Performance Enhancement of LTE Internet Access Network in Moving Vehicles

Hsu-Chieh Hu*, Jiun-Ren Lin*, Fan Bai†, Ozan K. Tonguz*
*Department of Electrical and Computer Engineering Carnegie Mellon University Pittsburgh, PA 15213-3890,USA
†General Motors Corporation, ECI Lab, Research and Development, Warren, MI 48092-2031, USA

***Abstract*—In-car Internet access using Long Term Evolution (LTE) technology has become a new trend in the automotive industry to support additional safety applications and enhanced infotainment systems on new vehicles. Car manufacturers may install an LTE modem with a WiFi router in a vehicle in order to provide ubiquitous Internet access to the passengers. However, the performance of the LTE network is highly variable due to the changing bandwidth and long latency, and the existing Transmission Control Protocol (TCP) may not be able to fully utilize the available bandwidth. Moreover, the high mobility of the vehicles can also exacerbate this issue. In this paper, we carefully investigate the performance degradation of the in-car wireless network through experimental studies, and we propose a solution to enhance the network performance. Specifically, the proposed approach utilizes direct information of the vehicle such as current speed and historical channel condition to adaptively adjust TCP parameters for optimal throughput and latency performance. Our empirical results show that the proposed approach can increase the LTE downlink throughput by 41% to 82% with different parameters, and the variance of round-trip time (RTT) of packets is significantly reduced.**

Keywords*—*Transmission Control Protocol (TCP), connected vehicles, Long Term Evolution (LTE), cellular networks

I. INTRODUCTION

As the demand of Internet access keeps increasing, car manufacturers have started to install in-cabin wireless networks that provide Internet access on production vehicles. Popular wireless broadband network technologies, such as Long Term Evolution (LTE) and Long Term Evolution-Advanced (LTE-A), are also added in order to provide better user experience. For example, starting from 2015, General Motors (GM) has started to provide the high-speed Internet access through LTE networks in their new vehicles. Fig. 1 depicts a possible network architecture of such an in-cabin wireless network. An LTE modem along with a WiFi router is installed in the vehicle, and all of the user devices (e.g., laptops, tablets, or smartphones) connect to the WiFi router to access the Internet. In fact, this architecture provides two major advantages: 1) since a larger LTE antenna can be installed on the vehicle, the Internet access provided by this architecture is more reliable than the case that each mobile handset uses its own LTE connection; 2) WiFi is more energy-efficient than direct LTE access, and hence the battery life of mobile devices could be longer. However, bandwidth under-utilization and long latency are the common issues of the LTE access from vehicles due to the highly mobile vehicular environment.

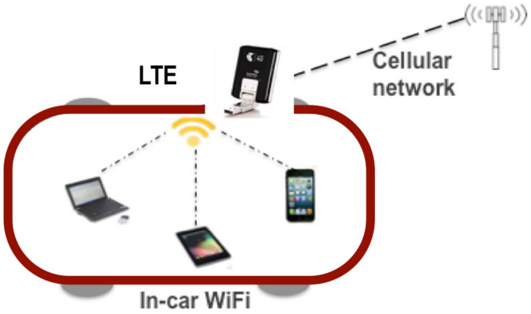


Fig. 1: In-cabin wireless networks.

The stringent latency and bandwidth requirements of real-time and steaming applications pose significant challenges to the design of in-cabin wireless networks, especially when the bandwidth of the LTE link is much smaller than the in-cabin WiFi network in the architecture shown in Fig 1. Also, in mobile networks such as LTE networks, the bandwidth and latency vary frequently at different time and locations [1]–[3]. Bandwidth usually fluctuates during handovers between base stations (BSs) due to different loading and channel conditions. If the transmitting rate at the source determined by the protocol is too low, the link will not be fully utilized. On the other hand, if the transmitting rate is too high, a large amount of packets will be queued along the way, and it results in long overall latency [4]–[6]. This latency could range from several hundred milliseconds to several seconds, which can be unacceptable for certain real-time applications. Furthermore, the vehicles can move at a high speed and in a large geographical area. Therefore, the use cases can be much more dynamic than typical LTE access from mobile devices, and a special protocol design is required to provide high quality service and great user experience, especially in the normal case that the LTE link is the throughput bottleneck.

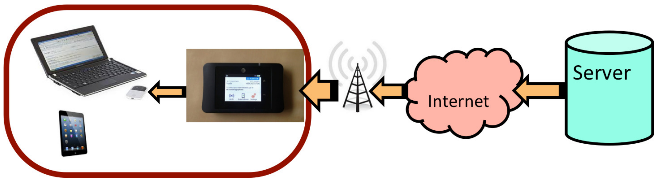
Since Transmission Control Protocol (TCP) was originally designed for wired networks, many researchers have been working on the issues of TCP over wireless networks. For example, Split TCP schemes such as Snoop protocol [7] and link-layer TCP-aware protocol [8] were proposed to enhance the performance of TCP over wireless networks. However, these techniques require special configurations on all the intermediate nodes and proxies in the network, and therefore, it is difficult to deploy them in the established mobile networks. Furthermore, car manufactures usually use the LTE modem directly provided by the LTE service provider and do not have the freedom to modify the network facilities outside the vehicles. From car manufacturers' point of view, it is important to have a solution to enhance the performance of in-cabin wireless networks without modifying the network components outside of the vehicle, which can be a difficult optimization problem. For in-cabin wireless networks, another major issue is the fact that TCP may not be able to provide reliable performance under the the highly mobile vehicular environment. Some modifications to TCP may be required in order to adjust the TCP parameters according to the variation of the observed bandwidth and latency. For instance, TCP Westwood [9] and TCP Vegas [10] measure protocol-related variables such as acknowledge (ACK) rate or round-trip time (RTT) variation and use these information to adjust the TCP parameters. However, such solutions could be slow and not effective for in-cabin wireless networks since they do not utilized the direct information related to the dynamic environment.

The major objective of this paper is to improve the performance of the current TCP in terms of both UL and DL throughput/latency for in-cabin wireless networks. Compared to the backbone networks of the Internet and the WiFi network inside the vehicle, the LTE link is often the bottleneck in the system, and the LTE bandwidth directly affects the throughput and delay that the users experience. To optimize the overall throughput and latency performance of in-cabin wireless network, we utilize the information related to the vehicle, e.g., current vehicle speed, and the measured RTT ratio to adjust the receive window and congestion window size accordingly. With the speed and channel information, our modified TCP can adapt to the variance of mobile networks in a timely manner. Based on our extensive experiments, we will show that the proposed method can achieve up to 55% throughput enhancement that is experienced by the users in the vehicle and maintain the delay in a reasonable range at the same time. The modified TCP proposed in this paper is shown to be efficient and effective, and it does not require any change on the intermediate nodes in the networks.

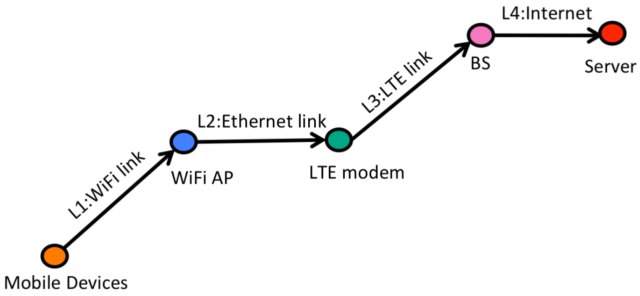
The remainder of this paper is organized as follows. Section II describes the detailed system model used in this paper. Section III discusses the bufferbloat and the throughput throttling problems, which are important issues related to the end-to-end performance of in-car wireless networks. Section IV introduces our novel adaptive window adjustment algorithm for improving the performance of in-cabin wireless networks. Section V provides the experimental results on the throughput and delay of LTE downlink (DL) and uplink (UL). Finally, the conclusions of this paper are given in Section VI.

II. SYSTEM MODEL

The system model of in-cabin wireless networks discussed in this paper is shown in Fig. 2; Fig. 2(a) depicts the actual components of our experimental testbed, and Fig. 2(b) conceptualizes the communication links among the components of the in-cabin wireless network. As briefly mentioned in Section I, the in-cabin wireless network consists of a WiFi router and an LTE modem. However, In the experimental testbed, the WiFi router and the LTE modem are both implemented in the single LTE hotspot device. In the DL, LTE modem relays packets to the WiFi AP through the wired connection. The WiFi AP then forwards the packets to the corresponding mobile devices in the vehicle. The UL communications work in a similar manner but in the reverse direction. At the time this paper was written, LTE release 8 could, in principle, support a maximum data rate of 75 Mbps in the DL and 37 Mbps in the UL with 10 MHz bandwidth and one single antenna. However, in reality, the average throughput that we measured on our testbed is around 13 Mbps in the DL and 6 Mbps in the UL due to network loading and propagation effect. It is also worth pointing out that, in our system model, the LTE link is usually the bottleneck of the in-cabin wireless networks.



(a) System components



(b) Communication links

Fig. 2: The system model of in-cabin wireless networks

III. THROUGHPUT THROTTLING AND BUFFERBLOAT

TCP was originally designed for wired networks, and it could encounter severe throughput throttling when being used in a highly mobile environment. This issue can be discussed from three different aspects: the throttling in the DL, the throttling in the UL, and the bufferbloat problem.

In the DL, the size of TCP congestion window at the sender side is determined by both the receive window and the TCP-related congestion algorithms such as the popular TCP CUBIC [11]. In recent research [5], it has been shown that the unfitting receive window can throttle 52.6% of DL flows. Therefore, a suitable receive window adjustment scheme is highly desirable in order to prevent underutilizing the available bandwidth from the server side. The DL throttling problem can be much more severe in the vehicular environment because of the larger bandwidth variation resulting from frequent handovers or changes of channel condition. For the vehicular environment, a method to adaptively adjust the receive window is needed to provide reliable performance under the large bandwidth variation.

In the UL, when packets are dropped, TCP would halve the congestion window since it presumes that congestion is the cause of lost packets. However, for in-cabin wireless networks, it is not always the case: the packets might be dropped due to bad channel condition. In such case, the available bandwidth of the link is actually still sufficient, and therefore, reducing the congestion window would cause throttling. To solve this issue, TCP Westwood was proposed, and it recalculates the congestion window size according to the ACK rate [9]. However, it does not work well when TCP applies delay ACK feature [12]. Therefore, we need an adaptive algorithm and a new indicator to adjust the congestion window when packets are dropped. Furthermore, it is also important to keep increasing the size of congestion window when random packet errors are observed as it would help utilize all of the available bandwidth in the UL.

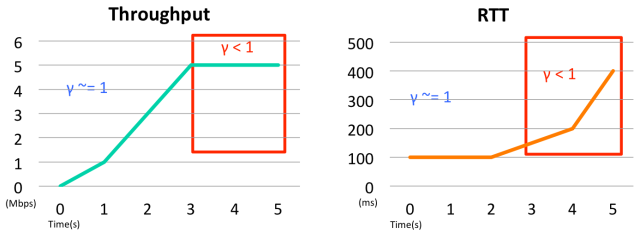
The chance of congestion happening in modern mobile networks is actually very small since most of the network devices are equipped with large buffers. However, the oversized buffers might result in extremely long queueing delay and performance degradation. For example, if a node (e.g., an LTE device in a vehicle) advertises a large receive window size in order to avoid throughput throttling, it is possible that the packet source would send too many packets that exceed the network capacity. In such case, those packets would be queued in the buffers along the way instead of being dropped, and the RTT will increase as the queueing delay grows. It has been shown that with small RTT, TCP can utilize over 95% of available bandwidth. On the other hand, when RTT is over 400 ms, the utilization ratio drops below 50%, which results in very bad performance. These observations show that longer RTT can significantly degrade the TCP performance in LTE networks.

IV. THE ADAPTIVE WINDOW ADJUSTMENT ALGORITHMS FOR IN-CABIN WIRELESS NETWORK

To address the throttling problem for in-cabin wireless networks, we propose an adaptive algorithm, which adjusts the size of both the receive window and the congestion window according to direct information related to the vehicle. As mentioned in the previous section, a small receive window could throttle throughput, whereas a large receive window could introduce the bufferbloat problem. Therefore, it is critical to select an optimal receive window size. The proposed algorithm is applicable to both the DL and the UL.

A. Adaptive Receive Window Determination Algorithm (ARWND)

In the current design of TCP, clients advertise their receive window size to the server, so the server can use the information to determine the suitable congestion window size. However, the available bandwidth of mobile networks usually fluctuates over time, and it is difficult for the clients to set an optimal receive window size. If the receive window is too large, the delay will increase due to the bufferbloat phenomenon. On the other hand, a small receive window could limit the throughput. In the experiments, we observed that as the size of receive window increases, a long RTT that is larger than 400 ms can be observed. Therefore, it is possible to determine the optimal receive window size based on historical RTT measurements. In other words, the historical record of RTT variation could help us better estimate the current network status.



(a) Relation between γ and throughput (b) Relation between γ and delay

Fig. 3: An example that demonstrates the physical interpretation of γ

The RTT ratio γ is defined as

$$\gamma = \frac{R_m}{R_t}, \tag{1}$$

where R_m is the minimal RTT observed and R_t represents the latest RTT measurement. R_m is calculated from the previous collected RTT samples within a predefined period, and it is used to estimate the RTT without any congestion in the network. γ can indicate the instant status of bandwidth utilization. If γ is close to one, it implies that the R_t is close to R_m. In such case, the RTT mainly consists of propagation delay and transmission delay, and it means that the queueing delay is very small. Therefore, it is possible to increase the receive window size in order to fully utilize the available bandwidth. In contrast, if γ is smaller than one, the queues along the LTE link start to accumulate packets, which leads a larger RTT. Under such situation, the throughput of the in-cabin wireless networks would degrade due to the long end-to-end delay. For better understanding of the physical interpretation of γ, Fig. 3(a) shows the relation between γ and throughput, and Fig. 3(b) shows the relation between γ and delay. The proposed algorithm will not decrease window size unless RTT is larger than a reasonable value, which could be multiple times of R_m. It ensures stable throughput during communications. However, if the delay grows significantly to an unacceptable level, the receive window size will be decreased by γ. Through the proposed adaptive algorithm, the advertised receive window would be maintained at a proper size. The flowchart of the proposed algorithm is depicted in the Fig. 4, which shows the detailed steps of the algorithm.

B. Incorporating Direct Information Related to the Vehicle

For in-cabin wireless networks, the available bandwidth from the LTE link is usually affected by the vehicle speed and the link condition. For example, the available bandwidth is smaller when the vehicle is traveling at a high speed or when the channel is none-line-of-sight (NLoS); the available bandwidth is larger when the speed is low or when the channel is line-of-sight (LoS). Therefore, in the proposed algorithm, in

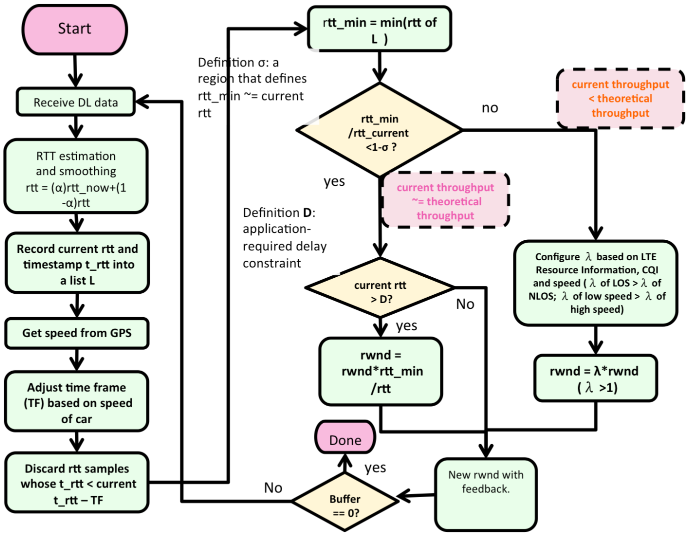


Fig. 4: Adaptive receive window determination algorithm

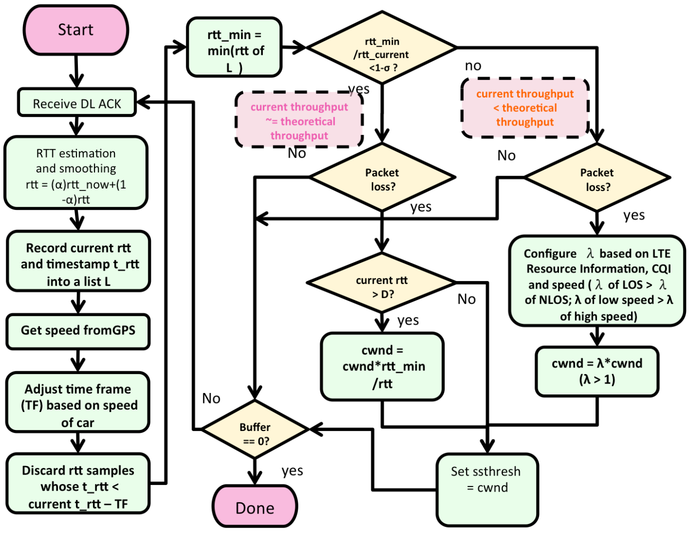


Fig. 5: Adaptive congestion window determination algorithm

order to adjust the receive window size more efficiently, we incorporate the information of vehicle speed and link condition. Consider the case when γ is close to 1. The proposed algorithm would increase the receive windows size to utilize the available bandwidth. At the moment, if the LTE modem detects a LoS channel or when the car is at a lower speed, a larger multiplier, i.e., λ, will be used to increase the receive window size. It means that the receive window size will increase at a higher rate to quickly reach the optimal level. On the other hand, if a NLoS channel is detected or the vehicle speed is high, a smaller and less aggressive multiplier will be applied.

As shown in Fig. 4, a large value of λ, which can be larger than 1, e.g., among 2–3, is chosen under a favorable channel condition or a low speed to rapidly increase the receive window size. This approach can effectively reduce the iterations needed to achieve the optimal window size. Similarly, a smaller value of λ, e.g., 1, is applied under a poor channel condition or a high speed to avoid excess packets being injected to the network.

C. Adaptive Congestion Window Determination Algorithm (ACWND)

In the UL, the traditional TCP CUBIC halves the congestion window size when some packets are dropped. However, for in-cabin wireless networks, a significant amount of lost packets are caused by bad link condition or handovers instead of congestion. Therefore, halving the window size without knowing the actual reason of the packet loss is not appropriate and can hurt the throughput significantly. To address this issue, we apply a similar technique as shown in Fig. 5, which uses the RTT ratio γ to recalculate the congestion window when the packets are lost. As γ is close to 1, the proposed algorithm would increase congestion window size to utilize the available bandwidth. Different values of multiplier λ are chosen under various conditions, which is similar to DL algorithm.

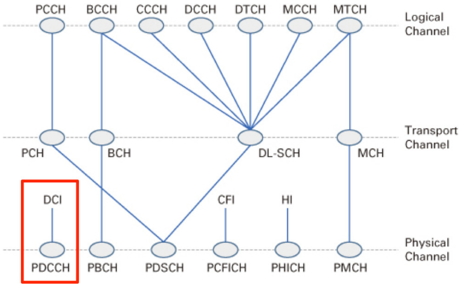
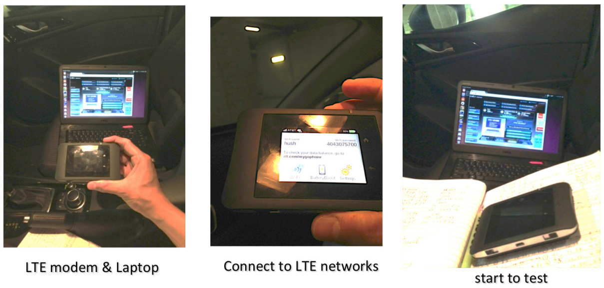


Fig. 6: LTE control information embedded in PDCCH

D. Incorporating LTE Control Information

According to our observations, the available bandwidth of LTE networks is highly variable, and the bandwidth could be fully utilized if the TCP window is configured to a correct size quickly and accurately. In fact, the current LTE standard provides another amenity that could improve the performance [13] – the LTE control channels contain related information including the DL resource allocations and the UL scheduling requests. For the DL, the control information of one or multiple mobile devices is contained in the Downlink Control Information (DCI) and is transmitted on the Physical Downlink Control Channel (PDCCH) [14] shown in Fig. 6. The DCI contains radio resource assignments, frequency hopping patterns, and modulation/coding schemes, which allow the devices to decode the data information on the data channels. Therefore, the control information can specify the bandwidth assigned by the BS, and the TCP window can be configured optimally by monitoring the control channels. To achieve this goal, the LTE physical layer needs to provide the transport layer with APIs to access the control information.



LTE modem & Laptop

Connect to LTE networks

start to test

Fig. 7: Experimental testbed

V. Experimental Results

A. Experimental Testbed

The testbed used in the experiments of this paper is shown in Fig. 7. It comprises a 4G LTE-WiFi hotspot, a laptop, and a Linux server. The hotspot combines a WiFi router and a LTE modem that supports AT&T's LTE network (LTE release 8). The hotspot provides Internet access through the LTE network to other WiFi clients in the vehicle. On the testbed, the RTT of the packets between the laptop in the vehicle and the server at Carnegie Mellon University is about 80 ms. We installed Linux Ubuntu version 14.04 as the operating system on the laptop, which represents the user device in the vehicle. Then we modified its TCP stack (i.e., the latest TCP CUBIC) in both receiver (DL) and sender (UL) parts and incorporated the proposed algorithm. It is worth pointing out that similar modifications could be easily done on a smartphone. The experiments were conducted in the urban area of Pittsburgh, PA, USA, where the AT&T LTE network provides 99% coverage in the area. We performed the same procedure three times during evening rush hours on three different days and average the results to make sure our results are statistically significant.

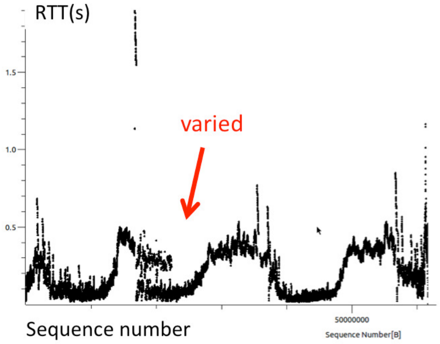
B. Experimental Results

Table I presents the observed throughput of the default TCP CUBIC and the proposed algorithms in a static environment. It shows that the proposed algorithm can provide a considerable improvement on the throughput of both UL and DL. For the DL, the maximum throughput gain could be up to 35%. With the proposed algorithm, since the receive window size is adjusted according to the RTT ratio, throughput is not throttled as in the case of the default TCP.

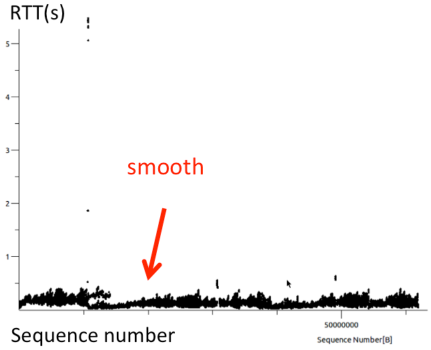
The results shows an 8% gain of throughput in the UL. The UL traffic is usually lighter and less variable than the DL traffic. Therefore, the gain we can achieve is smaller than the case of DL. In addition to the throughput, the proposed algorithm can keep the RTT within a range between 100 ms–200 ms, which is an acceptable range for most real-time applications. The observed RTT is shown in Fig. 8, and it can be seen that the variance of RTT is much smaller when the proposed algorithm is in place.

TABLE I: UL and DL throughput of the default TCP and the proposed algorithm

Version	Direction	Throughput (Mbps)
Default (TCP CUBIC)	UL	10.604
	DL	10.766
Proposed Algorithm (ARWND+ACWND)	UL	11.46
	DL	14.628



(a) RTT observed with the default TCP



(b) RTT observed with the proposed algorithm

Fig. 8: RTT observed in the experiments

C. Incorporating Vehicular Information

As discussed in Section IV-B, the parameter λ can be adjusted according to the direct information related to the vehicle in order to further enhance the benefits of the proposed algorithm. Fig. 9 shows the DL throughput improvement under two different channel conditions. On the left-hand-side of the figure, the Reference Signal Received Power (RSRP) is -81 dBm, and on the other side, the RSRP is -110 dBm. A higher RSRP (i.e, -81 dBm) implies a better channel condition. It can be observed that for the better channel condition, the proposed algorithm with a larger λ, i.e., 3, brings more throughput enhancement, whereas a smaller λ, i.e., 2, would be preferable for the worse channel condition. Furthermore, Fig. 10 shows the DL throughput improvement of the proposed algorithm when the vehicle is traveling at two different ranges of speed. Similarly, we can observe that a larger λ can bring more throughput enhancement to the DL when the vehicle is at a lower speed, whereas a smaller λ in the proposed algorithm

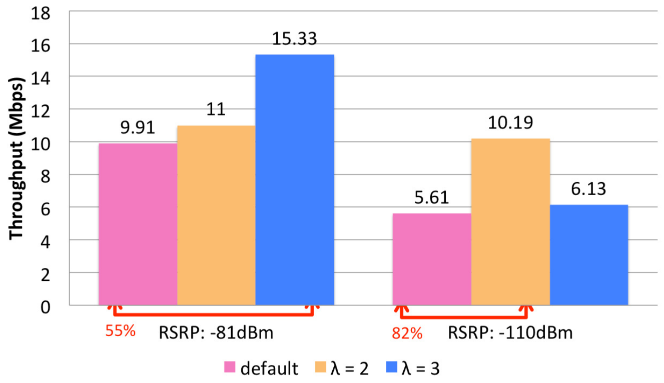


Fig. 9: DL throughput improvement under different channel conditions

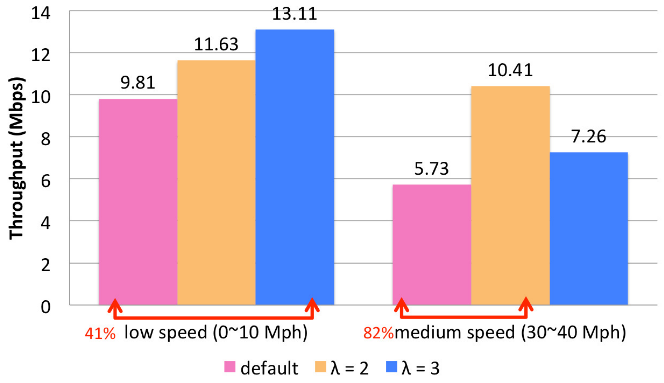


Fig. 10: DL throughput improvement under different vehicle speeds

performs better when the vehicle speed is higher.

VI. CONCLUSION AND FUTURE RESEARCH

Due to the increasing demand of Internet access while traveling, car manufacturers have started to install in-cabin wireless networks with WiFi and LTE technologies in production vehicles. However, since the vehicles are highly mobile, the LTE link can be unstable and is often the bottleneck of the communications. Furthermore, the mechanism of the current TCP might not be able to provide optimal performance under the highly variable LTE link condition. In this paper, we study the challenges of providing reliable Internet access through in-cabin wireless networks and propose a new approach to improve the performance of current TCP under the use case. Our modified TCP incorporates the information directly related to the vehicle and the RTT ratio, and it can adaptively adjust the parameters such as the congestion window size to optimize the overall throughput and delay. Through our extensive experimental studies, we demonstrate that the additional vehicular information, e.g., current speed, and the channel condition can effectively help improve the performance of in-cabin wireless networks in connected vehicles. With the proposed approach, the Internet users in the vehicle can experience significantly higher network throughput as well as shorter latency. Furthermore, it can be seen from the experimental results that for in-cabin wireless networks, our approach outperforms a generic TCP protocol, which does not apply any specific characteristics of the vehicular environment. With the promising results reported in this paper in mind, it could be very beneficial to incorporate our approach to TCP and even evolve to a completely new TCP framework that is tailored to connected vehicles in the future.

REFERENCES

[1] K. Lai and M. Baker, “Measuring bandwidth,” in *IEEE INFOCOM ’99. Conf. Computer Commun.*, vol. 1, 1999, pp. 235–245.

[2] R. Prasad, C. Dovrolis, M. Murray, and K. C. Claffy, “Bandwidth estimation: metrics, measurement mechniques, and tools,” *IEEE Network*, vol. 17, no. 6, pp. 27–35, 2003.

[3] J. Gettys, “Bufferbloat: dark buffers in the internet,” *Queue*, vol. 15, no. 3, pp. 96–96, 2011.

[4] H. Jiang, Y. Wang, K. Lee, and I. Rhee, “Tackling bufferbloat in 3G/4G mobile networks,” in *Proceedings of the 2012 ACM Conf. on Internet Meas.*, 2012.

[5] J. Huang, F. Qian, Y. Guo, Y. Zhou, Q. Xu, Z. M. Mao, and O. Spatscheck, “An in-depth study of LTE: effect of network protocol and application behavior on performance.” in *Proceedings of the ACM SIGCOMM 2013 Conf.*, 2013, pp. 364–374.

[6] K. Nichols and V. Jacobson, “Controlling queue delay,” *Communications of the ACM*, vol. 55, no. 7, pp. 42–50, 2012.

[7] H. Balakrishnan, S. Seshan, E. Amir, and R. H. Katz, “Improving TCP/IP performance over wireless networks,” in *MobiCom*, vol. 95, 1995, pp. 2–11.

[8] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, and R. H. Katz, “A comparison of mechanisms for improving TCP performance over wireless links.” *IEEE/ACM Trans. on Networking*, vol. 5, no. 6, pp. 756–769, 1997.

[9] S. Mascolo, C. Casetti, M. Gerla, M. Y. Sanadidi, and R. Wang, “TCP Westwood: bandwidth estimation for enhanced transport over wireless links,” in *Proceedings of the 7th annual international conference on Mobile computing and networking*, 2001, pp. 287–297.

[10] L. S. Brakmo and L. L. Peterson, “TCP Vegas: End to end congestion avoidance on a global internet.” *IEEE J. Sel. Areas Commun.*, vol. 13, no. 8, pp. 1465–1480, 1995.

[11] 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.

[12] J. Chen, G. M., Y. Lee, and M. Y. Sanadidi, “Tcp with delayed ack for wireless networks,” *Ad Hoc Networks*, vol. 6, no. 7, pp. 1098–1116, 2008.

[13] S. Kumar, E. Hamed, D. Katabi, and L. Erran Li, “Lte radio analytics made easy and accessible,” in *Proceedings of the 2014 ACM conference on SIGCOMM*. ACM, 2014, pp. 211–222.

[14] 3rd Generation Partnership Project. E-UTRA, *Multiplexing and channel coding, TS 36.212, v9.4*, 2011.