

Cross-Layer Design for TCP Performance Improvement in Vehicular Communication Networks

Zhaoran Wang*, Xianzhong Xie*, Dingxin Zhao *

** Institute of Broadband Access Networks, Chongqing University of Posts and Telecoms, Chongqing, China*

wzr_123@163.com, xiexianzhong@tom.com, zdx1314@163.com

Abstract—In Vehicular Communication Networks, the nodes move in high speed, the topology changes frequently and the channel quality is not stable, which are considered to be the most important factors leading to the deterioration of TCP performance. In this paper, we first analyze the problems of traditional TCP existed in Vehicular Communication Networks. Subsequently, a four-state transition diagram is given and a new TCP congestion control mechanism based on Cross-layer design is proposed, which can predict the reason of packet loss by providing TCP with helpful route and movement information, and can apply appropriate congestion control strategy under different conditions. The simulation results show that the improved TCP performs better than TCP Reno in Vehicular Communication Networks, especially when the channel quality is poor and the nodes move in high-speed.

Keywords—TCP, Cross-layer design, Vehicular Communication Networks, Four-State transition diagram, Ad Hoc

I. INTRODUCTION

In recent years, the explosive growth of traffic and ubiquitous information service have made communication networks closely integrate into the vehicular networks, so that the Vehicular Communication Networks emerged as the times require and became a research hotspot [1]. Vehicular Communication Networks has two parts: RVC (Road to Vehicle Communications) and IVC (Inter-Vehicle Communication) [2]. In the case of RVC, the vehicle access to the Internet to get network resources through the road infrastructure or mobile base station, while in the case of IVC, vehicles installed of wireless communication equipment can forward and exchange information through building up multi-hop Ad hoc Networks, which named VANET (Vehicular Ad hoc Networks). Since the latter is more widely used, in this paper, we mainly analyse TCP performance improvement in VANET environment.

VANET is a self-organized and open-structured inter-Vehicular Communication Networks, which not only has the characteristics of mobile ad hoc networks, such as dynamic, distributed, multi-hop, etc [3], but also has its own unique attributes, such as high speed of vehicle, frequently topology changes, predictable trajectory and no hardware constraints. These attributes affect the transmission performance in VANET directly [4]. Fortunately, in VANET environment, the trajectory of vehicle can be predicted since the movement of vehicle is limited by road conditions and the vehicle can

obtain useful traffic information conveniently. To make full use of these benefits, more efficient and reliable transmission control protocol can be designed.

There are many information services for vehicle safety in VANET, so it is necessary to choose a reliable transport protocol for these services. TCP (Transmission Control Protocol) is one of the core protocols and could provide reliable, ordered delivery for a stream of bytes in transport layer. However, the research about TCP in VANET is only a few at present [5-8], and improved scheme based on cross-layer design has not appeared yet. References [5] proposed an optimization scheme based on TCP agent, the TCP throughput is discussed in [6], references [7] analysed the impact of transmission power on TCP performance, and in [8], the goals and elements of transmission control design of VANET are proposed according to the applications and characteristics of VANET.

The TCP performance in Vehicular Communication Networks is not ideal because of the impact of route handover and transmission errors, so it is necessary to improve TCP performance in VANET environment. In this paper, we first analysed the characteristics and problems of traditional TCP in VANET environment, subsequently, a four-state transition diagram is given and a new TCP congestion control mechanism based on Cross-layer design is proposed, that is VC-TCP. Since the trajectory is predictable and real-time road information is available, so vehicles can obtain useful route and movement information to predict the reason of packet loss, and for different reasons, vehicles can take different TCP congestion control strategies. Simulation results show that the improved VC-TCP has greatly improved the performance in Vehicular Communication Networks, with a better throughput and latency value.

II. PROBLEMS OF TRADITIONAL TCP IN VEHICULAR COMMUNICATION NETWORKS

Traditional TCP is designed for the wired channel, such as TCP Reno, TCP New Reno, TCP Vegas and so on, which has high reliability, so all the packet losses are considered to be caused by network congestion. However, in VANET environment, the channel is wireless and unreliable and the transmission distance of node is short, while the communication nodes move in high speed and the route handover is frequent, so in addition to link congestion, both the route handover and transmission errors are important

reasons for package loss. If congestion control mechanism is running without correctly distinguish the reasons for packet loss, it will inevitably lead to decline of network performance.

A. The Basic Idea of TCP

TCP is a reliable and connection-oriented standard protocol in transport layer, the core of which is congestion control mechanisms, including slow start, congestion avoidance, fast retransmit and fast recovery. TCP sender confirms the packets send before and monitors network conditions by receiving ACK (Acknowledgement) from TCP receiver, and adjusts sending speed to avoid congestion. In the stage of slow start, when a ACK is received, $cwnd$ (congestion window) is increased by α , while in congestion avoidance, $cwnd$ is increased by $\alpha / cwnd$ for each ACK. As it is shown in (1), α means the window increase rate, $0 < \alpha \leq 1$, and in traditional TCP, $\alpha = 1$. $ssthresh$ is slow start threshold, that is used to determine whether the slow start or congestion avoidance algorithm should be used to control data transmission. When a packet is lost, TCP sender will start the congestion control process, then $ssthresh$ and $cwnd$ will be reset. In traditional TCP, $ssthresh$ is set to $cwnd / 2$, and $cwnd$ is set to 1.

$$cwnd(t + \Delta t) = \begin{cases} cwnd(t) + \alpha, & cwnd(t) < ssthresh \\ cwnd(t) + \frac{\alpha}{cwnd(t)}, & cwnd(t) \geq ssthresh \end{cases} \quad (1)$$

B. The Impact of Route Handoff on TCP

When a route is broken caused by fast-moving of nodes, their neighbour nodes will find link failure and send link failure messages upstream until the source node, then the relay nodes will discard the packet have not sent to the destination node yet. This may result in multiple and successive TCP packet losses. At the sender side, without receiving ACK from the receiver, the sender will treat it as network congestion and start the congestion control process. Since the sender can not receive duplicate ACK, it should wait until RTO (retransmission timeout) and retry the transmission task. If the route handover latency is less than RTO, however, the sender still required waiting RTO to begin retransmission, while if the route handover latency is greater, the retransmission is useless at this moment. Thus, without effective dealing with packet loss caused by the route handover, unreasonable congestion control and retransmission will start.

In this paper, we choose AODV (Ad Hoc on Demand Distance Vector Routing) protocol [9]. When a source node desires a route to a destination for which it does not already have a route, it will broadcast a route request (RREQ) packet across the network, and a node receiving the RREQ may send a route reply (RREP) if it is either the destination or if it has a route to the destination. Once the source node receives the RREP, it will begin to forward data packets to the destination.

C. The impact of transmission errors on TCP

When the packets are on flight, they will affect by the channel and get attenuation, and when the packets arrive at the receiver, they could be received if they are still strength

enough. TwoRayGround model is one of the radio propagation models [10], which considers both the direct path and the ground reflection path, as shown in (2), where P_t is the transmitted signal power, G_r and G_t are the antenna gains of the transmitter and the receiver respectively, h_r and h_t are the heights of the transmit and receive antennas respectively, and d is the communication distance. It is clear that, in this ideal condition, d varies directly with P_t when transmit and receiving antennas are the same state, and when P_t is constant, d can be determined.

$$p = \frac{P_t * G_t * G_r * (h_t^2 * h_r^2)}{d^4} \quad (2)$$

However, there are many other factors that bring signal attenuation in VANET environment, in addition to communication distance, the Doppler frequency shift caused by node movement, the multipath effect caused by buildings, roads, trees and other surroundings, as well as some other interference factors can affect the channel quality. Therefore, even if the distance between two nodes is less than the communication range, it may still appear packet loss due to the effects of transmission errors. In [11], the laws of VANET channel are analyzed and concluded, and an experiential model of packet loss rate is built. According to this paper, when the communication distance of two nodes is small, the channel quality is good and stable because there are only a few vehicles in the communication range. As the distance increased, the number of vehicles increased and interference factors enhanced, which result in a higher packet loss rate. When the distance reaches a threshold value (while still less than the communication range), packet loss rate will change irregularly, and it can be considered as according to uniform distribution. In the case of high BER (bit error rate), if all the packet losses are still considered to be the instruction of network congestion and congestion control process start as soon as packet loss occurred, it is bound to make the window in a smaller value and decline system performance.

III. THE IMPROVED TCP BASED ON CROSS-LAYER DESIGN

In this section, we introduce the cross-layer optimization method to improve TCP performances in VANET. By providing useful route and movement information, the transport layer could distinguish the causes of packet loss quickly. On this basis, we improved TCP Reno and proposed a TCP congestion control mechanism based on cross-layer design in Vehicular Communication Networks. We call it VC-TCP, which improved the throughput and system stability, while reduced the end-to-end delay. According to the characteristics of VANET, we make the following assumptions:

- 1) All of the vehicles in VANET are treated as communication nodes with a GPS, so the nodes can obtain real-time road information;
- 2) The nodes can determine their own location coordinates and speed through GPS;
- 3) The nodes can exchange packets between each other to get real-time coordinates and speed of their neighbour nodes;

4) The nodes have the same transmit power (P_t), from equation (2) we can infer that the nodes have the same communication range ($Range$).

In addition, according to [11], when the distance between two nodes is less than the threshold value S ($S < Range$), the packets can be regarded as receiving without error, and in this case, we thought the channel is at "good" state; on the other hand, when the communication distance is larger, the channel is at "bad" state, and packets will be lost at a certain probability (P_0).

A. Four-State Transition Diagram of VC-TCP

The traditional TCP only has two states: normal state and congestion control state. We added two new states: the waiting state and "bad" channel state. The waiting state triggered by predicting route handover and lasts during the whole route handover process. The four-state transition diagram based on new VC-TCP mechanism is as shown in Figure 1.

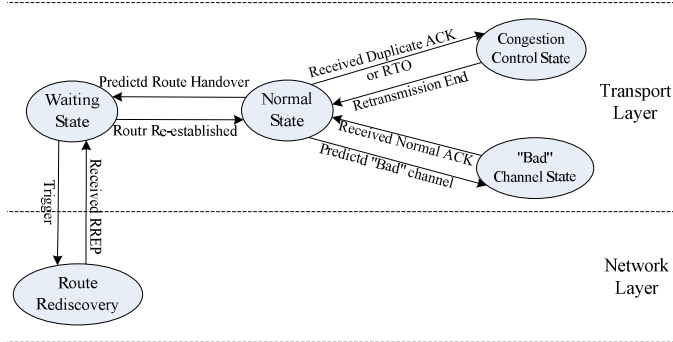


Figure 1. VC-TCP four-state transition diagram

The basic idea of VC-TCP based on four-state transition diagram is as follows:

- 1) Since the coordinates and speed of all the nodes and their neighbour nodes are available, so we can predict the coordinates of nodes in a SRTT (Smoothed round-trip time) later, and calculate the distance (Dis) between two neighbouring nodes;
- 2) Add two bits flag into the ACK packet header. When forwarding ACK from the receiver, each relay node on the route will first calculate Dis to the next hop in a SRTT later and then compared Dis with $Range$: if $Dis > Range$, it is indicate to be a route broken, and this will be recorded into the ACK flag bits, while if $S < Dis_{i,j} < Range$, there may occur packet loss at a certain rate(P_0), also record this into the ACK flag bits;
- 3) The sender detects the route instructions of each ACK received. If the sender fined that the route will broke, the waiting state will be triggered, and then the TCP sender will stop sending packet and inform the network layer to start route rediscovery immediately. In this case, the sender does not need to wait until receiving the feedback of link failure to start rediscovery, and unnecessary packet loss can be reduced. If the sender determined the packet loss due to transmission

errors, then we just need to reduce $ssthresh$ appropriately since the energy is supplied sufficiently in vehicular environment, in this case, the $cwnd$ could be in a larger state and the throughput could be improved;

- 4) For waiting state, when the network layer receives RREP, it is indicated that the route has been re-established well. The network layer will feedback this to TCP layer, and then the TCP sender will terminate the waiting state and return to normal state immediately. Since $cwnd$ of the old route may be too large for the new route, the packet should not be send at a full speed as before, but need be adjusted appropriately.

B. Improved Algorithm of VC-TCP

1) TCP Header Modification: In the standard format of TCP header, six unused bits are reserved with a default value 0. We modified two of the reserved bits, called VC bits, which is used to indicate the route handover and the "bad" channel state. As shown in Figure 2.

Source Port (16)				Destination Port (16)	
Sequence Number (32)					
Acknowledgement Number (32)					
Data Offset (4)	Reserved (6)	VC (2)	Flags (6)	Window (16)	
Checksum (16)				Urgent (16)	
Options and Padding (32)					
Data (Varies)					

Figure 2. Modified TCP header

2) Distance Prediction: The SRTT is given by:

$$SRTT_{n+1} = (1 - \alpha) \cdot SRTT_n + \alpha \cdot RTT_{n+1} \quad (3)$$

Where, RTT_{n+1} is the RTT value of current segment, and α is the smooth factor between 0 and 1. The values of α often is 1/8.

While forwarding a ACK packet, the receiver and ever relay nodes calculate their position coordinates (x'_i, y'_i) and coordinates of their next hop (x'_j, y'_j) in a SRTT later, according to their current position (x_i, y_i) and current velocity $\vec{v}_i$, and current position (x_j, y_j) and current velocity $\vec{v}_j$ of their next hop:

$$(x'_i, y'_i) = SRTT_{n,i} (\vec{v}_i \cdot \cos \theta, \vec{v}_i \cdot \sin \theta) + (x_i, y_i) \quad (4)$$

$$(x'_j, y'_j) = SRTT_{n,j} (\vec{v}_j \cdot \cos \theta, \vec{v}_j \cdot \sin \theta) + (x_j, y_j)$$

And the distance $Dis_{i,j}$ between them:

$$Dis_{i,j} = \sqrt{(x'_i - x'_j)^2 + (y'_i - y'_j)^2} \quad (5)$$

3) Channel State Prediction: For each node forwarding ACK packet, make a judgment as follows:

If $Dis_{i,j} > Range$, it is indicated to be a route handover, set the VC bits of TCP header to '10';

If $S < Dis_{i,j} < Range$, the channel is considered to be at "bad" state, and if the value of VC bits is not '10', set it to '01';

If $Dis_{i,j} < S$, the channel state is normal, and the VC bits should keep the default value '00'.

4) Cross-layer Mechanism for TCP-Network Layers: When the TCP sender received an ACK of which the VC is '10', it will inform the network layer immediately. Then the sender will send RRER to announce route failure, and send RREQ to re-initiate route discovery process. When the RREP is feedback back, the sender will immediately inform the TCP layer that the route has been re-established well. During this process, the TCP is at waiting state, and would not send a packet. The improved congestion control strategy is as described in (5).

5) The Improved Congestion Control Strategy: Based on TCP Reno, we improved the congestion control strategy. The sender will take different congestion control strategy according to different value VC bits:

If VC=10, the waiting state of TCP will be triggered, the TCP sender will record current $ssthresh$, $cwnd$, receive window ($rwnd$) and synchronize number (SYN), and set $ssthresh=1$, $cwnd=0$, $rwnd=0$ to suspend data transmission until receiving the feedback RREP from the network layer, then the TCP sender will immediately recover the $ssthresh$, $cwnd$, $rwnd$ and SYN as recorded before and send data at $4/5 cwnd$;

If VC=01, the channel is supposed to be a "bad" state, in this case, once the packet is lost, set $ssthresh = (4/5) * cwnd$ and $cwnd = cwnd / 2$, which could bring a relatively larger TCP window and improve system throughput;

If VC=00, TCP Reno congestion control strategy is applied, in which $ssthresh$ is set to $cwnd / 2$ and $cwnd$ is set to 1 when packet loss occurs.

IV. SIMULATIONS AND ANALYSIS

A. Traffic Scenario Simulation

The traffic simulation scenario is built by VanetMobisim, as shown in Figure 3. VanetMobisim is a traffic simulator, which is able to generate mobile trace file for network simulation software such as NS2 (For more information, please visit: <http://vanet.eurecom.fr/>). The scenario contains 60 vehicle nodes with a number from #0 to #59, and each node is adopting the intelligent driver model with intersection management and lane changing. Since the scenario is generated randomly, the faster the node is move, the more frequent the topology change is thought to be. The parameters are setting as Table 1.

TABLE 1. PARAMETERS OF TRAFFIC SIMULATION

Parameters	Value
Scenarios size	1000m*1000m
Number of vehicles	60
Maximal speed	30m/s
Minimal speed	0m/s
Average speed	0, 5, 10, 15, 20, 25 m/s
Maximal acceleration	0.5m/s ²
Density of roads	0.00001
Maximum speed allowed on the road	30m/s
Number of lanes	2

B. Network Scenario Simulation

The NS2 is used for network simulation, and 802.11Ext module is introduced to better simulate the Vehicular Communication Networks based on 802.11p. According to 802.11p standard [12], the parameters of physical layer and MAC layer are configured in detail. We select the 5.9GHz frequency and 10MHz bandwidth, and set the reception range to 250m and carrier sense range to 550m. The network layer chooses AODV protocol. The specific parameters are shown in Table 2.

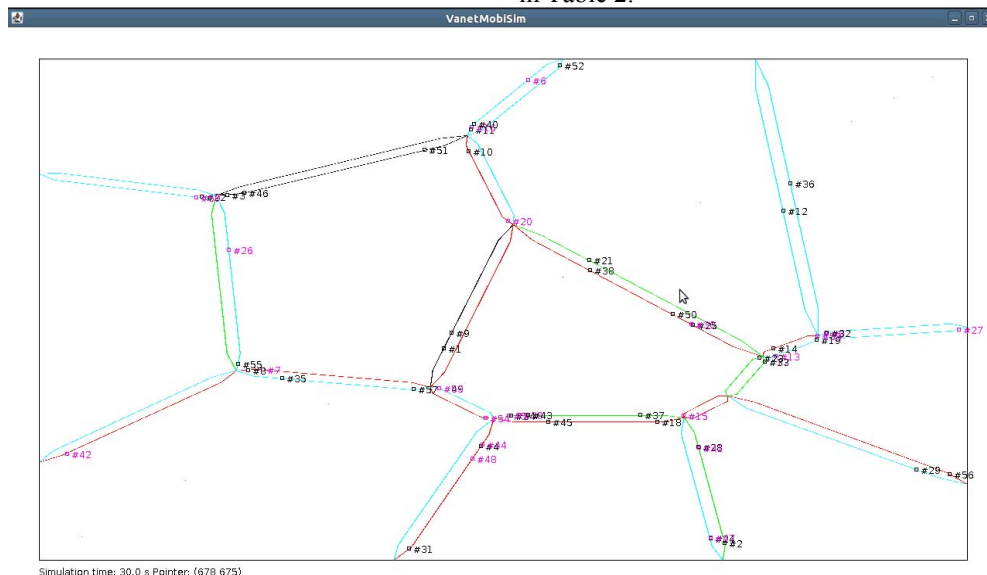


Figure 3. Traffic simulation scenario built by VanetMobisim

TABLE 2. PARAMETERS OF NETWORK SIMULATION

Parameters	Value
PHY layer	WirelessPhyExt
MAC layer	802_11Ext
Channel Propagation model	TwoRayGround
Frequency	5.9GHz
Bandwidth	10MHz
Reception range	250m
Carrier sense range	550m
Packet size	1000Byte
Routing protocol	AODV
Application layer	FTP

C. Simulation Result and Analysis

We used the traffic scenario and network scenario as above. The nodes are moving in six kinds of average speed: 0 m/s (stationary), 5 m/s, 10 m/s, 15 m/s, 20 m/s and 25 m/s. The simulation time is 500 seconds, and a successive transmission of FTP application is existed during the simulation. For each average speed, the traffic scenario simulation should run five times, and for each time, a trace files is generated. Apply each trace file into the network simulation with VC-TCP and TCP separately, and calculate the average throughput and end-to-end delay of each simulation.

Since reception range (*Range*) is 250m, we can set the threshold value (*S*) to 200m. In this case, when the distance between nodes is larger than 200m, the channel will be "bad" state, and we set the packet loss probability $P_0=0.1$. The simulation results are as shown in Figure 4, Figure 5.

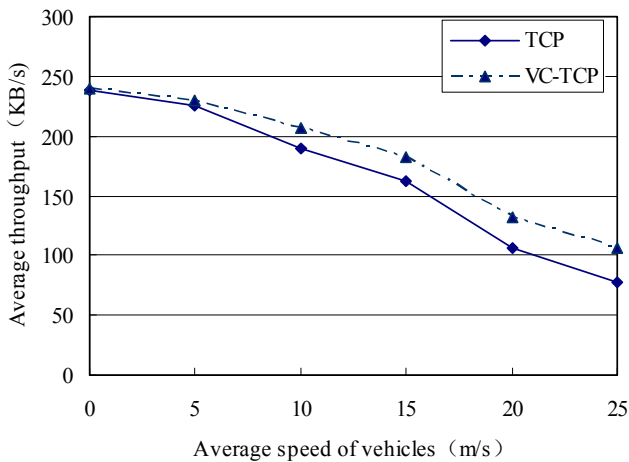


Figure 4. Average throughput comparison between VC-TCP and TCP Reno with different speed

Figure 4 shows the average throughput compared between VC-TCP and TCP with different speed. It can be seen from Figure 4 that, with the average speed increased, the average throughput of both VC-TCP and TCP tend to decrease, which mainly cause by more frequently route broke. However, the performance of VC-TCP has improved significantly, compared with TCP Reno at the same speed, because the VC-TCP not only can predict the route handover to avoid unnecessary packet loss and congestion control during the

handover, but also be able to slow down the process of congestion control when the channel is "bad", so that the window will not rapidly reduced.

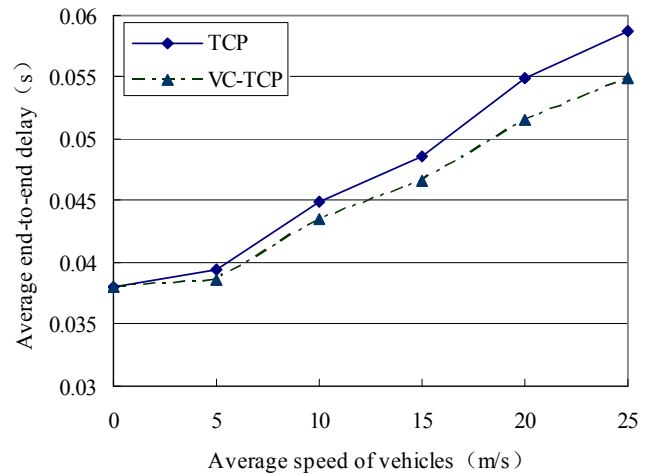


Figure 5. Average end-to-end delay comparison between VC-TCP and TCP Reno with different speed

The average end-to-end delay compared between VC-TCP and TCP with different speed is shown in Figure 5. The end-to-end delay mainly includes the waiting time in the transmit buffer, transmission time, relay nodes forward time and retransmission time. Once the route is broken, the packet will get lost and unnecessary retransmissions will be resulted in. Fortunately, the VC-TCP can aware route handover in advance to reducing unnecessary packet loss, and continue to send data as soon as the route is re-established, so the system end-to-end delay can be decreased. As shown in Figure 5, the average end-to-end delay of TCP and VC-TCP raised with the increase of average speed. However, the VC-TCP increased more slowly than TCP Reno, so we can conclude that, when the nodes move in high-speed and the topology changes frequently, the VC-TCP can effectively reduce the average latency.

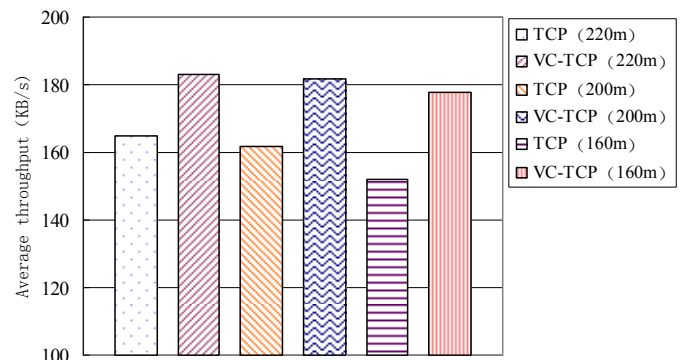


Figure 6. Average throughput of VC-TCP and TCP Reno with different *S*

When *Range* is given, the change of *S* will have an impact on TCP performance. Setting the average speed to 15m/s, the average throughput of VC-TCP and TCP with different *S*, respectively 160m, 200m, and 220m, is shown in Figure 6. According to [11], the value of *S* set to 200m is

appropriate. When S set to 160m, which means a even worse channel state, the packet loss will occur at a certain probability ($P_0=0.1$) once the distance between two nodes is larger than 160m, on the other hand, when S is 220m, the channel state will be in more favourable state. The simulation results in Figure 6 shows that, as the channel quality dropped, the probability of packet loss increased, both the performance of VC-TCP and TCP are affected and the throughput are decreased to a certain extent. Compared with the TCP Reno, the VC-TCP shows a better stability, because the VC-TCP could maintain a relatively high throughput when the quality of the channel is getting worse, furthermore, the worse the channel condition is, the more obvious this advantage is.

V. CONCLUSION

In Vehicular Communication Networks, the main reasons for causing packet loss are link congestion, route handover and transmission errors. The VC-TCP integrate the TCP protocol with routing protocol based on cross-layer design, which can predict the reasons for packet loss by analyzing movement information of vehicle. A four-state transition diagram is proposed, and on this basis, the TCP congestion control strategy is improved, which can take different TCP congestion control strategies for different reasons. The VC-TCP can improve the average throughput and end-to-end delay in Vehicular Communication Networks, especially when the nodes move in high-speed, the topology changes frequently and the channel quality is poor. In the next step, we will consider a joint optimization that integrates QoS of application layer and resource allocation of physical layer with the cross-layer scheme in this paper to further improve the network performance in Vehicular Communication Networks environment.

REFERENCES

- [1] Bernardos C.J., Festag A., Maxemchuk N.F., et al, "Advances in vehicular communications networks", *IEEE Network*, vol. 24, no. 1, pp. 4-5, 2010.
- [2] Theodore L. Willke, Patcharinee Tientrakool, Nicholas F. Maxemchuk, "A Survey of Inter-vehicle Communication Protocols and Their Applications", *IEEE Communication Surveys & Tutorial*, vol. 2, no. 11, pp.3-20, 2009.
- [3] WANG Zhaoran, XIE Xianzhong, ZHAO Dingxin, "Key Technique of Vehicular Ad Hoc Network", *Telecommunications Science*, vol. 27, no. 1, pp. 44-51, 2011.
- [4] Sommer C., Dressler F., "Progressing toward realistic mobility models in VANET simulations", *IEEE Communications Magazine*, vol. 46, no. 11, pp. 132-137, 2008.
- [5] Marc Bechler, Sven Jaap, Lars Wolf, "An Optimized TCP for Internet Access of Vehicular Ad Hoc Networks", *Lecture Notes in Computer Science*, vol. 3462, pp. 869-88, 2005.
- [6] ZHANG Xinming, XIE Fei, WANG Wenjing, et al, "TCP Throughput for Vehicle-to-Vehicle Communications", in *Proc. IEEE International Conference on Communications and Networking in China(ChinaCom)*, 2006, pp. 25-27.
- [7] Andrew Chen, Behrooz Khorashadi, Dipak Ghosal, et al, "Impact of Transmission Power on TCP Performance in Vehicular Ad Hoc Networks", in *Proc. IEEE Fourth Annual Conference on Wireless on Demand Network Systems and Services*, 2007, pp. 65-71.
- [8] CHEN Li-Jia, JIANG Hao, WU Jing, et al, "Research on Transmission Control on Vehicle Ad-Hoc Network", *Journal of Software*, vol. 18, no. 6, pp. 1477-149, 2007.
- [9] Perkins C., Belding-Royer E., Das S, "Ad hoc On-Demand Distance Vector (AODV) Routing", *IETF, RFC 356*, 2003, Retrieved 2010.
- [10] Eltahir I.K., "The Impact of Different Radio Propagation Models for Mobile Ad hoc NETWORKS (MANET) in Urban Area Environment", , in *Proc. The 2nd International Conference on Wireless Broadband and Ultra Wideband Communications*, pp. 27-30, 2007.
- [11] JIANG Hao, CHEN Lijia, YAN Puli, "Investigation and Modelling of VANET Channel Laws", *Journal of Wuhan University (Natural Science Edition)*, vol. 18, no. 6, pp. 114-118, 2008.
- [12] IEEE 802.11p Part 11: "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6", *Wireless Access in Vehicular Environments* [Online]. Available: <http://standards.ieee.org/getieee802/download/802.11p-2010.pdf>, July, 2010.