國立交通大學

資訊科學與工程研究所

碩 士 論 文

利用 IEEE 802.11p/1609 技術來縮短緊急車輛
到達事故地點的時間

Employing the IEEE 802.11p/1609 Technology to Shorten
the Arrival Times of Emergency Vehicles to Accidents

研 究 生：徐銘昇

指導教授：王協源　教授

中 華 民 國 九 十 八 年 六 月

利用 IEEE 802.11p/1609 技術來縮短緊急車輛到達事故地點的時間

Employing the IEEE 802.11p/1609 Technology to Shorten
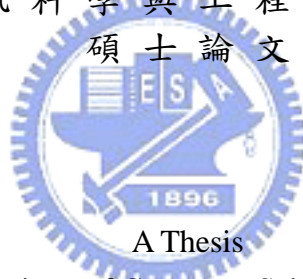the Arrival Times of Emergency Vehicles to Accidents

研 究 生：徐銘昇　　　　　Student：Ming-Sheng Hsu

指導教授：王協源　　　　　Advisor：Shie-Yuan Wang

國 立 交 通 大 學
資 訊 科 學 與 工 程 研 究 所
碩 士 論 文

A Thesis
Submitted to Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Computer Science

June 2009

Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 八 年 六 月

利用IEEE 802.11p/1609技術來縮短緊急車輛到達事故地點的時間

Employing the IEEE 802.11p/1609 Technology to Shorten
the Arrival Times of Emergency Vehicles to Accidents

研 究 生：徐銘昇　　　　　指導教授：王協源

國立交通大學

資訊科學與工程研究所

摘　要

　　事故救援對拯救生命和緊急救護是一個重要的論題。事故救援中的一項議題就是如何縮短緊急車輛（例如救護車、消防車、警車）到達事故現場的時間使得救護作業可以順利的進行。舉例來說，救護車應盡快地到達車禍現場，如此一來傷患才能盡早地被照顧到。然而，在大都市的區域裡，塞車可能使得緊急車輛難以快速地到達事故現場。這是因為在道路上的車輛有時會難以讓出道路讓緊急車輛經過。在這個情況下，由於無法預期的塞車，拯救傷患或消滅火災的寶貴時間將會被浪費掉而造成無法挽回的悲劇。

　　為了解決上述的問題，在此篇論文中，我們提出一種基於 IEEE 802.11p/1609 無線技術的交通控管機制來控制車流狀況和交通號誌來縮短緊急車輛到達事故現場的時間。在所提出的機制裡，行控中心用來監控道路上的車流情況並提供交通資訊給緊急車輛。行控中心會動態地改變道路上的交通號誌來幫助緊急車輛能快速地到達事故現場。

　　我們的模擬結果顯示，在我們所研究的案例中，我們所提出的機制能夠將緊急車輛到達事故現場的時間大幅地縮短 30% 至 50%。

　　關鍵字：網路模擬器、IEEE 802.11p/1609、緊急車輛。

Employing the IEEE 802.11p/1609 Technology to Shorten
the Arrival Times of Emergency Vehicles to Accidents

Student：Ming-Sheng Hsu   Advisor：Shie-Yuan Wang

Institute of Computer Science and Engineering
National Chiao Tung University

ABSTRACT

Accident rescue is an important topic for life saving and emergent handling. One issue of accident rescue is how to shorten the times for emergency vehicles (such as ambulances, fire engines, and police cars) to arrive at accident scenes so that the emergent events can be properly processed. For example, an ambulance should arrive at a traffic accident scene as fast as it can so that the wounded can be taken care of as soon as possible. However, in metropolitan areas traffic jams may make emergency vehicles difficult to arrive at the accident scenes. This is because cars on the roads may not have enough space to move themselves to empty a lane for the pass-through of emergency vehicles. In this condition, the valuable time for rescuing injured people or extinguishing fire will be wasted due to undesired traffic jams, which may result in unrecoverable tragedies.

To solve this problem, in this thesis we propose a traffic control mechanism based on the IEEE 802.11p/1609 radio technology to control the traffic condition and traffic lights to shorten the time required by emergency vehicles to arrive at accident scenes. In the proposed mechanism, a traffic control center is deployed to monitor the traffic conditions on the roads and provide the traffic information for emergency vehicles. The traffic control center can dynamically adjust the traffic lights on the roads to help emergency vehicles arrive at accident scenes more quickly.

Our simulation results show that the proposed mechanism can significantly reduce 30% to 50% arrival times required by emergency vehicles to arrive at accident scenes in our evaluated cases.

Keywords: network simulator, IEEE 802.11p/1609, emergency vehicle.

誌謝辭

首先必須感謝恩師王協源教授多年來對我的指導與照顧,從大學的專題開始,我就在王教授的實驗室下做研究,在專業領域上,經過這幾年的扎實訓練,讓我在研究與實作上均獲得了非常大的成長;而在立身處事方面,老師亦教予我們許多待人處事的道理。

感謝口試委員陳裕賢教授、鄭振牟教授以及藍崑展教授特地撥冗前來交通大學,聽取我們的論文報告並加以指導,在他們的建議下,修正不足之處,讓這篇論文得以更加完善。

感謝周智良與林志哲兩位學長,在平常的討論中讓我的論文架構更加的完善,也感謝網路與系統實驗室的每一位成員,與你們的相處與互相合作,讓我了解到團隊合作的方法與溝通技巧,也充實了我的研究所生活。

最後,感謝父母親對我的栽培和哥哥姐姐平時對我的關懷與照顧,有了你們的包容和支持,我才能順利的完成碩士學位。

Contents

List of Figures

List of Tables

Chapter 1. Introduction

Recently the advances of telecommunication systems and chip manufactures enable smart automobiles equipped with a variety of radios. For example, the Global Position System (GPS) has been widely deployed in vehicles for travel navigation and GPRS/3G cellular communication has be deployed in the buses of several public transportation system for tracking their locations. One promising new communication standard for such vehicular network is the IEEE 802.11p/1609 specification family, which is proposed for wireless access in vehicular environments based on DSRC (Dedicated Short Rang Communications) communication protocol. It aims to achieve the communications between vehicles and vehicles or vehicles and roadside units. The IEEE 802.11p/1609 network can be used to convey messages for applications for road safety, traffic condition report, traffic control, etc.

1.1. Motivation
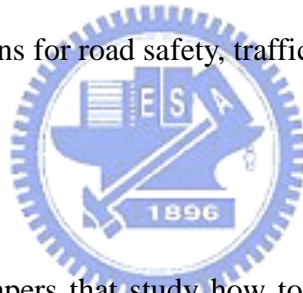
There have been many papers that study how to minimize the preparing time of the medicine for patient. However, rare papers consider the time required by emergency vehicles for arriving at accident scenes. Reducing the arrival times of emergency vehicles to accident scenes allows emergency events to be properly processed before they cause more damage. For example, if an ambulance arrives at a traffic accident scene in a very short time, injured people can be taken care of before their body conditions get worse. Another example is that if a fire engine arrives at the fire scene in a very short time, the fire can be eliminated before it causes more damage.

To achieve this goal, we propose a traffic control mechanism based on the IEEE 802.11p/1609 radios technology. Because the IEEE 802.11p/1609 specification is still under development, it is difficult to evaluate the performances of our proposed mechanism with

real 802.11p radios in the real world. Thus, in this thesis we use the simulation approach to evaluate the performances of the proposed mechanism.

1.2. Problem Description

In this thesis, a traffic control mechanism based on IEEE 802.11p/1609 is proposed. Each intersection is installed with a RSU (Road-Side Unit) and each vehicle is equipped with an OBU (On-Board Unit). The traffic control center monitors the traffic condition and controls the traffic light signal.

Note that, although an emergency vehicle can move itself to the nearby lane, frequently altering its moving lane is dangerous for itself and may cause traffic accidents. Especially, emergency vehicles such as ambulances, fire engines, and police cars may move at a high speed to save time to arrive at accident scenes. Therefore, evacuating vehicles for emergency vehicles on their way to the accident scenes not only saves the rescue time of injured people but also reduces the occurrences of accidents for emergency vehicles when they are moving to accident scenes.

The proposed traffic control mechanism uses three significant functions which are (1) Moving Path Arrangement, (2) Evacuation Message Delivery, and (3) Dynamic Traffic Lights Control. For moving path arrangement, the traffic control center will generate a shortest-time path to the accident scene for emergency vehicle to pass through so that the emergency vehicle will not waste too much time on choosing moving path. For evacuation message delivery, the traffic control center will send the evacuation message to a relevant RSU based on receiving the evacuation request sent from an emergency vehicle. The RSU will broadcast the evacuation message to nearby vehicles so that they can move themselves to other lanes and empty a lane in advance for facilitating emergency vehicles' passing through. For dynamic traffic lights control, the traffic control center can dynamically switch

the traffic lights into green to avoid the obstruction in the intersection so that the emergency vehicle will not be jammed when it approaches the intersection.

There are three interesting issues we will discuss in this thesis which are listed as follows.

(1) How will it affect the performance if we use different function combinations for the proposed traffic control mechanism: As described previously, there are three functions that can be used in the traffic control mechanism which are (1) moving path arrangement, (2) evacuation message delivery, and (3) dynamic traffic lights control. How will it affect the performance if we use different function combinations for our traffic control mechanism?

(2) How will it affect the performance of the proposed mechanism if we deploy vehicles on the road map with different vehicle separation distances: The average vehicle separation distance (i.e., vehicle density) is different during day and night. Therefore, it is interesting for us to know how it affects the performance if we deploy vehicles on the road map with different vehicle separation distances.

(3) How will it affect the performance under different traffic lights periods: In the real world, traffic lights on a main lane may have longer green time than the less important lane. Due to this reason, we also interest in the effect of the performance when applying different traffic lights periods for each scenario.

1.3. Organization

The rest of this thesis is organized as follows. A brief overview of IEEE 802.11p/1609, A* algorithm, related work, and the NCTUns network simulator is described in Chapter 2. In Chapter 3, the proposed design and implementation of the simulation environment is introduced. Using the module-based node location update and traffic signal control

mechanism makes the NCTUns network simulator simulate much more nodes on the roadmap. In Chapter 4, the design and implementation of the proposed traffic control mechanism is depicted. The results of the proposed traffic control mechanism applying distinct schemes for each scenario are shown in Chapter 5. In Chapter 6, the future work is discussed. Finally, in Chapter 7, the conclusion of this thesis is described.

Chapter 2. Background

This chapter mainly describes the concept of IEEE 802.11p/1609. At first the architecture of WAVE system is introduced. Then the operation of WAVE system is depicted. Finally, the A* algorithm, ESRI shapefile, NCTUns network simulator, and related work will be described in the rest of this chapter.

2.1. IEEE 802.11p/1609

IEEE 802.11p draft standard [4] is an amendment of the IEEE 802.11-2007 standard [5]. It adds wireless access in vehicular environments (WAVE) and defines the enhancements to the 802.11 standard that is required to support Intelligent Transportation Systems (ITS) applications. These enhancements include data exchange between vehicles with high-speed movement and between the vehicles and the roadside unit in the licensed ITS band of 5.9 GHz (5.85-5.925 GHz).

IEEE 802.11p standard is designed to collaborate with the IEEE 1609 trial-use standard suite. The required components of WAVE system were specified in IEEE 1609.1—Resource Manager [6], IEEE 1609.2—Security Services [7], IEEE 1609.3— Networking Services [8], and IEEE 1609.4—Multi-Channel Operations [9]. The architecture of the IEEE 802.11p/1609 network is depicted in Fig. 2.1. The new network architecture supports the TCP/UDP/IPv6 protocol suite and a new WAVE short message protocol (WSMP). The former is used to cooperate with existing IP-based network applications and the latter is used to disseminate small-sized packets that carry emergent safety event, location service, or traffic information, etc.

In the WAVE system, there are two device types which are On-Board Unit (OBU) and Road-Side Unit (RSU). OBU is a WAVE device that can operate when in motion and supports the information exchange with RSUs or other OBUs. RSU is a WAVE device that operates only on the stationary and supports information exchange with OBUs.
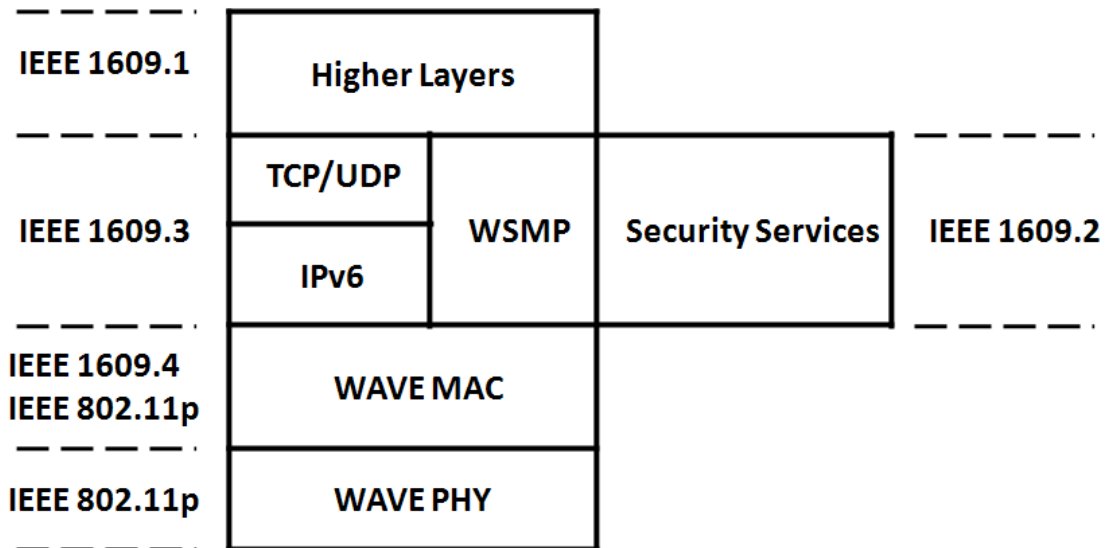
Fig. 2.1   WAVE System Protocol Stack

The WAVE system divides the link channel into two classes of radio channel: a control channel (CCH), and multiple service channels (SCH). As one can see in Fig. 2.2, the CCH and SCH are presented by black blocks and gray blocks, respectively. Each channel interval is 50 milliseconds suggested in IEEE 1609.4 draft standard [9]. The CCH is reserved for system control messages or WAVE short message (WSM). The SCHs are used for transmitting WSM or general purpose application data packets via a WAVE Basic Service Set (WBSS). The WAVE devices need to monitor CCH periodically to acquire system control messages. The operation of the WAVE system is explained below.
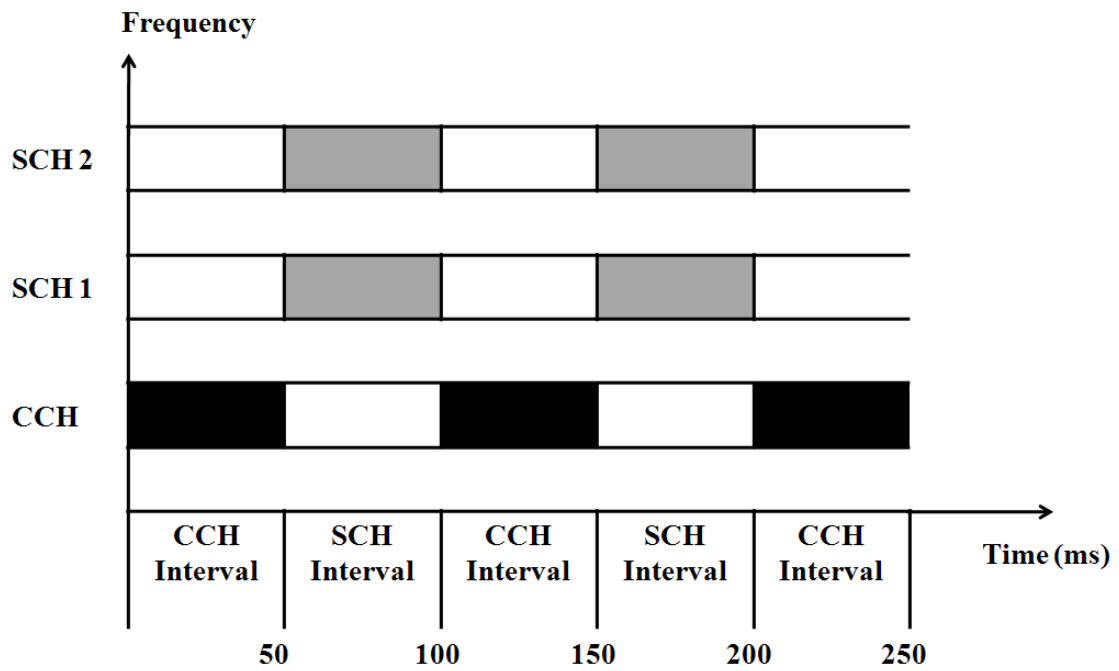
Fig. 2.2 Channel Intervals

In the WAVE system, there are two methods, or protocol stacks, to provide communication services for applications: WSMP and IPv6. WSMP is unique to IEEE 1609 standards and is expected to be used for special purpose. Applications using WSMP can initiate a WBSS to make a SCH available for their special use, but this may be unnecessary since WAVE short messages (WSMs) can be exchanged on the CCH without initiating a WBSS. As one can see in Fig. 2.3, the WSM can be transmitted from OBU to OBU and OBU to RSU.
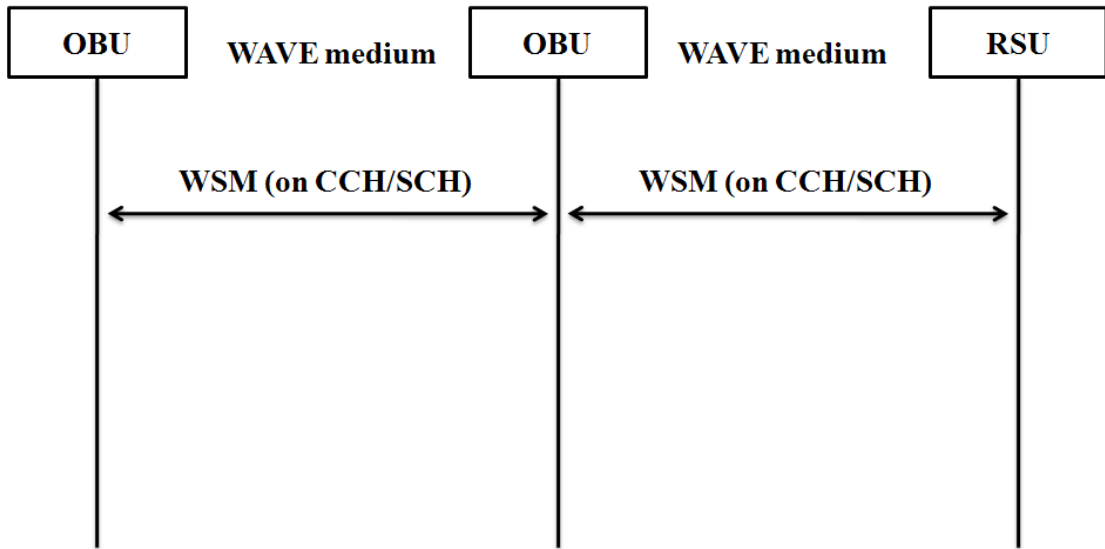
Fig. 2.3   WSM Transmission

The general application data packets can be transmitted via a WBSS using IPv6. The WAVE device that transmits the WAVE service advertisement (WSA) to initiate a WBSS is called the provider. A WAVE device is called the user if it participates in the WBSS by accepting the WSA.
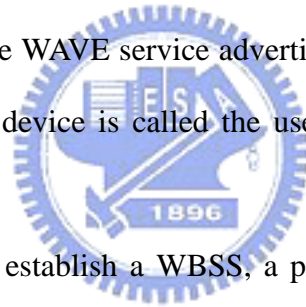
As shown in Fig. 2.4, to establish a WBSS, a provider needs to announce a WAVE service advertisement (WSA) on the CCH during CCH interval. The WSA contains the operational information of the WBSS such as the ID of this WBSS and the SCH number for data transmission. A WBSS can be non-persistent or persistent. The provider can initiate a non-persistent WBSS by announcing the WSA only on one CCH interval. If the provider wants to initiate a persistent WBSS, it has to announce the WSA on every CCH interval.
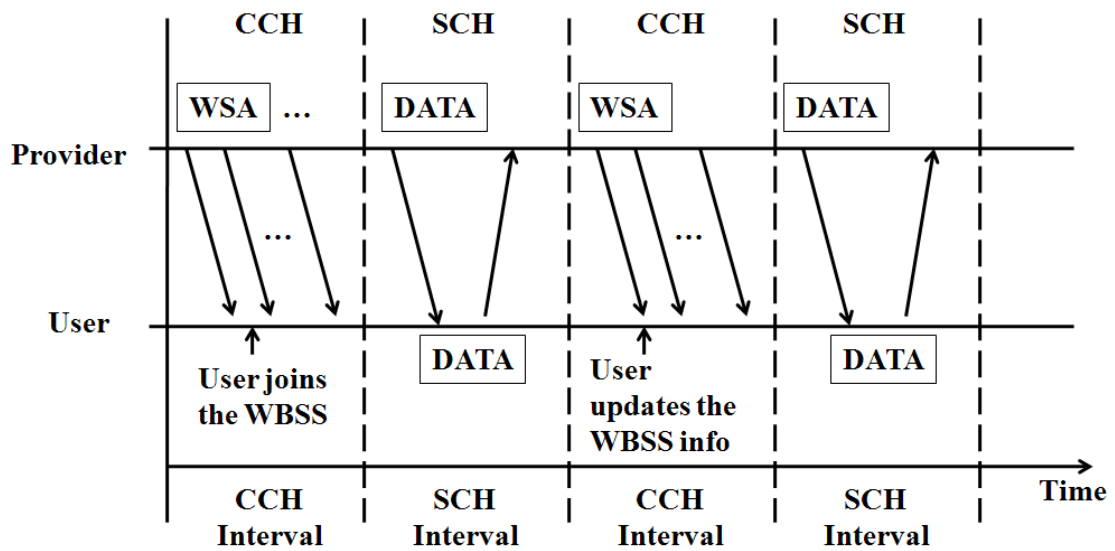
Fig. 2.4   Operation with a Persistent WBSS

2.2.   A* Algorithm

In this thesis, the traffic control center will generate the shortest-time path for the emergency vehicle to pass through. In order to generate the shortest-time path, A* algorithm is adopted in the traffic control center.

The A* algorithm [10] is a best-first, graph search algorithm that can find the path with the minimum cost from the source to the destination. This algorithm searches nodes from source to destination. For each node x, it contains 3 values:

$f(x)$: the heuristic function to determine which node has the highest priority to be searched, where $f(x) = g(x) + h(x)$

$g(x)$: the actual path-cost function from source to node x

$h(x)$: the admissible heuristic estimate of the distance from node n to destination

The algorithm maintains a priority queue called the "open set" which contains the nodes to be searched. The node with the smallest f value in the open set will be searched first. At each iteration of this algorithm, the node with the smallest f value is removed from

9.

the open set, the f and h values of its neighbors will be updated, and these neighbors will be added into the open set. This algorithm terminates when the destination node with the smallest f value is removed from the open set or the open set is empty. The pseudo code of A* algorithm is shown below.

```
A* algorithm pseudo code:
    Function InitA*(RoadMap)
        for all node x in RoadMap do
            g_value[x] := infinity
            f_value[x] := infinity
            frontNode[x] := undefined
        end for
    Function A*(RoadMap, Source, Destination)
        InitA*(RoadMap)
        add Source into OpenSet
        g_value[Source] := 0
        h_value[Source] := heuristic_distance(Source, Destination)
        f_value[Source] := g_value[Source] + h_value[Source]
        while OpenSet is not empty
            x := node in OpenSet with the smallest f_value
            if x = Destination
                return generateShortestTimePath(frontNode, Destination)
            remove x from OpenSet
            for each neighbor y of x
                trial_g_value := g_value[x] + edge_length (x, y)
                h_value[y] := heuristic_distance(y, Destination)
                if trial_g_value < g_value[y]
                    add y into OpenSet
                    frontNode [y] := x
                    g_value[y] := trial_g_value
                    f_value[y] := g_value[y] + h_value[y]
        return fail
    Function generateShortestTimePath(frontNode, currentNode)
        if frontNode[currentNode] is undefined
                return
        else
            generateShortestTimePath(frontNode, frontNode[currentNode])
            add currentNode into ShortestTimePath
```

heuristic_distance(x, y): the linear distance between x and y

edge_length(x, y): the actual edge length between x and y in the road map

2.3.   ESRI Shapefile

The GUI program of NCTUns can load the ESRI (Economic and Social Research Institute) shapefile. GUI will transform intersections and streets in the road map to nodes and edges and then save road map information into road map configuration file for later use.

The ESRI shapefile or normally called the shapefile is a common used geospatial data format for geographic information systems software. It is developed by ESRI as an open specification for data interoperability in ESRI and other software products. The "shapefile" commonly refers to a series of files with ".shp", ".shx", ".dbf" extensions. The actual shapefile relates specifically to files with the ".shp" extension. However, this file alone is incomplete for distribution. The other supporting files are also required.

Shapefiles spatially describes geometries such as points, polylines, and polygons. For instance, these could represent intersections, roads, loops, etc. Each item may also have attributes to describe items such as the road name or width.
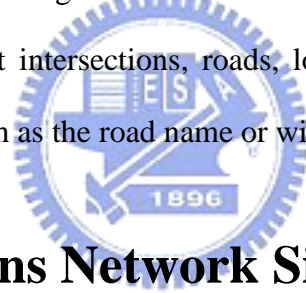
2.4.   The NCTUns Network Simulator

NCTUns network simulator [1] is an integrated simulation platform for vehicular traffic, communication, and network researches. It combines its original network simulation capabilities with traffic simulation capabilities, such as roadmap construction, vehicle driving behavior model, etc. The current released version of NCTUns is 5.0 [2]. It aids the ability to simulate IEEE 802.11p/1609 network protocol for vehicular wireless network, including the establishment of WBSS and transmission of WSM.

2.4.1. Application-based Node Location Update and Traffic Signal Control

Fig. 2.5 shows the architecture of application-based agent logic control over the current released NCTUns 5.0 network simulator [3]. The architecture consists of five major components, which are (1) GUI, (2) SE, (3) Command Server, (4) Car Agent(s), and (5) Signal Agent(s). The roles of these components and their functions are described below.

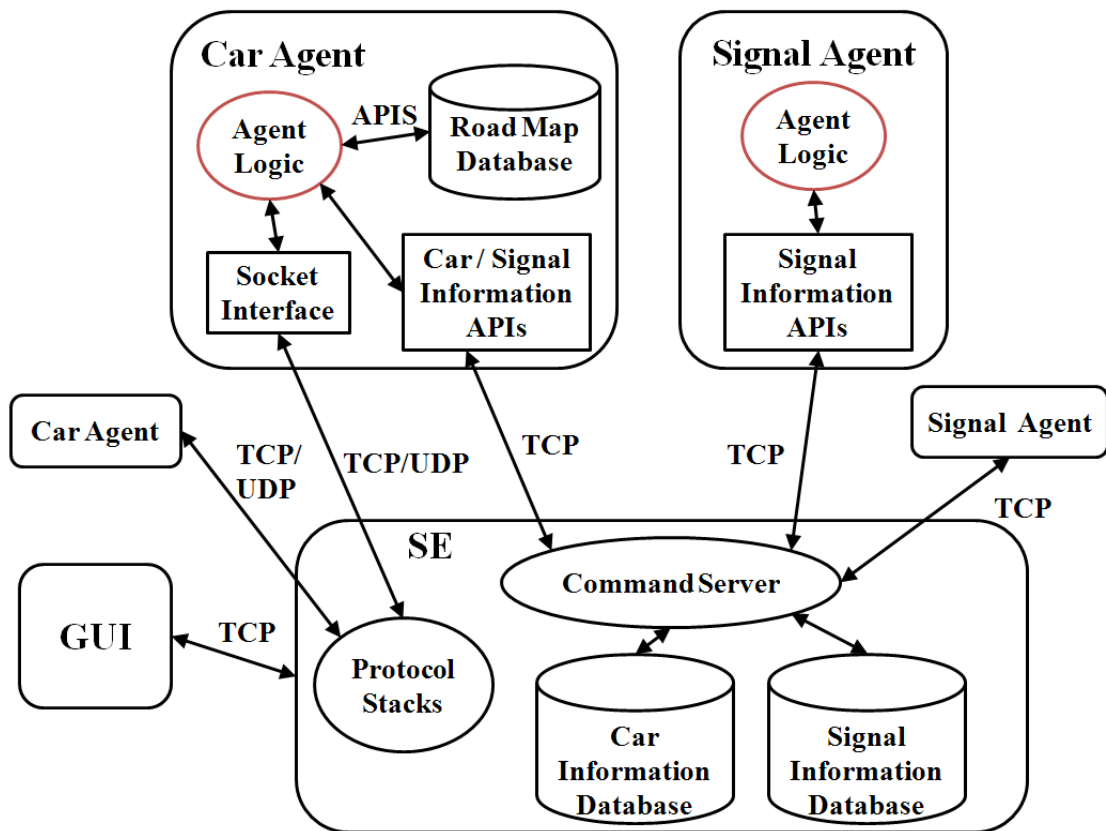Fig. 2.5　Architecture of Application-based Agent Logic Control

(1) GUI (Graphic User Interface)

The GUI process provides users an environment where they can easily construct their own road networks. For instance, road segment construction and vehicle deployment can be easily specified in a few steps of mouse operation. Users can construct the road map by themselves or by loading the shapefile using GUI.

Moreover, users can also select/replace network protocols and modify the network parameters in a few operations. After users finish all the settings of the road networks, the GUI program will generate necessary configuration files for later use.

With the help of the GUI program, users can save much time in constructing a network topology with many vehicles and roads. The GUI also has the ability to play back simulation results of vehicle movement and packet transmission. This play back ability can greatly help users verify the correctness of their design of vehicle driving behavior and network protocol.

(2) SE (Simulation Engine)

The main function of SE process is simulating MAC (Media Access Control) layer and PHY (Physical) layer protocols. For example, SE can simulate the MAC layer of IEEE 802.11p/1609 network protocol, the propagation delay and the power loss at the physical layer. In vehicular networks, SE stores car and signal information for servicing the requests issued from car and signal agents. Both car agent and signal agent are child processes of SE.

(3) Command Server

Command server is implemented inside the SE. Command server is responsible for accepting and responding to the requests from applications. Applications, such as car agent and signal agent, can update or retrieve the car/signal information from car/signal information database through the help of command server. For example, car agent can send a request to the command server to get locations of its surrounding vehicles. When command server receives the request, it will access the car/signal information database to get the requested surrounding vehicles location and then sends back the data to car agent.

(4) Car Agent

A car agent is a process run on each vehicle node. There are four important components in the car agent, which are (1) the car agent logic, (2) a road map database, (3) socket interfaces, and (4) car/signal information APIs. The car agent logic controls the automatic driving behavior of the vehicle node on which the car agent is run. When the simulation begins, the car agent will dynamically update the vehicle nodes location. As described before, the GUI program will generate a road map configuration file either by user defined or by loading the road map with shapefile format. Car agent will load the road map configuration file into its road map database. The road map database not only stores the location/direction of roads but also the relationship between each street and intersection. For example, we can know which street connects two specific intersections from the road map database. The socket interfaces provide TCP/UDP Internet connections for vehicles to exchange their information on the road. The car/signal information APIs are the functions that the car agent logic can utilize to access the car and signal information databases located in the SE. These car/signal information API functions internally use TCP/IP IPC (Inter-Process Communication) connections between the car agent and the command server. For instance, car agent can use the car/signal information API function to send a request to command server. While command server receives this request, it will access/update the car and signal information database and send back the data that car agent required.

(5) Signal Agent

A signal agent is a process run up for each traffic light at each intersection. It is responsible for controlling the signal state of the four traffic lights located at the intersection. Signal agent has two main components, which are (1) the signal agent

logic and (2) the signal information APIs. The signal agent logic governs when the signal state should be updated and what color of traffic lights to be changed into. The signal information APIs are called by the signal agent to update the signal information database stored in the SE. These signal information API functions internally use TCP/IP IPC (Inter-Process Communication) connections between the signal agent and the command server.

During simulation, the statuses of all traffic lights on the roadmap are changed on the same time and the time period of green and red lights are the same by default.

2.4.2.   Application-based WAVE Short Message Transmission

WSM (WAVE Short Message) can be implemented inside the car and signal agent. As such, car and signal agent can use WSM to transmit small sized information to each other. Fig. 2.6 shows how network protocol stacks and the transmission of WSM are simulated on NCTUns. In the example, two vehicles move on the road and they exchange WSM packets with each other. To simulate this case, two car agents are run up for each vehicle and control these two vehicles' moving behaviors. In addition, SE is run up to simulate the WME (WAVE Management Entity), MAC (Media Access Control) layer, and PHY (Physical) layer of IEEE 802.11p/1609 protocol stacks. Suppose that the left car agent sends a WSM packet to the right car agent, the detailed procedure of WSM packet delivery is described below.
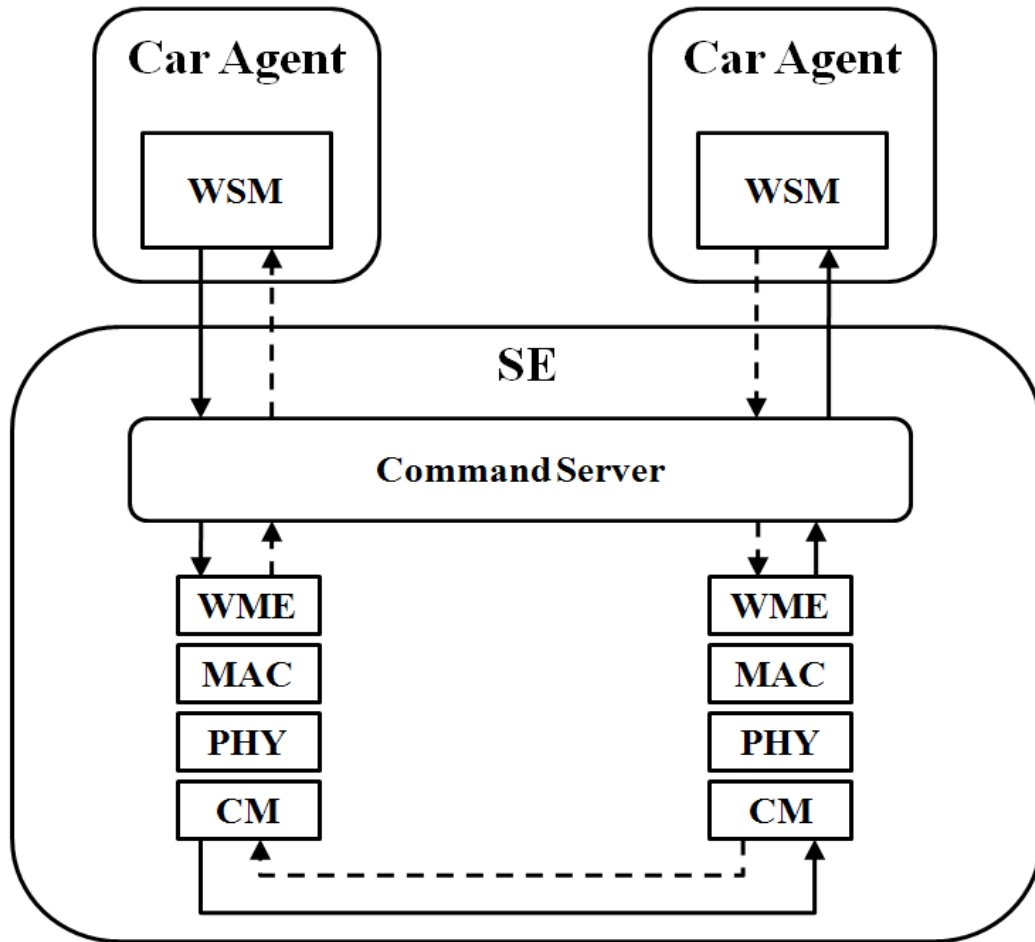
Fig. 2.6 Architecture of Application-based WSM Simulation

1. The car agent logic of the sending car encapsulates its data as a WSM packet and uses IPC command to send out this WSM data packet to the command server inside the SE.

2. When command server receives this IPC command from car agent, it will decapsulate this IPC command to retrieve the WSM packet and then send this WSM packet to the sending WME to start the procedure of WSM packet transmission.

3. The WME protocol, MAC protocol, and PHY protocol are simulated in SE. When this WSM packet arrives at the sending WME, this WSM packet will be encapsulated with a MAC header and then sent from the sending PHY to the

17.

receiving PHY. The channel model (CM) module is used to simulate the power loss of the packet on the air.

4. The MAC header of the WSM packet will be stripped off when this packet arrives at the receiving MAC. When this WSM packet arrives at the receiving WME module, it will be sent to the command server.

5. The command server will send out this WSM packet using IPC command to the receiving car agent.

6. Finally, the agent logic of the receiving car agent retrieves the WSM packet from the IPC command sent from the command server.

2.5. Related Work

In [11], the authors proposed a Fast Emergency Preemption Systems (FAST). The system can detect the approach of the emergency vehicles to the intersection and change the traffic light for emergency vehicles. However, the system doesn't provide the ability to pre-evacuate vehicles on the shortest-time path for emergency vehicles to pass through.

In [12], the authors used a centralized control mechanism to control the traffic condition for emergency vehicles to pass by. However, the mechanism doesn't provide the control of the traffic lights. Therefore, when the signal is red, there is still a great opportunity that the emergency vehicle may be blocked in the traffic jam. In the real world, if the emergency vehicles are close to the traffic lights, they can directly pass the intersection without concerning the traffic light. However, it is very dangerous and may cause another traffic accident.

In [13], the authors propose a mechanism that can reduce the traffic chaos by evacuating vehicles. However, it doesn't have the ability to control the traffic lights. In [14], the authors propose a learning real-time routing system for emergency vehicles. This system

can control the traffic light to make the emergency vehicles arrive at the accident fast and safely. However, it doesn't have the ability to make surrounding vehicles make the way for emergency vehicles to pass through. If the emergency vehicles switch lane in high speed frequently, it is very dangerous and may cause another traffic accident.

Chapter 3. Design and Implementation of the Simulation Environment

NCTUns network simulator can be used to simulate the vehicular networks. In this thesis, we want to simulate a large-scale vehicular network and therefore we need to deploy a lot of cars and traffic lights on the road map. However, as described in Chapter 2.4, the current released version of NCTUns 5.0 uses application-based agent logic control. Each car occupies one car agent process and each group of traffic lights at one intersection occupies one signal agent process. Therefore, SE needs to create a lot of child processes to simulate a vehicular network with a large amount of vehicles and traffic lights. It will cost a lot of waiting time for IPC communication and context switch between processes and thus lead to low performance of the simulation. Therefore, we need to modify the architecture of NCTUns's agent logic control mechanism to enhance its simulation performance.

3.1. Module-based Node Location Update and Traffic Signal Control

Fig. 3.1 shows the architecture of the module-based node location update and traffic signal logic control platform. The architecture consists of eight elements, which are (1) GUI, (2) SE, (3) Node Module, (4) Car Agent Logic, (5) Signal Agent Logic, (6) Car Information Database, (7) Signal Information Database, and (8) Road Map Database. The roles of these elements and their functions are described below.
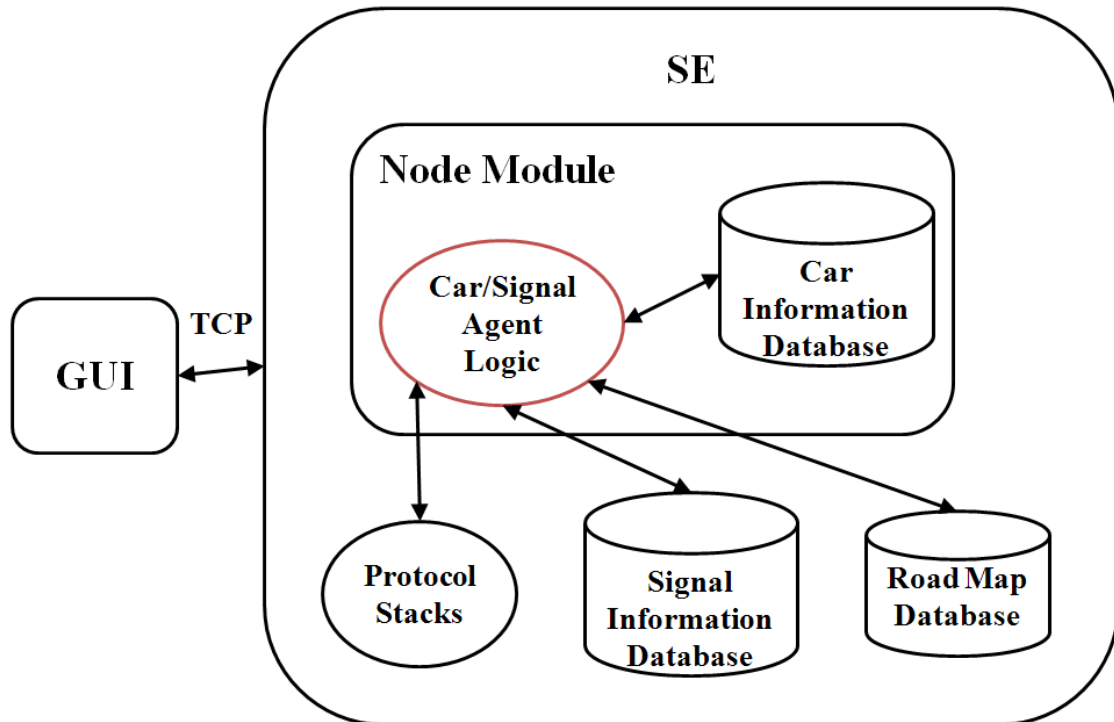
Fig. 3.1   Architecture of Module-based Agent Logic Control

(1) GUI (Graphic User Interface)

The GUI program provides users an convenient platform to construct a road network. Users can decide whether vehicles should be run by the application-based logic control or module-based logic control with a few operations. For example, if users want all vehicles are driven by the module-based logic control, they can deploy the car with the module-based logic control icon on the road map. By the help of GUI, users can also easily decide whether the traffic lights on the specified intersections should be controlled by the application or module. When all the settings are done, GUI will generate configuration files for later use.

(2) SE (Simulation Engine)

SE is not only responsible for simulating the MAC (Media Access Control) layer and PHY (Physical) layer protocols but also responsible for updating   the

node location and signal status. During the simulation, SE will periodically transmit the node location and signal status to GUI. As such, GUI program can use the information to display the simulation status during simulation and play back simulation results when simulation is done.

(3) Node Module

Node module is implemented in the SE. It also stores the car information database. Each vehicle node on the GUI is equipped with a node module. An intersection is represented as a node on the GUI and is also equipped with a node module, so that the traffic lights on the intersection can be controlled by the signal agent logic in node module.

(4) Car Agent Logic

Car agent logic is implemented in the node module. A vehicle controlled by car agent logic has the driving behaviors such as lane switching, accelerate/decelerate, car following, traffic lights recognition, etc. Car agent logic will periodically update vehicle's location by inserting the location update event into the event scheduler in SE. Each car can directly access the car information database to get necessary car information. For example, one vehicle can get other vehicles' locations by querying the car information database. The car can also access the signal information database to get the traffic lights information.

(5) Signal Agent Logic

Signal agent logic is also implemented in the node module. The node module which controls the traffic lights will update its traffic light periodically by inserting the signal status update event into the event scheduler in SE. Signal agent logic can directly access the signal information database to update its signal status.

(6) Car Information Database

Car information database stores the vehicle's current driving status, such as its current moving direction, the current road where it is located, its current moving speed, etc.

(7) Signal Information Database

Signal information database stores the signal's current traffic light status, such as the current status of traffic light and the signal's location.

(8) Road Map Database

Road map database is stored in SE. It provides the road map information to car agent, such as road's direction, road number, road's location, intersection location, the relationship between each roads and intersections, etc.

3.2. Module-based WAVE Short Message Transmission
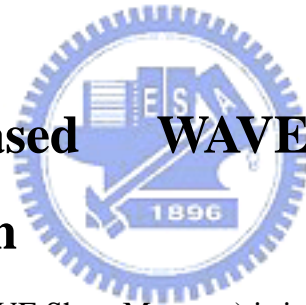
Module-based WSM (WAVE Short Message) is implemented in node module. As such, the module-based car and signal agent can use WSM to transmit small sized information to each other. Fig. 3.2 shows how network protocol stacks and the transmission of module-based WSM are simulated on NCTUns. In the example, two module controlled vehicles move on the road and they exchange WSM packets with each other. To simulate this case, two module-based car agents are responsible for each vehicle and control these two vehicles' moving behaviors. In addition, SE is run up to simulate the WME (WAVE Management Entity), MAC (Media Access Control) layer, and PHY (Physical) layer of IEEE 802.11p/1609 protocol stacks. Suppose that the left module-based car agent sends a WSM packet to the right module-based car agent, the detailed WSM packet delivery procedure is described below.

1. The module-based car agent logic of the sending node module encapsulates the WSM packet and sends this WSM packet to the WME inside the SE.

2. The WME, MAC, and PHY protocols are simulated in SE. When this WSM packet arrives at the sending WME, this WSM packet will be encapsulated with a MAC header and then sent from the sending PHY to the receiving PHY. The channel model (CM) module is used to simulate the power loss of the packet on the air.

3. The MAC header of the WSM packet will be stripped off when this packet arrives at the receiving MAC. When this WSM packet arrives at the receiving WME module, it will be sent to the receiving node module.

4. Finally, the module-based car agent logic of receiving node module retrieves the WSM data from this WSM packet.
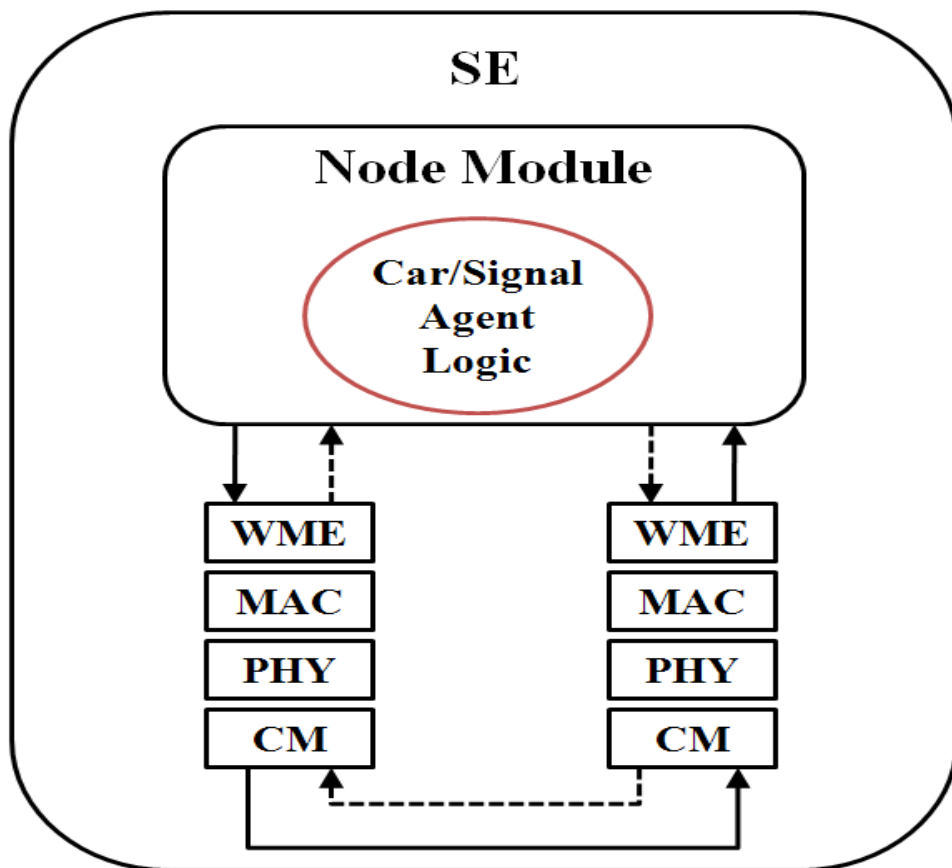

Fig. 3.2   Architecture of Module-based WSM Simulation

3.3. Advantages of Module-based Agent Logic Control

Module-based car agent and signal agent logic control has three main advantages, which are (1) break the limitation of simulated node numbers, (2) efficient information update, and (3) reduce required memory space. Those benefits are described below.

(1) Break the Limitation of Simulated Node Numbers

The IP address format on NCTUns is 1.0.subnetID.hostNumberOnThisSubnet, where subnetID and hostNumberOnThisSubnet are automatically assigned by NCTUns GUI program. Due to this address format, NCTUns allows a simulation case to have up to 254 subnets (subnetID 0 and 255 are excluded because they are used for broadcast purposes), each of which can have up to 254 nodes (hostNumberOnThisSubnet 0 and 255 are excluded because they are also used for broadcast purposes). Therefore, in total a maximum number of 254 * 254 = 64,516 layer-3 interfaces can be supported in a simulation case.

Although in theory, NCTUns can deploy up to 64,516 nodes in a simulation case, in practice it is rarely achieved. This is because each node requires a layer-3 interface and each layer-3 interface needs to be simulated by a tunnel network interface. However, currently the installation script creates only 4,096 tunnel interfaces on a UNIX machine by default. Therefore, the current NCTUns can support a simulation case using up to 4,096 layer-3 interfaces. This also means that currently the maximum number of IEEE 802.11p nodes in a vehicular network topology cannot exceed 4,096. (This is because each IEEE 802.11p node occupies a layer-3 interface.) This limitation can be easily raised to a number such as 8,192 by creating more tunnel interfaces on the UNIX machine. However, it is still not enough to simulate a large-scale vehicular network. We can use module-based agent logic control to simulate the moving

behaviors and WSM transmission of large-scale vehicles. Using the module-based agent logic control doesn't occupy any tunnel interface, and therefore we can simulate up to 64,516 nodes.

(2) Efficient Information Update

The original car and signal agent processes need to use IPC commands to get information from SE process. Because the IPC command is blocking system call, it will cost a lot of time to wait for the response from SE. For example, if the car agent sends a location update command to SE using IPC, it needs to wait till SE returns the success message of updating location. Also, when car/signal agent process finishes its job, it will return the CPU control to SE so that SE can continue its jobs such as refresh the virtual time, simulate network protocol, etc. However, when the number of car and signal agent processes increases, the context switch time between processes will be significant. Using module-based car and signal agent logic control instead can save the time of IPC communication and context switch between processes.

(3) Reduce Required Memory Space

Original car and signal agent are the processes forked by SE process. As one can see in Fig. 2.5, each car agent process has one copy of the road map database. Therefore, it will require plenty of memory to store the same roadmap data if there are a lot of vehicles deployed on the road map. Using module-based car agent logic control, there is only one copy of road map database in SE and it reduces the memory space occupied by agent processes.

Chapter 4. Design and Implementation of the Proposed Scheme

In this chapter, we will describe the design of the proposed traffic control mechanism, design of the scheme, and target scenarios.

4.1. Design of the Proposed Mechanism

The proposed traffic control mechanism utilizes three functions, which are (1) moving path arrangement, (2) evacuation message delivery, and (3) dynamic traffic lights control, to make emergency vehicle drive fast and smoothly. Designs of these three functions are described below.

(1) Moving Path Arrangement

The traffic control center will arrange a moving path for the emergency vehicle to reach the accident scene. This moving path must be the shortest-time path to avoid emergency vehicle driving on the longer-time path to the accident scene. The shortest-time path generation is based on the A* algorithm. A* algorithm is a best-first, graph search algorithm that can find the path with the minimum cost from the source to the destination. The procedure of generating shortest-time path for emergency vehicles is described below.

1. When Wan-Fang hospital receives an emergency call for an accident, it reports the location of the accident to the traffic control center.

2. When the traffic control center receives the report, it will generate a shortest-time path from the source (Wan-Fang hospital) to the destination (the reported location of the accident). The traffic control center then transmits the path back to the hospital so that the emergency vehicle can drive based on the shortest-time path.

(2) Evacuation Message Delivery

The evacuation message encapsulated in the WSM (WAVE Short Message) is used to inform the nearby vehicles of the approaching of emergency vehicles so that vehicles can make their way for emergency vehicles. The procedure of evacuation message delivery is described below.

1. The emergency vehicle periodically broadcasts the evacuation request message encapsulated in the WSM to the nearby RSU. The message contains the ID and the location of the emergency vehicle.

2. While the RSU near the emergency vehicle receives the evacuation request message, it delivers this message to the traffic control center through the wired link.

3. When the traffic control center receives the message, because it knows the moving path of the specified emergency vehicle ID, it sends a new evacuation message to the RSU which is installed at the intersection in front of the specified emergency vehicle. This new evacuation message contains the current and next road segment IDs which the emergency vehicle will pass through.

4. When the RSU which is installed at the intersection in front of the emergency vehicle receives the new evacuation message, it broadcasts the message using WSM to inform the nearby vehicles.

5. When the vehicle receives the WSM message, if it may block the way of the emergency vehicle based on the road segment IDs contained in the message, it will make the way for the emergency vehicle. So that the emergency vehicle can drive without been obstructed by other vehicles.
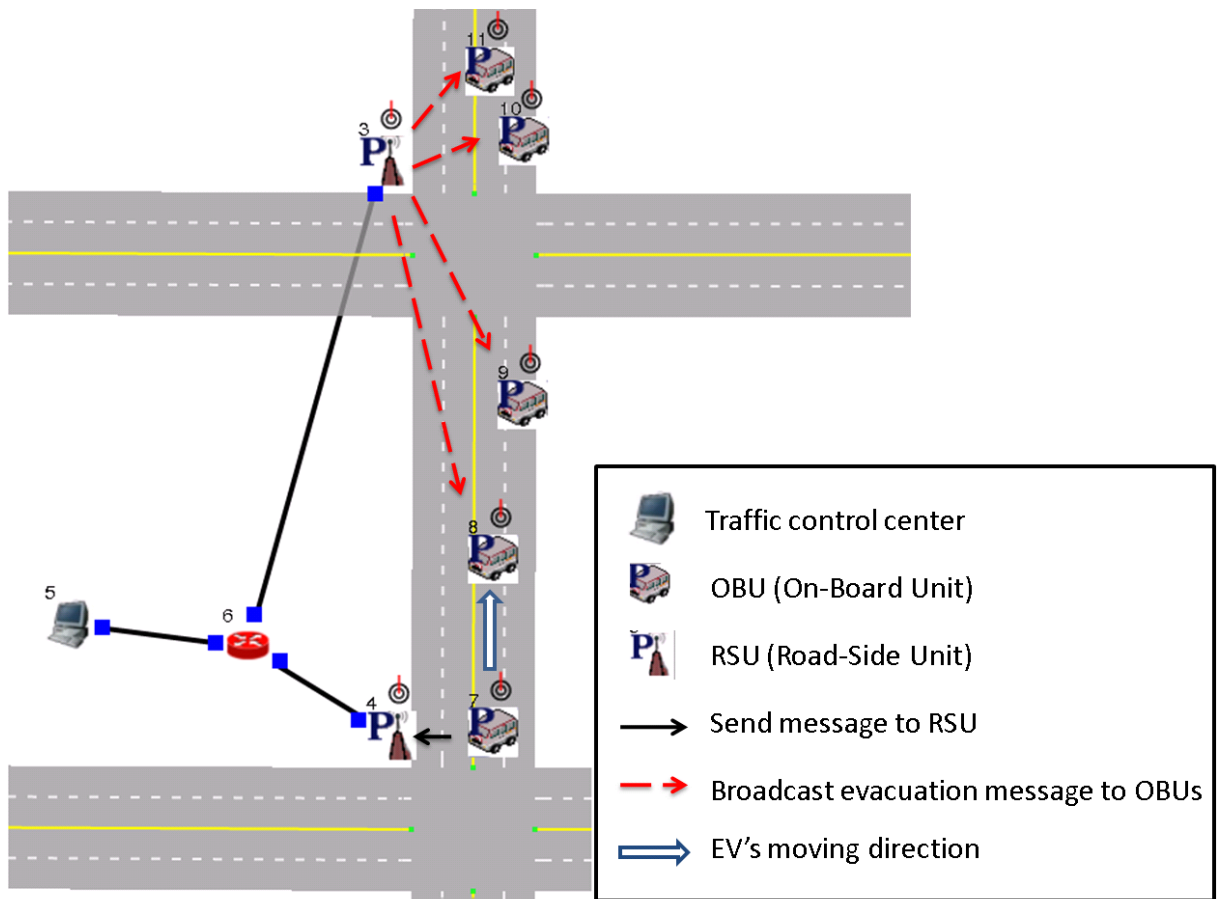
Fig. 4.1   A Simple Case of Evacuation Message Delivery

For example, as one can see in Fig. 4.1, the emergency vehicle (node 7) periodically broadcasts the evacuation request message using WSM which contains its vehicle ID and location. The RSU (node 4) installed on the nearest intersection will receive this WSM message and then deliver this message to the traffic control center (node 5). When the traffic control center receives the evacuation request, it will find out the current and next road segment IDs of the specified emergency vehicle based on the shortest-time path. Then, the traffic control center will sends the evacuation message which contains the current and next road segment IDs of the emergency vehicle to the RSU (node 3) which is installed in front of the specified emergency vehicle. The RSU (node 3) will then broadcast the evacuation message using WSM to nearby vehicles (node 8, 9, 10, and 11). When vehicles (node 8 and 11) receive the broadcasted WSM message from the RSU and find out they may block the

29.

way of emergency vehicle (node 7), they will make the way for the emergency vehicle to pass through. As such, the emergency vehicle can drive on a clear lane fast and safely.

(3) Dynamic Traffic Lights Control

The traffic lights are controlled by the traffic control center. The emergency vehicle periodically broadcasts the WSM message which containing the request of traffic lights control, its vehicle ID and location to the nearby RSU. The RSU will receive and relay this message to the traffic control center. When the traffic control center receives the request of traffic lights control, it will change the traffic lights on the intersection, which is in front of the emergency vehicle, into green to pre-evacuate vehicles waiting for the traffic lights. By doing so, we can avoid the emergency vehicle stuck in the traffic jam in front of the intersection.
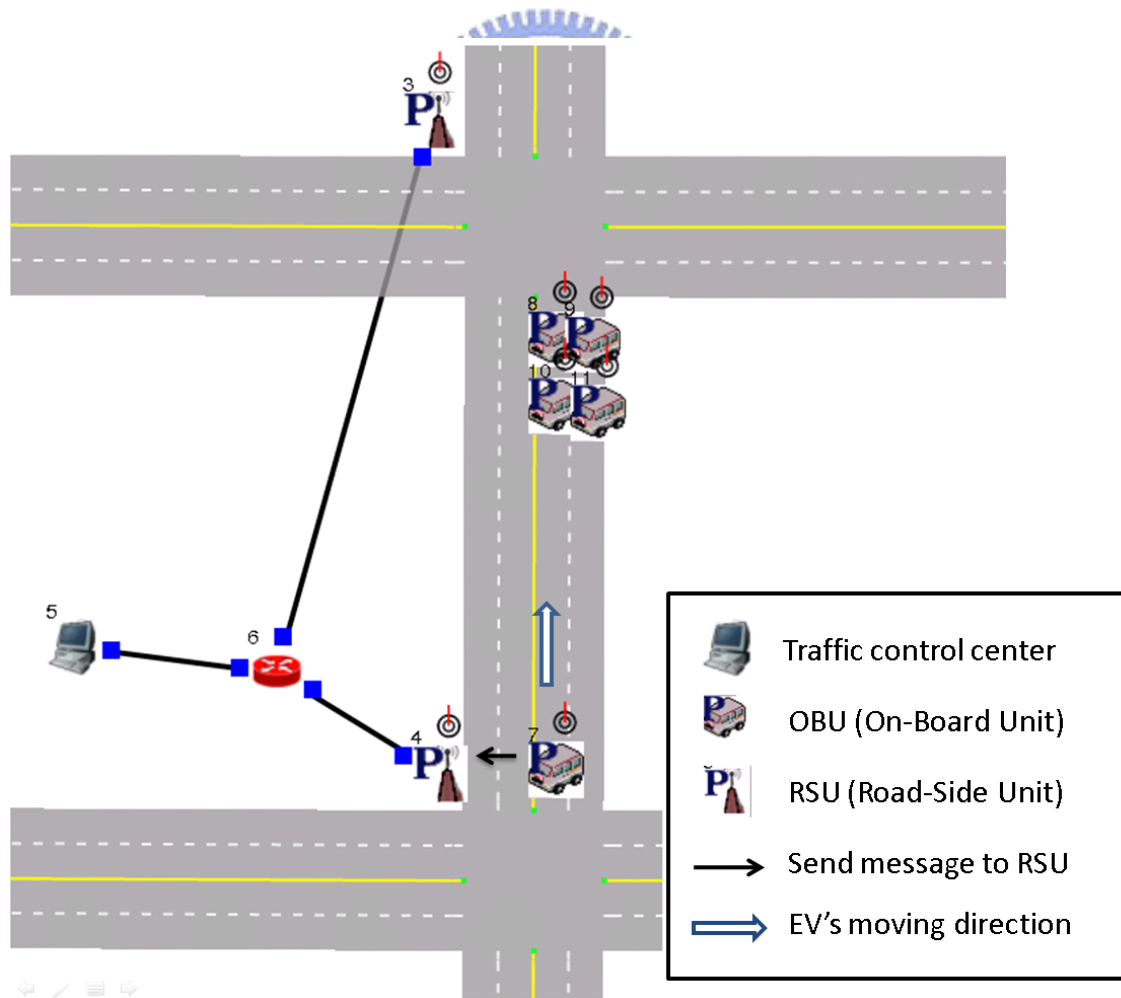

Fig. 4.2　A Simple Case of Dynamic Traffic Lights Control

30.

For instance, as one can see in Fig. 4.2, the emergency vehicle (node 7) periodically broadcasts a WSM message which contains the request of traffic light control, vehicle ID, and location. The nearby RSU (node 4) receives the WSM message and relays this message to the traffic control center (node 5). Because the traffic light control center has the shortest-time path information of the specified emergency vehicle ID, it changes the traffic lights in front of the emergency vehicle into green. As such, vehicles (node 8, 9, 10, and 11) can be pre-evacuated and will not make the emergency vehicle stuck in the traffic jam.

4.2. Design of the Scheme

The proposed traffic control mechanism is designed to reduce the arrival time of the emergency vehicles to accident scenes. As described before, the traffic control mechanism utilizes three major functions which are (1) moving path arrangement, (2) evacuation message delivery, and (3) dynamic traffic lights control. We design various schemes to evaluate how each scheme affects the arrival time of emergency vehicles to accident scenes.

As one can see in Table 4.1, we design four schemes to evaluate the arrival time of emergency vehicles to accident scenes. These schemes are described below.

Scheme A: The proposed traffic control mechanism only applies the moving path arrangement. The emergency vehicle will drive on the shortest-time path to the accident scene. This scheme is the basic scheme used to compare with other schemes so that we can evaluate the performance of each scheme.

Scheme B: The proposed traffic control mechanism applies both moving path arrangement and evacuation message delivery. This scheme can help us analyze the effect of the evacuation message delivery by comparing it to scheme A.

Scheme C: The proposed traffic control mechanism applies not only moving path arrangement but also dynamic traffic lights control. By comparing to scheme A, we can

evaluate the effect of the dynamic traffic lights control.

Scheme D: The proposed traffic control mechanism utilizes all three major functions which are moving path arrangement, evacuation message delivery, and dynamic traffic lights control. By comparing to scheme A, we can know the best performance improvement of the proposed mechanism for emergency vehicles.

Table 4.1 Functions Used in Each Scheme

	Scheme A	Scheme B	Scheme C	Scheme D
Moving Path Arrangement	∨	∨	∨	∨
Evacuation Message Delivery		∨		∨
Dynamic Traffic Lights Control			∨	∨

4.3. Target Scenario

The target scenario simulated in this thesis is the road map near the Wan-Fang hospital at Daan area in Taipei. Fig. 4.3 shows the road map near the Wan-Fang hospital captured from the electronic roadmap system, UrMap [15]. We set the source S on the Wan-Fang hospital and two accidents on position A and B. Therefore, there are two paths which are S -> A and S -> B we will use in our simulation cases.

Path	Hospital to Accident
Path 1	S to A
Path 2	S to B

Fig. 4.3   The Roadmap near Wan-Fang Hospital on UrMap

One can see the simulated road map on NCTUns in Fig. 4.4. With the help of NCTUns network simulator, we can construct the road network by loading the real world road map with shapefile format so that the moving paths of vehicles can be as real as possible. The parameter settings of the target scenarios are described below.

The figure shows a NCTUns 5.0 screenshot with a roadmap near Wan-Fang Hospital, including an embedded table:

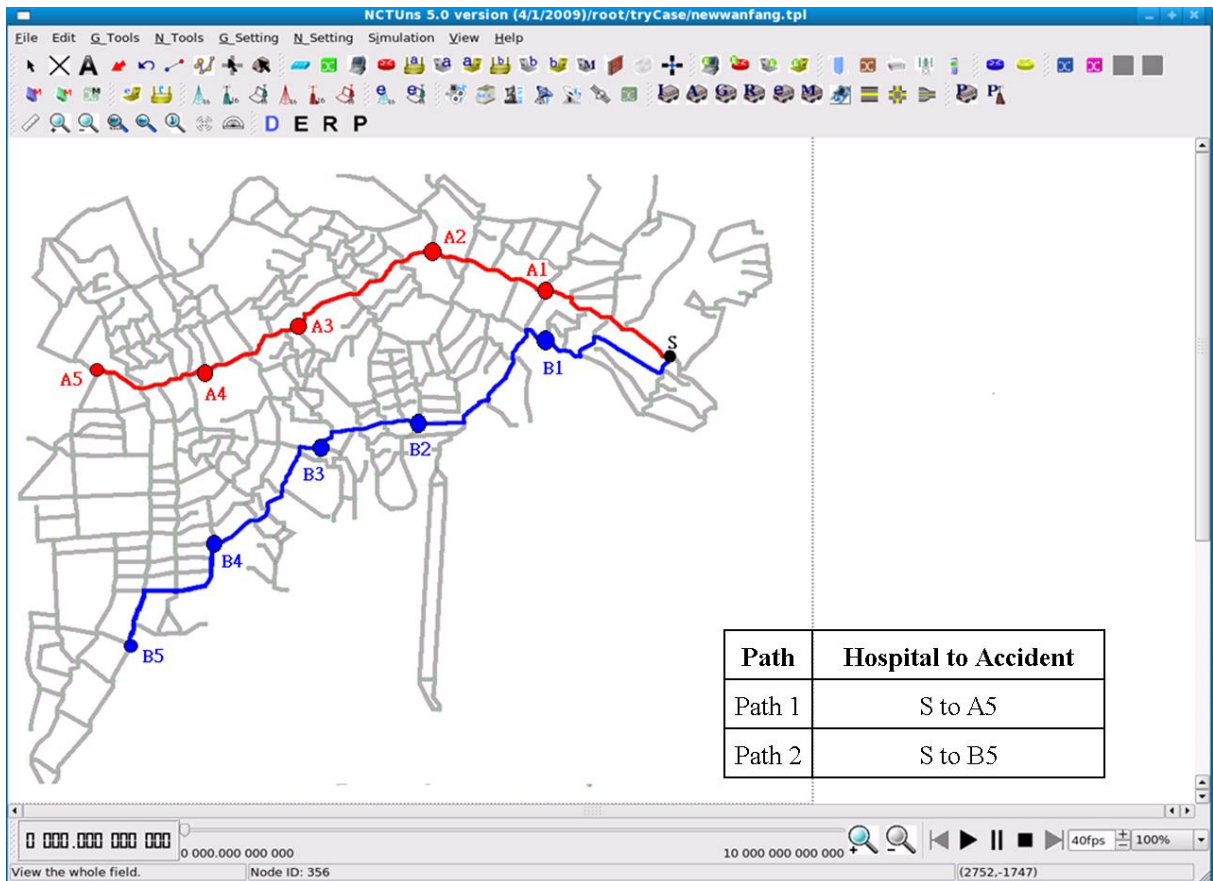| Path	Hospital to Accident
Path 1	S to A5
Path 2	S to B5

Fig. 4.4　The Roadmap near Wan-Fang Hospital on NCTUns

As one can see in Fig. 4.4, position S is the source location of the emergency vehicle at Wan-Fang hospital, position A5 and B5 are the locations which are 25 intersections away from the source S. One should know that the location A5/B5 is equal to the location A/B as shown in Fig. 4.3. The parameters used in the simulated scenarios are shown in Table 4.2. For each path, we set the destinations every 5, 10, 15, 20, and 25 intersections away from the source S which are A1 to A5 and B1 to B5. We also apply various time periods of traffic lights, which are 30, 60, 90 seconds, to analyze if time periods of traffic lights will affect the performance of the proposed traffic control mechanism for each scenario. As shown in Table 4.3, we evaluate two shortest time paths, which are path 1 and path 2, to see if driving on different paths will significantly affect the performance of the proposed mechanism. Table 4.4 shows the average vehicle separation distance for each scenario. We deploy

vehicles on the road map using various vehicle separation distances which are 10, 20, and 30 (meter/vehicle/lane), respectively. This is equivalent to 30, 60, 90 road surface area per vehicle when the road width is 3 meter. By doing so, we can analyze how average vehicle separation distance affects the performance for each scenario.

Table 4.2 Parameter Settings of the Simulated Scenarios

Parameter	Setting
Message Size (bytes)	< 1,400
Network Protocol	IEEE 802.11p/1609 WSMP
Source	Wan-Fang Hospital
Simulation Time	Source to destination
Network and Road Traffic Simulator	NCTUns
Terrain Size (meter^2)	2,000 * 1,500
Number of Vehicles	1,500 ~ 4,000
Road Width (meter)	3
Road Surface Area per Vehicle (meter^2/vehicle)	30, 60, 90
Destination (number of intersections away from the source)	5, 10, 15, 20, 25
Traffic Lights Period (second)	30, 60, 90

Table 4.3 Paths from Hospital to Accidents

Path	Hospital to Accident
Path 1	S to A
Path 2	S to B

Table 4.4 Average Vehicle Separation Distance

Traffic Condition	Average Vehicle Separation Distance (meter/vehicle/lane)
A	10
B	20
C	30

Chapter 5. Simulation Results

We use the arrival time reduction percentage (ATRP) based on scheme A to evaluate the performance of a scheme under different traffic conditions, paths, and traffic lights periods. The larger the ATRP, the better performance a scheme is. The definition of ATRP is describe below.

Assume the arrival time of scheme A needs X seconds and the arrival time of an evaluated scheme B, C, or D needs Y seconds, then the ATRP of the evaluated scheme is calculated follows:

$$\text{ATRP}\,(\%) = \frac{(X-Y)}{X} * 100\,\%$$

For conciseness, Table 5.1 shows the abbreviated terms that are used in the rest of this thesis.

Table 5.1 Abbreviations

Term	Abbreviation
Arrival Time Reduction Percentage	ATRP
Traffic Lights Period	TLP
Average Vehicle Separation Distance	AVSD

As one can see in Fig. 5.1, the ATRP results under traffic condition A and when TLP is 30 seconds show that the performance of scheme A < that of scheme B < that of scheme C < that of scheme D. The reasons are explained below.

On average, performances of scheme B, C, and D are better than scheme A. This is because both evacuation message delivery and dynamic traffic lights control can greatly help emergency vehicle drive fast and smoothly to the accident scene. The reason that scheme C is better than scheme B is because the dynamic traffic lights control can prevent

the emergency vehicle from being stuck in the traffic jam in front of the intersection. It also shows the effect of dynamic traffic lights control is greater than that of evacuation message delivery. Finally, the reason why scheme D has the best performance is because it not only avoids the traffic obstruction in front of the traffic lights but also clears the way for the emergency vehicle.

As shown in Fig. 5.1, in Fig. 5.2 and Fig. 5.3, one sees that when the AVSD increases, the performance of each scheme decreases. This is because when AVSD becomes larger, the emergency vehicle encounters less blocking of surrounding vehicles and therefore the arrival time of the emergency vehicle to the accident scene becomes shorter in scheme A. Compared with Fig. 5.4, in Fig. 5.5 and Fig. 5.6 one sees that the arrival time of scheme A increases when AVSD decreases which proves the above description.
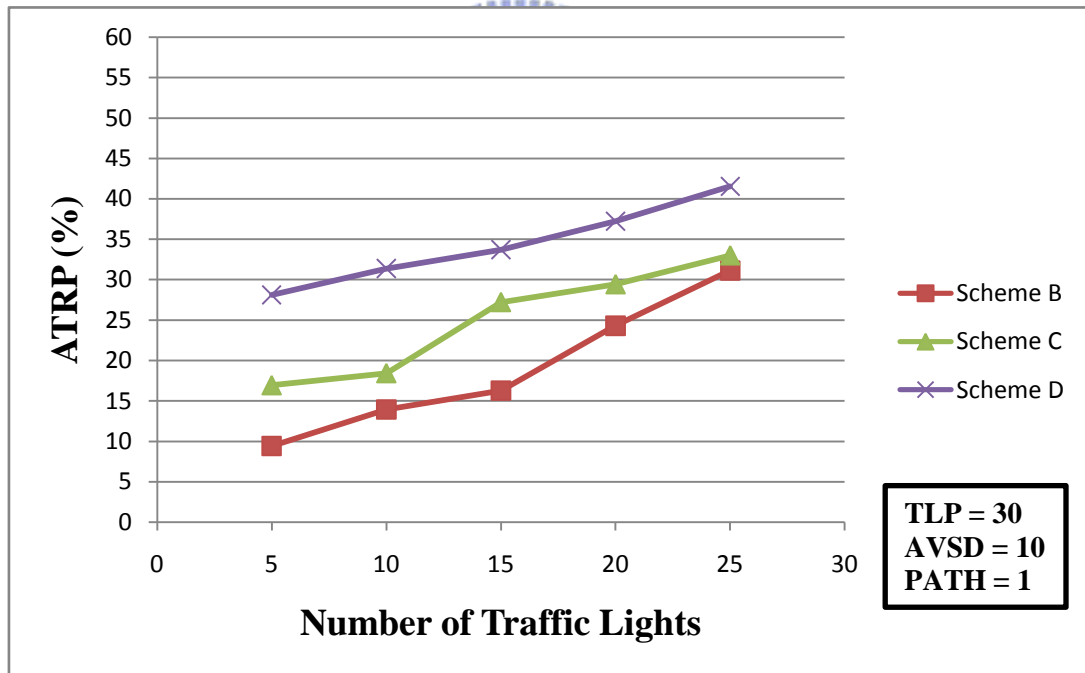

Fig. 5.1   The ATRP Results of Each Scheme on Path 1 under Traffic Condition A and when TLP is 30 seconds
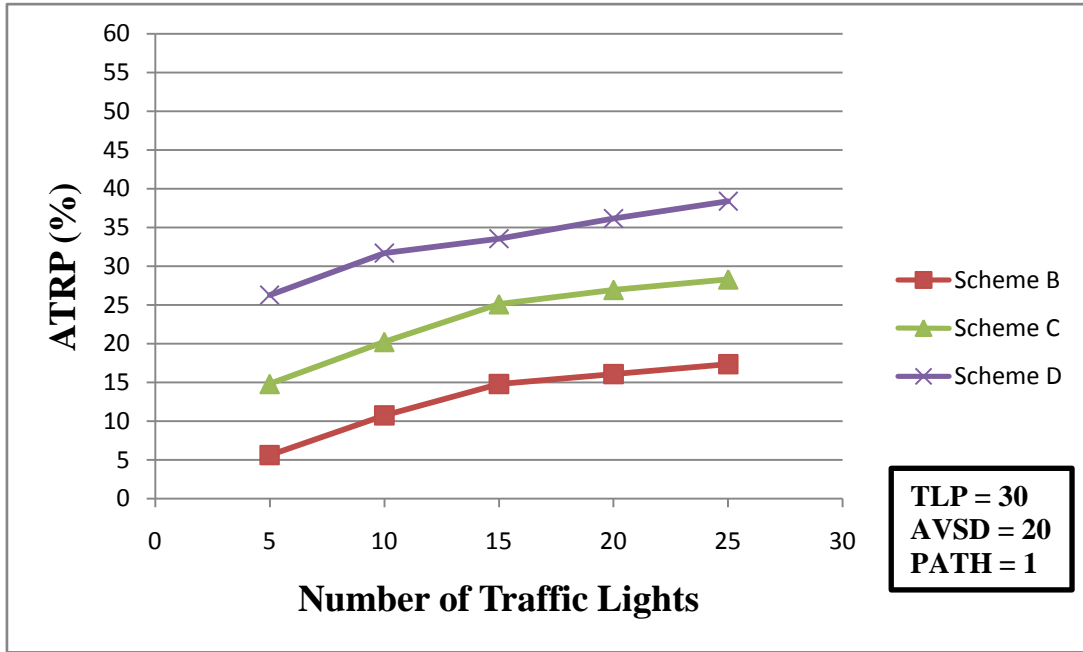
Fig. 5.2   The ATRP Results of Each Scheme on Path 1 under Traffic Condition B and when TLP is 30 seconds
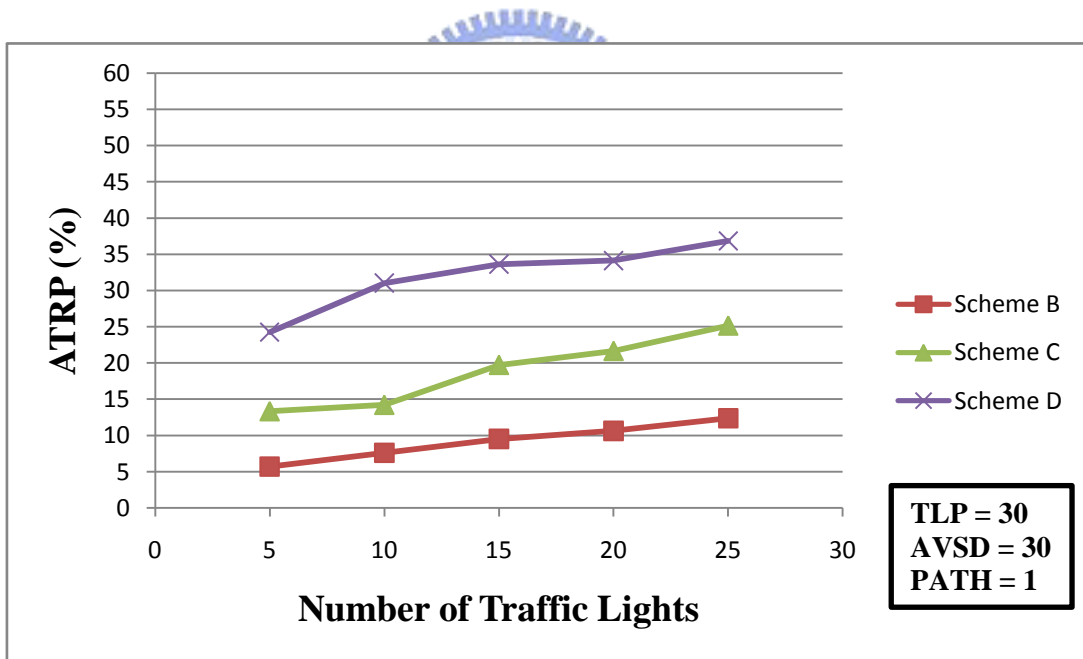

Fig. 5.3   The ATRP Results of Each Scheme on Path 1 under Traffic Condition C and when TLP is 30 seconds
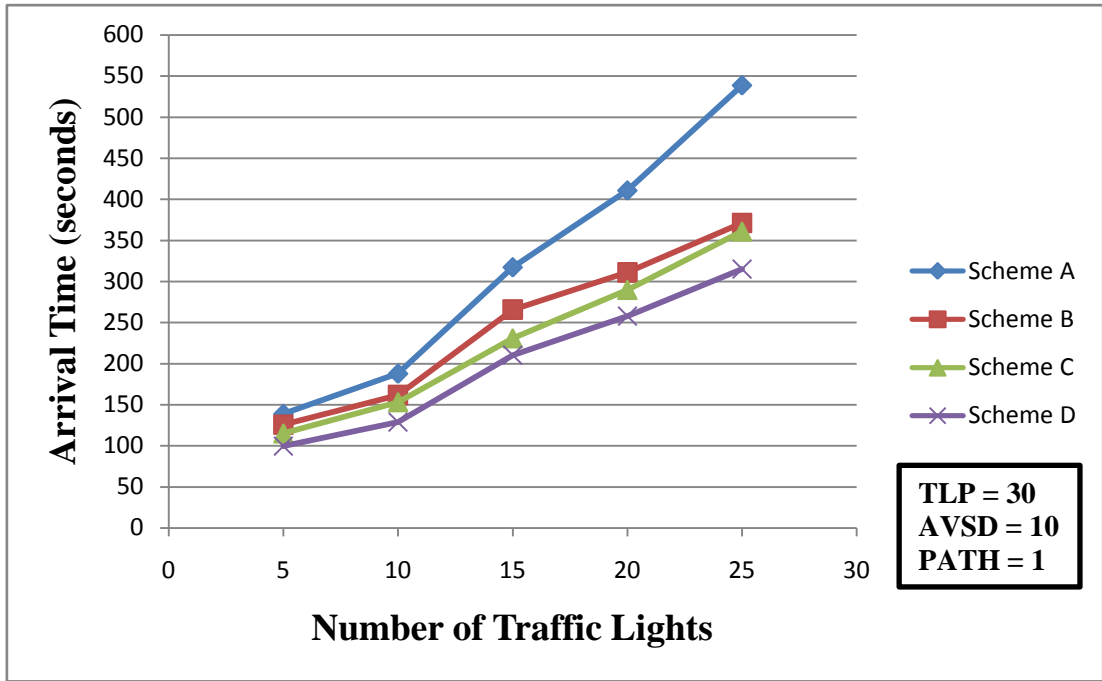
Fig. 5.4   The Arrival Time of Each Scheme on Path 1 under Traffic Condition A and when TLP is 30 seconds
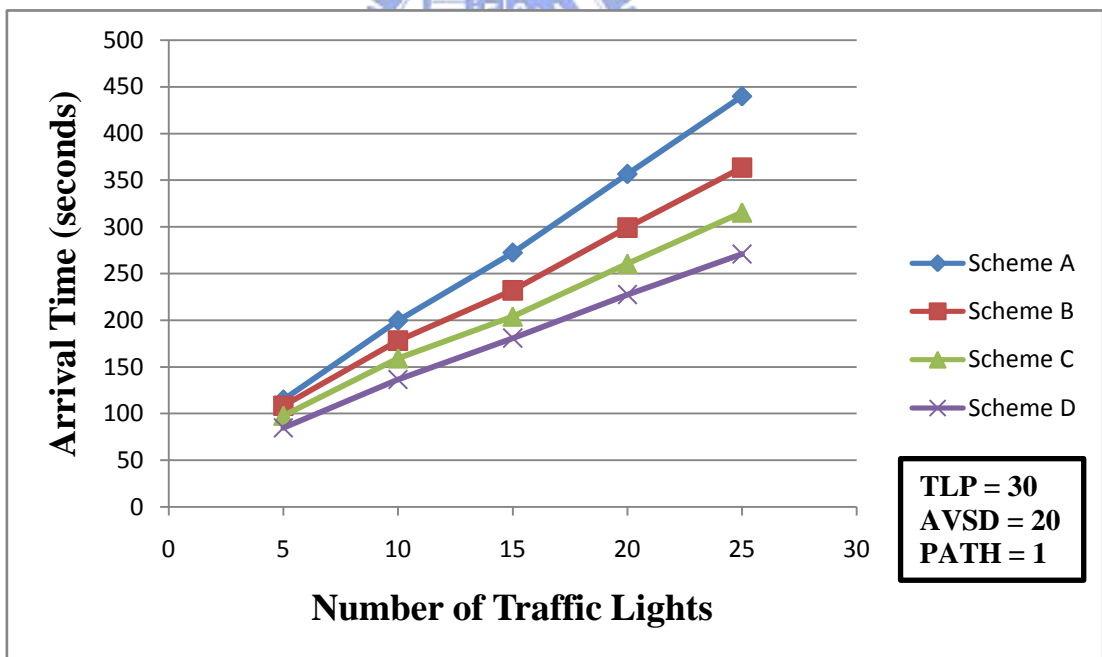

Fig. 5.5   The Arrival Time of Each Scheme on Path 1 under Traffic Condition B and when TLP is 30 seconds
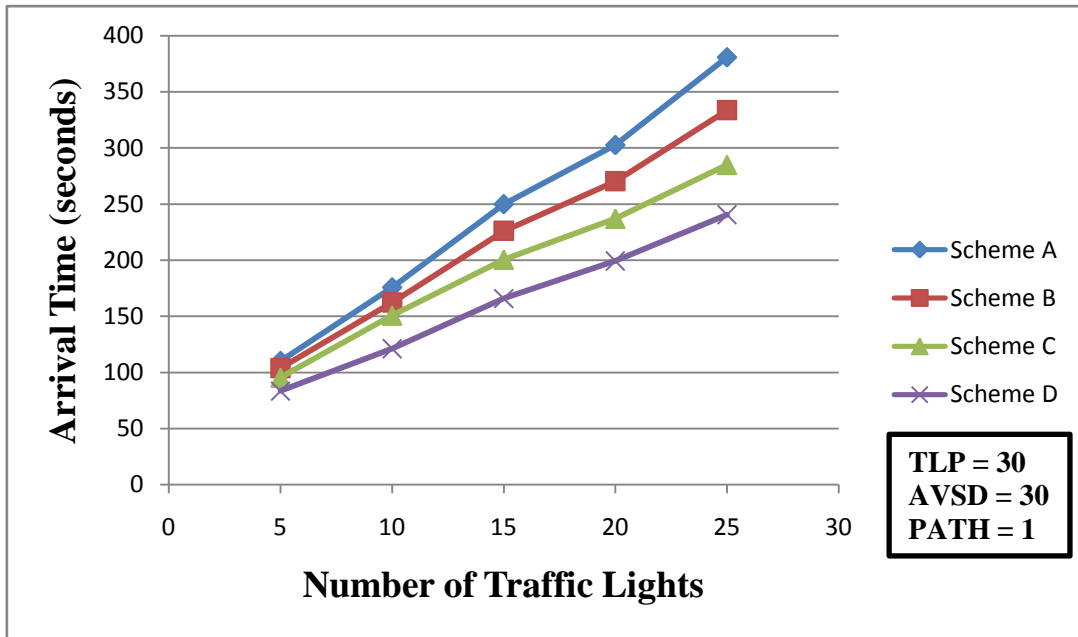
Fig. 5.6   The Arrival Time of Each Scheme on Path 1 under Traffic Condition C and when
TLP is 30 seconds

Compared with Fig. 5.1, in Fig. 5.8, one sees that the performances of scheme B, C, and D on path 2 are better than their performances on path 1 under traffic condition A and when TLP is 30 seconds. This is because the total distance of path 2 is longer than that of path 1 and therefore, in scheme A, emergency vehicle encounters more blockings caused by surrounding vehicles and needs more time to arrive at the accident scene on path 2. Fig. 5.4 and Fig. 5.7 show the same results as described above. One can see that the arrival time of scheme A in Fig. 5.7 is larger than in Fig. 5.4 but the arrival times of scheme B, C, and D in Fig. 5.7 are just a little more than their arrival times in Fig. 5.4. Therefore, the performances of scheme B, C, and D on path 2 are better than their performances on path 1 under traffic condition A and when TLP is 30 seconds.
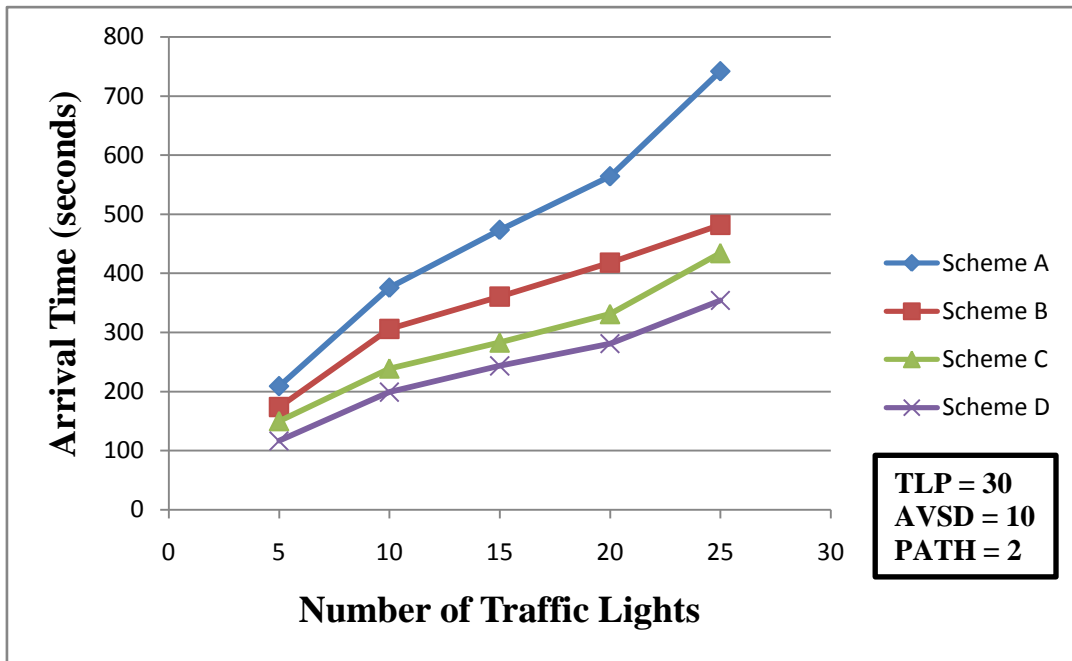
Fig. 5.7   The Arrival Time of Each Scheme on Path 2 under Traffic Condition A and when
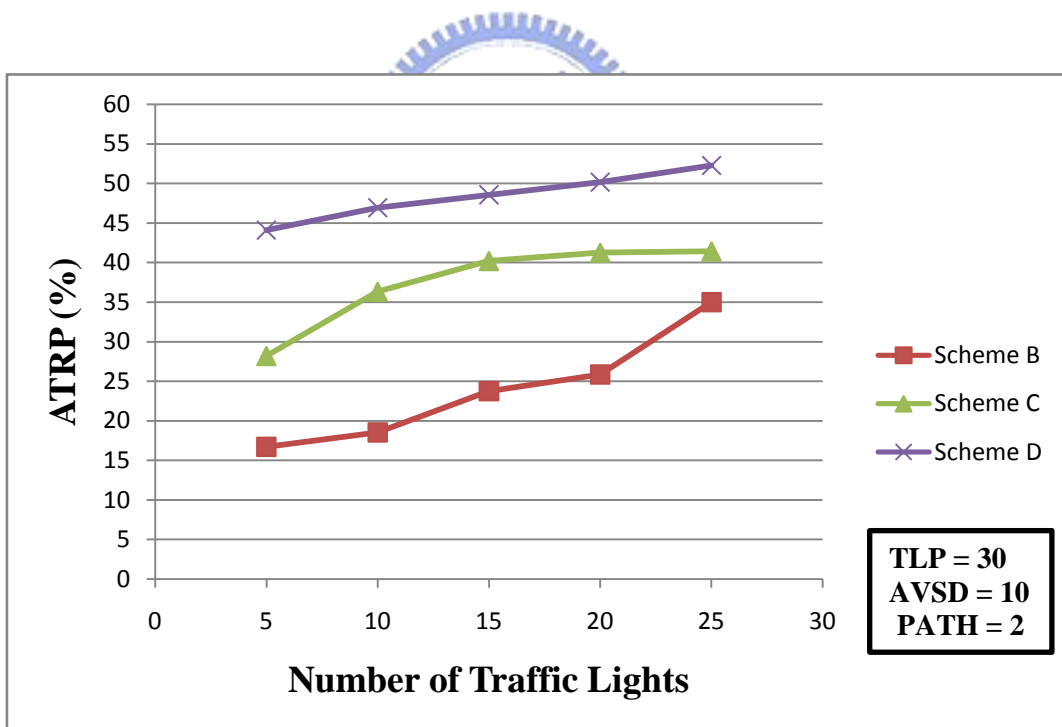TLP is 30 seconds

Fig. 5.8   The ATRP Results of Each Scheme on Path 2 under Traffic Condition A and when
TLP is 30 seconds

When the TLP increases, the emergency vehicle can pass more traffic lights and also

move smoothly during the long green time in scheme A. However, the traffic lights periods

will not significantly affect the arrival time of the emergency vehicle when the emergency vehicle uses scheme D. This is because the emergency vehicle using scheme D has the ability to control the traffic lights and empty a lane for itself. As shown in Fig. 5.9, one sees that the arrival time of scheme A increases when TLP increases. One also sees that the arrival time of scheme D is almost the same when TLP is variable in Fig. 5.10. Therefore, as shown in Fig. 5.11, the ATRP performance of scheme D decreases when TLP increases.
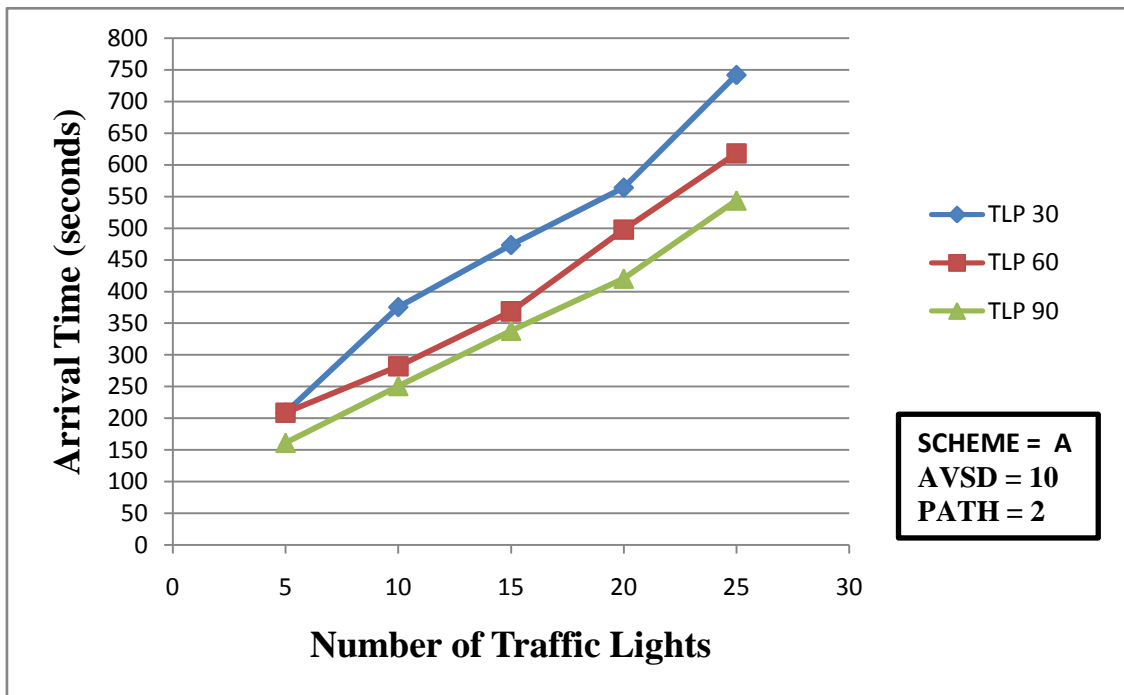

Fig. 5.9   The Arrival Time of Scheme A on Path 2 under Traffic Condition A when TLP is Variable
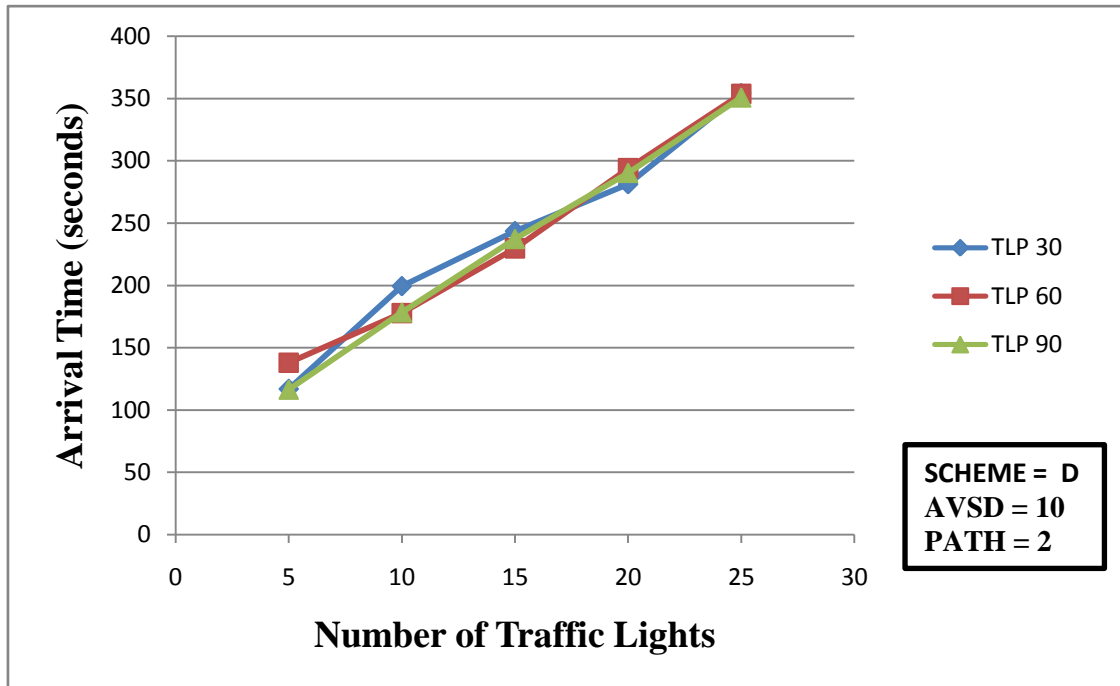
43.

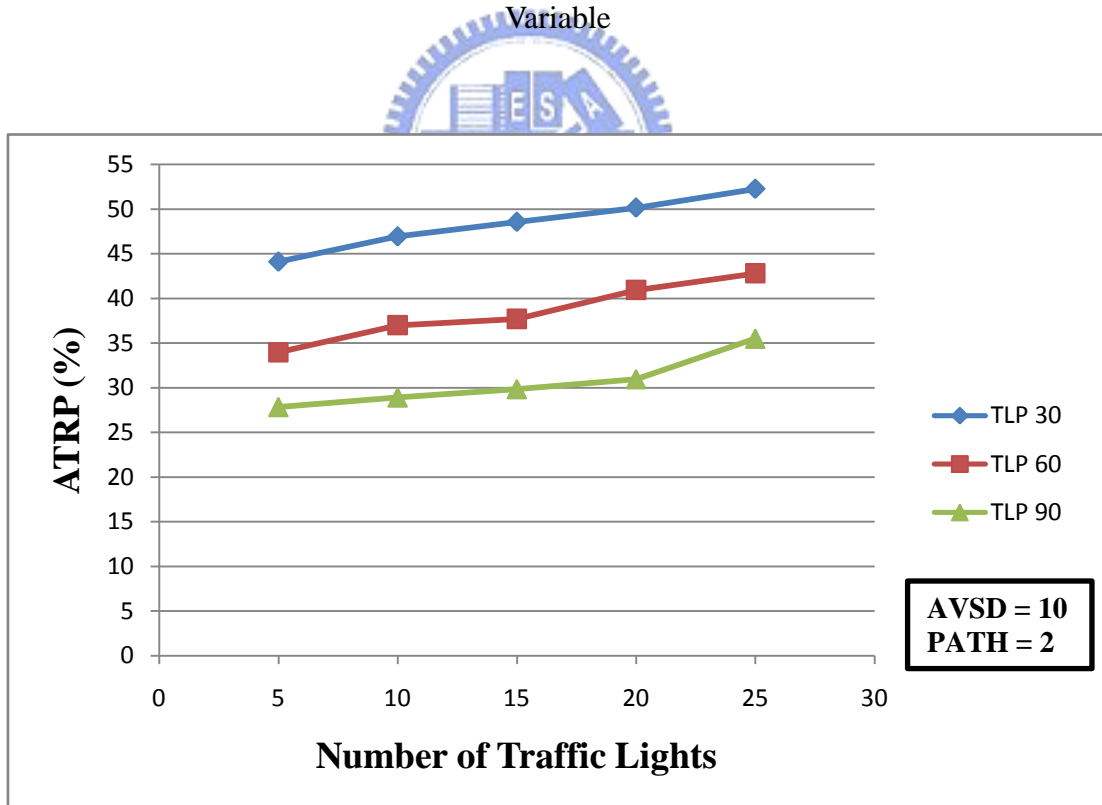Fig. 5.10 The Arrival Time of Scheme D on Path 2 under Traffic Condition A when TLP is Variable

Fig. 5.11 The ATRP Results of Scheme D on Path 2 using Various TLPs under Traffic Condition A

44.

Chapter 6. Future Work

There are two enhancements that can be done in the future.

➢ Use sensor-based locally controlled traffic lights switching

The traffic lights switching used in this thesis is globally synchronized. This means that all the traffic lights statuses on the roadmap are changed in the same time. We may develop a sensor-based locally controlled traffic lights switching which can sense the traffic in the intersection so that the traffic lights can locally switch to different signal status based on the different traffic condition in the intersection. We can evaluate the effects of globally synchronized traffic lights switching and sensor-based locally controlled traffic lights switching for our proposed traffic control mechanism.

➢ Assign the destination for every vehicle

In this thesis, surrounding vehicles use dynamic moving pattern to drive on the road map. Although the vehicle has abilities to accelerate, decelerate, switch lane, and detect the traffic lights, it doesn't have any pre-assigned destination but just randomly choose one direction at the intersection to move on. In the real world, every vehicle has its destination to go. We could enhance the NCTUns network simulation to support the assignment of destination for every vehicle so that the moving pattern of vehicles can be as reasonable as possible.

Chapter 7. Conclusion

In this thesis, a traffic control mechanism is proposed which utilizes three important functions: (1) Moving path arrangement, (2) Evacuation message delivery, and (3) Dynamic traffic lights control.

Our simulation results show that the proposed traffic control mechanism can significantly reduce the arrival times of emergency vehicles to accident scenes in our evaluated simulation cases. Our simulation results also show that the effect of dynamic traffic lights control is larger than that of evacuation message delivery.

In our simulation, the proposed traffic control mechanism has better performance on a dense vehicular network, i.e., when the average vehicle separation distance is small.   This is because the more vehicles on the roads, the more unexpected traffic blocking the emergency vehicle may encounter. The proposed traffic control mechanism can eliminate the unexpected traffic blocking and therefore an emergency vehicle can arrive at the accident scene much fast and smoothly.

We found that using different traffic lights periods can affect the performance of the proposed traffic control mechanism. Our results show that when the time period of traffic lights increases in a dense vehicular network, the performance of the proposed traffic control mechanism decreases. This is because when the green time increases, the traffic condition on the roads becomes better. As a result, the advantage of the proposed traffic control mechanism becomes less significant.

This thesis shows that our proposed traffic control mechanism can significantly reduce 30% to 50% arrival times required by emergency vehicles to arrive at accident scenes in our evaluated cases. It is a feasible and very useful mechanism for life saving.

Reference

[1] S.Y. Wang, C.L. Chou, C.H. Huang, C.C. Hwang, Z.M. Yang, C.C. Chiou, and C.C. Lin, "The Design and Implementation of the NCTUns 1.0 Network Simulator," *Computer Networks,* Vol. 42, Issue 2, June 2003, pp. 175-197.

[2] The NCTUns 5.0, available at *http://nsl10.csie.nctu.edu.tw/* .

[3] S.Y. Wang, C.L. Chou, Y.H. Chiu, Y.S. Tzeng, M.S. Hsu, Y.W. Cheng,W.L. Liu, and T.W. Ho, "NCTUns 4.0: An Integrated Simulation Platform for Vehicular Traffic, Communication, and Network Researches," *1st IEEE WiVec 2007 (International Symposium on Wireless Vehicular Communications),* Sep. 30, 2007, Baltimore, MD.

[4] "IEEE 802.11p/D3.0," *IEEE Standards Activities Department*. July 2007.

[5] "IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999," June 12, 2007.

[6] "IEEE 1609.1 Trial-Use Standard for Wireless Accesses in Vehicular Environments (WAVE)—Resource Manager," *IEEE Vehicular Technology Society.* October 2006.

[7] "IEEE 1609.2 Trial-Use Standard for Wireless Accesses in Vehicular Environments (WAVE)—Security Services for Applications and Management Messages," *IEEE Vehicular Technology Society.* October 2006.

[8] "IEEE 1609.3 Trial-Use Standard for Wireless Accesses in Vehicular Environments (WAVE)—Networking Services" *IEEE Vehicular Technology Society.* October 2006.

[9] "IEEE 1609.4 Trial-Use Standard for Wireless Accesses in Vehicular Environments (WAVE)—Multi-channel Operation" *IEEE Vehicular Technology Society.* October 2006.

[10] P. Hart, N. Nilsson, B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Transactions of Systems Science and Cybernetics,* Vol.

ssc-4, No. 2, July 1968.

[11] M. Miyawaki, Z. Yamashiro, T. Yoshida, "Fast Emergency Preemption Systems (FAST)," *Intelligent Transportation Systems, 1999. Proceedings. 1999 IEEE/IEEJ/JSAI International Conference on,* 5-8 Oct. 1999 Page(s):993 – 997.

[12] C.M. Huang, C.C. Yang, C.Y. Tseng, C.H. Chou, "A Centralized Traffic Control Mechanism for Evacuation of Emergency Vehicles Using the DSRC Protocol," *Wireless Pervasive Computing, 2009. ISWPC 2009. 4th International Symposium on,* 11-13 Feb. 2009 Page(s):1 – 5.

[13] S.R. Rizvi, S. Olariu, M.C. Weigle, M.E. Rizvi, "A Novel Approach to Reduce Traffic Chaos in Emergency and Evacuation Scenarios," *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th,* Sept. 30 2007-Oct. 3 2007 Page(s):1937 – 1941.

[14] R.C. Vlad, C. Morel, J.Y. Morel, S. Vlad, "A learning real-time routing system for emergency vehicles," *Automation, Quality and Testing, Robotics, 2008. AQTR 2008. IEEE International Conference on Volume 3,* 22-25 May 2008 Page(s):390 – 395.

[15] UrMap Electronic Roadmap System, available at *http://www.urmap.com/* .

[16] *CAR 2 CAR Communication Consortium,* "CAR 2 CAR Communication Con-sortium Manifesto, version 1.1," Aug. 2007.