to calculate the average MAPE for three prediction horizons (10, 30 and 60 minutes) and the results are shown in Figure 1. From Figure 2, we can see that the least error is obtained when \( p \) equals to 4, which means that the least error can be obtained by utilizing the previous 40 minutes travel pattern. The reason for this may lie in the fact that there is an optimal number of utilizing the historical data. When the number is too small, there is not enough information about the current traffic pattern. However, when the number is too large, some historical data are too old to bring any valuable information about the current traffic pattern but bring some disturbances for predicting the short-term travel time. Moreover, different numbers of particles from 10 to 400 are evaluated by calculating the average MAPE for three prediction horizons (10, 30 and 60 minutes). The results are shown in Figure 2 and the MAPE decreases as the number of particles increases, but when the number is larger than 100, the MAPE almost remains the same. Therefore, the optimal value of \( N \) is 100. Lastly, the ratio of resampling is investigated by calculating the MAPE for the value ranging from 10% to 90%. The results are shown in Figure 3 and the least error reaches when the value of the resampling ratio equals to 20%.

![Fig. 1. The error for different number of data sequence.](image-url)
3. Parameter determination

As a sensitivity test is conducted to demonstrate the impact of three parameters: the length of the data sequence \( p \), the number of particles \( N \) and the resampling ratio \( 1 - N_{\text{val}} / N \). The optimal value for these parameters are also determined in the sensitivity test. First, the value ranging from 1 to 10 for the length of the data sequence \( p \) are used to calculate the average MAPE for three prediction horizons (10, 30 and 60 minutes) and the results are shown in Figure 1. From Figure 2, we can see that the least error is obtained when \( p \) equals to 4, which means that the least error can be obtained by utilizing the previous 40 minutes travel pattern. The reason for this may lie in the fact that there is an optimal number of utilizing the historical data. When the number is too small, there is not enough information about the current traffic pattern. However, when the number is too large, some historical data are too old to bring any valuable information about the current traffic pattern but bring some disturbances for predicting the short-term travel time. Moreover, different numbers of particles from 10 to 400 are evaluated by calculating the average MAPE for three prediction horizons (10, 30 and 60 minutes). The results are shown in Figure 2 and the MAPE decreases as the number of particles increases, but when the number is larger than 100, the MAPE almost remains the same. Therefore, the optimal value of \( N \) is 100. Lastly, the ratio of resampling is investigated by calculating the MAPE for the value ranging from 10% to 90%. The results are shown in Figure 3 and the least error reaches when the value of the resampling ratio equals to 20%.

4 Conclusions

A travel time prediction algorithm based on particle filter is proposed in this paper to predict short-term travel time with historical floating car data and the concept of speed matrix is developed to illustrate the spatiotemporal properties of the traffic pattern. First, a large number of particles are initialized with each particle representing a historical data sequence with a randomly chosen day index and time index. Then, unlike previous travel time prediction methods, the proposed algorithm uses particles with corresponding weight to model the traffic trend in the historical data instead of using the state-transition function. Moreover, the weights for each particle are calculated with similarities between the speed matrix of the particle and the speed matrix of the current traffic pattern. Finally, a resampling process is developed to solve the degeneracy problem of the particles by removing the low-weight particles with historical data that has high similarity with current traffic pattern and the short-term travel time is predicted by the weighted travel time of all particles. However, the performances of the proposed algorithms is still not as good as the
The results of the travel time prediction in freeways for the complexity of the arterial traffic. Therefore, the future work should focus on modeling the complicated arterial traffic network and proposing a more precise way to represent the spatiotemporal properties of the arterial traffic.

References

1. Wang J, Indra-Payoong N, Sumalee A, et al. Vehicle reidentification with self-adaptive time windows for real-time travel time estimation [J]. Intelligent Transportation Systems, IEEE Transactions on, 2014, 15(2): 540-552.

2. Chen H, Rakha H A. Real-time travel time prediction using particle filtering with a non-explicit state-transition model [J]. Transportation Research Part C: Emerging Technologies, 2014, 43: 112-126.

3. Oda T. An algorithm for prediction of travel time using vehicle sensor data[C]//Road Traffic Control, 1990. Third International Conference on. IET, 1990: 40-44.

4. Al-Deek H M, D'Angelo M P, Wang M C. Travel time prediction with non-linear time series[C]//Fifth International Conference on Applications of Advanced Technologies in Transportation Engineering. 1998.

5. Anderson J M. The short term prediction of link travel times in signal controlled road networks [J]. Transportation systems: theory and application of advanced technology. Vol. 2, 1995.

6. Yuan, Jing, et al. "Driving with knowledge from the physical world. "Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2011.

7. Yuan, Jing, et al. "T-drive: driving directions based on taxi trajectories. “Proceedings of the 18th SIGSPATIAL International conference on advances in geographic information systems. ACM, 2010

8. Cawley G C, Talbot N L C. Fast exact leave-one-out cross-validation of sparse least-squares support vector machines [J]. Neural networks, 2004, 17(10): 1467-1475.

9. Cawley G C, Talbot N L C. Efficient leave-one-out cross-validation of kernel fisher discriminant classifiers [J]. Pattern Recognition, 2003, 36(11): 2585-2592.