A Measurement Study on TCP Behaviors in HSPA+ Networks on High-speed Rails

Li Li*†, Ke Xu*†, Dan Wang‡§, Chunyi Peng¶, Qingyang Xiao*† and Rashid Mijumbi‖
*Tsinghua National Laboratory for Information Science and Technology, Beijing, China
†Department of Computer Science and Technology, Tsinghua University, Beijing, China
‡Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China
§The Hong Kong Polytechnic University Shenzhen Research Institute, Shenzhen, China
¶Department of Computer Science and Engineering, The Ohio State University, Columbus, USA
‖Telematics Engineering Department, Universitat Politècnica de Catalunya, Barcelona, Spain
Email: ll-12@mails.tsinghua.edu.cn, xuke@tsinghua.edu.cn, csdwang@comp.polyu.edu.hk
chunyi@cse.ohio-state.edu, xqy13@mails.tsinghua.edu.cn and rashid@tsc.upc.edu

Abstract—**TCP has been the dominant transport protocol for mobile internet since its origin. Its behaviors play an essential role in determining quality of service/experience (QoS and QoE) for mobile apps. While TCP has been extensively studied in a static, walking, or driving mobility, it has not been well explored in high-speed (> 200 km/h) mobility cases. With increasing investment and deployment of high speed rails (HSRs), a critical demand of understanding TCP performance under extremely high-speed mobility arises. In this paper, we conduct an in-depth study to investigate TCP behaviors on HSR. We collect 90 GB of measurement data on HSPA+ networks in Chinese high-speed trains with a peak speed of 310 km/h, along various routes (covering 5,000 km) during an 8-month period. We analyze the impacts of high-speed mobility and handoff on performance metrics including RTT, packet loss and network disconnection. Then we demystify the grand challenges posed on TCP operations (TCP establishment, transmission, congestion control and termination). Our study shows that performance greatly declines in HSR, where RTT spikes, packet drops and network disconnections are more significant and occur more frequently, compared with static, slowly moving or driving mobility cases. Moreover, TCP fails to adapt well to such extremely high-speed and yields severely abnormal behaviors, such as high spurious RTO rate, aggressive congestion window reduction, long delay of connection establishment and closure, and transmission interruption. All these findings indicate that extremely high-speed indeed poses a big threat to today's TCP and it calls for urgent efforts to develop HSR-friendly protocols and wireless networks to address even more complicated challenges raised by faster trains/aircrafts in the foreseeable future.**

I. Introduction

It is well known that TCP is the core protocol to translate lower layer advances, or mask lower layer problems, to application layer performance. Many works have studied TCP performance in different scenarios, such as static, slowly moving, driving, subway and ferry [1]–[12]. However, the performance and behaviors of TCP are still quite uncertain in high-speed (>200 km/h) mobility cases. In the past few years, we have witnessed a significant worldwide development of HSR, reaching 22,000 km at end of 2013 [13]. In particular, China contributes more than a half of the world's HSR network in terms of length. Currently, the HSR speed in China reaches 310 km/h. It is natural to ask: how well can TCP perform under such physical scenarios?

In this paper, we make a contribution in this area by performing a comprehensive measurement to investigate TCP performance and behaviors on HSRs. Since, there is no long-distance deployment of 4G LTE network along HSR lines in China, we conduct measurements on HSPA+ (3.75G) [14] cellular networks of a major Chinese carrier, which has good coverage along long-distance railway lines nationwide. We carry out an 8-month measurement in the high-speed trains of four routes in China, with speeds reaching 310 km/h. We have covered a total distance of 5,000 km and collected more than 90 GB of data.

The main difference between our work and most previous studies lies in not only the high-speed scenarios we study and the large scale (covering a long distance) nature of our measurements, but also our focus on the following two issues:

- What is the main challenge brought by HSR to TCP? While the impact of mobility has been studied in prior works [7]–[12], but most of them are based on slowly moving mobile settings of end devices. Another deficiency is that these studies do not quantitatively analyze the independent effects (i.e considered separately) of mobility and handoff on TCP. It is therefore necessary to perform further studies on the challenges brought by high speed motion to TCP.
- Can TCP adapt well to the challenge? If not, what abnormal behaviors does it show? The adaptability has never been fully studied for all TCP aspects including connection establishment, transmission, congestion control and connection closure even in slowly moving mobile cases, not to mention high speed motion mobile scenarios. Besides, no prior study compares the performance among flows of various sizes in mobile scenarios. Then it is necessary to evaluate the adaptability of all aspects of TCP to high-speed motion and study the effects of flow size as well.

978-1-4799-8381-0/15/$31.00

To the best of our knowledge, this paper is the first comprehensive study on above two questions based on a real, large-scale measurement in high-speed mobile cases. In particular, there are two main challenges that this study has to overcome:

1) Since the carrier used in our measurements deploys various types of networks such as HSPA+ (3.75G), HSDPA (3.5G), UMTS (3G), EDGE (2.75G), and GPRS (2.5G) along the railway, the mobile device is connected to HSPA+ only in 70% of travel time. This change in network type is intertwined with mobility and handoff, making it very difficult to determine whether the TCP performance degradation is caused by network type change, mobility or handoff. To overcome this, we choose short-lived flows to perform the measurements because long-lived flows probably experience different types of networks during transmission.

2) Even in a pure HSPA+ network, mobility and handoff are intertwined, making it difficult to analyze the effect of each factor at the same time. For this reason, we compare the performance among TCP flows that suffer no handoff when the train runs at various speeds (parking, acceleration, full-speed running and deceleration) to study the effects of speed. Similarly, we study the effect of the number of handoffs when the train runs at a relatively stable high speed to quantitatively analyze the impacts of handoff.

Using these methods, we first investigate the main challenges brought by high-speed trains and then evaluate the adaptability of TCP to these challenges in almost all aspects, focusing on abnormal behaviors in each aspect. Finally, the effect of flow size is also studied. We mainly make the following findings:

- With an increase in speed and number of handoffs that a flow suffers, RTT and packet loss rate rise sharply and vary in a much wider range. For example, if a 3-minute flow suffers 11 to 16 handoffs when train speed is over 280 km/h, packet loss rate reaches 100% at a probability of 42%, RTT exceeds 4 s at a probability of 10%. Besides, due to handoff failures, passengers suffer repeated network disconnections, which can lead to TCP transmission interruption. In summary, RTT spikes, packet losses, and network disconnections are more significant and occur more frequently than in static or slowly moving mobile cases, which is the main challenge brought by HSR to TCP.
- The whole TCP spectrum shows serious inadaptability on HSR, including TCP establishment, transmission, congestion control and termination. Due to wide RTT variations, the spurious RTO rate is rather high, leading to many undesirable slow starts. Affected by frequent packet losses, TCP suffers a very aggressive congestion window (CWND) reduction. Furthermore, TCP encounters significant trouble in establishing or closing a connection. For example, if a 50 KB-sized flow suffers 1 to 9 handoffs when train speed is over 280 km/h, the probability of spending over 10 s to close the connection is as high as 40%. In addition, a big portion of connections are not closed normally by handshakes but RST, network disconnection or timeout, and among these connections, most are even closed before a file is completely transmitted, wasting time and energy.
- On HSR, big flows suffer more serious performance degradation than small flows, face much higher risk of transmission failures, and can no longer show an overwhelming advantage in CWND over small flows compared to stationary scenarios.
- Compared with high-speed mobility itself, frequent handoff brought by fast motion contributes more to TCP performance degradation. We observe that dense base station deployment in urban areas is a double-edged sword, which reduces the range of cells, improves network coverage and capability for stationary and slowly moving mobile users, but increases handoff frequency for fast moving mobile users, significantly hurting TCP performance.

All of these findings indicate that high-speed trains indeed pose a big threat to TCP, and it is urgent to develop HSR-friendly wireless networks and protocols to deal with even more complicated challenges brought by faster trains/aircrafts in the foreseeable future.

The rest of this paper is structured as follows. Section II covers related work. Section III describes the measurement method and data set. We investigate the main challenges brought by HSR to TCP in Section IV. Section V discusses abnormal behaviors in almost all aspects and effects of flow size before concluding the paper in Section VI.

II. Related Work

There are some theoretical studies on TCP in mobile scenarios. Pacifico et al. [7] determine that intra handoff in LTE during motion can hurt TCP performance significantly and propose an improvement mechanism. Wang et al. [8] design a fast adaptive congestion control scheme for improving TCP performance in soft vertical handoff between WLAN and 3G networks during the mobility of users. However, in real mobile environments, theoretical models are hard to formulate, so these theoretical studies are hard to apply.

There are multiple measurements in slowly moving mobile cases at speeds below 100 km/h. Litjens [9] evaluates data transfer performance in a UMTS/HSDPA network, with a principal focus on the impact of terminal mobility. Yao et al. [10] measure bandwidth and Derksen et al. [11] measure average downlink throughput in HSDPA networks in mobile vehicles. Tso et al. [12] conduct extensive measurements in HSDPA networks on trains, subways, self-driving vehicles, buses and ferries in Hong Kong, focusing on RTT and throughput performance and also explore impacts of mobility and handoff.

Merz et al. [15] measure LTE networks in trains with a peak speed of 200 km/h, finding that such a high speed causes significant negative effects on network performance.

As for speeds up to 300 km/h, only a few short-distanced measurements have been performed. Xiao et al. [16] measure LTE networks along 115 km high-speed railway, finding that TCP throughput and RTT are not only worse, but also have a large variance compared to the stationary and driving (100 km/h) scenarios. Jang et al. [17] analyze downlink throughput and ACK compression rate of TCP flows in CDMA-EVDO networks in 300 km/h trains, covering 450 km railway.

All of these measurements in slowly and high-speed moving mobile cases mainly show statistics of metrics such as throughput, RTT, packet loss rate and bandwidth, without quantitatively analyzing the effects of mobility and handoff independently and clearly. These studies neither fully study the behaviors of TCP in all aspects including TCP establishment, transmission, congestion control and closure, nor compare performance among flows with various sizes.

To summarize, the main difference between our work and all these studies lies not only in the high-speed motion scenarios we focus on and the large scale nature of our measurements, but also in our contributions towards exposing the challenges posed by HSR to TCP, as well as an evaluation of the adaptability of all aspects of TCP and a study on the impacts of flow size on TCP performance.

III. Measurement And Data Set

We perform measurements on a cellular network of a major carrier in China, which has 450 million users nationwide. In this Section, we first introduce the measurement setup, and then present the data set in detail.

A. Measurement Setup

We have developed a measurement tool, *MobiNet* [16], which consists of a client and a server program. The client program can run on mobile devices with Android OS and the server program can run on computers with either Windows or Linux OS. The client and server transmit TCP traffic in a client/server mode, and the data transmission algorithm is based on that of *Iperf* [18]. Besides data transmission, the client program can also get geographical location and speed of the train via GPS, read signal strength, network type, Location Area Code (LAC) and Cell ID (CID) of base stations from the Android OS, and record all of above information in log files.

We carry a Samsung Galaxy IV smart phone with Android 4.2 OS on HSR, and deploy a server running Ubuntu 12.04 with 3.2.0-36-generic Linux kernel in the backbone of CERNET [19], which is a dedicated education and research network that interconnects research institutes and universities in China. It is lightly loaded, directly connected to the core network of the carrier we measure, so the performance degradation from cross-traffic is expected to be minimal, which is helpful for measuring HSR's effects on TCP. The phone runs a client program of *MobiNet*, and the server runs the server program. We capture all the packets on both the phone and the server with *tcpdump* and *wireshark* respectively.

B. Data Set

From December 2013 to July 2014, we conducted experiments on 4 routes: Beijing-Guangzhou (B-G) line, Changsha-Shanghai (C-S) line, Shanghai-Beijing (S-B) line and Beijing-Tianjin (B-T) line. We define flow size as the total number of payload bytes within the flow (excluding IP/transport layer headers). Three types of TCP downlink flows are measured, including flows of 3-minute duration as well those of size 50 KB and 2 MB. The phone and the server only establish a single TCP connection, rather than multiple connections at the same time.

Table I shows the data set in detail. As an example, the length of the B-G line is 2,298 km, and the train stops at 17 stations along the line. We take 10 one-way trips on the line and the duration of each trip is 9.7 hours (excluding the parking time at the originating and terminal stations). The number of connections established when measuring 3-minute, 50 KB-sized and 2 MB-sized flows are 1,397, 17,410 and 3,722, and the total size of packets captured on both the server and the phone, and log files of *MobiNet* when testing each type of flow is 17.3, 1.6 and 12.5 GB respectively.

Through analysis of log files of *MobiNet*, we observe variations in train speed and network type.

1) Variation in train speed. Trains experience 4 phases of motion: parking at stations, acceleration, full-speed running and deceleration. As shown in Table I, in 10 trips on the B-G line, the train parks at 17 stations (including the originating and terminal station) along the line for 16.9 hours, and the train runs at a speed between 0 and 150 km/h, between 150 and 280 km/h, and between 280 and 310 km/h for 24.5, 19.5 and 46.3 hours respectively. Figure 1 shows the average proportion of various motion speeds on all the four HSR routes. Variation in train speed allows us to quantify the effects of motion speed on TCP. In each one-way trip, we get on the train and start the measurement about 30 minutes before the train leaves the originating station, and stay on the train for about 30 minutes after the train arrives at the terminal to continue the measurement. Although the duration of the B-T line is very short, we are able to perform static measurements for 44.5 hours at the originating and terminal stations in 52 trips.

2) Variation in network type. We observed that various types of network belonging to the carrier we measure coexist along the railway, including HSPA+ (3.75G), HSDPA (3.5G), UMTS (3G), EDGE (2.75G), and GPRS (2.5G). Figure 2 shows the average proportion of time that the phone connects to each type of network. Since the focus of this paper is the effect of high-speed mobility and handoff on TCP, rather than the effect of hybrid networks, we only analyze TCP flows transmitted in pure HSPA+ networks. We do not measure long-lived bulk flows but the 3 types in Table I because long-lived flows may experience different types of networks during transmission, not in a pure HSPA+ network.

IV. Challenges Posed To TCP

In this Section we first study the impact of high-speed mobility and handoff on packet loss rate and RTT, then analyze network disconnection due to handoff failure, and finally summarize the challenges posed by high speed trains to TCP.

High-speed trains can affect TCP in two ways. On one hand, due to Doppler frequency shift and fast multi-path fading, high-speed movement itself can cause fast signal fading [20], [21],

TABLE I. DATA SET

Route	Length (km)	Number of stations	Duration (hour)	Number of trips	Static (hour)	0-150 km/h (hour)	150-280 km/h (hour)	280-310 km/h (hour)	3-minute		50 KB-sized		2 MB-sized	
									number of flows	size (GB)	number of flows	size (GB)	number of flows	size (GB)
B-G	2,298	17	9.7	10	16.9	24.5	19.5	46.3	1,397	17.3	17,410	1.6	3,722	12.5
C-S	1,210	10	6.7	8	12.9	11.6	9.2	28.1	709	9.8	9,373	0.8	1,874	6.9
S-B	1,318	10	5.8	14	19.6	17.5	14.7	42.7	1,161	15.3	14,254	1.4	3,142	10.5
B-T	115	2	0.6	52	44.5	6.8	5.2	16.6	803	8.8	7,475	0.7	1,492	5.3

which can lead to bit error rate (BER) variation and bandwidth change. On the other hand, high-speed motion causes more frequent handoffs, resulting in sharp delays, consecutive packet losses, and network disconnections, hurting TCP performance badly.

Since mobility and handoff are intertwined, it is difficult to analyze the effect of each factor at the same time. We use the total number of handoffs that a flow experiences during transmission to quantify the degree of handoff that a flow suffers. Through measurements, we observe that flows suffer more frequent handoffs in urban areas than in suburban and rural areas. For example, when the train passes through Bejing, a 3-minute flow can experience handoff 12 times. However, as the train crosses large areas of farmland along the B-T line, sometimes a 3-minute flow suffers no handoff. This can be explained by variation in base station distribution density. Generally, due to difference in population and volume of network traffic, carriers deploy much denser base stations and smaller cells in urban areas than suburban and rural areas. Therefore, handoff frequency for mobile devices on high-speed trains varies significantly during the movement of the train along railway lines. This allows us to achieve variation in the number of handoffs suffered by flows. In addition, we make use of variation in train speed as shown in Figure 1 to perform measurements in static, slowly moving and high-speed motion mobile scenarios.

We use following methods to independently analyze the effect of speed and handoff: (1) we compare the performance among TCP flows that suffer no handoff when the train runs at various speed to study the effects of speed change alone, (2) we make a comparison in TCP performance among flows that suffer different number of handoffs when the train runs at a relatively stable high speed to quantitatively analyze the impacts of handoff.

A. Packet Loss Rate and RTT

We use 3-minute TCP downlink flows to analyze packet loss rate and RTT, because 3-minute duration is long enough for a flow to leave slow start, and the network type may keep unchanged and the train may run at only one of the 4 levels of speed during this time. We discard those flows that are transmitted in a network different from HSPA+ or those that experience train speed variations which span two speed levels, for example, from 120 km/h to 200 km/h. As hilly terrains can lead to complicated fading characteristics [20], we discard flows measured when the train passes through hills and tunnels, only use those measured in large areas of open plains.

Through comparison of packets captured on the server and the phone, we can determine the lost packets. Most previous works define packet loss rate as the proportion of total flow

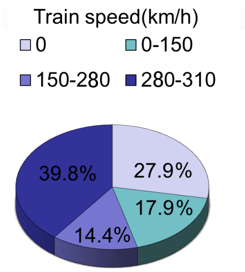


Fig. 1. Train speed

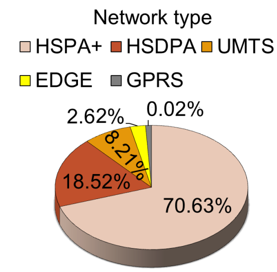


Fig. 2. Network type

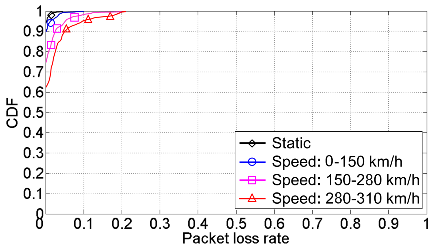
packets that are lost. This only shows an average packet loss rate for the duration of a flow. In order to precisely determine the time when packet loss bursts occur, we define packet loss rate as the percentage of packets sent out by the server per second that are lost.

In Section V, we shall show that the number of bytes in flight of a 3-minute flow is very small, so queuing delay is expected to be minimal, and we can use RTT to study the impacts of mobility and handoff on network delay. In our experiments, the server and the phone use options of Timestamp Value (TSV), Timestamp Echo Reply (TSER) and Selective Acknowledgement (SACK). We calculate RTT in the following way. When the server receives a normal ACK, we update RTT by TSER. While when the server receives a duplicate ACK (DupACK), since multiple DupACKs have the same TSER, we use SACK to accurately calculate RTT [22].

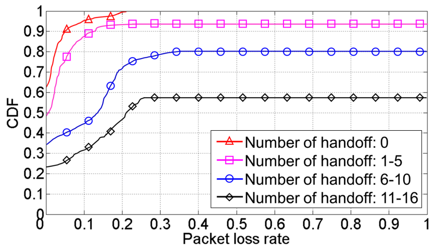
It is worth noting that serial (multiple) retransmissions, which mean that a packet is retransmitted several times and suffers from exponential back-offs, often happen when a flow suffers multiple handoffs. Due to exponential back-offs, there are intervals as long as dozens of seconds during which the server sends out no packet. We can not update RTT and packet loss rate in these intervals.

Among the 3-minute flows we analyze, 33.8% experience no handoff, 28.3% experience handoff 1 to 5 times, 25.2% suffer 6 to 10 handoffs, and 12.7% suffer handoff 11 to 16 times. Figures 3(a) and 4(a) show the CDF of packet loss rate and RTT respectively, of flows that suffer no handoff when the train moves at different speed. Figures 3(b) and 4(b) depict the CDF of the two metrics for flows that suffer various number of handoffs when the train runs at a relatively stable speed ranging from 280 to 310 km/h. We make following findings:

1) Effects of speed. For speeds below 150 km/h, compared to stationary scenarios, packet loss rate and RTT only rise slightly. However, when the speed is higher than 150 km/h, packet loss rate and RTT rise more significantly and vary within a wider range. This can be explained by big variation in BER and available bandwidth due to fast fading. With the fluctuation of BER and bandwidth, the rate of packet loss due to both bit error and congestion vary correspondingly. Then local retransmission

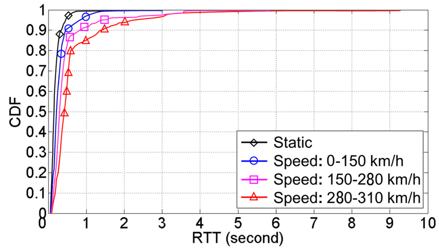


(a) Effects of speed

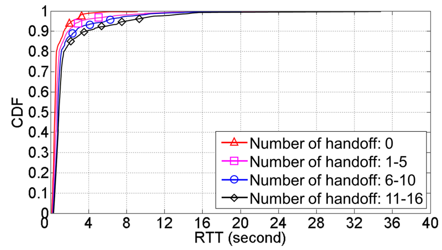


(b) Effects of handoff

Fig. 3. CDF of packet loss rate

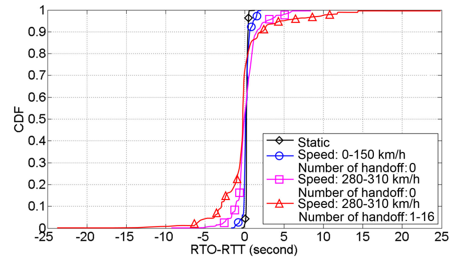


(a) Effects of speed

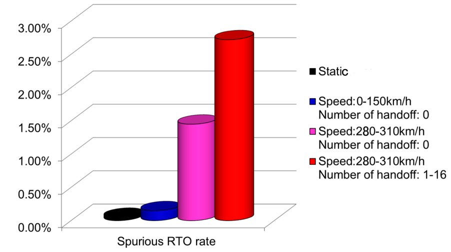


(b) Effects of handoff

Fig. 4. CDF of RTT



(a) CDF of difference between RTO and RTT



(b) Spurious RTO rate

Fig. 5. RTO estimation errors and spurious RTO

rate at the link layer rises and drops correspondingly with the variation of packet loss rate, leading to more RTT spikes and wider RTT variation.

2) Effects of handoff. Compared with high-speed mobility itself, the number of handoffs that a flow suffers during the movement of the train contributes more to performance degradation of packet loss rate and RTT. As the number of handoffs increases, packet loss rate and RTT rise drastically. When a 3-minute flow suffers 11 to 16 handoffs, the packet loss rate reaches 100% at a probability of 42%, RTT exceeds 4 s at a probability of 10%, and even ranges from 8 s to 35 s at a probability of 6%. This can be explained by a large number of consecutive packet losses during the process of handoff, and long delay for smartphones to choose a new base station, disconnect from the old one and reconnect to a new.

3) Effects of dense base station deployment. Since flows suffer much more handoffs in urban areas, we observe that dense base station deployment in urban areas is a double-edged sword. Although dense deployment of base stations and small cells improve network coverage and capability for stationary and low-speed motion mobile users in big cities where network traffic volume is very high, it can lead to high handoff frequency for passengers on HSR, resulting in poor network performance.

B. Network Disconnection

During the experiments, we observe that the phone repeatedly suffers network disconnections, which is caused by handoff failures in the following two cases:

1) Lack of a better choice. If signal quality of the current base station reaches the threshold to trigger a handoff yet no near base station can provide better signal quality, handoff failure will happen. In the network of the carrier we measure, deployment of base stations in rural areas is very sparse, and we observe that mobile devices are prone to disconnections when the train passes through these areas, which is due to the lack of better choice for a new base station during handoff.

2) Small cells. If the cell of current base station is very small, the train may run out of range of the cell before completion of the handoff, leading to handoff failure. The carrier deploys very dense base stations and quite small cells in urban areas, so the train may run out of the range of the cell at a high speed before completion of the handoff. We indeed observe many network disconnections in urban areas.

In conclusion, sparse and dense distribution of base stations along the railway both contribute to increase of network disconnections. In addition, current handoff schemes have not been well designed to adapt to the high-speed motion of end users, which also results in frequent disconnections.

C. Summary

Basing on the above results and analysis, we can characterize the main challenges brought by high-speed trains to TCP:

1) Wide RTT variations. Previous studies have found that delay variation can induce various problems [6]. As shown in Figure 4(a) and 4(b), due to both high-speed mobility and handoff, RTT varies in a much wider range with more spikes in high-speed motion mobile cases than in static and slowly moving mobile cases, which will cause more serious problems, such as high spurious RTO rate.

2) Heavy packet losses. Packet losses occur more frequently and loss rate is much higher in high-speed motion mobile cases than in static and slowly moving mobile cases. While authors in [23]–[25] found that TCP can not adapt well to high packet loss rate, we observe serious and frequent packet losses in high-speed trains probably lead to more significant problems, such as frequent and aggressive congestion window reduction and low utilization of bandwidth. During the time when packet loss rate is as high as 100%, serial retransmissions will happen and result in very long delay due to exponential back-offs.

3) Frequent disconnections. Affected by handoff failures, disconnections happen frequently, which probably causes transmission interruption of TCP flows and performance differences between big and small flows. Because duration of big flows is much longer than small flows, big flows may encounter network disconnections and suffer transmission interruptions with higher probability.

In Section V, we shall investigate abnormal behaviors in almost all aspects of TCP due to above challenges, and also compare the performance of big and small flows.

V. TCP BEHAVIORS

In this Section, we use 3-minute TCP downlink flows to analyze issues in transmission and congestion control, and use 50 KB-sized flows to study TCP connection establishment and closure because the duration of 50 KB-sized flows is short, which allows us to observe more connection establishment and closure instances. In addition, we also compare performance between 50 KB-sized and 2 MB-sized flows to study the effects of flow size. We discard those flows that are transmitted in a network different from HSPA+ or those that experience train speed variations which span two speed levels. Flows measured when the train passes through hills and tunnels are discarded as well. Among these available flows, 43.2% of 2 MB-sized flows suffer handoff 1 to 12 times, and the rest experience no handoff. 38.9% of 50 KB-sized flows suffer no handoff and the rest experience handoff 1 to 9 times during transmission.

A. Retransmission

If a sender neither receives the expected ACK，nor gets any indication of packet loss from 3 dupACKs or SACK, retransmission will be triggered by RTO. However, RTO may not be estimated accurately and spurious RTO probably occurs on HSR for two main reasons:

1) Estimation algorithm. In TCP, RTO is computed by the sender using smoothed RTT and RTT variation [26], which can work well in stationary and low-speed motion mobile scenarios since the variation of RTT is low. However, the accuracy of the algorithm may decrease when RTT variation is high. Because high-speed mobility and frequent handoff can lead to high variation of RTT, the RTO estimation algorithm may not adapt fast enough to the change of RTT, leading to serious inaccuracy.

2) DupACKs. TCP does not use DupACKs to update RTT and RTO. In stationary scenarios, the percentage of DupACKs is low, and RTT is relatively stable, so not using DupACKs to update RTT does not cause serious problems. However, in high-speed motion mobile scenarios, DupACKs account for a big proportion. For example, when the speed is ranging from 280 to 310 km/h, for 3-minute flows that suffer handoff 1 to 5 times, on average, 45% of ACKs are DupACKs. Besides, RTT fluctuates rapidly within a wide range, so when not using DupACKs to update RTT, TCP may not update RTO timely and suffer considerable estimation errors.

The difference between RTO and the latest RTT can reflect the estimation accuracy of RTO to some extent. Figure 5(a) exhibits the CDF of the difference between RTO and RTT of 3-minute flows measured under different train speeds with different number of handoffs suffered. We estimate RTO from packet retransmission intervals, and calculate RTT by SACK option, which can help us accurately update RTT after receiving a DupACK. As shown in the figure, when the train is parking, RTO is very close to RTT with a difference within hundreds of milliseconds. However, as the train accelerates, the difference becomes larger. We also observe that handoff leads to bigger differences between RTO and RTT. If RTO is estimated much shorter than the latest RTT, spurious RTO may occur. While if RTO is estimated much longer than the latest RTT, TCP's response to packet losses may be too slow.

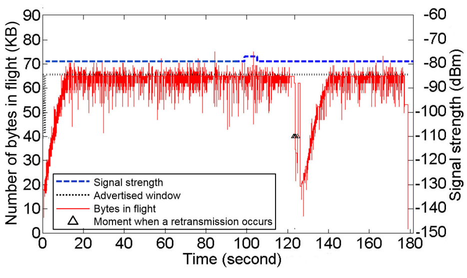
By comparing captured packets on the server and the phone, we can determine those packets that are retransmitted when no packet loss occurs. Among these spurious retransmissions, only those that are triggered by timeout rather than 3 DupACKs or SACK are spurious RTO. We define spurious RTO rate as the percentage of retransmitted packets triggered by spurious timeout in all the packets sent out from the server in a 3-minute flow. Figure 5(b) depicts average spurious RTO rate of 3-minute flows that suffer different number of handoffs and are measured at various train speeds. We note that both high speed mobility and handoff can lead to increase in spurious RTO rate.

B. Congestion Control

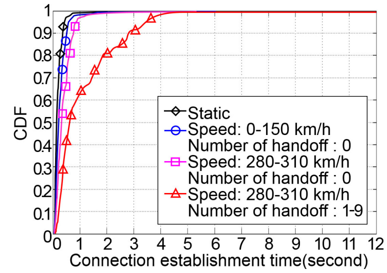
We use the number of bytes in flight to study congestion control. Bytes in flight are those bytes that have been sent out by the server, but not acknowledged yet by ACKs returned from the phone. The number of bytes in flight can reflect the size of slide window, which is the minimum of CWND and advertised window of the receiver (phone). Figures 6(a) and 6(b) show the changes in the number of bytes in flight over time of two 3-minute flows measured when the train is parking and almost running at a constant speed of 300 km/h respectively. The two figures also show signal strength variation over time and mark moments when handoffs and retransmissions occur. We have following findings.

1) Static case. Signal strength is relatively stable when the train is parking, and no handoff occurs during the 3-minute period. The number of bytes in flight can reach to the size of the advertised window of the phone at 10 s and remain at this size unless a retransmission occurs for packet loss. Only 3 retransmissions occur in the duration of the flow. Although the number of bytes in flight drops suddenly to a very low level due to congestion control, it can rise to the size of the advertised window again after about 10 s.

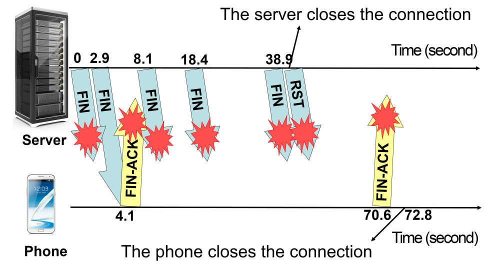
2) Mobile case. When the train runs at a speed of 300 km /h, we observe that 6 handoffs occur in the 3-minute duration and signal strength fluctuates within a wide range rapidly due to fast fading brought by high-speed mobility. Since packet losses and retransmissions happen frequently, CWND repeatedly experiences additive increase and multiplicative decrease (AIMD) [27]. Hence, the number of bytes in flight hardly reaches and keeps a size as big as the advertised window. Between 135 and 178 s, the number of bytes in flight is not updated, which is



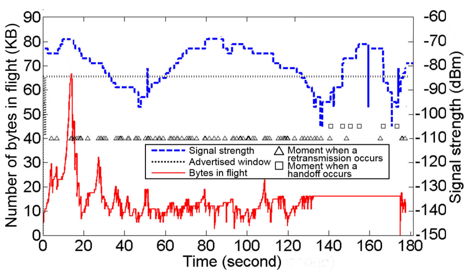
(a) Static



(a) Connection establishment

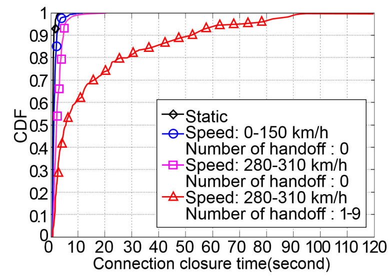


(a) Closure delay



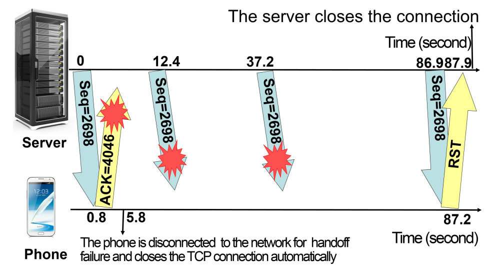
(b) Mobile

Fig. 6. Change of the number of bytes in flight



(b) Connection closure

Fig. 7. CDF of connection establishment and closure time



(b) Transmission failure

Fig. 8. Abnormal connection closure

because that during frequent handoff, the server only sends out 5 retransmitted packets and receives no ACK. Affected by bad network conditions in high-speed trains, a big portion of packet losses may not be due to congestion, but bit error. However, TCP always attributes packet loss to network congestion, hence conducts very aggressive CWND reduction, which may lead to very low utilization rate of bandwidth. Besides, as spurious RTO rate increases on HSR, many undesired slow starts are triggered, hurting TCP performance significantly.

C. Connection Establishment and Closure

Basing on analysis of 50 KB-sized flows, we observe that TCP also encounters problems when establishing and closing a connection. It takes a much longer time to set up or close a TCP connection on high-speed trains. Even more, sometimes connections can not even be established successfully. Some connections are closed abnormally before a 50 KB-sized file is completely received by the client, which is a waste of both time and energy.

1) Long delay. TCP uses three-way and four-way handshakes to establish and close a connection respectively. Due to high-speed mobility and frequent handoff, loss rate of handshakes is very high on HSR. Handshakes may be retransmitted even for multiple times and suffer from exponential back-offs, which will lead to very long delay to establish or close a connection. Figure 8(a) shows an example. For multiple handshake losses and retransmissions, the server and the phone finally send 7 handshakes (FIN, FIN-ACK) and 1 RST. The server closes the connection unilaterally after multiple retransmission failures at 38.9 s, and sends out a RST. However, for the loss of the RST, the phone still retransmits the second FIN-ACK, and has to close the connection when a timeout occurs at 72.8 s finally. In our measurements, connection establishment and closure are initiated by the phone and the server respectively. We define connection establishment time as the time span from when the first SYN is sent out by the phone to when the last ACK is received by the server, which acknowledges the SYN-ACK sent out by the server. Connection closure time is defined as the time span from when the first FIN is sent out by the server to the last packet of the flow. Figures 7(a) and 7(b) show CDF of connection establishment and closure time. Compared to high speed mobility, handoff contributes much more to the long delay of connection set-up and closure. As shown in Figure 3(b), handoff can lead to a very high packet loss rate even up to 100% at quite high probability, so handshakes may suffer multiple losses and serial retransmissions, similar to the example in Figure 8(a). The reason why connection closure time is much longer than set-up time is that RTO is much longer when the connection is to be closed than the initial RTO when connection is to be established, and the delay of closure becomes even much longer after exponential back-offs.

2) Establishment failure and abnormal closure. In addition to long delay, TCP connections also suffer establishment failures and abnormal closures on high-speed trains. When the train runs at a speed ranging from 280 to 310 km/h, for 50 KB-sized flows that suffer handoff 0 to 9 times, 1.1% of connections can not be established successfully even after a serial retransmission of handshakes and 3.2% of connections can not be properly closed by handshakes, but are closed abnormally by RST, network disconnection or timeout. Among these abnormally closed connections, most are closed before the phone completely receives the whole 50 KB-sized file. Figure 8(b) shows an example. The server sends out the first packet with a sequence number of 2698 at 0 s. At 5.8 s the phone suffers a network disconnection due to handoff failure, and the TCP connection is closed automatically for network unavailability. The phone reconnects to the network successfully at 10.3 s. However, the server does not know

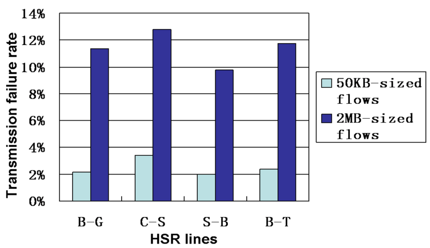


Fig. 9. Transmission failure rate

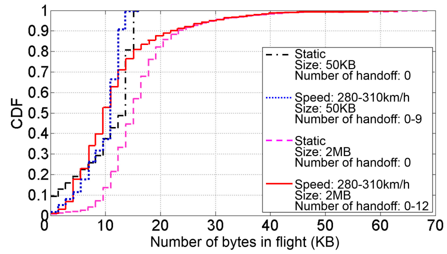


Fig. 10. CDF of the number of bytes in flight

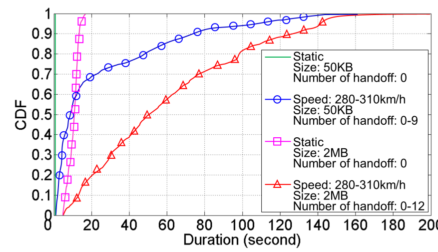


Fig. 11. CDF of duration

that the phone has closed the connection. Due to the loss of the ACK, the server retransmits the packet. The first two retransmitted packets are lost and only the third one successfully reaches the phone at 87.2 s. However, the phone has already closed this connection, so this retransmitted packet is useless and the phone sends a RST packet to inform the server to close the connection. Finally, the server closes this connection at 87.9 s after receiving the RST. This is an example of transmission failure. Although it is not difficult to resume broken transfers technically, through measurements we find that most applications such as web page browsing and video playing do not resume broken transfers when reconnecting to the network after a period of network disconnection. Therefore, after encountering a data transmission interruption on HSR, the incomplete part of a file is discarded, resulting in a serious waste of time and energy. In addition, due to the lack of real-time information about the phone's network condition, the server can not timely close the connection that has already been closed by the phone unilaterally, which unnecessarily consumes more time and energy.

D. Effects of Flow Size

In cellular networks, most flows are small. On the other hand, a very small fraction of large flows, which are known as "heavy-hitter" flows, contribute to the majority of the traffic volume [3]. Big flows often show overwhelming advantage in congestion window size and throughput over small flows in stationary scenarios, because small flows often finish transmission before leaving slow start. We set 50 KB and 2 MB as typical sizes of small and big flows respectively and want to determine if big flows can maintain the advantage in high-speed trains. We compare three characteristics between big and small flows: transmission failure rate, the number of bytes in flight, and duration. Transmission failure rate is the percentage of flows that are unusually closed before data transmission completion.

Figure 9 shows transmission failure rate during the movement of the train on four routes. We find that big flows are more prone to transmission failures. For example, on the B-G line, 11.2% of 2 MB-sized flows suffer transmission failures, while only 2.1% of 50 KB-sized flows are closed unusually before the phone completely receives the file. This can be explained by the long duration of big flows. Big flows are likely to suffer more from bad network conditions, especially network disconnections due to the long duration, so transmission failure rate is much higher.

Excluding those failed flows, we compare the number of bytes in flight and duration between successfully transmitted 50 KB-sized and 2 MB-sized flows. Figures 10 and 11 depict CDF of the number of bytes in flight and duration respectively. 50 KB-sized flows that suffer handoff 0 to 9 times include those flows that suffer no handoff and the rest flows that suffer handoff 1 to 9 times. Similarly, 2 MB-sized flows that suffer handoff 0 to 12 times include those flows that suffer no handoff and the rest that suffer handoff 1 to 12 times. We make following findings:

1) Big-sized flows no longer show an overwhelming advantage in the number of bytes in flight over small-sized flows in a fast running train. When the train is parking, 2 MB-sized flows have much more bytes in flight. However, when the train runs at a high speed, the overwhelming advantage of big flows can not be maintained. This can be explained by aggressive congestion control as shown in Figure 6(b). Due to frequent packet drops, CWND experiences additive increase and multiplicative decrease repeatedly, even drops to one segment frequently when slow start is triggered by RTO. Therefore, inspite of longer durations, the CWND of big-sized flows is unlikely to reach and maintain a very large size.

2) For both big and small flows, duration rises significantly on a fast running train. When the train is parking, passengers can always download a 50 KB-sized file in 0.7 s and download a 2 MB-sized file in 18 s. While when the train runs at a speed ranging from 280 to 310 km/h, the durations for both two types of flows become incredibly long, which is mainly caused by exponential back-offs during serial retransmissions of packets and handshakes. The long durations can lead to very poor QoE for real-time applications such as on-line games. Even for non-real time applications, QoE may be unacceptable too. For example, downloading a 2 MB-sized attachment from an E-mail. For the high transmission failure rate, users may suffer transmission interruption at an average probability of 11.54% on the four routes we measure. Even if they luckily avoid a transmission interruption, they will also suffer very poor QoE due to the long time needed to download the file and may give up before download is complete.

VI. Conclusion

In this paper, we for the first time present a comprehensive measurement study of the TCP performance and behaviors on HSR. We conduct an 8-month measurement on the HSPA+ networks in the high-speed trains of various routes in China, with speeds reaching 310 km/h. We have covered a total distance of 5,000 km and collected more than 90 GB of data.

We analyze the impacts of high-speed mobility on such metrics as RTT, packet loss rate and connectivity. We find that the RTT spikes, packet drops, and network disconnections occur more frequently and have a greater impact on the TCP performance than in static or slow movement scenarios. We then evaluate the adaptability of TCP under HSR. We study TCP establishment, transmission, congestion control and connection closure. We find that TCP shows serious abnormal behaviors, such as high spurious RTO rate, aggressive congestion window reduction, a long delay of connection establishment and closure, and transmission interruption. We also study the effect of flow size, and we show that big flows suffer higher performance degradation than small flows.

In summary, high-speed trains indeed pose a big threat to TCP. It is probable that with faster trains/aircrafts deployed in the foreseeable future, TCP will be faced with more serious challenges and may even collapse when the speed is high enough one day. Compared with high speed mobility, frequent handoff brought by fast motion contributes more to TCP performance degradation. We suggest that base stations be distributed with more proper density along the railway line. Furthermore, more intelligent handoff schemes should be deployed to adapt high-speed mobility, reducing both delay and packet loss rate. To cope with significant fast signal fading due to high-speed motion, advanced anti-fast fading technologies should be brought in. It is also urgent to develop HSR-friendly protocols to take better use of network resources and improve QoS and QoE of mobile apps.

Acknowledgment

This work has been supported in part by NSFC Project (61170292, 61472212), National Science and Technology Major Project (2012ZX03005001), 973 Project of China (2012CB315803), 863 Project of China (2013AA013302, 2015AA010203), EU MARIE CURIE ACTIONS EVANS (PIRSES-GA-2010-269323 and PIRSES-GA-2013-610524), Natural Science Foundation of China (No. 61272464), RGC/GRF PolyU 5264/13E, HK PolyU G-YM06, A-PK95, 1-ZVC2, and National Science Foundation under Grants No. CNS-1421440.

We would like to appreciate Wei Tan, Guodong Zhao, Qingfang Liu, Tong Li, Yi Qu and Fei Li, for their help in collecting data traces on HSR. We also appreciate engineers of CERNET who help us deploy the server for measurements.

References

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

[2] J. Huang, F. Qian, Y. Guo, Y. Zhou, Q. Xu, Z. M. Mao, S. Sen, and O. Spatscheck, "An In-depth Study of LTE: Effect of Network Protocol and Application Behavior on Performance," in *Proceedings of ACM SIGCOMM*, 2013.

[3] F. Qian, A. Gerber, Z. M. Mao, S. Sen, O. Spatscheck, and W. Willinger, "TCP Revisited: A Fresh Look at TCP in the Wild," in *Proceedings of ACM SIGCOMM conference on Internet Measurement Conference (IMC)*, 2009.

[4] C. Paasch, G. Detal, F. Duchene, C. Raiciu, and O. Bonaventure, "Exploring mobile/WiFi handover with multipath TCP," in *Proceedings of ACM SIGCOMM Workshop on Cellular Networks (CellNet)*, 2012.

[5] H. Jiang, Y. Wang, K. Lee, and I. Rhee, "Tackling Bufferbloat in 3G/4G Networks," in *Proceedings of ACM SIGCOMM conference on Internet Measurement Conference (IMC)*, 2012.

[6] M. C. Chan and R. Ramjee, "TCP/IP performance over 3G wireless links with rate and delay variation," in *Proceedings of ACM MOBICOM*, 2002.

[7] D. Pacifico, M. Pacifico, C. Fischione, H. Hjalrmasson, and K. H. Johansson, "Improving TCP Performance During the Intra LTE Handover," in *Proceedings of IEEE GLOBECOM*, 2009.

[8] N. C. Wang, Y. Y. Wang, and S. C. Chang, "A Fast Adaptive Congestion Control Scheme for Improving TCP Performance during Soft Vertical Handoff," in *Proceedings of IEEE WCNC*, 2007.

[9] R. Litjens, "HSDPA Flow Level Performance and the Impact of Terminal Mobility," in *Proceedings of IEEE WCNC*, 2005.

[10] J. Yao, S. S. Kanhere, and M. Hassan, "An Empirical Study of Bandwidth Predictability in Mobile Computing," in *Proceedings of Third ACM Int'l Workshop Wireless Network Testbeds, Experimental Evaluation and Characterization (WINTECH)*, 2008.

[11] J. Derksen, R. Jansen, M. Maijala, and E. Westerberg, "HSDPA Performance and Evolution," *Ericsson Rev.*, vol. 3, pp. 117–120, 2006.

[12] F. P. Tso, J. Teng, W. Jia, and D. Xuan, "Mobility: A Double-Edged Sword for HSPA Networks: A Large-Scale Test on Hong Kong Mobile HSPA Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 10, pp. 1895–1907, 2012.

[13] "International union of railways," http://www.uic.org.

[14] A. El Falou and S. E. Elayoubi, "Uplink Flow Level Capacity for HSPA+ Systems," in *IEEE Vehicular Technology Conference (VTC)*, 2010.

[15] R. Merz, D. Wenger, D. Scanferla, and S. Mauron, "Performance of LTE in a High-velocity Environment: A Measurement Study," in *SIGCOMM Workshop on All Things Cellular*, 2014.

[16] Q. Xiao, K. Xu, D. Wang, L. Li, and Y. Zhong, "TCP Performance over Mobile Networks in High-speed Mobility Scenarios," in *Proceedings of IEEE ICNP*, 2014.

[17] K. Jang, M. Han, S. Cho, H. K. Ryu, J. Lee, Y. Lee, and S. Moon, "3G and 3.5G Wireless Network Performance Measured from Moving Cars and High-Speed Trains," in *Proceedings of the 1st ACM Workshop on Mobile Internet through Cellular Networks (MICNET)*, 2009.

[18] "Iperf," https://iperf.fr/.

[19] "CERNET: China Education and Research Network," http://www.edu.cn/.

[20] F. Luan, Y. Zhang, L. Xiao, C. Zhou, and S. Zhou, "Fading Characteristics of Wireless Channel on High-Speed Railway in Hilly Terrain Scenario," *International Journal of Antennas and Propagation*, vol. 2013, 2013.

[21] Y. Yang and P. Fan, "Doppler frequency offset estimation and diversity reception scheme of high-speed railway with multiple antennas on separated carriage," *Journal of Modern Transportation*, vol. 20, no. 4, pp. 227–233, 2012.

[22] M. Mathis, J. Mahdavi, S. Floyd, and A. Romanow, "TCP Selective Acknowledgment Options," *RFC 2018*, 1996.

[23] H. Bai, D. Lilja, and M. Atiquzzaman, "Applying Speculative Technique to Improve TCP Throughput over Lossy Links," in *Proceedings of IEEE GLOBECOM*, 2005.

[24] S. W. Ng and E. Chan, "Equation-based TCP-friendly congestion control under lossy environment," *Systems Architecture*, vol. 51, pp. 542–569, 2005.

[25] J. Lee, H. Cha, and R. Ha, "Improving TCP fairness and performance with bulk transmission control over lossy wireless channel," *Computer Networks*, 2009.

[26] V. Paxson, M. Allman, J. Chu, and M. Sargent, "Computing TCP's retransmission timer," *RFC 6298*, 2011.

[27] M. Allman, V. Paxson, and E. Blanton, "TCP congestion control," *RFC 5681*, 2009.