

# Discovering Traffic Bottlenecks in an Urban Network by Spatiotemporal Data Mining on Location-Based Services

Wei-Hsun Lee, *Member, IEEE*, Shian-Shyong Tseng, *Member, IEEE*, Jin-Lih Shieh, and Hsiao-Han Chen

**Abstract**—Discovering traffic bottlenecks and taking action to alleviate congestion to enhance the performance of a traffic network are the most important tasks for the advanced traffic management system in the intelligent transportation system. However, traffic bottlenecks are affected by several factors and vary with spatial and temporal environments, which makes them difficult to define and discover. This paper proposes a three-phase *spatiotemporal traffic bottleneck mining* (STBM) model, including several spatiotemporal traffic patterns and STBM algorithms that use the raw data of location-based services to discover urban network spatiotemporal traffic bottlenecks. This paper implements an STBM prototype system based on a taxi dispatching system in a Taipei, Taiwan, urban network. The experimental results show that the congestion prediction capability of the proposed heuristic methods (congestion-propagation heuristic) is up to 79.6% during workdays and 72.1% on weekends, which outperforms other methods (e.g., the congestion-converge heuristic, the congestion-drop heuristic, and congested object item), and the discovered spatiotemporal bottlenecks match the travelers' experience.

**Index Terms**—Advanced traffic management system (ATMS), spatiotemporal data mining, spatiotemporal traffic patterns (STPs), traffic network bottleneck.

## I. INTRODUCTION

**T**RAFFIC congestion has been increasing worldwide due to increased motorization, urbanization, population growth, and changes in population density, particularly in urban networks. Congestion reduces the utilization of the transportation infrastructure and increases travel time, air pollution, and fuel consumption, which may cause various social, environmental, and economic problems. Most instances of congestion mainly

result from traffic bottlenecks; therefore, locating traffic bottlenecks and taking appropriate actions to alleviate congestion and improve the performance of the traffic network are goals of the advanced traffic management system (ATMS) in the intelligent transportation system (ITS). Traffic assignments such as traffic signal control, the changeable message sign (CMS), and the reversible lane can be exercised to alleviate congestions and enhance the traffic network performance. However, traffic bottlenecks in urban networks are difficult to define and discover, because they vary with spatial and temporal environments. For example, traffic bottlenecks on workdays differ from traffic bottlenecks on weekends, and traffic bottlenecks during A.M. peak hours are not the same as traffic bottlenecks during P.M. peak hours.

Traffic network comprises a set of network objects, each of which is either a link or an intersection where traffic congestions occur only on some network objects, in which the traffic demands cannot be fully serviced. Thus, a traffic bottleneck can be defined as a traffic network object where and when its capacity is less than the traffic demand. Traffic bottlenecks may be the root cause of neighborhood congestions; for example, a traffic bottleneck may generate a queue of vehicles that propagate to surrounding network objects, block intersections, and result in congestion chaining. However, traffic demands vary according to spatial and temporal environments, as do traffic bottlenecks. In this paper, we define spatiotemporal traffic bottleneck (STB) instead of traffic bottleneck to clearly identify where and when traffic bottlenecks could occur. An STB is a traffic bottleneck with spatial and temporal identification, which indicates when and where a network object could become the traffic bottleneck, may result from overloaded traffic demand and may be the root cause of related neighborhood congestion. Once an STB has been discovered, some traffic assignment actions such as traffic signal control, manual intersection control, reversible lane, or congestion message broadcasting by CMS can be exercised to relieve the congestion and enhance the network performance.

A spatiotemporal traffic pattern (STP) is a distribution of traffic-flow variables in space and time [1]. In the literature, several studies have been conducted to discuss or discover STPs [1]–[5] or to identify traffic states [6], [7] in the traffic network, and some researches worked on predicting the travel time to provide drivers with a route suggestion [8], [9] or detecting traffic incidents [10]. Kerner [1], [2] developed a three-phase traffic theory to discuss empirical spatiotemporal traffic features and characteristics that are applied to traffic

Manuscript received December 20, 2008; revised October 16, 2009, August 27, 2010, and February 22, 2011; accepted March 12, 2011. Date of publication May 12, 2011; date of current version December 5, 2011. The Associate Editor for this paper was H. Dia.

W.-H. Lee is with the Department of Transportation and Communication Management Science, National Cheng Kung University, Tainan 701, Taiwan (e-mail: leews@mail.ncku.edu.tw).

S.-S. Tseng is with the Department of Applied Informatics and Multimedia, Asia University, Taichung 413, Taiwan (e-mail: sstseng@asia.edu.tw).

J.-L. Shieh is with the Department of Management Information System, Far East University, Tainan 744, Taiwan, and also with the Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan 701, Taiwan (e-mail: shiehjl@cc.feu.edu.tw).

H.-H. Chen is with the Telecommunications Laboratory, Chunghwa Telecom Company Ltd., Taoyuan 326, Taiwan (e-mail: qemily@cht.com.tw).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TITS.2011.2144586

control and management. In [6], Kerner *et al.* proposed the floating-car data (FCD) method to recognize traffic states (e.g., congested or not) by FCD vehicles in urban networks but could not identify the bottleneck locations. Previous studies have tried to locate and control congestion patterns on highway/freeway bottlenecks [1], [2], [11], [12], which are typically static and clear, located near on- and off-ramps and decreasing lanes, or affected by road curves and road gradients [1]. However, locating STBs in urban networks is more difficult than on freeways because of an increased number of factors to consider, e.g., intersections, traffic signal controls, and network complexity. Analyzing traffic patterns in urban networks and discovering traffic bottlenecks is a complex and difficult task. First, the traffic network in urban areas is more complex than the traffic network on freeways or simple arterial networks. Second, traffic bottlenecks in urban networks are spatiotemporal dynamic and vary with traffic demands. Third, traffic factors and more nontraffic factors such as traffic signals, social events, and traffic incidents have to be considered more in urban networks than in freeways. In 2008, Long *et al.* attempted to recognize urban traffic congestion propagation and identify bottlenecks using the cell transmission model (CTM) [13], which discretizes each roadway into homogeneous sections (cells) and discretizes time into intervals. By providing network object capabilities, it tries to identify network bottlenecks by simulating the traffic demands of urban networks.

Location-based services (LBSs) provide appropriate location-aware information for users by exchanging location and related information between in-vehicle front-end devices and back-end systems. Communication raw data in LBS-based applications contain traffic network information to mine and reuse. The basic idea of this paper is to reuse raw data in LBS-based applications, transform these data into traffic information, and apply the spatiotemporal data-mining technique to generate traffic management knowledge, including STPs and STBs. The collected vehicle track data serve as traffic sensor data, and vehicles in the LBS-based applications function as traffic network probing vehicles. Compared with the traditional sensor-based traffic surveillance system, it is cost effective because of deriving traffic information by mining the raw data of LBS-based applications. Moreover, it has spatiotemporal coverage advantages due to the dynamic collection of traffic information in the LBS operation area 24 h per day in real time. By analyzing and mining the raw data, the network traffic status and vehicle journeys information can be derived. In this paper, several STPs are defined and discovered, and three STB discovering heuristics are proposed to discover STBs by analyzing STPs.

Common LBS-based applications comprise the following three system components: 1) an onboard unit (OBU); 2) a mobile network; and 3) a back-end system. An OBU is a small computer (or a smart phone) installed on the vehicle and integrates computing, positioning, communication, and human interface modules. The system locates the vehicle by Global Positioning System (GPS) signals, exchanges messages to and from the back-end system through a mobile network, and interacts with drivers through the human interface module. This paper applies the taxi-dispatching system (TDS) [14]–

[17], one of the most complicated LBS-based applications, as the data source, because the traffic information included in TDS is considered more plentiful than in other LBS-based applications, particularly in urban networks. The OBU automatically registers to the back-end dispatching center and reports the position, moving direction, speed, and status of the taxi according to predefined rules embedded in the OBU. The dispatching center maintains the status and positions of all vehicles according to the collected uplink packets of OBUs. TDS [14], applied in a Taipei urban network, has the following three uplink rules embedded in the OBU: 1) a periodical report (fixed-time interval); 2) a cross-boundary report (a vehicle that passes through the virtual geographical boundary); and 3) an event report (status changing or event triggering). The OBU automatically turns into the “dispatched” state when the taxi is dispatched, switches to the “occupied” state when passengers get into the vehicle, and turns into the “available” state when passengers get out of the vehicle. By decoding and analyzing the uplink packets and state transitions, the traffic status and traffic journey with origin–destination (O–D) information can be derived. This procedure utilizes all the vehicles in LBS-based applications as traffic probing vehicles for the traffic network.

Each uplink packet can be transformed into a traffic information spot (TIS), because the information that it contains represents a local traffic status according to the status of the vehicle, including the coordinates, moving speed, moving direction, and state. By integrating the road network database with the geographical information system (GIS), the GPS coordinate (latitude and longitude) of a vehicle can be interpolated to the nearest address [18]. Thus, traffic information can be derived by transforming the uplink packets into TISs. As illustrated in (1), a TIS  $S_k(O_i^j, V, D)$  transformed from the OBU uplink packet  $U_k$  of a vehicle (with status  $S$ ) consists of the object ID  $O_i^j$ , speed  $V$ , and moving direction  $D$  of a vehicle when it communicates with the LBS back-end system at time  $t$  and location coordinate  $(X, Y)$ , where  $O_i^j$  is a spatiotemporal network object spatially indexed by network object  $i$  (transformed from the location to address interpolation) and temporally indexed by time zone  $j$  (transformed from timestamp  $t$ )

$$U_k(X, Y, t, V, D, S) \xrightarrow{GIS} S_k(O_i^j, V, D). \quad (1)$$

Real-time traffic information of the urban network can be derived by aggregating all the collected TISs at a time interval, e.g., 15 min. In addition to TIS, which indicates the traffic status at one fixed point, a journey, which represents the tracks of a vehicle (starting from the origin to its destination) and partially reflects the traffic demand in the urban network, is a collection of consecutive TISs of a vehicle. For example, the “dispatched” state journey in TDS consists of a set of TISs, which starts from the dispatched location to the customer location, and the “occupied” state journey starts from the passengers’ location to their destination. A vehicle journey is defined as a set of consecutive  $n$  TISs collected from a vehicle, which reported its journey from origin  $S_1$  to destination  $S_n$ , i.e.,

$$J_k = \langle S_1, \dots, S_n \rangle. \quad (2)$$

A traffic network snapshot (TNS) provides a global view of the traffic status in a time period for the traffic network indexed by spatial and temporal dimensions. The spatial domain groups the TISs by the spatial area of the network objects (e.g., link), and the temporal domain groups the TISs by a time zone (e.g., 15 min). Let a TNS be composed of a set of a spatiotemporal network objects during a 15-min period. The traffic status during the workday A.M. peak hour (7~9 A.M.) includes eight snapshots. All the traffic information generated in this phase, including TISs, journeys, and TNSs, is stored in a traffic information database (TIDB) as the data source of subsequent data-mining processes. TNS can easily be presented by the map-based user interface to reflect a short period of network status. With the knowledge of STB, global network traffic status, and continuous-traffic-status variations presented by continuous TNSs, the network administrators can decide which traffic assignment actions are the best actions by their domain expertise.

The contributions of this paper are listed as follows.

- 1) A spatiotemporal traffic bottleneck mining (STBM) model is proposed, which defines STB and develops three heuristic methods to discover the STBs in an urban network.
- 2) Several hypothesis-based spatiotemporal data-mining methods are proposed to determine spatiotemporal traffic congestion patterns by reusing the communication raw data in LBS-based applications.
- 3) Two algorithms are developed, in which the congestion propagation pattern (CPP) discovering algorithm can be applied to predict the congestion area and thus contribute to ATMS.
- 4) The proposed STBM model can catch details of traffic information, including points (TISs), lines (traffic journeys), and planes (TNSs), and analyze the spatiotemporal relationships and features of the collected traffic information.
- 5) Traffic information transformation from LBS-based applications has cost and coverage advantages compared to traditional traffic surveillance systems, and the collected real-time traffic information and generated traffic knowledge (STP/STB) provides much knowledge, which can widely be applied to several ITS applications, e.g., an advanced traveler information system (ATIS), travel-time prediction, and travel routing path suggestion.

The rest of this paper is organized as follows. Section II proposes the three-phase STBM model for discovering STBs and discusses the traffic information generated from raw data of LBS-based applications (phase I). Section III defines and mines several traffic-congestion-related patterns from the TIDB (phase II). Section IV discusses, in detail, the proposed three STBM methods using different heuristics (phase III). We employed STBM to discover STBs in Taipei City, Taiwan, and compared the proposed three methods with the statistical method in Section V. Section VI offers the concluding remarks and future works.

TABLE I  
LIST OF ABBREVIATIONS IN STBM

CASR	Congestion Area Sequence Rule
CCP	Cross Conflict Pattern
CDP	Congestion Drop Pattern
CCH	Congestion Converge Heuristic
CDH	Congestion Drop Heuristic
CDP	Congestion Drop Pattern
CPH	Congestion Propagation Heuristic
COI	Congested Object Item
CPP	Congestion Propagation Pattern
DCP	Demand Conflict Pattern
DOR	Demand Overlap Ratio ( $\sigma$ )
IDP	Intersection Delay Pattern
SHC	Spatial Heuristic Clustering
STB	Spatiotemporal Traffic Bottleneck
STCA	Spatiotemporal Congested Area
STCO	Spatiotemporal Congested Object
STP	Spatiotemporal Traffic Pattern
TDS	Taxi Dispatching System
TIDB	Traffic Information Database
TIS	Traffic Information Spot
TNS	Traffic Network Snapshot

TABLE II  
LIST OF KEY VARIABLES USED IN STBM

$\tau$	Congestion Drop Ratio
$\lambda$	definition of ‘passing by’ concept of a journey ( $J_m$ ) and a STCA( $A_k^l$ )
$\sigma$	definition of demand overlap ratio of two STCAs by calculating the overlapped journeys which passes by these two STACs
$\theta(O_i^j)$	congestion factor: normalized traffic status of spatiotemporal object $O_i^j$
$A_i^j$	the $i$ -th STCA in snapshot $N_j$
$H_d$	congestion drop threshold
$H_u^s$	spatial boundary threshold
$H_u^d$	demand overlap ratio threshold
$J_m$	a journey indexed by $m$
$N_j$	Snapshot indexed by temporal index $j$
$O_i^j$	a spatiotemporal network object indexed by network object ID $i$ and time zone ID $j$
$T_k$	Time zone indexed by $k$
$S_k$	a TIS indexed by $k$

## II. SPATIOTEMPORAL BOTTLENECK MINING MODEL

To facilitate the model presentation, the abbreviations and notations used hereafter are summarized in Tables I and II, respectively. The proposed STBM model, as illustrated in Fig. 1, is a data-mining process that comprises the following three phases, where raw data in LBS-based applications and urban networks database in GIS are the two major data sources: 1) traffic information transformation and vehicle journey generation; 2) traffic congestion pattern recognition; and 3) STBM. The first phase collects, cleans, and transforms raw data in LBS-based applications into TISs and traffic journeys by identifying the vehicle ID and location and integrating the urban road network database in GIS. The generated traffic information, including the TIS, vehicle journey, and TNS, are then stored in the TIDB as sources for spatiotemporal data mining. The second phase categorizes several STPs by object- and area-level patterns, which are mined from the TIDB by a series of



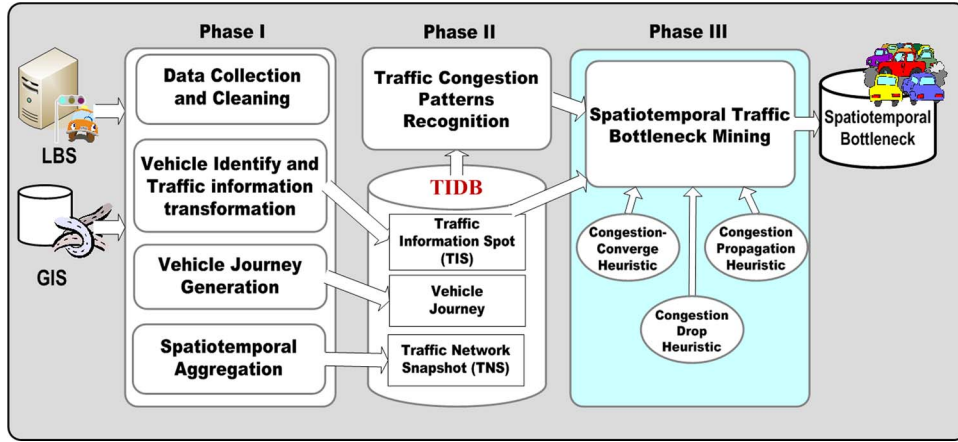


Fig. 1. STBM model.

hypothesis-based data-mining processes. Object-level patterns identify the features of congested network objects, which can directly be transformed into a STB. Area-level patterns identify the traffic demands, which will be analyzed and transformed into a STB by the proposed heuristic methods, as discussed in detail in phase III.

#### A. Phase I: Traffic Information Transformation and Spatiotemporal Congested Object (STCO)

Phase I comprises the following four modules, as illustrated in Fig. 1:

- 1) data collection and cleaning;
- 2) vehicle identification and traffic information transformation;
- 3) vehicle journey generation;
- 4) spatiotemporal modules.

Data collection is a batch process that periodically collects the communication log between front-end devices and the back-end system in the TDS and employs the data-cleaning module to filter out useless raw data, e.g., data that are not in the interested traffic network area, vehicles not in the moving status, incorrect GPS states, incorrect speed data, or incomplete data. The traffic information transformation module transforms the cleaned data into TISs by integrating the urban road network database in GIS, as illustrated in (1). The vehicle journey generation module extracts meaningful journeys from a vehicle (taxi) that is in the “dispatched” or “occupied” state. The spatiotemporal aggregation module summarizes and aggregates all the TISs by the spatial (link/intersection) and temporal domains; each temporal period view of the aggregation result of the traffic network is regarded as a TNS for that temporal period.

A network object is in a congested status when the traffic demand of the object cannot be served due to its capacity. We assume that the congestion level of a network object could be decided by the ratio of traffic demand and the capacity of the object. Unfortunately, the demand of network objects is unknown, and the capacity of network objects is unavailable, which makes the identification and prediction of network object congestion impracticable. According to [19] and [20], the ratio

of the average speed and the speed limit of a network object has a negative impact on the ratio of the demand and its capacity. The lower the former ratio value is, the greater the congestion of the network object becomes. Thus, the traffic status of a spatiotemporal object (STO; denoted by  $O_i^j$ , network object  $i$  in time zone  $T_j$ ) can be indicated by dividing the average speed of all the TISs within the same time zone  $T_j$  by the speed limit of the network object. Let congestion factor  $\theta(O_i^j)$  denote the normalized traffic status of the spatiotemporal network object  $O_i^j$ , as defined in (3), where  $\bar{V}$ ,  $L_i$  are the average speed and speed limit of the object  $O_i^j$ , respectively. The higher the  $\theta$  is, the greater will become the congestion level of the object. For example,  $\theta$  around 1 indicates that the object is in a serious congestion status, and  $\theta$  around zero ( $\theta$  may be a negative value, because  $\bar{V}$  may be larger than  $L$ , for example, in the midnight) indicates that the object is in a free-flow status. We have

$$\theta(O_i^j) = 1 - \frac{\bar{V}_i^j}{L_i}. \quad (3)$$

An object is defined as a STCO if  $\theta$  is higher than the congestion threshold ( $H_\theta$ ). The traffic status of an urban network can be in a snapshot by aggregating the network objects status in the spatial and temporal domains, where the spatial domain groups the TISs by the spatial area of the network object, and the temporal domain groups the TISs by time zone, e.g., 15 min. Let a TNS be composed of a set of spatiotemporal network objects during a 15-min period; therefore, the traffic status of an urban network during workday A.M. peak hours (7–9 A.M.) includes eight TNSs.

#### B. STCA and the Spatial Heuristic Clustering (SHC) Algorithm

Because the granularity of the object-level pattern is very small for global traffic network considerations, it is difficult to reflect the statistic meaning of O–D traffic demands. A set of STCOs that are in the neighborhood area in the same snapshot can be clustered into a spatiotemporal clustered area (STCA) to solve this problem so that area-level traffic patterns can be discovered to realize the relationship between O–D traffic demands and spatiotemporal congestions.

**Algorithm 1 : Spatial Heuristic Clustering (SHC)****Denotation :** $N_j$ : the snapshot in temporal period  $T_j$  $\langle O_1^j, O_2^j, \dots, O_k^j \rangle$ : top  $k$  congested STCOs in  $N_j$  $C_i^j$ : the  $i$ -th cluster in Snapshot  $N_j$  $C_1^j = \{\text{STCO with } \text{Max } \theta(O_i^j)\}$  $H_\theta$ : the congestion threshold of  $\theta$ **Input :** Snapshot  $N_j, i=1$ **Output :** Cluster Set  $\{C_1^j, C_2^j, C_3^j, \dots\}$ **Step1: Initialize Clusters**Create a new cluster  $C_i^j$  and add the STCO into  $C_i^j$ **Step2: Cluster union by spatial heuristic**For each object  $O_x$  in new cluster  $C_i^j$ For each connected objects  $O_y$  (by querying GIS)If ( $O_y$  does not belong to any cluster)and ( $\theta(O_y^j) > H_\theta$ )

then

$$C_i^j = C_i^j \cup \{O_y\}$$

Recursively iterate the union process of new object in  $C_i^j$ **Step3: Start new cluster** $i++$ If  $i < k$  then goto **Step 1**

else return the cluster set

Fig. 2. SHC algorithm.

We developed a SHC algorithm, as illustrated in Fig. 2, for the spatiotemporal clustering of neighborhood STCOs. In the SHC algorithm, the top  $k$  congested STCOs in the TNS  $N_j$  are selected as seeds of the candidate clusters. Each cluster initially searches and joins the neighborhood congested objects close to the seed by querying the traffic network database in GIS and recursively searches the neighborhood congested STCOs for the new member of the cluster until there is no neighborhood STCO or until the threshold is reached.

### III. PHASE II: SPATIOTEMPORAL TRAFFIC CONGESTION PATTERNS RECOGNITION

TISs, traffic journeys, and TNSs that were generated in phase I are stored in the TIDB as the spatiotemporal data-mining source for phases II and III. Several STPs that are categorized by object- and area-level patterns are defined and discovered from the TIDB in this phase, e.g., object-level patterns, including congested object item (COI), congestion drop pattern (CDP), and intersection delay pattern (IDP), and area-level patterns, including CPP and demand conflict pattern (DCP).

#### A. COI

The COI pattern indicates when and where a network object is in a congested status, and it fulfills the following two criteria: 1) The congestion factor  $\theta$  of the object is higher than the traffic congestion threshold  $H_\theta$ , and 2) the congestion confidence of the object is higher than the congestion confidence threshold

$H_c$ . Temporal periods (identified by  $P_k$ ) are the temporal sections with specific traffic characteristics in a week, e.g., A.M. or P.M. peak hours of the workday, normal hours, free-flow hours, and holiday hours. Some traffic regulation actions can be taken to alleviate the COI congestion, e.g., traffic signal control, enforcing the reversible lane, and manual traffic regulation. For the STCOs obtained in phase I, the congestion confidence threshold filter is then applied to each STCO to discover the COI pattern, as illustrated in (4). The confidence of network object  $O_i^j$  at temporal period  $P_k$  is defined by the ratio of the congested samples of the spatiotemporal network object  $O_i^j$  to all samples in the temporal period  $P_k$

$$\text{Conf}(O_i^j, k) = \frac{|\{O_i^j | T_j \in P_k, \theta(O_i^j) > H_\theta\}|}{|\{O_i^j | T_j \in P_k\}|}. \quad (4)$$

#### B. CDP

The basic idea of CDP is to calculate the significant congestion difference between a network object and its downstream objects, which indicates the bottleneck level of the object. The definition of congestion drop ratio function (denoted as  $\tau$ ) is illustrated in (5), which calculates the difference of congestion factor  $\theta$  between object  $O_i^j$  and the average congestion factor  $\theta$  of its  $m$  downstream objects ( $\{O_1^j, O_2^j, \dots, O_m^j\}$ )

$$\tau(O_i^j) = \theta(O_i^j) - \frac{\sum_{k=1}^m \theta(O_k^j)}{m} \quad (5)$$

where  $\{O_1^j, O_2^j, \dots, O_m^j\}$  is the set of the downstream objects of  $O_i^j$ .

If the congestion drop ratio  $\tau$  of an object approximates 1, it indicates that the traffic congestion of the object is more serious than its downstream objects. The traffic congestion is more serious in downstream objects than the object if the  $\tau$  of the object is smaller than 0. To discover the CDP in a TNS, the STBM model calculates the  $\tau(O_i^j)$  of all the STCOs, where an object is regarded as a CDP object when its  $\tau(O_i^j)$  is larger than the congestion drop threshold  $H_d$ .

#### C. IDP

Intersection delay is the delay between two consecutive links, primarily caused by signal and queuing delays. There are three types of intersection delays, which indicate three possible moving directions from one link connected to its downstream links, respectively: 1) through delay (TD); 2) right-turn delay (RTD); and 3) left-turn delay (LTD). The IDP samples are retrieved from two consecutive TISs of a journey with different links by calculating the link travel time and intersection delay. The IDP pattern is presented in the form of [IDP: ( $P, SO_{id}, SI_{id}, T_{id}, D_{avg}, S_{up}, C_{onf}$ )], where  $P$  is the pattern type (TD/LTD/RTD),  $SO_{id}$  and  $SI_{id}$  are the two consecutive link IDs, which indicate that the vehicle leaves from link  $SO_{id}$  and goes into link  $SI_{id}$ ,  $T_{id}$  is the temporal ID,  $D_{avg}$  denotes the average delay time of this IDP, and  $S_{up}$ ,  $C_{onf}$  are the support and confidence of the pattern, respectively.

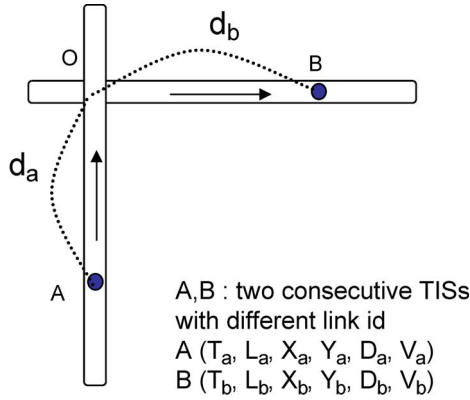


Fig. 3. Intersection delay example RTD.

For example,  $IDP(RTD, L1, L2, W, A.M., 40, 0.2\%, 75\%)$  represents that, during the A.M. peak hours of a workday (“W, A.M.”), it takes 40 s to complete a right turn from links  $L1$  to  $L2$ , with the support being 0.2% and the confidence being 75%.

IDPs can be discovered by sequential pattern mining on spatial and temporal sequences in the TIDB. Each sample of intersection delay must be in a journey, which contains two consecutive TISs with different links. Fig. 3 illustrates one example of the RTD pattern: a probing vehicle that drives north and then turns right to the east, reports TIS at location  $A$  of link  $L_a$ , and consecutively reports TISs at location  $B$  of link  $L_b$ . The symbols used in the TIS format  $(T, L, X, Y, D, V)$  in Fig. 3 represent the timestamp  $T$ , link ID  $L$ , coordinates  $(X, Y)$ , moving direction  $D$ , and moving speed  $V$ , respectively. The distances  $d_a, d_b$  in Fig. 3 represent the distance from  $A, B$  to the intersection of links  $L_a$  and  $L_b$ , respectively. Assuming that, in the short time interval between  $T_b$  and  $T_a$ , the vehicle is driving at a speed of  $V_a$  at link  $L_a$  and  $V_b$  at link  $L_b$ , the time to make a right turn from  $L_a$  to  $L_b$  can be calculated by simply extracting the two short periods of moving time from the time interval between  $T_b$  and  $T_a$ . A two-way cross-intersection object is typically composed of 12 intersection delays, including a TD/LTD/RTD for each E/W/S/N direction. For example, the RTD time from  $L_a$  to  $L_b$  can be estimated by subtracting the travel time of  $d_a$  and  $d_b$  from the elapsed time between two TIS  $(T_b - T_a)$ .

#### D. CPP

The CPP, denoted as  $CPP(A, B)$ , indicates a root cause for the congestion relationship between the following two STCAs: 1) the STCA  $A$  in time zone  $T_j$  and 2) the STCA  $B$  in time zone  $T_j$  or after  $T_j$ . In other words, it is highly possible that STCA  $B$  will be congested after STCA  $A$  has been congested if  $CPP(A, B)$  exists, and some of the traffic demands in STCA  $B$  comes from STCA  $A$ . To discover the congestion propagation relation between two STCAs, the CPP discovering algorithm is developed, as illustrated in Fig. 4, where the variable  $\Delta_t$  indicates the temporal distance between the TNS  $N_i$  and its consequent snapshots. The spatial boundary threshold  $H_u^s$  and demand overlap ratio (DOR) threshold  $H_u^d$  coupled with two heuristics are used for the threshold-filtering setting in discovering CPP patterns as follows.

#### Algorithm 2 : CPP Discovering

##### Denotation:

- $A_i^j$ : the  $i$ -th STCA in snapshot  $N_j$
- $\Delta_t$ : the temporal distance between two snapshots
- $H_t$ : the temporal boundary threshold
- $N = \{N_j, N_{j+1}, \dots, N_{j+H_t}\}$ : the set of traffic network snapshots within the temporal boundary threshold
- $H_u^s$ : the spatial boundary threshold under the temporal distance  $\Delta_t$
- $H_u^d$ : threshold of DOR under the temporal distance  $\Delta_t$

##### Input:

TNSs:  $\{N_j, \dots, N_{j+H_t}\}$

for each  $A_i^j \in N_j$

for each  $A_k^l \in N_m, m=j, j+1, \dots, j+H_t$

$\Delta_t = l - j$

/\* spatial boundary threshold ( $H_u^s$ ) filtering \*/

if spatial distance ( $A_i^j, A_k^l$ ) is smaller than  $H_u^s$

then

/\* DOR threshold ( $H_u^d$ ) filtering \*/

if  $\sigma(A_i^j, A_k^l) > H_u^d$

then output  $(A_i^j, A_k^l)$  as a CPP

Fig. 4. CPP patterns mining algorithm.

- Spatial boundary threshold  $H_u^s$ . This threshold is under the temporal distance  $\Delta_t$  and is positively related to temporal distance  $\Delta_t$ . For example, in the case of  $\Delta_t = 1$ ,  $H_1^s = 5$  indicates that the congestion that originates in one object in TNS  $N_0$  may propagate a five-network-object distance that is 15 min away in TNS  $N_1$ . In the case of  $\Delta_t = 2$ ,  $H_1^s = 12$  indicates that the congestion may propagate a 12-network-object distance away in TNS  $N_2$  from the original congested object after 30 min ( $\Delta_t = 2$ ), which indicates that the congestion in one object may result from the congestion of another object, where the spatial distance between these two objects is within a maximum 12-object distance, and the temporal distance is within 30 min. The concept of spatial boundary increasing with temporal distance matches the traffic congestion propagation experience.
- DOR threshold  $H_u^d$ . This threshold is the threshold of DOR under the temporal distance  $\Delta_t$  and is negatively related to  $\Delta_t$ , because the traffic demand scatters. The threshold decreases as the temporal distance increases.

As illustrated in (6)–(9), the DOR function (denoted as  $\sigma$ ) defines the journey overlapped ratio of two STCAs  $A_i^j$  and  $A_k^l$ , where  $J_m = \langle S_1, \dots, S_n \rangle$  is a journey of a vehicle indexed by  $m$ , which originates in  $S_1$  and ends at  $S_n$ . In (7),  $J_m \propto A_k^l$  is a relationship that indicates that the journey  $J_m$  is spatiotemporally overlapped with  $A_k^l$ , i.e., some TISs of the journey  $J_m$  are located in the STCA  $A_k^l$ . The  $\lambda$  function defines the “passing by” concept, as shown in (8), which obtains the minimum index of TISs in the sequence of  $J_m$  that are overlapped with  $A_k^l$ . In (9), the  $\sigma$  function determines the DOR of the two STCAs  $A_i^j$



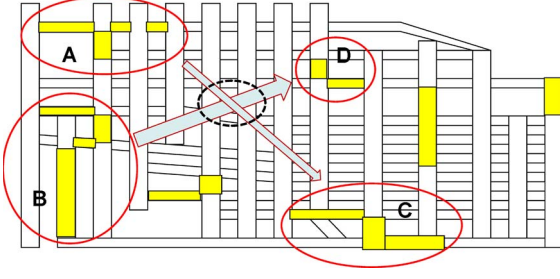


Fig. 5. DCP STB most likely exists in the spatial cross demand of two CPPs (black dotted area).

and  $A_k^l$  by calculating the journeys overlapped ratio of  $A_i^j$  and  $A_k^l$ , where all these journeys are overlapped with  $A_i^j$  before  $A_k^l$

$$J_m = \langle S_1, \dots, S_n \rangle \quad (6)$$

$$J_m \propto A_k^l \Leftrightarrow \exists i \in \{1, \dots, n\}, \quad S_i \in A_k^l \quad (7)$$

$$\lambda(J_m, A_k^l) = \min \{i | A_k^l \propto J_m, S_i \in J_m, \text{ and } S_i \in A_k^l\} \quad (8)$$

$$\begin{aligned} & \sigma(A_i^j, A_k^l) \\ &= \frac{\left| \left\{ J_m | A_i^j \propto J_m, A_k^l \propto J_m, \lambda(A_i^j, J_m) < \lambda(A_k^l, J_m) \right\} \right|}{\left| \{ J_m | A_k^l \propto J_m \} \right|}. \end{aligned} \quad (9)$$

#### E. DCP

DCP is constructed based on CPP and IDP. The concept of DCP is that, if two or more CPPs exist, which have spatial conflict traffic demands, then it may result in some traffic bottlenecks in the cross area of these two traffic demands. As illustrated in Fig. 5, there are two CPPs— $CPP(A, C)$  and  $CPP(B, D)$ . The traffic demand of  $C$  comes from  $A$ , and the traffic demand of  $D$  comes from  $B$ , which has a spatial-demand conflict area, as shown in the dotted circle in Fig. 5. The determination of the spatial conflict demand of two CPPs needs the support of the GIS engine with an urban network database. The bottleneck of DCP is most likely located at the intersection of two connected links and can be discovered by examining the intersection delay of intersection objects in the spatial-demand conflict area (see the dotted circle in Fig. 5).

With the assistance of GIS, the intersections located within the cross area of  $[A, B, C, D]$  are selected as the intersection bottleneck candidates. In this area, the traffic demands in the two CPP patterns may cross over each other in the intersection objects and result in the following two kinds of intersection delays, as illustrated in Fig. 6(a) and (b), respectively: 1) cross-conflict delay and 2) left-turn interlaced delay. Cross-conflict delay indicates that the two traffic demands cross over each other in the intersection, and left-turn interlaced delay shows that the two traffic demands meet and both turn left at the intersection. By checking the TD and LTD of intersection bottleneck candidates, the intersections that fit in with these two patterns can be found.

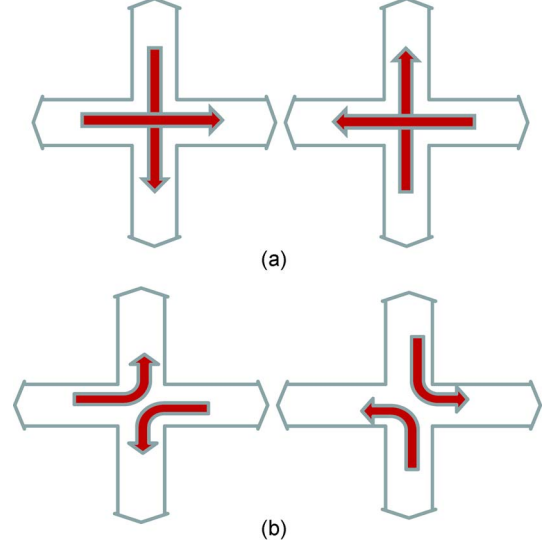


Fig. 6. (a) Cross-conflict delay. (b) Left-turn interlaced delay.

#### IV. PHASE III: SPATIOTEMPORAL BOTTLENECK MINING

Spatiotemporal traffic congestion patterns that were discovered in the previous phase are classified by object- and area-level patterns. COI, IDP, and CDP are object-level traffic patterns, which can directly be transformed into STB by threshold filtering. CPP and DCP, which are area-level traffic patterns, are further analyzed in this section. In phase III, the following three STBM heuristics are proposed to discover the STBs in the traffic network: 1) the congestion-propagation heuristic (CPH); 2) the congestion-converge heuristic (CCH); and 3) the congestion-drop heuristic (CDH). The first two heuristics derive from CPP, which then transforms into a congestion-area sequence rule (CASR), and the last heuristic derives from CDP. A CASR, defined as a three-tuple vector, as illustrated in (10), is a traffic-demand-oriented congestion association rule between two STCAs in the same or neighborhood TNSs, where  $A_i^j$ ,  $A_k^l$  indicate two STCAs located at TNS  $N_j$ ,  $N_l$ , respectively, the confidence  $C_{onf}$  of the CASR indicates the ratio of the days that support this rule compared to all days, and the DOR of these two STCAs is larger than the threshold  $H_u^d$ . We have

$$R = \langle A_i^j, A_k^l, C_{onf} \rangle$$

where

$$C_{onf} = \frac{\left| \left\{ \{ A_i^j, A_k^l \} | \sigma(A_i^j, A_k^l) > H_u^d \right\} \right|}{\left| \{ A_i^j, A_k^l \} \right|}. \quad (10)$$

##### A. CPH

The idea of CPH is that the traffic demand of an STCA may propagate to more STCAs in the neighborhood area. Therefore, if an STCA exists in the condition part (left-hand side) of several CASR rules, then it is regarded as a root-cause STCA. Thus, we assume that some bottlenecks exist in this root-cause STCA. The CPH bottleneck category can be found by

aggregating the condition part of all the discovered CASRs to find the root-cause STCA. All the network objects in the root-cause STCA are treated as STB candidates. By a detailed inspection of the traffic attributes, including the  $\theta$ , traffic demand direction, and congestion drop ratio ( $\tau$ ) of each candidate object, STBs can be determined by the threshold criteria.

### B. CCH

An STCA may result from some other prior neighborhood STCAs. The idea of CCH is that, if an STCA exists in the action part (right-hand side) of several CASR rules, then we assume that it contains some STBs. Aggregating the action part of all CASRs grouped by the temporal period finds the CCH type of the STCA. Again, every object in the CCH-type STCA is treated as an STB candidate, and the STB can be determined using the same criteria as in CPH.

### C. CDH

The idea of CDH is that, if the congested status of an object dramatically decreases or even disappears after the traffic flow has passed through the object, then it is regarded as a bottleneck. This type of bottleneck occurs, because the capacity of the bottleneck object is not enough to serve the traffic demand. Downstream objects do not become congested, because the capacities of these objects are sufficient to serve the same traffic demand. The objects in the discovered CDP patterns discussed in Section 4 can directly be transformed into CDH by threshold filtering  $H_d$ .

## V. EXPERIMENTS

The STBM prototype system was implemented based on a real-time TDS [14], which is an online  $7 \times 24$  LBS-based application. The target area of this prototype system, the same as TDS, focused on the urban network in Taipei City, where an arterial in the network consists of one or several links and intersections. A link object that comprises several attributes, including category, length, direction, speed limit, and intersection object, has the attributes of TD/LTD/RTD and the values of each direction, with default values given by domain experts to facilitate STB discovery. The taxi fleet in the TDS is the size of 500 taxis, and the OBU periodically reports its status at an interval of 30 s or when some events occur. Raw data were collected from the TDS from February 2006 to March 2007, where the average size was 0.5 million uplink reports per day. Traffic journeys were transformed from the raw data by combining the GIS road network. For example, the “dispatched” state journey comprises a set of TISs, beginning from the dispatched location to the customer location, and the “occupied” state journey begins from the customer location to the destination.

In phase I, the  $\theta$  of each link object is obtained by aggregating the TISs at the temporal and spatial dimensions and is normalized by category and speed-limit attributes. For example, the  $\theta$  of an object around 0 ( $\theta$  may be negative) indicates that the link is in the free-flow state and the average traveling speed is

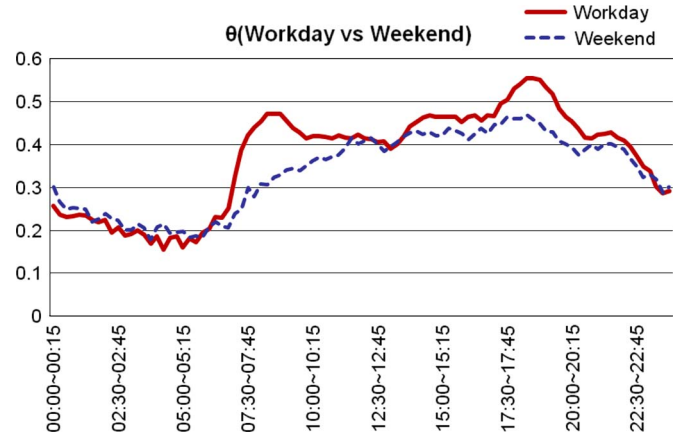


Fig. 7. Statistics of  $\theta$  for workdays and weekends.

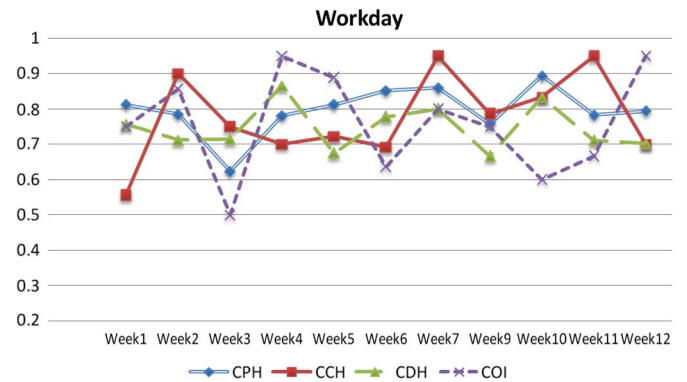


Fig. 8. Three heuristics compared with the COI method during workdays.

around the speed limit. If  $\theta$  is close to 1, it indicates that the link is in an extremely congested status. The average  $\theta$  of the network objects for workdays and weekends are summarized in Fig. 7, where each plot in the  $x$ -axis indicates a 15-min time slot. The workday curve consists of two peaks, which verify the common experience of A.M. and P.M. peak hours. The weekend curve does not have any obvious peak. To reduce the computing complexity, the search period for the STB in this experiment is limited to the two peak hours of a workday, where the average  $\theta$  of the A.M. peak hour (07:30~09:30) is 0.45, and the P.M. peak hour (17:30~19:30) is 0.54.

The raw data collected from March to December 2006 are used for discovering STB, and the remaining raw data from January to March 2007, which include 12 weeks, are used for comparison. The STB transformed from the COI, regarded as a statistic model, is compared with the STB discovered by the three heuristics discussed in Section 5. The experimental results for workdays and weekends are shown in Figs. 8 and 9, respectively. Fig. 8 shows the congestion prediction comparison of the three proposed heuristic methods (CPH, CCH, and CDH) with the COI method for workdays, and Fig. 9 shows the same comparison for weekends. The results show that the average accuracy of the CPH method has higher accuracy than the other methods (CCH, CDH, and COI) for both workdays (79.6%) and weekends (72.1%). Although the CDH method is as accurate (74.7% and 71.1% on workdays and weekends, respectively) as CPH, it is the most stable method (with a standard deviation



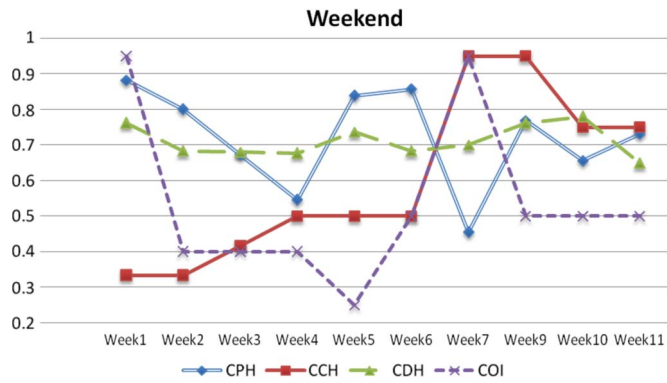


Fig. 9. Three heuristics compared with the COI method during weekends.

of 0.055) compared with the other three methods (CPH = 0.105, CCH = 0.179, and COI = 0.190). The congestion prediction accuracy of these methods, when applied on weekends, is more unstable than when applied on workdays, and the prediction accuracy is worse than when applied on workdays, as shown in Fig. 9. These results may be because traffic demands during weekends are diverse compared to traffic demands during workdays. For example, the following two types of traffic demand patterns mostly appear on workdays: 1) A.M. peak patterns, which are the inbound demands in the morning, and 2) P.M. peak patterns, which are the outbound demands in the evening.

In the experiment, we found that the CPH method performs better in terms of precision and stability. Fig. 10 shows the STBs discovered by CPH for the A.M. and P.M. peak hours. In the Taipei urban network, the southeast side of the city is the business center. The CPH mines the following three types of bottlenecks located at the Taipei urban network: 1) Pink arrows with a vertical sign (“||”) indicate the A.M.-peak-hour STBs from 7:30 to 9:30; 2) blue arrows with a horizontal sign (“-”) indicate the P.M.-peak-hour STBs from 17:30 to 19:30; and 3) yellow arrows with a plus sign (“++”) indicate the STBs in both A.M. and P.M. peak hours. The results show that the A.M.-peak-hour bottlenecks come from the suburban into the urban center, which reveals the inbound traffic demands of going to work, whereas the P.M.-peak-hour bottlenecks are only from the urban center to the suburban areas, which show the off-work outbound traffic demands of returning home.

## VI. CONCLUSION

Locating traffic bottlenecks in urban networks is more difficult than on freeways, because there are no intersection, and traffic signal controls on the freeways and bottlenecks in the freeway/highway are typically static and clear, are located around the on- and off- ramps and decreased lanes, or are affected by road curves and road gradients. In this paper, the proposed STBM model has focused on discovering traffic knowledge (STP/STB) in urban networks. The STBM model utilizes the raw data collected from LBS-based applications, which has the advantages of cost and coverage compared with the traditional sensor-based surveillance systems. The STPs and STCA rules discovered in phase II describe the relationships

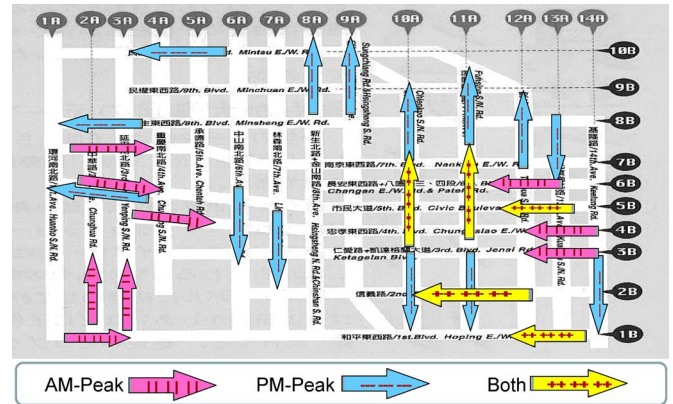


Fig. 10. Workday bottlenecks in Taipei City, Taiwan, (mined by CPH).

between traffic demands and congestion, which present congestion predictive capability. The discovered traffic knowledge also provides decision support information for the ATMS administrator to make appropriate traffic assignments to relieve traffic congestion and solve bottlenecks, thus enhancing global network performance. Although the experiments focus on discovering the STBs in the workday A.M. and P.M. peak hours, the proposed model can be applied to any spatiotemporal criteria for an urban network. Experimental results show that the average workday accuracy discovered by the proposed three heuristics is upward to 80%, which is better than the COI method (statistic model). The DCP-type STB discovering model is not included in the experiments due to the lack of GIS support to retrieve the spatial-demand conflict objects.

In the future, we plan to implement the STBM model as a real-time bottleneck detection and prediction system. Combining the discovered traffic knowledge (STPs and STBs) and domain expertise donated by traffic administration experts and extending the STBM system to cooperate with a knowledge-based system can provide further traffic assignment suggestions.

## ACKNOWLEDGMENT

The authors would like to thank the Telecommunication Laboratory, Chunghwa Telecom Company Ltd., for providing the raw data on the online taxi-dispatching system (TDS) system.

## REFERENCES

- [1] B. S. Kerner, *Introduction to Modern Traffic Flow Theory and Control*. Berlin, Germany: Springer-Verlag, 2009.
- [2] B. S. Kerner, *The Physics of Traffic: Empirical Freeway Pattern Features, Engineering Applications, and Theory*. Berlin, Germany: Springer-Verlag, 2004.
- [3] H. S. Zhang, Y. Zhang, Z. H. Li, and D. C. Hu, “Spatial-Temporal traffic data analysis based on global data management using MAS,” *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 267–275, Dec. 2004.
- [4] S. H. Tsai, W. H. Lee, and S. S. Tseng, “A spatiotemporal traffic patterns mining on LBS-based application,” in *Proc. 10th Conf. Artif. Intell. Appl.*, Kaohsiung, Taiwan, Dec. 2005.
- [5] E. Chung, “Classification of traffic patterns,” in *Proc. 11th World Congr. Intell. Transp. Syst.*, 2003, pp. 1–11.
- [6] B. S. Kerner, C. Demir, R. G. Herrtwich, S. L. Klenov, H. Rehborn, M. Aleks, A. Haug, and A. G. DaimlerChrysler, “Traffic state detection with floating car data in road networks,” in *Proc. 8th IEEE Conf. Intell. Transp. Syst.*, 2005, pp. 44–49.

- [7] M. Abdel-Aty and A. Pande, "ATMS implementation system for identifying traffic conditions leading to potential crashes," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 78–91, Mar. 2006.
- [8] W. H. Lee, S. S. Tseng, and S. H. Tsai, "A knowledge-based real-time travel time prediction system for urban network," *J. Expert Syst. Appl.*, vol. 36, pt. 1, no. 3, pp. 4239–4247, Apr. 2009.
- [9] T. Kawahara, S. Kamijo, and M. Sakauchi, "Travel time measuring by using vehicle sequence matching between adjacent intersections," in *Proc. IEEE Conf. Intell. Transp. Syst.*, 2005, pp. 712–717.
- [10] S. Kamran and O. Haas, "A multilevel traffic incidents detection approach: Identifying traffic patterns and vehicle behavior using real-time GPS data," in *Proc. IEEE Intell. Veh. Symp.*, Istanbul, Turkey, Jun. 2007, pp. 912–917.
- [11] B. S. Kerner and A. G. DaimlerChrysler, "Control of spatiotemporal congested traffic patterns at highway bottlenecks," in *Proc. 8th IEEE Conf. Intell. Transp. Syst.*, 2005, pp. 136–141.
- [12] B. S. Kerner, "Control of spatiotemporal congested traffic patterns at highway bottlenecks," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 308–320, Jun. 2007.
- [13] J. C. Long, Z. Y. Gao, H. L. Ren, and A. P. Lian, "Urban traffic congestion propagation and bottleneck identification," *Sci. China Ser. F: Inf. Sci.*, vol. 51, no. 7, pp. 948–964, Jul. 2008.
- [14] H. Y. Liu, C. H. Wang, V. S. Shieh, and B. S. Jeng, "An intelligent taxi dispatching management system," in *Proc. Cross Strait Conf. Intell. Transp. Syst.*, 2004, pp. 111–117.
- [15] Z. Liao, "Taxi dispatching via global positioning systems," *IEEE Trans. Eng. Manag.*, vol. 48, no. 3, pp. 342–347, Aug. 2001.
- [16] K. T. Seow and D. H. Lee, "Performance of multiagent taxi dispatch on extended-runtime taxi availability: A simulation study," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 231–236, Mar. 2010.
- [17] D. H. Lee, H. Wang, R. L. Cheu, and S. H. Teo, "A taxi dispatch system based on current demands and real-time traffic information," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1882, pp. 193–200, 2004.
- [18] C. W. Wang, C. C. Chiu, S. D. Jeng, S. R. Hsiao, L. G. Wei, C. H. Chao, and C. H. Hwang, "A geocoding application on GIS using address data: Case study of Taiwan address database," in *Proc. TGIS Annu. Conf.*, Taipei, Taiwan, pp. 25–26.
- [19] A. Hegyi, B. De Schutter, and J. Hellendoorn, "Optimal coordination of variable speed limits to suppress shock waves," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 1, pp. 102–112, Mar. 2005.
- [20] W. Brilon and M. Ponzlet, "Variability of speed–flow relationships on German autobahns," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1555, pp. 91–98, 1996.



**Wei-Hsun Lee** (M'09) received the Ph.D. degree in computer science from the National Chiao Tung University, Hsinchu, Taiwan, in 2009.

From 1993 to 2010, he was with the Telecommunication Laboratories, Chunghwa Telecom Company Ltd., Taoyuan, Taiwan, as an Associate Researcher. He is currently an Assistant Professor with the Department of Transportation and Communication Management Science, as well as with the Institute of Telecommunications and Management, National Cheng Kung University, Tainan, Taiwan.

His research interests include knowledge-based systems, spatiotemporal data mining, telematics systems, advanced traffic management systems, electronic toll collection systems, vehicle-positioning systems, and near-field communication techniques.



**Shian-Shyong Tseng** (M'84) received the Ph.D. degree in computer engineering from the National Chiao Tung University, Hsinchu, Taiwan, in 1984.

From 1983 to 2009, he was with the Department of Computer and Information Science, National Chiao Tung University. From 1988 to 1991, he was the Director of the Computer Center, National Chiao Tung University. From 1991 to 1992 and from 1996 to 1998, he was the Chairman of the Department of Computer and Information Science. From 1992 to 1996, he was the Director of the Computer Center, Ministry of Education, and the Chairman of the Taiwan Academic Network Management Committee. In December 1999, he founded the Taiwan Network Information Center (TWNIC). Since 2005, he has been with the Department of Applied Informatics and Multimedia, Asia University, Taichung, Taiwan. He is currently the Vice President of Asia University and the Chairman of the Board of Directors of TWNIC. His research interests include data mining, expert systems, computer algorithms, e-learning, and Internet-based applications.



**Jin-Lih Shieh** received the B.S. degree in information computer engineering from Chung Yuan Christian University, Chung Li, Taiwan, in 1990 and the M.S. degree in computer science and information engineering from the National Cheng Kung University, Tainan, Taiwan, in 1993. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan.

He is also a Lecturer with the Department of Management Information System, Far East University, Tainan. His research interests and works include overlay networks, intelligent transportation systems, peer to peer networking, and distributed systems.



**Hsiao-Han Chen** received the B.S. degree in information computer engineering from Chung Yuan Christian University, Chung Li, Taiwan, in 2005 and the M.S. degree from the National Chiao Tung University, Hsinchu, Taiwan, in 2007.

She is currently an Associate Researcher with the Telecommunication Laboratories, Chunghwa Telecom Company Ltd., Taipei, Taiwan. Her research interests and works include knowledge-based systems, intelligent transportation systems, mobile billing management systems, and mobile number portability

systems.