國立交通大學

資訊工程研究所

博士論文

都會區交通路網整體效能提升之研究

A Study of Enhancing the Global Traffic Network Performance for Urban Network

研究生：李威勳

指導教授：曾憲雄 博士

中華民國九十八年七月

都會區交通路網整體效能提升之研究

A Study of Enhancing the Global Traffic Network Performance for Urban Network

研究生：李威勳 Student: Wei-Hsun Lee

指導教授：曾憲雄 博士 Advisor: Dr. Shian-Shyong Tseng

國立交通大學

資訊學院 資訊工程學系

博士論文

A Dissertation
Submitted to Department of Computer Science
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy
in
Computer Science

July 2009
Hsinchu, Taiwan, Republic of China

中華民國九十八年七月

A Study of Enhancing the Global Traffic Network Performance for Urban Network

Student: Wei-Hsun Lee Advisor: Dr. Shian-Shyong Tseng

Department of Computer Science
College of Computer Science
National Chiao Tung University

Abstract

Global traffic network performance enhancement is one of the critical issues for the urban network administration since traffic congestion has been increasing world-wide as a result of increased motorization, urbanization, population growth and changes in population density, especially in urban network. Network administrators try to enhance the traffic network performance by overall considering all the related issues including current network traffic status, traffic demands, historical traffic patterns, traffic bottlenecks, and predicted traffic status. However, difficulties and issues are raised in traditional Dynamic Traffic Assignment (DTA) models, including traffic demand dynamics, network complexity, spatiotemporal traffic bottlenecks, lack of traffic event consideration, and network topologies evolvement problem.

In this dissertation, a traffic knowledge framework is proposed for enhancing the traffic network performance by considering four major factors including decreasing the traffic demand, discovering Spatiotemporal Traffic Patterns (STPs) and identifying the Spatiotemporal Traffic Bottlenecks (STBs), resolving the traffic bottlenecks, and creating a collaborative traffic information generation and sharing framework. There are four phases in the propose framework, which are traffic information generation, heterogeneous traffic information fusion, traffic knowledge extraction and traffic information applications. Two real-time traffic information collection schemes are proposed including traffic information derived from the LBS-based applications and collaborative traffic information generation and sharing framework. All the collected traffic information is kept in the Traffic Information Database (TIDB), from which STPs as well as STBs are mined by spatiotemporal data mining techniques. The discovered traffic knowledge is applied to travel time prediction system and Advanced Traffic Management System (ATMS) decision support system, where the former assists the travelers to select the shortest travel time path, and the latter provides traffic assignment suggestions to network administrators for enhancing traffic network performance. Knowledge based system technique is adopted for these two applications with pre-designed domain knowledge ontologies obtained from the domain experts. The collected as well as predicted traffic information, events, network topologies, constraints are regarded as the facts in the inference engine.

The contributions of this work can be summarized as: a traffic information and knowledge framework is built for the ground works of ITS, several hypothesis based spatiotemporal data mining methods are proposed for discovering STP/STB, the proposed travel time prediction system outperforms the real-time predictor as well as the historical predictor, the proposed ATMS decision support system is tractable to apply to real urban network comparing to traditional DTA-based approaches, and the knowledge base is continuously incremental updated to get more precise rules.

Keywords: intelligent transportation system (ITS), spatiotemporal data mining, spatiotemporal traffic patterns, spatiotemporal traffic bottlenecks, advanced traffic management system (ATMS)

摘要

由於車輛機動化、都市化、人口增加與人口密度提高的趨勢，導致都會區交通路網的壅塞成為世界性的趨勢，交通網路整體效能的提升是交通管理中心的關鍵性任務，交管中心管理者在考量交通需求、交通狀態、歷史資料、交通瓶頸點與預測資料等全盤狀況之後，做出提升整體交通路網效能的決策。然而以傳統的動態交通指派模型(DTA)方法，要提升整體交通路網效能卻可能面對許多的困難，包含交通需求的變化、路網的複雜度、時空變化的交通瓶頸點、缺乏事件應變機制、路網變更須重新定義模型等等問題。

在本論文中提出了一個以提升整體交通路網效能為目的的交通知識框架，考量了數個可以改善交通路網效能的要素，包含建立協同式的交通資訊產生與分享的架構、減低交通旅次需求、找出時空變化交通樣式(STP)與時空變化交通瓶頸點(STB)、利用系統找出的交通知識與專家提供的知識來改善交通路網的效能。此框架分成了交通資訊收集與產生、異質交通資訊的融合、交通知識的萃取與交通管理的應用等四個階段，在論文中提出了兩種即時交通資訊收集的方式，包含從車輛定位應用服務(LBS)資料中分析出交通資訊，與協同式交通資訊產生與分享架構，所收集到的交通資訊存放在交通資訊資料庫(TIDB)中，做為以時空資料探勘技術找出各種時空交通樣式(STP)與時空交通瓶頸點(STB)的資料源。在本論文中，我們將這些交通資訊與從這些資訊中找出的相關交通知識應用到兩個系統: 旅程時間預測系統與先進交通管理支援決策系統，前者提供用路人最短旅程時間路徑的資訊，後者提供路網管理中心有關於提升交通路網效能的交通指派建議。這兩個應用系統都採取了知識庫系統的技術，推論引擎中的推論法則(Rule)是由

領域專家協助設計的知識本體論以及發掘的交通知識(STP/STB)所轉換而來，而推論所需要的事實(Facts)則包含了系統收集到即時交通資訊與事件、預測的交通資訊、路網架構資訊與交通法規限制等等。

本研究的貢獻簡單彙整如下:一、建立一個交通資訊與知識的系統框架，為進一步發展智慧型運輸系統的基礎工程；二、提出數個以時空資料探勘方式找出時空交通樣式與交通瓶頸點的方法；三、提出的動態混合式旅程時間預測方法，其精準度比即時資料旅程時間預測與歷史資料旅程時間預測方法都來的高;四、以知識庫方法實現的交通管理決策系統比傳統的交通指派方法更容易實現在都會區路網上，並且系統可以持續更新知識庫以得到更精準的推論法則。

關鍵字: 智慧型運輸系統(ITS), 時空資料探勘, 時空交通樣式, 時空交通瓶頸點,先進交通管理系統(ATMS)

誌謝

本篇論文能夠順利完成付梓，最要感謝的當然是指導老師曾憲雄教授，老師嚴謹的治學態度與啓發式的指導，培養了我獨立思考、解決問題的能力，並得以領略學術研究的奧妙，本論文在研究方面若能有所突破與創新，都要歸功於 曾教授對教育理念的堅持及在學術研究的創見。同時也要感謝從論文研究計畫書審查、校內口試到最後口試一路給予我許多論文修改建議的 卓訓榮教授與 孫春在教授，讓我重新檢視本論文的研究貢獻。還有感謝在口試中， 郭耀煌教授、 洪宗貝教授、 鄭伯順教授與 張光耀教授所提出的寶貴意見，對於論文方法與結果呈現，給於精闢的見解與指導，並且指正論文的缺失，讓此論文最後的成果能夠更加完整並增加整體論文的可讀性。

在不惑之年，能夠以在職的身份順利的完成學業，要感謝任職單位中華電信研究所的許多長官栽培與同仁的支持，特別是主管王景弘博士與繆嘉新博士的栽培與指教，使得此論文研究能夠順利進行。另外也要感謝知識工程實驗室的眾多夥伴，蘇董、慶堯、順傑、衍旭、哲青、瑞鋒、宗儒、元昕與眾多同門師兄弟們，一起做研究討論讓彼此的研究能夠更爲精進，當然也要感謝和我一起做研究的學弟妹，昇翰、曉涵，沒有你們努力做實驗，論文與期刊的可看性就不會這麼完整，謝謝!

最要感謝的當然是家人一路的支持，謹將本論文獻給在天上的父親、在嘉義的母親、親愛的老婆惠瑜、可愛的女兒宜臻，以及眾親友，並且感謝每一個幫助過我、支持我的夥伴。

李威勳 謹誌

2009 年 7 月於桃園楊梅

Table of Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1 Introduction

1.1 Problems in traffic network performance enhancement

Traffic congestion has been increasing world-wide as a result of increased motorization, urbanization, population growth and changes in population density, especially in urban network. Congestion reduces utilization of the transportation infrastructure and increases travel time, air pollution and fuel consumption, which may cause various social, environmental, and economic problems. Global traffic network performance enhancement has become more and more important, especially in the heavy traffic urban area. It is the common hope of all the participants in the traffic network and the major goal of the transportation department in the government. Advanced traffic management system (ATMS), targets at improving the efficiency of utilization of existing infrastructures and enhancing the global traffic network performance for the urban network, is one of the major topics in intelligent transportation system (ITS). The goal of an ATMS is to efficiently manage existing transportation resources in response to dynamic traffic conditions. In the traffic network, the transportation resources including traffic signal control, changeable message sign (CMS), reversible lane, or manual control, etc. can be properly managed to alleviate the congestions and improve global traffic network performance.

Traffic network consists of a set of network objects, each of which is either a link or an intersection. Congestions occur on some network objects by which the traffic demand cannot be fully serviced. Thus traffic bottleneck can be defined as an object in traffic network 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 propagated to the surrounding network objects, blocking intersections and result in congestion chaining. Most traffic congestions mainly result from traffic bottlenecks; therefore locating the traffic bottlenecks and taking appropriate actions to alleviate the congestions and improving the global performance of traffic network is major tasks of the network administrators. However, the traffic demand varies with spatial and temporal environments, so does the traffic bottleneck. In this work, spatiotemporal traffic bottleneck (***STB***) is defined and used instead of traffic bottleneck in order to clearly identify where and when the traffic congestion could occur. An ***STB*** is a traffic bottleneck with spatial as well as temporal identification, which indicates when and where a network object could be congested and may be the root cause of other neighborhood congestions.

In order to enhance the global traffic network performance for the urban network, real-time traffic information and historical traffic patterns are the essential information for the network administrators. Real-time traffic information plays an important role in several fields of ITS such as ATMS, advanced traveler information system (ATIS), commercial vehicle operation (CVO) and emergency management system (EMS). It is one of the most useful features especially when users are travelling outside, because real-time traffic information can assist and support travelers and drivers to plan the trip before travelling and make route choices decision during traveling in order to reduce the travel time and improve travelling safety. Besides, the traffic information also can support traffic management administrators to make decisions and take appropriate actions to alleviate the congestions and improve the global performance of traffic network. However, the availability of real-time traffic information suffers from the cost, coverage, and real-time issues in the traditional data collection techniques, so compromised solution must be made among these considerations.

In this work, two real-time traffic information collection schemes are proposed, one is traffic information derived from the LBS-based applications, the other is traffic information collected from the collaborative traffic information generation and sharing framework. Comparing to the traditional sensor based or probing vehicle based traffic information collection system, it is cost effective because the traffic information is derived from the raw data of LBS-based applications and the smart traffic agents (***STA***) installed in the mobile devices (ex. mobile phone, personal navigation device). Moreover, the proposed traffic information collection schemes have the temporal as well as spatial coverage advantages because traffic information can be gathered from the ***STA*** and the LBS vehicles 24 hours per day in real time. By analyzing and mining the collected data, network traffic status as well as vehicle journey information can be generated.

Traditionally, traffic network performance enhancement is resolved by dynamic traffic assignment (***DTA***) model, which tries to achieve the goal of system optimal of the traffic network by regulating the traffic flow. However, ***DTA*** model is hardly practicable in urban network for several reasons: 1) complexity: traffic network in urban area is too complex to model and resolve by ***DTA***, 2) traffic demand unavailable: ***DTA*** makes traffic assignment by the origin-destination (***O-D***) traffic demand analysis; however, the real ***O-D*** data is not cost-effectively available for the most traffic surveillance systems. Most ***DTA***-based traffic network optimal assignment researches obtain the ***O-D*** data by assumption, estimation or simulation.

The difficulties for enhancing the global traffic network performance for the urban network can be summarized by the following issues.

1) Un-availability of sensor data in real-time

    In traditional traffic data collection scheme, real-time traffic information is either unavailable or not able to be collected cost effectively. Both the sensor-based and the probing vehicle based traditionally traffic data collection schemes have the spatiotemporal coverage problem. The former suffers from the sensor coverage area, and the latter suffers from the quantity and temporal coverage (working hours) of the probing vehicles.

2) Spatiotemporal dynamics in traffic demand

    In urban network, the traffic demand is spatiotemporally dynamic, and the precise traffic demand is never known in advance.

3) Network complexity

    The network complexity in urban area is far more complex than the freeway. Traditional ***DTA*** model is hard to apply to the complex urban network because of the complexity [ZW+04].

4) Spatiotemporal Traffic bottleneck (***STB***)

    There exists some ***STB***s in urban network; moreover, some of them resulted from the origin destination (***O-D***) demand conflict, which makes the traffic assignments more difficult.

5) Network topology evolution

    Traditional DTA-model is not evolvable as the network topology change. It has to redesign the network DTA model if the network topology is changed.

6) Lack of traffic event re-action mechanism

    In traditional DTA-model based solution, there is no traffic event re-action mechanism, so that if there exist some traffic events in the network, then the traffic assignment actions suggested by the traditional DTA-model may not conform with the expectation.

1.2 Major factors in global traffic network performance enhancement

For the problems and difficulties in enhancing the global traffic network performance discussed above, in this work, we try to enhance the global traffic network performance by considering the following four facets: 1) decreasing the traffic demand (discussed in this Section), 2) discovering traffic patterns and identifying traffic bottleneck (discussed in Chapter 5 and Chapter 6), 3) resolving the traffic bottlenecks by traffic assignment by decision support system (discussed in Chapter 8), and 4) creating a collaborative traffic information generation and sharing framework for two-ways information exchanging between the travelers and the backend traffic information center (discussed in Chapter 4).

1.2.1 Decreasing the traffic demands

Since the traffic congestions and bottlenecks primarily result from the traffic demand of the traffic network, decreasing the traffic demands or temporally distributing the traffic demand from the peak hours to the non-peak hour should be helpful for enhancing the global traffic network performance. ETC (electronic toll collection) [SL+05, SL+06, LTW08, LJ+04] system and HOV (high occupancy vehicle) system [TAO07] are the two effective mechanisms for achieving the goals of decreasing the traffic demands or temporally distributing the traffic demands. HOV system decreases the traffic demand of actual vehicle journeys by matching travelers who have same O-D demand into the same vehicle in order to ‘raise’ the occupancy of the vehicles. ETC system decreases the traffic demand by tolling the vehicles which enter the urban area, and distributes the traffic flow by setting different toll fee for different periods, for example, high toll fee for peak-hours and low toll fee for non-peak hours. Although these two techniques can effectively decrease the traffic demand, the technical details are beyond the scope of this study. In this dissertation, we focus on the other three factors, as discussed in the following sub-sections. As for the ETC system technique is not strongly related with the research topics in this dissertation, the discussion of design and implementation of a VPS-based ETC system is listed in Appendix A.

1.2.2 Discovering traffic patterns and identifying traffic bottlenecks

Although the traffic network evolves with the growth of the traffic demand and the dynamics in traffic demand makes the performance of the traffic network un-predictable. Somehow there is some traffic patterns can be discovered to reveal the traffic network behavior by data mining on the raw data collected from the LBS-based applications. For example, congestions usually take places at some links on some time period, or some links usually congestion because of in-town traffic demand at ***AM*** peak hours.

On the other hand, spatiotemporal traffic bottleneck (***STB***) is defined and discovered by analyzing the discovered spatiotemporal traffic patterns (***STP***). With the knowledge of ***STP*** and ***STB***, network administrators can take some actions to relieve the congestion and enhance the global performance of the traffic network. In this dissertation, an ***STP*** mining model based on spatiotemporal data mining on the raw data collected from the LBS-based applications is discussed in the Chapter 5, and three heuristic methods for identifying the ***STB*** is discussed in Chapter 6.

1.2.3 Resolving the traffic bottlenecks

In addition to traditional ***DTA*** actions such as traffic signal control, the concept of ***DTA*** can be generalized and extended to overall traffic network performance consideration. With the knowledge of the ***STP/STB*** and the traffic demand between congestion areas which results in the ***STP/STB***, network administrators can take some actions to relieve the congestion, for example, traffic signal control, changeable message sign (CMS), reversible lane, or assigning traffic polices for manually regulating the busy intersections. However, there are some difficulties for the network administrators. For example, which traffic assignment action is most suitable or what parameter setting is optimal (e.g., green time in traffic signal control)? Different actions may result in different effects for the traffic network. For enhancing the global network performance, network administrators need a decision support system provide decision support information for selecting the traffic assignment actions by integrating the real-time and historical traffic information as well as considering the ***STP/STB***.

1.2.4 Collaborative traffic information generation and sharing framework

In addition to central side traffic control and assignment which are carried out at the backend system, user centric behaviors such as path choice to avoid the congestion area, or departure time choice to avoid the congestion periods may also have positive effects to the global network performance. A wiki-like user centric collaborative real-time traffic information generation and sharing framework based on the high penetration rates of location aware mobile devices is proposed in this study. By this collective traffic information generation scheme, since more real-time traffic data will be collected cost-effectively and accurately, the spatiotemporal coverage is better than the traditional traffic information collection scheme. Front-end smart traffic agent (***STA***), traffic information exchange protocol (***TIEP***), and backend traffic information center (***TIC***) constitute the traffic information sharing framework, where ***STA*** and ***TIEP*** are designed for the location aware real-time traffic information exchanging between the ***STA*** and ***TIC***. The ***TIEP*** enables ***STA*** to automatically echo the local traffic information to the ***TIC***, automatically download the local real-time and predictive traffic information, and reveal the traffic events or traffic status to the ***TIC*** manually by the traveler.

1.3 Reader's guide

In this study, a novel approach rather than the traditional ***DTA*** based approaches is proposed, which has the philosophy of enhancing the traffic network performance progressively rather than the optimal traffic assignment.

Table 1-1 Comparisons of traditional *DTA* models with this study

	Traditional DTA models	**This Study**
Goal	Optimal traffic assignment	Continuously enhancing traffic network performance
Model Techniques	Math. Programming, Optimal control theory, Variational inequality	Spatiotemporal data mining, Knowledge based system, Decision Support System
Data source	Traffic demand obtained by estimation or simulation	Raw data collected from LBS-based applications
Apply to	Small traffic network, Network planning	Real urban network performance enhancement
Adv.	Optimal traffic assignment if the network can be properly modeled	Practicable, Knowledge is dynamic adjustment

Dis-adv.	Intractable for large network, Can’t capture realities of network due to simplification	Optimality is not guaranteed

The remainder of this dissertation is organized as follows. The related works about traffic data collection and traffic state recognition, traffic patterns mining and traffic bottlenecks discovering, and travel time prediction are discussed in Chapter 2. The background knowledge of LBS, data collection and cleansing technique, and traffic information generation from LBS issues are discussed in Chapter 3. A collaborative traffic information generation and sharing framework is proposed and designed in Chapter 4 for the portal of traffic information collection and distribution, where several techniques including smart traffic agent (***STA***) for the front-end devices, multi-sources heterogeneous traffic information fusion, and traffic status predicting are proposed and detailedly discussed. In Chapter 5, spatiotemporal traffic patterns (***STP***) mining issue is discussed, where several ***STP***s are defined and discovered by hypothesis based spatiotemporal data mining technique. The granularities of the discovered ***STP***s range from object level, area level, to the global network level. Based upon the discovered ***STP***s, three heuristic methods for discovering the spatiotemporal bottlenecks (***STB***) are proposed and discussed in Chapter 6, which includes congestion propagation heuristic (***CPH***), congestion converge heuristic (***CCH***), and congestion drop heuristic (***CDH***).

Two applications about the advanced traveler information system (ATIS) and advanced traffic management system (ATMS) are discussed in Chapter 7 and Chapter 8 respectively. In Chapter 7, a knowledge based real time travel time prediction (***TTP***) system is designed and implemented, which contains real-time and historical travel time predictors and a dynamic weighted combination scheme. The discovered spatiotemporal traffic patterns are transformed to the prediction rules in the historical travel time predictor, and real-time traffic information constitutes the real-time predictor and facts in the inference engine. In addition to the ***TTP*** system designed for the traveler, a decision support system targeted to enhance the global network performance which provides valuable traffic assignment suggestions for the traffic network administrators in Chapter 8. It utilizes the discovered traffic knowledge discussed in this work, such as traffic information database, ***STP***/***STB***, predicted traffic status, and combines the domain expertise in order to optimize the global network performance and make traffic assignment suggestions. Finally, the concluding remarks and future works are discussed in Chapter 9.

1.3.1 Abbreviations and notations

Many abbreviations, terminologies, and notations are defined and used in this dissertation. In order to facilitate reading, the abbreviations are summarized in Table 1-2 and the notations are summarized in Table 1-3.

Table 1-2 Abbreviations

Abbreviation	Meaning
ATIS	Advanced Traveler Information System
ATMS	Advanced Traffic Management System
AVI	Automatic Vehicle Identification
CASR	Congestion Area Sequence Rule
CCP	Cross Conflict Pattern
CDP	Congestion Drop Pattern
CMS	Changeable Message Sign
CCH	Congestion Converge Heuristic
CDH	Congestion Drop Heuristic
CDR	Congestion Drop Ration (τ)
CPH	Congestion Propagation Heuristic

CPP	Congestion Propagation Pattern
CTM	Cell Transmission Model
CRP	Conflict Resolution Principle
CVO	Commercial Vehicle Operation
DCP	Demand Conflict Pattern
DOR	Demand Overlap Ratio (σ)
DSRC	dedicated short range communication
DSS	Decision Support System
DTA	Dynamic Traffic Assignment
ETC	Electronic Toll Collection
FCD	Floating Car Data
FED	Front-End Device
GEP	Global Enhancement Principle
GIS	Geographical Information System
GPS	Global Positioning System
GWP	Give Way Principle
IBP	Information Broadcast Principle
IDP	Intersection Delay Pattern
ITS	Intelligent Transportation System
KB	Knowledge Base
KC	Knowledge Class
KBS	Knowledge based system
LBS	Location-Based Service
LTD	Left-Turn Delay
LTIP	Left-Turn Interlaced Pattern
NORM	New Object-oriented Rule Model
OBU	On-Board Unit
O-D	Origin Destination
PCP	Priority Control Principle
RTD	Right-Turn Delay
RC-STCA	Root Cause Spatiotemporal Congested Area
SAP	Spatiotemporal Aggregation Pattern
SCP	Sole Congested Pattern
SHC	Spatial Heuristic Clustering

STA	Smart Traffic Agent
STB	Spatiotemporal traffic bottleneck
STCA	Spatiotemporal Congested Area
STCO	Spatiotemporal Congested Object
STO	Spatiotemporal Object
STP	Spatiotemporal traffic pattern
TD	Through Delay
TDS	Taxi Dispatching System
TIC	Traffic Information Center
TIDB	Traffic Information Database
TIEP	Traffic Information Exchange Protocol
TIF	Traffic Index Factor (θ)
TIS	Traffic Information Spot
TNS	Traffic Network Snapshot
TSP	Traffic Smoothing Principle
TTP	Travel Time Prediction
VPS	Vehicle-Positioning System

Table 1-3 Notations

Notation	**Meaning**
$\boldsymbol{\alpha}$	Dynamic weighted combination control variable for real-time predictor in TTP
$\boldsymbol{\beta}$	Dynamic weighted combination control variable for historical predictor in TTP
$\boldsymbol{\tau}$	Congestion Drop Ratio
λ	Definition of 'passing by' concept of a journey ($\boldsymbol{J_m}$) and a STCA(A_k^l)
σ	Definition of demand overlap ratio of two STCAs by calculating the overlapped journeys which passes by these two STACs
θ	Traffic index factor
O_i^j	Variable of a spatiotemporal network object spatially indexed by network object id $\boldsymbol{i}$ and temporally indexed by time zone $\boldsymbol{j}$

$\theta(O_i^j)$	normalized traffic status of spatiotemporal object O_i^j
N_j	Snapshot indexed by temporal index j
H_d	congestion drop threshold
H_u^s	spatial boundary threshold
H_u^d	demand overlap ratio
T_k	Time zone indexed by k

Chapter 2 Related Works

2.1 Traffic data collection and traffic state recognition

Traditionally, real-time traffic information collection is categorized into three schemes [LZ05]: site-based, sensor-based and probing vehicle-based data collection schemes. Site-based measurement collects vehicle license plate characters and arrival times at various checkpoints through automatic vehicle identification (AVI) technologies, matches the license plates between consecutive checkpoints, and computes travel times from the difference between arrival times. Vehicle-based methods analyze the raw data collected from fleet of probe vehicles by matching the vehicle tracks with geographical information system (GIS). Sensor-based scheme collects raw data from the stationary sensors like loops detectors, transponders or radio beacons installed at arterial roads. However, each traffic information collection method has some drawbacks and limitations. For example, site-based and sensor based methods have the spatial coverage problem due to the fixed and limited sensors or AVI devices. Vehicle-based scheme [NT04, YAN05, CH+03] has cost, spatial and temporal coverage problems due to the very high cost for maintaining a dedicated fleet of urban network traffic probing vehicles. Besides, the cost of real-time transmission for the whole traffic network in each data collection scheme is also very high [LTT09].

2.2 Traffic patterns mining and traffic bottlenecks discovering

In the literature, many researches aimed to find out the traffic patterns[ZZ+04, TLT05, TSA06, CHU03, CMC07] or to identify traffic states[KD+05] in traffic network, and some researches worked on predicting the travel time to provide drivers route suggestion[LTT09, KKS05]. Kerner [KD+05] proposed FCD (Floating Car Data) method to recognize traffic state (e.g., congested or not) by FCD vehicles in urban network, but still cannot identify the locations of the bottlenecks. Till now, most of previous studies tried to locate and control congestion patterns on highway bottlenecks [KER05, KER07] which are usually static, clear and always located around the gateway, but locating the bottlenecks on the urban network is more difficult than that on the freeway because there are no intersections and traffic signal control on the freeway. In other words, the task of analyzing traffic patterns in urban network and finding out traffic bottlenecks is a complex and difficult mission due to the following reasons: first, traffic network in urban area is more complex than that in freeway or simple arterial network; second, bottlenecks in urban network which are spatiotemporal dynamic varied with spatial or temporal environment and varied with traffic demands; third, not only more traffic factors but also more non-traffic factors have to be concerned in urban network than in freeway, such as traffic signal, social event, and traffic incidents, etc. Recently, [LG+08] tried to recognize urban traffic congestion propagation and identify bottleneck based on Cell Transmission Model (CTM), which discretizes each roadway into homogeneous section (cell) and discretizes time into intervals. Given the network objects capacities, it tries to identify the network bottlenecks by simulating the traffic demands of the urban network. In this dissertation, we try to not only discover the statistical and congestion propagation types of network bottlenecks, which similar to [LG+08], but also define and discover several types of traffic patterns including object level patterns and area level patterns, which are then transformed to spatiotemporal traffic bottlenecks by our proposed heuristic methods.

2.3 Dynamic traffic assignment (***DTA***)

DTA researches have evolved substantially since the pioneering work of Merchant and Nemhauser [MN78A, MN78B], which are typically classified into two broad categories: analytical models and simulation-based heuristic models [ZW+04]. The analytical models can be further classified by three groups: mathematical programming, optimal control, variational inequality [PZ01]. Efforts in the analytical models include mathematical programming approaches by [JAN91] and [ZIL00], optimal control theory based formulations by [FL+90], and variational inequality approaches introduced by [FB+93] and [SMI93]. The evolutions and literature review of the related analytical as well as simulation based approaches was done by Peeta and Ziliaskopoulos [PZ01]. Most analytical formulations are extensions of their static formulations and seem to have two main disadvantages: 1) they cannot adequately capture the realities of street network due to simplifications, and 2) they tend to be intractable for realistic size networks [ZW+04]. Besides, most analytical researches obtain the ***O-D*** data by assumption, estimation or simulation, which cannot capture the spatiotemporal traffic demand of the traffic network. So that the assignment suggestions based upon the assumption, estimation or simulation won’t have good effect for network performance enhancement.

2.4 Travel time prediction (***TTP***)

Study in [IK01] found that level of reduction in congestion depends on the complexity of the road network. Vehicular flows on freeways are often treated as uninterrupted flows; flows on urban network are conceivably much more complicated since vehicles traveling on urban network are subject to not only queuing delays but also signal delays and turning delays. Thus, ***TTP*** for an urban network is more challenging than predicting the travel time for freeway or single arterial. Besides, the routing and path selection problems should be solved in ***TTP*** for urban network, i.e., the ***TTP*** model has to suggest a shortest travel time path on a given OD (origin, destination) pair as a request. Many models had been proposed for travel time prediction in these decades, but most of them focused on predicting the travel time on freeway [WHL04, CK03] or simple arterial network [JZ03, LKM04].

In the past, many ITS studies and transportation agencies use the traffic data from dual-loop detectors which are capable of archiving with traffic count (the number of vehicles that pass over the detector in that period of time), velocity, and occupancy (the fraction of time that vehicles are detected) and readily available in many locales of freeways and urban roadways [LZ05]. Nowadays, traffic data collecting techniques have made great progress and evolved to real-time collecting in order to improve traffic management efficiency. In [LZ05], traffic information collection and travel time measurement can be divided into three categories: site-based, vehicle-based and sensor-based measurement. Site-based measurement collects vehicle license plate characters and arrival times at various checkpoints through automatic vehicle identification (AVI) technologies, matches the license plates between consecutive checkpoints, and computes travel times from the difference between arrival times. Vehicle-based methods make ***TTP*** by analyzing raw data collected from fleet of probe vehicles. Sensor-based methods make ***TTP*** measurement by collecting raw data from the stationary sensors like loops detectors, transponders or radio beacons installed at arterial roads. However, each traffic information collection method used for ***TTP*** has some drawbacks and limitations. For example, site-based and sensor based ***TTP*** methods have the spatial coverage problem because the sensors or AVI devices are fixed and limited. Vehicle-based ***TTP*** methods [NT04, YAN05, CS+03] have cost, spatial and temporal coverage problems because the total cost is very high if a dedicated fleet of urban network traffic probing vehicle is maintained.

There are numerous previous ***TTP*** approaches based on the historical traffic data analysis in the literatures, which can be categorized as follows [LZ05]: regression method (mathematical model) [WHL04], time series estimation method, hybrid of data fusion or combinative model [WLC05] and artificial intelligence method like neural network [MSR04]. In [NT04], auto regression (AR) model and state space model for time series modeling were used to predict travel time. The Kalman filtering provides an efficient computational (recursive) in many ***TTP*** researches [LKM04, CHU03, YAN05], because it is very powerful in several aspects: it supports estimations of past, present, and even future states even if the precise nature of the modeled system is unknown. In [WHL04], the support vector regression model was used to predict travel time for highway users. In [BCK04], pattern matching technique was used for ***TTP***. Traffic patterns similar to the current traffic are searched among the historical patterns, and the closest matched patterns are used to extrapolate the present traffic condition. Chung et al. [CH+03] developed an O-D estimation method to make more accurate estimation of traffic flow and traffic volume in congestion traffic status. Moreover, the data fusion models of ***TTP*** integrated grey theory [TTI03] and neural network-based. Yang [YAN05] developed some hybrid models toward data treatment and data fusion for traffic detector data on freeway.

Chapter 3 Traffic Information Generated from LBS-based Applications

3.1 Location-based service (LBS)

LBS, a kind of service providing appropriate location-aware information for the users, has become the main stream of mobile commerce applications and telematics service. The technologies used in LBS are positioning, mobile communication, and GIS (Geographical Information System). Front end devices such as OBU (On-Board Unit) or smart phone exchanges the users' location information with LBS backend system through mobile communication network to accomplish the business processes. There are many LBS-based applications had been proposed, such as electronic toll-collection by vehicle-positioning system (VPS) [LTW08], telematics service, taxi dispatching system (TDS) [LW+04], and commercial fleet management system, etc.

As shown in Figure 3-1, the architecture of LBS system includes four system components: OBU, communication system, M-Server and E-Server. OBU, a small computer system installed on the vehicle, has computing, positioning, communication and human interface modules. It can locate the vehicle through receiving GPS satellites signal by positioning module, send and receive the messages to and from the backend system through the communication module, and interact with user via the human interface module. LBS-based applications accomplish the business processes by exchanging information between the in-vehicle front-end devices (OBU, smart-phone) and the backend systems. Information is transmitted by the uplink packets (sent from OBUs to backend) and downlink packets (sent from backend to OBUs) over the mobile network. Such interactions among OBUs and the backend system form the basis of the LBS-based applications. TDS, one of the most complicated LBS-based applications, is used as the data mining source of this work because the traffic information included in TDS is considered more plentiful than other LBS-based applications, especially in urban network.

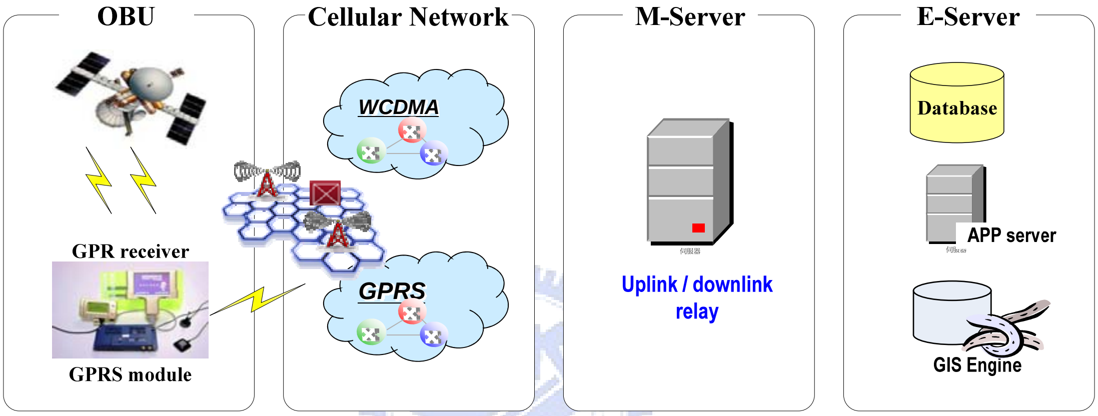


Figure 3-1 Components of LBS application

There are several participants in TDS, including customers (passengers), taxi drivers, operators and administrators. OBU automatically registers to the backend system through mobile network and turns into '***available***' state when it gets started. Taxi drivers can change the state of the taxi or interact with the backend system by the human interface on OBU. There are several functions designed for the driver to interact with the backend system and operator, including state change ('***available***' / '***occupied***' / '***scheduled***' / '***dispatched***' / '***rest***'), polling reply('**Y**' / '**N**' / [**minutes**]), emergency, and message request.

When OBU gets started, it automatically communicates and exchanges information with the dispatching center through the mobile network, and reports the position, direction, speed, and status of the taxi according to the predefined rules embedded in OBU. Therefore, the dispatching center keeps the latest statuses and positions of all the vehicles based upon all the collected uplink packets of OBUs. In TDS [LW+04], there are three uplink rules built in the OBU: periodically report (in fixed time interval), cross boundary report (on taxi driving through the geographical boundary), and event report (on status changing or event triggering). Customer requests a taxi via telephone call, fax, or Internet web site, operators key-in the requirements and location of the customers, and the backend system automatically searches the available taxi candidates nearby the location of the customer, probes the candidates that fulfill the requirements. The dispatched taxi driver responses the probing message from TDS, and moves on to the corresponding customer's location when the dispatching message is confirmed. The OBU turns into '***dispatched***' state when the taxi is dispatched, switches to '***occupied***' state when the customers get on the vehicle, and finally turns into '***available***' state when the customers get off the vehicle. By decoding and analyzing the uplink packets and state transitions, traffic status and traffic origin-destination demand in the urban network can be derived from communication raw data in LBS-based applications. Thus all the vehicles in the LBS-based applications can be regarded as the traffic status probing vehicles for the urban network.

3.2 Data collection and cleansing

Data collection module, a batch process, periodically collects the communication log between the front-end devices (OBU) in the vehicles and backend system in the LBS-based applications. Different data sources are collected and integrated by the data collection module, network connection, data transfer protocol and data format transformation issues are also done in this module. Real time data is collected and analyzed from the LBS-based applications, for different applications, data collection module and cleansing module have different ways of data collection and cleansing, depending on the application data format and domain knowledge. Take the TDS as an example, all the uplink packets are filtered out except OBU in "***occupied***" and "***dispatched***" states because only these two states can guarantee the vehicle is in moving status, which can reflect the real traffic status.

Data cleansing module takes care of the "***incomplete***", "***inconsistent***", "***outlier***", "***noise***" data preprocess issues, for example, data is not in the interested traffic network area, vehicle is not in moving status, incorrect GPS state, incorrect speed data, or incomplete data. Data normalization and those application dependent data preprocessing rules are also included in the data cleansing module.

3.2.1 Missing Values (incomplete)

In some cases, the uplink packets collected from the front-end devices are incomplete. This may happen due to malfunction of some components of the front-end device or the GPS signal is too weak to precisely locate the vehicle. The reasons for the weak GPS signal case might be that the vehicle is located in the weak signal area such as under an infrastructure (e.g., tunnel) or the vicinity of elevated structures (the so called urban canon). There are several cases in the GPS data missing, coordinates, speed, or direction missing. In the former two cases, the uplink packets are regarded as useless; however, in the case of missing direction data, it can be recovered by vehicle moving direction estimation by the assistance of the GIS.

3.2.2 Useless Data

Some collected data is regarded as useless data due to the following reasons: 1) the location of the uplink packet is not located in the interested urban area, 2) the outlier / incomplete / inconsistent data, 3) vehicle is not in moving status or 4) redundant data. The useless data is filtered out by the data cleansing module case by case. For example, for the case of vehicle not in the moving state, the vehicle continuously sends the speed “0” uplink packets to the traffic information center. The data collection module checks the real-time traffic status of the same link with ***TIC*** to verify that the link is in congested state or not. If the link is not in congested states, then the uplink packets from the vehicle are discarded due to they are not in moving state. This case will happen when the vehicle temporary stops for some business reasons, e.g., taxies in taxi stop, commercial vehicle (truck / wagon) stopped for goods loading / unloading, bus stopped at bus stop for people on / off the bus.

3.3 Traffic information generation

3.3.1 Traffic information spot (***TIS***)

Traffic information transformation module transforms the cleaned raw data into traffic information by integrating the urban road network database in GIS. Each uplink packet of the OBU can be transformed into a traffic information spot (***TIS***) because the information contained in the uplink packets includes location, moving speed, moving direction, and the state of the vehicle. By integrating the road network database with GIS, the coordinates of the GPS position of a vehicle can be interpolated to nearest address. Thus traffic information can be generated by transforming the uplink packet into ***TIS***. A ***TIS*** $S_k(\mathrm{O}_i^j, \boldsymbol{V}, \boldsymbol{D})$ transformed from the OBU uplink packet U_k of a vehicle, as illustrated in Equation (3.1), consists of object id (O_i^j), speed (***V***), direction (***D***) of the vehicle when it communicates with the LBS backend system at time ***t*** and location (***x**, **y***), where O_i^j is a spatiotemporal network object spatially indexed by network object id ***i*** (transformed from the location to address interpolation) and temporally indexed by time zone ***j*** (transformed from timestamp ***t***).

$$U_k(X,Y,t,V,D,S) \xrightarrow{Gis} \mathrm{S}_k(\mathrm{O}_i^j,V,D) \qquad \ldots (3.1)$$

3.3.2 Vehicle journey

In addition to ***TIS*** which indicates the traffic status at one fixed point, a vehicle journey represents the tracks of a vehicle starting from its origin to the destination. A vehicle journey, which partially reflects the traffic demand in the urban network, is a collection of consecutive ***TIS***s of a vehicle and can be extracted from the LBS raw data. For example, '***dispatched***' state journey extracted from TDS consists of a set of ***TIS***s which starts from the dispatched location to the customer's location, and '***occupied***' state journey starts from the customer's location to their destination.

Vehicle journey generation module extracts '***meaningful***' journey (indexed by ***k***) from the raw data collected from the LBS-based applications, for the TDS example, taxi that is in the '***dispatched***' or '***occupied***' states, where a journey of a vehicle consists of a set of consecutive ***TIS***s reported from the origin S_1 of the vehicle to its destination S_n, as illustrated in Equation (3.2).

$$J_k=<S_1,..,S_n> \qquad \dots (3.2)$$

3.3.3 Traffic network snapshot (***TNS***)

In order to enhance the global traffic network performance, network administrators have to realize the traffic status of the whole network. Traffic network snapshot (***TNS***) provides a global view of traffic status in a time period for the traffic network, which is indexed by spatial and temporal dimensions. Spatial domain groups the ***TIS***s by spatial area of the network object (e.g., link) and temporal domain groups the ***TIS***s by time zone (e.g., 15 minutes). Let a ***TNS*** be composed of a set of a spatiotemporal network objects during a ***15***-minute period. Therefore, the traffic status of urban network in morning workday peak hour (*7*~***9*** AM) includes eight snapshots. All the spatiotemporal traffic network objects and snapshots generated in this phase are stored in ***TIDB*** (traffic info. database) as the data source of the subsequent data mining processes.

TNS can be easily presented by map based user interface for reflecting a short period of network status by the assistance of the GIS. As illustrated in Figure 3-2, there are three continuous ***TNS***s of an urban network from *7***AM** to ***7:30*****AM**, where the objects marked as yellow lines indicate the congested links. With the knowledge of global network traffic status and continuous traffic status variations which can be presented by continuous ***TNS***s, the network administrators can decide which traffic assignment actions to be the best action by their domain expertise.

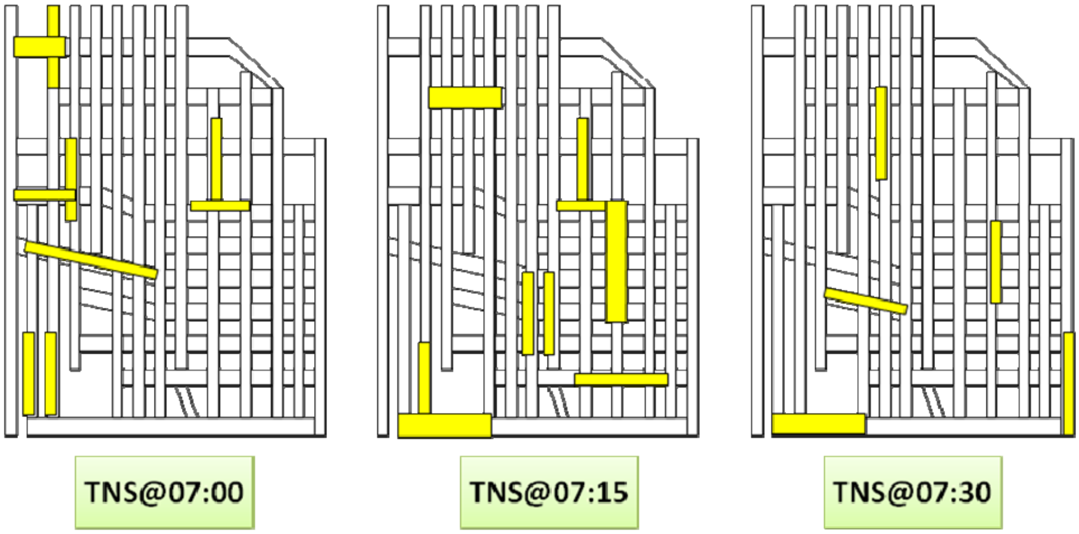

TNS@07:00 TNS@07:15 TNS@07:30

Figure 3-2 examples of three continuous *TNS*s (from 7AM~7:30AM)

3.3.4 Traffic information generation for freeway

The focus of traffic information for the freeway area is different from that for the urban area; travelers are more concerned about the traffic event messages and ramp to ramp traveling time than the route path choice in the freeway. On the contrary, in the urban area, travelers are concerned about the route path choices in order to avoid the congestion. Since the topology in the freeway is much different from that in the urban network, different sampling strategies are used to generate the traffic information depending on the location of the traffic information in freeway area or urban area. For example, if a vehicle is equipped with OBU or ***STA*** (smart traffic agent, discussed in the next chapter) traveling through the freeway, a series of samples uploaded from the same OBU/STA are collected by the backend traffic information center (***TIC***), and each sample can be transformed to the format of (M_x, T_x), where M_x indicates the mileage and T_x indicates the uploaded timestamp. By selecting the entry sample (M_s, T_s) and exit sample (M_e, T_e) of a vehicle, the average traveling speed ($\overline{V_{s,m}}$)

between these two ramps can be calculated by $(\boldsymbol{M_e}-\boldsymbol{M_s})/(\boldsymbol{T_e}-\boldsymbol{T_s})$, which is regarded as a case. For any two ramps $\boldsymbol{k}$ and $\boldsymbol{j}$, the overall average travelling speed can be calculated by the arithmetic mean value ($\overline{V_{k,j}}$) of for all the cases of ($\boldsymbol{k}$, $\boldsymbol{j}$) as shown in Equation (3.3).

$$\overline{\mathrm{V}}_{k,j} = \frac{\sum_{i=1}^{n}(T_{k,i} - T_{l,i})}{n \cdot (\left|\mathrm{M}_{k} - \mathrm{M}_{j}\right|)} \qquad \ldots (3.3)$$

3.3.5 Traffic information generation for urban network

Urban network consists of a set of network objects, each of which is either a link or an intersection where traffic congestions occur on some network objects in which the traffic demand cannot be fully serviced. The traffic status of the network which constitutes the network can then represent the traffic status of urban network.

(I) Traffic information of a link

The average speed of a link in a temporal period can be calculated by arithmetic mean value shown in Equation (3.3) of all the samples spatially falling in the link and temporally falling in the temporal range. However, in urban area, it seems not reasonable to represent link traffic status by the average speed for all the links because the service levels of different road grades are different. As shown in Table 3-1, the service levels of the three road categories in urban area of Taiwan are defined from level ***A*** to ***F*** [CLO01]. The link traffic information can be transformed to the service level by mapping the average speed to the road grade and service level defined in Table 3-1. For example, average speed of 30km/hr indicates the traffic status is good (level ***B***) in street (grade III), but slight congestion (level ***D***) in the expressway or arterial (road grade I).

Table 3-1 Service level classification for three categories of road in Taiwan

Road grade	Ⅰ	Ⅱ	Ⅲ
Free flow speed	**55 (kph)**	**45 (kph)**	**40 (kph)**
Service level	**Avg. speed (kph)**	**Avg. speed (kph)**	**Avg. speed (kph)**
A (90%)	~51	~43	~33
B (70%)	51~39	43~32	33~25
C (50%)	39~34	32~27	25~20
D (40%)	34~29	27~23	20~16
E (33%)	29~21	23~17	16~10
F (25%)	21~	17~	10~

(II) Traffic information of an intersection

On the other hand, the average intersection delay can represent the traffic information for the intersection, and the delay between two consecutive links, which is mostly caused by signal delay and queuing delay and can be classified by TD, LTD and RTD patterns according to the three possible directions from one link to the next link. Equation (3.4) shows the general format of intersection delays (TD/LTD/RTD), where $\boldsymbol{P}$ is the pattern type, $\boldsymbol{SO_{id}}$ and $\boldsymbol{SI_{id}}$ are the two consecutive links at the intersection where the vehicles leave out the link $\boldsymbol{SO_{id}}$ and come into the link $\boldsymbol{SI_{id}}$, $\boldsymbol{T_{id}}$ is the temporal id, $\boldsymbol{D_{avg}}$ is the average delay time of this intersection, and $\boldsymbol{Sup}$, $\boldsymbol{Con}$ are the support and confidence of the pattern, respectively.

[TD/LTD/RTD]: $(P, SO_{id}, SI_{id}, T_{id}, D_{avg}, Sup, Con)$ … (3.4)

For example, (‘*RTD*’, ‘*L1*’, ‘*L2*’, ‘*W,P*’, *40*, *0.2%*, *75%*) represents that in the peak hours of workday, it takes 40 seconds to do a right turn from link ‘*L1*’ to link ‘*L2*’, and the support is 0.2% , confidence is 75%. Intersection delay patterns can be discovered by sequential pattern mining or spatial and temporal sequence mining on all samples of intersection delay in a journey containing two consecutive samples with different links in the historical traffic information database. Figure 3-2 shows an example of RTD pattern: a probing vehicle driving north and then turning right to east, it reports ***TIS*** at location ***A*** of Link L_a and consecutively reports ***TIS*** at location ***B*** of Link L_b. The symbols of the ***TIS*** format (***T,L,X,Y,D,V***) in Figure 3-3 stand for timestamp (***T***), link id (***L***), coordinates (***X,Y***), direction (***D***) and speed (***V***), respectively. The distance d_a, d_b in Figure 3-2 stands for the distance from ***A*** or ***B*** to the intersection of links L_a or L_b respectively. Assuming that in the short period time interval between T_b and T_a, the vehicle is driving at the speed of V_a at link L_a and V_b at the link L_b. Then the right turn delay (RTD) time from L_a to L_b can be estimated by subtracting travel time of d_a and d_b from elapsed time between two ***TIS***s (T_b-T_a).

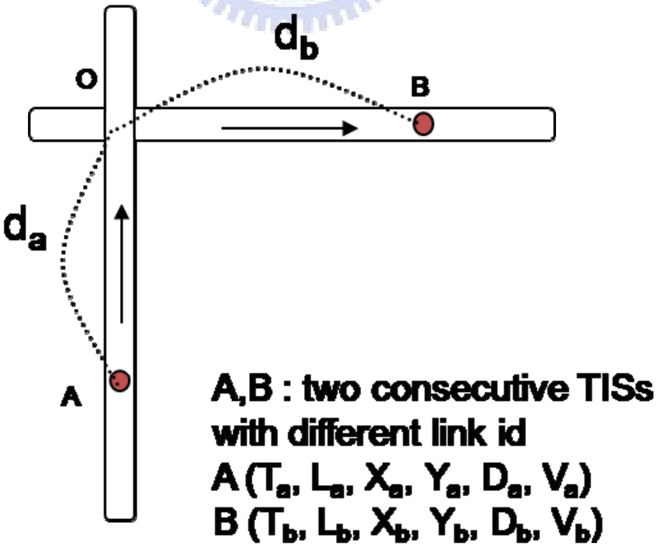


Figure 3-3 Intersection delay example: RTD

The status of each intersection delay in an intersection is also classified into six service levels (***A~F***) by normalizing each intersection delays, i.e., by equally dividing the intersection delay samples into six levels from the historical traffic information database. With the traffic status of all the links and intersections, the whole network status can be easily represented on map by coloring the objects in the user interface.

Chapter 4 Collaborative Traffic Information Generation and Sharing Framework

In addition to the generated traffic information from the LBS-based applications, there are two other traffic information sources: collaborative traffic information generation and external traffic information data sources integration. By integrating these two traffic information with the generated traffic information with LBS-based applications, a collaborative traffic information generation and sharing framework is proposed and discussed in this chapter.

More and more mobile devices such as mobile phones, PDAs are equipped with location capability by connecting to GPS module by wired or wireless scheme. On the other hand, more and more PNDs (personal navigation device) and UMPC (ultra mobile PC) which have location capability are now equipped with mobile communication module in order to retrieve the real-time traffic information. All these devices (GPS smart phone, PDA, PND, UMPC, etc.) have the features of mobility, communication and location capabilities. With the front end devices with these features, many modern mobile applications such as location-based service (LBS), telematics, and ITS related applications can be provided. On the basis of such high penetration of mobile communication devices with location capability, we proposed a wiki-like collaborative real-time traffic information generation and sharing framework for real-time traffic information collection, fusion and distribution. By this collaborative traffic information generation scheme, more real-time traffic data will be collected cost-effectively and accurately, and the spatiotemporal coverage is better than the traditional traffic information collection scheme.

As information exchanging and sharing through a public platform, e.g., the Wikipedia website is the major trend of Web 2.0 technology, our idea is to create a wiki-like real-time traffic information exchanging and sharing framework for different kinds of mobile devices users. Accordingly, a collaborative real-time traffic information collection, data fusion and distribution framework including a front-end smart traffic agent (***STA***), real-time traffic information exchange protocol (***TIEP***) and traffic data fusion and distribution backend center (***TIC***) is proposed. ***STA*** is a two-way real-time traffic information agent with a data exchange channel for location aware traffic information exchange through the ***TIEP*** is installed, and is also developed as a smart interface between the terminal and the driver for displaying the real-time and predictive traffic information, and providing the real-time traffic events reaction interface to share the local traffic information with other users.

Several external traffic information data sources including national traffic information center [EIOT], national freeway traffic information [TAN], and police station real-time traffic information [PRS] are integrated to the ***TIC*** in order to enhance the precision and coverage area, and an optimal weight traffic information fusion scheme is proposed to enhance the traffic information prediction capability. A knowledge based system combined with expert heuristic rules is also implemented in the ***TIC*** to dynamically decide the weight of different data sources including real-time collected traffic information, historical traffic data base, and multiple external traffic information data sources.

4.1 Architecture of the framework

The applications of the traffic information are generally divided into three stages, which are data collection and cleansing, data fusion and integration, and data distribution. In the data collection and cleansing stage, raw data is collected from different data sources, filtered and analyzed in the backend system. The collected data are then integrated with the geographical information system (GIS) and external traffic information data sources in the data fusion and integration phase. In the data distribution phase, several interfaces including web, radio, mobile subscribe, etc. are provided for users, where the mobile subscription scheme which is suitable for the users with mobile communication device can retrieve the real-time traffic information provided by the ***TIC*** according to the device location.

Three key components, front-end ***STA***, ***TIEP*** protocol, and backend ***TIC*** constitute the traffic information generation and sharing framework, where ***TIEP*** are designed for the location aware real-time traffic information exchanging between the ***STA*** and ***TIC***, which enables ***STA*** to automatically echo the local traffic information detected by ***STA*** to the ***TIC***, automatically download the local real-time and predictive traffic information, and react the traffic events or traffic status to the ***TIC*** by the traveler. As shown in Figure 4-1, the architecture of the collaborative traffic information generation and distribution framework includes three parts: location aware mobile device with ***STA***, ***TIC*** for data collection and fusion, and external traffic information data sources, where various versions of ***STA***s are developed for different front-end mobile devices depending on the different hardware and operating system, the generated real-time traffic information can be distributed by ***STA*** (request and response model), radio broadcast network (RDS), web site, etc, and ***TIC*** also connects to external real-time traffic information data sources, including e-IOT national traffic events database [EIOT], TANFB freeway traffic information [TAN], PRS public traffic service [PRS], and parking slot database.

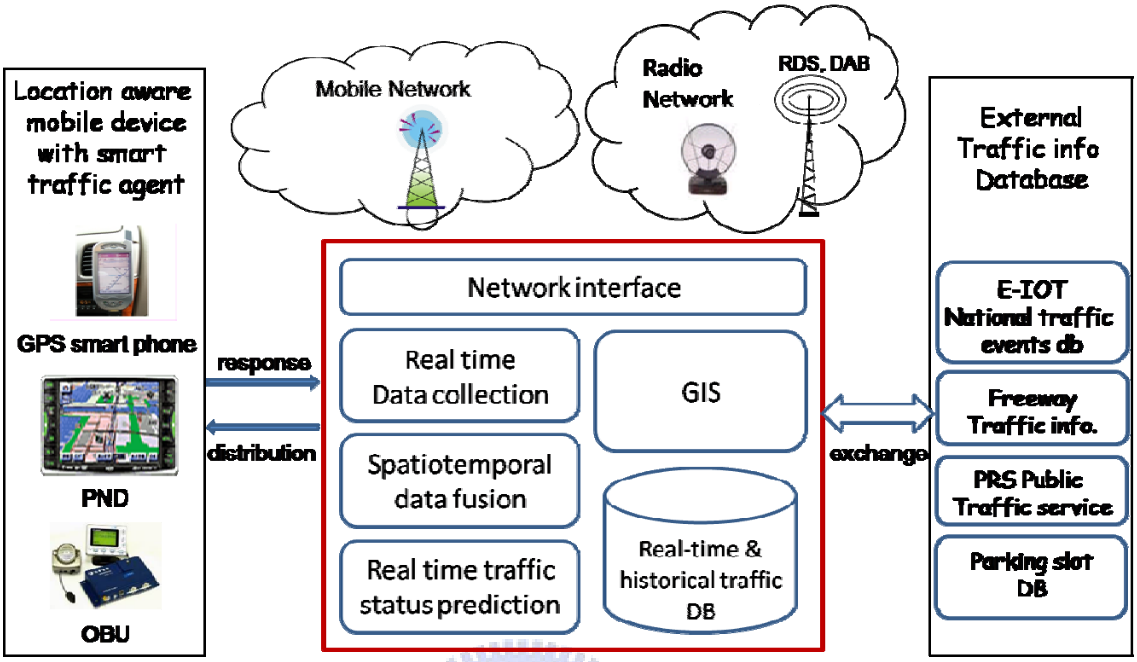


Figure 4-1 Collaborative traffic information generation and distribution framework

4.2 Smart traffic agent (***STA***)

STA, a software agent for the front-end devices, provides an interface between the traveler and the ***TIC***. It is installed on the front-end device with a two-way real-time data exchange channel based upon ***TIEP*** to 1) automatically echoing the local traffic information detected by ***STA*** to the ***TIC***, 2) automatically downloading the local real-time and predictive traffic information, and 3) manually reacting the traffic events or traffic status to the ***TIC*** by the traveler. The ***TIEP*** is based upon request-response model, where the request message includes GPS coordinate, moving direction, speed traffic status information, and request scope is defined in XML format to be sent to the ***TIC***. The frequency and scope of exchange the local traffic information can be determined by ***STA***, which is calculated depending on the traveling speed and moving direction. The higher moving speed of the ***STA***, the request scope is larger and the frequency is shorter in order to retrieve more up-to-date and more suitable traffic information.

The response message from ***TIC*** is also defined in XML format, which is a list of traffic messages consisting of two types: text message information and speed information. The former informs the traveler of the local traffic messages consisting of <***no***, ***message type***, ***data source***, ***message***, ***severity***, ***coordinate***> such as traffic events, CMS. The latter informs the current traveling speed and predictive traffic condition consisting of <***no***, ***type***, ***link/intersection id***, ***avg. speed***, ***current status***, ***predictive status***, ***coordinates***>. The response messages received is used to update the both the text and map in the user interface of ***STA***, which helps the traveler easily catch the local real-time and predictive traffic information. Real-time traffic status and predictive traffic status are combined in the ***TIC*** response message by considering the predictive ***STA*** location. As illustrated in Figure 4-2, after an ***STA*** request is sent from location ***A***, ***TIC*** responds the real-time (doted small rectangle) and predictive (doted large rectangle) traffic messages centralized by predictive location ***B*** by considering the moving direction, moving speed, and response packet size (response time). In the dotted small circle centralized by predictive ***STA*** location at point ***B***, the ***TIC*** responding message contains only real-time traffic information (average speed, traffic events) since the predictive information makes less sense to the current status. In the large dotted rectangle, both real-time and predictive traffic information is contained in the response message to inform the drivers of the current status and predictive traffic status around the ***STA***.

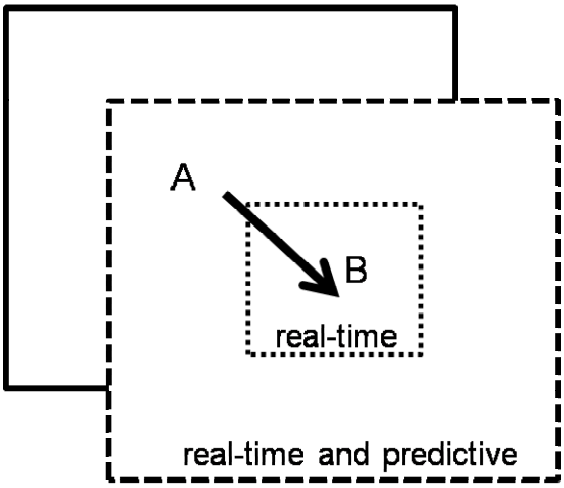
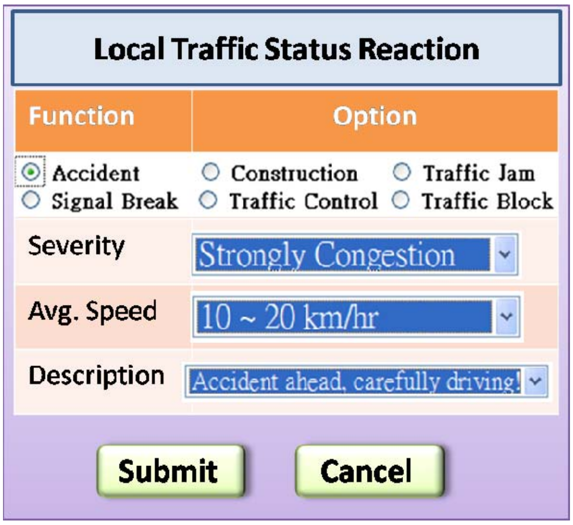


Figure 4-2 TIC real-time and predictive traffic information response message

There is a wiki-like user centric traffic information sharing mechanism in the framework, which provides a user interface in ***STA*** for traveler to share the local events or traffic status to other travelers via the framework, as shown in Figure 4-3. The reacting traffic event or status won’t be distributed instantly until it is confirmed by another traveler (through the same mechanism) or confirmed by the external data sources. The philosophy of this double check mechanism is to prevent from sending the fake messages.



Local Traffic Status Reaction

Function	Option	
● Accident	○ Construction	○ Traffic Jam
○ Signal Break	○ Traffic Control	○ Traffic Block
Severity	Strongly Congestion	
Avg. Speed	10 ~ 20 km/hr	
Description	Accident ahead, carefully driving!	

Figure 4-3 User centric traffic status reaction user interface in *STA*

4.3 Multi-sources heterogeneous traffic information fusion

In addition to the collected traffic information, several external real-time traffic information data sources are integrated in this framework, including E-IOT (national traffic information center) [EIOT], TANFB (Taiwan Area National Freeway Bureau) freeway traffic information system [TAN], public traffic service of PRS (police radio station) [PRS], and several parking slot databases are also used in the framework to retrieve more plentiful driving assistance information such as real-time CMS (changeable message sign) information, parking slot information, etc. Accordingly, the traffic information is sufficient for not only reporting real-time traffic information but also predicting the traffic status in the framework which can be further fed back to these external databases for sharing the information with the travelers without ***STA***.

In order to integrate the required external real-time traffic information data sources with the generated traffic information in the framework, an optimal weighting combination scheme based on Shannon entropy theory [SW64] for traffic data fusion approach [WU05] is adopted. The internal traffic information in the framework and each of the external traffic data source is regarded as a data source ($\boldsymbol{S_i}$). The weighted combination fusion equation for a network object is shown in Equation (4.1), where $\boldsymbol{O_j}$ can represent any attribute of a network object such as link average speed, intersection delays (TD/LTD/RTD), and W_i is the weight for the data source $\boldsymbol{S_i}$.

$$\overline{O_j} = \sum_i W_i \cdot S_i(O_j),$$
$$\sum_i W_i = 1 \qquad \ldots (4.1)$$

The Shannon entropy is adopted to measure the uncertainty and randomness of the collected data and external data sources, and the optimal weighting approach tries to minimize the overall uncertainty by assigning the lower weight to higher entropy data sources. The entropy with conditional probability of data source **S_i** , denoted as $h_i(\gamma)$, is defined as Equation (4.2), where $P(\theta_j|\gamma)$ is the probability of occurrence of the state θ_j given the observed state γ . The probability can be summarized from the historical traffic database, for example, the probability of service level '***C***' on road **L_j** can be calculated by the speed samples which are located in **L_j** and falls in the range of level '***C***' in the road category of **L_j** divided by the total samples located in **L_j**. The weight assignment equation in optimal weighting approach [WU05] is shown as in Equation (4.3), where ***S*** is the set of all data sources. It shows that the weight $W_i(\gamma)$ has a negative relationship with $\mathrm{h}_i^2(\gamma)$, so that data source ***i*** will be assigned a lower weight **W_i** if the entropy **h_i** (uncertainty) is high.

$$h_i(\gamma) = -\sum P(\theta_j|\gamma)\log P(\theta_j|\gamma) \qquad \ldots (4.2)$$

$$W_i(\gamma) = \frac{1}{\mathrm{h}_i^2(\gamma)\sum_{i\in S}\frac{1}{\mathrm{h}_i^2(\gamma)}} \qquad \ldots (4.3)$$

However, there are several traffic characteristics in temporal dimension such as peak hour, weekday, or midnight, etc. Without the consideration of temporal traffic characteristics, the probability calculation in the entropy $h_i(\gamma)$ equation defined in the Equation (4.2) may lose the precision for the “optimal weight” calculation. For example, traffic status characteristics in the workday are different from those in the holiday, and traffic characteristics in the ***AM*** peak hours are not the same as those in the ***PM*** peak hours. In order to take the traffic status characteristic, temporal dimension into consideration, we divide the collected samples in the historical traffic information database into five periods: {***AM*** peak hour, ***PM*** peak hour, normal, holiday, mid-night}. The probability of occurrence of the state θ_j given the observed state γ is modified as $\mathrm{P}_k(\theta_j^k|\gamma)$, where k is index for the temporal periods, and the entropy defined in the Equation (4.2) is then modified as in Equation (4.4), where $h_i^k(\gamma)$ denotes the entropy of network object ***i*** at temporal index ***k***.

$$h_i^k(\gamma) = -\sum P_k(\theta_j^k|\gamma)\log P_k(\theta_j^k|\gamma) \qquad \ldots (4.4)$$

4.4 Knowledge based traffic status predicting

The traffic information and event messages collected from the ***STA*** and external traffic data sources are used not only for message informing but also for the traffic status prediction. For each ***STA*** traffic information request, the prediction result is sent to the ***STA*** accompanied with the real-time traffic information, as illustrated in Figure 4-2. The traffic status prediction is implemented by the knowledge based system technology, in which heuristic rules are elicited from domain experts to enhance the precision of traffic status prediction. There are many rules for the traffic event to be considered for the traffic status prediction on an ***STA*** request, such as event type, severity, event elapsed time, and spatial related to the ***STA***, etc. For example, a traffic accident event may have severe impact to the near around ***STA***s, but the severity will be reduced as the distance increases.

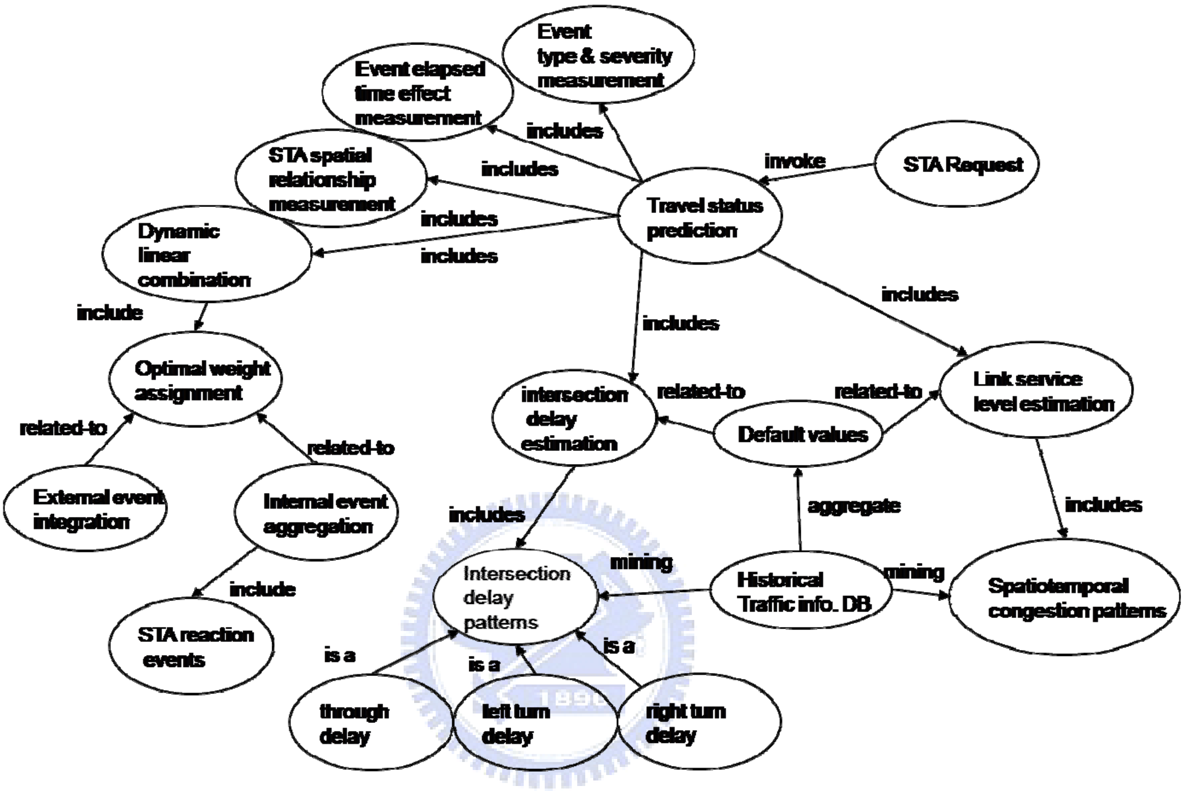


Figure 4-4 Design ontology for the traffic status prediction knowledge based system

The design ontology of the traffic status prediction knowledge base system is illustrated in Figure4-4, which includes several design concepts and relationships between these concepts, such as "***is-a***", "***includes***", "***modify***", "***related-to***", etc. These relationships connecting concepts in the ontology represent some interactions among them. For example, "***traffic status prediction***" concept includes several concepts such as "***dynamic linear combination***", "***intersection delay estimation***", "***link service level estimation***", and three traffic event factor measurement concepts. When the system receives an ***STA*** traffic status request, the "***traffic status prediction***" concept is then "invoked" to process the whole traffic status prediction process, which again invokes those concepts included in the "***traffic status prediction***" concept. The optimal weight assignment module integrates the internal ***STA*** reaction events and the external traffic event information sources by optimal weighting linear combination scheme. On the other hand, the real-time link service level and intersection delays estimation modules retrieve the current link service level and intersection delay information from the real-time traffic information database. "***Traffic status prediction***" returns the current traffic status near around the ***STA*** (real-time area in Figure 4-2) and predict the traffic status (real-time and prediction area in Figure 4-2) by fine tuning the current traffic status and considering the localized ***STA*** and traffic event impacts consideration modules including "***event type and severity measurement***", "***event elapsed time effect measurement***", and "***event location to STA spatial relationship measurement***".

The "***event type and severity measurement***" module evaluates the effect of the event to the ***STA*** by its type and severity and tunes the real-time evaluation result, for example, the service level prediction of the link and related intersections near around the traffic accident event may be downgraded one or two levels than the current link service level and intersection delay estimation. The "***event elapsed time effect measurement***" and "***event location to STA spatial relationship measurement***" modules tunes the link service level and intersection delay by considering the temporal distance (event elapsed time from the ***STA*** request time) and spatial relationship (distance from the event to the ***STA***) respectively. The traffic status prediction knowledge based prototype system has been implemented by the expert system shell, DRAMA [CO03], a NORM (New Object-oriented Rule Model) [LTT03] knowledge modeled rule base system platform implemented using pure Java language, includes Drama Server, Console, Knowledge Extractor, and Rule Editor. Heuristic rules donated by the traffic domain experts are categorized to several knowledge classes (KC), and stored at knowledge base of DRAMA server.

Chapter 5 Spatiotemporal Traffic Patterns Mining

All the traffic information collected from the LBS-based applications, framework and external traffic sources are stored in the ***TIDB***, which is the data source of spatiotemporal traffic patterns (***STP***) mining and spatiotemporal traffic bottleneck (***STB***) mining. All the traffic information and traffic events in ***TIDB*** have the attributes of location and timestamp information, so the spatial and temporal relationship between different traffic objects can be discovered by spatiotemporal data mining techniques.

Spatiotemporal congestion patterns are mined from the ***TIDB*** by integrating the traffic network database in the GIS engine, which are categorized by object level patterns and area level patterns. The former considers about the frequent behavior of single network object (discussed in Section 5.1), while the latter tries to find out the congestion area and the relation between different areas (discussed in Section 5.3). The spatiotemporal congested areas are discovered by the spatial heuristic clustering algorithm, which is discussed in Section 5.2.

5.1 Object level traffic patterns mining

Traffic network consists of a set of network objects, each of which is either a link or an intersection where traffic congestions only occur on some network objects by which the traffic demand cannot be fully serviced. Unfortunately, the demand of the network objects is never known in advance and the capacity of the network objects is hardly available, which makes the identification and prediction of network objects congestion impracticable. According to the expert's heuristic, the ratio of average speed and speed limit of a network object has a negative impact to the ratio of the demand and its capacity. The lower of the former ratio value indicates that the network object is in the higher congested situation. Thus traffic status of a spatiotemporal object (***STO***, notated by O_i^j, network object ***i*** in time zone $\boldsymbol{T_j}$) can be calculated by dividing the average speed of all the ***TIS***s during the same time zone $\boldsymbol{T_j}$ by the speed limit of the network object. Let traffic index factor (***TIF***), $\theta(O_i^j)$, denote the normalized traffic status of O_i^j, as listed in Equation (5.1), where $\overline{\mathrm{V}_\mathrm{i}^\mathrm{j}}$, $\boldsymbol{L_i}$ are the average speed and speed limit of the object O_i^j respectively. The higher the $\boldsymbol{\theta}$, the more serious congestion level of the object is, for example, $\boldsymbol{\theta}$=1 indicates the object is in a serious congestion status and θ near around zero means the object is in a free flow status.

$$\theta(O_i^j) = 1 - \frac{\overline{V_\mathrm{i}^\mathrm{j}}}{L_i} \qquad \text{... (5.1)}$$

The traffic status of urban network can be aggregated to snapshot by spatial and temporal domains, where spatial domain groups the ***TIS***s by spatial area of the network object and temporal domain groups the ***TIS***s by time zone, for example, 15 minutes. Let a traffic network snapshot (***TNS***) be composed of a set of a spatiotemporal network objects during a 15-minute period. Therefore, the traffic status of urban network in morning workday peak hour (7~9 AM) includes eight snapshots. All the spatiotemporal traffic network objects and snapshots generated in this phase are stored in ***TIDB*** as the data source of the subsequent phases.

5.1.1 Spatiotemporal Aggregation Pattern (***SAP***)

The spatiotemporal aggregation pattern (***SAP***), also named as congested object item (***COI***), represents the network object which fulfils two threshold criteria: (1) traffic congestion threshold (H_{θ}), and (2) congestion confidence threshold ($\mathbf{H}_c$). ***STP*** are mined from historical traffic database by aggregating the ***TIS***s by spatial, temporal, and event dimensions, classifying the congestion level of the link by the attributes and traveling speed of that link, and calculating the support and confidence for each ***SAP***. Spatial dimension stands for the link identification attribute in the urban road network, and temporal dimension is the classified indices of time domain (identified by P_k), which includes: ***AM*** peak or ***PM*** peak hour, holiday or workday, etc. Congestion level is determined by the ratio of average speed and speed limit of the link in the same spatial and temporal condition. For example, average speed of 40 km/hr in the ***PM*** peak hour in workday is classified as free flow state in urban street, but will be classified as strongly congestion in freeway. Support and confidence can be calculated by aggregating the ***TIS***s at the same spatiotemporal conditions. The format of ***SAP*** is listed as in Equation (5.2): S_{id}, stands for link id, T_{id} is temporal id, E_{id} means the event condition of the pattern, such as normal, car accident, road construction, etc. C_g stands for congestion level, which is normalized by the link attribute (speed limit, link type) of target link (S_{id}); S_{up} and C_{on} stand for support and confidence of the ***SAP*** respectively. An example of ***SAP*** (*'L1', 'W,N', 'N', 9, 0.3%, 65%*) means that the link [*L1*] is in free flow state (congestion level ***9***) at non-peak hours of workday and the confidence of this pattern is 65%, support is 0.3%.

[SAP]: (S_{id}, T_{id}, E_{id}, C_g, S_{up}, C_{on}) … (5.2)

For the network objects where the ***TIF*** (θ) larger than the threshold (H_θ), which are named as spatiotemporal congested object (STCO), congestion confidence threshold filter is then applied to each STCO to discover the ***SAP***, as illustrated in Equation (5.3). The confidence of network object O_i^j at temporal period $\boldsymbol{P_k}$ is calculated by the ratio of the congested samples of the spatiotemporal network object O_i^j over total samples in the temporal period $\boldsymbol{P_k}$.

$$\mathrm{Conf}(\mathrm{O}_\mathrm{i}^\mathrm{j}, k) = \frac{\left|\{O_i^j \middle| T_j \in \mathrm{P}_k, \theta(\mathrm{O}_\mathrm{i}^\mathrm{j}) > \mathrm{H}_\theta\}\right|}{\left|\{O_i^j \middle| T_j \in \mathrm{P}_k\}\right|} \quad \text{… (5.3)}$$

For the ***SAP***, some traffic regulation actions can be taken to alleviate the congestion, such as extend the green time signal control, enforce the reversible lane, etc.

5.1.3 Congestion Drop Pattern (*CDP*)

The basic idea of ***CDP*** is to calculate the significant congestion difference between an object and its downstream objects, which indicates the bottleneck level of the object. In an typical intersection, an object usually has three downstream objects, as illustrated in Figure 5-1, {$\boldsymbol{O_{i,1}}$, $\boldsymbol{O_{i,2}}$, $\boldsymbol{O_{i,3}}$} are the downstream objects of $\boldsymbol{O_i}$ (upstream object), which are the three candidate moving directions while a vehicle leaving out the object $\boldsymbol{O_i}$.

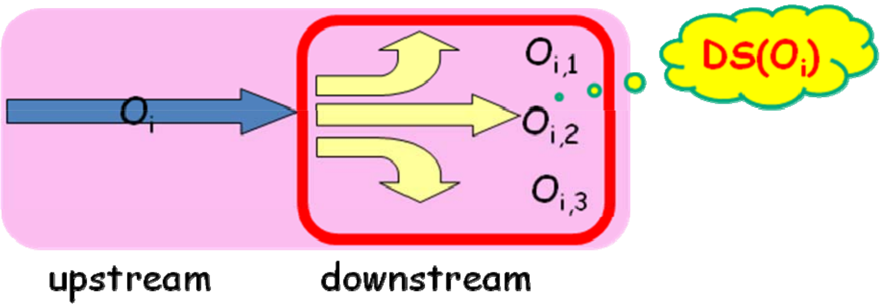


Figure 5-1 Congestion drop pattern concept

The definition of congestion drop ratio (***CDR***, notated as τ) is illustrated in Equation (5.4), which calculates the difference of ***TIF***(θ) between object O_i^j and the average θ of its *m* downstream objects ($\{O'^j_1, O'^j_2, ..., O'^j_m\}$).

If the ***CDR*** (τ) of an object is closed to ***1***, it indicates the traffic congestion of the object is more serious than its downstream objects. On the other side, the traffic congestion is more serious in downstream objects than itself if the ***CDR*** of the object is smaller than ***0***. When the ***CDR*** of an object is larger than the congestion drop threshold (***H_d***), then this object is regarded as a ***CDP*** object

$$\tau(O_i^j) = \theta(O_i^j) - \frac{\sum_{k=1}^{m} \theta(O'^j_k)}{m} \quad \dots (5.4)$$

where $\{O'^j_1, O'^j_2, ..., O'^j_m\}$ is the set of downstream objects of O_i^j

5.1.4 Intersection Delay Pattern (***IDP***)

Intersection delay is the delay between two consecutive links (i.e, upstream object and downstream object), as illustrated in Figure 5-1, which is mostly caused by signal delay and queuing delay. Intersection delays are categorized by through delay (TD), right-turn delay (RTD), and left-turn delay (LTD), which indicate the three possible directions from one link connected to its downstream links respectively. The ***IDP*** samples can be retrieved from two consecutive ***TIS***s of a journey with different link by calculating the link travel time and intersection delay. As illustrated in Equation (5.5), ***P*** is the pattern type (TD/LTD/RTD), $\boldsymbol{SO_{id}}$ and $\boldsymbol{SI_{id}}$ are the two consecutive links ID from the link $\boldsymbol{SO_{id}}$ to the link $\boldsymbol{SI_{id}}$, $\boldsymbol{T_{id}}$ is the temporal *id*, $\boldsymbol{D_{avg}}$ denotes the average delay time of this intersection, and ***Sup***, ***Con*** are the support and confidence of the pattern, respectively. For example, (‘*RTD*’, ‘L_1’, ‘L_2’, ‘*W,P*’, *40*, *0.2%*, *75%*) represents that in the peak hours of workday (‘*W,P*’), it takes 40 seconds to do a right turn from link ‘L_1’ to link ‘L_2’, and the support is 0.2% , confidence is 75%.

$$\textbf{IDP} : (P,\ SO_{id},\ SI_{id},\ T_{id},\ D_{avg},\ S_{up},\ C_{onf}) \qquad \ldots(5.5)$$

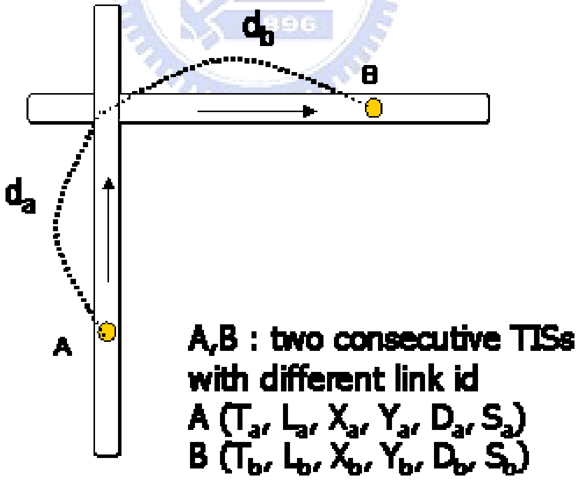


Figure 5-2 Intersection delay example: RTD

Intersection delay patterns can be discovered by sequential pattern mining on spatial and temporal sequences in ***TIDB***. Each sample of intersection delay must be in a journey which contains two consecutive ***TIS***s with different links. Figure 5-2 illustrates an example of RTD pattern: a probing vehicle driving north and then turn right to east, it reports ***TIS*** at location ***A*** of Link L_a and consecutively reports ***TIS*** at location ***B*** of Link L_b. The symbols used in the ***TIS*** format (***T,L,X,Y,D,V***) in Figure 5-2 stand for timestamp (***T***), link id (***L***), coordinates (***X,Y***), direction (***D***) and speed (***V***) respectively. The distances d_a, d_b in Figure 5-2 stand for the distance from ***A***, ***B*** to the intersection of links L_a and L_b respectively. Assuming that in the short period time interval between T_b and T_a, the vehicle is driving at the speed of V_a at link L_a and V_b at the link L_b. There are at most twelve intersection delays in a two way cross intersection object, including TD/LTD/RTD for each E/W/S/N direction. For example, the right turn delay (RTD) time from the L_a to L_b can be estimated by subtracting travel time of d_a and d_b from elapsed time between two ***TIS*** (T_b-T_a).

5.2 Spatial heuristic clustering (***SHC***)

Since the granularity of object level pattern is too small to reflect the overall traffic network status, it is difficult to reflect the statistic meaning of ***O-D*** (origin destination) traffic demand. A set of STCOs which are in neighborhood area in the same snapshot can be clustered into a spatiotemporal clustered area (***STCA***) to solve this problem, so that area level of traffic patterns can be discovered to realize the relationship between ***O-D*** traffic demand and spatiotemporal congestion.

A heuristic STCO clustering algorithm (***SHC***), as illustrated in Figure 5-3, is developed for the spatiotemporal clustering. In ***SHC*** algorithm, the top k congested STCOs in the snapshot N_j are selected as the seeds of the candidate clusters. At the initial of each cluster, it searches and joins the neighborhood congested objects nearby the seed by querying 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 the threshold reached.

Algorithm 5-1 Spatial Heuristic Clustering (*SHC*) Algorithm

```
Spatial Heuristic Clustering (SHC) Algorithm
Denotation :
$N_j$ : the snapshot in temporal period $T_j$
$< O_1^j, O_2^j, ..., O_k^j >$ denote the top $k$ congested STCOs in $N_j$
$C_i^j$ : the $i$-th cluster in Snapshot $N_j$
$C_1^j$ ={STCO with $\max_i \theta(O_i^j)$ }
$H_\theta$: the congestion threshold of $\theta$
Input : Snapshot $N_j$, $i$=1
Output : Cluster Set { $C_1^j, C_2^j, C_3^j, ...$ }
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
```

5.3 Area level traffic patterns mining

5.3.1 Congestion Propagation Pattern (***CPP***)

Congestion propagation pattern, notated as ***CPP***($\boldsymbol{A_j}$, $\boldsymbol{B_k}$), is that there is root cause congestion relationship between STCA ***A*** in time zone $\boldsymbol{T_j}$ and STCA ***B*** in the time zone $\boldsymbol{T_k}$, where $\boldsymbol{j \leq k}$.

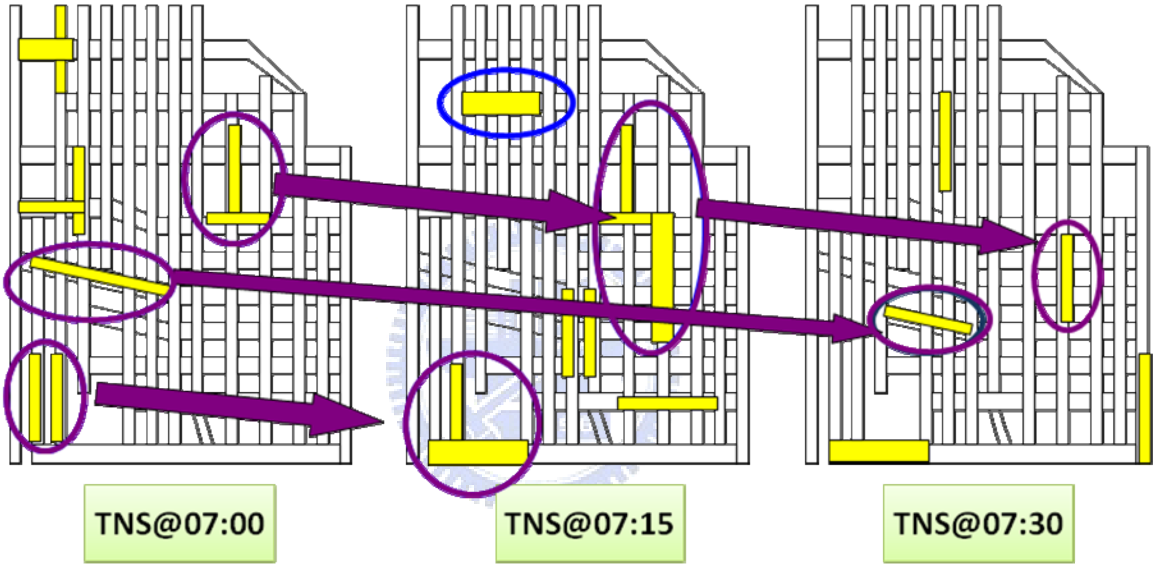


Figure 5-3 Example of *CPP* discovering algorithm

To discover the congestion propagation relation between two STCAs, the ***CPP*** discovering algorithm is illustrated in Figure 5-3, where the variable $\boldsymbol{\Delta_t}$ indicates the temporal distance between the snapshot $\boldsymbol{N_i}$ and its consequent snapshots, 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.

- Spatial boundary threshold (H_u^s): it is positively related to temporal distance $\boldsymbol{\Delta_t}$. For example, in the case of $\Delta_t = 1$, $\mathrm{H}_1^s = 5$ the congestion originated from one object in snapshot N_0 may propagate **5** network objects distance away in **15** minutes in snapshot N_1. In the case of $\Delta_t = 2$, $\mathrm{H}_1^s = 12$ the congestion may propagate **12** network objects distance away in snapshot N_2 from the origin congested object after **30** minutes ($\Delta_t = 2$), which indicates the congestion in one object may result from the other object, where the spatial distance between these two objects is within at most **12** objects distance and the temporal distance is within **30** minutes. The spatial boundary increasing with temporal distance concept matches the traffic congestion propagation experience.

- ***DOR*** threshold (H_u^d): it is negatively related to Δ_t because traffic demand scatters. It decreases as the temporal distance increases.

As illustrated in Equation (5.6) ~ (5.9), ***DOR*** function (notated as σ) defines the journey overlapped ratio of two STCAs A_i^j and A_k^l, where $J_m = <S_1, .., S_n>$ is a journey of a vehicle indexed by m which is originated from S_1 and ended at S_n. In Equation (5.7), $J_m \propto \mathrm{A}_k^l$ is a relationship which indicates the journey J_m is spatiotemporally overlapped with A_k^l, i.e., that some ***TIS***s of the journey J_m are located in STCA A_k^l. λ function defines the ‘passing by’ concept as in Equation (5.8), which finds out the minimum index of ***TIS***s in the sequence of J_m that is overlapped with A_k^l. In Equation (5.9), σ function determines the ***DOR*** of the two STCAs A_i^j 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.

$$J_m = \langle S_1,.., S_n \rangle \qquad \cdots \quad (5.6)$$

$$J_m \propto A_k^l \Leftrightarrow \exists\, i \in \{1,..,n\} \text{ s.t. } S_i \in A_k^l \qquad \cdots \quad (5.7)$$

$$\lambda(J_m, \mathrm{A}_k^l) = \min\left\{ i \,\middle|\, A_k^l \propto J_m, S_i \ in\ J_m, \text{ and } S_i \ in\ A_k^l \right\} \qquad \cdots \quad (5.8)$$

$$\sigma(A_i^j, A_k^l) = \frac{\left| \left\{ J_m \middle| A_i^j \propto J_m, A_\mathrm{k}^1 \propto J_m, \lambda(A_i^j, J_m) < \lambda(A_k^l, J_m) \right\} \right|}{\left| \left\{ J_m \middle| A_\mathrm{k}^1 \propto J_m \right\} \right|} \qquad \cdots \quad (5.9)$$

Algorithm 5-2 *CPP* Patterns Mining Algorithm

Congestion Propagation Pattern Mining

Denotation:

A_i^j : the i-th STCA in snapshot $\mathrm{N_j}$

Δ_t : the temporal distance between two snapshots

$\mathrm{H_t}$: the temporal boundary threshold

N=$\{N_j, N_{j+1}, \ldots, N_{j+Ht}\}$ the set of traffic network snapshots within the temporal boundary threshold

H_u^s : the spatial boundary threshold under the temporal distance Δ_t

H_u^d : threshold of DOR under the temporal distance Δ_t

Input:

snapshots $\{N_j,..,N_{j+Ht}\}$

```
for each  $A_i^j \in \mathrm{N_j}$
    for each  $A_k^l \in \mathrm{N_m}$, m=j, j+1,.., j+$\mathrm{H_t}$
$\Delta_t = l-j$
/* spatial boundary threshold ($\mathrm{H}_u^s$) filtering */
if spatial distance ($\mathrm{A}_i^j, A_k^l$) is smaller than  $\mathrm{H}_{\Delta_t}^s$
then
/* DOR threshold ($\mathrm{H}_u^d$) filtering */
if  $\sigma(A_i^j, A_k^l) >$  $\mathrm{H}_{\Delta_t}^s$
then output ($\mathrm{A}_i^j, A_k^l$) as a CPP
```

5.3.2 Demand Conflict Pattern (***DCP***)

DCP is constructed on the basis of ***CPP*** and ***IDP***. The concept of ***DCP*** is that if there exists two or more ***CPP***s which have the spatial conflict traffic demand, then it may result in some bottlenecks in the cross area of these two traffic demands. As illustrated in Figure 5-6, there are two ***CPP***s [***A,C***] and [***B,D***]: traffic demand of ***C*** comes from ***A***, and traffic demand of ***D*** comes from ***B***, which has a spatial demand conflict area as dotted circle in Figure 5-4. The determination of the spatial conflict demand of two ***CPP*** needs the support of GIS engine with 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 the intersection objects in the spatial demand conflict area (dotted circle in Figure 5-4).

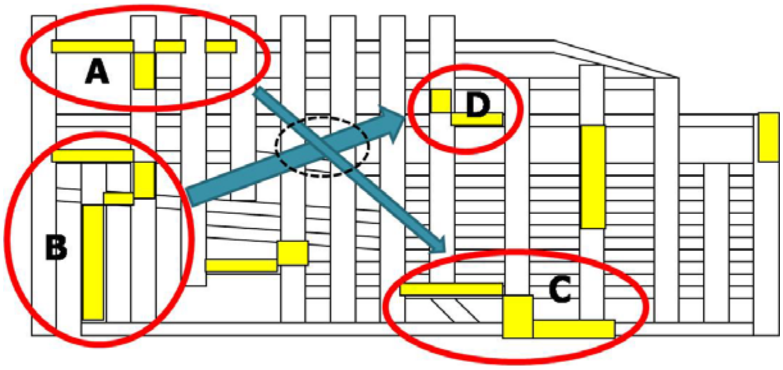


Figure 5-4 *DCP*: most likely existing *STB* in the spatial cross demand of two *CPP*s (black dotted circle area)

By 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, traffic demand in the two ***CPP*** patterns may go across each other in the intersection object, and result in two kinds of intersection delays: cross conflict delay and left turn interlaced delay, as illustrated in Figure 5-5(a) and 5-5(b) respectively. Cross conflict delay indicates the two traffic demands go cross 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.

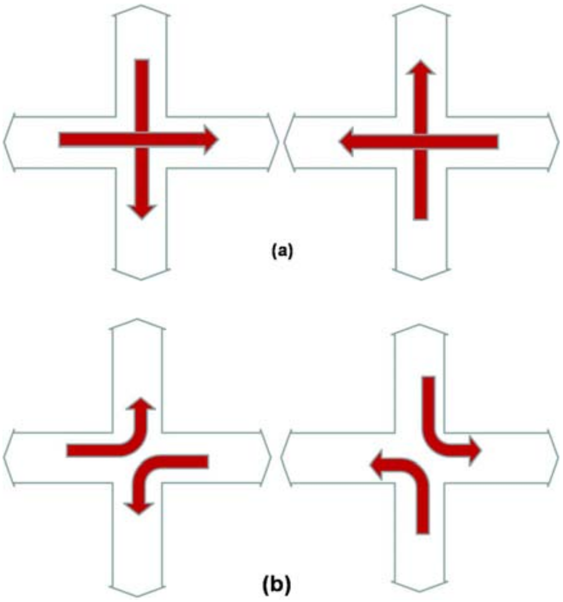


Figure 5-5 (a) cross conflict delay, (b) left turn interlaced delay

Chapter 6 Spatiotemporal Traffic Bottlenecks Discovering

Spatiotemporal traffic bottleneck (***STB***), which indicates when and where the network object is in congestion status, may result from overloaded traffic demand, and may be the root cause of other related neighborhood congestions. Once an ***STB*** is discovered, some actions including traffic signal control, changeable message sign and traffic assignment can be taken to relieve the congestion. Therefore, discovering the ***STB*** and taking appropriate actions to improve the overall traffic performance is the major task of official traffic administration agency.

In this Chapter, a process for spatiotemporal traffic bottlenecks discovering is discussed to find out ***STB***. Based upon the traffic patterns defined and discovered in the Chapter 5, ***STB*** can be discovered by hypothesis based analysis. Three ***STB*** discovering heuristics, Congestion-Propagation Heuristic (***CPH***), Congestion-Converge Heuristic (***CCH***), and Congestion-Drop Heuristic (***CDH***) are proposed in this chapter, which are detailedly discussed in Section 6.3.

6.1 Spatiotemporal traffic bottlenecks (***STB***)

As discussed in Chapter 5, traffic bottleneck can be defined as an object in traffic network where and when its capacity is less than the traffic demand, and the congestion in this object may be the root cause of neighborhood congestion. For example, a traffic bottleneck may generate a queue of vehicles propagated to the surrounding network objects, blocking intersections and result in congestion chaining. However, the traffic demand varies with spatial and temporal environments, so does the traffic bottleneck. Traffic bottlenecks in the urban network are not easy to define and discover because they may vary with spatial and temporal environment. For example, traffic bottlenecks in the workday are different from those in the holiday, and traffic bottlenecks in the ***AM*** peak hours are not the same as those in the ***PM*** peak hours.

In this work, spatiotemporal traffic bottleneck (***STB)*** is defined and used instead of traffic bottleneck in order to clearly identify where and when the traffic bottleneck could occur. An ***STB*** is a traffic object with spatial as well as temporal identification, which indicates when and where a network object could be congested and the congestion may propagate to the neighborhood objects. In other words, ***STB*** may be the root cause of other related neighborhood congestions, and the influence of the ***STB*** is much larger than the other congested objects. Once an ***STB*** is discovered, some actions including traffic signal control, changeable message sign and traffic assignment can be taken to relieve the congestion. Therefore, discovering the ***STB*** and taking appropriate actions to improve the overall traffic performance is the major task of official traffic administration agency.

6.2 Congestion area sequence rule (***CASR***)

The congestion propagation pattern (***CPP***) discussed in the Section 5.3, which depicts the relationship between two congested area, can be transformed into a congestion area sequence rule (***CASR***) directly. ***CASR*** is defined as a three tuples vectors as illustrated in

Equation (6.1), which is a traffic demand oriented congestion association rule between two ***STCA***s in neighborhood snapshots, where A_i^j, A_k^l indicates two ***STCA*** located at snapshot N_j, N_l respectively, confidence (c_{onf}) of the ***CASR*** is the ratio of the days where their ***DOR*** (σ, defined in Equation 5.8) of these two ***STCA***s is larger than the threshold H_u^d .

$$R = < A_i^j , A_k^l , C_{onf} >, \quad C_{onf} = \frac{\left|\left\{\{A_i^j, A_k^l\} \middle| \sigma(A_i^j, A_k^l) > H_u^d\right\}\right|}{\left|\{A_i^j, A_k^l\}\right|} \qquad \ldots (6.1)$$

6.3 Heuristics for discovering *STB*

Traffic congestion related patterns discovered in the Chapter 5 are classified by object level pattern and area level pattern according to the granularity. In this section, three ***STB*** mining heuristics are proposed in order to discover the ***STB*** by analyze the area level patterns, which are congestion propagation heuristic (***CPH***), congestion converge heuristic (***CCH***), congestion drop heuristic (***CDH***). The first two are derived from ***CPP*** which is then transformed into congestion area sequence rule (***CASR***), and the last one is derived from ***CDP***.

6.3.1 Congestion-Propagation Heuristic (*CPH*)

The idea of ***CPH*** for discovering urban network ***STB*** is that traffic demand of a ***STCA*** may propagate to more ***STCA***s in neighborhood area. As illustrated in Figure 6-1, if an ***STCA*** exists in the condition part of many CASR rules then it is heuristically regarded as a root cause ***STCA*** (***RC-STCA***); therefore there may exist some bottlenecks in it.

The ***CPH*** category of bottleneck can be found by aggregating the condition part of all the discovered ***CASR***s to find the ***RC-STCA***. All the network objects in the ***RC-STCA*** are treated as an ***STB*** candidate. By detailedly inspecting the traffic attributes including includes ***θ***, traffic demand direction, and ***CDR*** of each candidate object, ***STB*** can be determined by criteria thresholds.

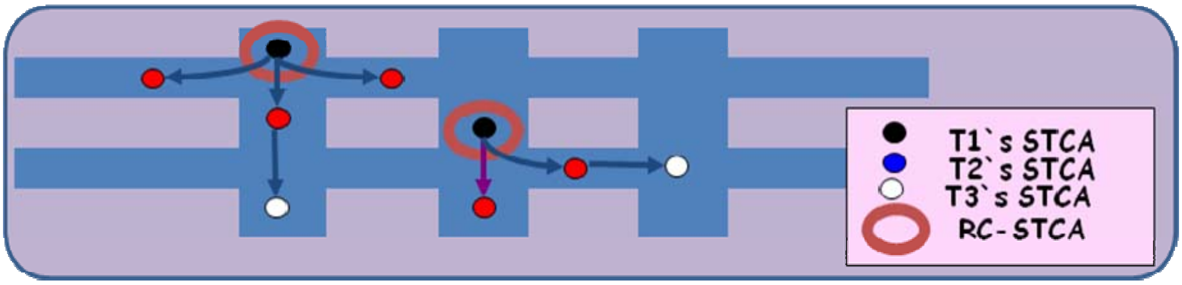


Figure 6-1 Congestion propagation heuristic for discovering *STB*

6.3.2 Congestion-Converge Heuristic (*CCH*)

On the other hand, an ***STCA*** might be caused by some other prior ***STCA***s, which constitutes the concept of congestion converge heuristic (***CCH***). As illustrated in Fig 6-2, if the traffic demands from several ***STCA***s go to the same destination ***STCA***, there may exist some ***STB*** in the destination ***STCA***. That is to say, if an ***STCA*** exists in the action part of many ***CASR*** rules then there exists some ***STB*** in it. By aggregating the action part of all ***CASR*** rule, ***CCH*** type of ***RC-STCA*** can be found. Again, each object in the root cause ***STCA*** is treated as an ***STB*** candidate and the ***STB*** can be determined the same criteria as in ***CPH***.

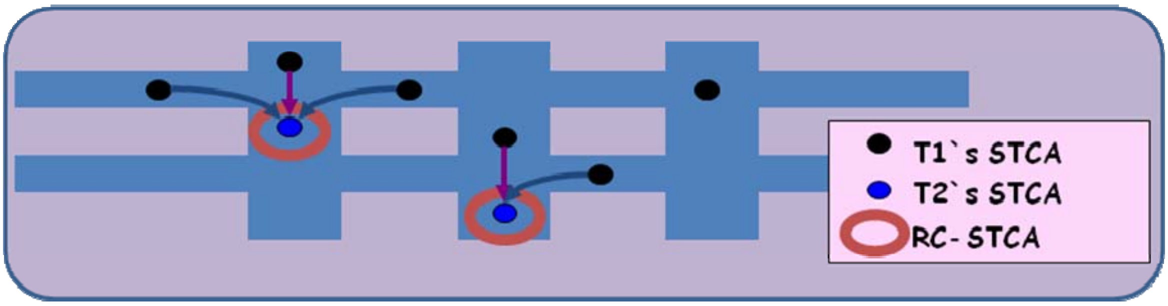


Figure 6-2 Congestion converge heuristic for discovering *STB*

6.3.3 Congestion-Drop Heuristic (***CDH***)

The idea of ***CDH*** is that if the congested status of an object decreases dramatically or even disappears after the traffic flow pass through the object, the object is treated as a bottleneck. This kind of bottleneck resulted from the capacity of the bottleneck object is not enough for the traffic demand, but afterwards, downstream objects do not congested because the capacity of these objects are enough for the traffic demand. The objects in the discovered ***CDP*** discussed in Section 5.1.2 can be directly transformed into ***CDH***.

6.4 Experiment

The ***STB*** discovering prototype was implemented based on a real time LBS-based application: *taxi dispatch system* (***TDS***) [LW+04], which is an online 7×24 system. The target area of this TDS system focused on the urban network in Taipei city, as shown in Figure 6-3, where an arterial in the network consists of one or several links as well as intersections. A link object consists of several attributes including category, length, direction, speed limit, and intersection object has the attributes of ***TD***/***LTD***/***RTD*** values of each direction with default values which are given by domain expert to facilitate ***STB*** discovering. The taxi fleet in the ***TDS*** has the size of 500 taxis, and the OBU reports its current status periodically at the interval of 30 seconds or when some events occur. Raw data was collected from the TDS during Feb. 2006 to Mar. 2007, where the average size is half a million uplink reports per day and total 13,000,000 records are collected. Traffic journeys are transformed from the raw data by combining the GIS road network. For example, ‘*dispatched*’ state journey consists of a set of traffic information spot which starts from the dispatched location to the customer’s location, and ‘*occupied*’ state journey starts from the customer’s location to their destination.

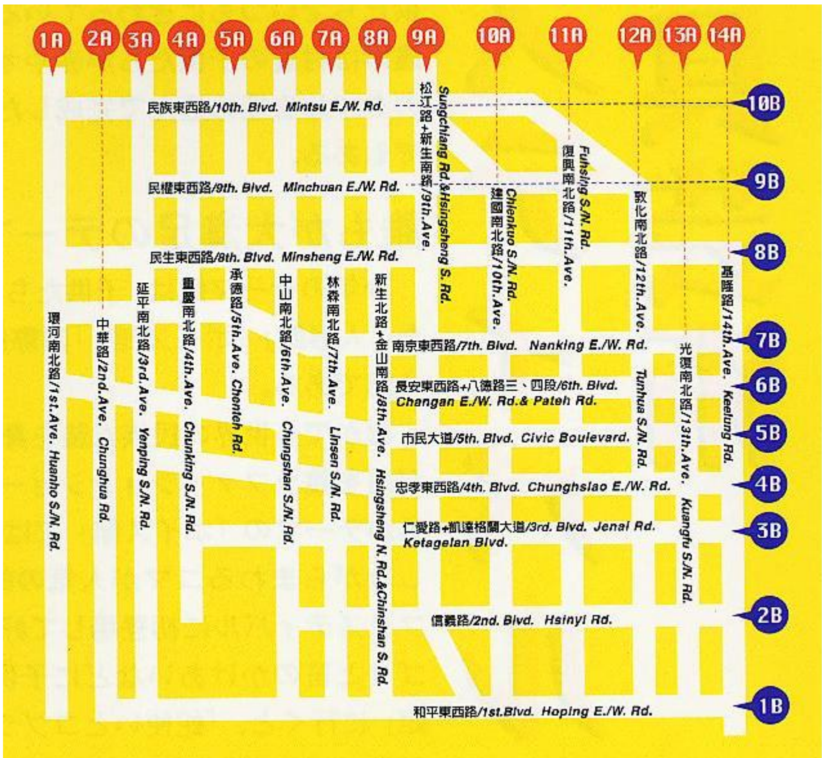


Figure 6-3 Road Network in Taipei Urban Area

In the traffic information generation phase, θ is obtained by aggregating the ***TIS***s at temporal and spatial dimensions and normalized by category and speed limit attributes. For example, the θ of an object close to **0** means the link is in free flow state and the traveling speed is near around the speed limit; on the other hand, if ***θ*** near around **1** indicates that the link is in extremely congestion status. The average ***θ*** of the network objects for workday and weekend are summarized as depicted in Figure 6-4, where each plot in X-axis indicates a **15** minutes time slot. There are two peaks in the curve of workday, which verify the common experience of ***AM*** and ***PM*** peak hours. On the other hand, the curve of weekend does not have the obvious peak due to different patterns on workday and weekend. In order to reduce the computing complexity, the search period for the ***STB*** is limited to the two peak hours of the workday, where the average ***θ*** of the ***AM*** peak hour (07:30~09:30) is 0.45, and ***PM*** peak hour (17:30~19:30) is 0.54.

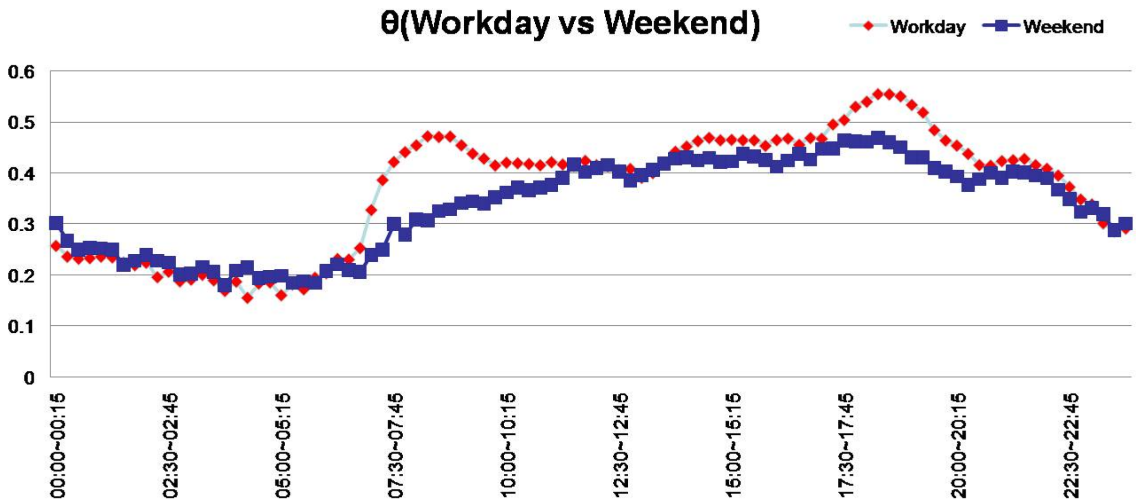
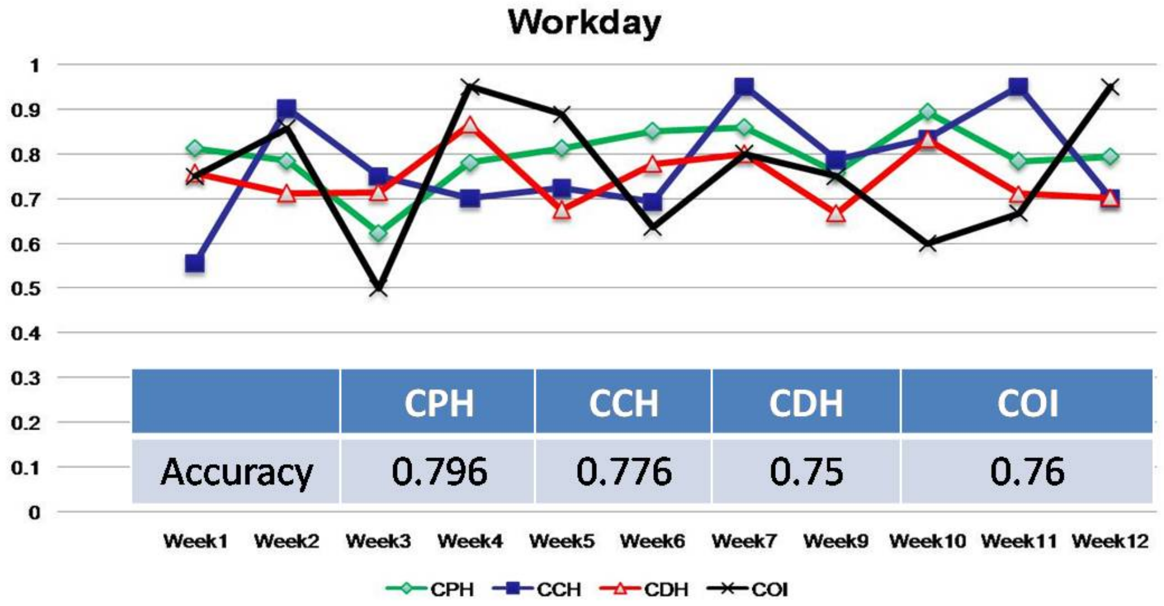


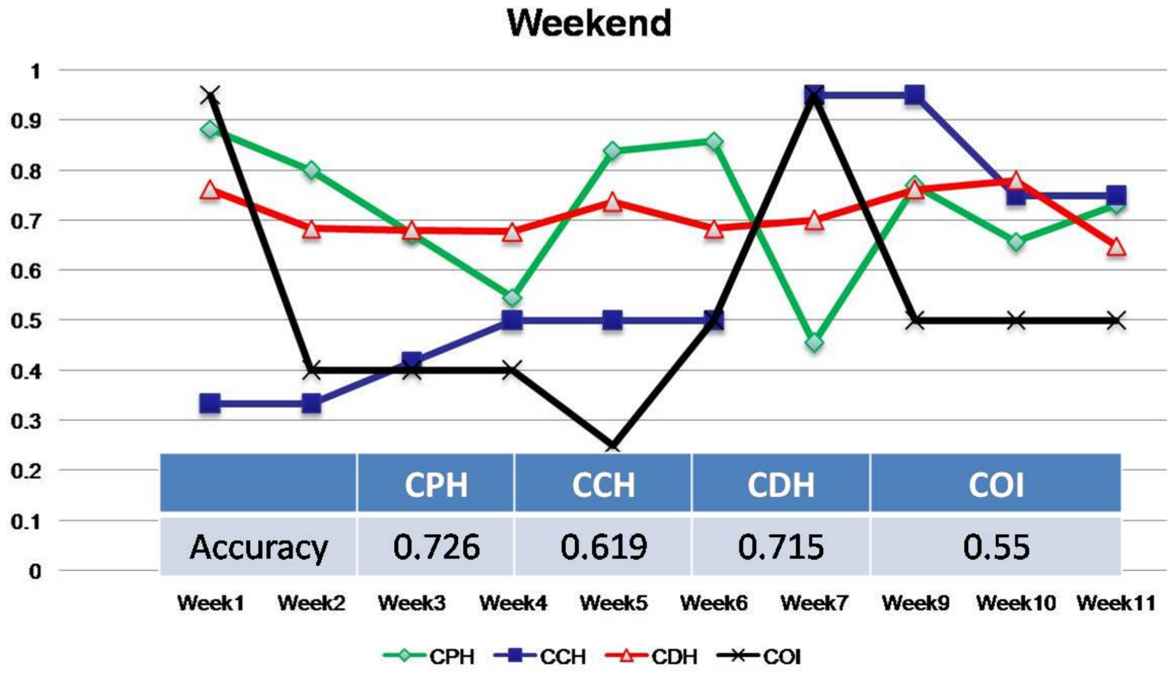
Figure 6-4 Traffic index factor (θ) for workday and weekend

The raw data during the Mar. 2006 to Dec. 2006 is used for STB discovering, and the remaining raw data from Jan. to Mar. in 2007, which includes twelve weeks, is used for comparing. ***STB*** transformed from the ***COI*** (congested object item) is compared with the ***STB*** discovered by three heuristics discussed in this experiment, where the results in workday and weekend are shown in Figure 6-5 and Figure 6-6 respectively. The former shows the congestion prediction comparison of the three heuristics method (***CPH***/***CCH***/***CDH***) with ***COI*** method in workday, and the latter shows the same comparison for weekend. The result shows that the average accuracy of ***CPH*** and ***CCH*** methods are more stable than other two methods (***CDH***, ***COI***) in workday, and also has higher accuracy. On the other hand, the congestion prediction accuracy of these methods which apply in weekend is much unstable than in workday, as shown in Figure 6-6. The prediction accuracy is worse than that apply in workday, which may be due to the traffic demand in weekend is diverse and does not have the general traffic pattern as the workday traffic demand. There are two traffic demand patterns in workday: (1) on-duty pattern which is the in-town demand in the morning and (2) off-duty pattern which is the out-town demand in the evening.



	CPH	CCH	CDH	COI
Accuracy	0.796	0.776	0.75	0.76

Figure 6-5 Three heuristic methods compared to COI method in workday



	CPH	CCH	CDH	COI
Accuracy	0.726	0.619	0.715	0.55

Figure 6-6 Three heuristic methods compared to COI method in weekend

In the experiment, we found that ***CPH*** method has the advantage in precision and stability, and the STBs discovered by ***CPH*** for the ***AM*** and ***PM*** peak hour is shown in Figure 6-7. In the Taipei Urban network the south-east side of the urban is the center and there are three types of arrow mined by ***CPH*** located on Taipei urban network: arrows in '*pink*' indicate the ***AM*** peak ***STB*** on 7:30 to 9:30 in the morning; arrows in '*blue*' indicate the ***PM*** peak ***STB*** on 17:30 to 19:30 in the evening; and arrows in '*yellow*' indicate the bottlenecks both in the morning and evening. The result shows the ***AM*** peak bottlenecks are coming from the suburban into the urban center which reveals the traffic demand of going on duty; furthermore the ***PM*** peak bottlenecks is just from the urban center to the suburban areas which shows the traffic demands of returning home.

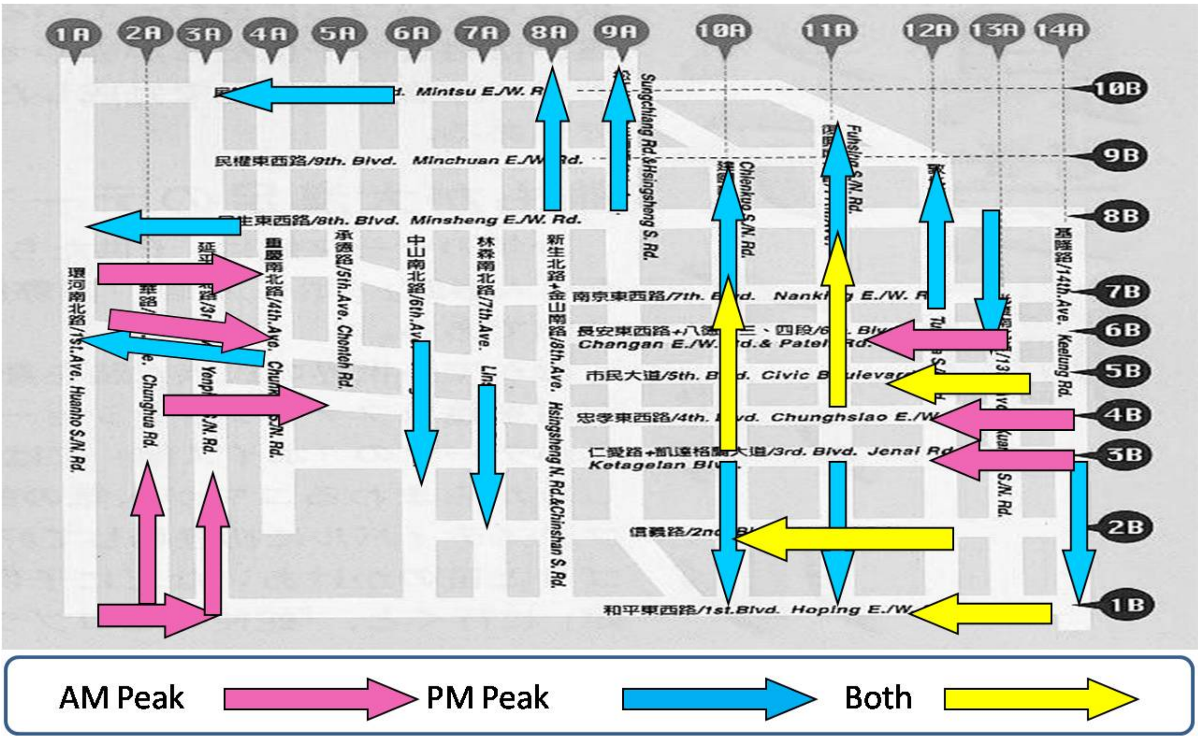


Figure 6-7 STBs discovered by CPH in workday

Chapter 7 Application: Real-time Travel Time Prediction

Many approaches had been proposed for travel time prediction (***TTP***) in these decades; most of them focus on the predicting the travel time on freeway or simple arterial network. However, most of the previous works only consider the static models of spatial network and predicted travel time based on the historical collected data, and thus lack the consideration of real-time events and traffic status. In other words, none of real-time events (e.g., detours, traffic congestions, etc.) affecting the spatial network can be reflected in the prediction result. Travel time prediction for urban network in real time is hard to achieve due to the following reasons: 1) network complexity, 2) path routing and selection problem in road network, 3) the collection of sensor data in real time is not available or cost-effective, 4) spatiotemporal data coverage problem of sensor or vehicle based travel time prediction, and 5) low precision due to lack of event response mechanism.

In this chapter, we propose a knowledge based real-time ***TTP*** model which contains real-time and historical travel time predictors, and a dynamic linear combination mechanism for the combination of these two predictors. Based upon the discovered spatiotemporal traffic patterns and the real-time traffic information database discussed in the previous chapters, the ***TTP*** model takes the advantage of independent knowledge base so that ***TTP*** knowledge can be dynamically adjusted to fit the changing requirement and response to external events. It results in that prior knowledge contributed by domain expert (meta-rules) and pattern knowledge mining from LBS-based applications can be evolved with the environment.

7.1 Architecture of ***TTP*** system

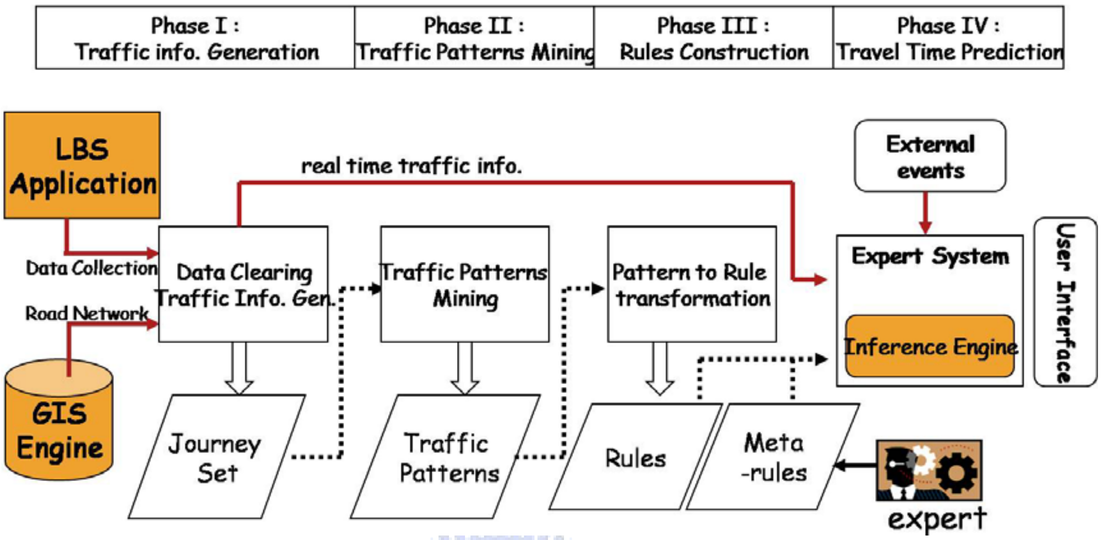


Figure 7-1 Architecture of *TTP* system

The system architecture of proposed ***TTP*** system is shown in Figure 7-1, which has two data sources: LBS-based applications and GIS engine. There are four information and knowledge processing phases in the ***TTP*** system, including:

- Phase I: Traffic information generation
- Phase II: Traffic patterns mining
- Phase III: Rules construction
- Phase IV: Travel time prediction

In Phase I, the traffic information is collected from the LBS-based applications, and transformed to traffic information by combining the road network information retrieved from GIS engine. The collected traffic information is stored in the ***TIDB***, such as ***TIS***, vehicle journey, and ***TNS***, as discussed in Chapter 3. In Phase II, several spatiotemporal patterns such as ***SAP*/*CDP*/*IDP***, etc., are mined from the ***TIDB*** which is discussed in Chapter 5. Rules construction Phase (Phase III) constructs two types of ***TTP*** rules: meta-rules donated by the domain experts, and rules transformed from the traffic patterns discovered in Phase II. In Phase IV, knowledge base technique is applied to ***TTP*** inference, where external events are fed into the inference engine as facts to trigger the dynamic weight adjustment mechanism.

7.2 Dynamic weighted combination ***TTP*** scheme

The basic idea of the proposed ***TTP*** model is that travel time along a selected path can be estimated by summing up the links travel time with intersections delay, as shown in Equation (7.1), where link travel time can be estimated by linear combination of current (real time) and historical predictors. As the example illustrated in Figure 7-2, where upper part shows the link travel time of the real-time predictor and the lower part shows the link travel time of the historical predictor. Linear combination of the real-time predictor and historical predictor with dynamic weight adjustment mechanism is adopted for the final prediction result.

Origin ($\boldsymbol{O}$), destination ($\boldsymbol{D}$) and journey start time ($\boldsymbol{t}$) are the input parameters of the prediction formula $T(\boldsymbol{O,D,t})$, as shown in Equation (7.1). Two sub-functions $\boldsymbol{T_c}$ and $\boldsymbol{T_h}$ are two link travel time predictors based on real-time and historical traffic information respectively, and $\boldsymbol{T_d}$ presents the total intersection delays of the passing through intersections in the path. Two control variables $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are the weighted combination variables for current ($\boldsymbol{T_c}$) and historical ($\boldsymbol{T_h}$) predictors, respectively. The weights for these two predictors are dynamically decided by meta-rules donated by domain experts, which are discussed in Section 7.3.4.

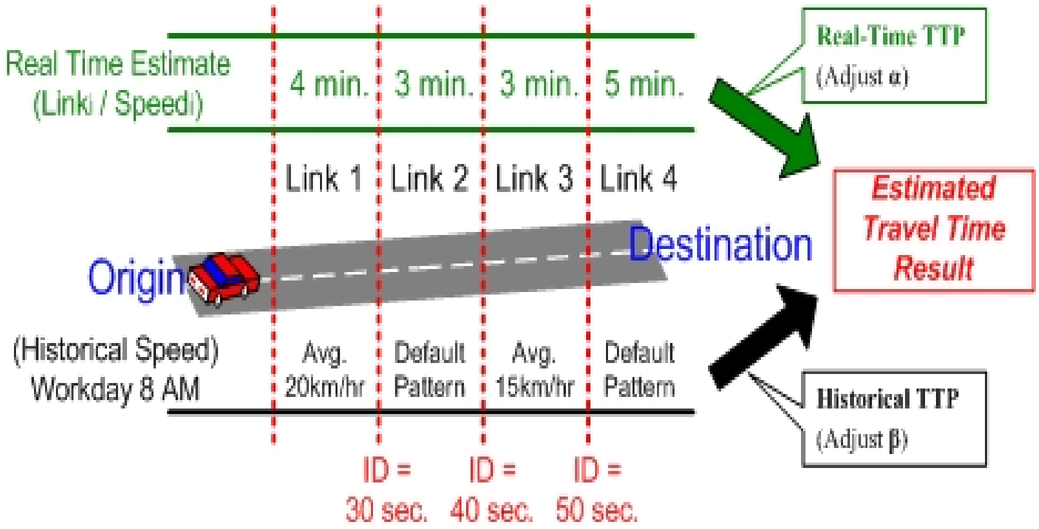


Figure 7-2 Linear combination of real-time and historical travel time predictors

$$T(O,D,t) = \alpha \cdot T_c(O,D) + \beta \cdot T_h(O,D,t) + T_d(O,D,t) \qquad \ldots (7.1)$$
$$where \quad \alpha + \beta = 1$$

The proposed real-time knowledge based ***TTP*** model predicts travel time based on the knowledge based system and data mining technologies. There are two categories of knowledge in this model: (a) general rules for real-time and historical ***TTP*** are obtained by mining the LBS-based applications, (b) meta-rules donated by the human domain experts. Meta rules play a key position in the ***TTP*** model for following four purposes:

1) dynamic weight decision depending on the external events impact factors fusion for the linear combinations of two predictors

2) general rules modification and prediction adjustment for the higher precision

3) traffic rules application, such as left/right turn forbidden, speed limit, etc.

4) default rule actions for missing values.

As mentioned above, travel time along a selected path can be estimated by summing up the links travel time and intersection delays, as shown in Equation (7.1). In the road network of an urban area, an (***O, D***) pair may have many path choices, and each path may consist of several road links. As we know, there are many strategies for choosing the candidate paths, such as shortest path first, expressway first strategies, etc. In this dissertation, we adopt the top *k* used paths selection heuristic, i.e., each candidate path will be evaluated by our model to estimate the travel time, and the top ***k*** candidate paths from the historical database with the ***1***-st to ***k***-th lowest evaluated travel time will be suggested. Once the candidate paths have been decided, travel time along each candidate path can be predicted by summing up the travel time of the links in the path and the intersection delays between consecutive links. Assume P_j(***O,D***) is a link set in the candidate path ***j***, each $\boldsymbol{L_i}$ is the ***i***-th link from the origin in P_j(***O,D***). The estimated travel time for path *j* can be formulated as Equation (7.2), where $\boldsymbol{T_c}$ and $\boldsymbol{T_h}$ are the two predictors for real-time and historical prediction respectively; $\boldsymbol{D_{Li,Li+1}}$ stands for the intersection delay between two adjacent link $\boldsymbol{L_i}$ and $\boldsymbol{L_{i+1}}$, and $\boldsymbol{\alpha_i}$, $\boldsymbol{\beta_i}$ are the weight control variables of the link L_i. As the travel time of each candidate path has been estimated, the travel time of a given ***(O, D)*** pair can be estimated by choosing the minimum estimated travel time from the ***1***-st candidate to the ***k***-th candidate path, as depicted in Equation (7.3), and thus become the suggested path.

$$
\begin{aligned}
&T_j(O,D,t) = \sum_{\forall L_i \in P_j(O,D)} (\alpha_i \cdot T_c(L_i) + \beta_i \cdot T_h(L_i,t)) + \sum_{\forall L_i \in P_j(O,D)} D_{L_i,L_{i+1}} \\
&where \qquad \alpha_i + \beta_i = 1, \qquad \forall i
\end{aligned}
\qquad \dots (7.2)
$$

$$
T(O,D,t) = \underset{j \in \{1..k\}}{Min} \{T_j(O,D,t)\} \qquad \dots (7.3)
$$

7.3 Ontology for travel time prediction

Ontology building mainly depends on the contribution of domain experts in the knowledge creation activity. Metadata extraction and merging is carried out manually by domain experts. As shown in Figure 7-3, the ***TTP*** ontology designed by cooperation of the traffic domain experts, knowledge acquisition engineers, and system design engineers to organize the ***TTP*** inference structure consists of several design concepts and the relationships among these concepts, where one type of concepts is meta-concept drawn by red circle encrypting the domain knowledge donated by the traffic domain expert, or process control knowledge designed by system engineer, and the other is normal concept drawn in black circle representing the static knowledge or mining knowledge generated in the rules construction phase.

There are several relationships between these concepts, such as “***is-a***”, “***includes***”, “***modify***”, “***related-to***”. These relationships connecting concepts in the ontology represent some interactions among them. For example, “***real time TTP predictor***” concept includes a “***Path selection***” concept, and “***Path selection***” concept includes “***Path to link decomposition***” concept. When the ***TTP*** predictor concept is started, path selection process is then fired to find out the candidate paths between origin and destination, and the path selection process then fires the path to link decomposition process to get the links and intersections to be used to evaluate the travel time in each candidate path. In the following subsections, some major concepts in the ***TTP*** are detailedly discussed to outline this ***TTP*** ontology workflow.

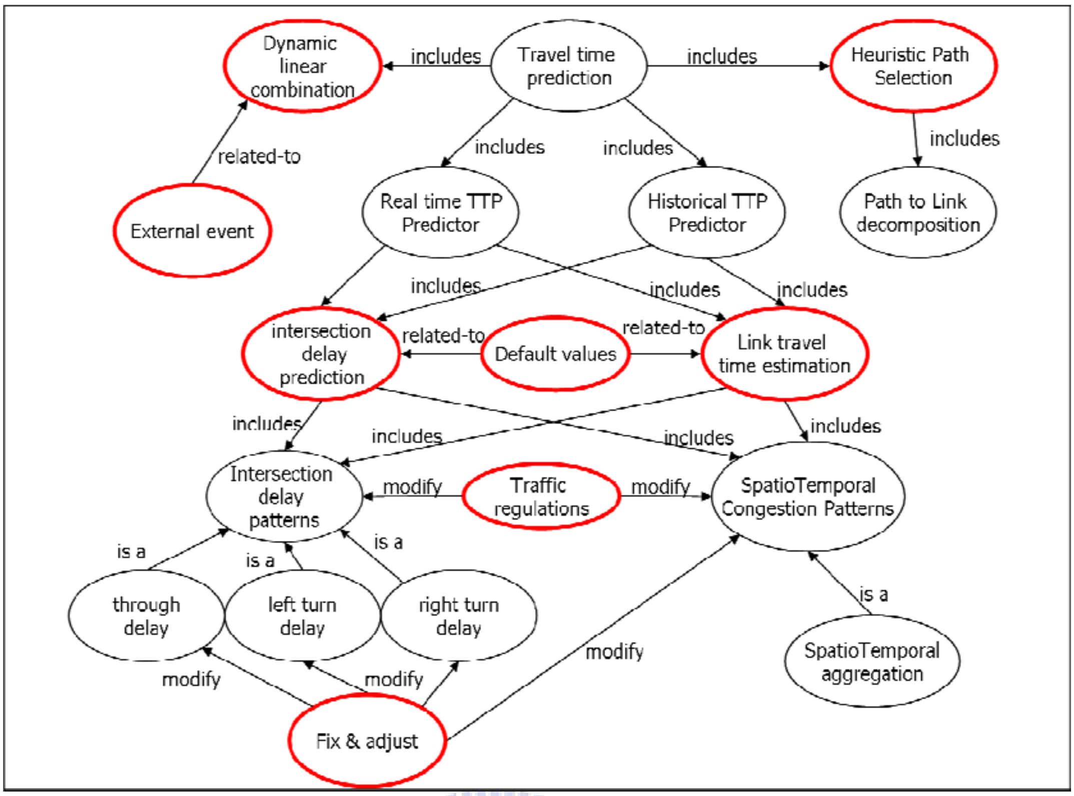


Figure 7-3 Design ontology of the *TTP* system

7.2.1 ***TTP*** system workflow

The ***TTP*** system workflow which depicts the relationship of several system modules for the ***TTP*** is shown in Figure 7-4. "***Heuristic path selection***" module analyzes the (***O***, ***D***, ***t***) input data and generates the heuristic candidate paths from the origin to the destination with the assistance of the GIS engine and ***TIDB***. The "***path to link decomposition***" module decomposes each candidate path to a set of continuous network objects from the origin to the destination, where a network object may be link or an intersection. The link travel time is then estimated by the "***real-time predictor***" and "***historical predictor***", and the intersection objects travel time is estimated by the "***intersection delay***" module. After all the objects in the candidate paths are estimated, the travel time of a candidate path can be estimated by summing up all the network object travel time in the candidate path, including links travel time and intersections delay. The "***meta-rule weighted combination***" module integrate the dynamic weight decision mechanism donated by the traffic domain experts, which decides the weight of real-time as well as historical predictors by considering the effect of the external events and spatiotemporal traffic patterns. Finally the "***path selection***" module select the shortest travel time path as the system suggested path and shows the estimated travel time of the suggested path.

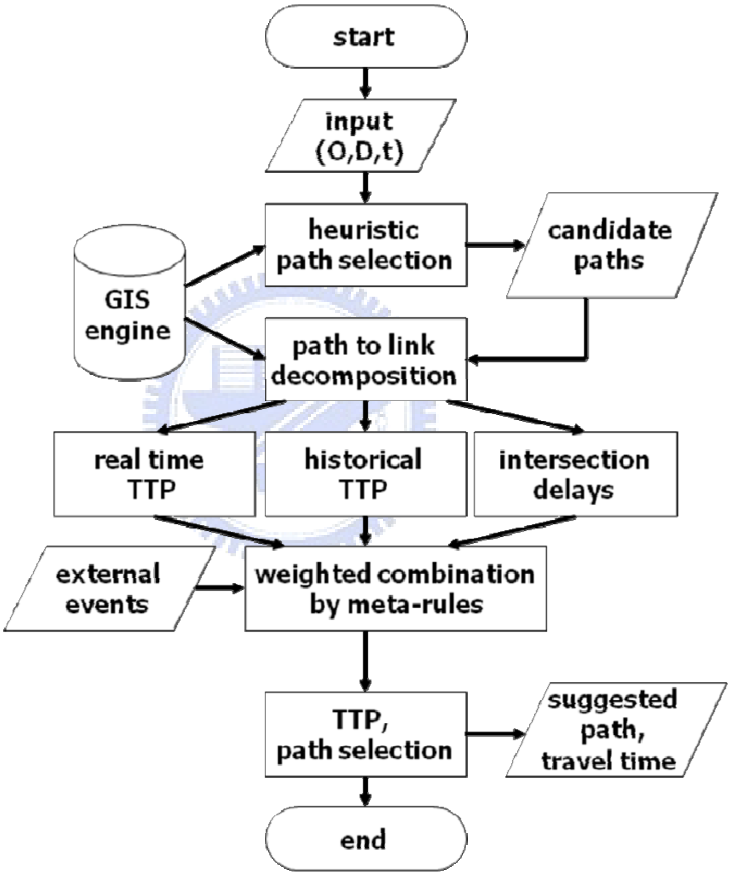


Figure 7-4 travel time prediction inference workflow

7.2.2 Heuristic Path Selection

Path selection module is activated by the travel time predictor modules by the “***includes***” relationship in the ***TTP*** ontology. Path selection problem in the urban road network is much more complicated than that in freeway since only few paths can be chosen in freeway path routing for a given ***(O, D, t)*** input. To minimize the path travel time, many strategies had been proposed for a routing path selection based on a given ***(O, D, t)*** input in the urban network, such as shortest path first, expressway first or signal less path first, etc. The common goal in these strategies is to minimize the path traveling time. To cope with the path selection problem, heuristic and domain knowledge are used in this dissertation. Since most of the taxi drivers can most likely select the heuristic optimal path according to their experience and the current traffic status. Our idea is to select top ***k*** paths from the journey set in the historical database as the candidate paths according to the request ***(O, D, t)***.

1896

7.2.3 Historical Predictor

Historical predictor and real-time predictor inherited from the ***TTP*** concept include two parts: link travel time estimation and intersections delay prediction. After historical predictor inferring the travel time of each link and intersection delays in the candidate path, the total amount of the links travel time and intersection delays is the predict result. Link travel time estimation is inferred by rules obtained from the spatiotemporal congestion patterns. Intersection delay prediction is reasoned by the rules obtained from the intersection delay patterns, which are classified by through delay (TD), right turn delay (RTD) and left turn delay (LTD). The rules transformed from these patterns are stored into the knowledge base (KB). When a ***(O, D, t)*** request is fed into the expert system as facts, the inference engine automatically fires the rules related to the historical ***TTP*** prediction. Meta concept components in this ontology help to fill out the missing values and fixed some outliers. For example, if one rule shows that one link can be traveled at a speed in peak hour which is much higher than the speed limit, the meta-rule in "***fix & adjust***" component might fix and replace it with the speed limit of that link.

7.2.4 Real-Time Predictor

Real time ***TTP*** of a candidate path can be done by summing up the travel time of each links and intersections delay which constitute the path, where the current travel time of a link can be easily done by dividing the link length with current average traveling speed of that link, and the real time intersection delay is obtained from the real time traffic information generated in Phase I. In the case of missing current speed of some links, a speed evaluation meta-rule given by domain expert in the "***default values***" concept is fired to give a default speed depending on their heuristic experience and spatiotemporal conditions. For example, domain expert may give default speed of midnight on non-holiday as 20% more than the speed limit of the link. The process of determining the missing value in intersection delay can be done in the similar way.

7.2.5 Dynamic Linear Combination

"***Dynamic linear combination***" concept incorporates with "***external event***" concept to provide the real time event response ability for the ***TTP*** model, which raises the precision of ***TTP*** and makes it more practicable. Real time event response mechanism consists of several meta-rules, which is designed to handle the external events by dynamic tuning the weight of two predictors: T_c, T_h through weight control variables: α and β. For example, if the system receives a current external event, such as car accident on a link in the candidate path, event handling meta-rules will then reduce the weight of historical predictor (***T_h***) and raise the weight of real time predictor (***T_c***). Because the effect of that car accident will be reflected at the corresponding link immediately, raising the weight of real time predictor might get higher precision. On the other hand, some meta-rules may raise the weight of historical predictor if the following two conditions are satisfied: if there is no current event, and the support and confidence of the related patterns are higher than the threshold set by the expert. It means that there is a strong support that traffic status most likely regresses to the intents of related historical patterns. Therefore, raise the weight of historical predictor might get higher precision.

7.4 Knowledge based ***TTP***

7.3.1 Rules constructed from traffic patterns

Two kinds of rules are constructed in this phase: rules transformed from the traffic patterns, and meta-rules donated by domain experts. The traffic patterns, mining from the historical journey set in previous phase, will then be transformed to the format of ***if-then*** rules by mapping the condition and action parts of the rule from attributes of the patterns, and decide the support and confidence of each generated rule by computing the probabilities of the rule. The if-then format rules are stored in the knowledge base, and can be easily read by inference engine and then make ***TTP*** inference. For example, ***SAP*** of Equation (5.2) can be transformed to if-then rules (***SAP*** rules) by combining the link attribute in the traffic network database, as illustrated in Equation (7.4). The ***S_{id}, T_{id}, E_{id}*** information are transformed to the condition part of the if-then rule, and the congestion level (***C_g***) can be transformed to estimated speed in the action part by considering the link attributes (speed limit, category). And thus the estimated travel time can be calculated by dividing the length of the link with the estimated speed. Equation (7.5) shows the link travel time (***LTT***) estimation rule, which cooperates with ***STA*** rule and link attributes to compute the link travel time.

[***SAP*** rule] … (7.4)

IF

(S_{id} and T_{id} and E_{id})

THEN

Congestion level at the Link(S_{id}) = C_g,

Support = Sup,

Confidence = Con

[***LTT*** rule] … (7.5)

IF

(C_g = L) and Cate(S_{id})

THEN

Speed at Link(S_{id}) = S,

LTT at Link(S_{id}) = Length(S_{id}) / S

7.3.2 Historical travel-time predictor

Historical predictor and real-time predictor inherited from the ***TTP*** concept include two parts: link travel time estimation and intersections delay prediction. After historical predictor inferring the travel time of each link and intersection delays in the candidate path, the total amount of the links travel time and intersection delays is the predict result. Link travel time estimation is inferred by rules obtained from the spatiotemporal congestion patterns. Intersection delay prediction is reasoned by the rules obtained from the intersection delay patterns, which are classified by through delay (TD), right turn delay (RTD) and left turn delay (LTD). The rules transformed from these patterns are stored into the knowledge base (KB). When ***(O, D, t)*** request is fed into the expert system as facts, the inference engine automatically fires the rules related to the historical ***TTP*** prediction. Meta concept components in this ontology help to fill out the missing values and fixed some outliers. For example, if one rule shows that one link can be traveled at a speed in peak hour which is much higher than the speed limit, the meta-rule in "***fix & adjust***" component might fix and replace it with the speed limit of that link.

7.3.3 Real-time travel-time predictor

Real time ***TTP*** of a candidate path can be done by summing up the travel time of each links and intersections delay which constitute the path, where the current travel time of a link can be easily done by dividing the link length with current average traveling speed of that link, and the real time intersection delay is obtained from the real time traffic information generated in Phase I. In the case of missing current speed of some links, a speed evaluation meta-rule given by domain expert in the "***default values***" concept is fired to give a default speed depending on their heuristic experience and spatiotemporal conditions. For example, domain expert may give default speed of midnight on non-holiday as 20% more than the speed limit of the link. The process of determine the missing value in intersection delay is done in the similar way.

7.3.4 Dynamic weighted combination scheme

"***Dynamic linear combination***" concept incorporates with "***external event***" concept to provide the real time event response ability for the ***TTP*** model, which raises the precision of ***TTP*** and makes it more practicable. Events are collected from external data sources and saved in ***TIDB*** (traffic information database) as discussed in the traffic information generation and sharing framework as discussed in Chapter 4. Real time event response mechanism consists of several meta-rules, which is designed to handle the external events by dynamic tuning the weight of two predictors: T_c, T_h through weight control variables: α and β. Table 7-1 illustrates some examples of the weight assignment meta-rules donated by domain experts.

For the "*event consideration*" as an example, if the system receives a current external event, such as car accident on a link in the candidate path, event handling meta-rules will then reduce the weight of historical predictor (T_h) and raise the weight of real time predictor (T_c). Because the effect of that car accident will be reflected at the corresponding link immediately, raise the weight of real time predictor might get higher precision. On the other hand, "historical support is strong" meta-rule raise the weight of historical predictor if the following two conditions are satisfied: if there is no current event, and the support and confidence of the related patterns are higher than the threshold set by the expert. It means that there is a strong support that traffic status most likely regresses to the intents of related historical patterns. Therefore, raising the weight of historical predictor might get higher precision.

Table 7-1 Examples of the dynamic weight assignment meta-rules

Weight Assignment Meta-Rule	**Condition**	**Action**
Lack of real-time info.	Prediction time is much earlier than the departure time.	α =0, β=1
Temporal distance to real-time info. is short.	Real-time info. in TIDB is close to the link entry time.	Raise the weight of α
Temporal distance to real-time info. is long.	Real-time info. in TIDB is far from the link entry time.	Raise the weight of α
Event consideration	A real-time event is happened at some link, and some probing car just passes by the link.	Raise the weight of α
Historical support is strong	The support and confidence of historical patterns are strong	Raise the weight of β

7.5 ***TTP*** prototype system

The ***TTP*** prototype system was implemented based on a real time LBS-based application: taxi dispatch system (TDS) [LW+04]. The TDS is an online 7*24 system operated in Taipei urban area (as shown in Figure 7-2), and the current fleet size is about 500 taxis, where the OBU reports its current status based on following events: periodically (30 sec), spatial trigger event, or some business events, such as dispatch/response event, customer on/off taxi events, etc. The raw data in TDS system had half a million records per day, which was a good data source for this prototype ***TTP*** system. At the data collecting and clearing phase, the OBU report raw data has been collected and translated to ***TIS***s in a period of 5 minutes in order to catch the real time traffic information, and all the ***TIS***s are filtered out except the OBU in ‘***dispatched***’ or ‘***occupied***’ state since the traffic information extracted from these two states is meaningful. Besides, the OBU report will be filtered out if the location of its ***TIS*** is not in the interested links (links in Figure 7-5). Historical traffic information consists of journey sets by combining the raw data and the GIS road network. A journey is a tour consisting of a set of sequence ***TIS***s between origin and destination. The ‘dispatch’ state journey starts from the dispatch location to the customer’s location, and ‘occupied’ state journey starts from the customer’s location to customer’s destination.

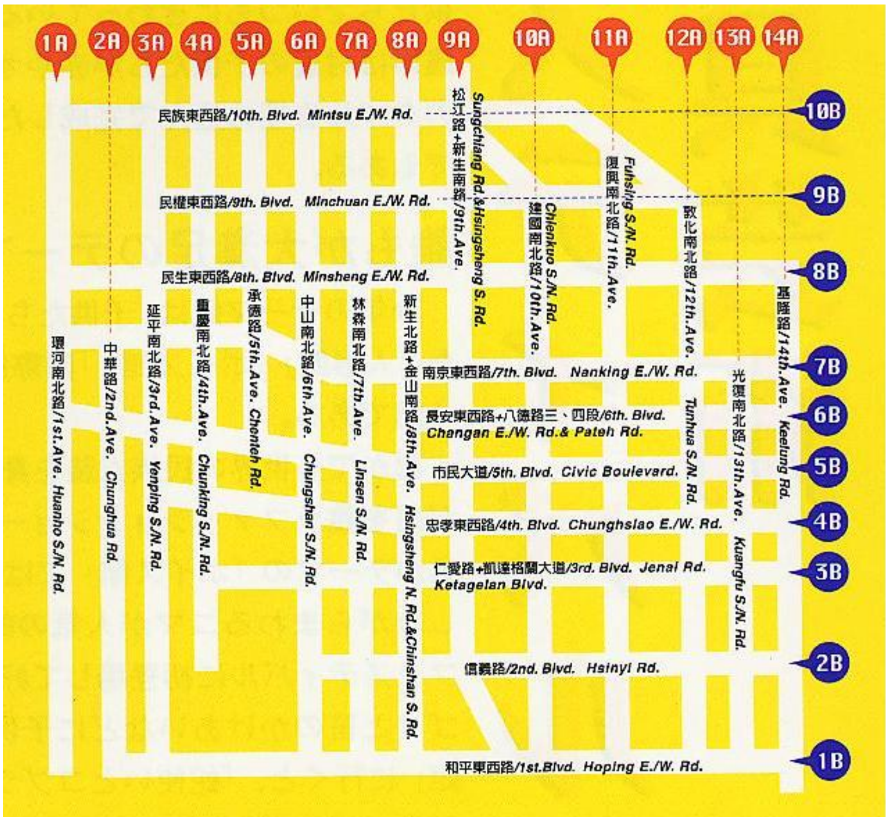


Figure 7-5 Road Network in Taipei Urban Area

As shown in Figure 7-5, the target area of this prototype system is focused on the arterials in Taipei urban area, and each arterial may have one or several links. Link attributes including link category, length, direction, speed limit, average signal delays and geographical coordinate vectors are defined and default values are given by domain experts to facilitate link travel time estimation. At the traffic patterns mining phase (phase II), traffic level (**1**-**9**) is classified by aggregating the ***TIS***s at temporal and spatial dimension and normalized by two link attributes: category, speed limit. For example, traffic level **9** means the link is in free flow state and the traveling speed is near around the speed limit; on the other hand, traffic level **1** represents that the link is in extremely congestion status.

The ***TTP*** prototype system was implemented by the expert system shell, DRAMA [CO03], a NORM (New Object-oriented Rule Model) [LTT03] knowledge modeled rule base system platform implemented using pure Java language, includes Drama Server, Console, Knowledge Extractor, and Rule Editor. ***TTP*** rules generated from the Phase III and meta-rules donated by traffic experts are categorized to several knowledge classes (KC), and stored at knowledge base of DRAMA server. A real time traffic information database: e-IOT [EIOT] is connected as an external data source. E-IOT is a centralized real time traffic information database provided by institute of traffic of Taiwan government, which provides various traffic event information, including traffic block, traffic jam, construction, signal break, disaster, and accident. The external traffic information is used as a trigger input for weight combination knowledge classes (meta-rules), which dynamically tunes the weighted combination of current and historical travel time predictors. The essence of the weight tuning meta-rules can be induced in two principles: meta-rules raise the ratio ($\boldsymbol{\alpha}$) of current predictor when a current event is happening at some links, and return to the origin ratio when the event has been relieved. The second principle is that the weight of historical predictor ($\boldsymbol{\beta}$) will be raised when the support and confidence of related patterns are higher than the threshold set by domain experts.

7.6 Experiment

Five months (2006/02 ~ 2006/06) LBS raw data is collected for the experiment, the data of first four months is for mining the traffic patterns and the fifth is for testing the TTP. In the experiment, three predictors: current-time predictor, historical traffic pattern predictor and weighted combination predictor had been implemented and compared. Current travel time predictor makes the prediction by summing up the travel time and intersection delays based on current speed on the links of the candidate path and related intersection delays. Historical traffic pattern predictor predicts travel time by reasoning the historical traffic patterns. The weight combination predictor predicts travel time by weighted combination of these two predictors based on the external traffic events and meta-rules. Two performance indices: relative mean errors (**RME**) and root mean squared errors (**RMSE**) are applied for comparing these predictors and listed as Equations (7.6), (7.7) where n is the number of predictions, $\boldsymbol{X_i}$ and $\overline{X_i}$ present the travel time and prediction time respectively. The last of linear combination predictor use the meta-rules to dynamically adjust $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ variables with real-time events consideration, as mentioned before.

$$RME = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{X_i - \overline{X_i}}{X_i}\right| \qquad \text{… (7.6)}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left|\frac{X_i - \overline{X_i}}{X_i}\right|^2} \qquad \text{… (7.7)}$$

Table 7-2 *TTP* precision of three predictors (by IDPs)

PREDICTOR	HISTORICAL PREDICTOR	CURRENT TIME PREDICTOR	WEIGHT COMBINATION PREDICTOR
RME	*20.5%*	*13%*	*10.8%*
RMSE	*24.82%*	*18.13%*	*15.92%*

Table 7-3 *TTP* precision of three predictors (by expertise in ID)

PREDICTOR	HISTORICAL PREDICTOR	CURRENT TIME PREDICTOR	WEIGHT COMBINATION PREDICTOR
RME	*29.6%*	*21%*	*20%*
RMSE	*35.78%*	*32.21%*	*30.55%*

Table 7-2 shows the **RME** and **RMSE** results of different predictors on workday (Mon. to Fri.). In this experiment, we randomly choose different ***(O, D)*** pairs at two peak hour sections: 7:30~9:00AM and 6:00~7:30PM and then calculate the **RME** and **RMSE** with each predictor. The result shows that current time predictor has better precision than historical predictor. By combining domain expert knowledge in meta-rules, weight combination predictor has better performance than other two predictors in both **RME** and **RMSE**.

In the above experiment, the intersection delay is regarded as fixed variable so that all the three predictors use the same ***IDP***s for calculating intersection delays. In order to compare the ***IDP***s and human expertise, the experiment with the same data set has been made again by replacing ***IDP***s with human expertise. That is, human expert gives a default value for average turning delays instead of mining ***IDP***s from historical traffic database. The result in Table 7-3 shows that precision of using human expertise is not as good as ***IDP***s. This might be improved by tuning the turning delay default value by domain experts.

1896

Chapter 8 Application: Decision Support System for Advanced Traffic Management System (ATMS)

The discovered knowledge discussed in the previous Chapters, such as spatiotemporal traffic patterns, spatiotemporal traffic bottlenecks, predicted traffic status are valuable information and can be further utilized for improving the global traffic performance in urban network. In Chapter 7, the discovered knowledge are applied for the travel time prediction, which assists travelers for choosing the shortest travel time routing path or avoids the congestion area before or during the journey. In this Chapter, the knowledge is applied to a decision support system (DSS) for the network traffic assignments, which assists network administrators to make the right actions to enhance the global network performance. This DSS is categorized as advanced traffic management system (ATMS) in the intelligent transportation system (ITS), thus it is named as ATMS DSS.

Knowledge based system (KBS) technique is adopted for the ATMS DSS, where an ontology is designed as the kernel of the ATMS DSS based upon a three-layer system optimizing traffic assignment principles. Top level principles consider the global network performance enhancement as well as demand conflict resolution issues, mid level principles targets at local area traffic flow smoothing and entry level principles deal with network object optimization. The granularities and considerations of the hierarchical three-layer traffic assignment principles is illustrated in Table 8-1, where the possible actions show the strategies and approaches for the goals.

Table 8-1 Hierarchical three-layer traffic assignment principles

Traffic assignment principle	**Granularity**	**Goal / Consideration**	**Possible Actions**
Top Level	Global network Area to Area	Global performance enhancement, Area level congestion prevention	area level traffic assignment
Mid Level	Area	Area level traffic smoothing Demand conflict resolution	directed signal chaining reversible lane
Entry Level	Object	object level control for local optimization	Signal control manual control

8.1 System architecture

The system architecture of ATMS DSS is shown in Figure 8-1, which integrates several components including knowledge base, inference engine, traffic information database, urban network database, GIS, and a web-based user interface. It connects to the spatiotemporal traffic patterns mining system, spatiotemporal traffic bottlenecks discovering system, collaborative traffic information generation and sharing framework as discussed in the previous Chapters, and transforms the knowledge discovered by these systems into knowledge classes. Domain expertise combined with discovered knowledge constitutes the ATMS ontology, which is designed based upon the expertise donated by the traffic management domain expert for the purpose of global traffic network performance enhancement. It is the kernel of the ATMS DSS and directs the whole inference process to generate traffic assignment suggestions.

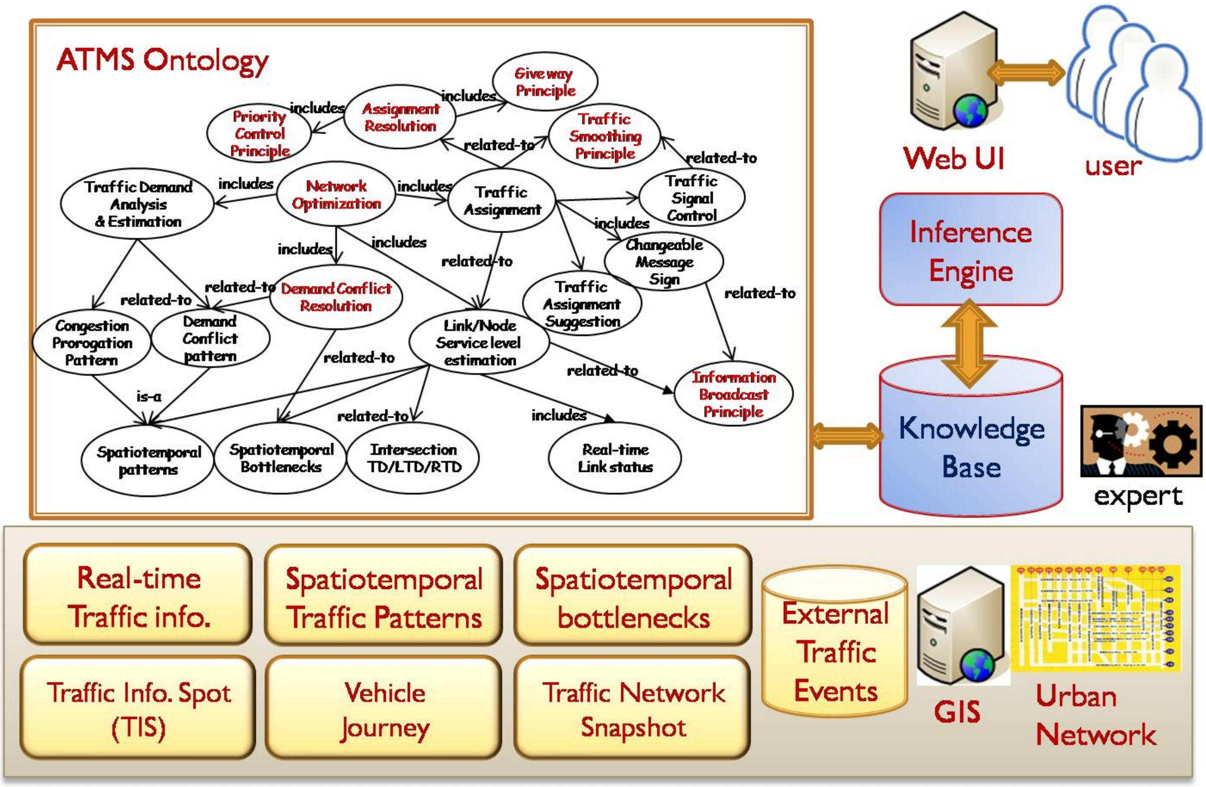


Figure 8-1 System architecture of ATMS DSS

The knowledge structure of the ATMS DSS is classified by five categories, as shown in Table 8-2, including traffic assignment principles, ontology, discovered traffic knowledge (e.g. ***STP***/***STB***), collected traffic information, and urban network topologies in GIS. In this table, the source and store of the knowledge are listed, and the inference types (rules or facts) existed in the inference engine are illustrated. For example, the ontology is designed based upon the heuristic expertise, and it is stored in the knowledge base as rules, and traffic information collected from the LBS is stored at ***TIDB*** and treated as facts in the inference engine.

Hierarchical three-layer traffic assignment principles guided the direction of the inference from the top level global view to the entry level local view, which are detailedly discussed in Section 8.2. The principles and the traffic assignment heuristics constitute the

ATMS ontology, which is discussed in Section 8.3.

Table 8-2 Hierarchical knowledge structure in ATMS DSS

Knowledge / Information	Source	Knowledge Hierarchy	Inference Engine	Store
Traffic assignment principles	Domain expertise	Human expertise	-	Ontology
Ontology	Heuristic expertise	Human expertise	Rules	Rule classes in knowledge base
STP/STB	Mining from TIDB	Machine learning rule	Rules	Rule classes in knowledge base
Traffic info.	LBS, Traffic Framework	Data	Facts	Database
Network database	GIS	Data	Facts	GIS

8.2 Hierarchical three-layer traffic assignment principles

To achieve the goal of global traffic network performance enhancement, an ATMS ontology is designed based upon hierarchical three-layer traffic assignment principles. The targets and possible traffic assignment actions of the three-layer principles are summarized in Table 8-2, where the top level principle considers global system optimal and area level congestion prevention issues, mid level considers the area level traffic smoothing, and the entry level deals with the object level control for local optimization.

For the discovered ***STP/STB*** discussed in the Chapter 5 and Chapter 6, the features of these knowledge can be further analyzed to find out the possible actions of enhance global traffic network performance. The solution concepts and possible traffic assignment actions of the discovered ***STP/STB*** are summarized in Table 8-3, where the traffic patterns are categorized by their granularities and classified to different levels by their granularities. For example, ***SCO*** (sole congested object) pattern does not related to the neighborhood objects so that it is an entry level pattern and the possible traffic assignment solutions for this pattern are signal control and CMS. ***CPP*** (congestion propagation pattern) concerns about the spatiotemporal association relationship between two areas which is an area level pattern and several possible actions can be considered, such as CMS, signal control chaining, or reversible lane.

Table 8-3 Spatiotemporal traffic patterns and handling strategies

Spatiotemporal Traffic Patterns	**Granularity / Relation**	**Level**	**Possible Actions**
Sole Congested Object	Object	Entry Level	Signal Control, CMS
Intersection Delay	Object	Entry Level	Conflict check, Signal control
Congestion Drop (STB)	Obj2Obj	Mid Level	Conflict check, Signal control
Congestion Propagation	Area2Area	Top Level	CMS, signal control chaining, reversible lane
Demand Conflict (CCP)	Area2Area	Top Level	CMS, manual signal control, traffic regulation
Demand Conflict (LTIP)	Area2Area	Top Level	CMS, manual signal control, traffic regulation

8.2.1 Top level: Global network performance enhancement

For the top level traffic assignment principles, global network performance enhancement is the major issue, which can be considered in two aspects global performance enhancement and demand conflict resolution. These considerations are practiced by two principles, which are global enhancement principle (***GEP***) and conflict resolution principle (***CRP***).

(A) Global enhancement principle (***GEP***)

GEP is a general concept for all the possible actions to improve the traffic network system performance. Our idea is to apply the 80-20 rule in the management science, that is, major traffic demands serve first, especially the congested demands. The system should consider re-organizing any possible available resources to improve the global network performance, such as directed signal chaining control, reversible lane, priority manual traffic control, etc. In other words, improve the efficiency of the congested major traffic demands may enhance the global traffic network performance. The congestion propagation pattern (***CPP***) , discussed in the Section 5.3.1, which fits the criteria.

In the implementation of the ATMS DSS, the ***GEP*** module should have the following two processes. First, select all the ***CPP***s and sorts them by their demands. Second, find out the ***CPP***s without demand conflict and arrange more resources for handling the ***CPP***, such as prioritized directed signal chaining, reversible lane, etc. The demand conflict ***CPP***s are left to the conflict resolution principle module.

(B) Conflict resolution principle (***CRP***)

For the demand conflict pattern (***DCP***), the two traffic demands cannot be satisfied simultaneously. So that a decision must be made to compromise on the major demand appropriately in order to resolve the conflict and enhance the global performance. For example, as illustrated in Figure 8-2, two ***CCP*** demands ***A to C***, and ***B to D*** are conflict, where the demand of ***B to D*** is larger than ***A to C*** and thus the traffic assignments will be conceded more resources to demand ***B to D***.

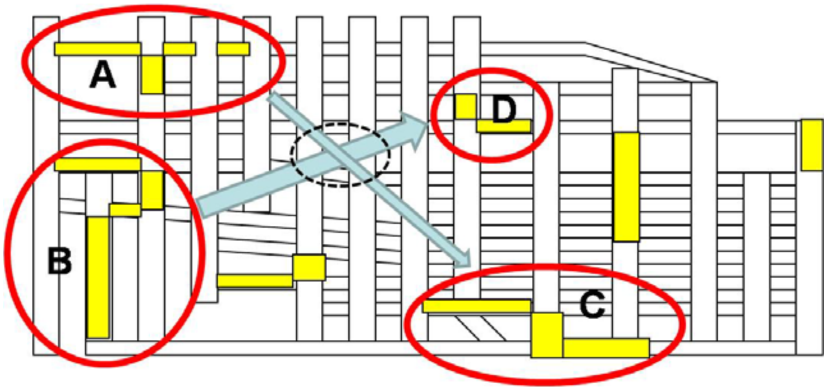


Figure 8-2 Demand conflict pattern

In the ***DCP***, there are two types of conflicts when the traffic demands meet at the intersection, which are cross conflict pattern (***CCP***) and left-turn interlaced pattern (***LTIP***), as shown in Figure 8-3. The solution for the ***CCP*** is compromised on the major demand, as illustrated in Figure 8-3(a), and the solution for the ***LTIP*** is increasing the left turn green time, as illustrated in Figure 8-3(b).

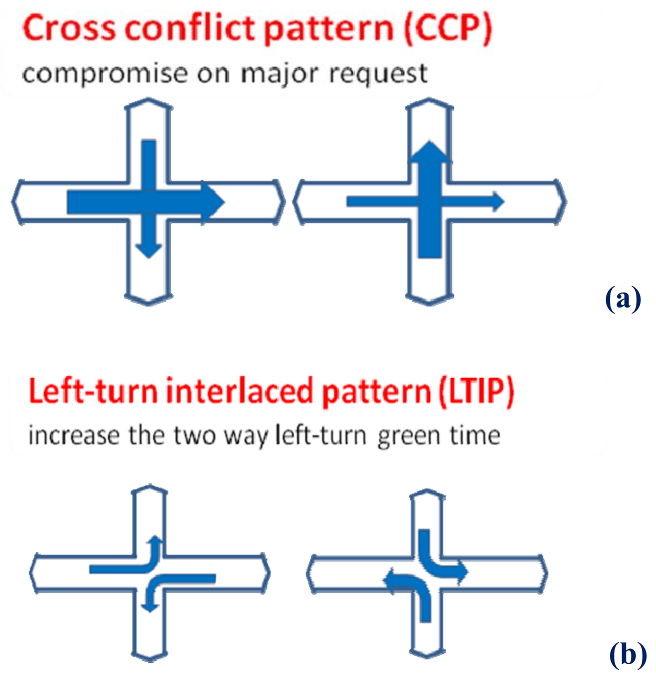


Figure 8-3 (a) compromised on major request, (b) increasing the two way lef-turn green time

8.2.2 Mid level: Traffic smoothing

As illustrated in Table 8-2, the top level traffic assignment principle handles the area to area traffic demands assignment, and the mid level principle considers the area level traffic smoothing. In this level, two principles are designed including information broadcast principle (***IBP***) and traffic smoothing principle (***TSP***).

(A) Information Broadcast Principle (***IBP***)

In Chapter 4, the collaborative traffic information generation and sharing framework built-up a traffic information broadcasting mechanism, which provides several channels of traffic information provisioning, such as CMS (changeable message sign), ***STA*** (smart traffic agent), radio broadcasting, etc. The knowledge of the traffic information helps the travelers to decide the travelling path, and thus avoid the congestion and decrease the journey traveling time. In the ATMS DSS, important congestion message are handled by the information broadcast principle module, such as traffic events, predicted congested area, ***STB***, etc.

(B) Traffic Smoothing Principle (***TSP***)

Traffic smoothing principle (***TSP***) considers the area level traffic smoothing assignment, while ***CPP*** is one of the major targets. ***TSP*** allocates more resources to resolve the congestion propagation, such as increasing signal green time at the directed intersection. As illustrated in Figure 8-4, ***TSP*** suggests increasing the demand based directed green time of each intersection from the ***Origin*** area to the ***Destination*** area in order to smooth the congested traffic flows, which is the major cause of the ***CPP***.

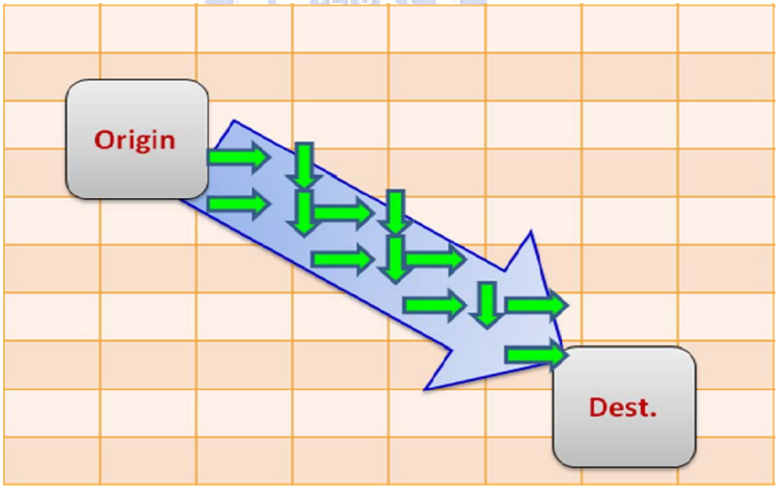


Figure 8-4 Traffic smoothing assignment for congestion propagation pattern

Another example is the reversible lane, which allocates lanes from the reverse direction to the congested direction in order to serve the congested traffic flow. In the ***TSP***, if several continuous ***SCO***s located in the path of ***CPP***, then connection of these ***SCO***s is the suggested reversible lane candidate. If reversible lane is not easy to apply, the signal chaining can be a good candidate solution. As illustrated in Figure 8-5, the traffic demand of the east direction is much larger than the other three directions, hence results in congestion in the east direction links; increase the green time of the east direction in the related intersections will smoothly harmonize the congestions.

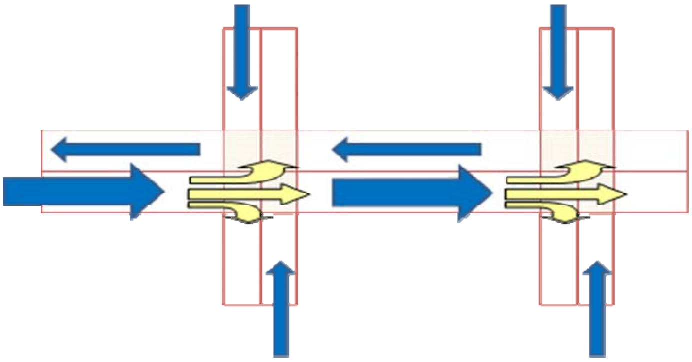


Figure 8-5 Signal chaining traffic smoothing assignment

8.2.3 Entry level: Local optimization

The entry level principles have two purposes: compromises with upper level principles and consider the local optimum of the single network object, such as intersection or link. In the entry level, two principles are designed for these purposes including priority control principle (***PCP***) and give way principle (***GWP***).

(A) Priority Control Principle (***PCP***)

PCP practices the concept of compromise, that is to say, if lower principle does not obey the upper level principle, then the ***PCP*** sacrifices it to the upper level principle. The ***PCP*** can be implemented as a priority control rule: "*If the actions suggested by lower level principles are conflict with those suggested by higher level principles, then abort lower level actions*".

(B) Give Way Principle (***GWP***)

The ***GWP*** deals with the network object congestion from the object level point of view. For the intersection object as an example, as illustrated in Figure 8-6, east direction demand is larger than the other three directions, then the ***GWP*** suggests to increase the green time for the longest wait queue (east direction in the Figure) to enhance the performance of the intersection.

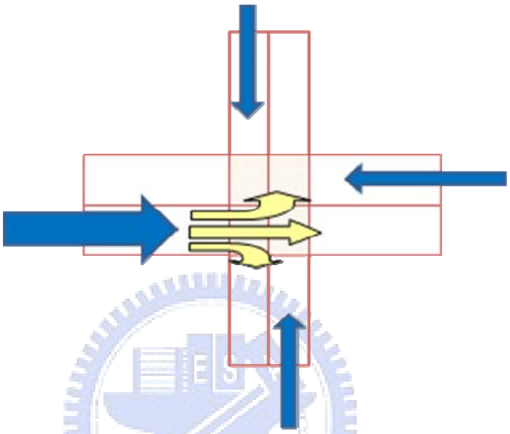


Figure 8-6 Intersection give way signal control

8.3 ATMS ontology

Based upon the three-layer traffic assignment principles, an ATMS ontology is designed for the purpose of global traffic network performance enhancement by cooperation of the traffic domain experts, knowledge acquisition engineers, and system design engineers. As shown in Figure 8-7, the ATMS ontology is composed of several concepts and several types of relationships among these concepts. The relationships connecting concepts represent some interactions or connections between these concepts, such as “***is-a***”, “***includes***”, “***related-to***”.

For example, “***traffic assignment***” concept **includes** “***traffic signal control***”, “***changeable message sign***”, and “***traffic assignment suggestion***” concepts, and **related to** “***link/node service level estimation***” concept. Each concept is implemented by one or several knowledge classes and stored in the knowledge base, and the relationship is implemented as inheritance, interface, or inclusion relationships between these knowledge classes. The concepts in red title indicate that these concepts implement the three-layer traffic assignment principles, as shown in Figure 8-7.

The ontology of the ATMS DSS can be divided into two parts, as shown in Figure 8-7, where the concepts of the bottom part are transformed from the traffic knowledge generated from STP/STB mining system, traffic status prediction, travel time prediction systems which are discussed in the previous Chapters. The concepts in the bottom part includes several traffic knowledge concepts derived from two systems, where “***spatiotemporal traffic pattern***”, “***spatiotemporal bottleneck***”, “***intersection delay***”, “***congestion propagation pattern***”, and “***demand conflict pattern***” concepts are transformed from ***STP***/***STB*** mining system (discussed in Chapter 5 and Chapter 6), and “***link/node service level estimation***”, “***real-time link status***” concepts are derived from the traffic status prediction / travel time prediction system (discussed in Chapter 4 and Chapter 7). These transformation processes are batch executed so that the traffic knowledge are periodically updated to catch the latest real status of the urban network.

The upper part of the ATMS DSS implements the hierarchical three-layer traffic assignment principles, where the principle concepts will invoke the related concepts for deep reasoning to find out the solution. The process of the ATMS DSS is a knowledge based inference process which is triggered by the “***network optimization***” concept, it then triggers related concepts and the multi thread inference processes are separately going on in order to generate the traffic assignment suggestions. All these assignment suggestions are aggregated at the “***assignment resolution***”, which implements an aggregation and resolution mechanism to reconcile the conflict traffic assignments. It collects all the traffic assignments generated by different principles, and decides which traffic assignments should be exported as a formal result by considering the conflict resolution issue through the “***give way principle***” and “***priority control principle***” concepts.

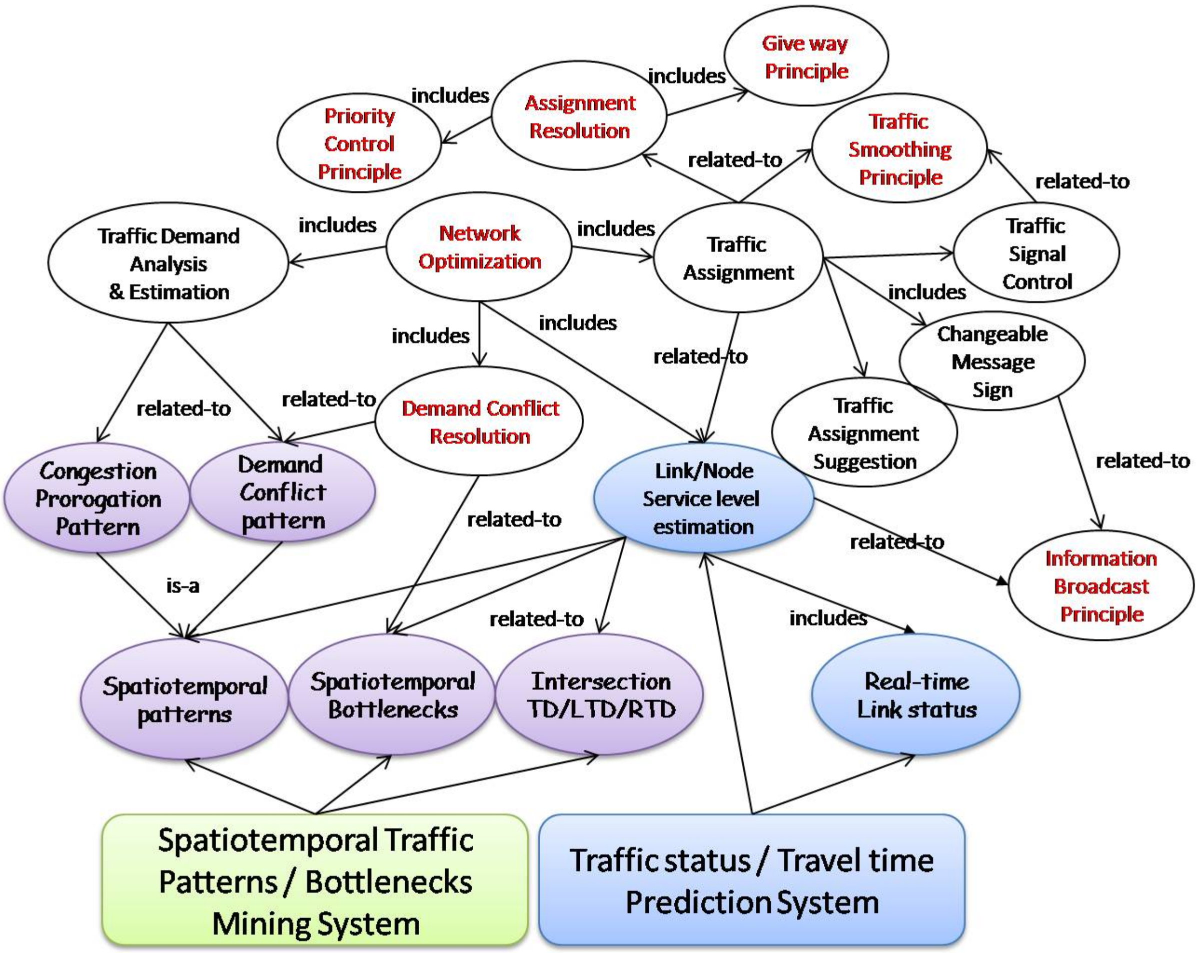


Figure 8-7 Ontology for the ATMS Decision Support System

Chapter 9 Conclusions and Future Works

9.1 Conclusions

Global traffic network performance enhancement is one of the major tasks in ATMS, and it is the common hope of all the traffic network travelers as well as the network administrators. However, this task is not only complex but also difficult due to several factors: network complexity in urban area, real-time sensor data unavailability, spatiotemporal traffic demand dynamics, spatiotemporal traffic bottleneck, and traffic demand conflict. These factors result in that traditional ***DTA***-based approaches are hardly practical to real urban network.

In this dissertation, a novel approach based upon different philosophy of thinking is proposed, which builds the solution based on the concept of "*continuously enhancing the global network performance*" rather than the "*optimal traffic assignment*" concept in the traditional ***DTA***-based approach. We try to enhance the global traffic network performance by four facets, including decreasing the traffic demand, discovering traffic patterns and identifying traffic bottleneck, resolving the traffic bottlenecks, and creating a collaborative traffic information generation and sharing framework for two-ways information exchanging between the travelers and the backend traffic information center. A series of works toward to the goal of enhancing the global network performance are discussed in this dissertation, including traffic information collection (Section 2.1), spatiotemporal traffic patterns and bottleneck discovering (Chapter 5 and Chapter 6), traffic information generation and sharing framework (Chapter 4), multi-sources heterogeneous traffic information fusion (Section 4.3), traffic knowledge to inference rules transformation, travel time prediction system (Chapter 7) and ATMS decision support system (Chapter 8). These works are organized as a three-layered traffic knowledge framework, as illustrated in Figure 9-1, which includes **data process layer**, **data mining layer**, and **application layer**. As shown in the Figure, data collection, cleansing and traffic information generation are categorized in the **data process layer**, which collects raw data from the LBS-based applications, and the generated traffic information such as ***TIS***, vehicle journey and ***TNS*** are stored in the ***TIDB***. In the **data mining layer**, several hypothesis based spatiotemporal data mining models and heuristics are proposed for ***STP***s mining and ***STB*** discovering to find out the traffic knowledge, which are transformed into rule classes and stored in the knowledge base. The traffic information, discovered traffic knowledge, domain expertise, and ontology constitute the hierarchical knowledge structure of the applications, as illustrated in Table 8-2. In the **application layer**, two practical applications are discussed: real-time travel time prediction system for travelers and ATMS decision support system for the network administrators. Knowledge based system technique is adopted in these two applications, where the discovered traffic knowledge combined with domain expertise constitutes the knowledge base, and real-time as well as historical traffic information, external traffic events data sources are regarded as facts in the inference engine.

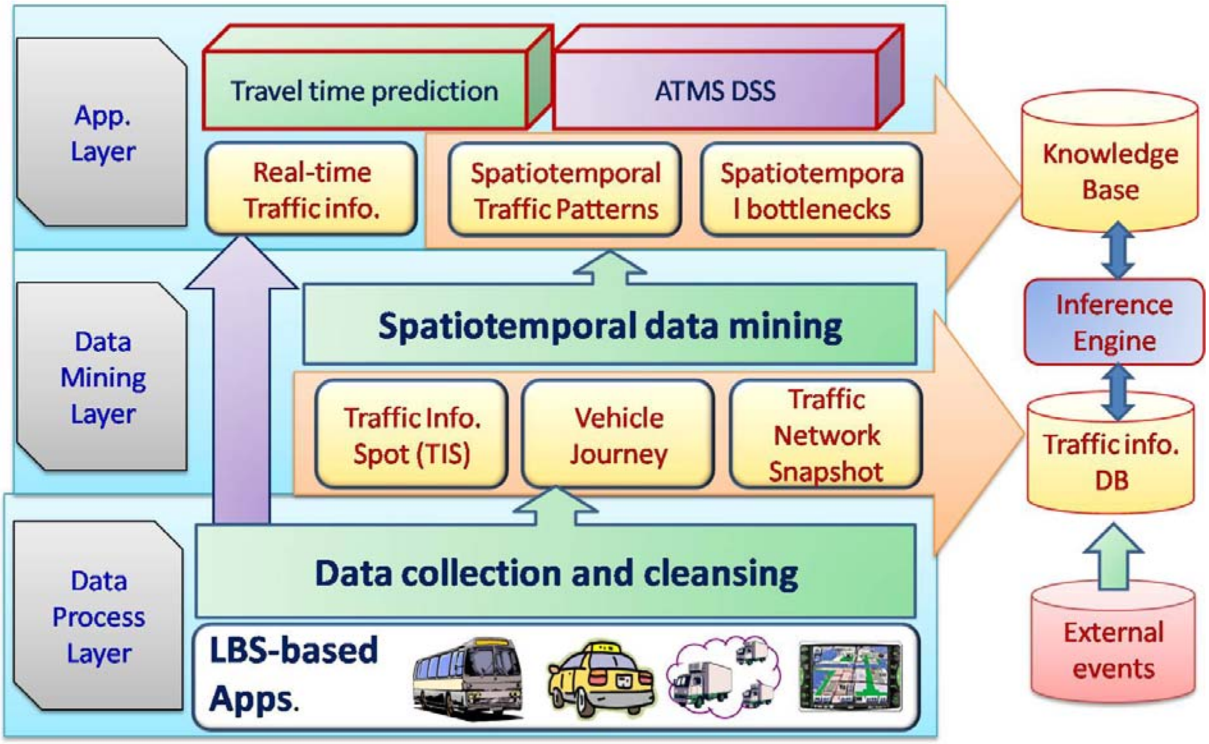


Figure 9-1 Three-layer traffic knowledge framework

By the traffic information derived from LBS-based applications as discussed in Chapter 3, the vehicles in the LBS-applications can be regarded as traffic probing vehicles, which solve the cost, coverage, and real-time issues in traditional sensor-based or probing vehicle based surveillance system. In addition to traffic information collected and derived from the LBS-based applications, a wiki-like collaborative real-time traffic information collection and sharing framework based on the high penetration rates of location aware mobile devices is discussed in Chapter 4. By this collective traffic information generation scheme, more real-time traffic data will be collected cost-effectively and accurately, and the spatiotemporal coverage is better than the traditional traffic information collection scheme. Moreover, the framework integrates the generated traffic information with external traffic information sources into ***TIDB*** based on optimal weighting approach to minimize the data uncertainty, and shares the traffic information with all the framework users by various data broadcasting channels.

The ***STP*** mining and ***STB*** discovering models discussed in Chapter 5 and Chapter 6 reveal the traffic bottleneck and traffic demand conflict issues in global traffic network performance enhancement task. Several spatiotemporal traffic patterns are defined and discovered from the ***TIDB*** based on the hypothesis based data mining techniques, and three heuristics are proposed to discover the ***STB*** in urban network by analyzing the discovered traffic patterns. On the other hand, ***CASR*** rules derived from the ***CPP*** can be a good explanation utility for describing the relationship between ***O-D*** traffic demand and congestions. Moreover, the discovered traffic patterns and ***O-D*** demand relationships present the congestion predictive capability, which provide decision support information for the ATMS administrator to take appropriate actions to solve the bottlenecks and thus enhance the global network performance.

In Chapter 7, a knowledge based real time travel time prediction (***TTP***) system is designed and implemented based upon the model proposed in this work, which contains real-time and historical travel time predictors and a dynamic weighted combination scheme. The discovered spatiotemporal traffic patterns are transformed to the prediction rules in the historical travel time predictor, and real-time traffic information constitutes the real-time predictor and facts in the inference engine. Meta rules donated by traffic domain experts dynamically adjust the weight control variables for the linear combination of these two predictors by considering the effects and severity of the traffic events collected from the external traffic data sources. The experiments result show that real time predictor has better precision than historical predictor. By combining domain expert knowledge in meta-rules, weight combination predictor has better performance than other two predictors in both RME and RMSE. In addition to the ***TTP*** system designed for the travelers, a knowledge based advanced traffic management system (ATMS) decision support system (DSS) targeted to improve the global network performance is discussed in Chapter 8, which provides valuable traffic assignment suggestions for the traffic network administrators. It utilizes the discovered traffic knowledge discussed in this work, such as traffic information database, ***STP***/***STB***, predicted traffic status, and combines the domain expertise in order to enhance the global network performance and make traffic assignment suggestions. A hierarchical three-layered traffic assignment principles model combined with several traffic assignment principles is discussed to handle the different level of granularities of traffic patterns and traffic congestions. These traffic assignment principles donated by domain experts try to relieve traffic congestion and resolve traffic bottlenecks as well as traffic demand conflict. Compromise concept is practiced in these principles to let the actions suggested by higher level principle have the major priority due to the higher level principles deal with the major traffic demands. The kernel of the ATMS DSS is the domain ontology which consists of two parts: meta-rules donated by the traffic domain experts and rules transformed from the spatiotemporal traffic patterns and bottleneck. The ATMS DSS is implemented as a knowledge-based system, where the rules are transformed from the ontology and facts come from the real-time traffic information in ***TIDB***.

9.2 Future works

In this dissertation, a series of works which constitute the foundation of the traffic information framework from the traffic information processing layer, traffic knowledge generation layer to the traffic application layer have been discussed. Two applications including travel time prediction and ATMS traffic assignment are designed based upon knowledge based technique. The ontologies donated by the domain experts, which are the kernel of these two applications, need to be continuous updated in order to improve the precision and performance of the systems. In the knowledge based system, knowledge mined from the ***TIDB*** is combined with the expert knowledge for the inference and usually the knowledge donated by domain experts has the priority over the knowledge extracted from the system when the knowledge have different actions on the same objects. However, domain expertise may not 100% accurate in every case so that a more sophisticated rules conflict resolution mechanism is needed.

Comparing to the traditional ***DTA***-based approaches, the works discussed in this dissertation are more applicable to the real traffic network, especially for the large urban network. However, due to the insufficient of the real road network and topologies data, the experiments and applications are implemented based upon a simplified road network model in Taipei urban area, as shown in Figure 7-5. In the future, these works should be revised based upon more details and precise urban road network model, which should be established by the assistance of the professional GIS organizations.

Finally, the ground works, traffic information database, travel time prediction and traffic assignment and management knowledge base, and traffic information framework built in this work are well structured and can be applied to more ITS applications in the future, such as advanced traveler information system (ATIS), commercial vehicle operation (CVO), advanced public transportation system (APTS), or emergency management system (EMS), etc.

References

[AI05] Xu Aigong, “Research on a new GPS/GIS based ERP system”, in *Proceedings of IEEE Int. Conf. Intelligent Transportation Systems,* pp. 455-457, 2005.

[BCK04] S.I. Bajwa, E. Chung, and M. Kuwahara, “An Adaptive Travel Time Prediction Model Based On Pattern Matching”, in *Proceedings of 11th World Congress on ITS,* Oct. 2004.

[CAT00] Catling, I., “Road user charging using vehicle positioning systems”, in *Proceedings of IEEE Int. Conf. Road Transport Information and Control,* pp. 126-130, 2000.

[CH+03] E. Chung, et al., “Travel Time Prediction: Issues and Benefits”, in *Proceedings of the 10th World Conference on Transport Research*, 2003.

[CHU03] E. Chung, “Classification of Traffic Pattern”, in *Proceedings of 11th World Congress on ITS*, 2003.

[CHE07] Hsiao-Han Chen, “A Spatiotemporal Traffic Bottleneck Mining Model for Discovering Bottlenecks in Urban Network”, *Thesis of Master of NCTU*, 2007.

[CIT06A] Commission for Integrated Transport, “World review of road pricing: Phase 2 - case studies”, http://www.cfit.gov.uk/docs/2006/wrrp/wrrp2/case/index.htm, UK, Dec. 2006.

[CIT06B] Commission for Integrated Transport, “World review of road pricing: Phase 2 – final report”, http://www.cfit.gov.uk/docs/2006/wrrp/wrrp2/index.htm, UK, Dec. 2006.

[CK03] S. Chien and C. M. Kuchipudi, “Dynamic Travel Time Prediction with Real-Time and Historic Data”, *Journal of Transportation Engineering*, Vol. 129, Issue 6**,** pp. 608–616, 2003.

[CLO01] Tzu-Jung Cheng, Tyh-ming Lin, and Yu-Ching Ou-Yang, “2001 Highway Capacity Manual in Taiwan”, Government Publication No.:1009001056, 2001, Available: http://www.iot.gov.tw/.

[CMC07] Huiping Cao, Nikos Mamoulis, David W. Cheung, “Discovery of Periodic Patterns in Spatiotemporal Sequences”, *IEEE Trans. on Knowledge and Data Engineering*, Vol. 19, No. 4, pp. 308-320, Apr. 2007.

[CO03] DRAMA rule based expert system shell, Coretech Inc, Available: http://www.ctknow.com.tw/rule.htm, 2003

[CS+03] E. Chung, M. Sarvi, Y. Murakami, R. Horiguchi and M. Kuwahara, ”Cleansing of Probe Car Data to determine Trip OD,” in *Proceedings of 21st ARRB and 11th REAAA Conf.*, Cairns, Australia, 2003.

[DAK06] Davis, Aubrey and Matthew Kitchen, “Puget Sound Traffic Choices Study”, Puget Sound Regional Council, Washington State Transportation Commission, November, 2006.

[DE06] Toll Collect, website of Toll Collect, http://www.toll-collect.de/, 2006.

[DOT06] U.S. DOT's ITS website: http://www.itsoverview.its.dot.gov/, 2006.

[EIOT] National traffic information center, Available: http://e-iot.iot.gov.tw/, Institute of Transportation MOTC.

[FB+93] T. L. Friesz, D. Bernstein, T. E. Smith, R. L. Tobin and B. W. Wie, “A variational inequality formulation of the dynamic network user equilibrium problem”, *Operations Research*, Vol. 41, No. 1, pp. 179-191, 1993.

[FB03] Bolte F. and Wolfgang Beier, “Fee collection for heavy goods vehicles on German motorways”, *Proceedings of IEE Seminar on Road User Charging*, pp. 10/0~10/7, 2003.

[FL+90] T. L. Friesz, J. Luque, R. L. Tobin and Byung-Wook Wie, “Dynamic Network Traffic Assignment Considered as a Continuous Time Optimal Control Problem”, *Operations*

Research, Vol. 37, No. 6, pp. 893-901, 1990.

[HO+99] H. Hashimoto, H. Ohno, M. Konishi, K. Morishita, and K. Sugimoto, “Multi-lane electronic road pricing system in Singapore,” *Mitsubishi Juko Giho, Mitsubishi Heavy Industries Ltd*., vol. 36, no. 1, pp. 36–45, 1999.

[HVF07] HVF, website of Federal Office for Spatial Development ARE, Switzerland, http://www.are.admin.ch/, October 2007.

[IK01] T. Iroy and M. Kuwahara, “Simulation Analyses of Traffic Congestion Alleviation by Demand Spreading over Time”, *Journal of the Eastern Asia Society for Transportation Studies*, Vol.4, No.4, pp.161-174, 2001.

[IV+05] Safa I., Fernando V.L., Richard T.S., and Bongki M., “Spatiotemporal Aggregate Computation: A Survey,” *IEEE Trans. on Knowledge and Data Engineering*, Vol. 17, No. 2, pp 271-286, 2005.

[JAN91] B.N. Janson, “Dynamic Traffic Assignment for Urban Road Networks”, *Transportation Research, Part B: Methodological*, Vol. 25, No.2-3, pp. 143-161, 1990.

[JZ03] G. Jiang, R. Zhang, “Travel Time Prediction for Urban Arterial Road”, in *Proceedings of IEEE International Conf. on Intelligent Transportation Systems*, Vol. 2, pp 1459–1462, 2003.

[JOR01] Juan Guillermo Jordan, etc.,”A comparison of different technologies for EFC and other ITS applications”, in *Proceedings of IEEE Intelligent Transportation System Conference*, Aug. 2001.

[JW+00] Bor-Shenn Jeng, Ching-Hung Wang, Wei-Shiun Lee, Wern-Sheng Shieh, Ming-Cheng Chen, Jynn-Yu Wang, Kang-Chieh Hsiao, Mei-Yu Liu, and Chuang-Wen Hsu, “An Integrated Service for Backend Systems in Taiwan Electronic Toll Collection Project”, in *Proceedings of International Conference on ITS-2000*, 2000.

[KD+05] B. S. Kerner, C. Demir, R. G. Herrtwich, S. L Klenov, H. Rehborn,M. Aleksi, A. Haug, G. DaimlerChrysler A, “Traffic State Detection with Floating Car Data in Road Networks”, in *Proceedings of 8th IEEE International Conference on Intelligent Transportation Systems*, 2005.

[KER05] B. S. Kerner, ”Control of Spatiotemporal Congested Traffic Patterns at Highway Bottlenecks”, *Elsevier Physica A: Statistical Mechanics and its Applications*, Vol. 355, Issues 2-4, pp. 565-601, 2005.

[KER07] B. S. Kerner, “Control of Spatiotemporal Congested Traffic Patterns at Highway Bottlenecks”, *IEEE Trans. on Intelligent Transportation System*, Vol. 8, No. 2, pp. 308-320, 2007.

[KH07] Shoaib Kamran and Olivier Haas, “A Multilevel Traffic Incidents Detection Approach: Identifying Traffic Patterns and Vehicle Behaviours using real-time GPS data”, in *Proceedings of IEEE Intelligent Vehicle Symposium,* Istanbul, Turkey, pp. 912-917, 2007.

[KKS05] T. Kawahara, S. Kamijo and M. Sakauchi, “Travel time measuring by using vehicle sequence matching between adjacent intersections,” in *Proceedings of IEEE International Conference on Intelligent Transportation Systems*, pp. 712-717, 2005.

[KZ+05] Wei-Shinn Ku, Roger Zimmermann, Haojun Wang, and Chi-Ngai Wan, “Adaptive nearest neighbor queries in travel time networks”, in *Proceedings of ACM GIS*, pp. 210-29, 2005.

[LC03] D Levinson and E Chang, “A model for optimizing electronic toll collection systems”, *Transportation Research*, 37A(4), pp. 293-314, 2003.

[LJ+04] Wei-Hsun Lee, Bor-Shenn Jeng, Shian-Shyong Tseng, and Ching-Hung Wang, “Electronic Toll Collection Based on Vehical-Positioning System Techniques,” in Proceedings of *IEEE International Conf. on Networking Sensing and Control*, pp. 643-648, 2004.

[LG+08] J.C. Long, Z.Y. Gao , H.L. Ren and A.P. Lian, “Urban traffic congestion propagation and bottleneck identification”, *Science in China Series F: Information Sciences*, Vol.51, No. 7, pp. 948-964, 2008.

[LKM04] W. H. Lin, A. Kulkarni, and P. Mirchandani, “Short-Term Arterial Travel Time Prediction for Advanced Traveler Information Systems”, *Journal of Intelligent Transportation Systems*, Vol. 8, Issue 3, pp. 143–154, 2004.

[LTT03] Y. T. Lin, S. S. Tseng and C. F. Tsai, “Design and implement of new object-oriented rule base management system,” *Elsevier Expert Systems with Applications*, Vol. 25, pp. 369–385, 2003.

[LTT09] Wei-Hsun Lee, Shian-Shyong Tseng, and Sheng-Han Tsai, “A Knowledge base Real-Time Travel Time Prediction System for Urban Network”, *Elsevier Expert System with Applications*, Vol. 36, Issue 3, Part 1, pp. 4239-4247, 2009.

[LTW08] Wei-Hsun Lee, Shian-Shyong Tseng, and Ching-Hung Wang, “Design and Implementation of Electronic Toll Collection System Based on Vehicle Positioning System Techniques”, *Elsevier Computer and Communications*, Vol. 31, Issue 12, pp. 2725-2933, 2008.

[LW+04] H. Y. Liu, C. H. Wang, V. S. Shieh, and B. S. Jeng, “An intelligent taxi dispatching management system,” in *Proceedings of Cross Strait Conference on ITS*, pp. 111-117, 2004.

[LZ05] H. E. Lin and R. Zito, “A review of travel-time prediction in transport and logistics”, in *Proceedings of the Eastern Asia Society for Transportation Studies,* Vol. 5, pp. 1433–1448, 2005.

[LZC+05] H. Liu, H. Zulen, H. Lint, Y. Chen and K. Zhang, “Preditction of Urban Travel Times with Intersection Delays,” in *Proceedings of IEEE International conference on Intelligent Transportation Systems*, pp. 1062-1068, 2005.

[MAO07] C. K. Mao, “My Handset can be ETCized – (Taiwan’s VPS-ETC project report)”, in *Proceedings of Intelligent Transportation System World Congress*, 2007.

[MB04] Rapp, M.H. and U. Balmer, “The Swiss Distance Related Heavy Vehicle Fee (LSVA) – A Novel Approach to Area-Wide Road Charging”, Rapp Trans, AG, Basel, 2004.

[MN78A] D.K Merchant and G.L. Nemhauser, “A Model and an Algorithm for the Dynamic Traffic Assignment Problems”, *Transportation Science*, Vol. 12, No. 3, pp. 183-199, 1978.

[MN78B] D.K Merchant and G.L. Nemhauser, “Optimality Conditions for a Dynamic Traffic Assignment Model”, *Transportation Science*, Vol. 12, No. 3, pp. 200-207, 1978.

[MSR04] C. D. Mark, A. W. Sadek, and D. Rizzo, “Predicting Experienced Travel Time with Neural Networks: a PARAMICS Simulation Study”, in *Proceedings of IEEE conference on Intelligent Transportation Systems*, pp.906-911, Oct. 2004.

[NAG99] K. Nagaoka, ”Travel time System by using Vehicle Information and Communication System (VICS),” in *Proceedings of IEEE conference on Intelligent Transportation Systems*, pp. 816-819, 1999

[NT04] T. Nakata, and J. Takeuchi, “Mining Traffic Data form Probe-Car System for Travel Time Prediction,” in *Proceedings of ACM SIGKDD international conference on Knowledge discovery and data mining*, Seattle, WA, USA, pp. 817-822, 2004.

[PRO00] PRoGRESS Project 2000-CM.10390 – “Pricing Road Use for Greater Responsibility, Efficiency and Sustainability in Cities”, Main Project Report, http://www.progress-project.org/, July 2004.

[PRS] Police station public traffic service, Available: http://www.prs.gov.tw/, Police radio station.

[PZ01] Srinivas Peeta and Athanasios Ziliaskopoulos, “Foundations of Dynamic Traffic Assignment: The Past, the Present and the Future”, *Network and Spatial Economics*, Vol. 1, pp. 233-265, 2001.

[RZ04] John Rice and Erik van Zwet, “A Simple and Effective Method for Predicting Travel Times on Freeway”, *IEEE Trans. on Intelligent Transportation System,* Vol. 5, No. 3, pp 200–207, 2004.

[SK07] DSRC Toll collection system in South Korea, http://www.hipassplus.co.kr/, 2007

[SL+05] Wern-Yarng Shieh, Wei-Hsun Lee, Shen-Lung Tung, and Chung-Ding Ho, “A Novel Architecture for Multilane-Free-Flow Electronic-Toll-Collection Systems in the Millimeter-Wave Range", *IEEE Trans. on Intelligent Transportation System*, Vol. 6, No. 3, pp. 294-301, 2005.

[SL+06] Wern-Yarng Shieh, Wei-Hsun Lee, Shen-Lung Tung, Bor-Shenn Jeng, and Cheng-Hsin Liu, “Analysis of Road-Side-Unit and On-Board-Unit Optimum Configuration in Dedicated Short-Range Communication Systems", *IEEE Trans. on Intelligent Transportation System*, Vol. 7, No. 4, pp. 565-571, 2006.

[SW64] Shannon C, Weaver W: “The Mathematical Theory of Communication”, Urbana, University of Illinois, 1964

[SH+03] M. Sarvi, R. Horiguchi, M. Kuwahara, Y. Shimizu, A. Sato, and Y. Sugisaki “A Methodology to Identify Traffic Condition Using Intelligent Probe Vehicles,” in *Proceedings of 10th World Congress on ITS*, Madrid, 2003.

[SMI93] M.J. Smith, “A new dynamic traffic model and the existence and calculation of dynamic user equilibria on congested capacity-constrained road networks”, *Transportation Research, Part B: Methodological*, Vol. 27, No.1, pp. 49-63, 1993.

[TAN] TANFB Traffic information system, Available: http://1968.nfreeway.gov.tw/, Taiwan Area National Freeway Bureau, MOTC.

[TAO07] Chi-Chung Tao, “Dynamic Taxi-sharing Service Using Intelligent Transportation

System Technologies", in *Proceedings of IEEE Wireless Communications, Networking and Mobile Computing,* pp. 3209-3212, 2007.

[TLT05] Sheng-Han Tsai, Wei-Hsun Lee, and Shian-Shyong Tseng, "A spatiotemporal traffic patterns mining on LBS-based application", in *Proceedings of 10th Conf. on Artificial Intelligence and Applications*, Kaohsiung, Taiwan, Dec. 2005.

[TSA06] Sheng-Han Tsai, "A Knowledge Based Real-Time Travel Time Prediction System using Location Based Service", *Thesis of Master of NCTU*, 2006.

[TTI03] H. Takahashi, S. Takahashi, and T. Izumi, "Travel Time measurement using Genetic Algorithm in Vehicle Sequence Matching Processing", in *Proceedings of 2nd ITS Symposium*, Osaka, 2003

[TW07] DSRC Toll collection system in Taiwan, http://www.fetc.net.tw/, 2007

[WC+03] 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 *Proceedings of TGIS 2003 Annual Conference*, Taipei, Taiwan, pp.25-26, 2003.

[WHL04] C. H. Wu, J. M. Ho, and D. T. Lee, "Travel Time Prediction with Support Vector Regression", *IEEE Trans. Intelligent Transportation System*, Vol. 5, No. 4, pp 276–281, 2004.

[WLC05] Y. H. Wen, T. T. Lee, and H. J. Cho, "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, 2005.

[WU05] Ruei-Feng Wu, "The Optimal Weighting Approach for Traffic Data Fusion", Thesis of Master of NCTU, 2005.

[YAN05] J. S. Yang, "Travel time prediction using the GPS test vehicle and Kalman filtering techniques", in *Proceedings of American Control Conference,* pp. 2128–2133, 2005.

[YKP05] B. S. Yoo, S. P. Kang, and C. H. Park, “Travel Time Estimation Using Mobile Data”, in *Proceedings of the Eastern Asia Society for Transportation Studies*, Vol. 5, pp. 1533-1547, 2005.

[YX+02] DK Yang, AG Xu, QS Zhang, JT Ong, CL Law, “Some technical considerations on next ERP system in Singapore”, in *Proceedings of IEEE Int. Conf. Intelligent Transportation Systems,* pp. 407-410, 2002.

[ZIL00] A. K. Ziliaskopoulos, “A Linear Programming Model for the Single Destination System Optimum Dynamic Traffic Assignment Problem”, *Transportation Science*, Vol. 34, No. 1, pp. 37-49, 2000.

[ZL97] Yifeng Zhou, Henry Leung, "Minimum Entropy Approach for Multisensor Data Fusion", in *Proceedings of IEEE Signal Processing Workshop on Higher-Order Statistics* (SPW-HOS '97), pp. 0336, 1997.

[ZW+04] A. K. Ziliaskopoulos, S. T. Waller, Y. Li, and M. Byram “Large-Scale Dynamic Traffic Assignment: Implementation Issues and Computational Analysis”, *Journal of Transportation Engineering*, Vol. 130, No. 5, pp. 585-593, 2004.

[ZZ+04] H. S. Zhang, Yi Zhang, Z. H. Li, and D. C. Hu, “Spatial-Temporal Traffic Data Analysis Based on Global Data Management Using MAS,” *IEEE Trans. on ITS*, Vol. 5, No. 4, pp. 267-275, 2004.

Appendix A: Design and Implementation of a Vehicle-Positioning Based Electronic Toll Collection System

A.1 Introduction

Electronic toll collection (ETC), also known as electronic payment and pricing system, is one of the major research topics in intelligent transportation system (ITS) [DOT06]. ETC is an implementation of a road pricing concept in order to create benefits such as increasing the capacity of toll stations, reducing toll paying time, enhancing the convenience and safety of travelers, and minimizing air pollution and fuel consumption. It enables freeway toll plaza, bridge, tunnel, and turnpike operators to save on staffing costs while reducing delay for travelers and improve overall traffic performance. ETC system determines whether the vehicles passing are enrolled in the program, alerts enforcers for those that are not, and debits electronically the accounts or the amount in the IC card of registered cars without their stopping. The traditional technologies used in ETC system are classified as DSRC (dedicated short range communication) system since the on board unit (OBU) installed in the vehicle can only communicate with the road side unit (RSU) within a short range area, for example, 30 meters. The technologies used in DSRC-based ETC system are classified by two categories: infrared and microwave, which are named by their communication media. The evolution of ETC technology has brought DSRC-based ETC system from SLFF (single lane free flow) to

MLFF (multilane free flow), which do not constrain the vehicle moving on single lane while moving through the tolling zone. However, there are several drawbacks in DSRC-based ETC system, including complexity, cost ineffective, difficulty in system integration, and lack of RSU re-location flexibility.

Area wide integrated MLFF road charging system is now currently on its development to replace DSRC-based ETC systems. Vehicle positioning system (VPS) technology has become the new trend for road charging system, which implements ETC system based on positioning and mobile communication technologies. There are two major differences between VPS-based and DSRC-based ETC systems: communication mechanism and toll collection media. The communication mechanism used in VPS is mobile communication such as GPRS / UMTS / HSPA, which are the standard mobile communication protocols. Although there are some standards or protocols for DSRC-based ETC, most of them cannot cooperate with each other. Comparing to DSRC-based ETC system, VPS-based ETC system has following advantages: cost effective, RSU simplification, no communication zone restriction, service extensibility, and easy to migrate from lane-based to distance-based toll collection scheme. The toll mechanism in VPS is based on interaction between OBU and backend system through mobile network instead of the communication with RSU in the DSRC-based ETC system. The advantage of this mechanism is that there is no need to build up complex RSU as in DSRC-based ETC system, which has the flexibility of tolling zone relocation. The cost of system construction and maintenance can be largely reduced. For the extensibility issue, since there are mobile communication, positioning and electronic payment mechanisms in VPS-based ETC system, it can be easily extended to telematics service or m-commerce service without extra facilities in in-vehicle device (OBU), for example, park fee collection, vehicle navigation, etc. Furthermore, OBU in DSRC-based system is usually proprietarily owned by operator or manufacturer, but it is users' choices in VPS-based ETC system, such as dedicated OBU, VPS enabled smartphone/PDA/PND (personal navigation device)/UMPC (ultra mobile PC), etc.

ETC system is operated by the coordination of several subsystems, including debit transaction subsystem, enforcement system, OBU, mobile communication system and the backend system. Enforcement subsystem consists of AVI (automatic vehicle identification) and license number recognition. In this work, we discuss the design issues about VPS-based ETC system, and a prototype system including backend system (debit transaction subsystem) and frontend devices (OBU) are designed and implemented for field test. The system adopts GPRS mobile network and GPS positioning system as the basis. Several types of OBU including dedicated OBU, smart phone, PDA are implemented. Field test is carried out in the National freeway No.1 and No.3 in Taiwan for several months and a testing fleet consisting of 10 freeway scheduled buses is created in cooperation with the bus company.

A.2 VPS system background and related works

DSRC-based ETC technique is in its development towards to the precision of MLFF [SL+05] and optimum configuration [SL+06]. Currently in worldwide, many countries and cities have conducted the DSRC-based ETC system for their road pricing or congestion pricing policies, and many schemes have been adopted: Stop and Go, SLFF, and MLFF [CIT06A, CIT06B]. For example, Taiwan [TW07] and South KOREA [SK07] have announced the national wide ETC service using the DSRC-based ETC system since 2006. On the other hand, ANPR (automatic number plate recognition) technique is also applied for the urban cities road pricing, which is adopted by several cities: London, Rome, Edinburgh, etc.

However, the identification precision of ANPR technique which is not as good as the DSRC or VPS technique may result in extra manual operation cost. It is considered more suitable to apply to the enforcement system as a backup of toll collection system.

Vehicle positioning technique currently has been applied to many popular applications, including vehicle navigation, vehicle tracking, fleet management, location based services and telematics services. VPS-based ETC system is a category of location based service which tolls the vehicles by determining if they move into the tolling zone, and it can be used for the freeway toll collection or the road pricing in the urban area. The concept of toll collection by VPS scheme began in [CAT00], and the first VPS field test project ERP (electronic road pricing) was carried on in Hong Kong in 1997. The overview and design issues of the VPS-based ETC system were further discussed in [LJ+04] and [AI05]. Currently, there are some countries have conducted or prepared to conduct VPS based toll collection scheme for road pricing. Toll Collect Project [FB03, DE06] for the trucks on Autobahns in Germany has been in operation since 2005, which tolls the trucks by mileage based scheme calculated by the OBU, but the enforcement system is still based on the DSRC technique. HVF (heavy vehicle fee) project [HVF07, MB04] in Switzerland combines DSRC and VPS technique to collect toll fee for heavy vehicle. In US, traffic choices study project had a field test for road pricing by VPS technique in Puget Sound [DAK06]. Eight cities in Europe had joined the PRoGRESS project [PRO00] for urban road pricing, which has the goal of “demonstrate and evaluate the effectiveness and acceptance of integrated urban transport pricing schemes to achieve transport goals and raise revenue” [PRO00]. The comparison of technology and the toll schemes adopted by these cities are listed in Table A-1. Among them, three cities adopted VPS technique including Copenhagen, Bristol, and Gothenburg for different toll schemes: cordon-based, zone-based, and distance-based.

Table A-1 Comparisons of the toll scheme and technology in PRoGRESS [PRO00]

Scheme concept	**Road-pricing technology basis**		
	DSRC - electronic tag	**ANPR**	**GPS**
Cordon (per trip)	Rome, Helsinki	Bristol, Genoa, Rome	Copenhagen Bristol
Cordon (per day)	--	Edinburgh	--
Zone (per trip)	Trondheim, Helsinki	--	Copenhagen
Distance-based	--	--	Copenhagen Gothenburg Bristol

A.3 VPS system architecture

The system architecture of VPS-based ETC system is illustrated in Figure A-1, which includes four key components: OBU, enforcement system, mobile communication system, and backend system. It combines several technologies including vehicle positioning, mobile communication, vehicle detecting and classification, and auto license plate recognition, OBU and backend system. OBU is a device installed in the vehicle, with computing, positioning (GPS) and communication modules. Mobile communication network, such as GPRS/UMTS/HSPA, provides the communication media between OBU and the backend system. Enforcement system is used in reducing unpaid tolls by vehicle image capture, automatic vehicle classification and automatic number plate recognition. Backend system is in charge of all the information processing including online data collection and processing, billing transactions, customer service center, and traffic information, debit transaction and enforcement post matching, etc. OBU and backend system communicate each other through the dedicated VPN (virtual private network) based on GPRS/UMTS/HSPA mobile network provided by telecommunication operator.

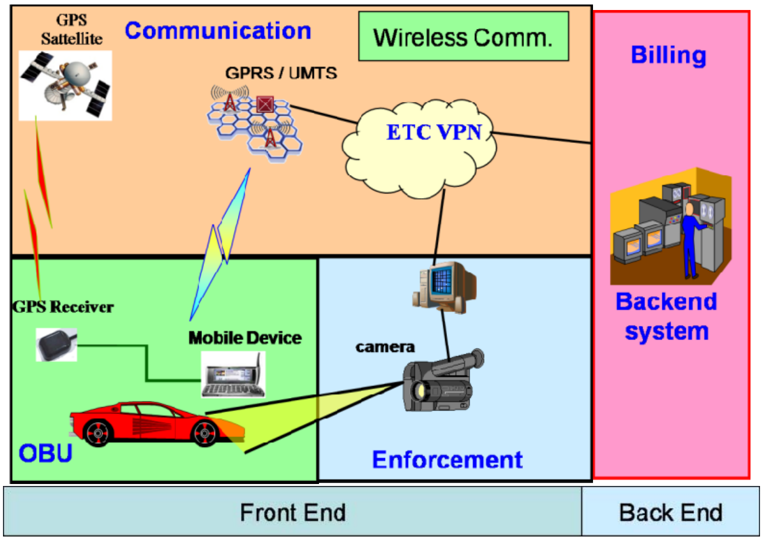


Figure A-1 Architecture of VPS-based ETC system

The tolling process is carried on at predefined toll areas, known as virtual toll stations, where vehicles will be tolled when it moves into these toll areas. As shown in Figure A-2, a toll zone is a rectangle area identified by a pair of coordinates {(x_1,y_1), (x_2,y_2)}, which covers all the vehicles moving possibilities area in the same moving direction. There are two parts in the tolling zone: notification area and toll area. The vehicle installed with an OBU moving into the notification area will be notified that a tolling transaction is going to be carried out, and the debit transaction is going on when the vehicle passing through the toll area. A virtual toll station has two tolling zones, one for each direction. Enforcement line is a gantry lies across these two tolling zones and has enforcement devices such as cameras, vehicle classification sensors installed on it in order to identify the vehicle class and capture the vehicle license plates. Camera modules in the enforcement system takes pictures for every vehicle entering the toll area and license plate recognition module recognizes license plate number by image processing technology in order to discriminate the registered and un-registered vehicle by matching the license number with the tolling transaction information.

The matched results provide the information and evidence for post processing and law enforcement. If unregistered vehicles enter into the tolling zone, pictures and license plate taken by enforcement system cannot be matched with the debit information. Based on the license plate number, the system can get more detailed information about the violated vehicles such as their names and addresses through the support of the vehicle management system of the government for the violation processing.

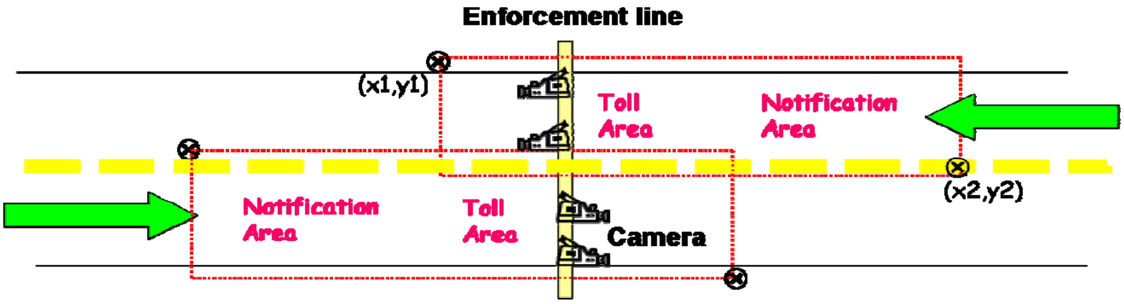


Figure A-2 Virtual toll zone configuration in VPS system

All the virtual toll stations coordinates are secure packed as toll zone table and kept in the storage of OBU. OBU detects whether if the vehicle (OBU) is entering into the toll zone (notification area and toll area) by the current vehicle position coordinate read from the GPS receiver, and sends transaction request message to the backend system by the communication module through mobile network when it is confirmed that the vehicle is entering in the toll area. The registered vehicle license number information is sent to the enforcement system in the RSU for the license plate recognition and matching when the backend system receives the transaction request, and the account of the registered vehicle will be debited by querying the toll table and the tolling transaction is processed by the backend system. The OBU then receives and displays the transaction result and local traffic information replied by the backend system after the transaction request is done.

A.3 VPS components design

VPS system and components as well as the communication protocol between OBU design issues are detailedly discussed in this section. As illustrated in Figure A-1, the key components in VPS system include OBU, enforcement system, and the backend system. In order to meet localized ETC requirements, there are several issues and principles in the VPS system design, including: (1) support both tollbooth based and distance based ETC, (2) adopt standard mobile communication, such as GPRS/UMTS/HSPA, where users are free to choose mobile network service providers, (3) support flexible toll fee management for different area and different time division, (4) safety of the debit transaction must be assured, (5) violation vehicles should be punished by the enforcement system, and (6) debit transaction should be survivable when the mobile network signal is unavailable or the communication module malfunctions. These key components as well as VPS message protocol design and implementation details are discussed in the following subsections.

A.3.1 OBU

There are several modules in OBU: positioning, mobile communication, computing, and human interface. The computing capability is used for running the VPS client program. There are several design principles of OBU to be considered: 1) OTA (over the air) software update capability, 2) GPS and cell ID dual mode positioning, and 3) transaction message request resend capability. Two types of OBU, smart phone OBU and dedicated OBU, are designed and implemented for the VPS system, as shown in Figure A-3(a) and Figure A-3(b) respectively. The smart phone OBU designed for testing functionality and compatible with client consumer products such as smart phone, PDA, etc, has the goal of minimizing the hardware effort in OBU. The dedicated type, on the other hand, is designed for testing stability and solidity, and is used for the field test project discussed in next section.

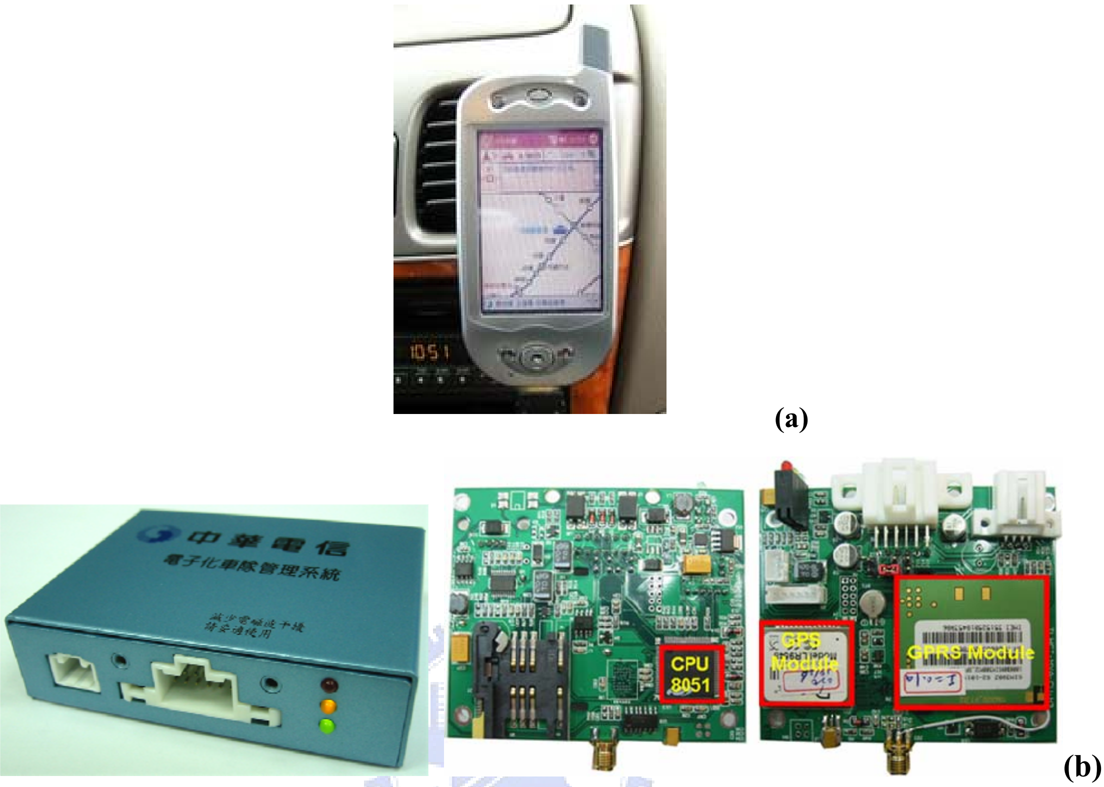
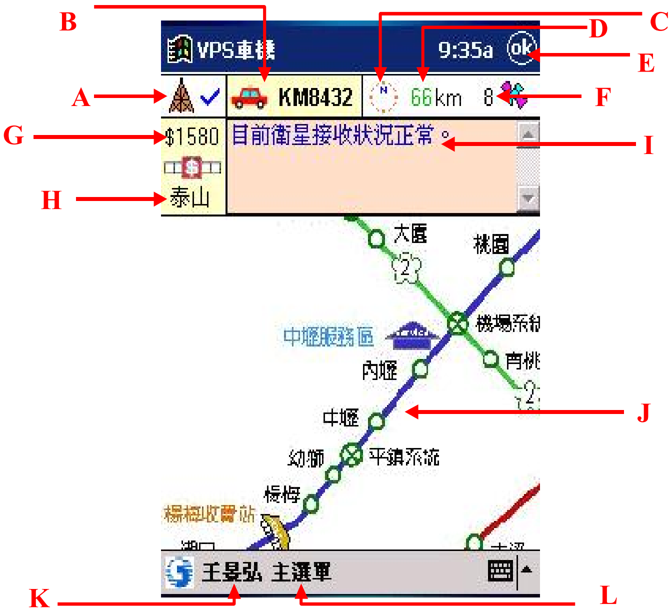


Figure A-3 (a) Smart phone OBU, (b) dedicated type OBU

A smart phone version is implemented as a sample for open platform OBU. Windows Mobile® operation system and Windows Dot Net compact framework is chosen as the VPS client program execution environment. As illustrated in Figure A-4, there are several indicators of user interface listed in Table A-2 to illustrate the main window. For example, item G shows the user’s account balance and item I is the message area which shows the VPS client status or traffic information messages. The implementation of dedicated type OBU consists of hardware design as well as software design. Figure A-3(b) shows the hardware figure of the dedicated type OBU, which is a single chip computer added on with several peripheral modules, such as GPS positioning module, GPRS communication module, etc.

There are three LED signal indicators in this type OBU, which indicate working status, GPS status and GPRS status, respectively.



目前衛星接收狀況正常。

Figure A-4 user interface of the open platform type OBU

Table A-2 Indicators and descriptions of OBU main window

item	Name	Description
A	status	indicate the communication status between OBU and backend system
B	vehicle info.	show the registered vehicle type and its license number
C	Direction	instant direction of the vehicle
D	Speed	instant speed of the vehicle
E	Close	the VPS program exits
F	GPS	GPS status /* the number of received GPS satellites signals */
G	Balance	the current balance of the ETC purse
H	Station	the name of next nearest toll station
I	message area	messages including the VPS status, traffic information, etc.
J	map area	show the map and routing choice near around the vehicle
K	Name	name of registered user for the OBU
L	main menu	advanced functions /* registered profile option, traffic information detail, and debit transaction history, etc. */

A.3.2 VPS Message Protocols

The message protocol plays key position in VPS system. OBU as well as the backend system must follow this protocol to interact each other. There are several messages defined for the VPS-based ETC system, as listed in Table A-3. Each message definition consists of a pair of request and response. The request messages sent from OBU to the backend are named as tailed with ‘***S***’, and the backend response messages are named as tailed with ‘***R***’. For example, ‘***BS***’ message stands for the OBU registration message which is automatically sent to the backend for checking the client registration and OBU version every time when the OBU gets started. The ‘***BS***’ message contains information such as OBU ID, profile, version, and OTA message. The data server in the backend will then verify the user’s profile (balance, vehicle type, etc.), check the version information, change the status and then return ‘***BR***’ message to the OBU. If there is a new version for VPS client program or new definition of the toll zone tables, OTA software update procedure is then activated to update the OBU. A new ‘***BS***’ message is resent to the backend after the update procedure is done and the OBU is restarted. The debit transaction is done by the ‘***TS***’ or the ‘***RS***’ message, where ‘***TS***’ is the online transaction request message sent by OBU to the backend when the OBU detects itself is entering into the tolling zone. A debit transaction is being processing when the backend gets the ‘***TS***’ message, and returns ‘***TR***’ message to the OBU when the transaction is confirmed. Local real time traffic information is appended to the ‘***TR***’ message returns to the OBU to inform the driver local traffic status. If the OBU does not receive the ‘***TR***’ message, the debit transaction is kept in the storage of OBU and resend as ‘***RS***’ (batch transaction) request message after a sleep time period, where a ‘***RS***’ request message may contain one or more debit transactions which are not finished by online transaction '***TR***'. These mechanisms ensure the toll transaction will finally be done whenever the vehicle entering into the tolling zone, even when the mobile network communication is not good in the tolling zone.

Table A-3 Messages protocol definition between OBU and backend in VPS

OBU	Message from OBU to backend	Data Server	Message from backend to OBU
BS	OBU power on msg., registration	**BR**	version checking and user profile verification
CS	profile modification msg.	**CR**	confirm modification msg.
TS	online debit msg. send tracks in toll zone	**TR**	debit confirmation msg., return account balance
SS	shutdown msg.	**SR**	shutdown confirm
RS	batch debit msg.	**RR**	batch debit confirmation msg., return account balance.
QS	traffic info. Request msg.	**QR**	traffic info. reply

A.3.3 Enforcement System

The enforcement system is the key component of VPS-based ETC system because it provides a mechanism to deal with the violation. The design objective of enforcement system is that it should support lane-based free flow as well as multilane free flow enforcement mode, catching the license plate pictures of vehicles, and recognizing the license number in nearly real time. The actions of the enforcement system includes 1) identifying vehicle: detects the vehicles which entering the enforcement line in real-time and classifies the category of the vehicle, 2) taking picture: the camera module must catch the license plate of each vehicle correctly, 3) recognizing the license plate: automatic license plate recognition module should recognize the license plate number of vehicle by OCR, and 4) match pre-processing: the debit transaction data and enforcement information must be matched correctly.

It is a computing critical task for the enforcement system because all the missions should be done in nearly real time (within 1 or 2 seconds), and the computing capacity must accommodate all the vehicles passing through the enforcement line. In order to reduce the computing capacity in license plate recognition and raise the pre-matching accuracy, a heuristic skill is adopted in our implementation. The license plate number is sent to the enforcement system immediately by querying the registered user database in the backend system when it receives '***TS***' message from an OBU. Thus whenever the OBU (vehicle) passes through the enforcement line, the enforcement system has the capability of 'prediction' the next coming registered vehicle license plate. This action improves the efficiency and accuracy of automatic license plate recognition and match pre-processing. The match pre-processing is done when the license number received from the backend is matched to the license plate number recognition, and all the images of unrecognized vehicle is stored in the backend database for the post matching and violation processing.

A.3.4 Backend System

Backend system is a general term for ETC, which consists of several subsystems, such as billing, clearing, customer service, web site, OBU management, system monitoring, violation processing, traffic management supporting, etc. As illustrated in Figure A-5, the major components in the backend system include data server, database, application server and web server. Data server is response for the data sending and receiving to OBU via mobile network and connects to enforcement system via ETC private network. Database keeps the registered user profile, communication logs, transaction records and all the information in the system. Web server provides introduction pages and customer service web pages for the Internet users, sample pages are shown in Figure A-7. Registered users can modify personal profile or list personal transaction history by logining the web site. Application server plays a vital role in the VPS backend system. Several kernel jobs of the backend system including cache response module, debit transaction module, post matching module, violation processing module, etc. are processed in the application server, as illustrated in Figure A-5.

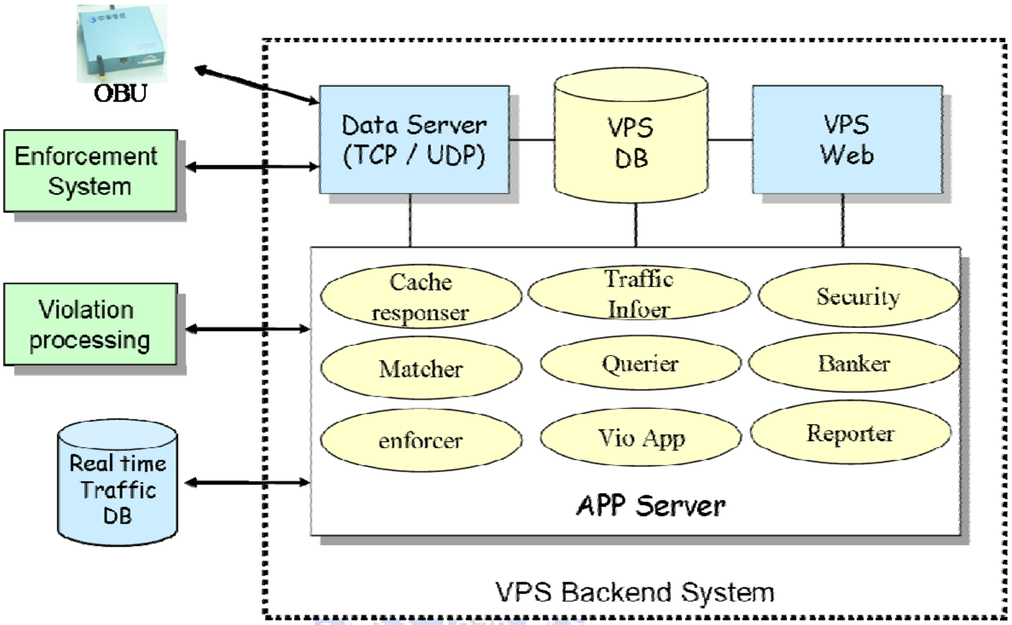


Figure A-5 The servers and modules in the backend system

Debit transaction processing is one of the most important tasks in the backend system. The debit transaction module interacts with OBU to get the debit information (‘***TS***’ or ‘***RS***’ message packet) when OBU entering into the tolling zone. As illustrated in Figure A-6, the data server invokes the *do_TS()* function in the debit transaction module in the application server to complete the debit transaction when it receives the transaction request (***TS***) message from OBU. The debit transaction processes includes checks user account, makes a debit transaction, queries traffic information and returns the ‘***TR***’ message packet which includes transaction result, account balance and local real-time traffic information to the data server. The traffic information includes the estimated journey time in minutes to the next toll station and real time events ahead of the vehicle to notify the user, for example, a passenger car accident ahead two kilometers away. OBU will vocally speak out the message by the text to speech module and also display it on the message area of terminal (smart phone type OBU).

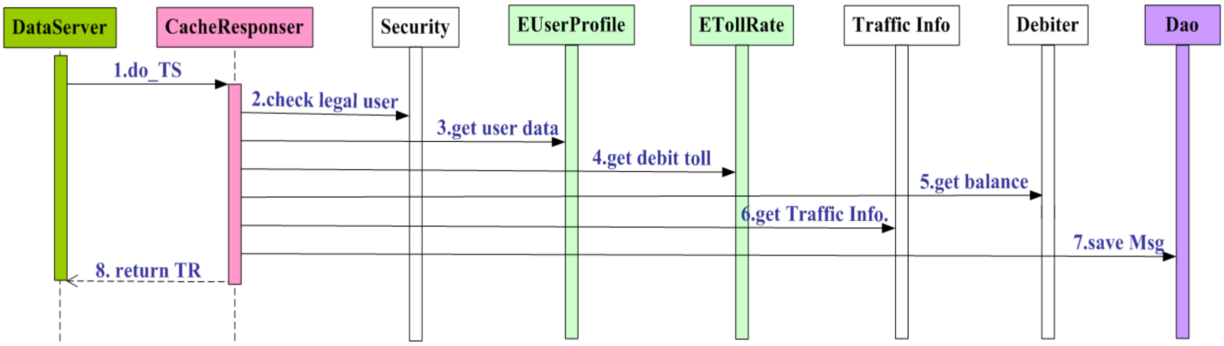


Figure A-6 Debit message (TS) process flow in application server

The backend system includes a web site for the customer service of registered user and the internet guest. It provides many services such as debit transaction history query, traffic information, and estimated traveling time, as illustrated in Figure A-7. The remaining jobs of the backend system, such as matching, billing, clearing, violation processing, will be automatically done as batch jobs in the backend system.

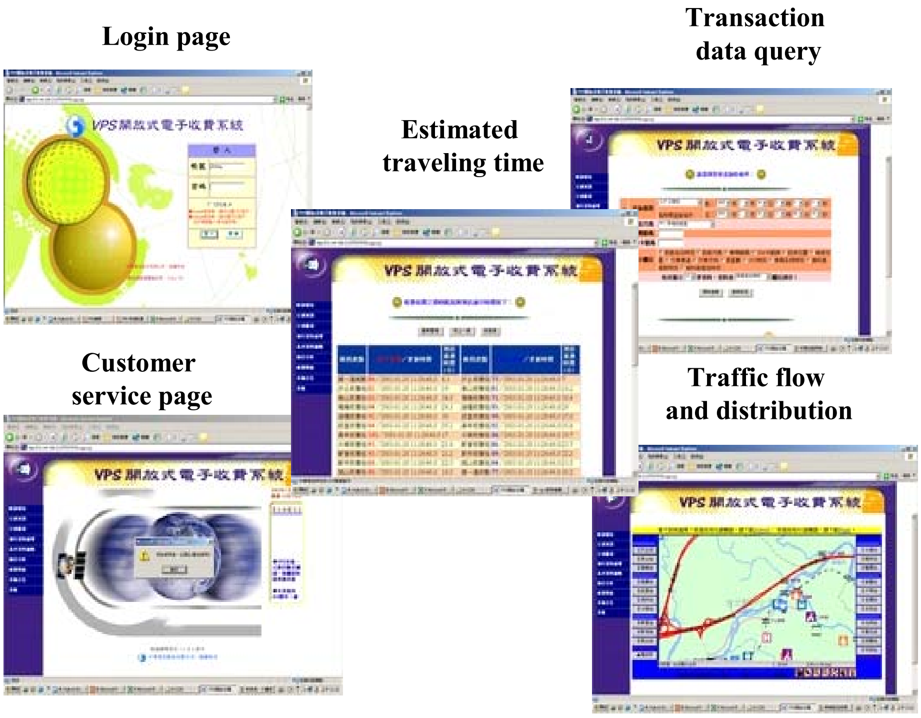


Figure A-7 web site for customer service in VPS

A.5 Unit Test and Field Test

A.5.1 Mobile communication and positioning unit test

Since mobile communication and positioning modules play the key roles in VPS-based ETC system. Unit tests of these two modules are necessary in order to know how well the characteristics of them. The purpose of mobile communication unit test is to summarize the average delay time of GPRS packet from the OBU to the data server. It facilitates to comprehend the time out threshold between the message request and response in OBU. On the other hand, the positioning unit test helps understanding the GPS signal variation and average shift deviation in a fixed location.

The unit test of communication and positioning modules are carried on together at some fixed locations and running unit test program for a period of time. For the communication unit test, in order to calculate the packets lost rate and average packet delay of the adopted GPRS mobile network, the testing program in OBU pings two data servers in the backend by sending three TCP/IP packets via GPRS network, each packet contains 16 bytes data, and there are 10 ping actions in one single test. The time delay of each test is the average delay of all the 30 packets time difference between the GPS satellites timestamp and the system timestamp in the data server. For the time delay calculation accuracy, the data server must be synchronized to universal time by NTP protocol connecting to time server. The GPRS unit test result is shown in Figure A-8(a), where total 12,347 tests are proceeded. The response time threshold is set to 5 seconds, and the packet response time over the threshold is regarded as GPRS packet lost because the packets must return to OBU in the toll zone to confirm the user transaction is done in this time threshold. In the GPRS unit test, the response time over the threshold (lost rate) is 3.64%, and for the packets response within the threshold, the average response time is 3.9 seconds.

The positioning unit test is proceeding in parallel with the communication unit test because the GPS positioning signal and GPS satellites timestamp information are read from GPS module concurrently and sent by the GPRS same packet to the data server. The positioning unit test helps understanding the GPS satellite signal status in a long period of time at fixed location, and gaining the knowledge of average location shifting deviation. The result of positioning unit test is illustrated in Figure A-8(b). For the number of satellite signal over 4 satellites signal then the OBU can read the correct coordinates from GPS module. So the number of satellite signal less than 3 is regarded as positioning failure, which is 3.68 %.

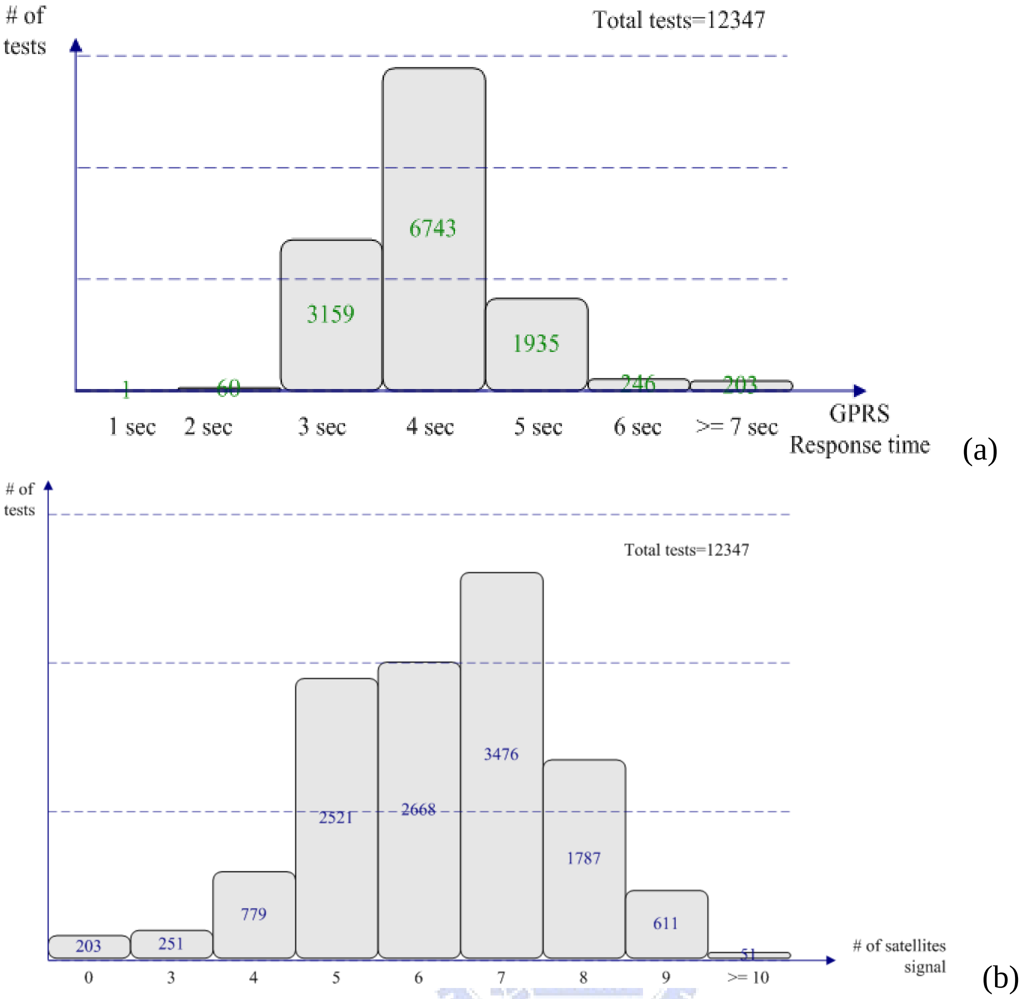


Figure 8 (a) response time of GPRS unit test, (b) satellites signal of GPS unit test

A.5.2 Freeway field test

In order to test the functionality and stability of VPS system, field test was carried out with a 10 vehicles fleet in the national freeway No.1 in Taiwan, and the field tests are done in two parts: first test (2006/12/09 ~ 2007/01/02) and second test (2007/07/13~2007/07/19). In these field tests, only debit transaction system (OBU and backend system) is test, enforcement system is excluded due to complexity and cost issues. Ten dedicated type OBUs (as shown in Figure A-3(b)) were installed on a fleet consists of ten freeway passenger buses which have daily schedule journeys to and from Taipei to Kaohsiung, two large cities located at north and south of the island respectively. There are nine toll stations between these two cities, which all have DSRC-based ETC system [TW07] in operation, and the ten test vehicles equipped with DSRC OBU must drive through the DSRC ETC lane on every scheduled journey. In this freeway field test, the DSRC-based ETC system is the benchmark of this field test. We assume that the data provided by DSRC-based ETC system is 100% accurate. By comparing to DSRC-based ETC, these two phase test results are shown in Table A-4 and Table A-5, respectively.

There are three possible cases when OBU moving through the virtual tolling zone, including 1) transaction request message (***TS***) correctly sent to the backend system and acknowledge (‘***TR***’ message) received by OBU, 2) OBU resend transaction request because OBU does not receive ‘***TR***’ message before timeout, and 3) OBU does not send request or the request packet had lost during mobile network transmission. The first case listed in the column C in Table A-4 and Table A-5 is a normal case, where backend system receives the transaction request (TS), makes a debit transaction, and returns ‘TR’ (transaction confirmed) message to OBU. But in Case (2), listed in column D of Table A-4 and Table A-5, the resent transaction requests as well as original requests are received by the backend system. This may result in duplicated debit transactions, which is not acceptable. Fortunately, this case can be solved by adding a spatiotemporal checking rule in the debit transaction module in the application server. It filters out the duplicate transaction request by checking the recently requests by each OBU. The result of Case (3) is listed in the column E of Table A-4 and Table A-5, the lost packet won’t be collected by the backend and thus results in un-debit case. The accuracy of VPS debit transaction system is calculated by the ratio of correct debit and DSRC-based ETC records, as listed in the last column of Table A-4 and Table A-5. The average accuracy the two field tests are 86.87%, 84.96% respectively. However, the accuracy of two vehicles (No.1 and No.7) are much worse than the others. After recalling the vehicles and checking the OBU, we found that the GPS positioning modules of OBU installed in these two vehicles were malfunctioned. This can be corrected by replacing the positioning module. If the test samples of these two vehicles are excluded, the accuracy of VPS debit transaction system in these two field tests can be achieved to 99.59%, 99.47% respectively.

Table A-4 First field test result of VPS debit system (in the view of vehicle)

Vehicle No.	DSRC ETC Records (A)	VPS ETC Records (B)	VPS Debit Correct (C)	VPS Debit Duplicate (D)	VPS Un-debit (E)	Debit Accuracy (C/A)
(1)	629	38	38	0	591	(6.04%)
2	596	596	596	0	0	100.00%
3	594	594	594	0	0	100.00%
4	597	596	596	0	1	99.83%
5	647	629	629	0	18	97.22%
6	587	587	587	0	0	100.00%
(7)	636	435	435	0	201	(68.40%)
8	631	631	631	0	0	100.00%
9	631	631	631	0	0	100.00%
10	635	634	634	0	1	99.84%
Sum./Avg.	6183	5371	5371	0	812	86.87%
Sum./Avg. (without No.1,No.7)	4918	4898	4898	0	20	99.59%

Table A-5 Second field test result of VPS debit system (in the view of vehicle)

Vehicle No.	DSRC ETC Records (A)	VPS ETC Records (B)	VPS Debit Correct (C)	VPS Debit Duplicate (D)	VPS Un-debit (E)	Debit Accuracy (C/A)
(1)	181	41	41	0	140	(22.65%)
2	133	133	133	0	0	100
3	124	124	124	0	0	100%
4	44	44	44	0	0	100%
5	185	180	180	0	5	97.29%
6	97	97	97	0	0	100%
(7)	117	76	76	0	41	(64.96%)
8	108	108	108	0	0	100%
9	117	117	117	0	0	100%
10	131	131	131	0	0	100%
Sum./Avg.	1237	1051	1051	0	186	84.96%
Sum./Avg. (without No.1,No.7)	939	934	934	0	5	99.47%

It is possible that OBU does not send transaction request message (‘***TS***’) while the toll zone definition is not correct. This is because the coordinates of tolling zones have not been correctly surveyed. In this case, all the vehicles going through the tolling zone will not be debited. Spatial dimension test result analysis (in the view of toll station) for the first field test is shown in Table A-6. The accuracy of all the tolling zones are almost the same and the un-debit records are almost equally spreading to all the toll stations. This indicates that the incorrect toll zone definition case does not exist in this field test, and the un-debit records are mainly caused by two the mal-function OBUs.

Table A-6 Field test result of VPS debit system (in the view of toll station)

Tolling zone ID (Direction)	DSRC ETC Records (A)	VPS ETC Records (B)	VPS Debit Correct (C)	VPS Debit Duplicate (D)	VPS un-debit (E)	Debit Accuracy (C/A)
102(N)	342	293	293	0	49	85.67%
102(S)	342	296	296	0	46	86.55%
103(N)	355	307	307	0	48	86.48%
103(S)	356	313	313	0	43	87.92%
104(N)	359	308	308	0	51	85.79%
104(S)	361	313	313	0	48	86.70%
105(N)	360	311	311	0	49	86.39%
105(S)	361	311	311	0	50	86.15%
106(N)	361	313	313	0	48	86.70%
106(S)	136	119	119	0	17	87.50%
107(N)	361	315	315	0	46	87.26%
107(S)	359	309	309	0	50	86.07%
108(N)	351	310	310	0	41	88.32%
108(S)	349	304	304	0	45	87.11%
109(N)	360	315	315	0	45	87.50%
109(S)	359	311	311	0	48	86.63%
110(N)	353	312	312	0	41	88.39%
110(S)	358	311	311	0	47	86.87%
Sum./Avg.	6183	5371	5371	0	812	86.87%

On the other hand, the online transaction may not success during the short period when the vehicle passing through the tolling zone due to several reasons: delay of mobile network (congestion, handover, weak radio signal, etc), weak GPS signal, or delay of the backend system. The batch transaction message (‘***RS***’) is designed for the backup of the online transaction request message (‘***TS***’). If a ‘***RS***’ message is received by the backend system, it implies a ‘***TS***’ transaction request does not get a ‘***TR***’ confirmation during the timeout threshold period when the vehicle moving through the tolling zone. The ratio of the ‘***TS***’ message compares to the total transactions record (‘***TS***’+’***RS***’) received by the backend system indicates the successful rate of online transaction request, which can be further analyzed in two dimensions: OBU and tolling zone. In the OBU point of view, if the successful rate of one OBU is lower than the average, then the communication module of that OBU may have problems. As shown in Table A-7(a), the summary of 97.65% online transaction successful rate. Besides, there is no big difference between each OBU and average online transaction successful rate, which indicates the communication module of these OBUs are all in good working status. In the tolling zone point of view, lower successful rate in tolling zone indicates the tolling zone may be too small to complete the online transaction, which is important information for re-defining the coordination of the zone. Table A-7(b) shows the statistics of each tolling zone are near around the average value except the toll station ID *109(N)*, which indicates the zone area of *109(N)* should be revised to raise the successful rate.

Table A-7 Online debit transaction successful rate

(a) By the OBU point of view

Vehicle No.	TS (A)	RS (B)	transaction successful rate A/(A+B)
1	894	24	97.39%
2	5337	88	98.38%
3	5466	97	98.26%
4	4316	106	97.60%
5	5249	132	97.55%
6	5654	136	97.65%
7	1145	29	97.53%
8	4846	118	97.62%
9	5633	140	97.57%
10	5440	189	96.64%
Sum./Avg.	43980	1059	97.65%

Table A-7 (b) By the tolling zone point of view

Tolling Zone ID (Direction)	**TS (A)**	**RS (B)**	**transaction successful rate A/(A+B)**
102(N)	2287	39	98.32%
102(S)	2265	34	98.52%
103(N)	2444	34	98.63%
103(S)	2448	30	98.79%
104(N)	2482	41	98.37%
104(S)	2456	62	97.54%
105(N)	2493	47	98.15%
105(S)	2466	57	97.74%
106(N)	2713	73	97.38%
106(S)	2699	69	97.51%
107(N)	2746	41	98.53%
107(S)	2697	57	97.93%
108(N)	2306	44	98.13%
108(S)	2333	34	98.56%
109(N)	**2164**	**245**	**89.83%**
109(S)	2284	56	97.61%
110(N)	2356	36	98.49%
110(S)	2312	60	97.47%
Sum./Avg.	43951	1059	97.65%

A.6 Concluding Remarks

VPS is an evolutional technology for area wide integrated road charging solution, it achieves the goal of electronic payment and electronic toll collection by a totally different scheme comparing to traditional DSRC-based ETC technology. It is cost effective and has the advantages of simplicity and flexibility comparing to DSRC-based ETC, and the virtual toll zone can be changed easily by modifying the toll zone coordinates table. This feature provides a way to migrate from the lane-based toll collection scheme to the distance-based or area wide toll collection scheme, which is fit in with the policy of toll collection scheme in many countries. Besides, OBU is easy to integrate with telematics service because they have the same key components: positioning, mobile communication, computing, and human interface. The transaction capability also enables the mobile electronic commerce. In this work, we discussed the design and implementation details of VPS-based ETC system, and unit tests as well as a debit transaction field test had been practiced in the freeway of Taiwan. The results show that the accuracy of debit transaction is satisfiable if the malfunction OBU data is excluded. In addition to the debit transaction, enforcement system and the matching mechanism for enforcement and debit transaction are also playing a key position in the VPS system, which will be the major issues of our future work.