International Journal of Applied Mathematics and Computing Science. Volume 4, Issue 1, (2015) 42-58
http://www.cayley-nielson.com/ijamcs

Missing Data Compensation Model in Real-Time System of Floating Car Data

Xu Qiang[1], Bo Li[2]

[1]School of Electronic Engineering, Beijing University of Posts and Telecommunications, No 10, Xitucheng Road, Haidian District, Beijing 100876, China

[2]Key Laboratory of Data Storage Systems, Ministry of EducationHuazhong University of Science & Technology, 1037 Luoyu Road, Wuhan, China

E-mail: [1]*xuqiang@bupt.edu.cn*, [2]*ielibo@hust.edu.cn*

Abstract

Nowadays the Floating Car Data (FCD) is playing a more and more important role in dynamic data collection, because its coverage and precision. Urban traffic system is a typical nonlinearity system. With increasing data availability from FCD, it is increasingly possible to develop road network real-time performance measures. Base on it, we have the ability to construct the evaluation of city-scale traffic conditions using system dynamics. However, the level of accuracy expected from FCD highly depends on the level of error related to the vehicle positioning and the road network cannot be covered by available real-time FCD. How to make use of them to perform trusty in traffic information estimation is a key issue. In this work, a multi-pattern compensating model based on History FCD (HFCD) is proposed to compensate the missing real-time traffic data. The experiment which involved the data of 15,000 taxies for 6 months was carried out in several ways and the result suggests that this method raised the road coverage while guaranteed the accuracy and it can be applied in real-time systems to manage large amount of data.

Keywords: Missing Data Compensation, Floating Car data (FCD), pattern matching.

1. Introduction

As the rapid population growth and urbanization process, millions of people who work and shopping in metropolis but live in the surrounding suburbs have to spend plenty of time on travelling. For residents, understanding real-time and reliable traffic information concerning the best routes from their location to their destination (OD) is a requirement in their daily lives. OD travel time based on Real-time Traffic Information Digital Map (RTIDM), which describes real-time traffic information of the road network, can be evaluated by the dynamic route guidance. For people who would like to travel with optimized route and take full advantage of the road network resources, it is one of the methods to alleviate the traffic pressure on the basis of the present traffic foundation facilities.

Supposing the scene of commuters to work from their living place, either traveling by private car, by bus, by subway or by taxi need to estimated travel time basis real-time information determined the route that their goes on a journey, and chooses the fast one. A network is made of basic pieces usually referred to as links, and OD travel time estimation also means that, link travel time should be calculate timely, correctly and sustainability.

For several decades, fixed sensors technologies are mature to provide precise and stable data on the current traffic situation. It plays a key role in traffic engineering analysis (e.g. traffic operation, road planning, and policy making purpose, etc.). However, fixed detectors' capabilities are limited due to high

ISSN 2333-2425
March, 2015

costs for setting up and maintaining the required infrastructure and their poor road network coverage. For practical reasons, it is suffering from bad weather conditions and traffic interruptions. Fortunately, with the advent of Floating Car Data (FCD) and specifically the GPS based tracking data component, a means was found to derive accurate and up-to-date travel time. What it was concept years ago, it is now becoming routine all over the world. The strength of this technology stems from high quality real-time data collected from thousands of vehicles over a large road network and for much less cost than traditional methods. Base on it, important parameters like average speed, OD travel time, and intersection delay can be used for performance monitoring of the transportation system. Therefore, the methods for assessing and reporting traffic characteristics and conditions have begun to shift. These improvements are not only expected to sever commuters, but also to benefit all the transportation actors, although at different degrees. For instance, road managers will have a cost-effective tool to obtain continuous and wide-covering data leading to better traffic monitoring in real time, better understanding of the traffic patterns.

However, due to floating car erratic driving and unusual behaviors, it still suffers from limited time/spatial coverage. For details, the road network is made of basic pieces usually referred to as links, and the trajectories are collected by map matching (MM) method to the urban network on a fix rolling basis (such as 5min). The issue is straightforward: On the one hand, floating cars sometimes have an optional halt behavior, such as speeding down or waiting for passengers, stopping at a red light, these can cause error fluctuation; on the other hand, during each rolling period, the difficulties was trying to cover all links of the road network, because that's nearly impossible, especially at nighttime. Consequently, if these vacant data are not compensated, major problem in dynamic navigation systems is due to the unreliable OD travel time estimated with the more complicated fluctuation data.

Recently, a lot of methods about the problem based on fixed detector data are proposed. Yuh-Horng WEN [1] raised a grey time-series model and a grey-theory-based pseudo-nearest-neighbor method to compensate vacant data. Jianwei Wang [2] developed two imputation approaches to compensate. However, these approaches based on fixed detector data (FDD) are not applicable to recover missing floating car data (FCD), because FCD describes the position and velocity of floating cars, whereas, FDD expresses traffic flow of roads. Moreover, some methods of statistics [3-6], such as neural network, Kalman filtering, are also not applicable, because they could not meet the need of real-time processing of FCD.

This paper has two contributions: one is the definition of Link Traffic Pattern to present to describe different characters of a link in dynamical ways, and the other is proposed a cost-efficiency method to compensate vacant link information in real-time. This paper starts by summarizing related research, and then describes the requirements and associated problems, second, the definition of Link Traffic Pattern is raised to describe all the trends appeared in HFCD of driving speed of a road in a time period and using it to construct Link Traffic Pattern Database (LTPD). Next Temporal-Spatial Compensation Model (TSCM) based on HFCD is proposed after carrying out plenty of data analysis and experiments on FCD. And then a series of evolution is exploited to verify the accuracy and coverage of our model. Conclusion is mentioned at last.

2. Problem Description

The related definition and problem description are introduced in this section.

2.1 Network structure

Road network is composed of links and road nodes. A link is an atomic road portion such as a piece of road between two intersections. Nodes are the points in the map which are connected by links.

Each link has a unique direction. Therefore, links of the road network can be represented in vector data. A link is associated with a dynamic weight ρ, which reflects the speed variety in a sampling interval. The weight of link l ρ_l can be written as:

$$\rho_l = \frac{L}{T} \tag{1}$$

L describes the length of *link l* and T describes the travel time needed in crossing *link l.*

The weight ρ is determined by the trajectories, which covers the links in the latest sampling interval. The vector data of *link l* is described as $\phi(l)$, which has the start node $u(l)$ and the end node $v(l)$. The entrance-links of link l are the links which $u(l)$ works as their end nodes and $id(l)$is accounted as the numbers of such links. The exit-links of link l are the links which $v(l)$works as their start nodes and $od(l)$is accounted as the numbers of such links. The road network model R can be described as follows:

$$\begin{cases} R = (I, E) \\ E = \left\{ e = \left\langle p, E_{pre}, E_{next} \right\rangle \mid E_{pre}, E_{next} \subset E \right\} \\ E_{pre} = \left\{ e_\phi \mid \phi_p(x), 0 \le x < 360 \right\} \\ E_{next} = \left\{ e_\phi \mid \phi_n(x), 0 \le x < 360 \right\} \\ I = \left\{ \left\langle e, e_\phi \right\rangle \mid e, e_\phi \in E \right\} \end{cases} \tag{2}$$

where I means the set of nodes, E means the set of the links, and each link e can be described as a triplet cardinality $\left\langle p, E_{pre}, E_{next} \right\rangle$ E_{pre} is the entrance-links set, and E_{next} is the exit-links set; p describes the attributes of e. $\phi_p(x)$ is the function to describe the angle relationship between e and its entrance-links; $\phi_p(x)$ is the function to describe the angle relationship between e and its exit-links. The road structure can be shown in Figure 1 as follows:

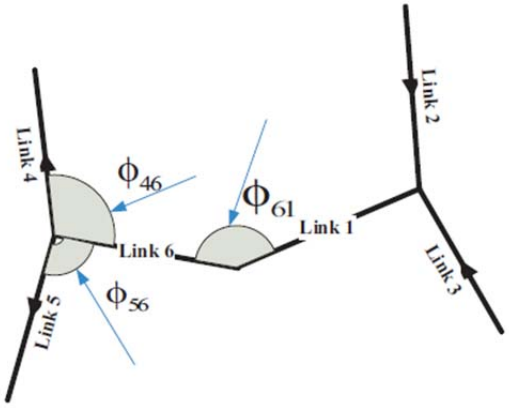

Figure 1. Road structure

Link 1 is the entrance-link of *Link 6; Link 4* and *Link 5* are the exit-links of *Link 6; the* angel between *Link 4* and *Link 6* is ϕ_{46}; the angel between *Link 5* and *Link 6* is ϕ_{56}, ϕ_{61} is the angel between *Link 6* and *Link 1*.

Definition: Time slice

The weight ρ_l is updating in cycle and each updating cycle is defined as a time slice. For example, if the time slice is 5min and the time slices of a day are 288.

2.2 Problem description

A dynamic weight ρ of each link is implemented by means of the trajectories calculated by the system in the latest sampling interval. Because the scale of covered trajectories is constrained by the generate time and the number of floating cars, the entire road network is impossible to be covered by trajectories in a sampling interval. The circumstance is showed in Figure 2.

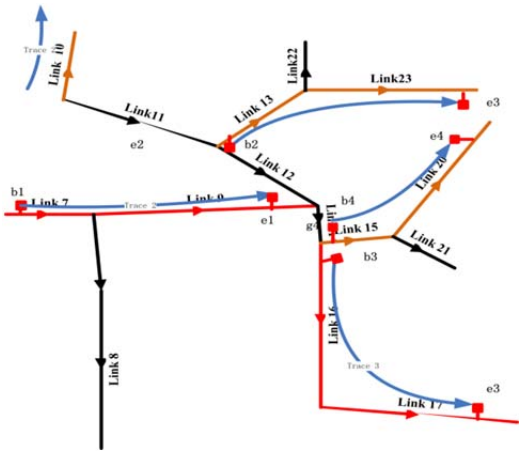

Figure 2. Dynamic weight ρ generated by trajectories of FCD

Where red points relate to the GPS positions of vehicle, a trajectory is the path composed of every 2 GPS position and the link between them. Red and yellow lines describe the link covered by trajectories, and black lines describe the link without any cover; blue lines express the vehicles' travel direction. As is shown in the figure above, the blank-data links exists within the travel routes. For example, the trajectories from the northwest to southeast, from *Link 11*, *Link 12* to *Link 16*, *Link 17*, has two blank-data links, *Link 11* and *Link 12*. For other links like *Link 17*, the travel time can be calculated as they have data on themselves. But for the vacant links, they don't have the source data to calculate the travel time, and errors will occur at last. According the related study of FCD, the factor of travel time has something to do with temporal factors (weekly and sampling interval of the day) and spatial factors (portion of the road network), as is shown in Figure 3.

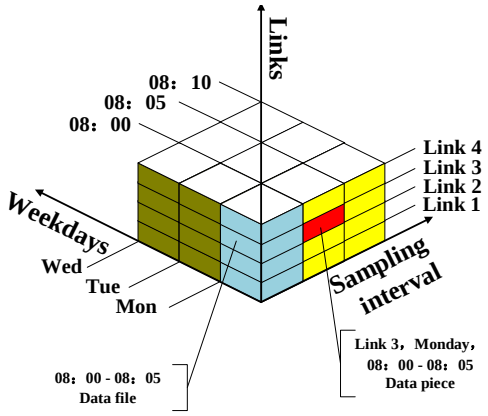

Figure 3. The factor of travel time

Every Data piece (DP) is the weight ρ of link l in a time slice (e.g. the red DP shown in Figure 3)

and record as D_i, where i describes the time slice index. For example, D_1 represents the data collected from 0:00am to 0:05am. $P_i(w)$ is the weight of D_i, which is defined as Road Network Traffic Point (RNTP) and can be expressed by:

$$P_i(w) = \frac{1}{n}\sum_{k=1}^{n} \rho_k, i = 1, 2, ..., T \tag{3}$$

Where n is the number of the non-blank-data links which have data within the lasted sampling interval, w describes the day type (means day of week, such as Sunday, Monday, and so on).

Definition 5: Link Traffic Pattern (LTP)

LTP is a vector of dynamic weight; it describes the variation trend of dynamic weight of a link in a time period. The LTP can be expressed as:

$$V_l^d(t_0, t_0 + h) = \left(\rho_{t_0}, \rho_{t_0+1}, ..., \rho_{t_0+h}\right), 0 \le t_0 < t_0 + h \le T \tag{4}$$

Where d is the data type, such as Monday or Sunday, l is the link ID in road network, t_0 is used to express the time slice and h offset of time slice and T is the count of time slice in a day? The LTP is composed of dynamic weight of the same day, as is shown in Figure 4.

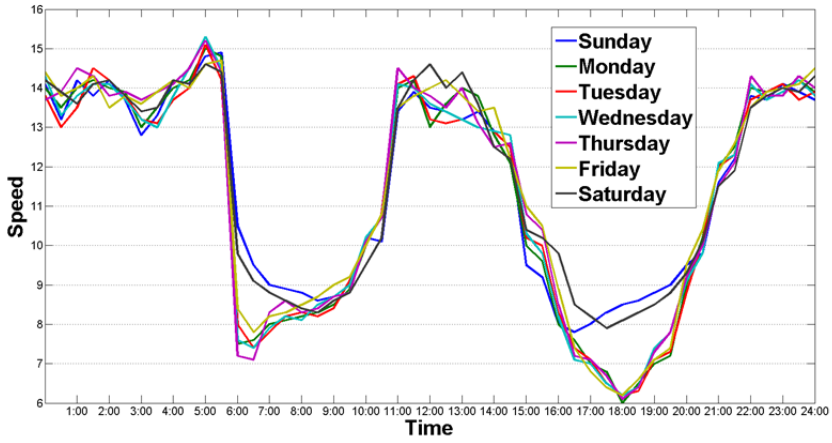

Figure 4. Link Traffic Patterns of a week

There are two kinds of factors affecting LTP: one is local event, which is caused by unusual road congestion, such as accident or traffic control at special time of day. Reference [8] proposed the concept of EZ (evacuation zone from regional) to describe such an effect: congestion in road traffic would usually be a regional impact. The other is global event, which is caused by special days or system errors. The data generated by system error cannot reflect the correct traffic flow trend changes, therefore, it should be removed; the data generated by special factor (like holidays) which reflects a special traffic flow trend, should be treated as an independent kind. Reference [7] proposed a method to compensate for missing data based on historical data, but it did not take these factors into consider. By analyzing the FCD historical data of 1.5 months generated by 15000 cars in Beijing, an example is shown in Figure 4. It expresses the LTP of the same day type for several weeks, which appears to be similar all time of day. But one of the LTP differs significantly during the period of day time, maybe in sake of some global events. In conclusion, Massive, accurate FCD will provide a strong basis to compensate for missing data in the latest

sampling interval. These data come from the high-quality historical data which has the similar LTP with the real time data [2]. And factors such as events and weather will also have great influence on the data, so when the processing of compensating is going on, such kinds of data should be taken out and processed independently.

3. Assistant data

Only the highly effective and accurate methods of compensating can meet the requirements of the real-time processing. Consequently, it is needed to pick out some of the traffic information which has similar characteristics with the real-time data, as assistant information source in compensating. The entire process can mainly be divided into two steps: the optimizing of the road network, and the classified storing and organizing of the assistant data source.

3.1 Road network optimizing

The first work should be done is to optimize the road network. This means an appropriate model for the network should be built. A digital map has different levels of links which play different Role in Traffic (RIT). The RIT depends on 3 parameters: the function of the road, the percentage of the road of level k in the road network and the utilization of the different levels. While the sample probing cars keep at a stable quantity, utilization reflects the importance of a specific road level within a given period. For example, the levels of the highway should be different from the lanes. Utilization can be described as follows:

$$U_k = \frac{1}{n}\sum_{j=1}^{n}\left(\frac{1}{m}\sum_{i=1}^{m}L_{ij}\right) \qquad L_{ij} \in \{0,1\} \tag{5}$$

Utilization shows how many links of a certain level has been covered by trajectories according to a certain amount of historical data files. In the formula, U_k is the utilization of level *k.* As n is the amount of links of level k and m is the numbers of historical data files, L_{ij} shows the condition of the j^{th} link of level k in the i^{th} file. L_{ij} returns 1 when the link is covered by trajectories, and 0 when not.

In the digital map of Beijing, each link belongs to a unique level, and different levels express different temporal feature and different passing rules, so they should be treated separately. The utilizations of different levels are shown in Figure 5, which is derived from 12 months' 24h historical data (2011.2-2012.2). The x-coordinate is the levels of the links and the y-coordinate is the utilization of different levels. The difference among the levels can be seen from Figure5.

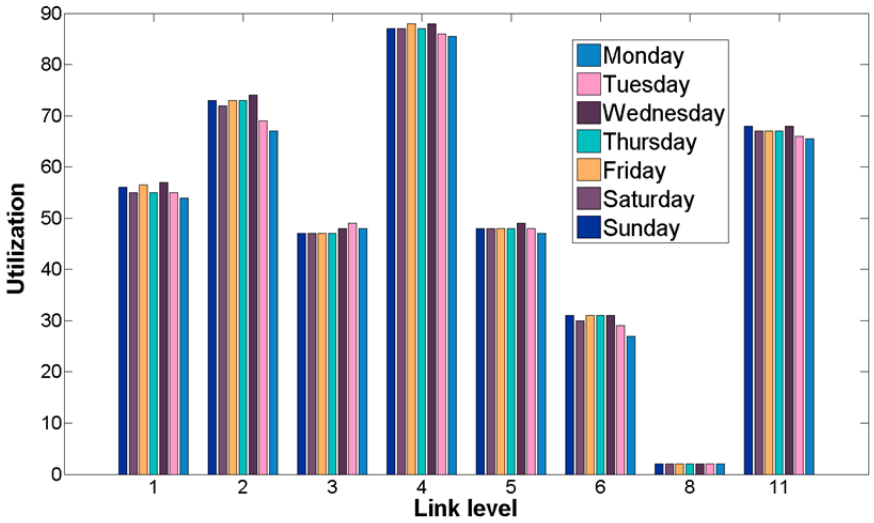

Figure 5: Utilization of different levels

However, the percentage of these levels in the road network is quite different. Figure 6 is the distribution.

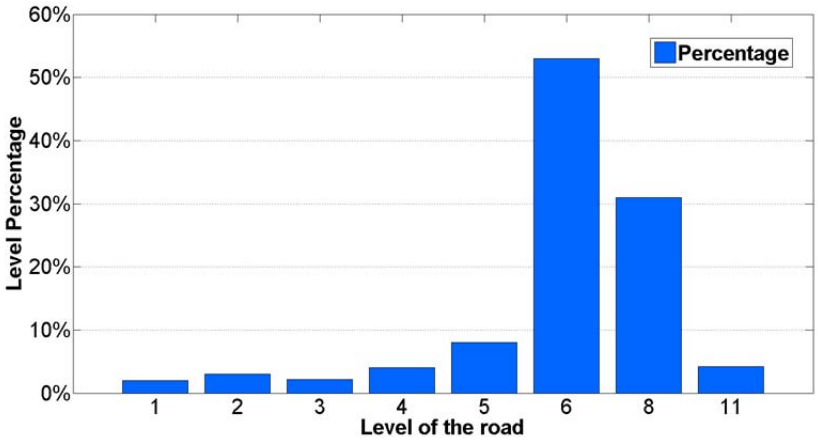

Figure 6. Percentage of different link level

It can be seen from this figure that the trunk roads and speedway accounts for low percentage but high utilization; the circumstance of the lanes and paths performs to be the opposite. Therefore, synthesizing the effect of levels and utilization, the levels of *1,2,3,4,5,11* are put into the main link set, and levels of *6,8* into the assistant link set. But such classification still has its own problems which can't be ignored. In the analysis of the historical data, it can be found that many of them even don't have any trajectory cover in quite a long time. Those were always the links which do not have the ability for passing vehicles. These links not only reduce the data coverage of the entire road network but also mislead the dynamic route guidance. Consequently, these links should be treated as an independent class called unreachable link set.

In summary, the road network is optimized as follows: the unreachable links L_{unre}, the reachable main links L_{main}, and the reachable assistant links L_{assi}. For set L of all links of the road network, this relation can be expressed like this: $L = L_{main} \bigcup L_{assi} \bigcup L_{unre}$. As the real-time data is derived every sampling interval, the three sets will also be updating dynamically: if one link in L_{unre} gets covered, it should be

taken it out and move it into L_{main} or L_{assi} according to its own levels; if one link out of L_{unre} gets no data in a quite a period (e.g. 2 months), they should be moved it into L_{unre}. Such a process can highly improve the accuracy and optimize the efficiency of compensating.

What mentioned above is the analysis of the historical data. And for the real-time data, there will be a missing data link set L'. Referring to the classification above, this relation is also obvious: $L' = L'_{main} \bigcup L'_{assi}$ .However, this doesn't contain the unreachable link set for the sake of real-time data. And what needs to be done next is to build the compensating models for the two classes separately.

3.2 Assistant data source

The assistant data source used in compensating should be derived from the historical data. As what Section 2 mentioned, DP is the basic unit of the historical data. And a data file contains the information of all the non-blank-data links within a certain sampling interval.

Therefore, what can be done first is to classify the historical data by day type, from Sunday to Saturday. Then if there's need to compensate for the data of a certain timestamp, just find the latest files of the same day type as the assistant data source. And by the study of the historical data, there are several factors which may result in signal noise, such as holidays and accidents. Therefore, in the storing of the historical data, a special part should be set to hold such data.

In conclusion, the data should be classified by day type and special factors. According to the classification, each type of the data represents a kind of traffic flow trend. In order to improve the system efficiency in compensating for missing data, the assistant data source should be preprocessed, which is derived from the historical data according to LTP of the real-time. The data used to compensate should have similar LTP in spatial dimension and the same period in temporal dimension. It is constrained by 2 conditions: day type and time offset. The way to meet the requirements of day type was mentioned above. And the time offset problem can be described that similar LTP may not be exactly synchronized in time, and there's always an offset. This circumstance can be expressed as follows:

$$T_c = T_h + \Delta t \tag{6}$$

where $T_c$describes the sampling interval when the realtime data file is generated, $T_h$describes the time when generating the data with similar LTP compared to the real time data file; Δt describes the time offset between realtime FCD and historical FCD which has similar LTP. Therefore, the factor of time should be adjusted in order to make sure that the real-time FCD be compensated correctly.

Suppose the max value of Δt is $\Delta t_{\max}$ and then the actual time offset must be fluctuating within $\left[-\Delta t_{\max}, \Delta t_{\max}\right]$. To guarantee the accuracy of the assistant data, the amount of historical data should be big enough to adjust the error. Therefore, the number of files is expressed as follows:

$$k = int\left[\frac{\Delta t_{max}}{t_{interval}}\right] \tag{7}$$

$\Delta t_{\max}$ should be as long as *k* sampling intervals $t_{\text{int}erval}$. Consequently, *2k* files on both edges and the file in the middle, which is in all *2k+1* files, can fully cover $\left[-\Delta t_{\max}, \Delta t_{\max}\right]$ and will accurately reflect the traffic flow trend. Such a group of files can be organized as an assistant data source to compensate one real-time data file. Consequently, all the files waiting to be compensated have their own *2k+1* files as the data source.

4. The patterns of compensating for missing data

In HFCD, a lot of data do not reflect the real traffic condition of links as a result of the erratic driving and some special driving character [8] of floating cars, and the effect of traffic incident. This data is called abnormal data in this paper. If these abnormal data are not identified and corrected, they will affect the accuracy of the TSCM seriously. In this section, the method of identifying and correcting abnormal data is given. According to the theory of traffic flow, the changing trend of driving speed of a road will not fluctuate tempestuously in a short time period. Thus, we can estimate whether or not the driving speed of a link in time point t is an abnormal value in the light of the driving speed of the link in the neighboring time point of *t*.

This chapter will put forward a Temporal-Spatial Compensation Model (TSCM). The model includes two kinds of compensation methods according to the situations that the situation is continuous data missing and the situation is scatter data missing. The process describes as Figure 7.

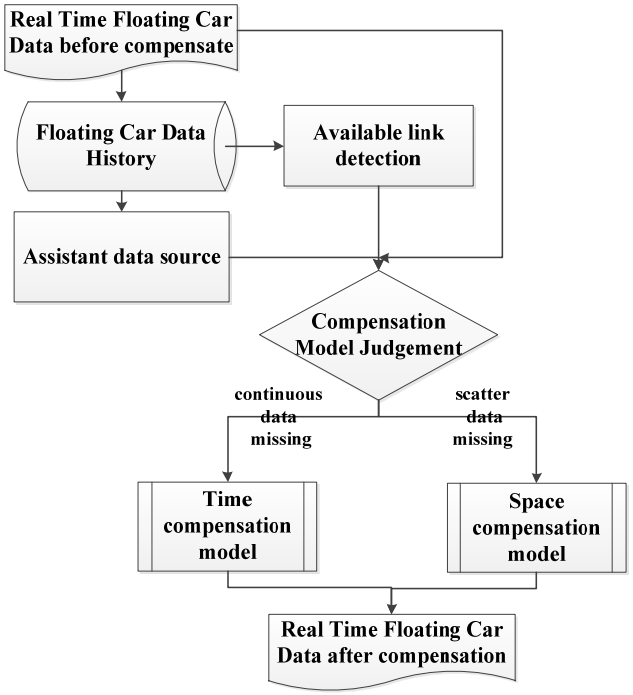

Figure 7. The process for compensation

Continuous data missing means that the information of the link is missing at the time from t_0 to $t_0 + h$ where $h \geq 2$, and we adopt the temporal compensation model to compensate. Scatter data missing means that the information of the link is missing at the time t_0 and the information of the link at the time from $t_0 - h$ to $t_0 - 1$ exists where $h \geq 2$, and under this circumstance we adopt spatial compensation model.

4.1 Temporal compensation model

HFCD can be used to compensate the vacant real-time traffic information, in that the operation of traffic flow will occur periodically according to the theory of traffic flow. Moreover, the theory also tells us that most of the changing trends of driving speed of a link in different days which share a same day of week are similar. Besides, as LTP reflects the trend of driving speed of a link in a time section, LTP is treated as the basic unit of analyzing HFCD. Besides, a time section means some continuous time slice, for example from t_0 to $t_0 + h$ where $h \geq 2$.

On basis of these facts, firstly, regular trends reflected by many LTPs of different days which share a

same day of week are analyzed. Then, similar trends are divided into a mode by cluster analysis. As is stated above, most of these regular trends are similar, thus, the amount of modes derived by cluster analysis is limited. After that, curve fitting method on all these mode is adopted to decrease the side-effect of random error, as real data have random error. And all the curves of these modes constitute the History Data Pattern Library (HDPL), the result of analyzing HFCD. And how to generate the HDPL will be discussed later. In the end, by matching the trend of real-time FCD in latest time range with modes in HDPL, the best matched mode which can be used to compute the vacant real-time driving speed of a link at current time can be derived.

(1) Regular trends reflected by LTPs analysis

As is stated above, firstly, we count the regular trends reflected by many LTPs of different days which share a same time section and a same day of week, and derive the modes after carrying out cluster analysis.

In this paper, as an LTP has n relatively independent values of speed, a LTP could be viewed as an n -dimensional vector. Therefore, suppose there are m days in HFCD which share a same day of week, namely, the amount of these n -dimensional vectors is m, the changing modes of driving speed of a link in a time section of some day of week can be deduced by cluster analysis on m vectors, where day of week means Monday, Tuesday and so on. For the convenience of description, each of these m vectors (or LTP) in a time section is called trend vector, and the set of these m vectors is called trend vector set below.

(2) Cluster analysis and HDPL

From the discussion stated above, it is known that, by cluster analysis on trend vectors, the changing modes of driving speed of a link in a time section of some day of week can be figured out.

Above all, it is crucial to choose an appropriate clustering algorism. K-means clustering is not suitable in that it should be given the amount of clustered classes in advance and the wrong value of the amount will affect the effect of clustering seriously. Besides, Clustering based on model cannot meet our need, because it is hard to conclude a model to express the changing trend of driving speed, as is shown in Figure 2. In this paper, min-max clustering algorism [9] is adopted, in that, this algorism can classify all the entered vectors to reasonable classes according to the relationship among these vectors, and do not need the amount of classes as well as some model in advance. Next, some related definitions are given below.

Definition 6 Neighboring vector

Suppose there are two n–dimensional vectors $X=(x_1,x_2,\ldots,x_n)$ and $Y=(y_1,y_2,\ldots,y_n)$, d is the Euclidean distance of X and Y, say $d=\sqrt{\sum_{k=1}^{n}(x_k-y_k)^2}$. If $\mathrm{d}<=\mathrm{m}\times\sqrt{\mathrm{n}}$, where $10<=\mathbf{m}<=20$, then X and Y call each other neighboring vector.

Definition 7 Vector density

Suppose there are several n -dimensional vectors, the amount of neighboring vectors of a vector is called density of the vector.

The basal principle of Min-max clustering algorism can be stated that, to all the entered vectors, it chooses a new clustering core by maximizing the distance between clusters and group these vectors by minimizing the distance in a cluster. Though the algorism has a lot of advantages, it is sensitive to the

value of initial clustering core. As is known to all, a clustering core should be close to a lot of the entered vectors in a local scope.

In this paper, to every vector of the trend vector set, the vector which has the most neighboring vectors is chosen as the first initial clustering core, and the vector whose Euclidean distance to the first initial clustering core is the biggest one is selected as another initial clustering core.

Definition 8 Average vector

Suppose there are m n–dimensional vectors $X_1=(x_{11},x_{12},\ldots,x_{1n})$, $X_2=(x_{21},x_{22},\ldots,x_{2n})$,…, $X_m=(x_{m1},x_{m2},\ldots,x_{mn})$, the average value of these vectors is defined as $\overline{X}$,

$$\bar{X}=\frac{1}{m}\sum_{i=1}^{m}X_i=\left(\frac{1}{m}\sum_{i=1}^{m}x_{i1},\frac{1}{m}\sum_{i=1}^{m}x_{i2},\ldots,\frac{1}{m}\sum_{i=1}^{m}x_{in}\right) \tag{8}$$

After the clustering algorism being carried out, the trend vectors which have smaller Euclidean distance are classified to a class. And average vector of the trend vectors of a class is used to represent the class. Therefore, the changing mode reflected by a class corresponds to the average vector, the final result of cluster analysis.

In conclusion, all of changing modes of driving speed of a link in time section of some day of week could be obtained by cluster analysis. The mode in a time section could be described as an n -dimensional vector as follows:

$$R=\{v_1(l,dw,t_1),v_2(l,dw,t_2)\ldots,v_n(l,dw,t_n)\} \tag{9}$$

where *l* shows a link, dw means some day of week (say Monday, Tuesday and so on), t_i expresses a time point of the time section, $v_i(l,dw,t_i)$ is the driving speed of link l at time point t_i of dw , n represents the amount of time points in a time section.

As the modes derived by cluster analysis are made up of a lot of relatively independent values of speeds and the existence of random error in real data of HFCD, larger error could be made when real-time traffic condition is matched with these modes. It is known to all that the method of curving fitting [10] based on the method of Least-squares[11] could make the least sum of squares of difference between data and its fitted curve, so it reduces the side-effect of the random errors. Besides, as the time range in a time section is short, the changing mode of driving speed of a link in a time section could be described with polynomial function. Therefore, the method of curve fitting based on m -polynomial function is adopted in this paper.

By fitting the modes derived by cluster analysis with curves based on m-polynomial function, each of the mode could be described as a $m+1$ dimensional vector which constituted by $m+1$ coefficients of the m–polynomial function. A $m+1$ dimensional vector *C* could be described as follows:

$$C=\{coe_1(l,dw,ts),coe_2(l,dw,ts)\ldots,coe_{m+1}(l,dw,ts)\} \tag{10}$$

Where l shows a link, dw means some day of week (say Monday, Tuesday and so on), ts expresses a time section, and $coe_i(l,dw,ts)$ means the i^{th} coefficient of the m-polynomial function expressed by C . Finally, by permuting and combining of all of l in road networks, all dw and all ts of a day in a library, a set which is composed of C is derived, and it is called History Data Pattern Library (HDPL) which is the final result of HFCD analysis.

And History Data Pattern Library (HDPL) can be described as followed:

$$\begin{cases} H=\left\{H^{t_0,h}(l,d)\,|\,l\in L,d\in D,h\in N\right\} \\ H^{t_0,h}(l,d)=\left\{C_1^{t_0,h}(l,d),C_2^{t_0,h}(l,d),...,C_n^{t_0,h}(l,d)\right\} \end{cases} \tag{11}$$

Where *l* shows a link, *d* means some day of week (say Monday, Tuesday and so on), t_0 expresses the time slice, *h* means the offset, and $C_i^{t_0,h}(l,d)$ means the i^{th} curve where the link is *l* ,the day type is d and the time from t_0 to t_0+h.

When compensating, firstly analysis the thread of speed before the information of the link missing, then match this thread with the patterns in HDPL, and find the best match. Finally, calculate the speed of the link at the current time according to the best match pattern.

(3) Missing data compensation

When HDPL is derived from the analysis of HFCD, vacant real-time driving speed of a link could be estimated by matching the trend of real-time FCD in latest time range with modes in HDPL.

Suppose in current real-time FCD (the current date is d which corresponds to dw day of week) the driving speed of *link l* is vacant at time point *t* (*t* belongs to time section ts) and other the driving speeds of link *l* before *t* is v_{t-1}, v_{t-2}, v_{t-t_0} (where $1<=t_0<=t$) in time section *ts* , the mode matching method based on the method of Least-squares is proposed to compensate the vacant data. As the matching method is based on the method of Least-squares, it could decrease matching error.

Firstly, all the corresponding curves of link *l* at day of week dw and time section ts are searched from HDPL. Suppose the amount of these curves is *n*. As all these curves are described by m-polynomial function, the function could be expressed as follows:

$$y_i=a_{i,m}x^m+a_{i,m-1}x^{m-1}+\ldots+a_{i,1}x+a_{i,0} \tag{12}$$

Where $1<=i<=n, a_m\in R, m>=0; x\in Z, x>=0; y_i>=0$, *x* shows a time point of time section ts, y_i means the driving speed of i^{th} curve C_i at x.

Then, v_{t-1}, v_{t-2}, v_{t-t_0} are viewed as a vector $\vec{v}$, $\vec{v}=\left(v_{t-1},v_{t-2}\ \ldots\ v_{t-t_0}\right)$. By calculating the least sum of squares of difference between $\vec{v}$ and y_i, the best matched curve C_b is obtained. Therefore, the estimation of the vacant driving speed at time point t is y_b which corresponds to curve C_b, $y_b=a_{b,m}t^m+a_{b,m-1}t^{m-1}+\ldots+a_{b,1}t+a_{b,0}$

4.2 Spatial compensation model

According to the theory of traffic flow, the changing trend of driving speed of a road has features related to time and space. Space correlation reflects that the current link condition has certain linear correlation with the adjacent link at same level. In this paper, we use Pearson Correlation Coefficient to measure time and space correlation.

For space correlation, Pearson Correlation Coefficient can be described as follows:

$$\omega_z^{spa}=\left|r_{ij}^s(s,n)\right|=\frac{\sum_{o=t_0-h}^{t_0-1}\left(v_o^j-\overline{v}^i\right)\left(v_o^j-\overline{v}^j\right)}{\sqrt{\sum_{o=t_0-h}^{t_0-1}\left(v_o^j-\overline{v}^i\right)^2}\sqrt{\sum_{o=t_0-h}^{t_0-1}\left(v_o^j-\overline{v}^j\right)^2}} \tag{13}$$

Where t_0-1 and t_0-h are the time slice, v_o^i is the speed of current link when time slice is *o*, v_o^j is the speed of adjacent link when time slice is *o* , $\overline{v}^i$ is the average speed of current link, $\overline{v}^j$ is the average speed of current link.

When missing the information of one link, we can use the information of adjacent link at same level to

compensate according to the space correlation.

Suppose now we link the speed information of link *l* in the t_0 time.

Firstly, select the speed information of link *l* in $t_0-h, t_0-(h-1), \ldots, t_0-2, t_0-1$ time, and then find the best match in the HDPL.

Secondly, after finding out the best match, calculate the Pearson coefficients between link *l* and the adjacent and the same level link in this best match.

Thirdly, multiply the Pearson coefficients and the speeds of the adjacent link at same level with link *l*, and then average the average speed of the link *l* as the speed of the missing link.

5. Evaluation

In order to verify the model presented in this paper, a number of experiments were implemented to prove it.

5.1 Coverage

We implemented the compensating experiment 6 times in the same way. However, each time the sample is different. The difference lies on the amount, or in another way, time period of the historical data, including 1 week, 2 weeks, 4 weeks, 8 weeks, 15 weeks and 20 weeks. Beijing road network composed by 127,541 links is used in our experiment, where 13,264 links are put in the unreachable links set L_{unre}, with the total length about 10.4% of the road network. 25,967 links with 164,726 corresponding matching trajectories of the road network are computed out through real-time FCD of the latest sampling interval. Afterwards, the historical data is used to compensate for the real-time data, and the result is shown in Table 1.

Table 1. Different historical data

Amount of data samples(in weeks)	Compensating links
1	22831
2	29904
4	35371
8	41947
15	46679
20	49087

The coverage *C* of road network is defined by the following equation:

$$C = \frac{\sum_{i=1}^{k} L_i}{\sum_{i=1}^{k} T_i} \tag{14}$$

Where *k*=1,2,3,4,5,6,8,11 describes the level of the links, L_i describes the length of the i^{th} level links which has data, and T_i describes the total length of the i^{th} level links . The variety of the coverage is shown in Figure 8.

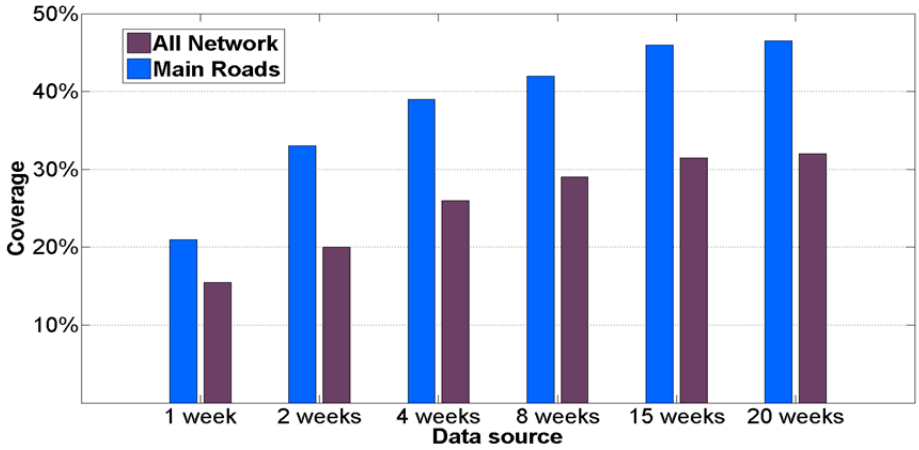

Figure 8. Coverage of the network

The result shows that the coverage of the road network has grown seriously after being compensated. 20 weeks' sample data gained about 47.6% coverage raise comparing to the real-time data. However, 15 weeks' sample data also achieved a similar result. Figure 9 shows the coverage changes of the RTIDM generated by different time period of assistant historical data.

Figure 9. Coverage Performance

Table 2 is an analysis of the performance of this model. The test environment is a PC with Pentium (R) 4 CPU 3.0GHz, 2G memory.

Table 2. Performance of TSCM

	CPU	IO
Data source generated	3740ms	5220ms
Real-time process	1690m	2470ms
Real-time compensate	2420ms	2490ms

5.2 Accuracy

We implemented the travel time measurement experiment by collecting the GPS data returned by 20 taxies. As the travel routes are fixed as planned, the process of trajectory matching can be skipped and the travel time can be calculated directly. We tested the error rate of the dynamic route guidance by a road experiment. In the experiment, we recorded the time when the vehicle passes the original location and the destination location (OD) of every route. Then we calculated the travel time of the vehicle from the system, and compared the results to the actual time we recorded. In this way, we could value the accuracy of information from the system. This equation describes the error rate of the traffic information:

$$E = \frac{|t_e - t_a|}{t_a} \tag{15}$$

Where t_e describes the travel time of some parts of the route calculated from the system, t_a describes the actual travel time we recorded.

In reference [7], it puts forward a method to compensate using the arithmetic mean, we call it as arithmetic mean method (AMM). And then we calculate the error rate using this method and our model to compare the difference.

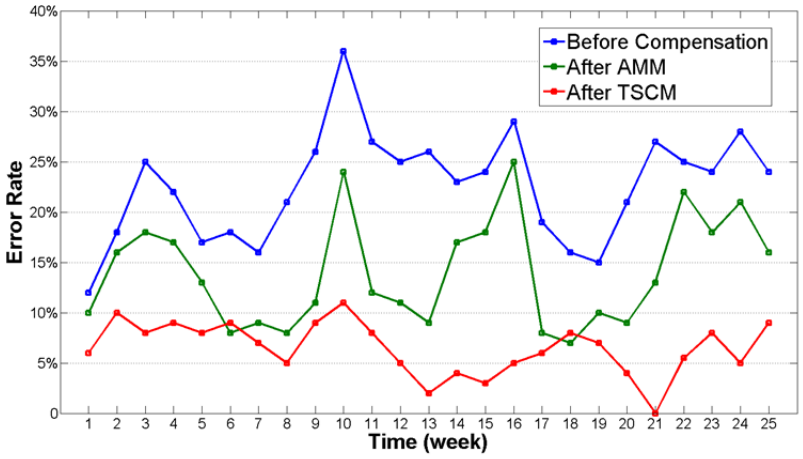

Figure 10. Error rate of the route guidance

Figure 10 shows that the great difference between the calculated travel time using the AMM method and the TSCM model and the actual travel time. The error rate of the real-time FCD is sometimes beyond 37% before using any method to compensate, but then it keeps in a steady low level when using the TSCM model. The error rate using AMM is between them. And the error rate using AMM is approaching that using TSCM at the morning peak and evening peak because there are more vehicles in that moment. So, our model can gain more accuracy in the dynamic route guidance.

In order to obtain the relative error rate, firstly, at a time point, the data of links' travel time is calculated by former system with no compensation. Then, the links whose travel time is vacant are put into a set called *L*. And each link of *L* is called a sample. After that, the system based on the TSCM is used to calculate the links' travel time at the same day and same time point, namely t_e is obtained.

In order to gain t_a, twenty floating cars are arranged to drive on the links in *L*. These floating cars are equipped with GPS devices of high precision and high sampling rate, thus, by taking these GPS data returned by these floating cars, the travel time of these links could be easily acquired, namely t_a is

obtained. After calculating t_e and t_a of all samples (namely, all links of L), the result is shown in Figure 11.

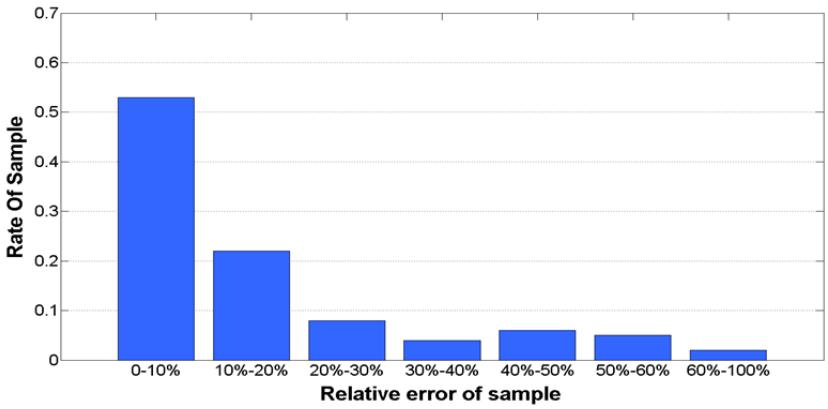

Figure 11. Relative error rate of samples

As is shown in Figure 11, about 75% of samples' relative error rate is smaller than 20%. Thus, the accuracy of the TSCM is confirmed. Moreover, there are few samples whose relative error rate is high, because of the existence of abnormal data in real-time FCD which is used to match the mode of HDPL.

6. Conclusion

The collection technology of FCD which get rapid development these years has been widely used because of its higher coverage and more accurate travel time than the traditional loop data. However, the biggest problem of FCD is that when the sample size can't be satisfied, part of the links will miss data, which can't guarantee the accuracy of the travel time calculation.

This paper presents a model (TSCM) which solved the problem of link data loss when sample amount become unstable. 1) It optimized the road network structure, classified the links which belong to different levels, and get rid of the links which don't have the ability for passing vehicles temporarily in order to raise the accuracy of the algorithm. 2) The method of selecting high-quality historical FCD as assistant data by LTP matching, can not only overcome the influence of special day, special weather and special event on the traffic trend, but also increase the matching efficiency. 3) The temporal-spatial compensation model deals with different RIT and different information missing condition separately, and this can guarantee the accuracy of compensating.

From the evaluation, the conclusion can be made that the method of compensating proposed in this paper can compensate for the blank traffic information on base of guaranteeing the processing efficiency and accuracy. The stability of data amount is guaranteed and the trajectory can also cover most of the road network which can satisfy the requirement of dynamic route guidance. On the aspect of efficiency, because of the low performance of the test PC (P4 2.8G and 2G memory) and the single-thread style of the program, the actual performance can be raised further if we change it into a better one and introduce the method of concurrent processing. At the same time, the historical data from 15 weeks can best reach the optimal efficiency and processing result, according to the evaluation.

References

[1] Wen Y. H., Lee, T. T., Cho, H. j., "Missing Data Treatment and Data Fusion Toward Travel Time

Estimation For ATIS", Journal of the Eastern Asia Society for Transportation Studies, Vol. 6, pp. 2546-2560, 2004.

[2] Jianwei Wang, Nan Zou, Gang-Len Chang, "Empirical anaylysis of missing data issues for ATIS applications: travel time prediction", the 87th meeting of the Transportation Research Board, Washington D.C., USA.

[3] Kalman, R.E. (1960). "A new approach to linear filtering and prediction problems". Journal of Basic Engineering 82 (1): 35–45.

[4] Burke, H.B., Rosen, D.B., Goodman, P.H., "Comparing artificial neural networks to other statistical methods for medical outcome prediction", Neural Networks, 1994. IEEE World Congress on Computational Intelligence., 1994 IEEE International Conference on Volume 4, 27 June-2 July 1994 Page(s):2213 - 2216 vol.4.

[5] Lymann Ott, Michael Longnecker, "An Introduction to Statistical Methods and Data Analysis", Duxbury Press, 2000.

[6] John W. Pratt, Howard Raiffa, Robert Schlaifer, "Introduction to Statistical Decision Theory", The MIT Press, 1995

[7] Dieter Pfoser, Nectaria Tryfona, Agn‘es Voisard. Dynamic Travel Time Maps - Enabling Efficient Navigation. IEEE Proceedings of the 18th International Conference on Scientific and Statistical Database Management, 2006.

[8] Georgiana L.Hamza-Lup, Kien A.Hua ,Minh Le, Rui Peng. Enhancing Intelligent Transportation Systems to Improve and Support homeland Security. 2004 IEEE Intelligent Transportation Systems Conference,Washington D.C., USA, 2004.

[9] Jixiang Sun, Modern Pattern Recognition, Higher Education Press, 2008.

[10] Coope, I.D., "Circle fitting by linear and nonlinear least squares", Journal of Optimization Theory and Applications Volume 76, Issue 2, New York: Plenum Press, February 1993.

[11] J. Wolberg, "Data Analysis Using the Method of Least Squares: Extracting the Most Information from Experiments", Springer, 2005.