國立交通大學

資訊科學與工程研究所

碩士論文

都市型車載隨意網路之可靠視訊串流機制

Resilient Video Streaming for Urban VANETs

研究生：何軒甫

指導教授：王國禎　博士

中華民國一百年六月

都市型車載隨意網路之可靠視訊串流機制

Resilient Video Streaming for Urban VANETs

研究生：何軒甫　Student: Hsuan-Fu Ho

指導教授：王國禎　Advisor: Kuochen Wang

國立交通大學
資訊學院
資訊科學與工程研究所
碩士論文

A Thesis

Submitted to Institute of Computer Science and Engineering

Department of Computer Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Sciencmae

June 2011
Hsinchu, Taiwan, Republic of China

中華民國一百年六月

都市型車載隨意網路之可靠視訊串流機制

學生：何軒甫　　　指導教授：王國禎 博士

國立交通大學 資訊學院 資訊科學與工程研究所

摘 要

在車載網路(VANET)上傳遞視訊串流是近幾年來熱門興起的研究議題，它能提供乘客更安全的行車環境及娛樂服務。但由於隨意網路中拓樸快速改變的特性，如何在車載網路的環境中提供穩定的車輛間資料傳輸路徑便成了一個很大的挑戰。也由於車載網路中的車輛使用無線網路作為傳輸媒介互相傳遞資料，而無線傳輸易於出現錯誤的特性，使得資料在多次傳輸的過程之中遺失，造成了接收端收到的影像畫質下降。在本篇論文中，我們提出了都市型車載隨意網路之可靠視訊串流機制(RVS)，它結合了車輛移動相似度及彈性巨方塊順序。我們使用容錯的機制來復原在傳輸過程中遺失的資料，藉以提升接收端的影像畫質，並且找出較為穩定的資料傳輸路徑，使車輛間資料傳輸更為可靠。資料傳輸路徑是由路段所組成而不是由車輛所組成，這樣可以減少路徑斷裂並且提升路徑的可靠度。彈性巨方塊順序將影像編碼成為多個片組，而每個片組是由許多巨方塊所組成，以提

供接收端影像容錯的能力。藉著車輛移動相似度可以選擇較穩定的相鄰車輛作為傳遞中繼點，可使車輛間資料傳輸更為可靠。為了評估本篇論文所提出的方法,我們比較了F-AODV ，它結合了彈性巨方塊順序及AODV的方法，而所比較的參數是封包傳輸率以及峰值信噪比。模擬的結果顯示，在都市環境下,與F-AODV 相比,我們提出的方法提升了 7% 的封包傳輸率及1.3 dB的峰值信噪比。

關鍵詞：視訊串流、移動相似度、彈性巨方塊順序、都市車載網路、基於路段

Resilient Video Streaming for Urban VANETs

Student: Hsuan-Fu Ho　　**Advisor: Dr. Kuochen Wang**

Department of Computer Science
National Chiao Tung University

Abstract

In recent years, video streaming in the vehicular ad hoc network (VANET) has become a popular research issue. Because of the rapid changing network topology, maintaining stable paths to support inter-vehicle video streaming transmissions is a great challenge in VANETs. Due to the error-prone characteristic of wireless communication, routing packets over multiple hops results in packet loss and causes reconstructed video of poor quality at the receiver. In this thesis, we propose a road-based resilient video streaming (RVS) scheme that integrates flexible macroblock ordering (FMO) and vehicle moving similarity. We intend to enhance video streaming quality at the receiver by recovering lost packets with error resilience and choose more stable routing paths to make inter-vehicle data transmissions more reliable. In our scheme, routing paths consisting of road blocks rather than vehicular nodes can reduce broken links to enhance route stability. FMO encodes images into slice groups consisting of macroblocks to provide error resilience at the receiver. Based on vehicle moving similarity our scheme selects more stable neighbors as relay nodes to make inter-vehicle transmissions more reliable. To evaluate the performance of the proposed RVS, we compare it with a classical ad hoc protocol, F-AODV, which combines FMO and AODV, in terms of delivery ratio and peak signal-to-noise ratio (PSNR). Simulation results under urban settings show that the proposed RVS improves the delivery ratio by 7% and the received video stream quality (in terms of PSNR) by 1.3 dB compared with F-AODV.

Keywords: Video streaming, flexible macroblock ordering, moving similarity, road-based, urban VANETs.

Acknowledgements

Many people have helped me with this thesis. I am in debt of gratitude to my thesis advisor, Dr. Kuochen Wang, for his intensive advice and guidance. I would also like to show my appreciation for all the classmates in the *Mobile Computing and Broadband Networking Laboratory* for their invaluable assistance and inspirations. The support by the National Science Council under Grants NSC 99-2219-E-009-006 and NSC 99-2218-E-009-002. is also gratefully acknowledged. Finally, I thank my father, my mother and my friends for their endless love and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The concept of leveraging wireless communication in vehicles has fascinated researchers since 1980s [1]. In the last few years, with the availability of global positioning system (GPS) receivers and wireless local area network (WLAN) transceivers, researches exploring the possible applications in the field of inter-vehicular communication draw much attention [2]. Those potential VANET applications can be categorized into three major types [2]: *road safety*, *transportation efficiency*, and *infotainment applications*. As to road safety, related applications aim to decrease significantly the number of road accidents. According to some studies [3], 60 percent of accidents could be avoided if drivers were provided with a warning half a second before the moment of collision [4]. There have been a lot of researches about increasing road safety, for instance, to avoid pile-ups by alerting the following cars. Regarding to transport efficiency, related applications can provide drivers more information to choose better routes to their destinations. This would mitigate road congestion and maintain a smooth traffic flow, and then increase the capacity of the roads.

In addition to the traffic safety and transport efficiency issues, infotainment will become an important and potential application type in VANETs in the near future. Its applications can provide additional information or entertainment, such as advertisement or multimedia streaming, to the passengers. With the advanced multimedia streaming technology, multimedia files can be shared within vehicles in a VANET. There are large market potential of advertisements with the VANET technology, such as reception of data from commercial vehicles and roadside infrastructure about local businesses (*wireless advertising*). Enterprises (shopping malls, fast food, gas stations, and hotels) can set up stationary gateways to transmit marketing data to potential customers passing by.

1.1 Motivation

As mentioned above, VANETs applications can bring a lot of conveniences to our daily life, especially entertainment applications. However, transferring video streaming in VANET scenarios has many challenges. In urban VANET, each vehicle is independent and moves in constrained areas, the city streets. It is a self-configuring network of vehicles connected with wireless links. Because of high mobility in VANETs, wireless links can be disconnected frequently and routing paths may be very unstable most of the time. Due to the error-prone characteristic of wireless communication, routing packets over multiple hops results in packet loss and causes poor quality of reconstructed video at the receiver [7]. By resolving these problems, we can provide more reliable video streaming for urban VANET.

1.2 Research objective

In this thesis, a road-based resilient video streaming scheme (RVS) that integrates flexible macroblock ordering (FMO) and vehicle moving similarity is proposed, which focuses on enhancing video streaming quality at the receiver by recovering lost packets with error resilience and choosing more reliable routing paths to make inter-vehicle data transmissions more reliable.

1.3 Thesis organization

The rest of this thesis is organized as follows. We describe the background in Chapter 2. In Chapter 3, the related work is introduced. In Chapter 4, we detail our RVS scheme. Simulation results are discussed in Chapter 5. In Chapter 6, we give concluding remarks and outline future work.

Chapter 2

Background

2.1 Vehicle moving similarity

The vehicle moving similarity (VMS) method [5] is to select nearby vehicles having similar velocities with the source vehicle as relay nodes. Vehicle persistence score (VPS) is used to reflect vehicle moving similarity. A vehicle with a high VPS implies that it has stayed long with the sender, and it is a better relay node in an inter-vehicle transmission path.

2.1.1 Hello messages exchange

In order to choose more reliable neighbors, information of neighbors within the transmission range needs to be maintained. Thus, vehicles periodically send HELLO messages to their one-hop neighbors. The neighbor information is recorded in a neighbor list. When a node receives a HELLO message, it refreshes or adds the neighbor information of the sender to the neighbor list and the routing table. Figure 1 illustrates HELLO messages exchange between vehicles. In [5], a new field *position* is added to the original HELLO message, where *position* is the GPS coordinate (x, y) of a vehicle.

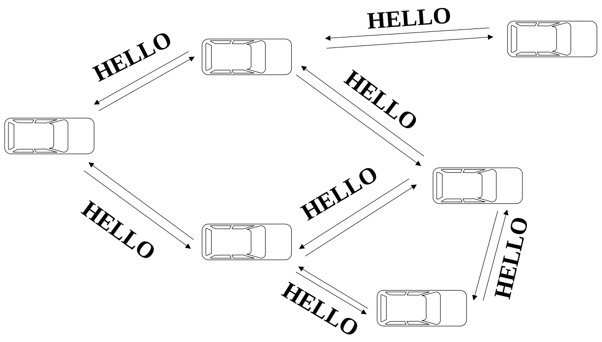


Figure 1. HELLO messages exchange between vehicles.

2.1.2 Vehicle persistence score

In [5], the speed of neighbors is not recorded. Instead, vehicle persistence score (VPS) mechanism is used to choose relay nodes. When a vehicle receives a HELLO message from a neighbor vehicle, the VPS table is updated.

The original format of each entry in the VPS table is

<*ID*, *position*, *distance*, *type*, *VPS* >, where

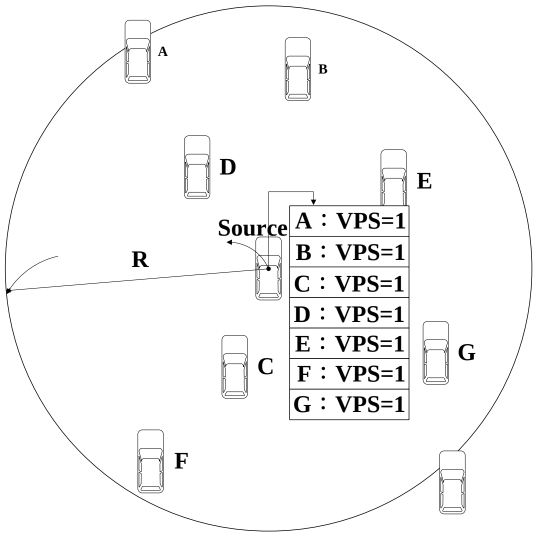
- *ID*: a neighbor's ID.
- *Position***:** the GPS coordinate (x, y) which stands for the position of a vehicle. We use this information to determine the distance between two vehicles.
- *Distance*: the distance between a vehicle and the neighbor.
- *Type*: the type of this neighbor.
- *VPS***:** the value used to reflect a vehicle's stability; Vehicles use this parameter to select rebroadcast nodes.

In this thesis, the format of each entry in the VPS table is modified and some fields are added for the selecting of relay nodes. The modified format of each entry in the VPS table is <*ID, position, block, direction, VPS*>, where only block and direction are two new fields:

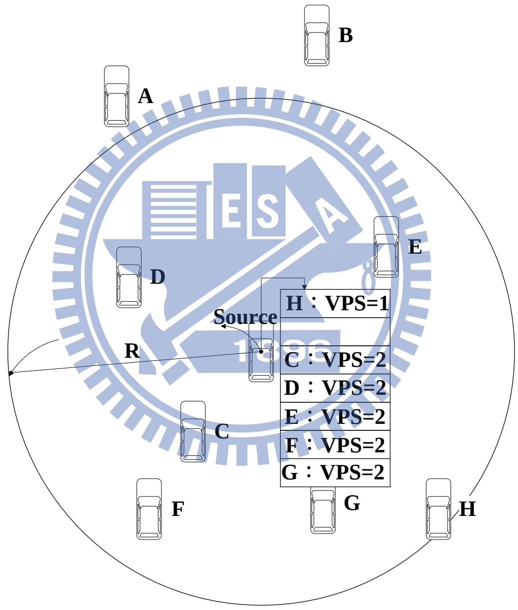
- *Block*: the block identifier that the neighbor is located.
- *Direction*: the neighbor's moving direction.

How the VPS in a vehicle maintained is described in the following. When a vehicle receives a HELLO message from a neighbor, it searches the VPS table. If the neighbor's identifier cannot be found in the VPS table, the vehicle adds the neighbor's related information to the VPS table, and initializes the neighbor's VPS to 1. If the neighbor's identifier can be found in the VPS table, the vehicle increases the neighbor's VPS by 1.

An example of VPS maintenance is showed in Figure 2. In Figure 2(a), the source receives HELLO messages from vehicles A, B, C, D, E, F and G for the first time, so the VPS's of these vehicles are initialized to 1. Figure 2(b) shows the VPS values change after the second update. Vehicles C, D, E, F and G still stay in the transmission range and the source receives their HELLO messages, so their VPS's are incremented to 2. Vehicles A and B are not in the transmission range anymore so they are removed from the VPS table. A new vehicle H is added to the neighbor list and its VPS is initialized to 1. According to the VPS information, it indicates that vehicles with higher VPS tend to stay in the source's transmission range longer. With the VPS information, we can select a stable vehicle to be a relay node.



(a) VPSs are initialized when receiving HELLO messages for the first time.



(b) VPSs are updated when receiving a second HELLO message.

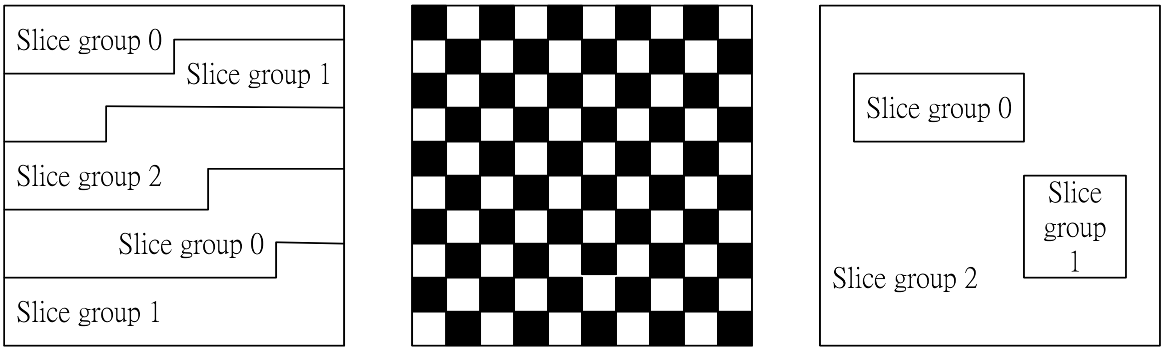
Figure 2. Initialization and update of VPS when receiving HELLO messages.

2.2 Flexible macroblock ordering (FMO)

H.264/MPEG-4 AVC is a new standard for digital video compression jointly developed by ITU-T's Video Coding Experts Group (VCEG) and ISO/IEC's Moving Picture Experts Group (MPEG). The H.264/AVC specifications define a set of error resilience techniques. FMO [6] is one of error resilience techniques defined by the H.264/AVC specifications.

In FMO, an image is divided into slice groups, and each slice group can be divided into several slices, consisting of a sequence of macroblocks. If a slice is lost during transmission, it is possible to reconstruct the lost blocks with the information of the neighboring blocks. The power of FMO depends on how the macroblocks are ordered. In H.264/AVC specifications, there are seven FMO map types, referred to as Type 0 through Type 6. Type 6 is the most general type and allows full flexibility to the user. The others use specific pattern rules. In Figure 3, three types of FMO are illustrated, which are described as follows:

- **Type 0**: It uses run lengths for slice groups which are applied consecutively until the map is complete. Therefore, the run lengths are needed to rebuild the image on the decoder side.
- **Type 1**: It is also known as scattered slices which uses a mathematical function to spread the macroblocks. It is one common case that macroblocks are spread forming a chess board.
- **Type 2**: It is used to mark rectangular areas, so-called *regions of interest*. In this type, the coordinates top-left and bottom-right of the rectangles are saved.
- **(Types 3-5 (not shown in Figure 3)**: They are dynamic types that let the slice groups grow and shrink over the different pictures in a cyclic way. Only the growth rate, the direction and the position in the cycle have to be known.)



(a) Type 0 (b) Type 1 (c) Type 2

Figure 3. FMO type (illustration of only three types).

Chapter 3

Related Work

3.1 F-AODV [7]

In this work, the capability of H.264 codec Flexible Macroblock Ordering (FMO) with receiver error concealment was demonstrated to be capable of streaming good-quality video across a VANET. It showed that the performance of FMO is superior to other error resilience techniques. A common routing protocol, AODV was selected for routing.

Due to the error-prone characteristic of wireless communication, routing packets over multiple hops result in packet loss. There are a variety of error-control methods to protect data transmission. A basic method is the Automatic Repeat Request (ARQ) that uses acknowledgements and timeouts to achieve reliable data transmission over an unreliable service. If the sender does not receive an acknowledgment of a packet before the timeout, it re-transmits the packet until it receives the acknowledgment. It is a useful technique; however, sending an ARQ packet back to the video sender through multi-hops in VANET leads to huge delay which is not tolerable for video streaming applications.

Error resilience methods have better options as they can recover lost data rather than retransmit the data. In forward error correction (FEC), a sender adds redundant data to messages that allows the receiver to detect and correct a limited number of errors without asking the sender for retransmission. However, compared with FMO, FEC needs extra bandwidth for the redundant data. In FMO, compressed frame data are split into a number of slices, consisting of a sequence of macroblocks, which gives a way of reconstructing a frame even if one or more slices are lost.

3.2 RBVT [8]

A number of road-based routing protocols have been proposed. However, some early proposed protocols create routes composed of road segments using the shortest road path between the source and the destination. It is possible that there are no vehicles on the road segments of the shortest path. Some other methods try to use historical data such as average traffic flow. Unfortunately, historical data can not accurately indicate the current road traffic conditions.

In this research, the routing protocol called road-based using vehicular traffic (RBVT) routing was proposed. RBVT protocol uses real-time vehicular traffic information to create paths consisting of road intersections which may have network connectivity among them with higher probability. To reduce the path’s sensitivity to individual node movements, geographical forwarding is chosen to transfer packets between intersections on the path. Simulation results show that it outperforms existing routing protocols in city-based VANETs. Since this work targets at the network layer, it will not be compared with our work, which targets at the application layer.

Chapter 4

Proposed Road-based Resilient Video Streaming Scheme

The goal of the proposed road-based *resilient video streaming* scheme (RVS) is to enhance video streaming quality at the receiver by recovering lost packets with error resilience and choose more stable routing paths to make inter-vehicle data transmissions more reliable. The method can be divided into four phases, *video encoding* phase, *route discovery* phase, *data forwarding* phase, and *video decoding* phase. Figure 4 shows the four phases of the proposed RVS.

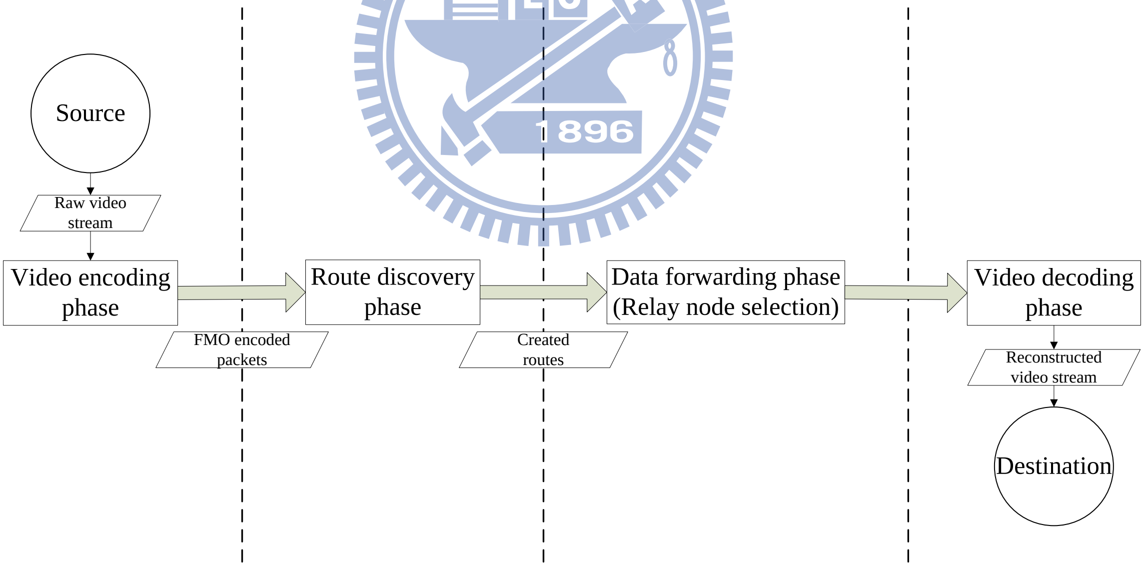


Figure 4. The procedure of proposed RVS.

4.1 Video encoding phase

Before a sender starts transmitting to a receiver, it first encodes the row video stream with FMO type 1, which splits the video streaming into multiple slices for error resilience at the receiver.

4.2 Route discovery phase

Before a sender sends a video stream to a receiver, it first checks if it has routes to the receiver. If there are existing routes, the sender will send FMO type 1 encoded packets through a selected route. If there is no existing route, the sender will broadcast RREQ to neighbors to find the routes to the receiver. Figure 5 shows the procedure of a sender handling a video request.

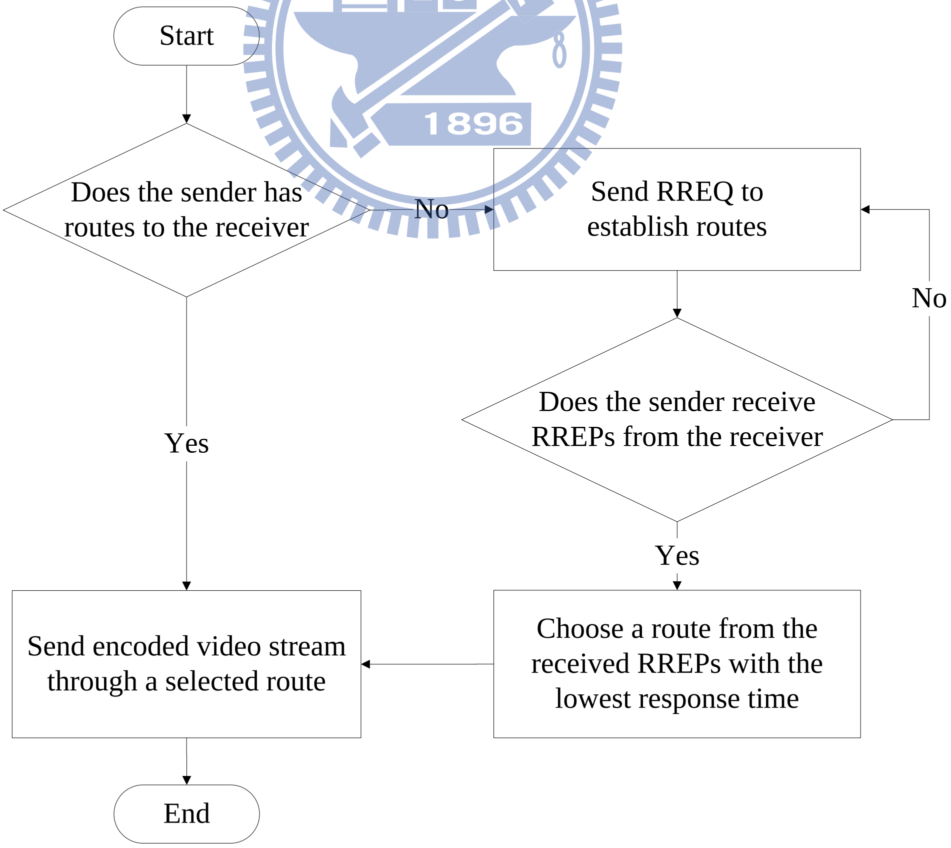


Figure 5. Procedure of a sender handling a video request.

When a node receives an RREQ, it first checks if it is the first time to receive the RREQ. If it is the first time, it then checks if the block number that the node locates has been added in the packet header. If the block number is already in the RREQ, which means it is not the first node in the block relaying the request, it just broadcasts the RREQ to others. If the block number is not in the RREQ, which means it is the first node in the block relaying the request, it adds the block number in the RREQ and then broadcasts it. Figure 6 shows the procedure of node handling RREQ request.

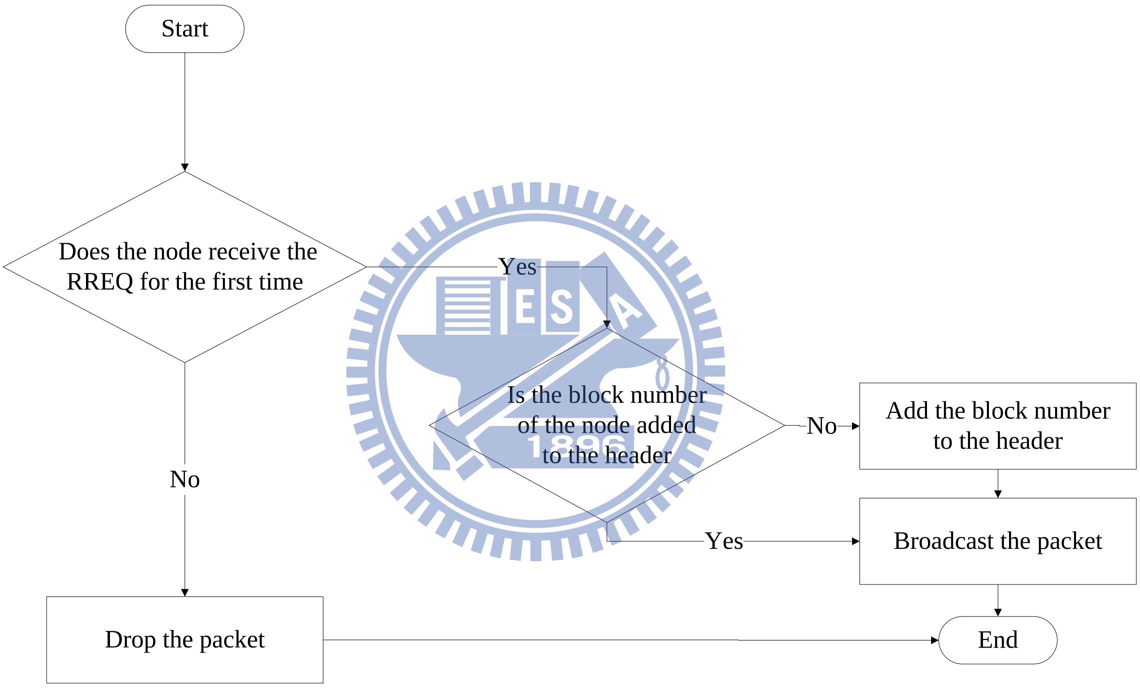
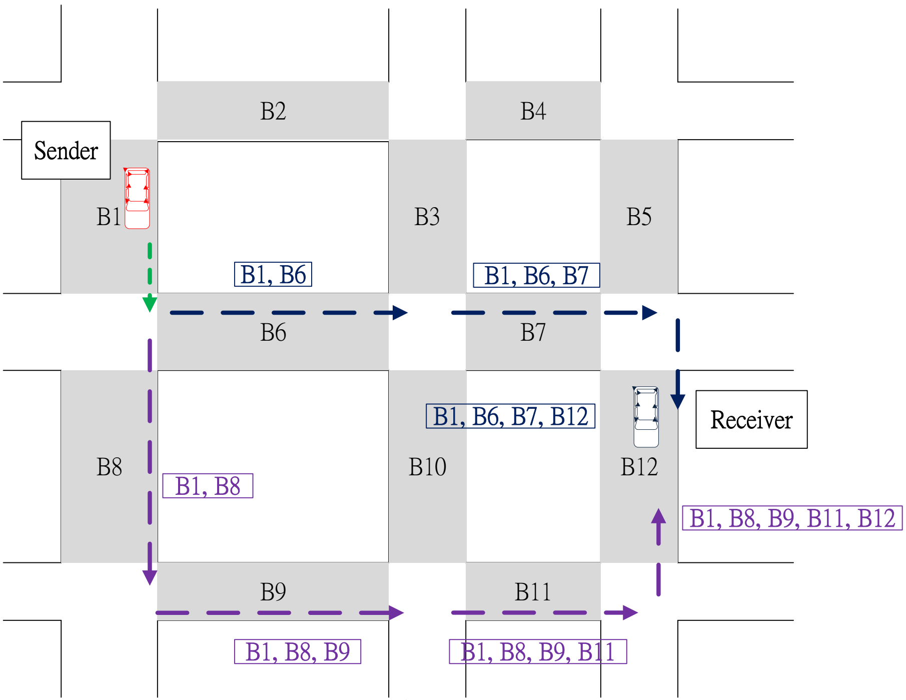
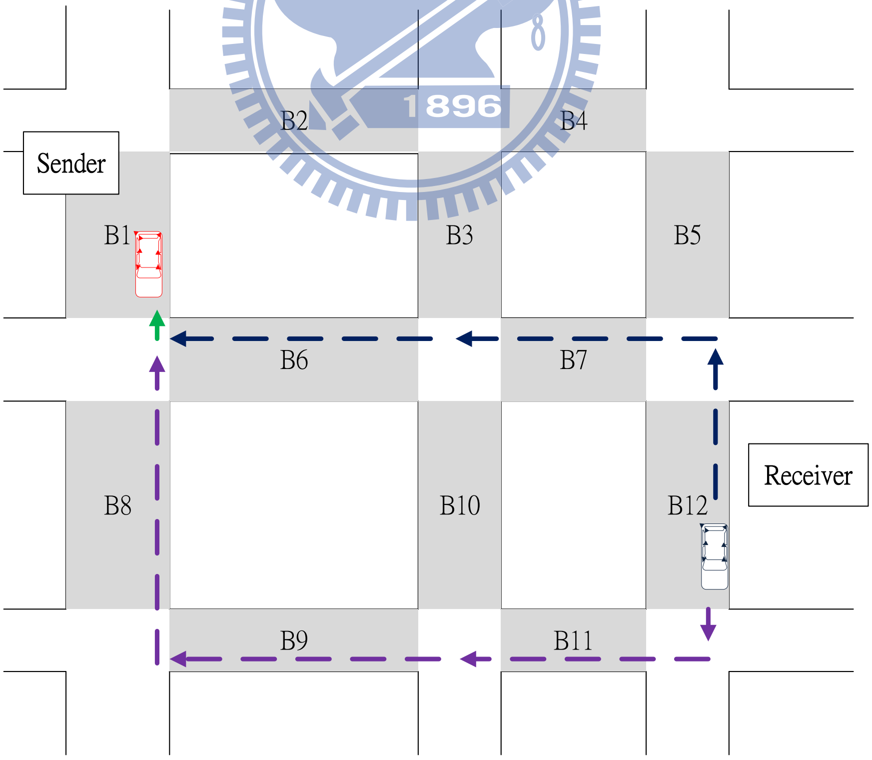


Figure 6. Procedure of a node handling RREQ request.

When the receiver receives an RREQ, it would send an RREP to each node from which it received an RREQ. Figure 7 shows the creation of routes from sender to receiver. In Figure 7(a), block identifiers that the RREQ request passed through are recorded in the header. The receiver uses the block identifiers in the RREQ header to create a path to the sender. The block identifiers are added in the RREP header and sent back to the sender. Figure 7(b) shows RREP returned based on the reverse block identifiers to the sender.



(a) Block identifiers recorded in the RREQ header.



(b) RREP returned based on the reverse block identifiers.

Figure 7. The creation of routes from sender to receiver.

4.3 Data forwarding phase

In the data forwarding phase, the most important issue is the selection of relay nodes. Relay nodes are selected from the VPS table according to the data stored in the VPS table. Fields in the VPS table used for selection are:

- *Block*: used to choose a relay node located in the next block in the header.
- *Direction*: used to choose a relay node with the moving direction that will move toward the receiver.
- *VPS*: a node with a high VPS may be selected.

Figure 8 shows the procedure of relay node selection. When a node has a packet and wants to choose relay nodes to forward, it enters the relay node selection procedure. Firstly, it selects a neighbor based on the VPS table and checks if the neighbor’s Block is equal to next block identifier in the header. If the neighbor’s Block is equal to next block identifier in the header, it then checks if the direction of the neighbor is toward the receiver. If the direction of the neighbor is toward the receiver, the neighbor is selected as a relay node and is added to the relay node list. If all the neighbors in the VPS table are checked, it will enter the next part. Secondly, all neighbors in the VPS table are sorted by VPSs. Then, the neighbor with a higher VPS is selected and is added to the relay node list. If all VPSs of the neighbors in the VPS table are checked, it will exit the procedure.

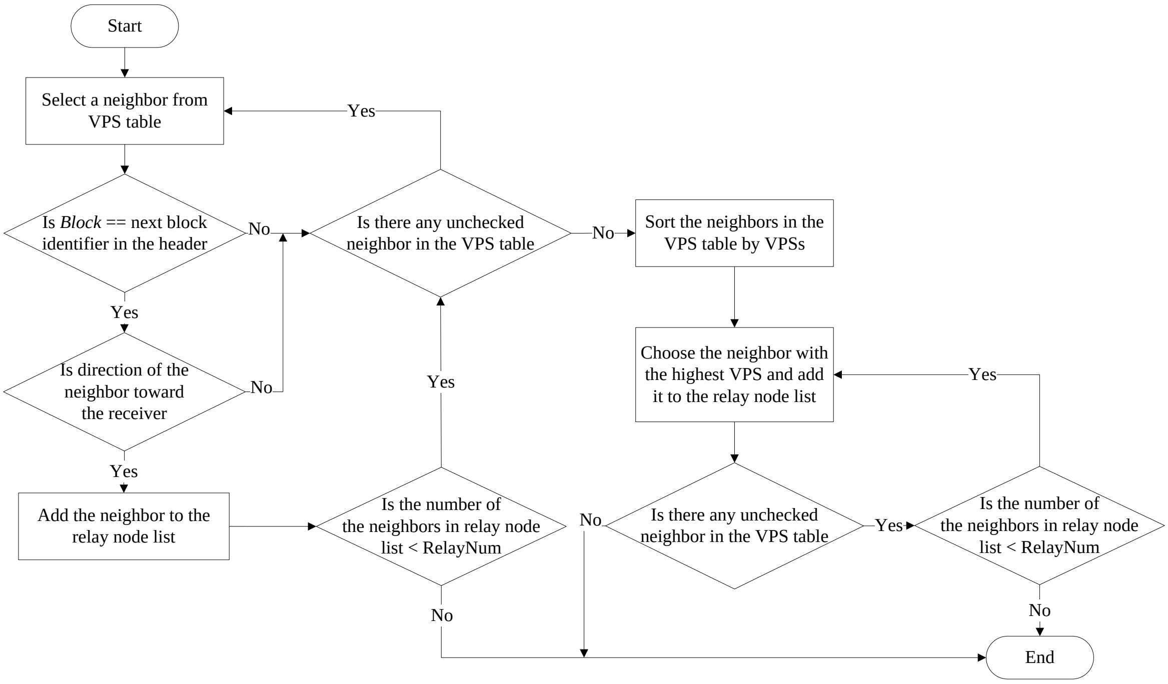


Figure 8. The procedure of relay node selection.

4.4 Video decoding phase

Once a receiver receives encoded packets transmitted from a sender, it decodes the received packets with FMO, recovers lost packets during transmission, and creates reconstructed video stream.

Chapter 5

Simulation and Discussion

In this chapter, we first describe simulation setup and evaluation metrics. Then, we compare the proposed RVS with F-AODV.

5.1 Simulation setup

We consider a urban scenario to evaluate the performance of the proposed RVS. The scenario is 1 *km* *1 *km*, and the width of each lane is 5 *m*. The simulation was done in NS2.29 [10]. The simulation parameters are shown in Table 1. The simulation results were obtained from the average of five simulation runs. To evaluate the performance of the proposed RVS, we compare our approach RVS with a classical ad hoc protocol, F-AODV, in terms of delivery ratio and peak signal-to-noise ratio (PSNR).

- *Packet delivery ratio*: It is the number of data packets successfully delivered to the receiver divided by the total number of data packets generated by the sender.

$$PacketDeliveryRatio = \frac{Successfully\ received\ data\ packets}{Total\ number\ of\ data\ packets generated}$$

- *Peak signal-to-noise ratio (PSNR)*: PSNR is the most commonly used parameter as a measure of video stream quality. The calculation formula is as below.

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left[I(i,j)-K(i,j)\right]^2$$

where *m*, *n* are the height and width of a frame and *I* is the original frame and *K* is the reconstructed frame.

$$PSNR = 10\times\log_{10}\left(\frac{MAX_I^2}{MSE}\right)$$

where MAX_I is the maximum possible pixel value of the image. If the pixels are represented using 8 bits per sample, MAX_I is 255.

Table 1. Simulation settings for NS2 [10].

Parameter	**Value**
Transmission range	250 *m*
MAC Protocol	IEEE 802.11
Network area	1000 *m* x 1000 *m*
Lane width	5 *m*
Number of vehicles	30, 40, 50, 60, 70
Connection type	CBR
Data packet size	1000 bytes
Mobility model	VanetMobiSim
Simulation time	100 *s*

We choose the VanetMobiSim [11] mobility model to generate mobility traces for our simulation. In the simulations, a 1000 m^2 area was defined and vehicles were initially randomly placed in the area. Time interval between traffic lights change was set to be 10 sec. The number of vehicles range from 30 to 70 nodes. The detail settings are given in Table 2.

Table 2. Simulation settings for VanetMobiSim [11].

Parameter	**Value**
Terrain dimension	1000 m^2
Max. traffic lights	6
Time interval between traffic lights change	10 *s*
Number of lanes	2
Nodes (vehicles)	30, 40, 50, 60, 70
Min. speed	3.2 *m/s* (7 *mph*)
Max. speed	13.5 *m/s* (30 *mph*)
Length of vehicle	5 *m*
Max. acceleration	0.6 m/s^2
Normal deceleration	0.5 m/s^2

We used H.264/AVC reference software [12] to implement the FMO encoding and decoding procedure. The Foreman film was selected as a test film and was encoded at QCIF resolution with 4:2:0 sampling. The frame rate of the video stream was set at the rate of 15 Hz. The detailed settings are given in Table 3.

Table 3. Simulation settings for H.264/AVC reference software [12].

Parameter	**Value**
Test film	Foreman
Resolution	QCIF (176 x 144)
Sampling	4:2:0
Frame rate of the video stream	15 Hz
Profile	Baseline
GOP (group of pictures) structure	15 pictures
Number of slice groups	2
FMO type	Dispersed (type 1)

5.2 Simulation results and discussion

In Figure 9, we compare the delivery ratio under different node numbers between F-AODV and RVS. We compare these two protocols in the same scenario with a different number of vehicle nodes varying from 30 to 70 nodes. Simulation results show that the proposed RVS improves the delivery ratio by 7% compared to F-AODV.

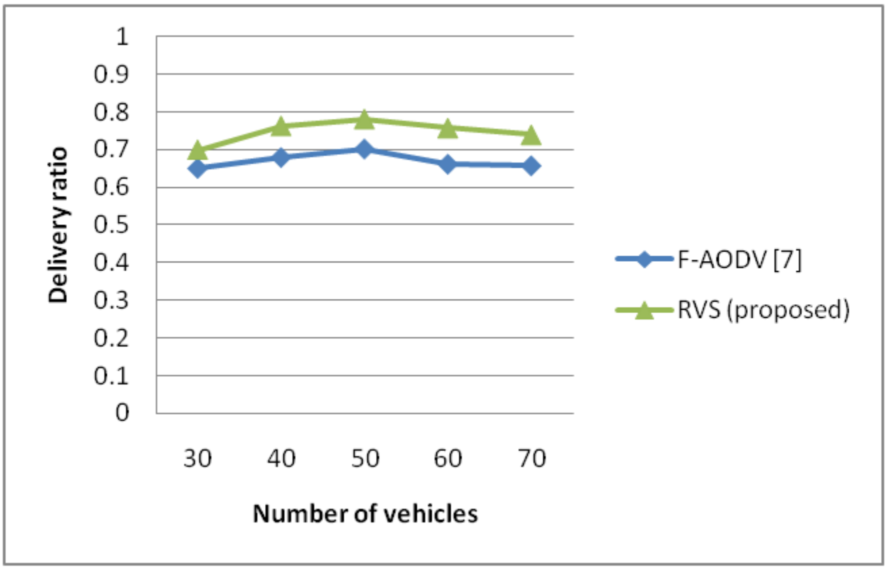


Figure 9. Delivery ratio under a different number of vehicles.

In Figure 10, we compare the PSNR under a different node number of vehicles between F-AODV and RVS. We compare these two protocols in the same scenario with a different number of vehicle nodes varying from 30 to 70 nodes. The proposed RVS improves the PSNR by 1.3 dB compared with F-AODV.

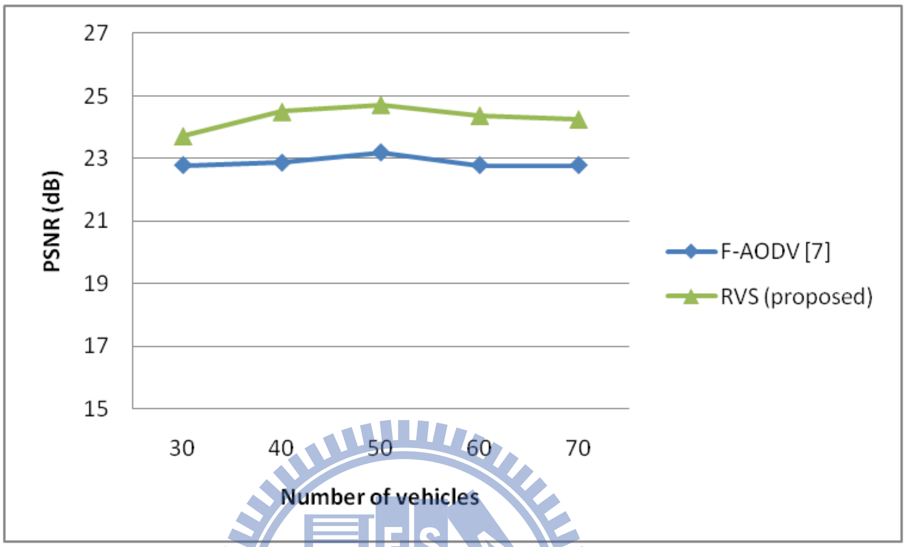


Figure 10. PSNR under a different number of vehicles.

Chapter 6

Conclusion

6.1 Concluding remarks

In this thesis, we have presented a resilient video streaming scheme (RVS) to enhance video streaming for Urban VANETs. RVS integrates flexible macroblock ordering (FMO) and vehicle moving similarity. It can enhance video streaming quality at the receiver with error resilience and choose more stable routing paths to make inter-vehicle data transmissions more reliable. FMO encodes images to provide error resilience at the receiver. Vehicle moving similarity selects more stable neighbors to make inter-vehicle data transmissions more reliable. The proposed RVS improves the delivery ratio by 7% and the received video stream quality (in terms of PSNR) by 1.3 dB compared with F-AODV. That is, simulation results show that the proposed RVS performs well in city environments.

6.2 Future work

We may make RVSS be able to establish multiple paths and transfer encoded video stream through those paths to provide a more reliable inter-vehicle transmission environment. Moreover, we may take the available bandwidth of each neighbor into consideration to choose more suitable neighbor and provide better performance.

Bibliography

[1] H. Kawashima, “Japanese perspective of driver information systems,” *Transportation*, vol. 17, no. 3, Sept. 1990, pp. 263–84.

[2] H. Hartenstein and K. P. Laberteaux, “A tutorial survey on vehicular ad hoc networks,” *IEEE Commun. Mag*., vol. 46, no. 6, pp. 164–171, Jun. 2008.

[3] C. D. Wang and J. P. Thompson, “Apparatus and method for motion detection and tracking of objects in a region for collision avoidance utilizing a real-time adaptive probabilistic neural network,“ U.S. patent no. 5,613,039, 1997.

[4] Y. Toor, P. Muhlethaler, and A. Laouiti, “Vehicle ad hoc networks: applications and related technical issues,” *IEEE Commun. Surveys & Tutorials*, vol. 10, no. 3, pp. 74-88, 2008.

[5] M.H. Wei, K.C. Wang, and I.L. Hsieh “A reliable routing scheme based on vehicle moving similarity for VANETs,” in Proc. *IEEE TENCON*, 2011.

[6] S. Wenger, “H.264/AVC over IP,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 645-656, 2003.

[7] N. Qadri, M. Altaf, M. Fleury, M. Ghanbari, and H. Sammak, “Robust video streaming over an urban VANET,” in *Proc. IEEE International Conference on Wireless and Mobile Computing, Networking and Communication*, pp. 429–434, 2009.

[8] J. Nzouonta, N. Rajgure, A. Guiling Wang, and C. Borcea, “VANET routing on city roads using real-time vehicular traffic information ,” *IEEE Trans. Veh. Commun.* pp. 3609 - 3626, 2009.

[9] N. N. Qadri, M. Fleury, and M. Ghanbari, "Approaching P2P communication in a vehicular ad hoc network," in *Proc. IEEE 34th Conference on Local Computer Networks*, pp. 695-701 2009.

[10] "The network simulator (NS2)," [Online]. Available: http://www.isi.edu/nsnam/ns/.

[11] M. Fiore, J. Härri, F. Filali, and C. Bonnet, "Vehicular mobility simulation for VANETs," in *Proc. 40th Annual Simulation Symp.*, Mar. 2007, pp. 301-307.

[12] "H.264/AVC Software," [Online]. Available: http://iphome.hhi.de/suehring/tml/.

[13] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," Network Working Group, RFC 3561, 2003.

[14] M. Sun, W. Feng, T. H. Lai, K. Yamada, H. Okada, and K. Fujimura, "GPS-based message broadcasting for inter-vehicle communication," In *Proc. International Conference on Parallel Processing*, 2000, pp. 2685-2692.

[15] J. Jakubiak and Y. Koucheryavy, "State of the art and research challenges for VANETs," In *Proc. Consumer Communications and Networking Conference*, pp.912-916, 2008.