

Research on Dynamic OD Analysis for Urban Traffic

DONG Hong-zhao^{1,2}, ZHOU Min^{1,2}, GUO Ming-fei^{1,2}, Qian Xiao-hong²

¹MOE Key Laboratory of Mechanical Manufacture and Automation,
Zhejiang University of Technology, Hangzhou 310014, Zhejiang Province, China

²ZJUT-ENJOYOR ITS Joint Institute, Zhejiang University of Technology,
Hangzhou 310014, Zhejiang Province, China
zhouminchinahz@gmail.com

Abstract. Traditional dynamic OD (Origin-destination) analysis is an important method of acquiring traffic information in urban traffic studies. However, just one single parameter of the traffic flow could be obtained from the traffic flow database, and it is always fractional and uncertain. A new dynamic OD analysis method based on time slice is put forward here to obtain traffic information more comprehensive than before. First, the length of time slice, a necessary factor to preprocess the dynamic OD data could be estimated in terms of historical OD matrix. Consequently, the OD data can be analyzed to get traffic information within each time slice. More traffic parameters within an interval of time such as travel time, vehicle volume etc., can be obtained by combining the results of each time slice. Finally, a practical dynamic OD analysis system is developed. The results showed the comprehensive traffic parameters can be obtained resorting to the proposed method.

1 Introduction

In recent years, the research and application of intelligent transportation systems (ITS) have spread all over the world. The dynamic self-adapting traffic management is the goal of ITS. To attain the object, time-based OD matrix is used to reflect traffic demand with various intervals (such as per hour, per minute, etc.).

The OD matrix can reflect the traffic status in the road grid effectively. The OD analysis becomes a mainstream method to obtain traffic parameters. Generally, to obtain OD matrix, there are two main methods including direct and indirect methods as shown in Fig1. ^{[1][2]}

The dynamic OD matrix has the features of time variance and uncertainty. The previous research on OD matrix is mainly focused on static OD matrix., in which traffic information between any OD pairs can be obtained by static multi-route assignment model ^[3].

Nowadays, the research is confined in the estimation and prediction of the dynamic OD matrix, which is mostly applied in the simple road grid such as highways and belt

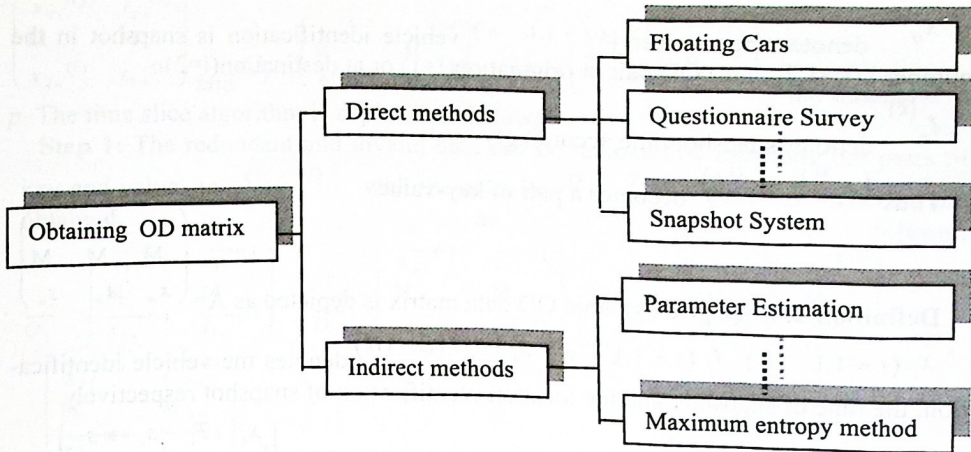


Fig. 1. The methods of getting OD matrix

highways^[4]. The methods such as the least square method^[5] and the predication of dynamic route travel time based on fuzzy judgment^[6], etc. can not suit to the complex urban road grid.

A new dynamic OD analysis method is put forward here. An interval is further divided into short time slices. In each time slice, with the calculation of vehicle data including its location and time coming from the snapshot, the traffic parameters within an interval of time can be obtained by combining each time slice. The method can effectively obtain more traffic parameters in traffic grid such as route travel time, vehicle volume, and so on.

2 Establishment of Model

Because there is the uncertainty in the dynamic OD matrix, a new strategy is proposed that the whole period is divided into small time slices. In each slice, the dynamic OD matrix is approximately considered as a static OD matrix, which is processed to acquire some parameters. The actual traffic information of whole period can be imitated through integrating the parameters in all time slices.

2.1 Definition of Concepts

Based on the strategy above, the following concepts are defined:

Definition 1: The set of OD pairs is depicted as $\{(O_k, D_k) | k = 1, 2, L, m\}$, and

the snapshot records are $O_k = \begin{pmatrix} x_{11}^{(k)} & t_{11}^{(k)} \\ M & M \\ x_{1n}^{(k)} & t_{1n}^{(k)} \end{pmatrix}$, $D_k = \begin{pmatrix} x_{21}^{(k)} & t_{21}^{(k)} \\ M & M \\ x_{2n}^{(k)} & t_{2n}^{(k)} \end{pmatrix}$.

$x_{ij}^{(k)}$ denotes the number j ($j = 1, L, n$) vehicle identification is snapshot in the number k ($k=1,2,\dots,m$) OD pair at origination($i=1$) or at destination($i=2$).

$t_{ij}^{(k)}$ denotes snapshot time accordingly.

Thus $(x_{ij}^{(k)}, t_{ij}^{(k)})$ becomes a pair of key-value.

Definition 2: Road grid dynamic OD data matrix is depicted as $A = \begin{pmatrix} x_1 & t_1 & s_1 \\ M & M & M \\ x_n & t_n & s_n \end{pmatrix}$.

x_i ($i = 1, L, n$), t_i ($i = 1, L, n$), s_i ($i = 1, L, n$) denotes the vehicle identification, the time of snapshot, and the location identification of snapshot respectively.

Definition 3: The road travel time matrix is depicted as $C = \begin{pmatrix} x_1 & \overline{T}_1 & s_i \rightarrow s_j \\ M & M & M \\ x_n & \overline{T}_n & s_k \rightarrow s_n \end{pmatrix}$,

x_i ($i = 1, L, n$) denotes the vehicle identification.

$\overline{T}_i$ ($i = 1, L, n$) denotes the time interval of the two snapshot time.

$s_i \rightarrow s_j$ ($i = 1, L, n; j = 1, L, n$) denotes the route from s_i to s_j that the same vehicle passes.

2.2 Selection of Time Slice

The time slice can be acquired by analyzing the history data of every cross-road. Because the traffic flow has a periodicity feature, the target OD pair's time slice can be obtained by matching with history OD matrix under appropriate threshold of vehicle volume^[8]. The selection of threshold is very important. If the threshold becomes larger, the precision will be decreased. If it becomes smaller, then process time spending will increase, even induces the "Runger phenomenon"—the smaller the time slice is, the larger the error becomes.

2.2.1 Time slice algorithm of two OD pairs

Considering two OD pairs--- (O_1, D_1) , (O_2, D_2) , their history matrixes of snapshot records are $O_1 = \begin{pmatrix} x_{11}^{(1)} & t_{11}^{(1)} \\ M & M \\ x_{1n}^{(1)} & t_{1n}^{(1)} \end{pmatrix}$ D_1 .

$\begin{pmatrix} x_{21}^{(1)} & t_{21}^{(1)} \\ M & M \\ x_{2n}^{(1)} & t_{2n}^{(1)} \end{pmatrix}$ and $O_2 \begin{pmatrix} x_{11}^{(2)} & t_{11}^{(2)} \\ M & M \\ x_{1n}^{(2)} & t_{1n}^{(2)} \end{pmatrix} D_2 \begin{pmatrix} x_{21}^{(2)} & t_{21}^{(2)} \\ M & M \\ x_{2n}^{(2)} & t_{2n}^{(2)} \end{pmatrix}$, the threshold is p . The time slice algorithm is depicted as follows:

Step 1: The redundant and invalid data can be removed by matching the pairs of key and value in $O_1 D_1$ with the ones in $O_2 D_2$, then new matrixes are obtained as follows:

$$\begin{aligned}
 \overline{O_1} &= \begin{pmatrix} \overline{x_{11}^{(1)}} & \overline{t_{11}^{(1)}} \\ M & M \\ \overline{x_{1m}^{(1)}} & \overline{t_{1m}^{(1)}} \end{pmatrix} \quad \overline{D_1} = \begin{pmatrix} \overline{x_{21}^{(1)}} & \overline{t_{21}^{(1)}} \\ M & M \\ \overline{x_{2n}^{(1)}} & \overline{t_{2n}^{(1)}} \end{pmatrix} \quad \text{and} \quad \overline{O_2} = \begin{pmatrix} \overline{x_{11}^{(2)}} & \overline{t_{11}^{(2)}} \\ M & M \\ \overline{x_{1h}^{(2)}} & \overline{t_{1h}^{(2)}} \end{pmatrix} \quad \overline{D_2} \\
 &= \begin{pmatrix} \overline{x_{21}^{(2)}} & \overline{t_{21}^{(2)}} \\ M & M \\ \overline{x_{2h}^{(2)}} & \overline{t_{2h}^{(2)}} \end{pmatrix}.
 \end{aligned}$$

The constraints are depicted as follows:

$$m \leq n, h \leq n$$

$$\begin{aligned}
 \overline{x_{1i}^{(1)}} &= \overline{x_{2i}^{(1)}} \in \{x_{1i}^{(1)} \mid i = 1, \dots, m\} \cap \{x_{2i}^{(1)} \mid i = 1, \dots, m\} & \overline{t_{1i}^{(1)}} &< \overline{t_{2i}^{(1)}} \\
 \overline{x_{1i}^{(2)}} &= \overline{x_{2i}^{(2)}} \in \{x_{1i}^{(2)} \mid i = 1, \dots, h\} \cap \{x_{2i}^{(2)} \mid i = 1, \dots, h\} \\
 \overline{t_{1i}^{(2)}} &< \overline{t_{2i}^{(2)}};
 \end{aligned}$$

Step 2: Assuming the threshold is p , $m \leq h$, there are three conditions below:

- if $p > h$, then the OD pair $O_1 D_1$ and $O_2 D_2$ don't satisfy the threshold condition, the algorithm stop.
- If $p \in (m, h]$, then only $O_2 D_2$ satisfies the condition. So p corresponding records can be chosen randomly from $\overline{O_2} \overline{D_2}$. The result is depicted as follows:

$$\overline{\overline{O_2}} = \begin{pmatrix} \overline{\overline{x_{11}^{(2)}}} & \overline{\overline{t_{11}^{(2)}}} \\ M & M \\ \overline{\overline{x_{1p}^{(2)}}} & \overline{\overline{t_{1p}^{(2)}}} \end{pmatrix} \quad \overline{\overline{D_2}} = \begin{pmatrix} \overline{\overline{x_{21}^{(2)}}} & \overline{\overline{t_{21}^{(2)}}} \\ M & M \\ \overline{\overline{x_{2p}^{(2)}}} & \overline{\overline{t_{2p}^{(2)}}} \end{pmatrix};$$

- If $p \leq m$, then the two pairs both satisfy the condition. So p corresponding records can be chosen randomly from $\overline{O_1} \overline{D_1} \overline{O_2} \overline{D_2}$. The result is depicted as follows:

$$\overline{\overline{O_1}} = \begin{pmatrix} \overline{\overline{x_{11}^{(1)}}} & \overline{\overline{t_{11}^{(1)}}} \\ M & M \\ \overline{\overline{x_{1p}^{(1)}}} & \overline{\overline{t_{1p}^{(1)}}} \end{pmatrix} \quad \overline{\overline{D_1}} = \begin{pmatrix} \overline{\overline{x_{21}^{(1)}}} & \overline{\overline{t_{21}^{(1)}}} \\ M & M \\ \overline{\overline{x_{2p}^{(1)}}} & \overline{\overline{t_{2p}^{(1)}}} \end{pmatrix}; \quad \overline{\overline{O_2}}$$

$$\begin{pmatrix} \overline{x_{1,1}}^{(2)} & \overline{t_{1,1}}^{(2)} \\ M & M \\ \overline{x_{1,p}}^{(2)} & \overline{t_{1,p}}^{(2)} \end{pmatrix} \overline{D_2} \begin{pmatrix} \overline{x_{2,1}}^{(2)} & \overline{t_{2,1}}^{(2)} \\ M & M \\ \overline{x_{2,p}}^{(2)} & \overline{t_{2,p}}^{(2)} \end{pmatrix}$$

Step 3: If $p \in (m, h]$, the time interval between the earliest time of $\overline{O_2}$ and the latest time of $\overline{D_2}$ can be obtained. For example, the earliest time of $\overline{O_2}$ is $t_{o_2} = \min(\overline{t_{1,i}}^{(2)})$, and the latest time of $\overline{D_2}$ is $t_{D_2} = \max(\overline{t_{2,i}}^{(2)})$ $i = 1, 2, \dots, p$, the time interval is $t_{D_2} - t_{o_2}$. So the time slice is Δt ; If $p \leq m$, then the two time interval between $\overline{O_1}$ and $\overline{D_1}$, $\overline{O_2}$ and $\overline{D_2}$ can be obtained. For example, the earliest time of $\overline{O_1}$ is $t_{o_1} = \min(\overline{t_{1,i}}^{(1)})$, the latest time of $\overline{D_1}$ is $t_{D_1} = \max(\overline{t_{2,i}}^{(1)})$; and the earliest time of $\overline{O_2}$ is $t_{o_2} = \min(\overline{t_{1,i}}^{(2)})$, the latest time of $\overline{D_2}$ is $t_{D_2} = \max(\overline{t_{2,i}}^{(2)})$ $i = 1, 2, \dots, p$, the two time intervals are respectively $t_{D_1} - t_{o_1}$ and $t_{D_2} - t_{o_2}$. To satisfy the threshold condition, $\Delta t = \max(t_{D_1} - t_{o_1}, t_{D_2} - t_{o_2})$ is selected as the time slice.

2.2.2 Time slice algorithm of multiple OD pairs

If the set of OD pairs is depicted as $\{(O_k, D_k) | k = 1, 2, L, m\} (m \geq 3)$, then the algorithm can be shown as follows:

Step 1: Because of the space similarity of traffic volume, the two neighboring OD pairs can be treated as a group: $\{(OD_k, OD_{k+1}) | k = 1, 3, L, m-1\}$ OD_k denotes (O_k, D_k)

Step 2: Call the algorithm depicted in section 2.2.1 to figure out the time slice of each group. The result is $\{\Delta t_k | k = 1, 3, L, m-1\}$;

Step 3: The time slice of the area road grid can be obtained as follows:
 $\Delta t = \sup \{\Delta t_k | k = 1, 3, L, m-1\}$

2.3 Algorithm of Travel Time

The travel time refers to the average time that the vehicles spend when passing the neighboring road crosses. The dynamic OD matrix constructed from snapshot data is divided by the time slice. The dynamic OD matrix is analyzed with static OD analysis

during each time slice. The algorithm can be shown as follows:

Step 1: The dynamic OD matrix data are filtered in each time slice Δt according to the initial time t in order to obtain the domain that the algorithm runs. For example, the OD matrix data transmitted from optical receiver is depicted as

$$A = \begin{pmatrix} x_1 & t_1 & s_1 \\ M & M & M \\ x_n & t_n & s_n \end{pmatrix} \quad \text{then the filtrated data is}$$

$$\overline{R} : \overline{R} = \begin{pmatrix} x_1 & t_1 & s_1 \\ M & M & M \\ x_m & t_m & s_m \end{pmatrix} (m \leq n, t \leq t_i \leq t + \Delta t)$$

Step 2: Because there exists redundancy in the matrix data, the data must be pre-processed with the vehicle identification matching to remove the abnormal or single data. The algorithm is depicted as follows:

- i. $x = x_1$
- ii. Traverse the record matrix $\overline{R}$ with x
- iii. If $\exists x_i = x$, add the x and x_i corresponding records into the set R , and at the same time delete those corresponding records in $\overline{R}$. If there is only one record left in $\overline{R}$, go to the step ; otherwise rearrange the records and give value x_1 to the first record, then go to step i;
- iv. If $\forall x_i \neq x$, delete the record that the x corresponds to in $\overline{R}$. If there is only one record in the $\overline{R}$, go to the step ; else rearrange the records and give value x_1 to the first record, then go to step i;
- v. The algorithm ends;

The final result is $R: R = \begin{pmatrix} x_1 & t_1 & s_1 \\ M & M & M \\ x_h & t_h & s_h \end{pmatrix} (h \leq m, \text{ the vehicle identification } x_i, i \in [1, n] \text{ corresponds to multiple records})$

Step 3: Compute the travel time from the OD matrix data obtained from step 2. If

the count of records that x_i corresponds to is k , the k records are depicted as R_i :

$$R_i = \begin{pmatrix} u & t_i & s_i \\ M & M & M \\ u & t_{i+k-1} & s_{i+k-1} \end{pmatrix} \quad \text{Then do subtraction of the time field of every two re-}$$

cords in $R_i \cdot C_k^2$ records can be got. The result is depicted as

$$C_i : C_i \begin{pmatrix} u & T_i & s_i \rightarrow s_i \\ M & M & M \\ u & T_{i+C_i^2-1} & s_{i+k-1} \rightarrow s_{i+k-1} \end{pmatrix}$$

Step 4: Filtrate C_i with the use of adjacent relationship of two road-crosses in road grid topological structure. The result is

$$\overline{C_i} : \overline{C_i} \begin{pmatrix} u & \overline{T}_1 & s_i \rightarrow s_j \\ M & M & M \\ u & \overline{T}_m & s_k \rightarrow s_n \end{pmatrix} (\overline{T}_i > 0 \quad 0 < i < m \quad s_i \text{ adjacent to } s_j \quad s_k \text{ adjacent to } s_n)$$

Step 5: Loop step 3 and 4. The travel time matrix $\overline{C_i} \quad 1 \leq i \leq h$ of every group data are obtained. The travel time matrix between two adjacent road-crosses of one vehicle can be depicted as partitioned matrix $\overline{C} : \overline{C} = (\overline{C_1} \quad \overline{C_2} \quad L \quad \overline{C_h})^T$

Step 6: The records in $\overline{C}$ are the travel time of single vehicle, the road average travel time is the weighted mean of the records in $\overline{C}$. For example, the records

the road $s_i \rightarrow s_j$ corresponds to in $\overline{C}$ are $D_i : D_i \begin{pmatrix} x_m & \overline{T}_m & s_i \rightarrow s_j \\ M & M & M \\ x_n & \overline{T}_n & s_i \rightarrow s_j \end{pmatrix}$.

Vehicle i is assigned a weight α_i according to the vehicle type x_i . The weight α_i can be determined by many methods such as statistical mean value method binomial coefficient method paired comparison method CCM analytic hierarchy process etc. Now the matrix is depicted

as $D_i : D_i \begin{pmatrix} x_m & \overline{T}_m & s_i \rightarrow s_j & \alpha_m \\ M & M & M & M \\ x_n & \overline{T}_n & s_i \rightarrow s_j & \alpha_n \end{pmatrix}$. The road average travel time of $s_i \rightarrow s_j$ is $\overline{\overline{T}}_{ij} : \overline{\overline{T}}_{ij} = \sum_{k=m}^n \overline{T}_k \alpha_k$. The matrix D_i only includes the information of road and

travel average time. The final result C is combined from $D_i : C = (D_1 \quad D_2 \quad L \quad D_h)^T$

3 Application

This research has been applied in the intelligent traffic guidance information platform of Hangzhou city China. The system's main functions are include the inquiry of traffic flow, Graphic inquiry and Exportation of excel.

3.1 Running Procedure of the System

3.1.1 Computation of Time Slice

In this part, two road segments are taken for example which are from Yan'an-Qingcun roadcross to Qingcun-Zhonghe roadcross and from Linying-Shuguang roadcross to Huanglong-Shuguang roadcross respectively. The OD matrix data are sampled from the four roadcrosses above. The time segment is from 10:10:00 a.m., Jan 3th, 2006 to 10:20:00 a.m., Jan 3th, 2006. The threshold is 10 vehicles. 10 records are selected from those samples randomly. The matching results are shown in Table 1 and Table 2:

From the two tables above, the earliest time of 10 vehicles passed Yan'an-Qingcun roadcross is $t_{a_1}=10:12:11$, the latest time of Qingcun-Zhonghe roadcross is $t_{b_1}=10:16:24$. Then the time interval is $t_1 - t_{b_1} - t_{a_1}=4$ minutes and 13 seconds.

Similarity, the earliest time of 10 vehicles passed Linying-Shuguang roadcross is $t_{a_1}=10:10:48$, the latest time of Huanglong-Shuguang roadcross is $t_{b_1}=10:15:48$. The time interval is $t_1 - t_{b_1} - t_{a_1}=5$ minutes.

According to the algorithm above, time slice is $\Delta t: \Delta t = \max t_1 - t_2 = 5 \text{ min}$.

3.1.2 Computation of Travel Time

With the time slice Δt , we can obtain the travel time from Yan'an-Qingcun roadcross to Qingcun-Zhonghe roadcross in the time between 10:10:00 a.m., Jan 10th, 2006 and 10:20:00 a.m., Jan.10th, 2006.

First, the data are further filtrated according to the time slice and system trigger time at which the system starts to run. For example, if the system trigger time is 10:17:00 a.m. Jan. 10th,2006, then the processing period is from 10:12:00 a.m. Jan 10th,2006 to 10:17:00 a.m. Jan. 10th,2006. Second, every two of the filtered data make subtraction with each other. Then the mean value can be acquired (the vehicle type of the vehicles in this sample are considered the same). Table 3 shows process.

The time intervals of each sample are as follows: $\overline{T_1}=1\text{min } 40\text{sec}$ $\overline{T_2}=2 \text{ min } 57 \text{ sec}$ $\overline{T_3}=2 \text{ min } 44 \text{ sec}$ $\overline{T_4}=3 \text{ min } 20 \text{ sec}$ $\overline{T_5}=2 \text{ min } 29 \text{ sec}$ $\overline{T_6}=2\text{min } 52\text{sec}$ $\overline{T_7}=3\text{min}4\text{sec}$ $\overline{T_8}=2\text{min}41\text{sec}$ $\overline{T_9}=2 \text{ min } 52 \text{ sec}$ $\overline{T_{10}}=2 \text{ min } 06 \text{ sec}$ $\overline{T_{11}}=2 \text{ min } 38 \text{ sec}$. The weight is $\alpha_k: \alpha_k=1/11$ $k=1 \ 2 \ \dots \ 11$. And finally

the average travel time is depicted as $\bar{T} = \frac{\sum_{i=1}^{11} \bar{T}_i \alpha_i}{11} = 2 \text{ min } 40 \text{ sec}$ with the algorithm above.

Table 1. The 10 samples randomly obtained from Yan'an-Qingcun No.1 and Qingcun-Zhonghe(No2) roadcrosses

Samples	1	2	3	4	5	6	7	8	9	10	Extreme early/late time	Interval
Vehicle ID	A36BXX	A3T6XX	A56JXX	A56UXX	B7E4XX	AT79XX	A70XXX	AT78XX	AT5XX	B33IXX		
No.1	10:12:37	10:13:20	10:14:20	10:13:12	10:13:22	10:12:11	10:12:30	10:14:20	10:12:54	10:12:34	10:12:11	4min13sec
No.2	10:14:39	10:16:07	10:16:20	10:16:03	10:16:24	10:15:55	10:15:55	10:16:10	10:14:59	10:15:51	10:16:24	

Table 2. The 10 samples randomly obtained from Linyin-Shuguang(No.1) and Huanglong-Shuguang(No.2) roadcrosses

Samples	1	2	3	4	5	6	7	8	9	10	Extreme early/late time	Interval
Vehicle ID	A456XX	AT36XX	A54YXX	B674XX	B454XX	AT56XX	AT77IX	AT9P3XX	A165XX	A6HJXX		
No.1	10:11:26	10:11:20	10:14:20	10:10:48	10:11:20	10:10:37	10:11:20	10:11:15	10:13:20	10:14:20	10:10:48	5min
No.2	10:13:37	10:14:13	10:15:42	10:13:37	10:14:55	10:13:18	10:14:20	10:14:11	10:14:10	10:15:48	10:15:48	

Table 3. The computation of travel time

Samples	1	2	3	4	5	6	7	8	9	10	11	Time mean value
Vehicle ID	AT28XX	A471XX	A13XX	A53XX	A57QXX	AT63XX	B23XX	B67XX	A546XX	B465XX	AT10XX	
No.1	10:14:35	10:12:24	10:13:55	10:13:11	10:14:11	10:12:45	10:12:12	10:14:42	10:13:56	10:14:20	10:13:20	
No.2	10:16:15	10:15:21	10:16:39	10:16:31	10:16:40	10:15:37	10:15:26	10:16:23	10:16:48	10:16:26	10:15:58	
Time interval	1min40sec	2min57sec	2min44sec	3min20sec	2min29sec	2min52sec	3min41sec	2min41sec	2min52sec	2min06sec	2min38sec	2min40sec

3.2 Exhibition of Results

The graphic inquiry is a distinguishing function of the system, which is shown in Fig 2(in Chinese). In the query interface, the names of roadcrosses that we concern are input by clicking the location on the map. They are displayed in the two text boxes on the first line. The inquiry period between the input start time and the end time are displayed in the two text boxes on the second line. After the requisite information is all input, the road travel time will be shown as in Fig. 3(in Chinese). The dynamic OD analysis is a basic and effective method of acquiring traffic parameters of road grid. To reduce the complexity and inaccuracy in traditional OD analysis, an improved dynamic OD analysis is put forward in this article which provides an effective approach to obtain the traffic parameters.

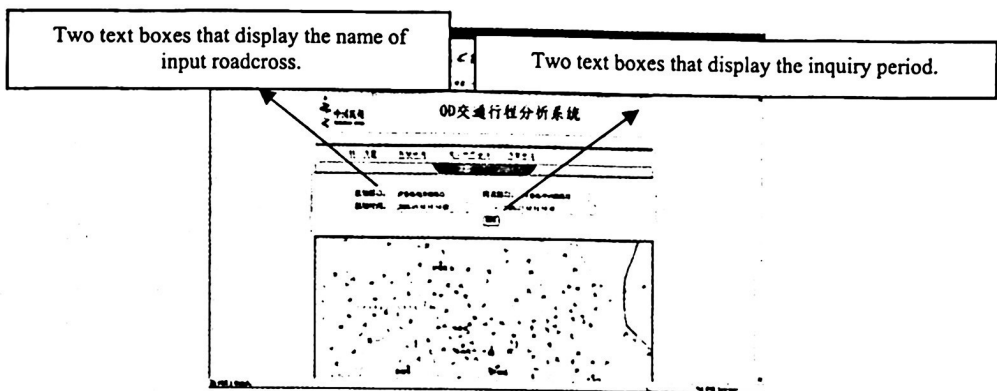


Fig. 2. The interface of graphic input

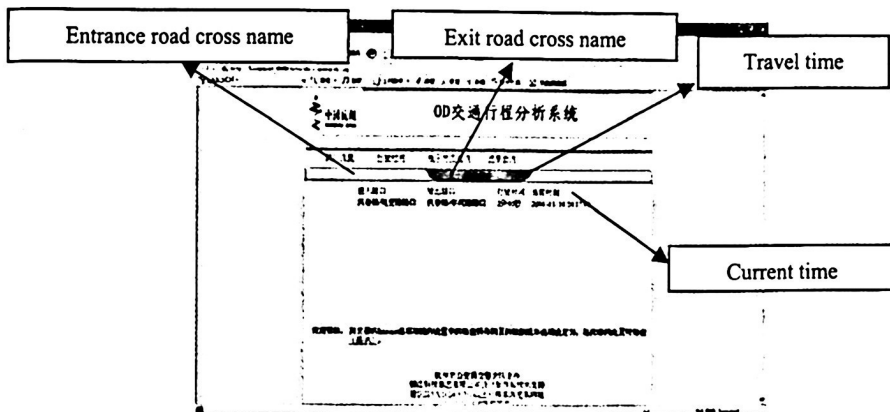


Fig. 3. Interface of Query Result

The approach has been put into practice in the intelligent traffic guidance information platform of Hangzhou, Zhejiang province, China. Results show that the method can provide travel time correctly and effectively.

Acknowledgment

This research was supported by the Important Project for Key Subject of Zhejiang Province, China 2006C13100 and the Projects of the Key Subject Open Fund of Zhejiang Province, China AMT200506-018. The authors would like to thank Ms. WEN Xiao-yue and Mr. XU Jian-jun of the ZJUT-ENJOYOR ITS Joint Institute for their kind assistance in the research.

References

1. Zhenguo Zhou, Dynamic. OD Matrix Estimate and Utilization Research[D]. Master's thesis of University Southeast, 2001.
2. Chen Chaogang, Ma Zhu. The Improvement for The OD Investigation of Road Vehicle[J]. EAST CHINA HIGHWAY, 2003, 140: 69-73.
3. Luping Pan, Jingchang Chen. The Application of Static Mul-route Assignment Model on Freeway Plan[J]. Journal of Engineering Graphics, 2003, 3: 37-42.
4. Taifeng Ma. Traffic Control and Vehicle Route Guidance in Intelligent Transport Systems[D]. Doctor's thesis of Tianjin University, 1999.
5. Yang Shao-hui, Wang Dai-hai, Wang Ying-ping, Dong Bin. Fitting Travel Time by Least Square Method[J]. Journal of Highway and Transportation Research and Development, 2006, 9(23): 90-93.
6. Wensheng Zhang, Qian Yin, Lixin Wu, Lei Xu, Zhigang Zang. The Prediction Model of Dynamic Rode Running Time Based on FCE[J]. Geography and Geo-Information Science, 2006, 22(4): 25-27.
7. Cai Yonghua. The Research of OD Matrix in ITS[D]. Master's thesis of wuhan university of technology, 2005.
8. Yong Lin, Yuanli Cai, Yongxuan Huang. Dynamic Origin-destination Matrix Estimation for Freeways[J]. Journal of Chang'an University, 2003, 23(6): 83-86.
9. Haihu Wang, Zhixiang Xu. Intelligent Traffic Monitoring and Measuring[J]. TV Applications, 2000.
10. M. Bierlaire, F. Crittin. An Efficient Algorithm for Real-Time Estimation and Prediction of Dynamic OD Tables[J]. Transportation Science EPFL, 2004, 1(52).
11. Alan L. Erera, Carlos F. Daganzo, David J. Lovell. The Access-Control Problem on Capacitated FIFO Networks With Unique O-D Paths is Hard. Transportation Science[J], 2002, 4(50): 736-743.