Congestion Control vs. Link Failure: TCP Behavior in mmWave Connected Vehicular Networks

Woongsoo Na, Demeke Shumeye Lakew, Jihoon Lee, and Sungrae Cho

W. Na is with Electronics and Telecommunications Research Institute (ETRI), Yuseong-gu, Daejeon, Gajeong-ro 218, South Korea; (E-mail: wsna@etri.re.kr).

D. S. Lakew and S. Cho are with the School of Computer Science and Engineering, Chung-Ang University, 156-756 Seoul, South Korea; (E-mail: wsna@uclab.re.kr; demeke@uclab.re.kr; srcho@cau.ac.kr).

J. Lee is with the Department of Information and Telecommunication Engineering SangMyung University, 330-720 Cheonan, South Korea; (E-mail: vincent@smu.ac.kr).

Abstract

The millimeter-wave (mmWave) frequency band presents a viable communication platform for the exchange of large amount of data and communications of Connected Vehicular Networks (CVNs). Its ability to support multi-gigabit data rates and high spatial reuse is especially crucial because of the large volume of data produced by physical entity each second. However, its susceptibility the deafness problem, blockages, and beam misalignment is a major impediment, especially in highly mobile environment such as CVNs. Especially, CVNs are related to the life of human being, network reliability is one of the most important issue. To resolve this problem, recent studies on mmWaves have focused more on the PHY/MAC Layers, while fewer studies have examined the transport layer. In this paper, we examine the performance of TCP in mmWave based vehicular networks, focusing particularly on the effects of congestion and link failure. In addition, we highlight the problems of using conventional TCP in an mmWave CVNs and propose an real-time wireless TCP (RTW-TCP) implementation such mmWave environment. Finally, we compare the performance of the proposed RTW-TCP and conventional TCP. The simulation results show that the proposed scheme maximizes the total network resources by distinguishing link failure from network congestion in an mmWave CVNs.

Keywords: mmWave, TCP, congestion control, wireless link failure, and connected vehicular networks.

---

1. Introduction

Mobile data traffic is growing at an exponential rate. According to Cisco, global mobile data traffic is expected to reach about 49 exabytes per month by 2021, while a 5G connection will generate almost five times more traffic than a regular 4G connection. Consequently, to accommodate global mobile traffic demand of this magnitude, the millimeter-wave (mmWave) band in 5G mobile networks has been identified as a potential solution. The mmWave band lies between 30 and 300 GHz on the frequency spectrum, and offers tremendous amounts of bandwidth and data rates (multi-gigabit) to support services such as high definition and ultra-high definition television (HDTV and UHDTV, respectively), high quality virtual reality (VR), and in-vehicular infotainment systems [1]. Thanks to the wide bandwidth of mmWave, it has become a core technology of 5G mobile networks and enables very high speed device-to-device (D2D) communications, massive machine communications (MMC), and ultra-dense networks (UDNs).

Based on the aforementioned qualities, mmWave frequency has immense potential in Connected Vehicular Networks (CVNs) communications in terms of the exchange of large volume of sensing data to improve traffic safety and to support applications for in-vehicle entertainment and an augmented driving experience [2]. This requirement is highlighted by the volume of data that vehicular on-board sensors create. According to [1], 750 MB of sensor data are generated every second by Google's self-driving car. This translates to over 2 TB of generated sensor data per hour. Although the whole of generated data does not need to share with other vehicles[1], there is still a large amount of data should be transmitted. Unfortunately, the current vehicular communication standard, dedicated short-range communication (DSRC) [3], which has a theoretical maximum data rate of about 27 Mbps, is insufficient for such high volumes of data. Research on vehicular communications using mmWaves started about three decades ago. A European ITS project, which ran between 1987 and 1994, called PROMETHEUS (PROgraMme for European Traffic with Highest Efficiency and Unprecedented Safety) utilized the 57 GHz frequency to achieve cooperative driving among vehicles using vehicle-to-vehicle communications [2]. Similarly, a joint research project on mmWaves (60 GHz) for inter-vehicle communication was carried out at Yokosuka Research Park in Japan in the late 1990s [3]. Experiments were also carried out by the Communications Research Laboratory in Japan for the transmission of high quality TV between vehicles over a short distance using 60 GHz mmWaves [2]. These projects confirm the

---

[1] Compression algorithm or feature selection can be applied to reduce data size

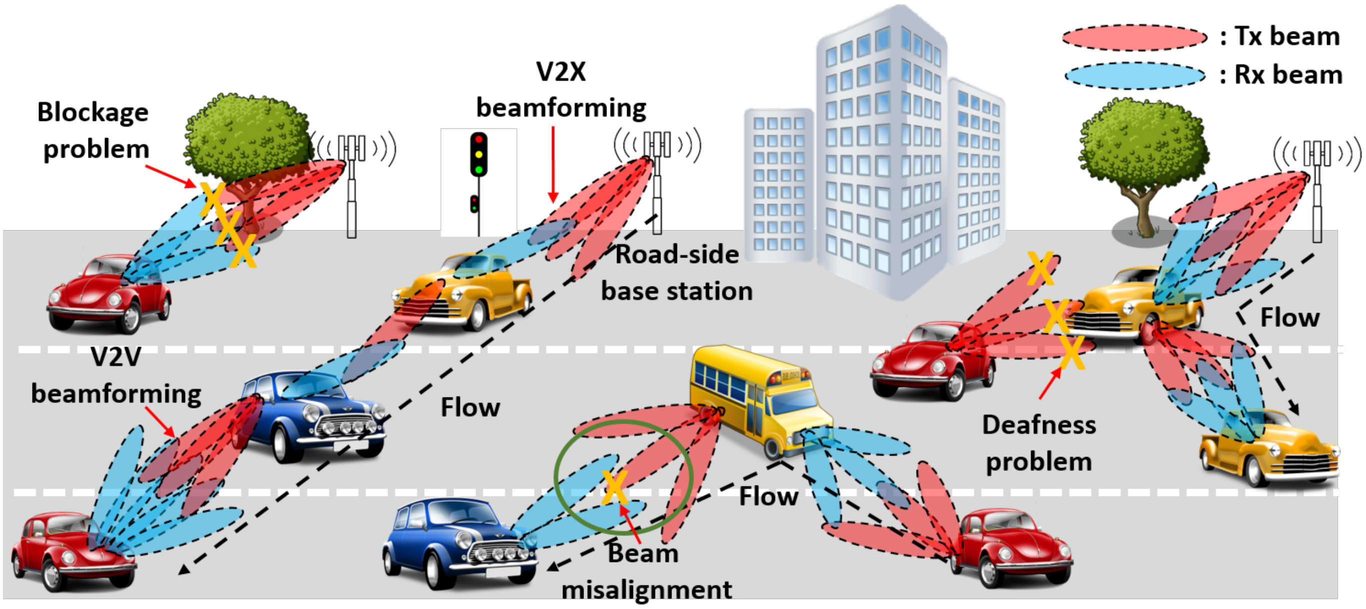


Figure 1: V2X communications and technical issues related to the mmWave band

viability of the mmWave frequency in vehicular communications. However, despite these benefits, mmWave communication presents several significant technical challenges, including beam misalignment, sensitivity to blockages, the deafness problem, directivity, and high propagation loss, as illustrated in Figure 1. Therefore, in order to harness the potential of mmWave communications, both industry and academia are attempting to resolve these challenges.

Additionally, the reliability of end-to-end data transmission in CVNs is very necessary feature since CVNs can be considered as real-time Internet of Things (IoT) system, which requires end-to-end real-time performance. Thus, research on the TCP responsible for the reliability of end-to-end data transmission is one of the important issues to be addressed. In the future CVNs, there are a number of potential use cases of end-to-end communications. For instance, the automotive sensors can identify the traffic volume around the area and reports the remote server which controls the timing of the traffic light. Moreover, they can quickly identify the roads requiring repairs and reports to repairman or road management corporation before the danger appears to improve road safety and reduce the inconvenience.

Thus, in this paper, we assume that vehicles are embedded with mmWave directional antennas, which enable communications with neighboring vehicles and road-side base station (RBS) units by beamforming. We examine the behavior of TCP in mmWave CVNs. We also propose real-time wireless TCP (RTW-TCP) to solve the above problems and measure its performance compared with the conventional TCP in CVNs.

The contributions of this paper can be summarized as follows:

- We investigated various existing TCP and analyzed whether it is suitable for mmWave band for each TCP characteristic.
- We analyzed various link failure phenomena occurring in mmWave band and simulated the behavior of existing TCP in the mmWave. In simulation, we investigated the rate at which link failure and congestion occur.
- We propose a RTW-TCP suitable for the characteristics of the mmWave band, and compare it with the existing TCP through the experiment in the CVNs environment.

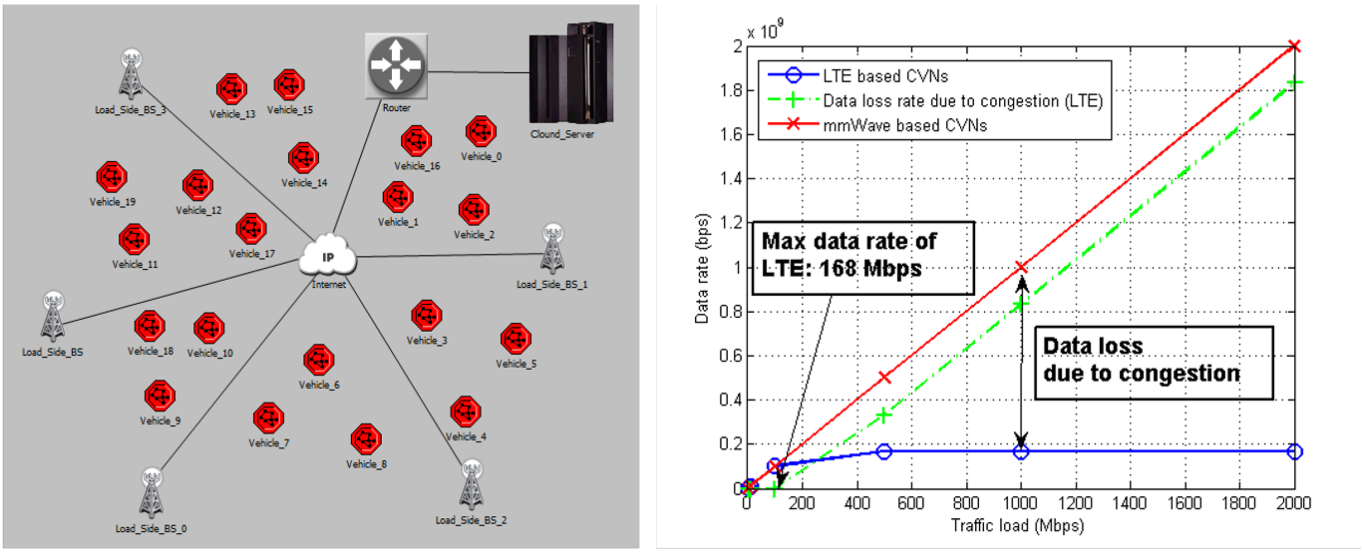
Section 2 presents the technical challenges for TCP design in mmWave-based CVNs. In Section 3, we introduce existing wireless TCP and propose an enhanced wireless TCP for mmWave CVNs. Then, in Section 4, we evaluate the performance of the conventional TCP schemes (TCP Cubic and TCP Compound) and the proposed real-time wireless TCP (RTW-TCP) using case studies (rural area and urban area with Manhattan mobility and random mobility). Finally, Section 5 concludes the paper and suggests possible future research directions.

2. Technical issues in mmWave CVNs

In this section, we discuss the technical challenges for mmWave TCP design in CVNs. Principally, we examine how conventional TCP functions, such as congestion control, error control, and flow control, work in the CVNs.

2.1. How much congestion occurs in mmWave CVNs?

To answer this question, we analyze the behavior of conventional TCP in mmWave CVNs by means of a case study. We begin our analysis with a comparison of the network congestion in LTE and mmWave bands for CVNs. We deployed 20 vehicle



(a) Topology scenario (20 vehicles, 5 load-side BS)

(b) Data rate (bps) for LTE and mmWave CVNs

Figure 2: Data rate and loss rate for LTE and mmWave-based CVNs

Table 1: Simulation parameters for mmWave-based CVNs

Parameters	**Values**
Topology size	100m x 100m
Speed of vehicles	5 m/s - 30 m/s (Uniformly Distributed)
Traffic loads	200 Mbps - 2 Gbps
Operating frequency	28 GHz
Number of symbols per slot	24
Packet length	1400 bytes
Core network delay	1 ms
RLC buffer size	50 MB

nodes and five road-side base stations in a 100m x 100m topology using the OPNET simulator[2]. In this simple scenario, all vehicles are randomly deployed and do not have any mobility[3]. Additionally, we assume that each vehicle had one passenger (1 UE) receiving data from an external cloud server. Both the data rate and the loss rate due to congestion were measured by varying the traffic in the order of 1 Mbps, 10 Mbps, 100 Mbps, 500 Mbps, 1 Gbps and 2 Gbps. For LTE model, we applied LTE High Speed Packet Access+ model (3GPP Rel.11), which supports to 168 Mbps, while 5G architecture is applied for the mmWave model. More detailed simulation parameters are listed in Table 1.

Figures 2 (a) and (b) show the simulation topology and the data rate and loss rate due to network congestion in CVNs, respectively. Note that, in this scenario model, all devices use conventional TCP protocol. As shown in the figure, data loss due to congestion increases rapidly with increasing amounts of traffic in the LTE-based CVNs. Conversely, owing to the very wide bandwidth (5.6 Gbps) present in the mmWave environment, congestion-related data losses do not occur.

2.2. *The Deafness problem in mmWave CVNs*

The deafness problem is an issue related to directional antennas that occurs if a node (vehicle or RBS) fails to reply to a directional request frame addressed to it, leading to a retransmission by the originator [4]. This increases the contention window, during which time messages to other nodes are blocked. This problem occurs frequently in single-antenna systems in the mmWave band, and has a high probability of occurring in CVNs owing to the recurrent nature of V2V communications.

Recently, various studies have investigated how to prevent this problem or to minimize its damage. However, it is inevitable that communication between vehicles cannot be performed, even within a short period [4].

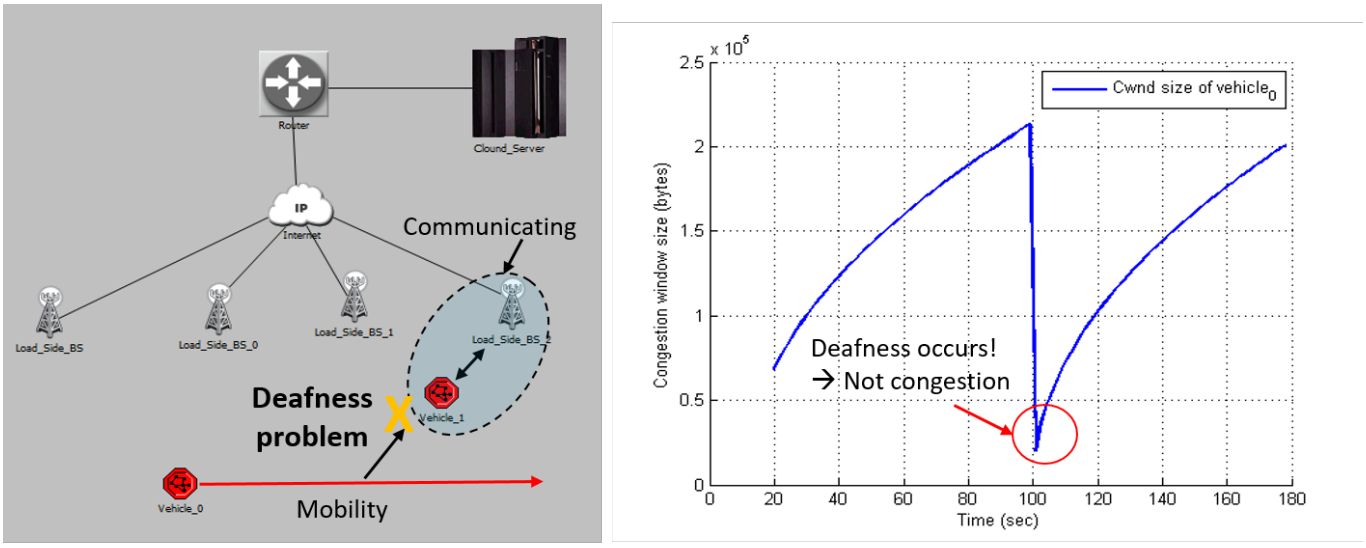
Although the problem of deafness increases the delay between terminals, it is not a factor that causes network congestion. Unfortunately, conventional TCP identifies the deafness problem as network congestion, and invokes it congestion control mechanisms by adjusting the congestion window (*cwnd*) size, thereby reducing network throughput. Therefore, despite the existence of available network resources, they are not able to be utilized efficiently.

Figures 3 (a) and (b) show the simulation topology and the trend of *cwnd* variation in conventional TCP when deafness occurs, respectively. In this scenario, assume that *Vehicle 0* is moving and receiving data from the cloud server by communicating with the RBS. *Vehicle 0* moves away from the RBS and sends a connection request frame to continue to receive the service through *Vehicle 1*. However, *Vehicle 1* does not hear

---

[2] All vehicles maintain one connection to one RBS. Thus, vehicles receive data from only one RBS even if they are located in the service range of more than two RBSs.

[3] Mobile scenarios are described in Section 4.



(a) Topology scenario (Deafness scenario)

(b) *Cwnd* size of vehicle_0

Figure 3: Congestion window size variation in deafness scenario

the message. As shown in the figure, after the deafness occurs, *cwnd* drops rapidly, even though it can be resolved.

2.3. *The Blockage Problem in mmWave CVNs*

In mmWave CVNs, a signal between nodes may be interrupted by a building, sign, or tree, or by obstacles on the road, because the very high frequency results in attenuation of the received SNR. Because the vehicles and the RBS in the mmWave link need to use a directional antenna to concentrate power and compensate for the high path loss, this problem becomes particularly serious [5].

In addition, in metropolitan areas, blockage problems can occur more frequently because of high-rise buildings, crowds, and crowded vehicles. Even if the intensity of the signal is reduced by obstacles, it can escape within a short time because of the mobility. However, TCP mistakenly interprets a break in the short link as network congestion and, thus adjust *cwnd*. In addition, as a result of interference and signal attenuation due to the characteristics of wireless channels, it is very difficult to apply TCP properly to mmWave CVNs. As shown in Figures 4 (a) and (b), a blockage problem causes *cwnd* to return to its initial value after a temporary link disconnection at the wireless link level.

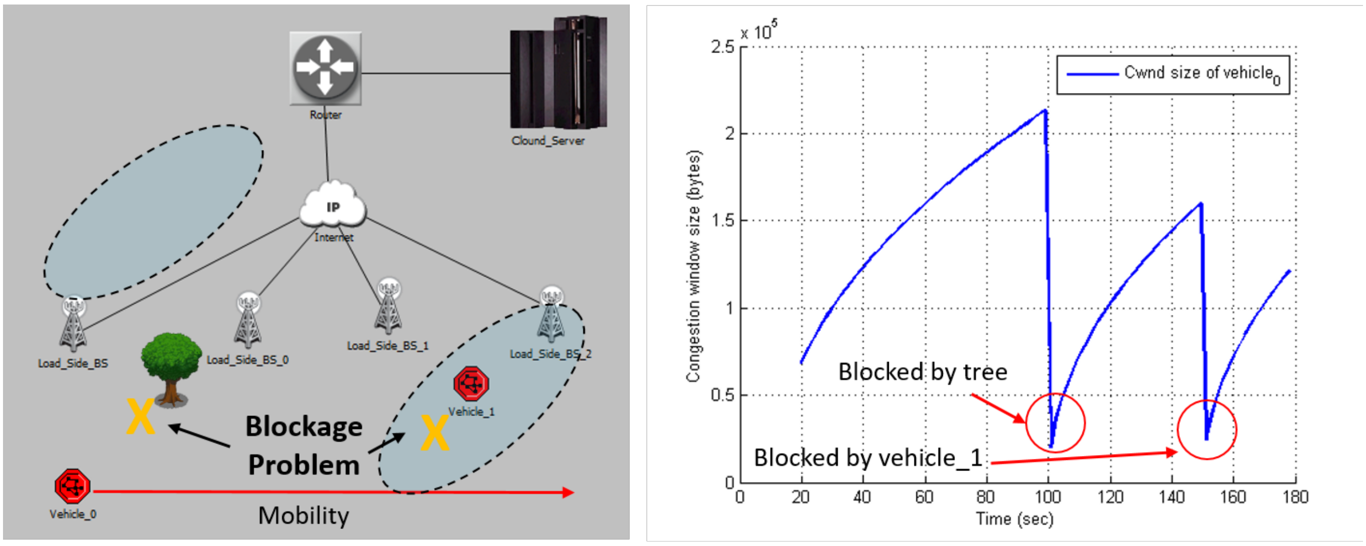
2.4. *Beam misalignment in mmWave CVNs*

Millimeter-wave frequencies suffer from significant path loss. To compensate for this loss, mmWave radios use highly directional antennas to focus the signal power in a narrow beam. Such directional antennas can be implemented using phased arrays. In fact, because the wavelength is very small (of the order of a millimeter), tens or hundreds of such antennas can be packed into a small space, creating a pencil-beam antenna. The beam can be steered electronically within a few microseconds. However, the real challenge is to identify the correct spatial direction that aligns the transmitters beam with the receiver's beam [6].

In CVNs, the position of a vehicle changes frequently owing to the movement of the vehicle, often causing the beam misalignment problem. If the beam between V2V or V2I is not aligned, the signal quality will be temporarily degraded and the link can be disconnected. Recently, in vehicular environments, a large amount of side information has been available from various sources, such as automotive radars, visual cameras, LIDARs, or even DSRC devices. Based on this information, efforts have been made to reduce the beam switching overhead by tracking the beam in real time, but the problem of temporary link disconnections still exists. In this case, the existing TCP invokes the congestion control algorithm because it interprets the delays caused by disconnected links as network congestion. However, this is a temporary link disconnection, which results in wasting available resources. As shown in Figures 5 (a) and (b), a beam misalignment problem causes *cwnd* to return to its initial value after a temporary link disconnection at the wireless link level. In addition, in CVNs, because the phases of the transmission beam and the reception beam between vehicles change continuously, temporary disconnections occur frequently. The figure shows the results when the beamwidth is 30 degrees. Furthermore, this problem occurs more frequently here than in other experimental environments.

3. TCP Behavior in the Literature

Conventional congestion control mechanisms for TCP were mainly designed for wired networks. In such networks, all segment losses are the result of congestion, with those due to random bit error rates (BERs) being negligible. Therefore, these the congestion control mechanisms consider all segment losses as congestive losses and non-congestive losses are assumed



(a) Topology scenario (Blockage scenario)

(b) *Cwnd* size of vehicle_0

Figure 4: Congestion window size variation in blockage scenario

to be negligible. However, this assumption does not hold in wireless networks, in which segment losses are often induced by mobility, signal contention, signal fading, or reasons other than network congestion. The incapability of traditional congestion control mechanisms to differentiate between random packet losses and congestive losses results in significant performance degradation in wireless networks.

In this section, we describe some of the existing congestion control mechanisms, as well as proposed new solutions for TCP in wireless networks.

3.1. *Conventional TCP schemes*

TCP-Tahoe [7] and TCP-Reno are the two most popular conventional congestion control mechanisms for TCP. TCP-Reno implements a fast recovery algorithm in addition to the slow start, congestion avoidance, and fast retransmit algorithms of TCP-Tahoe. Until either a slow start threshold (*ssthresh*) is reached or a packet loss is detected, the size of the *cwnd* grows exponentially in round-trip times (RTT). Both Tahoe and Reno consider a retransmission timeout (RTO) and duplicate ACKs as packet loss events due to congestion. The difference with the two protocols is mainly in how they react to duplicate ACKs. If duplicate ACKS are received, Tahoe carries out a fast retransmit, sets the *ssthresh* to half the current *cwnd*, reduces the *cwnd* to 1 maximum segment size (MSS), and resets to the slow start state. In contrast, Reno saves half the current *cwnd* as *ssthresh* and as the new *cwnd*, thus, skipping the slow start and entering a fast recovery phase to speed up the recovery process [7]. However, both mechanisms are limited in their ability to handle multiple packet losses in a single *cwnd*, which is common in wireless networks. TCP-New Reno improves the retransmission during the fast recovery phase of Reno to alleviate this problem [8]. On the other hand, to solve the limitation of the cumulative ACKs scheme in TCP, TCP-SACK is proposed as a selective ACK (SACK) option for TCP. SACK implements a selective retransmission policy. Thus, the sender needs to retransmit only those segments that were actually lost, based on the selective notification of the receiver.

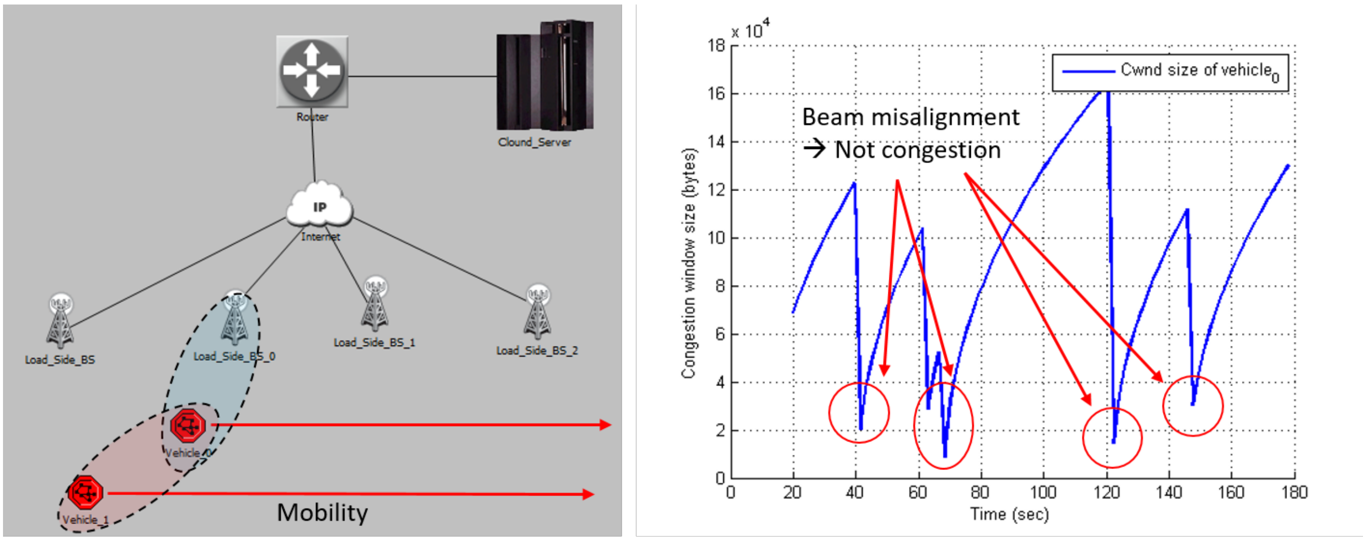
Conventional congestion control mechanisms implicitly assume that all packet losses are due to congestion. Consequently, the congestion window is reduced by half. However, as noted, this assumption does not hold for wireless networks, because packet loss due to the high bit error rate of wireless link would make the sender reduce the transmission rate unnecessarily, resulting in performance degradation. In particular, in an mmWave CVNs environment, this problem becomes severe because of the high path loss and mobility of vehicles.

3.2. *Wireless TCP schemes*

In this subsection, we introduce the representative TCP schemes for wireless networks. Basically, wireless TCP aims to differentiate between packet loss due to congestion and packet loss due to link failure. Based on the techniques they employ, we categorize wireless TCP as follows.

3.2.1. *Traffic measurement-based TCP*

TCP schemes in this category are techniques for distinguishing between congestion and link failure based on the queue backlog of routers and intermediate nodes [9, 10]. TCP-Veno [9] has been proposed as an end-to-end congestion control mechanism, with sender-side refinements on top of Reno to deal with random packet loss in wireless access networks. TCP-Veno adopts the same mechanism as TCP-Vegas to estimate the backlogged packets in the network. The estimated value is used to identify the cause of the packet loss. If the estimated backlog is less than a certain threshold, the loss is assumed to be random (link failure). Otherwise, the loss is categorized as congestive loss. Therefore, all of the Reno congestion control schemes are



(a) Topology scenario (Beam misalignment)

(b) *Cwnd* size of vehicle_0

Figure 5: Congestion window size variation in beam misalignment scenario

adopted. TCP-Veno utilizes the estimated backlog information to enhance two of Reno's congestion mechanisms.

The additive increase algorithm is modified such that if the estimated backlog packet exceeds the threshold, the congestion window is increased by one for every two round trip times, rather than for each round trip. The other modification is to the multiplicative decrease algorithm, such that if the connection is not in the congestive state and random loss is inferred, the slow start threshold is set to 4/5 of the congestion window, rather than 1/2. Veno achieves a far higher throughput than Reno, is less aggressive than Reno in terms of the number of timeouts, fast retransmits triggered, and retransmitted packets [9].

3.2.2. Biased queue management-based TCP

Differentiating congestive losses from non-congestive losses is difficult when wireless congestion is light. TCP-Casablanca [11] implements a simple biased queue management scheme on its to distinguish congestion losses from random losses in order to improve the performance of TCP in a wired−wireless network. The main idea of TCP-Casablanca is to de-randomize congestion losses such that the distribution of such losses differs from that of wireless error losses. TCP-Casablanca uses a discriminator function to distinguish congestive losses from random losses. This helps a TCP receiver to determine accurately the cause of a loss and to notify the TCP sender to react accordingly. TCP-Casablanca identified congestion losses with an accuracy of more than 95 percent, and wireless losses with an accuracy of more than 75 percent. This improves the performance of TCP in terms of throughput compared with those of TCP-New Reno [8] and TCP-Westwood [12]. However the classification accuracy decreases when the level of congestion is high.

3.2.3. Congestion timeout delay-based TCP

This type of TCP algorithm resolves link level errors through retransmission before a TCP timeout occurs owing to congestion.

The delayed congestion response TCP (TCP-DCR) [13] was proposed to improve the TCP performance for non-congestive events in a wireless network. The main idea of this protocol is to delay the triggering of congestion response algorithm for a small bounded period after the first duplicate acknowledgment is received by the receiver. The delay time is used to allow the link-level retransmissions to recover non-congestive losses due to channel errors. Upon receiving the first duplicate ACK, the TCP-DCR sender starts a delayed response timer of one RTT, and allows the link-level retransmission to recover the lost packet. However, if the packet is not recovered within the delayed period, the loss is considered to be a congestive loss and TCP-DCR triggers the fast retransmit and recovery algorithms to recover the packet.

TCP-DCR [13] does not require modifications at the receiver side and the performance of TCP-DCR is much better than that of TCP-Reno and TCP-Westwood in the presence of wireless channel errors. However, other network factors that affect the degree of packet reordering are not considered, whereas the chosen delayed time is dependent only on RTT.

Apart from the high bit error rate in a wireless network, the absence of infrastructure and high node mobility mean that frequent route changes and network partitions in wireless ad hoc networks also pose considerable challenges to the performance of TCP. Frequent route changes can cause out-of-order delivery of packets at the destination, which results in further misinterpretation by the congestion control mechanism of TCP.

3.2.4. Freeze-based TCP

TCP-Feedback [14] intends to distinguish route failure from

Table 2: Comparative analysis of TCP techniques and operability in the mmWave band

	Ability to distinguish link failure from congestion	**Behavior in the mmWave CVNs**
Conventional TCP (wired TCP)	- Cannot distinguish between congestion and link errors.	- It is difficult to achieve maximum throughput because it does not distinguish link errors from congestion.
Traffic measurement-based TCP	- Distinguishes between congestion and link errors based on queue backlog.	- Because of vehicle mobility, network backlog measurement is very difficult.
Biased queue management-based TCP	- Distorted congestion distributions are generated to identify congestion and link errors.	- It is very difficult to generate a consistent congestion distribution because of the mobility of the vehicle and the unpredictability of the link.
Congestion timeout delay-based TCP	- By delaying the RTO timer, the RTO does not occur in the link layer owing to retransmission at the MAC layer.	- It is difficult to predict the time required for error recovery - If the vehicle moves at low speed, the blockage duration can be long.
Freeze based-TCP	- When routing failure occurs, RTO is not generated by freezing the timer for a while. - Cannot distinguish wireless link failures.	- It is difficult to achieve maximum throughput because it does not distinguish link errors from congestion.

congestion. It depends on the network layer at the intermediate mobile hosts to detect route failures, owing to the mobility of the next mobile host along the route. As soon as an intermediate mobile host detects a route failure, it sends an explicit route failure notification (RFN) packet to the source and records this event. If the intermediate host that receives the RFN packet knows an alternate route to the destination, these packets can be rerouted through that alternate route to the destination, and the RFN is discarded. Otherwise, the host relays the RFN to the source.

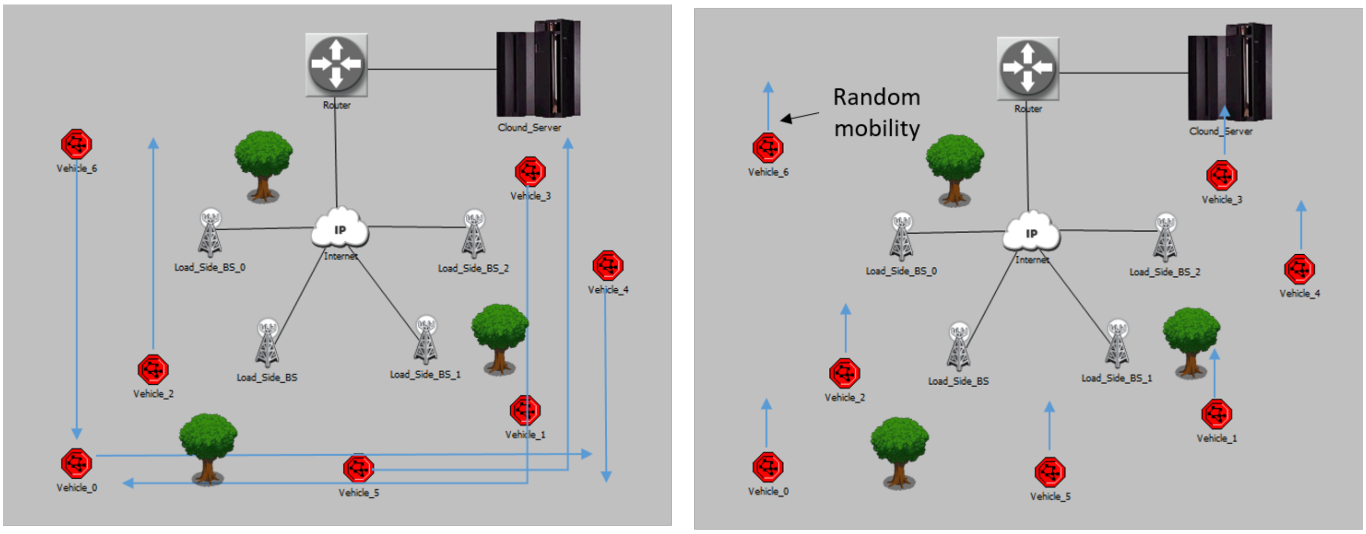
Upon receiving the RFN, the source goes into snooze state, and then stops sending further packets and freezes the value of state variables, such as the retransmission timer and the congestion window size. The source remains in the snooze state until it is made aware of the restoration of the route through a route reestablishment notification (RRN) packet. Once the source receives the RRN packet from an intermediate host it re-enters the active state, where transport is controlled by the normal TCP [14]. TCP-Feedback outperforms TCP as the route reestablishment delay increases. Even though it can avoid performance deterioration from route failure, it cannot avoid or identify occasional packet losses from signal fading.

3.2.5. *mmWave Wireless TCP*

The previously introduced TCP protocol distinguishes between congestion and errors at the wireless link level in frequency bands other than the mmWave band, and congestion control is performed only when congestion occurs. However, because the existing wireless TCP was designed without considering the problems caused by the mmWave CVNs introduced in Section 2, it is limited in its applicability to the mmWave frequency band. In [5, 15, 16, 17], the authors simulated the operation of the conventional TCP (Tahoe, Reno, and Cubic) in the mmWave band. In their experiments, conventional TCP in the mmWave frequency band 1) takes a long time to achieve full throughput owing to the wide bandwidth, 2) increases the end-to-end latency because of the high link error rate, and 3) suffers from frequent RTOs (retransmission timeout) and the re-initializing of *cwnd*. Additionally, the authors performed TCP performance analysis with SNR reduction due to obstacles when the node simply moved in a straight line (pedestrian and train mobility).

There are some TCP studies that take into account the mmWave bandwidth characteristics rather than simple experiments. In [18] and [19], the authors focused on the *bufferbloat* problem in the mmWave band. Bufferbloat is a problem in that if the buffer size of the intermediate node is large, packets are trapped and the total delay increases. In the mmWave band, this problem is caused by the wide bandwidth that exceeds the output speed of the buffer. In [18], the authors investigated the impact of queue management scheme on TCP performance and other authors proposed queue management scheme to alleviate this problems in [19]. In another study [20], the authors proposed cross-layer based TCP (X-TCP) that controls TCP windows using information the uplink flow. Their research has been designed to adjust the *cwnd* based on the estimated RTT to adapt to the mmWave band, where channel conditions change frequently. However, in their study, the bufferbloat problem was addressed efficiently, but the problem with link failure was not addressed.

TCP Cubic [21] and TCP Bottleneck Bandwidth and Round-trip Propagation Time (BBR) [22], which are based on network traffic, are the most popular TCP techniques currently used in Linux systems. TCP Cubic is a feature that increases the congestion window quickly based on the congestion event that occurred immediately before. TCP BBR also keeps the congestion window constant based on the bandwidth and RTT of the bottleneck link. Both techniques are suitable for a variety of network environments, including the mmWave band, due to the advantage of keeping the congestion window to match the net-



(a) Rural area with Manhattan mobility model

(b) Rural area with random mobility model

Figure 6: mmWave TCP case study scenario (rural area)

work conditions.

However, as shown in Section 2, mmWave CVNs have more link failure than traffic congestion. Therefore, it is obvious that mmWave TCP with wide bandwidth has more RTO due to link failure than congestion. If the link layer can recover the link before the RTO occurs, it will improve TCP performance. Fortunately, studies of the MAC layer in the mmWave band show that the deafness and beam misalignment problems can be resolved by retransmission within 10 ms, even though many nodes are crowded [2, 4]. However, the above techniques assume that the node is fixed, and the link disconnection time is dependent on the mobility of the vehicle in CVNs for blockage problem. In order to solve this problem, it is possible to consider a mobility management technique to rapidly switch to a serving base station in which a LOS is formed when a blockage problem occurs in the 5G environment [23]. However, in the base station is limited topology (e.g., rural area), the above technique cannot be a fundamental solution. Thus, the mmWave TCP technique based on mobility should be designed.

Thus, we proposed a simple TCP technique for mmWave CVNs based on vehicle's mobility. The proposed TCP solution is based on TCP Cubic [21] when packet loss event does not occur. However, our proposed TCP operates differently when packet loss event occur according to the mobility of the vehicle and channel quality information (CQI). The link layer sends its current mobility to the TCP layer, which determines the behavior of TCP in real-time[4]. Each RBS can collect CQI for each region (beam) from the vehicle (or UE) through the physical uplink shared channel (PUSCH) in 5G networks. Then, the RBS can predict the future blockage duration based on vehicle's mobility. Thus, the TCP sender can adjust the *cwnd* size appropriately based on received information from the gNB. In general, if the blockage duration is long[5], the TCP sender reduces the size of *cwnd*. Otherwise, if the time in which the blockage is experienced is short, the TCP sender holds the size of *cwnd* because network connection can be recovered in a short time. In the vehicle-to-vehicle (V2V) environment, since the blockage problem occurs relatively less, the use of conventional TCP does not cause any major problems. Thus, the proposed TCP solution can be a suitable model for CVNs.

4. Case study: TCP behavior in mmWave CVNs

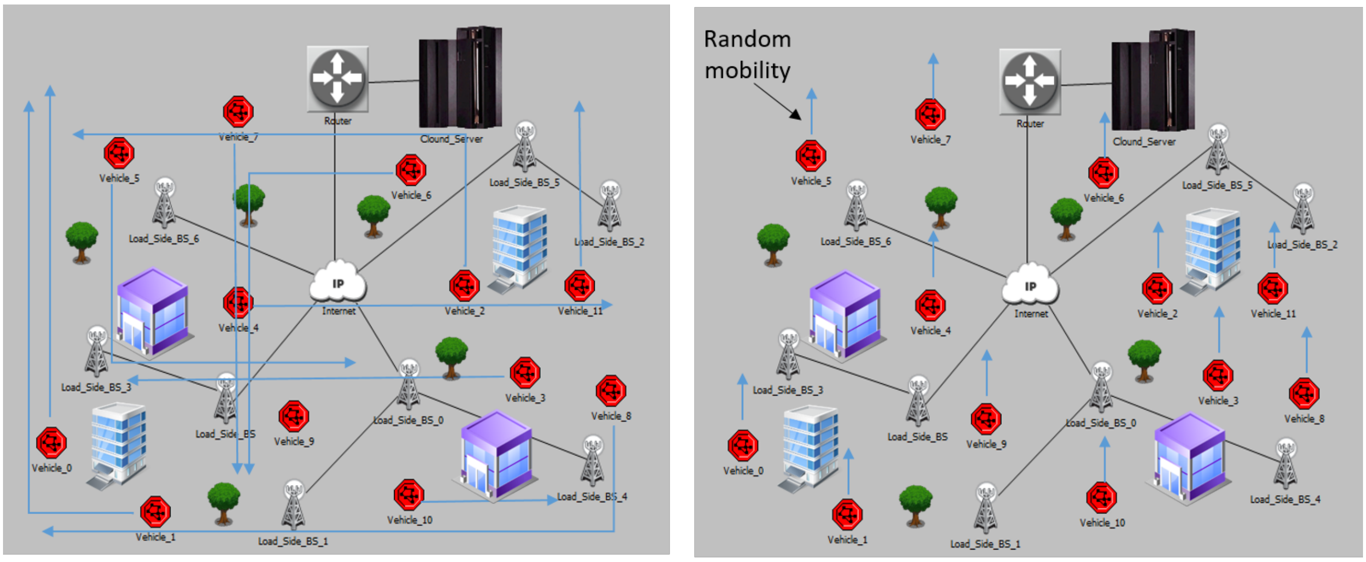
In this section, we compare the performance of existing TCP schemes and the proposed Real-Time Wireless TCP (RTW-TCP) in the mmWave CVNs environment. The simulation parameters used in this section is listed in 1. We measure total aggregate throughput and *cwnd* variation for the comparative analysis. For this purpose, the RBS and the vehicle are placed in the simulation scenario and obstacles are arranged to be similar to an actual environment. The carrier frequency is 28 GHz, the transmit power is 30 dBm, the total bandwidth is 1 GHz, and there are four TX and RX antennas. In addition, we apply the MAC scheme, which uses the beaming algorithm and the deafness aware algorithm for mmWave communication to respond to the above problems. Obstacles were placed randomly with 10-20m x 10-20m size and the speed of vehicle is set randomly to 5-30 m/s (18-108 km/h). We also apply the Manhattan mobility model and the random mobility model as vehicle motion models. The detailed scenario of the model is shown in Figures 6 and 7.

- *Rural area with Manhattan mobility model*: In this model, there are few obstacles (low blockage problem), and the

---

[4] In 5G network architecture, Mobility Management Entity (MME), which manages the mobility of the UEs connected to the core network, can perform this role.

[5] If the blockage duration is longer than retransmission timeout of TCP, it is recognized blockage duration is long.



(c) Urban area with Manhattan mobility model

(d) Urban area with random mobility model

Figure 7: mmWave TCP case study scenario (urban area)

probability of deafness and beam misalignment is relatively small, because vehicles travel along a straight path.

- *Rural area with random mobility model*: In this model, there are few obstacles (low blockage problem), and the probability of deafness and beam misalignment is relatively high, because vehicles determine their path randomly.
- *Urban area with Manhattan mobility model*: In this model, there are many obstacles, such as trees and buildings (high blockage problem), and the probability of deafness and beam misalignment is relatively, small because vehicles travel along a straight path.
- *Urban area with random mobility model*: In this model, there are many obstacles, such as trees and buildings (high blockage problem), and the probability of deafness and beam misalignment is relatively high, because vehicles determine their path randomly.

Figure 8 shows the overall network aggregate throughput versus the generated traffic. The x-axis is the normalized traffic load. For example, a traffic load of one indicates that the coming traffic requires that the sum of the rate of vehicles in the CVNs be one, as the average capacity of a single link. Thus, the overall throughput increases linearly until the traffic load is one in all simulation scenarios. In addition, when the traffic load is lower than one, the overall throughput in the rural area is about 1−2% higher than the throughput in the urban area. This is because rural areas have fewer vehicles and obstacles and, thus, fewer packet losses in the wireless channels. In addition, Manhattan mobility has a higher throughput owing to fewer link disconnections that in the case of random mobility.

When traffic is heavy in the network, throughput is slightly reduced, owing to congestion and collisions between packets.

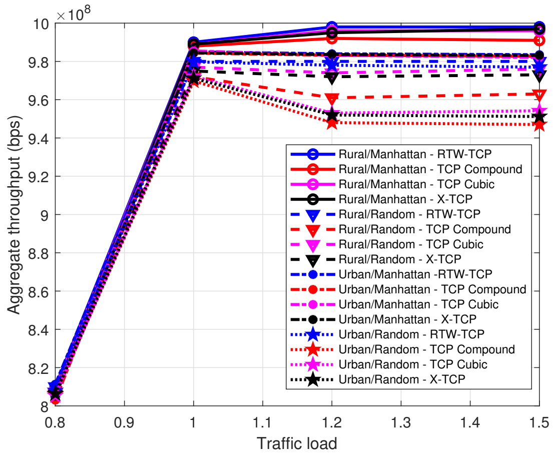


Figure 8: Aggregate throughput versus traffic load.

In particular, in the case of random movement in an urban area, we find that conventional TCP shows a 7% reduction in throughput compared to that of RTW-TCP. High-traffic environment, include the problems in the mmWave band, as well as collisions between packets and bottlenecks in the wireless link, causing many timeouts for conventional TCP schemes. On the other hand, RTW-TCP prevents timeouts between end-to-end links after resolving transient link errors by performing a fast retransmission on the wireless link. As a result, RTW-TCP can increase the throughput by maintaining the *cwnd* size, even if transient link errors occur. Basically, Manhattan mobility achieves higher throughput than random mobility. However, considering that the proposed scheme has a difference of about 2% of the performance, while the other scheme shows a difference of about 5%, the proposed scheme is less dependent on the mobility of the vehicle. In realistic road environments,

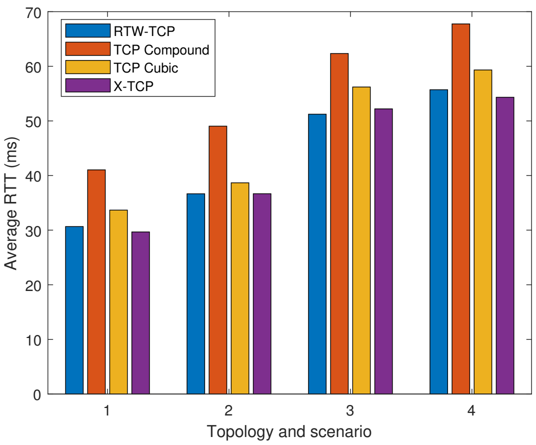


Figure 9: Round-Trip-Time (RTT) for mmWave CVNs (traffic load: 1.0).

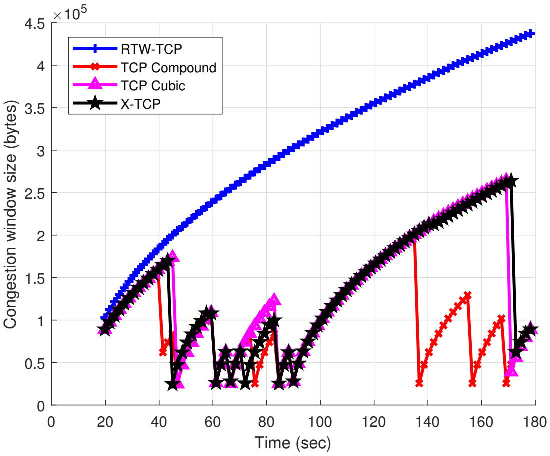


Figure 10: Congestion window size variation of TCP in CVNs (urban area with Manhattan mobility; traffic load: 1.0).

Manhattan mobility is more important because there are roads for vehicles and the direction of travel is deterministic. An interesting result is the dramatic reduction in performance in the urban-Manhattan scenario when using conventional TCPs (including mmWave X-TCP). On the other hand, when RTW-TCP is used, there is little change in performance. This means that link failures causing performance degradation are the result of more than congestion in mmWave CVNs.

Figure 9 shows the average Round-Trip-Time (RTT) for various mobility/scenario models when traffic load is 1.0. As shown in the figure, random mobility showed higher RTT than Manhattan mobility, and urban area showed higher RTT than rural area. This means that random mobility and the urban area generate more link failures. Moreover, the proposed RTW-TCP showed about 7% -10% better performance than the existing Cubic and Compound schemes. The RTT results are very similar to the throughput results, indicating that TCP schemes that do not take into account the nodes' mobility are not suitable for mmWave CVNs.

Figures 10 – 11 show the variation of *cwnd* with time. As

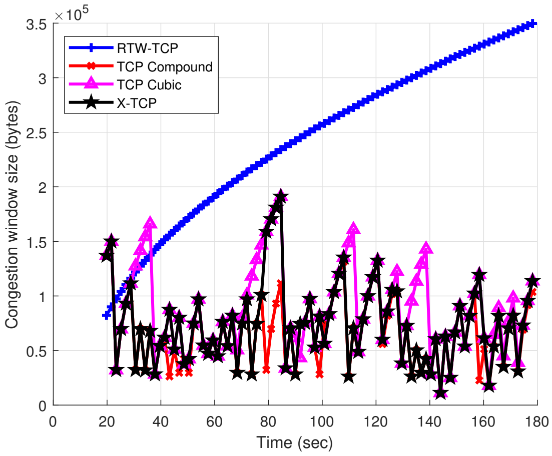


Figure 11: Congestion window size variation of TCP in CVNs (Urban area with Random mobility; traffic load: 1.0).

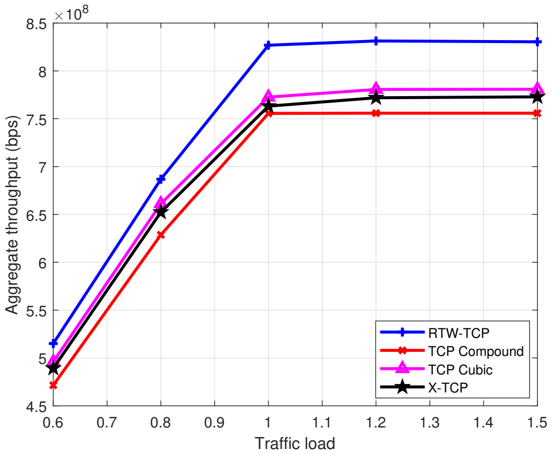


Figure 12: Aggregate throughput versus traffic load in realistic scenario.

shown, when using the conventional TCP, the *cwnd* change is largest when moving randomly in an urban area. However, the congestion delay-based TCP algorithm (RTW-TCP for low mobility), MAC layer techniques (e.g., smart beam tracking), and deafness awareness prevent RTO from occurring. In addition, in the proposed RTW-TCP technique, the TCP sender only finds out about the network congestion from the TCP receiver when the congestion occurs. Therefore, even if a blockage is temporary, *cwnd* continues to increase because network congestion does not occur in the intermediate node.

For more realistic evaluation, we conducted further experiments using traffic traces in [24, 25]. They used the traffic traces presented in the congested highway scenario. In this scenario model, vehicles move with an average speed of 50.08 km/h. Average density of vehicles is 0.119 vehicles/m and traffic generation used the ON/OFF traffic generation model to reflect realistic network traffic [26]. Figure 12 shows the average throughput in realistic scenario. In the realistic scenario, due to the relatively high moving speed and high vehicle densities, blockage due to neighboring vehicles occurs in large numbers, so there

Table 3: RTT for mmWave CVNs in realistic scenario.

	RTW-TCP	TCP Compound	TCP Cubic	X-TCP
RTT (msec)	45 msec	61 msec	47 msec	44 msec

is much performance difference between the proposed TCP and other TCPs. As you can see, the proposed TCP achieved about 11% performance improvement.

Table 4 shows RTT values measured for each scheme in realistic scenario. As shown in the table, the results are slightly higher than those of the previous simulation, and trends in the graphs are not very different. In realistic scenarios, relatively high RTT is observed because handover occurs frequently due to relatively wide and fast mobility. The proposed TCP achieves RTT similar to X-TCP which adjusts the transmission rate based on the current data rate. This shows that our technique does not cause a buffering delay (bufferbloat problem) even if it supports a large amount of throughput.

5. Conclusion

Communication via the mmWave band will be a key communication technology for future CVNs because of the very high data rate of such networks. Various PHY/MAC technologies have been studied to reflect the features of mmWave CVNs, but research on the transport layer is limited. In future mmWave CVNs, network congestion will be rare, owing to the wide bandwidth and rapidly evolving hardware. However, in the mmWave band, various link errors may occurs owing to frequency characteristics, which means conventional TCP is not suitable because it cannot distinguish these errors from network congestion. In this paper, we examined the features of existing wired/wireless TCP and whether they are suitable for mmWave CVNs. In addition, we proposed a simple wireless TCP protocol for mmWave CVNs by taking advantage of the existing wireless TCP, and simulated the performance of the proposed protocol in various simulation environments. The simulation results show that existing TCPs cannot accurately detect network congestion due to various link errors in mmWave CVNs. Even if,we have experimented with a simple technique for switching different congestion control schemes according to mobility, we have confirmed that the proposed technique efficiently uses network resources by coping appropriately. However, the actual TCP behavior of mmWave CVNs should be designed taking into account other parameters (handover delay, Tx/Rx buffer size) besides mobility or link capacity. Our future work will develop a completely new TCP protocol for mmWave CVNs, taking into account all the parameters that affect this TCP performance.

References

[1] J. Guo, B. Song, Y. He, F. R. Yu, and M. Sookhak, “A Survey on Compressed Sensing in Vehicular Infotainment Systems,” *IEEE Communications Surveys & Tutorials*, 2017.

[2] C. Perfecto, , J. Del Ser, and M. Bennis, “Millimeter-Wave V2V Communications: Distributed Association and Beam Alignment,” *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 9, pp. 2148-2162, 2017.

[3] J. Choi, V. Va, N. Gonzalez-Prelcic, R. Daniels, C. R. Bhat, and R. W. Heath “Millimeter-wave vehicular communication to support massive automotive sensing,” *IEEE Communications Magazine*, vol. 54, no. 12, pp. 160-167, 2016.

[4] W. Na, L. Park, and S. Cho, “Deafness-aware MAC Protocol for Directional Antennas in Wireless Ad Hoc Networks,” *Elsevier Ad Hoc Networks Journal*, vol. 24, Part A, pp. 121-134, 2015.

[5] M. Zhang, M. Mezzavilla, R. Ford, S. Rangan, S. Panwar, E. Mellios, D. Kong, A. Nix, and M. Zorzi, “Transport layer performance in 5G mmWave cellular,” in Proc. of *IEEE Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 730-735, 2016.

[6] Y. Zhou, V. W. Wong, and R. Schober, “Performance analysis of millimeter wave NOMA networks with beam misalignment,” in Proc. of IEEE International Conference on Communications (ICC), pp. 1-7, 2018.

[7] K. Fall and S. Floyd, “Simulation-based comparisons of Tahoe, Reno and SACK TCP,” *ACM SIGCOMM Computer Communication Review*, vol. 26, no. 3, pp. 5-21, 1996.

[8] J. C. Hoe, “Improving the start-up behavior of a congestion control scheme for TCP,” in Proc. of *ACM SIGCOMM Computer Communication Review*, Vol. 26, No. 4, pp. 270-280, Aug. 1996.

[9] C. P. Fu, and S. C. Liew, “TCP Veno: TCP enhancement for transmission over wireless access networks,” *IEEE Journal on selected areas in communications*, vol. 21, no. 2, pp. 216-228, 2003.

[10] K. Tan, J. Song, Q. Zhang, and M. Sridharan, “A compound TCP approach for high-speed and long distance networks,” in Proc. of *IEEE INFOCOM,* 2006.

[11] S. Biaz, and N. H. Vaidya, “”De-randomizing” congestion losses to improve TCP performance over wired-wireless networks,” *IEEE/ACM Transactions on networking*, no. 13, no. 3, pp. 596-608, 2005.

[12] S. Mascolo, C. Casetti, M. Gerla, M. Y. Sanadidi, and R. Wang, “TCP westwood: Bandwidth estimation for enhanced transport over wireless links,” in Proc. of *ACM 7th annual international conference on Mobile computing and networking* pp. 287-297, 2001.

[13] S. Bhandarkar, N. E. Sadry, A. N. Reddy, and N. H. Vaidya, “TCP-DCR: A novel protocol for tolerating wireless channel errors,” *IEEE Transactions on Mobile Computing*, vol. 4, no. 5, pp. 517-529, 2005.

[14] K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash, “A feedback-based scheme for improving TCP performance in ad hoc wireless networks,” *IEEE Personal communications*, vol. 8, no. 1, pp. 34-39, 2001.

[15] M. Polese, , R. Jana, and M. Zorzi, “TCP and MP-TCP in 5G mmWave Networks,” *IEEE Internet Computing*, vol. 21, no. 5, pp. 12-19, 2017.

[16] M. Pieska, and A. Kassler, “TCP performance over 5G mmWave linksTradeoff between capacity and latency,” in Proc. of *Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 385-394, 2017.

[17] R. Ford, M. Zhang, S. Dutta, M. Mezzavilla, S. Rangan, and M. Zorzi, “A framework for end-to-end evaluation of 5G mmWave cellular networks in ns-3,” in Proc. of *the Workshop on ns-3 ACM*, pp. 85-92, 2016.

[18] M. Pieska and A. Kassler, “TCP performance over 5G mmWave linksTradeoff between capacity and latency,” in Proc. of *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 385-394, 2017.

[19] M. Zhang, M. Mezzavilla, J. Zhu, S. Rangan, and S. Panwar, “TCP dynamics over mmWave links,” in Proc. of *IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1-6, 2017.

[20] T. Azzino, M. Drago, M. Polese, A. Zanella, and M. Zorzi, “X-TCP: a cross layer approach for TCP uplink flows in mmwave networks,” in Proc. of *IEEE Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)* , pp. 1-6, 2017.

[21] S. Ha, I. Rhee, and L. Xu, “CUBIC: a new TCP-friendly high-speed TCP variant,” *ACM SIGOPS operating systems review*, vol. 42, no. 5, pp. 64-74, 2008.

[22] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, “BBR: Congestion-based congestion control,” *Queue*, vol. 14, no. 5, pp. 20-53, 2016.

[23] M. Polese, M. Mezzavilla, S. Rangan, and M. Zorzi, “Mobility Man-

agement for TCP in mmWave Networks" in Proc. of *ACM Workshop on Millimeter-Wave Networks and Sensing Systems* pp. 11-16, 2017.

[24] G. Cecchini, A. Bazzi, B. M. Masini, and A. Zanella, "LTEV2Vsim: An LTE-V2V simulator for the investigation of resource allocation for cooperative awareness," in Proc. of *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, pp. 80-85, 2017.

[25] A. Bazzi, B. M. Masini, A. Zanella, and A. Calisti, "Visible light communications as a complementary technology for the internet of vehicles," *Elsevier Computer Communications*, vol. 93, pp. 39-51, 2016.

[26] A. Varet and N. Larrieu, "How to Generate Realistic Network Traffic?," in Proc. of *IEEE 38th Annual Computer Software and Applications Conference*, pp. 299-304, 2014.