An In-depth Analysis of Subflow Degradation for Multi-path TCP on High Speed Rails

Tong Li[†], Li Li[‡], Xiangxiang Wang[§], Xu Zhang[¶], Feng Zhang[†], and Kao Wan[‖]
Renmin University of China[†], Tsinghua University[‡], Simon Fraser University[§], University of Exeter[¶], PCL[‖]
Email: tong.li@ruc.edu.cn, ll-12@tsinghua.org.cn, xwa203@sfu.ca, x.zhang6@exeter.ac.uk,
fengzhang@ruc.edu.cn, wankao@hotmail.com

***Abstract*—Recent advances in high-speed rails (HSRs), coupled with user demands for communication on the move, are propelling the need for acceptable quality of experience (QoE) in high-speed mobility environments. However, with throughput declining significantly the QoE on existing HSRs is still far from satisfactory. In order to improve QoE on HSRs, this paper seeks to answer the question regarding which is better of two options: the selection of the best cellular carrier applying single-path TCP or the conjunction of multiple carriers applying Multi-path TCP (MPTCP). To this end, we carefully design comparison experiments using the two approaches on HSRs with a peak speed of 310 km/h. Measurement study on MPTCP performance shows that generally carrier conjunction gives similar performance as carrier selection. We take an in-depth analysis of the details of the instances, and for the first time expose the phenomenon called *subflow degradation*. We further confirm that subflow degradation of MPTCP occurs due to its poor adaptability to frequent handoffs. We believe these insights can provide valuable guidance for the design, implementation, and deployment of transmission protocols in high-speed mobility environments.**

***Index Terms*—multiple cellular carriers, high speed rails, multi-path TCP, subflow degradation**

I. Introduction

Recent years have witnessed a significant worldwide progress in the development and deployment of high speed rail (HSR), which has reached over 56,000 km before June, 2021. HSR is a new type of rail transport with higher speed and better experience than traditional rail traffic. On the other hand, with rapid progress in transmission capabilities of both wireless networks and mobile devices [1], users require keeping online anytime and anywhere, such as online gaming [2], web-browsing, Internet telephony, online data analytics [3]–[5], and video streaming [6]–[8]. However, state-of-the-art measurements [9]–[13] indicate that throughput declines significantly under such extremely high-speed conditions, resulting in poor network quality of experience (QoE) on existing HSR environments.

Measurement results [9]–[12] have demonstrated that frequent handoff is the main cause of poor QoE on HSRs. The general category of single-path solutions [14]–[19] can hardly adapt to the frequent handoffs on HSRs. A fundamental solution to deal with frequent handoff is developing seamless handoff techniques for high-speed mobility environments [20]. However, this requires evolution of technologies on the side of operators, which will likely take time.

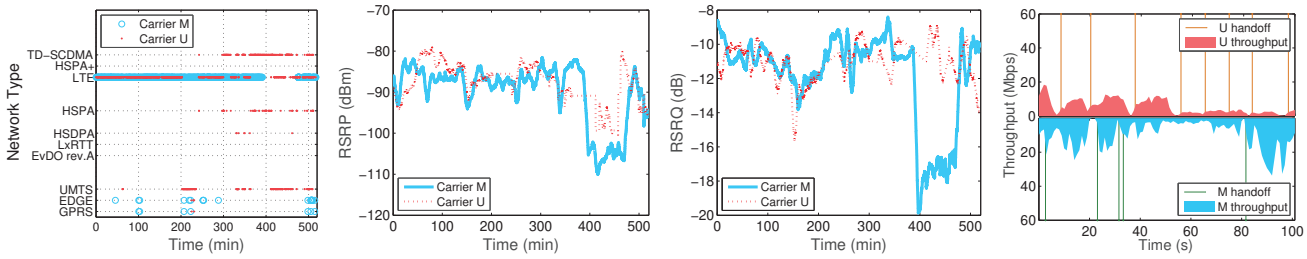
On the other hand, it is supposed that better QoE can be achieved through carrier *complementarity* in the distribution of network types, signal strength and quality, and handoff time span (as detailed in Section II-B). There are two options for making use of the complementarity between multiple cellular carriers, *i.e.*, dynamically switching to the best one of multiple carriers on demand, or simultaneously using all the available carriers. The former is called *carrier selection*, and the later *carrier conjunction*.

On-demand carrier selection requires seamless handover from one carrier to another, which is currently an unsettled issue. In addition to carrier selection, this paper mainly explores multi-path transfer via carrier conjunction. However, the ordinary single-path TCP may fail to take full advantage of multi-path packet deliver. This is because TCP is designed for connections that traverse a single path between host pairs [21]. As a natural side effect of multi-path routing, single-path TCP rarely optimizes packet reordering at the receiver [22].

Multi-path TCP (MPTCP) [23] is a relatively mature solution to support multiple carriers. It moves traffic away from congested paths onto less congested ones using its coupled congestion control algorithms [24], [25]. MPTCP has been proved to be effective in improving the efficiency and robustness of network transmission in static and low-speed mobility environments [26]–[30]. Companies have been enthusiastically adopting MPTCP. For example, Apple, Huawei, and Samsung are implementing this revolutionary protocol into their latest iOS or Android systems. The natural question is then on how well MPTCP would behave in a multi-carrier environment in high-speed mobility scenarios. Li *et al.* [13] conducted the first MPTCP performance study of two cellular carriers on HSRs, they exposed that MPTCP's value is proven mostly in reliability enhancement rather than bandwidth aggregation. This paper takes one step further to explore three cellular carriers and clarify the reason why MPTCP fails to achieve bandwidth aggregation from the perspective of *subflow degradation* (see Section IV).

Estimating the performance of MPTCP does not only require a significant amount of efforts, but is also technically challenging. First, to access multiple carrier networks simul-

This work is supported by the National Natural Science Foundation of China (62072459), the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement (898588).



(a) (b) (c) (d)

Fig. 1: Carrier complementarity. (a), (b) and (c) depict the LTE coverage, signal strength and signal quality in one-way trip on the B-F line, respectively. (d) illustrates the handoff of Carriers M and U along the B-F line. (To avoid measurement bias, we also investigate carrier complementarity along other HSR lines. Since results remain consistent, we just take the B-F line as an example.)

taneously is not straightforward. Second, to analyze the large-scale measurement data, development of dedicated measurement and analytical tools is required. Third, root cause analysis on the MPTCP performance is not straightforward either.

This paper makes the following contributions.

First, we have overcome the challenges on experiment design, and conducted a comprehensive measurement study on MPTCP performance "in the wild", exploring multi-carrier data transmission on HSRs with a peak speed of 310 km/h during six months.

Second, measurement results show that network performance is poorer in mobility environments, and carrier conjunction performs similarly to carrier selection. However, MPTCP suffers from the utilization imbalance between subflows in high-speed mobility environments. We then look into the details of the instances, and find events of subflow degradation, *i.e.*, some subflows encounter a sudden fall in throughput and suffer from sustained inefficiency afterwards. In-depth analysis reveals that the subflow degradation is because of MPTCP's poor adaptability to frequent handoffs.

The rest of this paper is structured as follows. Section II motivates the carrier selection and conjunction. Section III introduces the measurement methodology of multi-carrier MPTCP. Section IV compares the performance between carrier selection and carrier conjunction along HSR lines. Section V discusses the next research directions and future work. We introduce the related work in Section VI, and then conclude the paper in Section VII.

II. Background and Motivation

A. HSR Environments

In high-speed mobility environments, there are some particular inherent network characteristics that do not exist or are not so significant in low-speed mobility environments. 1) High-speed mobility. The current peak operating speed is 310 km/h. Such a high speed leads to serious multi-path fading and Doppler shift on wireless channels. 2) Constantly changing network types. Any HSR line is usually served by different network types including GPRS, HSPA+, LTE, *etc*. Each supports a different QoE. 3) Fluctuating signal strength and quality. Terrain along long distance HSR routes is diverse. It includes plains, hills, valleys and tunnels, all of which affects cellular signal quality and shadow fading characteristics [31]. Therefore, users on HSRs may experience highly varying signal strength and signal quality when the train crosses different terrain types. 4) Frequent handoffs. At high speed, HSR trains go through the coverage of multiple base stations in a short time. Therefore, mobile devices have to handoff very frequently, which may lead to significant delays and consecutive packet losses.

State-of-the-art measurements [9]–[13] have shown that throughput declines greatly and disconnection occurs frequently under such extremely high speed conditions, resulting in poor QoE. For example, it is demonstrated that although the peak throughput with TD-LTE technology can reach as high as 100 Mbps with good network conditions, it can decrease to 0 Mbps at cell edges or during handoffs, leading to RTT spikes, packet drops and network disconnections [32].

B. Carrier Complementarity

We consider three representative cellular carriers (represented as M, U and T) in China[1]. As mentioned above, there are some particular inherent network features on HSRs, such as constantly changing network types, fluctuating signal strength and quality, and frequent handoffs. These diverse features lead to unstable QoE. If we think of these from another perspective, inspiration of solutions for QoE improvement could be obtained by considering the *complementarity* of multiple cellular carriers in terms of network type, signal feature, and handoffs.

Network Type. Our investigation indicates that there are sections with poor infrastructures along the HSR routes for each carrier. We define *LTE coverage rate* as the proportion of total amount of time when LTE is detected along HSRs. For example, Figure 1(a) illustrates the time-variant network types of Carriers M and U on the Beijing-Futian (B-F) HSR line. It can be determined that, individually, the LTE coverage rates of Carriers M and U are 79% and 76%, respectively.

[1]Although there exist many other carriers, only M, U and T have full coverage along most of the HSR routes in China, and at the end of 2020 their accumulative number of subscribers is 942 million, 305.8 million and 351 million, respectively.

However, if we could use any of the two carriers along the line, then the coverage rate would increase to 94%.

Signal Feature. Signal feature includes signal strength and signal quality, which can be characterized by the Reference Signal Received Power (RSRP) and Reference Signal Receiving Quality (RSRQ), respectively. Signal strength is in the "Good" category if the RSRP is in the range -105 dBm to -90 dBm, and signal quality is in the "Good" category if the RSRQ is in the range -12 dB to -9 dB [33]. For example, in Figure 1(b), single Carrier M and single Carrier U obtain "Good" signal strength during 64% and 60% of the trip, respectively. Furthermore, the proportion of the complementary "Good" signal strength between Carriers M and U increases to 72%. This increase can also be demonstrated when analyzing the complementary signal quality, which is shown in Figure 1(c).

Handoff. Although frequent, base station handoff might happen in different time span. The differences in handoff frequency between multiple carriers can be explained by the differences in their base station distribution density, due to differences in population and volume of network traffic. As a typical example, Figure 1(d) shows the handoff of Carriers M and U along the B-F line during a period of 100 seconds. It is observed that handoff may lead to throughput decline. However, it is also potential to take advantage of the handoff frequency diversity of multiple carriers in the time dimension to avoid throughput decline.

In conclusion, carrier complementarity might contribute to better QoE on HSRs. In particular, it allows users to access a network with extensive LTE coverage and better signal quality. Moreover, it has the potential to improve robustness when encountering handoffs. There are two options for making use of carrier complementarity on HSRs, *i.e.*, *carrier selection* and *carrier conjunction*, which will be discussed next.

C. Carrier Selection and Carrier Conjunction

As analyzed above, if we can make use of carrier complementarity on HSRs, there is potential to achieve better QoE. This paper suggests two options, *i.e.*, carrier selection and carrier conjunction.

Carrier Selection. By selecting the best single path among multiple carriers to optimize availability [34], carrier selection could be a way for QoE improvement. For example, both Huawei and Xiaomi provide the option to select from at least two cellular carriers. However, on-demand carrier selection requires seamless handover from one carrier to another, which is currently an unsettled issue. In this paper, we just use the carrier selection as a baseline for evaluation in comparison experiments.

Carrier Conjunction. The other way to improve QoE might be carrier conjunction. However, the legacy single-path TCP may fail to take full advantage of multi-path transfer. This is because TCP is designed for connections that traverse a single path between host pairs. Out-of-data delivery via multiple paths degrades TCP throughput [22]. In this case, MPTCP [23] is an effort towards enabling the simultaneous use of Network Interface Controllers (NICs) achieved through a modification of TCP that presents a regular TCP interface to applications, while in fact spreading data across several subflows. That is, instead of just using a single network for data transfer, it is possible to have connectivity to multiple networks simultaneously. In this case, MPTCP is suggested to support carrier conjunction, which is called *multi-carrier MPTCP*.

This paper aims to answer the question: Carrier selection or carrier conjunction, which is better? To compare the performance between carrier selection and conjunction, we conduct various kinds of experiments as below. 1) Single-path TCP. Experiments using Carrier M, U and T are denoted by M single-path, U single-path and T single-path cases, respectively. 2) Dual-path MPTCP. We conduct experiments using Carriers M and U (denoted by *(M, U) dual-path cases*), Carriers M and T (denoted by *(M, T) dual-path cases*), and Carriers U and T (denoted by *(U, T) dual-path cases*), respectively. 3) Three-path MPTCP. Experiments are performed using Carrier M, U and T (denoted by *(M, U, T) three-path cases*). Besides, since MPTCP creates multiple subflows to support multi-path transfer, we use *M-subflow*, *U-subflow* and *T-subflow* interchangeably when referring to the MPTCP subflows using Carriers M, U and T, respectively.

III. Measurement Methodology

To explore multi-carrier MPTCP in high-speed mobility environments, in this section, we first describe the MPTCP provisioning and then the measurement approach. Then we discuss the challenges we meet and their corresponding solutions.

A. MPTCP Provisioning

The most popular multi-path solution with off-the-shelf Linux kernel MPTCP implementation is released by the IP Networking Lab [35], which includes configurable modules such as congestion control, scheduler, and buffer size. Our previous work [13] has explored the impact of scheduler and buffer size, this paper mainly focuses on the impact of congestion control.

MPTCP allows for adoption of different congestion controllers. We study two representatives: Reno [36] and LIA [24]. The former is studied as a baseline using TCP NewReno over each subflow. The latter is the default MPTCP congestion controller, which couples the additive increase function of subflows to manage congestion balancing. Reno is regarded as *uncoupled* congestion control, and LIA is regarded as *coupled* congestion control.

By default, the MPTCP in this paper adopts the data scheduler called *Default* [35]. *Default* is the default MPTCP scheduler which first sends data on a subflow with the lowest RTT until its congestion window (cwnd) is full. Then, it starts transmitting on the subflow with the next higher RTT. Both sending and receiving buffer sizes are set 16 MB to ensure that buffer does not become the bottleneck. As for single-path

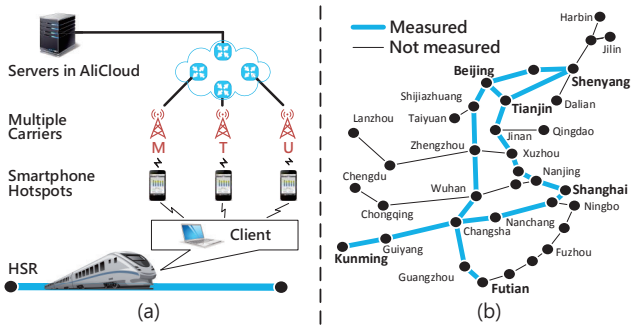


Fig. 2: Measurement setup and HSR footprints. (a) An example of multi-carrier MPTCP on HSRs (M, T and U are three different carriers). (b) Map of China HSR lines.

TCP, we set congestion controller as TCP NewReno, and set both sending and receiving buffers at 16 MB.

B. Measurement Setup and HSR Footprints

Figure 2 (a) illustrates an example of HSR measurement for MPTCP transfer via three different carriers. Mobile clients (*e.g.*, a wireless laptop) are in an HSR train and use the `wget` toolkit to download HTTP data via port 80 from servers. The servers are dedicated and lightly loaded, running in the AliCloud ECS [37]. Both the server and client run Ubuntu Linux 14.04 with MPTCP enabled.

HSR Line Footprints. As Figure 2 (b) shows, measurements are conducted on more than half of all the China HSR lines including six popular HSR routes, *i.e.*, Beijing-Futian (B-F), Beijing-Shenyang (B-Y), Beijing-Shanghai (B-H), Beijing-Tianjin (B-T), Beijing-Kunming (B-K), and Shanghai-Kunming (H-K) line. Table I shows the HSR line footprints in detail. Taking the B-F line as an example, we accumulate a mileage of 14,430 km in 6 one-way trips (the length of each one-way trip is 2,405 km).

TABLE I: HSR lines.

Route	Total Length (km)	*Static* (h)	*Low-speed* (h)	150-280 km/h (h)	*High-speed* (h)
B-F	14,430	11.70	11.90	12.00	32.68
B-Y	6,048	12.93	9.28	9.60	16.19
B-H	15,816	17.40	15.40	15.30	35.90
B-T	4,370	38.00	4.43	4.50	13.87
B-K	11,928	11.20	10.50	10.80	33.50
H-K	13,512	12.50	16.80	17.00	30.20

Trains experience four phases of motion: parking at stations, acceleration, running at full speed and deceleration. Since the duration of each one-way trip is 8.70 hours (excluding the parking time at the originating and terminal stations), in the B-F trips, the train parks at stations along the line for 11.70 hours (called *Static*), and the train runs at a speed between 0 and 150 km/h (called *Low-speed*), between 150 and 280 km/h, and between 280 and 310 km/h (called *High-speed*) for 11.90, 12.00 and 32.68 hours, respectively. Note that measurements are started about 30 minutes before the train leaves the originating station, and are maintained for about 30 minutes after the train arrives at the terminal, which ensures enough time for performing static measurements.

C. Measurement Challenges and Solutions

We are faced with several challenges in carrying out these measurements, which is expected to truly reflect the performance of MPTCP on HSRs and to facilitate data collection and analysis. First, neither the current implementation of MPTCP nor the smartphone has the built-in module that supports data transmission through two or more carrier networks simultaneously. Second, comparison experiments between MPTCP and single-path TCP should be carefully designed due to the tradeoff between synchronization and interference. Third, numerous network indicators during the data transmission intervals, such as signal strength, network type and LAC, need to be monitored and recorded for a multi-dimensional analysis, which requires dedicated measurement and analysis tools.

Multi-carrier Support. To deploy multi-carrier MPTCP, the first thing is to support multi-carrier network access. Although USB cellular modems (*e.g.*, ZTE MF832S) are commonly used for laptop to access cellular networks, we use smartphone hotspots instead[2]. This is because the open Application Program Interfaces(APIs) are enabled in smartphones to monitor indicators such as signal strength, network type, and base station parameters, while for USB cellular modems, APIs for signal monitoring are closed to developers. Figure 2 (a) illustrates the setup details. To further avoid measurement bias, each client (*e.g.*, Lenovo XiaoXin 510S-14ISK) disables its built-in wireless network card, and access the smartphone hotspot via a USB adapter instead.

Synchronization and Interference. The comparability between different cases demands for simultaneous data transmission, while interferences may exist among clients[3].

For synchronization, when comparing the throughput of both single-path TCP and dual-path MPTCP (*e.g.*, using Carriers M and U), we simultaneously start three experiments, *i.e.*, M single-path, U single-path, and (M, U) dual-path cases. Specifically, in order to ensure synchronism, each client operates clock synchronization via the Internet and starts data transmission at fixed time points such as 9:00, 9:05, 9:10, *etc*.

On the other hand, in theory this synchronization might lead to interference from the cellular network. That is, clients connected to the same carrier may compete for downlink bandwidth. For example, the client in the M single-path case and the client in the dual-path case (*i.e.*, M-subflow) might compete for the bandwidth of Carrier M. However, our measurements in both static and high-speed mobility environments demonstrate that this two-client interference in the cellular network is negligible[4]. We believe this is because base station usually allocates per-user resource separately in cellular networks [38], [39].

[2]Our comparison experiment shows that the performance of the hotspots approximates that of the USB cellular modems.

[3]Note that each client is allocated a dedicated server in the AliCloud ECS, thus we ignore the interferences in servers.

[4]For static measurements, we add an extra client when a client has been performing single-path case for 120 s. While in high-speed mobility environments, we conduct one single-path case for 10 one-way trips along the B-T line, and two single-path cases together for 10 one-way trips as well.

TABLE II: Dataset.

Route	Dataset: M-S		Dataset: T-S		Dataset: U-S		Dataset: MT-D		Dataset: MU-D		Dataset: TU-D		Dataset: MUT-T	
	# of flows	Size (GB)	# of flows	Size (GB)	# of flows	Size (GB)	# of flows	Size (GB)	# of flows	Size (GB)	# of flows	Size (GB)	# of flows	Size (GB)
B-F	78	35.48	109	50.86	223	102.91	333	154.95	133	61.51	477	222.37	0	0.00
B-Y	165	76.88	220	101.13	220	101.72	0	0.00	367	170.33	85	39.03	179	82.80
B-H	218	101.72	116	52.04	234	108.82	136	61.51	248	113.55	124	56.78	96	42.58
B-T	72	33.12	160	73.33	84	37.85	132	61.56	328	151.40	116	52.04	144	66.24
B-K	334	130.62	111	51.84	259	120.11	201	93.00	153	70.52	57	25.92	56	25.55
H-K	462	215.27	128	59.14	301	139.57	231	107.64	177	81.61	66	30.75	65	29.57

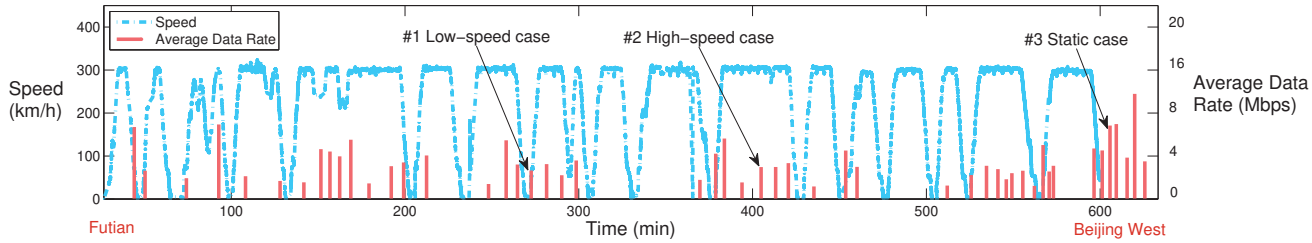


Fig. 3: Average data rate of each case along the B-F line.

Data Collection. Data is collected in three ways: 1) Packet capture. We capture all the packets at both clients and servers using `tcpdump` [40] for packet level analysis. 2) Communication log. We have developed a monitoring tool on the client to record the measurement parameters in the client such as carrier types, case IDs, start/end time, transmission states, *etc*. 3) Smartphone signal monitoring. We have further developed an android application on hotspot smartphones to get the geographical location and speed of the train via GPS, reads signal strength, network type, LAC and CID of base stations from the Android OS, and records all this in log files.

Table II shows details of the main dataset. According to carrier combinations and path numbers, we divide the dataset into seven sub-datasets: M-S, T-S, U-S, MT-D, MU-D, TU-D and MUT-T. For example, sub-dataset M-S contains data of M single-path cases, MU-D contains data of (M, U) dual-path cases, and MUT-T contains data of (M, U, T) three-path cases. For example, we have captured 223 data flows in U single-path cases on the B-F line, and the size of data collected is 102.91 GB, including captured packets, client technical information logs, and log files of hotspot smartphones.

IV. Comparison between MPTCP and TCP

In this section, we analyze single-path TCP (Datasets: M-S, U-S, and T-S), dual-path MPTCP (Datasets: MU-D, MT-D, and UT-D), and three-path MPTCP (Dataset: MUT-T) to estimate the MPTCP's performance in different mobility environments (*i.e.*, static, low-speed and high-speed) with different carrier combinations. Figure 3 gives an example of the dual-path MPTCP experiment in a whole one-way trip along the B-F line. Across fifteen stations from Futian to Beijing West, the train experiences four phases of motion: parking at stations, acceleration, running at full speed and deceleration. Each instance is conducted for 120 s, whose average data rate is illustrated by red stems. According to the speed range, we carefully pick out the test cases for performance analysis. For example, the marks #1, #2, and #3 in Figure 3 refer to cases in low-speed mobility environments, high-speed mobility environments and static environments, respectively.

A. Conjunction Efficiency

Before answering the question regarding whether and when MPTCP is superior to single-path TCP on HSRs, we first estimate the *conjunction efficiency* of MPTCP, U_c, which is defined as the ratio of the total average data rate of all subflows in MPTCP to the highest average data rate among N single-path TCP.

$$U_c = \frac{\sum R'_n}{\max(R_n)}, n = 1, 2, ..., N. \quad (1)$$

Where N is the number of carriers (each single-path or subflow uses one carrier). R_n is the average data rate of a single-path TCP, and R'_n is the average data rate of the subflow in MPTCP. When $U_c > 1$, the carrier conjunction is supposed to be superior to the carrier selection.

For overall performance evaluation of MPTCP, we conducted comparison experiments between single-path cases and MPTCP (*i.e.*, dual-path and three-path) cases, investigating MPTCP's conjunction efficiency in different mobility environments with different carrier combinations. As illustrated in Figure 4, we make three observations as detailed below.

First, the median variation of U_c from "static" to "high-speed" indicates that mobility reduces the conjunction efficiency of MPTCP. For example in Figure 4 (a), the median of U_c is as large as 1.4 in "static", while it is only nearly 1.0 in "high-speed". Other violin sub-figures in Figure 4 also draw the similar conclusions.

The second observation from Figure 4 relates to the fluctuation range of U_c. We see that mobility leads to a larger fluctuation range[5] of conjunction efficiency in "high-speed" than that in "static". For example, in Figure 4 (a), U_c in "static" ranges from 0.5 to 2.4, while in "high-speed" it fluctuates in a

[5]According to the definition, we have $U_c \in [0, 2]$. However, in practice, U_c may exceed its theoretical range (*i.e.*, $U_c > 2$) due to the instability of wireless environments.

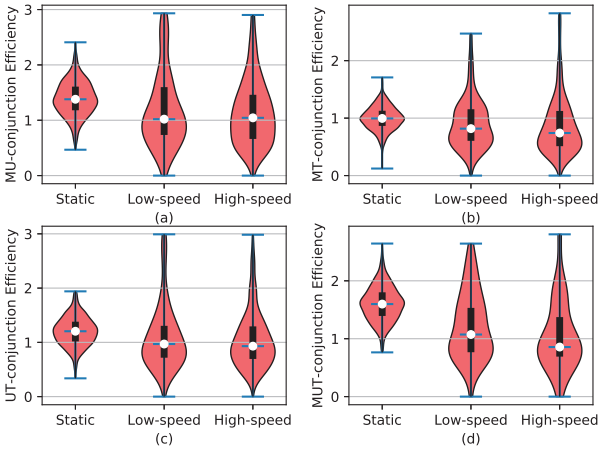


Fig. 4: Conjunction efficiency of MPTCP in different mobility environments (static: 0 km/h, low-speed: 0 $\sim$ 150 km/h, hign-speed: 280 $\sim$ 310 km/h) along HSRs. (a) (M, U) Dual-path cases. (b) (M, T) Dual-path cases. (c) (U, T) Dual-path cases. (d) (M, U, T) Three-path cases.

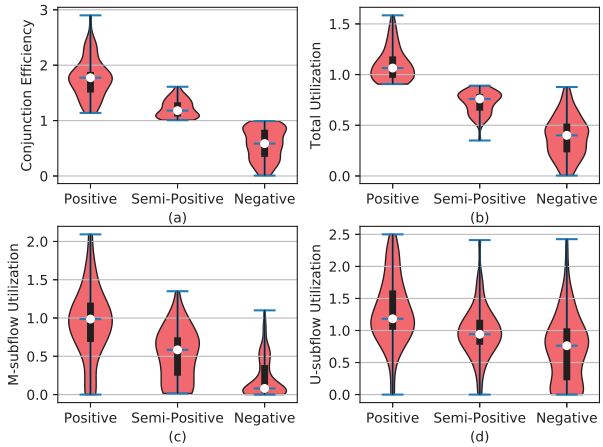


Fig. 5: Details of the classified instances of (M, U) dual-path in high-speed mobility environments. (a) Conjunction efficiency. (b) Total utilization. (c) Utilization of M-subflow denoted by $\frac{R'_M}{R_M}$. (d) Utilization of U-subflow denoted by $\frac{R'_U}{R_U}$.

wide range from 0 to 2.9. The same conclusion can be obtained in MPTCP cases using other carrier combinations.

Finally, according to all the violin sub-figures (*i.e.*, Figures 4 (a), (b), (c) and (d)), we can see the probability that $U_c > 1$ approximates 50% in "high-speed" regarding different carrier combinations (except for (M, T)). Specifically, the probability approximates 51%, 32%, 49% and 47% in (M, U), (M, T) and (U, T) dual-path cases, and in (M, U, T) three-path cases, respectively. It is worth mentioning that (M, T) shows a low conjunction efficiency, which indicates (M, T) has poorer carrier complementarity compared with the other carrier combinations. According to the definition of U_c, we therefore draw the conclusion that carrier conjunction is superior to carrier selection in nearly half of the cases with appropriate carrier combinations. In other words, carrier conjunction performs similarly to carrier selection in high-speed mobility environments.

Instance Classification. In addition to the conjunction efficiency, we further define *total utilization*, U_t as the ratio of the total average data rate of all subflows to the total average data rate of N single-path TCP.

$$U_t = \frac{\sum R'_n}{\sum (R_n)}, n = 1, 2, ..., N. \quad (2)$$

Test cases satisfying the condition $U_c < 1$ are referred to as *negative instances*, those that satisfy $(U_c \geq 1 \cap U_t > 90\%)$ *positive instances*[6], and all others *semi-positive instances*.

Take the (M , U) dual-path cases in "high-speed" as an example. The proportion of each kind of instance is 21% (positive), 29% (semi-positive) and 50% (negative), respectively. Figure 5 gives an in-depth performance analysis of the classified instances. Figure 5(a) indicates that all kinds of instances (*i.e.*, positive, semi-positive and negative instances) exist on HSRs. Figure 5(b) illustrates the total utilization of MPTCP. For the negative instances, we can see that over 75% of the negative instances suffer from poor performance ($U_t < 50\%$). We further investigate the utilization of subflows (denoted by $\frac{R'_n}{R_n}$) as shown in Figures 5(c) and (d). It can be observed that the positive instances can achieve high utilization of both subflows, and both the semi-positive and negative instances suffer from *utilization imbalance* between subflows.

Instance Analysis. To explore the utilization imbalance between subflows, we further analyze instances in terms of throughput calculated using the bits downloaded from the server per second.

Figures 6(a), 6(b) and 6(c) show the throughput of a positive, semi-positive and negative instance, respectively. In particular, Figure 6(a) demonstrates that in the positive instance the throughput of both subflows approximate to those of the single-path flows. Therefore, the positive instance can achieve high utilization of both subflows, resulting in a high conjunction efficiency. For the semi-positive instance shown in Figure 6(b), the U-subflow achieves a high throughput for the duration. However, the M-subflow encounters a sudden decrease in throughput at 65 s and suffers from durative low throughput afterwards. In this paper, this event is called *subflow degradation* of MPTCP. Figure 6(c) depicts the throughput of a negative instance, where it is observed that the subflow degradation occurs earlier, resulting in poor conjunction efficiency as well as low total utilization. Figure 6(d) further reveals that subflow degradations are ubiquitous in all kinds of instances.

Subflow degradation. Statistical results show that the events of subflow degradation occur in 39.7% of the positive instances, in 62.5% of the semi-positive instances, and in

[6]The barrier of U_t can be configured according to different application scenarios. For example, the barrier for video streaming should be set larger than that for web-browsing. In our analysis, we set the barrier 90% for a strict mode.

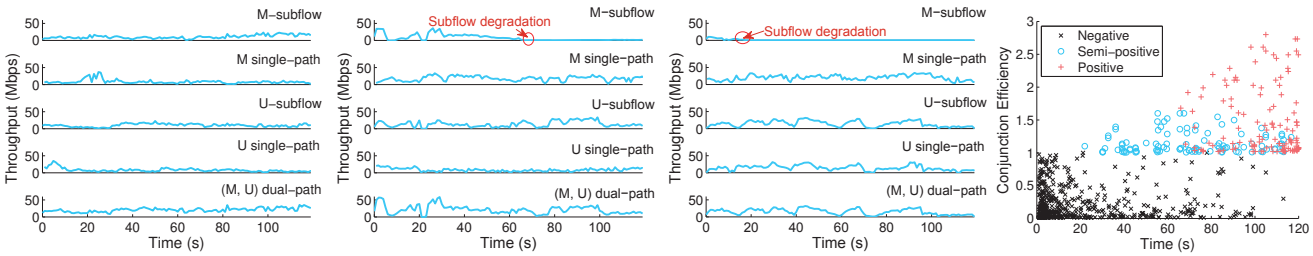


Fig. 6: Subflow degradation analysis. (a), (b) and (c) depict the throughput of a positive, semi-positive and negative instance, respectively. (d) illustrates the time points of subflow degradations.

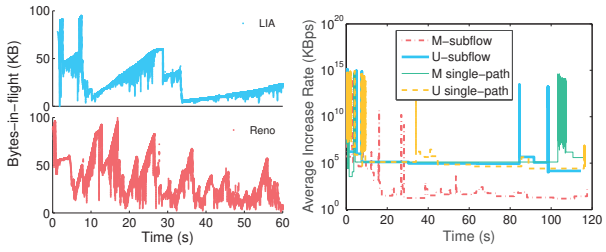


Fig. 7: Subflow degradation analysis. (a) depicts the difference in linear increase rate between LIA and Reno, and (b) illustrates the average cwnd increase rate.

100% of the negative instances, respectively. Our measurement shows that the proportion will further increase with a longer data transmission ($>$ 120 s). It is demonstrated that the earlier the subflow degradation occurs, the poorer the conjunction efficiency. Since subflow degradation leads to low efficiency of MPTCP, and our statistical results indicate that single-path cases can easily recover from a sudden decrease in throughput, this poses the question on why subflow degradations occur in MPTCP rather than in single-path TCP on HSRs.

As mentioned above, we use LIA and Reno as the congestion controllers for MPTCP and single-path TCP, respectively. Packet loss, as we know, is regarded as the congestion signal by both LIA and Reno. When encountering packet loss, the cwnd decreases by half, then increases linearly, namely additive increase and multiplicative decrease (AIMD). We denote I as the set of all subflows. w_i and rtt_i are the cwnd size and RTT of subflow i ($i \in I$), respectively. w is the total cwnd size over all the subflows. LIA uses unmodified TCP behavior in the case of a packet loss. During the congestion avoidance, it couples the additive increase function of the subflows [41]. Therefore, for each ACK received on subflow i, it increases w_i by $\min(\frac{\alpha b M_i}{w}, \frac{bM_i}{w_i})$. Where α is a function of w_i and rtt_i, b is the bytes acknowledged, and M_i is the maximum segment size, for all $i \in I$. Besides, as to single-path TCP, for each ACK received, Reno increases its cwnd w_{reno} by $\frac{bM_i}{w_{reno}}$.

Figure 7(a) gives an example of two flows using LIA and Reno, respectively. We find that after multiple multiplicative decreases, the linear increase rate of LIA becomes lower and lower, while that of Reno is relatively stable. Besides, our statistical results shows that the increase rate of LIA is much slower than that of Reno on HSRs. This reveals that LIA's additive increase function does not adapt well to the high-speed mobility.

On the other hand, as demonstrated above, mobility declines the conjunction efficiency of MPTCP. Although it is obvious that higher train speed results in more frequent handoffs, how does these handoffs affect the conjunction efficiency is still "in the wild". In order to explore what happened to MPTCP when the subflow degradation occurs, for the negative instance discussed in Figure 6(c), we take a depth analysis into its cwnd variation in Figure 7(b). First of all, we define *bytes-in-flight* as the number of bytes that has been sent but not yet acknowledged. Bytes-in-flight can reflect the size of slide window (swnd), which is the minimum of cwnd and receive window (rwnd) [42]. We find that when the sending and receiving buffer are large enough (*e.g.*, 16 MB), the rwnd for both subflows is always much larger than the bytes-in-flight, so swnd is throttled by cwnd. That is to say, the number of bytes-in-flight can be used to approximate the cwnd of each subflow.

To estimate the cwnd variation, we define k_t, the average increase rate of cwnd between two times of packet loss (time interval is t s), which is calculated as follow:

$$k_t = \frac{BIF_t}{t} \tag{3}$$

where BIF_t is the total number of bytes-in-flight during t s. As shown in Figure 7(b), the average increase rate of the M-subflow encounters a continuous decline before 20 s (time point of subflow degradation as illustrated in Figure 6(c)). After that, the average cwnd increase rate of the M-subflow maintains much lower than those of the U-subflow and the single-path TCP. Under these circumstances, MPTCP shows poor adaptability to frequent packet loss, which is usually caused by handoffs in high-speed mobility environments.

Summary. Based on the above analysis, we highlight three insights. First, network performance is poorer in mobility environments than in static ones, and carrier conjunction shows similar performance to carrier selection on HSRs. Second, the low utilization of semi-positive and negative instances are mainly caused by subflow degradations of MPTCP. Finally, MPTCP's poor adaptability to frequent packet loss leads to

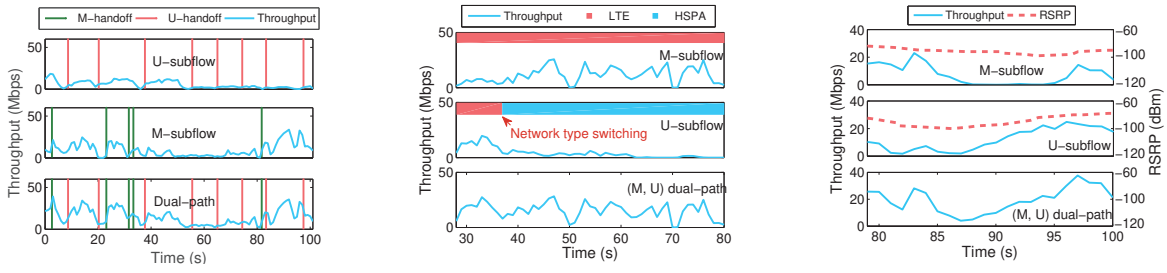


Fig. 8: Complementarity analysis. (a), (b) and (c) illustrate the complementarity in handoff, network type, and signal without handoffs, respectively.

subflow degradations, and its performance is still far from satisfactory in mobility environments.

B. Complementarity

It is observed that subflow degradations lead to low conjunction efficiency of multi-carrier MPTCP, *i.e.*, increasing the proportion of semi-positive and negative instances. In this section, we focus on the instance durations without subflow degradations to estimate the benefit of carrier conjunction.

Complementarity in Handoff. As mentioned in Section II, handoffs on subflows may happen asynchronously. For example in Figure 8(a), at the moments 8 s, 20 s and 35 s, the U-subflow suffers from low-throughput transmission due to handoffs, while the M-subflow can still provide an acceptable data rate. Under these circumstances, the total throughput of the dual-path case is enhanced by handoff complementarity.

Complementarity in Network Type. For network type handover, Figure 8(b) gives an example of the correlation of LTE coverage and throughput. It can be observed that when covered by LTE, the throughput is relatively higher than without LTE coverage. Although throughput of the U-subflow drops at 35 s due to network type handover, the total throughput of MPTCP does not decrease because the M-subflow is always transmitted in LTE networks. This indicates the benefit of network type complementarity.

Complementarity in Signal. On the other hand, to explore the complementarity in network signal, we investigate the instance durations without handoffs. Figure 8(c) illustrates the correlation of signal strength and throughput. It is observed that the signal strength of the M-subflow decreases while that of the U-subflow increases from 87 s to 93 s. Therefore, we can see that signal complementarity benefits the total throughput of the dual-path case in this duration.

In summary, it can be demonstrated that without subflow degradations, multi-carrier MPTCP is capable of making use of the complementarity of subflows, achieving higher conjunction efficiency. Similar results can be obtained by analyzing dual-path cases using other combinations of carriers (*i.e.*, (M, T) and (U, T)) as well as three-path cases (*i.e.*, (M, U, T)).

V. Discussions and Future Work

In this paper, we find that, subflow degradations significantly impact the complementarity of multiple cellular carriers. This is because the coupled congestion control algorithms are designed to transfer traffic from a congested path to a less congested path, lead to more significant window distribution imbalance. Extreme window distribution imbalance leads to significant traffic distribution imbalance, preventing MPTCP from showing enough advantages. Since MPTCP cannot differentiate between congestion and handoff, its congestion control cannot adapt well to frequent handoffs, hurting the performance significantly. To migrate subflow degradations, we give some possible directions on improving MPTCP below.

Window adaption. The additive increase and multiplicative decrease approach used by MPTCP Reno and LIA cannot adapt fast enough to the rapid network capability variation on HSRs. It is recommended to apply more adaptive solution to increase/decrease the window or sending rate [43]. For example, the approach used by Sprout [44]. Since packet loss has significant impact on window increase, it is also recommended to adopt new coupled congestion control that is not loss-based [45], for example, the state-of-the-art non-loss-based congestion controller called BBR [46].

Mobility Management. Designing robust schemes for mobility management should be a feasible way to improve multi-carrier MPTCP performance on HSRs. For example, Li *et al.* [47] has demonstrated that the fundamental cause of unreliable cellular transmission in extreme mobility is its wireless signal strength-based design. They proposed a movement-based mobility management scheme, which is a signaling overlay in the delay-Doppler domain, extracting client movement and multi-path profile with the recently proposed orthogonal time-frequency space (OTFS) modulation.

Differentiating between congestion and handoff. MPTCP always attributes packet loss to congestion, hence conducting very aggressive congestion control when heavy packet drops occur due to handoff. MPTCP should differentiate between congestion and handoff in an end-to-end way. For example, Sinky *et al.* detect handoff with cross-layer assistance [48]. By detecting handoffs, we can retransmit lost packet of the handoff path via other path that is not suffering a handoff. Xu *et al.* [49] also design a cross-layer aided mechanism based on deep reinforcement learning to alleviate performance degradation problems induced by handover.

Enhanced coupled congestion control. Some state-of-the-art works [50], [51] have focused on coupled congestion control for MPTCP on HSRs, which is proved to be effective for MPTCP performance enhancement. In order to make good use of the difference in handoff time between multiple carriers, it is also recommended that MPTCP pauses the timeout timer and freezes data transmission of a path suffering a handoff. At the same time, MPTCP can allow the window of other paths not suffering a handoff to rise much faster than usual. Using such a coupled congestion control, traffic can be transferred timely and accurately from paths suffering a handoff to paths not suffering a handoff. This would ensure that total throughput of all sub-flows can keep relatively stable even when any path suffers a handoff, and the resistance of MPTCP to frequent handoff is enhanced.

Applying uncoupled congestion control. Only coupled congestion control such as LIA induces subflow degradations. Hence, it is recommended to adopt uncoupled congestion control directly if fairness is not the dominant factor.

VI. Related Work

With regard to scenarios with speeds of 200 km/h or more, Merz *et al.* [9] test an LTE system in a train with velocities up to 200 km/h, and the results show that the performance of LTE decreases with increasing velocity. Jang *et al.* [10] take short-distance measurements in a fast moving car at 300 km/h, and analyzes of the performance of UDP and TCP over 3G and 3.5G wireless networks. They point out that mobile nodes in such a high speed will suffer from far worse performance than static nodes even in the same network. Xiao *et al.* [11] carry out a measurement study in mobile data networks under high-speed mobility, and the results expose the degradation and large variance of throughput and RTT. Li *et al.* [12], Wang *et al.* [52], and Cui *et al.* [53] investigate TCP behaviors on HSR with a peak speed of 310 km/h. All these measurements obtain an corresponding conclusion that TCP's performance declines greatly when encountering frequent handoffs. The measurement results show that performance declines greatly under such extremely high speed conditions. For example, the RTO rate is high spurious, aggressive congestion window reduces, connection establishment and closure is slow, transmissions are interrupted.

All the above measurements obtain an accordant conclusion that performance of single-path TCP declines greatly when encountering frequent handoffs, and the inefficiency of the traditional single-path TCP turns out to be inadaptable on HSRs.

Paasch *et al.* [26] enable smooth handovers for WiFi and Cellular, and proves MPTCP's effectiveness in the current Internet. For performance measurements, Chen *et al.* [27] compare the latency measurements under MPTCP (over WiFi and Cellular) and the single-path TCP (over WiFi or Cellular), indicating that MPTCP reduces the variability in download latencies. Nguyen *et al.* [28] adopt MPTCP and present a cross-layer wireless virtualization approach, which improves performance of mobile WiFi users and achieves seamless handover. Another significant work for measurement study on multi-path transfer over mobile devices are conducted by Nikravesh *et al.* [54], whose goal is to provide key knowledge and vital clues for evolving the mobile multi-path design. For the wired cases, Baidya *et al.* [29] disclose some inefficiencies of MPTCP, for example, MPTCP throughput decreases below the level of single-path TCP throughput when paths have significantly different bandwidths. In addition, some literature focuses on MPTCP in mobility environments. Williams *et al.* [30] conduct experiments to augment cellular 3G connections and find that MPTCP provides benefits in the vehicle-based field test. Although the above literature covers most aspects of MPTCP, they are all conducted in the static or low-speed environments.

Li *et al.* [13] fill the void by making a detailed measurement study on MPTCP via two carrier networks in high-speed (> 300 km/h) mobility scenarios. They have demonstrated that MPTCP's value is proven mostly in reliability enhancement rather than bandwidth aggregation. However, our work takes one step further to explore up to three cellular carriers and takes an in-depth analysis on the reason why MPTCP fails to achieve bandwidth aggregation from the perspective of subflow degradation.

VII. Conclusion

In this paper, we focus on the comparison between carrier conjunction and carrier selection among up to three cellular carriers. We have conducted experiments that are carefully designed on various HSR routes in China. Measurement results show that network performance is poorer in mobility environments, and generally carrier conjunction performs similarly to carrier selection. This reveals that increasing the number of carriers cannot improve performance in the context of the legacy MPTCP. To the best of our knowledge, this is the first work that presents the concept of *subflow degradation* and its corresponding phenomenons of MPTCP behaviors in the wild. Furthermore, based on the observations on the instance details, we confirm that subflow degradation occurs due to MPTCP's poor adaptability to frequent handoffs. Thus, only by migrating subflow degradation in the MPTCP protocol design, can the carrier conjunction gain benefits. Based on these observations, in order to improve MPTCP performance in high speed mobility scenarios, we discuss the possible directions such as window adaption, mobility management, differentiating between congestion and handoff, enhanced coupled congestion control, and applying uncoupled congestion control.

VIII. Acknowledgment

We thank Ke Xu, Kai Zheng, Dan Wang, Meng Shen, and Chunyi Peng for feedback throughout this project. This article reflects only the authors' view. The European Union Commission is not responsible for any use that may be made of the information it contains. Li Li is the corresponding author.

References

[1] T. Li, K. Xu, H. Huang, X. Du, and K. Zheng, "Wip: When rdma meets wireless," in *IEEE WoWMoM*, 2022, pp. 1–4.

[2] H. Wang, T. Li, R. Shea, X. Ma, F. Wang, J. Liu, and K. Xu, "Toward cloud-based distributed interactive applications: measurement, modeling, and analysis," *IEEE/ACM TON*, vol. 26, no. 1, pp. 3–16, 2017.

[3] F. Zhang, J. Zhai, X. Shen, O. Mutlu, and X. Du, "POCLib: a high-performance framework for enabling near orthogonal processing on compression," *IEEE TPDS*, vol. 33, no. 2, pp. 459–475, 2022.

[4] F. Zhang, J. Zhai, X. Shen, D. Wang, Z. Chen, O. Mutlu, W. Chen, and X. Du, "TADOC: Text analytics directly on compression," *The VLDB Journal*, vol. 30, no. 2, pp. 163–188, 2021.

[5] F. Zhang, Z. Chen, C. Zhang, A. C. Zhou, J. Zhai, and X. Du, "An efficient parallel secure machine learning framework on GPUs," *IEEE TPDS*, vol. 32, no. 9, pp. 2262–2276, 2021.

[6] K. Xu, T. Li, H. Wang, H. Li, W. Zhu, J. Liu, and S. Lin, "Modeling, analysis, and implementation of universal acceleration platform across online video sharing sites," *IEEE TSC*, vol. 11, no. 3, pp. 534–548, 2018.

[7] K. Xu, L. Lv, T. Li, M. Shen, H. Wang, and K. Yang, "Minimizing tardiness for data-intensive applications in heterogeneous systems: A matching theory perspective," *IEEE TPDS*, vol. 31, no. 1, pp. 144–158, 2019.

[8] T. Li, K. Xu, M. Sheng, H. Wang, K. Yang, and Y. Zhang, "Towards minimal tardiness of data-intensive applications in heterogeneous networks," in *IEEE ICCCN*, 2016, pp. 1–9.

[9] R. Merz, D. Wenger, D. Scanferla, and S. Mauron, "Performance of lte in a high-velocity environment: a measurement study," in *ACM SIGCOMM Workshop*, 2014, pp. 47–52.

[10] 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 *ACM MICNET*, 2009.

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

[12] L. Li, K. Xu, D. Wang, C. Peng, Q. Xiao, and R. Mijumbi, "A Measurement Study on TCP Behaviors in HSPA+ Networks on High-speed Rails," in *IEEE INFOCOM*, 2015.

[13] L. Li, K. Xu, T. Li, K. Zheng, C. Peng, D. Wang, X. Wang, M. Shen, and R. Mijumbi, "A measurement study on multi-path tcp with multiple cellular carriers on high speed rails," in *ACM SIGCOMM*, 2018, p. 161–175.

[14] T. Li, K. Zheng, K. Xu, R. A. Jadhav, T. Xiong, K. Winstein, and K. Tan, "Tack: Improving wireless transport performance by taming acknowledgments," in *ACM SIGCOMM*, 2020, pp. 15–30.

[15] M. Alizadeh, A. Greenberg, D. A. Maltz, J. Padhye, P. Patel, B. Prabhakar, S. Sengupta, and M. Sridharan, "Data center tcp (dctcp)," in *ACM SIGCOMM*, 2010, pp. 63–74.

[16] C. Wilson, H. Ballani, T. Karagiannis, and A. Rowtron, "Better never than late: Meeting deadlines in datacenter networks," in *ACM SIGCOMM*, 2011, pp. 50–61.

[17] B. Vamanan, J. Hasan, and T. Vijaykumar, "Deadline-aware datacenter tcp (d2tcp)," *ACM SIGCOMM*, pp. 115–126, 2012.

[18] M. Alizadeh, S. Yang, M. Sharif, S. Katti, N. McKeown, B. Prabhakar, and S. Shenker, "pfabric: Minimal near-optimal datacenter transport," in *ACM SIGCOMM*, 2013, pp. 435–446.

[19] F. R. Dogar, T. Karagiannis, H. Ballani, and A. Rowstron, "Decentralized task-aware scheduling for data center networks," in *ACM SIGCOMM*, 2014, pp. 431–442.

[20] C. W. Lee, M. C. Chen, and Y. S. Sun, "A novel network mobility management scheme supporting seamless handover for high-speed trains," *Computer Communications*, vol. 37, no. 1, pp. 53–63, 2014.

[21] T. Li, K. Zheng, and K. Xu, "Acknowledgment on demand for transport control," *IEEE Internet Computing*, vol. 25, no. 2, pp. 109–115, 2021.

[22] C. Barakat, E. Altman, and W. Dabbous, "On tcp performance in a heterogeneous network: a survey," *IEEE Communications Magazine*, vol. 38, no. 1, pp. 40–46, 2000.

[23] A. Ford, C. Raiciu, and M. Handley, "RFC 6824: TCP extensions for multipath operation with multiple addresses," 2013.

[24] C. Raiciu, M. Handly, and D. Wischik, "RFC 6356: Coupled congestion control for multipath transport protocols," 2011.

[25] N. Gast, R. Khalili, J. Y. L. Boudec, and M. Popovic, "Opportunistic linked-increases congestion control algorithm for mptcp," *IETF Draft*, 2014.

[26] C. Paasch, G. Detal, F. Duchene, C. Raiciu, and O. Bonaventure, "Exploring mobile/wifi handover with multipath tcp," in *ACM SIGCOMM workshop*, 2012, pp. 31–36.

[27] Y.-C. Chen, Y.-s. Lim, R. J. Gibbens, E. M. Nahum, R. Khalili, and D. Towsley, "A measurement-based study of multipath tcp performance over wireless networks," in *ACM IMC*, 2013, pp. 455–468.

[28] K. Nguyen, Y. Ji, and S. Yamada, "Improving wifi networking with concurrent connections and multipath tcp," in *IEEE WoWMoM*, 2013, pp. 1–3.

[29] S. H. Baidya and R. Prakash, "Improving the performance of multipath tcp over heterogeneous paths using slow path adaptation," in *IEEE ICC*, 2014, pp. 3222–3227.

[30] N. Williams, P. Abeysekera, N. Dyer, H. Vu, and G. Armitage, "Multipath tcp in vehicular to infrastructure communications," Tech. Rep., 2014.

[31] F. Luan, Y. Zhang, L. Xiao, and C. Zhou, "Fading characteristics of wireless channel on high-speed railway in hilly terrain scenario," *International Journal of Antennas and Propagation*, vol. 2013, no. 3, pp. 188–192, 2013.

[32] X. Zhang, Z. Qi, G. Min, W. Miao, Q. Fan, and Z. Ma, "Cooperative edge caching based on temporal convolutional networks," *IEEE TPDS*, vol. 33, no. 9, pp. 2093–2105, 2022.

[33] "Signal Strength," http://blog.industrialnetworking.com/2014/04/making-sense-of-signal-strengthsignal.html.

[34] A. Dhamdhere and C. Dovrolis, "Isp and egress path selection for multihomed networks," in *IEEE INFOCOM*, 2006, pp. 1–12.

[35] "Multipath TCP in the Linux Kernel," http://www.multipath-tcp.org.

[36] S. Floyd and T. Henderson, "The newreno modification to tcp's fast recovery algorithm," *Expires*, vol. 345, no. 2, pp. 414–418, 1999.

[37] "AliCloud ECS," https://www.aliyun.com/.

[38] T. Li, C. S. Magurawalage, K. Wang, K. Xu, K. Yang, and H. Wang, "On efficient offloading control in cloud radio access network with mobile edge computing," in *IEEE ICDCS*, 2017, pp. 2258–2263.

[39] T. Li, K. Wang, K. Xu, K. Yang, C. S. Magurawalage, and H. Wang, "Communication and computation cooperation in cloud radio access network with mobile edge computing," *Springer CTON*, vol. 2, no. 1, pp. 43–56, 2019.

[40] "TCPDUMP," http://www.tcpdump.org/.

[41] D. Wischik, C. Raiciu, A. Greenhalgh, and M. Handley, "Design, implementation and evaluation of congestion control for multipath tcp," in *Usenix NSDI*, 2011, pp. 99–112.

[42] T. Li, K. Zheng, K. Xu, R. A. Jadhav, T. Xiong, K. Winstein, and K. Tan, "Revisiting acknowledgment mechanism for transport control: Modeling, analysis, and implementation," *IEEE/ACM TON*, vol. 29, no. 6, pp. 2678–2692, 2021.

[43] X. Du, T. Li, L. Xu, K. Zheng, M. Shen, B. Wu, and K. Xu, "R-aqm: Reverse ack active queue management in multi-tenant data centers," in *IEEE ICNP*, 2021, pp. 1–12.

[44] K. Winstein, A. Sivaraman, and H. Balakrishnan, "Stochastic forecasts achieve high throughput and low delay over cellular networks," in *USENIX NSDI*, 2013, pp. 459–471.

[45] H. Xie and T. Li, "Revisiting loss recovery for high-speed transmission," in *IEEE WCNC*, 2022, pp. 1–6.

[46] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, "Bbr: Congestion-based congestion control," *ACM Queue*, vol. 14, no. 5, p. 20–53, 2016.

[47] Y. Li, Q. Li, Z. Zhang, G. Baig, L. Qiu, and S. Lu, "Beyond 5g: Reliable extreme mobility management," in *ACM SIGCOMM*, 2020, p. 344–358.

[48] H. Sinky, B. Hamdaoui, and M. Guizani, "Proactive multipath tcp for seamless handoff in heterogeneous wireless access networks," *IEEE TWC*, vol. 15, no. 7, pp. 4754–4764, 2016.

[49] J. Xu and B. Ai, "Deep reinforcement learning for handover-aware mptcp congestion control in space-ground integrated network of railways," *IEEE Wireless Communications*, pp. 1–8, 2021.

[50] J. Xu, B. Ai, L. Chen, L. Pei, Y. Li, and Y. Y. Nazaruddin, "When high-speed railway networks meet multipath tcp: Supporting dependable communications," *IEEE Wireless Communications Letters*, vol. 9, no. 2, pp. 202–205, 2020.

[51] J. Han, K. Xue, Y. Xing, P. Hong, and D. S. Wei, "Measurement and redesign of bbr-based mptcp," in *ACM SIGCOMM Posters and Demos*, 2019, p. 75–77.

[52] J. Wang, Y. Zheng, Y. Ni, C. Xu, F. Qian, W. Li, W. Jiang, Y. Cheng, Z. Cheng, Y. Li, X. Xie, Y. Sun, and Z. Wang, "An active-passive measurement study of tcp performance over lte on high-speed rails," in *ACM MobiCom*, 2019, pp. 1–16.

[53] L. Cui, Z. Yuan, Z. Ming, and S. Yang, "Improving the congestion control performance for mobile networks in high-speed railway via deep reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 5864–5875, 2020.

[54] A. Nikravesh, Y. Guo, F. Qian, Z. M. Mao, and S. Sen, "An in-depth understanding of multipath tcp on mobile devices: measurement and system design," in *ACM MOBICOM*, 2016, pp. 189–201.