社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2007－DPS－130 (28)
2007－CSEC－ 36 (28)
2007／3／1

Car-to-Car Communication System Using a Dual Channel Mobile Ad-Hoc Network

de Silva Heethaka Pradeep Ruwantha[†], Hisashi Kawase[†], Akira Iwata[†], Kimitake Wakayama[††], Hidekazu Umeda[†††]

[†] Nagoya Institute of Technology.
[††] Nagoya University of Foreign Studies.
[†††] SKYLEY NETWORKS.
e-mail: ray@mars.elcom.nitech.ac.jp, iwata@nitech.ac.jp

Abstract: We propose a new Car-to-Car communication system using a Dual Channel Mobile Ad-Hoc network. In the proposed system, Car-to-Car communications will be divided in to 2 categories. Normal mode transmissions and Emergency mode transmissions. Emergency mode transmissions will be done using a dedicated channel thus overcoming packet loss due to packet collisions. The proposed system will also adopt PKI technology to prevent unauthorized access thus making the system more reliable. We report that packet routing for up to 9 hops was possible within 50miliseconds. We also report that adoption of PKI technology would not impair the system performance.

2チャンネル・モバイル・アドホック・ネットワークを用いた車車間通信システムの提案と検証

デシルワ・ヒータカ・プラディープ・ルワンタ[†] 川瀬 悠[†] 岩田 彰[†]
若山 公威[††] 梅田 英和[†††]

[†]名古屋工業大学
[††]名古屋外国語大学
[†††](株)スカイリー・ネットワークス

概要:新型車車間通信方法として異なる無線LANチャンネル二つを用い通信を行うモバイル・アドホック・ネットワークを使用するシステムを提案する。システムの通信を通常モード通信と緊急モード通信の二種類に分ける。通常モード通信と緊急モード通信のため異なる無線LANチャンネル二つを用いることにより、緊急情報のブロードキャスト時のパケットコリジョンを防ぎ効率よく通信できる。またPKI技術を使用する事により信頼度の高いシステムにする。この提案システムを検証するため実験を行い、メッセージロス率、通信遅延等の問題点、及び解決方法を考察する。

1 Introduction

When considering Car-to-Car Communication methods, Mobile Ad-Hoc Networks(*MANETs*) appear to be the obvious solution. Mobile Ad-Hoc Networks do not require any roadside infrastructure to route the packets which makes it a very economical solution. Moreover this makes a Car-to-Car Ad-Hoc Network easily adoptable.

A Car-to-Car Mobile Ad-Hoc Network will enable the vehicles in the network to exchange information such as traffic congestion data, weather information like rain, snow, and road surface freezing and a system like this can also warn vehicles about other vehicles approaching them from blind corners. A Car-to-Car network can also be used to warn the vehicles about accidents, breakdowns and sudden traffic blocks helping to prevent pileup accidents making the road a safer place. Unlike the other information exchanged in the network, warnings like these need to be transmitted to all the nodes in the network with high priority. Broadcasting can be used in such a scenario.

Many Car-to-Car MANET systems have been proposed in the recent years. For instance Peter Davis et al.[1] suggest a single channel Car-to-Car

MANET which will flood emergency data. For this solution they carried out an experiment utilizing a 50 node single channel Ad-Hoc network with 20-25 nodes within 1 hop range and a maximum hop count of 5.

We had the opportunity to conduct an experiment by creating a MANET of 130 nodes at the EXPO 2005 AICHI, JAPAN. The data collected during this experiment showed that packet collision can occur in a high node density Mobile Ad-Hoc Network[2]. So it can be predicted that packet collision will become a factor for a Car-to-Car Ad-Hoc networks, increasing packet loss and bringing down the transmission speeds.

In this paper we propose a Dual Channel Mobile Ad-Hoc network to prevent the packet loss due to collision during emergency message broadcasts. This System will adopt Public Key Infrastructure (*PKI*) technology making the transmissions tamper proof. We also conduct evaluation experiments to verify the loss rates in common channel broadcastings and dedicated channel broadcastings, to measure the transmission time up to 9 hops and to measure the loss rate at the 9th hop.

2 Proposed System

In the proposed system, each vehicle would have two separate Wireless LAN(*WLAN*) adaptors. One of these adaptors will be used for the normal mode communications. The other adaptor will be used for Emergency mode transmissions and will use a different dedicated channel for the transmissions as shown in Fig 1. "Dual Mode DECENTRA Advanced" developed by SKYLEY NETWORKS [3]was used as the Ad-Hoc routing middleware for this work.

2.1 Dual Mode DECENTRA Advanced

Dual Mode DECENTRA Advanced is an Ad-Hoc routing middleware which creates a Dual Channel Mobile Ad-Hoc Network. Dual Mode DECENTRA Advanced is a Hybrid routing protocol. It has a parameter called "Scope" and packets will be routed to the nodes within the scope specified number of hops using proactive routing and routing beyond the scope value number of hops will be done using reactive routing.

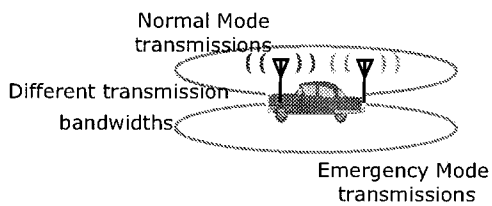

Figure 1: Transmission model

2.2 Normal Mode Transmissions

Each node will transmit hello messages in the normal mode channel to advertise their presence and create an Ad-Hoc network. After entering the network each node will then exchange the link_state data they have. These data will be used in the packet routing process. The normal mode transmissions Which consists of traffic data, weather data etc. will be exchanged using normal mode transmissions.

All the transmissions in this channel will be encoded with PKI technology. This is to prevent the data from being accessed by third party nodes.

2.3 Emergency Mode Transmissions

This mode is used to transmit the emergency warnings when ever they are generated. These transmissions will be done using a dedicated channel. Emergency mode does not broadcast hello messages. It assumes the network topology of the normal mode transmissions.

There are two methods of transmissions available in the emergency mode. One is the broadcast method where each node broadcasts the packet to all the nodes within 1 hop range. Each receiving node will re-broadcast the packet. The other method is a unicast based broadcasting (*unicastXn*) where the packet will be transmitted to all the nodes within 1 hop range one node at a time. This way if there are 4 nodes within 1 hop range the sending node will have to transmit the same message 4 times instead of the one time broadcasting. The link_state data from the normal mode will be used to identify the nodes within 1 hop range.

In order to increase the reliability of the system each emergency data packet will be retransmitted 5 times. All the emergency data packets generated will be digitally certified by the node who generated them, to prevent foul play before broadcasting.

2.4 Security

In order to prevent any unauthorized accessing and alterations, this system use PKI technology. A digital certificate will be issued to each node by a certification entity. The nodes will exchange their cer-

tificate with the other nodes in the network. When ever a node (Node A) receives link_state data containing a new node ID (Node C) from another node (Node B) Node A will request the certificate of Node C from Node B. If node B does not have Node C's certificate then Node A will request it from the next node who send link_state data contain the node ID C. Meanwhile Node B will be requesting the node C's certificate from the node which sent link_state data containing node C's ID.

A Certificate Revocation List (*CRL*) released by the issuing authority of the certificates will be downloaded when ever the node has access to roadside infrastructure like Electronic Toll Collection (*ETC*) sensors or Dedicated Short Range Communications (*DSRC*) transmitters. Each node will periodically check the validity of the certificates it has by comparing them with the CRL . The certificates collected will be deleted once a fixed time has passed since the last interaction with the corresponding node.

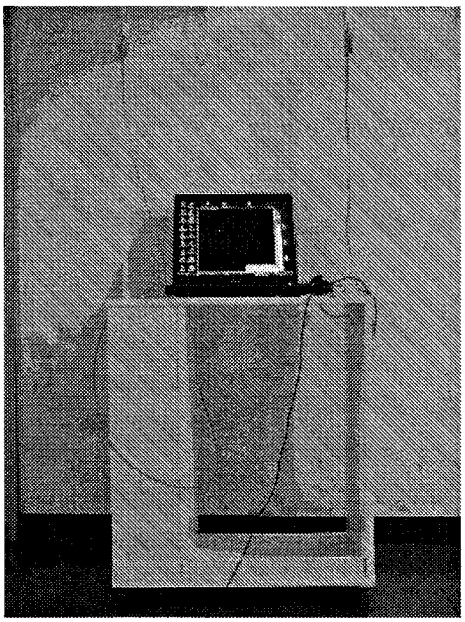

Figure 2: The experiment set up.

3 EXPERIMENTS

This section will provide a detailed description of the verification experiments we conducted.

1. ***Experiment 1*** was conducted to measure the normal mode packet loss rates at 9 hops and the transmission times for 7, 8 and 9 hops, in a 10 node 9 hop MANET.

2. ***Experiment 2*** was conducted to measure the loss rates and transmission times at the 10th node for emergency mode transmissions. The same network as Experiment 1 was used for this experiment.

3. ***Experiment 3*** was conducted to verify the certificate distribution system. This experiment was repeated at different the Advertise_rate (hello packet interval) and Refresh_rate (link_state packet interval) parameters to see their effect on the certificate distribution times.

4. ***Experiment 4*** was conducted to verify that encoding or digitally certifying the packets would not affect the system performance.

3.1 Experiment setup

We used up to 10 notebook computers (Epson endeavor NT2850, Windows XP SP2) and 20 USB WLAN adaptors (Buffalo WLI-U2-KG54) for these experiments. Each notebook computer was placed on a wooden trolley 70cm high and the 2 WLAN adaptors were placed 75cm away from the computer as shown in Fig 2 Dual Mode DECENTRA Advanced was used as the middleware implementing the Dual Channel MANET. The application used for the experiments was developed on VC++2005 express edition. AiCrypto [4] a cryptographic library developed by Akira Iwata Laboratory of Nagoya Institute of Technology was used as the cryptographic library for these experiments.

3.2 Experiment 1

10 nodes were placed inside the Nagoya Institute of Technology (*NIT*) campus as shown in Fig 3. Using buildings as barriers to make sure that each node could only transmit with the 2 nodes on its either sides, we created a 9 hop Ad-Hoc network. Then 100 packets with a payload of 1024 Bytes were transmitted from node 1 to node 10 at 2000ms intervals. The times taken by the packets to reach node 10 and the number of packets lost en route (loss rate) was calculated. This procedure was repeated 5 times. Table 1 shows the loss rates at the 10th node. All the transmissions were done with plain text data packets.

As shown in Table 1 . Nearly 50% of the packets are lost by the time they reach the 10th node. The Ad-Hoc Network used for this experiment has only

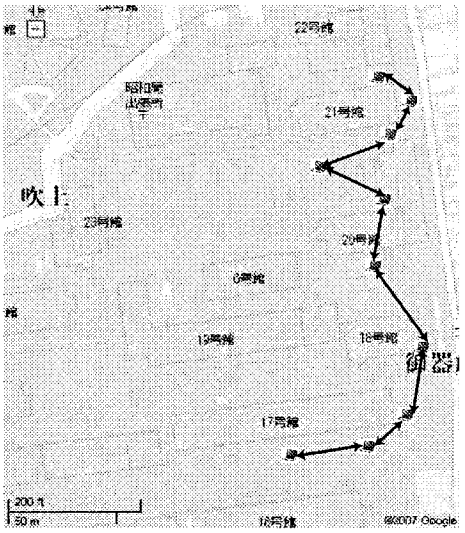

Figure 3: Node distribution for experiment 1 and 2.

Table 1: Loss rate at the 10th node.

Experiment no	1	2	3	4	5	Avg
Loss Rate %	43	47	54	25	57	45.2

one possible route from node 1 to node 10 and this plays a significant part in the high packet loss rate. In order to achieve 9 hops with 10 nodes we had to keep them at the absolute edges of their wireless ranges making the network very unstable. A detailed inspection of the collected data showed us that most of the lost packets were consecutive ones, pointing towards an unstable network.

Fig 4 shows the transmission times for 7, 8, and 9 hops. The results show that 7 hops can be made in around 32ms-35ms, 8 hops in 36ms-40ms and 9 hops in 42ms-46ms.

3.3 Experiment 2

Using the same node distribution as in Fig 3. We measured Emergency mode transmission times and loss rates at the 10th node. We transmitted 100 packets with a payload of 1024 Bytes from node 1 to node 10 using broadcasting, and another 100 packets were transmitted using the unicastXn method. Each packet was re-transmitted 5 times and each one of these packets were broadcasted with a 2000ms interval. All the transmissions were done with plain text data packets.

The same experiment was repeated thrice. First

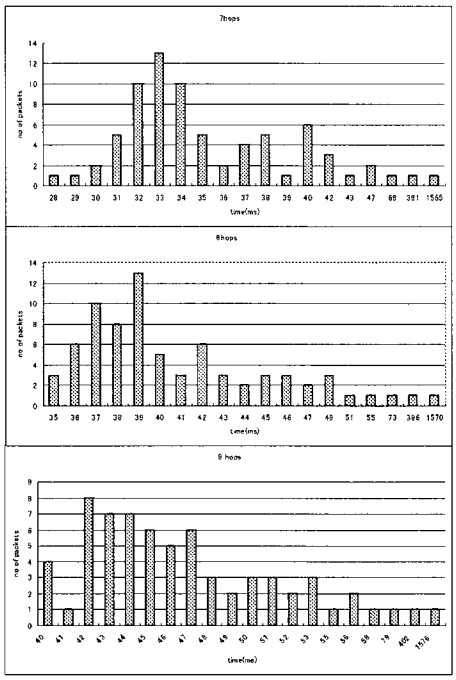

Figure 4: Transmission times for 7, 8, and 9 hops.

each of the 100 packets were retransmitted 5 times consecutively (0ms interval) then the packet re-transmitting interval was increased form 0ms to 5ms. Next the packet re-transmitting interval was increased to 10ms. The loss rates and transmission times for all 3 different re-transmission times were compared.

Table 2: Emergency mode Loss rate.

Re-transmitting interval(ms)	0	5	10
Broadcasting loss rate %	49	45	82
UnicastXn loss rate %	41	37	28

Table 2 shows the loss rates for emergency mode transmissions. The loss rates in broadcasting is quite high. Again the obvious reason here is the availability of only one possible route for the packets. In a lean network like this unicastXn method shows more promise with a lower loss rate. That is because Unicast packets are re-transmitted at the network level in the case of a packet loss. However unicastXn has its drawbacks. When we look at the transmission times in Fig 5 we can see that the unicastXn method requires much longer transmission times than the broadcast method. This can be

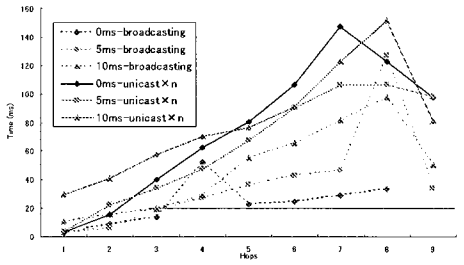

Figure 5: Emergency mode transmission times.

explained by the fact that unicastXn method is a 1-to-1 type transmission which does re-transmissions in the case of lost packets. However broadcasting is a 1-to-many type transmission where no re-transmissions are done. This difference in routing makes the unicastXn method more time consuming than the broadcast method.

3.4 Experiment 3

In this experiment the time taken for the certificate distribution was measured. 4 nodes were placed on the 5th floor of Building number 20 of NIT as shown in Fig 6. At start, each node had their own certificates only. The nodes were switched on from node 1 to node 5 and in each case the time taken for

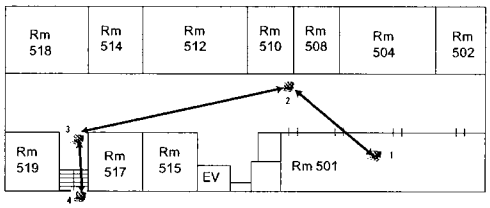

Figure 6: Node distribution for experiment 3.

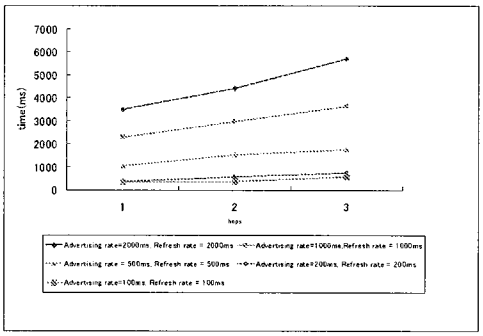

Figure 7: Certificate distribution times.

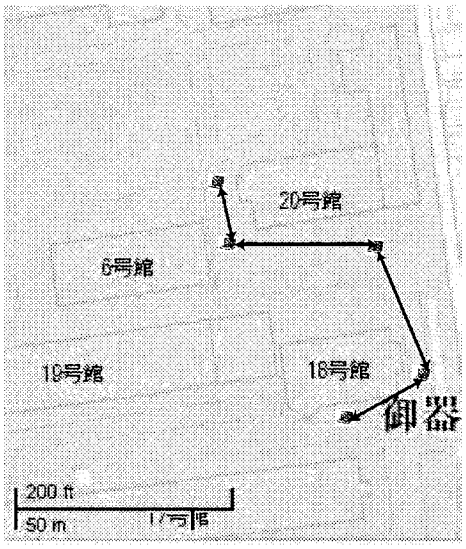

Figure 8: Node distribution for experiment 4.

the certificate to reach node 1 was measured. The same experiment was repeated at Advertise_rate (hello packet interval) and Refresh_rate (link_state packet interval) values of 2000ms, 1000ms, 500ms, 200ms and 100ms.

Fig 7 shows the distribution times for the certificates. We can see that the certificate transmission time is a function of the Advertise_rate and Refresh_rate. Also the Certificate distribution times can be brought down by reducing the Advertise_rate and Refresh_rate.

3.5 Experiment 4

Five nodes were placed in locations as shown in Fig 8. Then 100 packets of 1024 bit plain text data was transmitted from node 1 to node 5. Next the same experiment was repeated with the data of 1024 Bytes encoded at each transmission. The encoded transmission times were measured and compared with plain text transmission times. During the experiment all the nodes had acquired the other 4 node's digital certificates and the data was encoded with the public key of the destination node (node 5) included in the respective digital certificate.

Fig 9 shows the transmission times for encoded transmissions and transmissions with digital signature. We can see that these extra security measures do not have a negative effect on the transmission times.

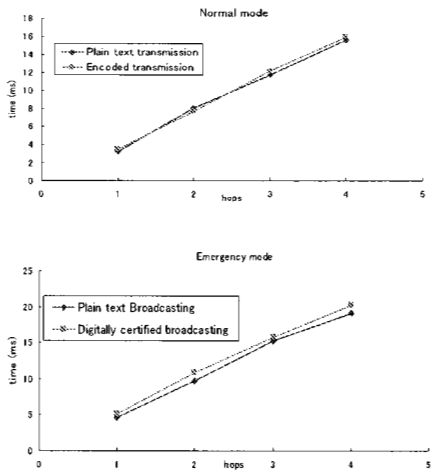

Figure 9: Secure transmission times.

4 Discussions

The experimental analysis we did shows us that we can rout data packets even 9 hops away in less than 50ms. This speed is quite sufficient to warn a motor vehicle on a highway of a imminent danger (A car traveling at 108km/h travels only 3 meters in 100ms). We also established that PKI could be easily adopted in to an Ad-Hoc network and that encrypting or digitally signing data packets do not increase transmission times. However the packet loss rates at 9 hops are still too high for this system. Further experiments with a higher number of nodes providing multiple routs for the packets are needed to verify the expected loss rates in a real situation of Car-to-Car Ad-Hoc network.

5 Conclusions

In conclusion, we proposed a Car-to-Car communication system using a Dual Channel Mobile Ad-Hoc network. We showed that this system can transmit data packets up to 9 hops within 50ms, and that PKI can be adopted into this system. The results so far look very promising for practical implementation in a Car-to-Car network.

6 Acknowledgements

This work was supported by Joint Research Program of DENSO CORPORATION.

References

[1] Peter Davis,Satoko Itaya,Jun Hasegawa, Akiyo Hasegawa, Naoto Kawowaki, Akira Yamaguchi, Sadao Obana
Analysis of Characteristics of Flooding for Inter-Vehicle Communications, IPSJ SIG Notes, 2005-MBL-35, 2005-ITS-23 pp.99-104 November 2005.

[2] Kenji Ito, Kimitake Wakayama, Akira Iwata, Hidekazu Umeda
The Inspection of Large Scale Mobile Ad hoc Network Proof Experiment at EXPO 2005 Aichi,JAPAN, IPSJ SIG Notes, 2006-DPS-126, 2006-CSEC-32, pp.147-152, March 2006.

[3] "SKYLEY NETWORKS"
http://www.skyley.com/

[4] "A.I.Lab Web Site"
http://mars.elcom.nitech.ac.jp/security/aicrypto.html