Improving Onboard Internet Services for High-Speed Vehicles by Multipath Transmission in Heterogeneous Wireless Networks

Ping Dong, *Member, IEEE,* Bin Song*, Hongke Zhang, Xiaojiang Du, *Senior Member, IEEE*

***Abstract*—With the development of modern high-speed vehicles and mobile communication systems, there is a strong demand for operators to provide stable and high-quality onboard Internet services. However, the vehicle-to-ground links may suffer from several problems, such as insufficient bandwidth and long round-trip time. Although multiple wireless networks, such as WiFi, 3G and 4G, may be available along a track, onboard users hardly have good experiences if they can only visit the Internet via one wireless network, which may be subject to coverage gaps and signal attenuation. In this paper, the link qualities of multiple existing 3G and 4G technologies are first measured in a typical high-speed environment. These 3G/4G networks are candidates for vehicle-to-ground communication. Then, a concurrent multipath transmission scheme together with a network adaptive scheduling algorithm is proposed. The scheme is independent of the protocol stack so that it is easy to deploy. It also provides transparent Internet services for users without requiring the participation of the user devices in any multipath signaling. Meanwhile, the scheduling algorithm works at the network layer, instead of the transport layer, to meet the diverse transmission requirements of the connection-oriented and connectionless user applications. The algorithm can effectively aggregate the bandwidth of multiple available wireless links, as well as avoid the reordering of packets based on both the practical tracing databases and active path monitoring. Analysis and experiments show that the proposed algorithm can provide better onboard Internet services with lower cache requirements than the Earliest Delivery Path First (EDPF) and Weighted Round Robin scheduling algorithms, in terms of bandwidth improvement and packet disorder reduction.**

***Index Terms*—mobile Internet, vehicles, schedule, multiple paths**

This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant 2014JBM004 and 2015JBM001, by the National Natural Science Foundation of China under Grant 61271202, by the National 973 Program of China under Grant 2013CB329100, by the US National Science Foundation under Grant CNS-1065444, as well as by the US Army Research Office under Grant WF911NF-14-1-0518.

Ping Dong is with Beijing Jiaotong University, Beijing 100044, China (e-mail: pdong@bjtu.edu.cn).

Bin Song is with Xidian University, Xi'an 710071, China (e-mail: bsong@mail.xidian.edu.cn). (Corresponding author)

Hongke Zhang is with Beijing Jiaotong University, Beijing 100044, China (e-mail: hkzhang@bjtu.edu.cn).

Xiaojiang Du is with Temple University, Philadelphia, PA 19122, USA (e-mail: xjdu@temple.edu).

I. Introduction

Today, with the rapid development of modern high-speed vehicles, providing stable and high-quality mobile Internet services on board is becoming a rising and inspiring trend for both passengers and operators. Vehicles with increased speeds, such as High-Speed Trains (HSTs) that exceed 300km/h, are becoming popular among long-distance travelers because of their convenient, stable, and relatively spacious environment. Nowadays, China, Japan, USA, Germany, and France are rapidly deploying their national high-speed rail networks [1]. According to an official statistic [2] published by the National Railway Administration in April 2015, the number of passengers has reached 2.357 billion in 2014 in China. In USA, a plan [3] for a 17,000 mile national high-speed rail system has been outlined for completion by 2030.

Meanwhile, with the boom of mobile communication technologies, such as WiFi and Long-Term Evolution (LTE), people have become accustomed to using various mobile applications. A report [4] from the China Internet Network Information Center shows that at the end of 2014, the amount of mobile network users has reached 557 million, and 85.8% of the mobile users visit the Internet by their smart mobile terminals. A report from WiFi Alliance, the certification authority of WiFi devices, shows that about 4.5 billion WiFi products are in use all over the world today [5]. Therefore, there is a strong demand to provide high-quality onboard Internet services to the passengers on high-speed vehicles.

However, there are several special issues associated with high-speed mobility [6] such as: high penetration losses of signals, severe Doppler shift, and frequent handoffs. All of these issues are faced by the mobile network systems for the high-speed vehicles. These problems may lead to the instability of popular Internet services and result in poor Quality of Service (QoS).

Although multiple wireless networks, such as WiFi, 3G, 4G, may be available along a track, unfortunately, due to the limitations of user devices, onboard users hardly have good experiences if their devices can only visit the Internet via one wireless network (e.g., laptops with WiFi, which may be subject to coverage gaps and signal attenuation).

To solve these problems, some proposed onboard network schemes designed mobile routers with multiple interfaces. These interfaces may support various wireless technologies (e.g., WiFi, cellular network and satellite technology). Seamless handover [7] extends NEMO [8] by using multiple interfaces of the same technology (LTE) to support high-speed mobility. Dynamic network selection [9] was proposed to switch from a poor performance network to a better one. However, most of the research in this domain has been confined to single interface usage at any given time, while other interfaces work as backups or assistances for fast handover.

The use of multiple wireless interfaces simultaneously opens a new way of solving some problems in high-speed scenarios and can bring some new and interesting opportunities, such as bandwidth aggregation, seamless handover, and reliability. A concurrent multipath transmission scheme can satisfy the desire of operators to achieve stable and high-quality vehicle-to-ground communications. A survey found that 78% of the business travelers asked said they would use Wireless Local Area Network (WLAN) if it was available [10]. However, until recently, passengers on high-speed vehicles, such as bullet trains, have not been able to achieve stable high-speed Internet services. The multipath transmission scheme can provide a faster, more reliable Internet service for onboard users. It also allows the vehicle operators to deliver information to the passengers. In addition to these benefits, broadband access on vehicles is also imperative in vehicle control and safety by allowing an operations center to carry out fault diagnostics, video surveillance, and so on.

In this paper, we introduce a scheme of concurrent multipath transmission based on multiple interfaces for high-speed mobility. A network adaptive multipath scheduling algorithm is also proposed. Our main contributions are as follows:

(1) Some practical experiments were performed to evaluate the characteristics of multiple 3G/4G networks, i.e., EV-DO, FDD-LTE/HSPA+ and TD-LTE, under different statuses (static and high-speed mobility).

(2) A multipath transmission architecture, together with a network adaptive multipath scheduling algorithm, is proposed, which provides transparent Internet services for the users through multiple wireless technologies without requiring the participation of the user devices in any multipath signaling.

(3) The proposed algorithm can adapt to the changing of path conditions, and achieve good bandwidth aggregation while avoiding the reorder of packets with lower cache requirement than Earliest Delivery Path First (EDPF) and Weighted Round Robin scheduling algorithms.

The remainder of this paper is organized as follows. Some related work is introduced in Section II. The results of our practical experiments on high-speed trains are shown in Section III. A multipath transmission architecture, together with a network adaptive multipath scheduling algorithm, is proposed in Section IV. Some of the properties of our algorithm are analyzed in Section V. Simulations and experimental evaluations are provided in Section VI followed by the conclusion and Appendix A where a lemma is proven in detail.

II. Related Work

Some contributions were proposed to take advantage of multiple interfaces and multiple paths. They work at different protocol layers and require different changes to different network elements. We briefly discuss the solutions related to high-speed mobility in this section.

A. Link-layer Solutions

A scheme based on multiple interfaces was adopted early for vehicle-to-ground communication by Fast Train Hosts (FIFTH) project [11] in 2003. It provides Internet services for trains by satellites and fills the coverage gaps by WiFi. However, the expensive price, limited bandwidth and long propagation delay limit the large-scale application of a satellite-based approach. Compared to the literature's work, our method employs multiple wireless technologies, and selects all (or some) of the interfaces to work simultaneously for vehicle-to-ground communication.

With the popularity of WLAN technology, a link-layer design called *information raining* [12] was proposed in 2005, which uses a number of repeaters placed along a track and multiple antennas installed on the roof of a vehicle to enhance the throughput on board. An optimal antenna assignment strategy [13] was proposed for employing maximum ratio combining. Along with the rise of Wireless Metropolitan Area Networks (WirelessMAN), the idea of *information raining* was extended in [14] to support network coding through IEEE 802.16j. However, it will produce a high deployment cost to provide seamless coverage by the *information raining* infostations along a track. Otherwise, coverage gaps will make the resource allocation challenging. In contrast, our method focuses on making full use of the various existing infrastructures to improve the onboard Internet services.

J. Zhang, et al [15] proposed a multi-system-based access architecture for high-speed trains. The architecture introduces Access Service Gateways (ASGs) in a ground network to support multiple wireless systems. It also deploys Radio Antenna Units (RAUs) along a rail track that serve as a ground-to-train network. However, the proposed architecture will produce a high deployment cost for both the ground network and the ground-to-train network. In contrast, our method does not require the deployment of new infrastructures.

Recently, with the rapid development of 4G and 5G techniques, which may provide higher bandwidth and lower transmission latency at less cost, the cellular-based solutions are generating more interest. MEN-NEMO [7] proposed an LTE femtocell-based scheme that uses multiple interfaces to support seamless mobility for high-speed rail systems. Different from MEN-NEMO, our method does not require any modifications of the existing LTE networks so that it is easy to deploy.

Overall, to achieve multipath transmission in heterogeneous wireless networks for high-speed vehicles, link-layer approaches are infeasible because the various networks in question may belong to different providers. Compared to the link-layer literatures, our scheme works at the network layer so that it can avoid the complex interworking at the link layer and take advantage of a variety of network resources managed by different operators.

B. Network-layer Solutions

Several standardized protocols working at the network layer (or 3.5 Layer) can support multi-homing, such as HIP [16], Shim6 [17] and LISP [18]. These protocols are locator/identifier separation solutions, which enable continuity of communications across IP address changes in mobile scenarios. However, they are not designed for concurrent multipath transmission, nor suitable for fast-moving vehicles. In contrast, we aim at concurrent multipath transmission for high-speed vehicles in heterogeneous networks.

IETF Multiple Interfaces (MIF) working group is now working on the standards [19] for nodes with multiple interfaces. However, it mainly focuses on the conflicting configuration problems associated with different networks, such as DNS server and HTTP proxy, but not the multipath transmission issues.

MAR [20] introduced a wireless multi-homed device similar to our mobile router to exploit wireless diversity. However, different from MAR, our mobile router is not configured as the user devices' default DNS server that may overly complicate the mobile router's design. Moreover, our mobile router does not work as a source-NAT box for user packets, instead, user packets are routed to an access router through logic tunnels. In addition, without designing or specifying any scheduling algorithm, MAR is assumed to rely on custom-built scheduling protocols, while we design a scheduling algorithm adapting to wireless paths.

An IP-in-IP encapsulation method [21] [22] was proposed to provide bandwidth aggregation by splitting a data flow across multiple interfaces at the IP level. However, it simply assumes that the effective bandwidth of a path is constant throughout the lifetime of a TCP connection. It also assumes all the packets sent along a path have the same size. These assumptions do not hold in high-speed mobility scenarios. EDPF [23] also uses IP-in-IP encapsulation to tunnel packets and proposes a bandwidth aggregation scheme for the real-time applications. It schedules the packets based on their estimated delivery time. However, EDPF ignores the wireless propagation delay, instead, they only consider the wireline delay on the paths from the proxy to the base stations, which is not accurate in high-speed mobility scenarios. In contrast, for more accurate scheduling in high-speed mobility scenarios, our algorithm considers the total delay on the paths from the mobile router to the access router, and takes the variable path delays into consideration when showing the properties of our algorithm. Furthermore, our algorithm employs rescheduling mechanisms (also called compensation mechanisms) that automatically adapt to the unexpected changes of link parameters.

Several literatures [24] [25] have shown that the historical tracing data can accurately forecast the wireless network capabilities and is of great value for intelligent scheduling. BreadCrumbs [24] predicts the changes of networks with past observations of wireless network capabilities. It evaluated the efficacy of the forecasts with several weeks of real-world usage, and found that the scheme can provide significantly improved performance. Y Jun, et al [25] found that the past bandwidth information is a good indicator of the actual bandwidth experienced at a given location. They stored past bandwidth data in the form of bandwidth maps and demonstrated the usefulness of these maps with two case studies (i.e., adaptive multimedia and mobile vehicular Internet). Results showed that the use of past location-specific bandwidth knowledge can significantly improve the QoS of multimedia applications in high-speed mobility. However, weighted round robin scheduling is adopted by [25], which may result in the disorder of packets. In addition, no method is designed to deal with unexpected channel fluctuations. In contrast, our scheduling algorithm selects the path that can deliver a packet fastest. Rescheduling mechanisms are also designed to deal with the unexpected channel fluctuations.

To meet with the special requirements in high-speed mobility scenarios, we extend EDPF by adding the parameter of wireless transmission delay into our algorithm. In addition, our algorithm automatically adapts to the path status, in terms of bandwidth and delay, according to both historical tracing databases and active path monitoring. Moreover, a rescheduling mechanism is designed to deal with unexpected channel fluctuations.

C. *Transport-layer Solutions*

Although standardized SCTP [26] [27] supports multi-homing, it does not transmit data over multiple paths simultaneously; instead, it selects a primary data transmission path with enabling transparent fail-over between redundant network paths. Some approaches extend the multi-homing capability of SCTP to aggregating the bandwidth of multiple paths, including WiMP-SCTP [28] and cmpSCTP [29]. However, most of these non-standardized extensions of SCTP require significantly changing the protocol stacks of both endpoints, which makes it difficult for them to be accepted and rapidly deployed. In addition, the existence of various types of middleboxes greatly constrains the deployability of SCTP. In contrast, our method is independent of the protocol stack and transparent to the transport-layer protocols.

Recently, a multipath extension to TCP, named MPTCP [30], has been standardized to transmit data on multiple paths simultaneously. As declared by its architectural guidelines [31], MPTCP needs to meet the goals of improved reliability and throughput. Its performance benefits were verified in [32], and the results show that MPTCP achieves both good throughput and fairness. However, to achieve multipath transmission for high-speed vehicles, MPTCP may not be a suitable solution for the following reasons: 1) Various applications on different user terminals share the multiple vehicle-to-ground paths. Some of the applications, however, are not based on the connection-oriented service provided by MPTCP, such as Domain Name System (DNS) and most voice/video services. Thus, it is unreasonable to select MPTCP as the multipath scheme since all applications will be forced to use a connection-oriented service. 2) The retransmission timer may expire when there is a high path delay. As a result, MTCP usually reduces its congestion window to 1 and enters a slow transmission phase. Consequently, all the packets of all applications will be blocked because they share the vehicle-to-ground paths. In contrast, a network-layer scheme is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2581020, IEEE Transactions on Vehicular Technology

more flexible because its best effort delivery service can fully utilize the limited resources of the vehicle-to-ground paths. 3) In practice though, the existence of various types of middleboxes greatly constrains the deployability of MPTCP [33], especially in mobility scenarios where the middleboxes on the vehicle-to-ground paths vary with the location of a moving vehicle. Thus, we employ a network-layer multipath transmission scheme, which is more flexible for user applications to choose their own transport-layer protocols.

MPTCP throughput is known to decrease significantly under the circumstances of path heterogeneity, especially when there are some bottleneck paths [33] [34]. Thus, several algorithms were proposed recently to improve MPTCP's performance. These algorithms can be roughly divided into two categories: 1) Algorithms closely related to MPTCP, which improve the MPTCP congestion control scheme. 2) More general algorithms, which are not limited to use in transport layer. The first category includes congestion window adaption algorithm [34] and fairness design with congestion balancing [35]. They try to solve the throughput decrease problem of MPTCP, but they are not aimed at the high-speed mobility scenario. They will not be further discussed because MPTCP itself is not selected to be the vehicle-to-ground solution as discussed before. The second category includes Round Robin (RR) scheduling algorithm and its variants, such as Load Sharing for SCTP (LS-SCTP) [36] and Delay-Aware Packet Scheduling (DAPS) [37]. LS-SCTP [36] supports weighted round robin and assigns data to each path according to the ratio $cwnd/RTT$ (congestion window/round trip time). DAPS [37] distributes data to two different paths depending on the ratio RTT_{slow}/RTT_{fast} and $cwnd$. However, an ideal ratio is hard to achieve because the ratio is limited by the $cwnd$ size. No packet can be delivered on a path if the $cwnd$ is full, even if the path may deliver the subsequent packet earliest to the destination. Moreover, large variation of end-to-end delay will make LS-SCTP and DAPS fragile. In contrast, we employ a network-layer scheduling algorithm that can adjust the size of sending buffers as needed. Moreover, we design a rescheduling mechanism to deal with unexpected delay fluctuations.

III. Measurements of 3G/4G Wireless Network Characteristics in High-Speed Scenario

Fig. 1 shows the experimental environment. Two types of our mobile routers are shown in Fig. 2. We measured the network characteristics of 3G and 4G technologies by establishing communications between the clients, the mobile router, and the server [38].

In our experiments shown in Fig. 1, WiFi was used inside of a train for data transmission between the user devices and the mobile router, while 3G/4G technologies were used for vehicle-to-ground communications.

WiFi communication inside of the train may suffer from performance degradation when the WiFi link is shared by multiple users. This problem is mainly caused by the collision avoidance scheme of WiFi, but it is beyond the scope of this paper since we focus on improving vehicle-to-ground

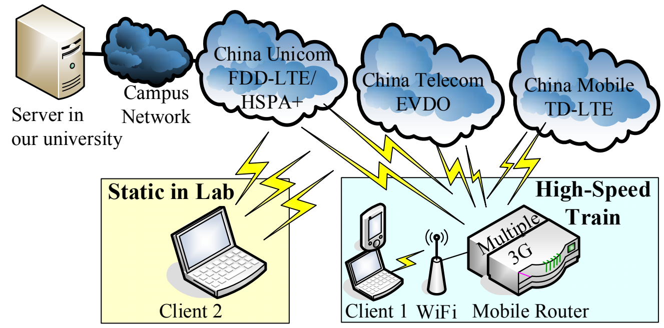

Fig. 1. Experimental topology

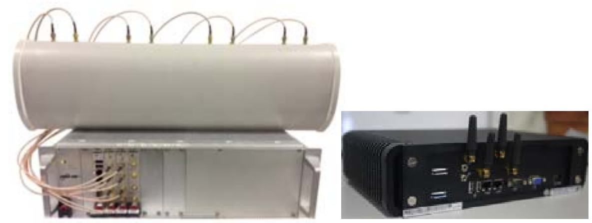

Fig. 2. Mobile routers

communication by making full use of various existing infrastructures.

As a candidate for vehicle-to-ground communication, WiFi can provide high bandwidth. However, WiFi is typically used as gap fillers due to the handover issues and high deployment costs [6]. In practice, WiFi is more suitable to be deployed at the stations where vehicles slow down, rather than the roadside where vehicles are under high-speed mobility. Our mobile router has both external WiFi and 3G/4G interfaces for vehicle-to-ground communication, and can utilize all these interfaces simultaneously based on the proposed method when a train is near a station. However, no WiFi AP could be accessed by the external WiFi interfaces when the train was under high-speed mobility. Thus, only 3G/4G network characteristics were compared between stationary and high-speed status, as shown in the following.

There are three major mobile network providers in China, i.e., China Telecom, China Unicom and China Mobile. Today, EV-DO (3G), FDD-LTE/HSPA+ (4G/3G) and TD-LTE (4G) are the main mobile technologies adopted by their networks, respectively. We evaluated all these technologies in our experiments.

The mobile router, in which plugged multiple MiniPCI-E 3G/4G cards, was placed on a high-speed train and could visit the Internet through multiple networks simultaneously. In particular, the mobile router installed Ubuntu 12.04 LTS with Linux kernel 3.6.0. WvDial was used to make 3G/4G connections to the Internet. All 3G/4G cards were at their default configurations because we aimed at making full use of the existing infrastructures. Iptraf was used to monitor the network parameters. For producing historical tracing databases, an extension to Iptraf was programmed to support MySQL.

Note that for China Unicom, the card may change its mode between FDD-LTE and HSPA+ automatically due to the incomplete coverage of FDD-LTE. Each interface was allocated IP addresses dynamically when the train moved along

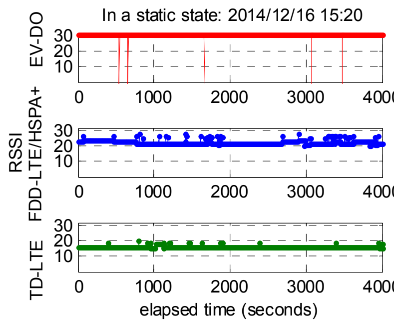

Fig. 3. RSSI in a static state

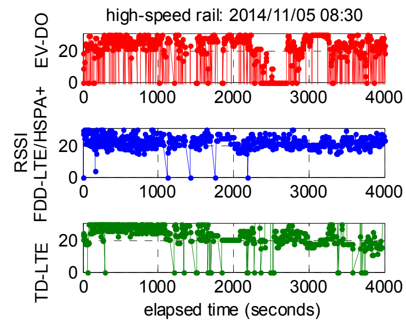

Fig. 4. RSSI on a high-speed train

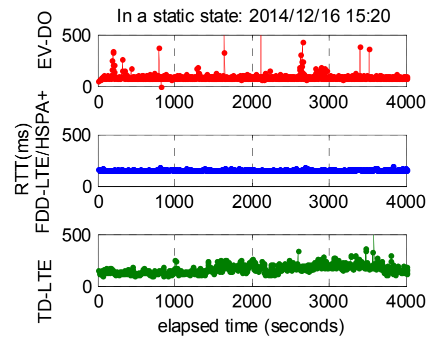

Fig. 5. RTT in a static state

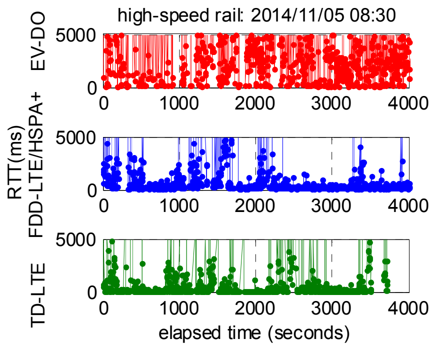

Fig. 6. RTT on a high-speed train

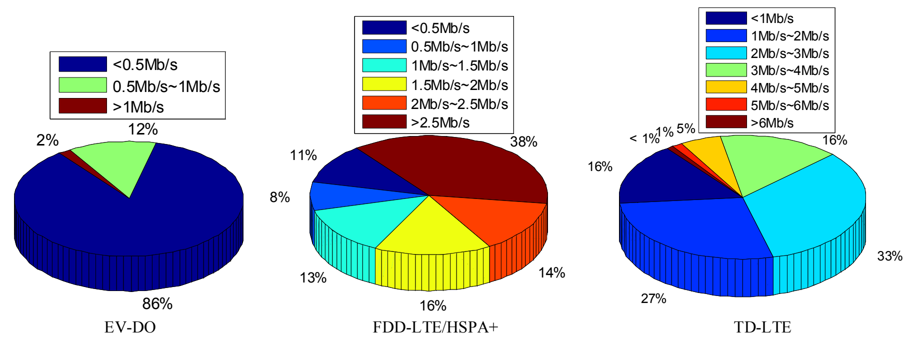

Fig. 7. Bandwidth distribution on a high-speed train

the rail. The server was placed in the Information Center of Beijing Jiaotong University and could visit the Internet through the campus network.

During the experiments, we recorded the Received Signal Strength Indicator (RSSI), Round-Trip Time (RTT) and bandwidth of different 3G/4G networks. For comparison, we performed the experiments both in our laboratory in a static state and on a high-speed train with the speed of 300km/h.

The RSSIs were obtained by sending AT commands to the 3G/4G cards plugged into the mobile router and clients. For EV-DO, FDD-LTE/HSPA+ and TD-LTE, the AT commands returned values between 0 and 31, which is a relative quality of the received signal and it is unique to each cellular technology. For the 3G/4G cards in use, the power level in dBm is calculated by $(\text{rssi} \times 2 - 113)$. For example, 10 corresponds to -93dBm. We use these values to evaluate the stability of each technology under different statuses. From Fig. 3, we can observe that the RSSIs of all the 3G/4G networks keep stable at a relatively high level in a static state. However, as shown in Fig. 4, the RSSIs fluctuate remarkably and can even be 0 at times. Fortunately, we can also observe that different 3G/4G networks can complement each other well at some geographic coordinates.

RTT was obtained by a modified "ping" program. We modified the source codes by adding a timestamp in ICMP messages, and recorded the RTT in a MySQL database every 5 seconds. Fig. 5 shows the RTT between the static client and the server. The RTTs are relatively low (around 100ms~200ms) and stable for all the networks in a static state, while from Fig. 6 we can conclude that in the high-speed environment, the RTTs become much larger and unstable. Bandwidth was evaluated using the tool iPerf [39], which is developed for active

TABLE I. COMPARISONS BETWEEN STATIC AND HIGH-SPEED EVIRONMENTS FOR EV-DO, FDD-LTE/HSPA+ AND TD-LTE

Parameters	3G/4G	High-Speed Train	Static
Average RSSI	EV-DO	17.7	30.8
	FDD-LTE/HSPA+	15.6	22.6
	TD-LTE	18.3	22.7
Average RTT	EV-DO	1255ms	90ms
	FDD-LTE/HSPA+	763ms	155ms
	TD-LTE	1163ms	151ms
Average Bandwidth	EV-DO	127kbps	1.5Mbps
	FDD-LTE/HSPA+	1.8Mbps	12Mbps
	TD-LTE	2.1Mbps	15Mbps

measurements of the maximum achievable bandwidth on IP networks. Fig. 7 shows the bandwidth distribution of EV-DO, FDD-LTE/HSPA+ and TD-LTE obtained on the train.

Table I shows the average RSSI, RTT, and bandwidth of EV-DO, FDD-LTE/HSPA+ and TD-LTE in the experiments. Although the results observed along different trails may differ from each other due to the different network coverage and geographical environments, our results reflect the highly variability of wireless links in highly mobile scenarios.

Several literatures [24] [25] have shown that the historical tracing data can forecast the wireless network capabilities and is of great value for intelligent scheduling. We evaluated the efficacy of these forecasts. Fig. 8 shows the bandwidth on two trips. We can find a similar conclusion as [25], i.e., the past bandwidth information is a good indicator of the actual bandwidth experienced at a given location. For example, there is a sharp bandwidth decrease from location 8 to location 9. Thus, to forecast the bandwidth at location 9 on trip 2, the result will be inaccurate if the bandwidth observation at location 8 is used. In contrast, the result will be more accurate if the bandwidth observation at location 9 on trip 1 is used, because the two bandwidth observations at location 9 on the two trips

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2581020, IEEE Transactions on Vehicular Technology

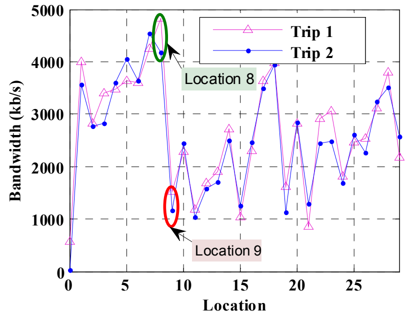

Fig. 8. Past versus present bandwidth

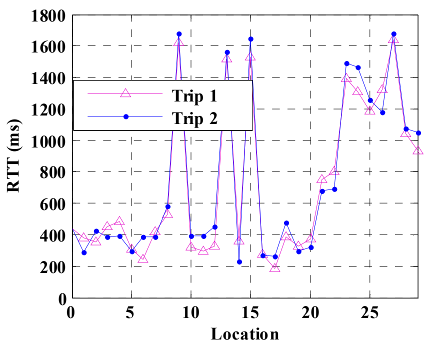

Fig. 9. Past versus present RTT

are much closer. The RTTs on two trips illustrated in Fig. 9 show a similar trend.

To conclude, there are two key observations to guide us in the design of a multipath method for vehicles in high-speed mobility scenarios:

First, compared to stationary vehicles, vehicles moving at a high speed face a more challenging wireless environment. The wireless characteristics, in terms of RSSI, RTT, and bandwidth, vary widely with location. The network parameters experienced at a location cannot accurately indicate those of the next location. Thus, most scheduling algorithms based on link probe will be fragile.

Second, the fixed route of a vehicle gives us an opportunity to predict the network characteristics based on historical data, i.e., the past trip tells us more than the previous location on the current trip [24] [25]. Although it is impossible to predict the network characteristics without error by the historical tracing databases, the forecast by this means is of great help for an efficient scheduling.

Thus, we design our multipath method, as well as our scheduling algorithm, based on the two observations. Furthermore, rescheduling mechanisms (also called compensation mechanisms) are designed to deal with unexpected channel fluctuations.

IV. Network Adaptive Multipath Transmission

In this section, we first present an overview of the proposed multipath transmission architecture for high-speed vehicles in heterogeneous wireless networks. Then, a network adaptive scheduling algorithm is introduced to meet the requirements in terms of aggregating bandwidth and avoiding the disorder of packets.

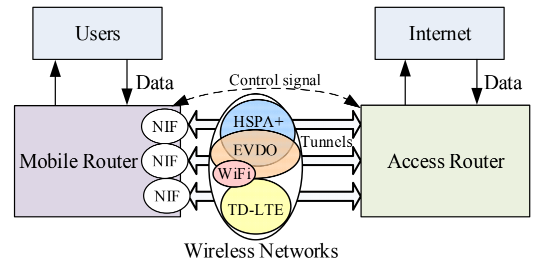

Fig. 10. Overview of the concurrent multipath transmission scheme

A. *Multipath Architecture with Logic Tunnels*

Fig. 10 shows an overview of the multipath transmission architecture. It is proposed to provide more stable Internet services for the onboard users by multiple available wireless technologies, without requiring the participation of the user devices in any multipath signaling.

In the architecture, one or more mobile routers are deployed on a high-speed vehicle. Moreover, a mobile router can connect to the Internet by accessing multiple wireless networks that are available along a track, including WiFi, 3G, 4G, and satellite. The mobile router then provides transparent Internet services for onboard users.

An access router is a device located on the ground. It visits the Internet via wired links, and maintains logic tunnels with multiple mobile routers.

The logic tunnels, which are established between a mobile router and an access router, provide transparent concurrent multipath data transmission for onboard users.

At each available interface of a mobile router, a logic tunnel is established. The number of available interfaces may change with the moving of the vehicle. The number of logic tunnels will change correspondingly. A logic tunnel at an interface will be removed if the interface becomes inactive. Meanwhile, a new tunnel will be established if an interface becomes active, e.g., getting a new IP address.

The logic tunnels are independent of each other. Each tunnel simply forwards the packets in its sending queue. The proposed scheduling algorithm is responsible for putting packets into the sending queues of all logic tunnels.

The proposed scheduling algorithm receives a user packet at the IP layer when the packet arrives at the mobile router. Then the algorithm puts the IP packet into the sending queue of the selected tunnel. Afterwards, the tunnel encapsulates the packet and forwards it to the access router. At the access router, the encapsulated packet is firstly put into the receiving buffer for reordering. Then, the packet is decapsulated, and the original IP packet is returned to the IP layer of the protocol stack. After that, the protocol stack forwards the original IP packet to its destination.

In the architecture, scheduling is a key method to improve the network efficiency. For the onboard applications, especially those based on connection-oriented protocols (e.g., TCP), the scheduling algorithm not only improves the total available bandwidth by aggregating the bandwidth of multiple interfaces, but also avoids unnecessary packet reordering caused by dynamic path characteristics.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2581020, IEEE Transactions on Vehicular Technology

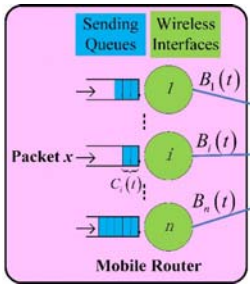

Fig. 11. Simplified view of the multipath network between the mobile router and the access router in heterogeneous wireless networks

TABLE II. SUMMARY OF IMPORTANT SYMBOLS

Symbols	Meaning of the symbols
i	A path between a wireless interface of a mobile router and a wired interface of an access router
$D_i(t)$	The one-way delay of path i at time t
$B_i(t)$	The available bandwidth of path i at time t
$C_i(t)$	Queueing time on path i. Over a period of time $C_i(t)$ after t, the sending queue on path i will be empty
$Q_i(t)$	Total size of packets in the sending queue on path i at time t
a^x	The arrival time of the x^{th} packet at the mobile router
b_i^x	The arrival time of packet x at the access router on path i
c^x	The leaving time of the x^{th} packet from the access router
$t_{q\_i}^x(t)$	The duration that packet x waits in the sending queue on path i when arriving at t
$t_{s\_i}^x(t)$	Packet sending duration at path i when the sending begun at t
$t_{d\_i}^x(t)$	One-way transmission delay at t
$w_i(t)$	The normalize average bandwidth of path i.
S^x	The size of the x^{th} packet arriving at the mobile router
O^x	The buffering time of the x^{th} packet at the access router
D_{max}	The maximal acceptable delay of the paths
B_{max}	The highest bandwidth of the paths
B_{min}	The lowest bandwidth of the paths
S^{max}	The maximum packet size

Next, we introduce a scheduling algorithm that meets our requirements.

B. Network Adaptive Multipath Scheduling Algorithm

To provide transparent multipath Internet services for the users on board, many challenges exist. A major one is how to schedule packets reasonably on multiple unstable paths, as discussed in section III.

If the packets belonging to the same flow are scheduled on different paths, a high packet out-of-order is likely to occur without a good scheduling algorithm. EDPF[23] can guarantee the packets arrive in order when the packets are of the same size. However, it does not take into account the delay on wireless links.

Our algorithm is a variation of EDPF. We extend it by adding the wireless transmission delay into the algorithm. Furthermore, based on our practical experiments on the high-speed train, we got the tracing databases about the delay and bandwidth of various wireless networks along a specific trail. Depending on the recorded network conditions, we can divide the database into mini-timeslots during which the delay and bandwidth remain fairly constant. We also monitor the wireless link conditions and perform rescheduling in case of unexpected changes of path parameters, in terms of bandwidth and delay.

The fundamental thought of our algorithm is to (1) take account of the queueing time at the mobile router, bandwidth of the interfaces, and delays of the paths between the mobile router and the access router, (2) based on the tracing databases and the monitoring of path parameters, select the path that takes the shortest time to deliver a packet to the access router, and (3) reschedule the packets in sending queues once a changing of path conditions is detected.

Algorithm 1: Network adaptive multipath scheduling algorithm

Input: $D_i(t)$, $B_i(t)$, $Q_i(t)$, $C_i(t)$, a^x, S^x
Output: forwarding path r for each packet x;
1 : initialize $Q_i(0)$=0, a^1=0;
2 : update $D_i(t)$ and $B_i(t)$ per timeslot;
3 :
4 : **while** (a new packet x arrives) **do**
5 : r calculation;
6 : packet x is put in the sending queue on path r, and waits to be forwarded;
7 : **end** while
8 :
9 : **for** any path i
10 : monitoring the path parameters $D_i^{mo}(t)$, $B_i^{mo}(t)$ periodically;
11 : **if** $((|\Delta D = D_i^{mo}(t) - D_i(t)| > D_{THR})$ or $(|\Delta B = B_i^{mo}(t) - B_i(t)| > B_{THR}))$
12 : $D_i(t) = D_i(t) + \Delta D$; $B_i(t) = B_i(t) + \Delta B$;
13 : **for** any packet y in the sending queues
14 : r re-calculation;
15 : packet y is put in the sending queue on path r, and waits to be forwarded;
16 : **end** for
17 : **end** if
18 : **end** for

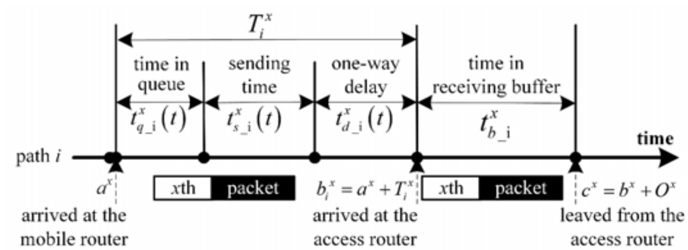

Fig. 12. Distribution of the time points

The network between the mobile router and the access router can be simplified as shown in Fig. 11. The meaning of symbols can be found in TABLE II. The network adaptive multipath scheduling algorithm is detailed in Algorithm 1. The r calculations in step 5 and step 14 are shown in Eq. (1).

Phase 1: Path selection at the arrival of a packet

As depicted from step 4 to step 7, Algorithm 1 selects the forwarding path r for packet x at its arrival. In step 5,

$$r = \arg\min_i b_i^x, (1 \le i \le n) \quad (1)$$

where n is the number of available paths. b_i^x is the arrival time of packet x at the access router on path i.

To make it clear, the transmission procedure of the x^{th} packet is illustrated in Fig. 12. Let a^x be the arrival time of packet x at the mobile router. Then, the estimated arrival time of packet x at the access router on path i is evaluated as $b_i^x = a^x + T_i^x$, where

$$T_i^x = t_{q\_i}^x(t) + t_{s\_i}^x(t) + t_{d\_i}^x(t)$$

where $t_{q\_i}^x(t)$ is the duration that packet x waits in the sending queue on path i. In other words, the forwarding of the x^{th} packet begins at $a^x + t_{q\_i}^x(t)$. $t_{s\_i}^x(t)$ is the duration of the packet sending time on path i, i.e., $t_{s\_i}^{x}(t) = S^{x}/B_i(t)$. $t_{d\_i}^{x}(t)$ is the one-way transmission delay from the mobile router to the access router at time t, i.e., $t_{d\_i}^{x}(t) = D_i(t)$.

In step 5, the values of parameter t for calculating $t_{q\_i}^{x}(t)$, $t_{s\_i}^{x}(t)$, and $t_{d\_i}^{x}(t)$ are different from each other. In addition, $D_i(t)$ and $B_i(t)$ are updated per timeslot along with the moving of a vehicle in step 2. Since the values of $D_i(t)$ and $B_i(t)$ are the dynamic path parameters stored in the tracing databases for a fixed track, r calculation in step 5 will automatically choose the appropriate values according to the speed and location of the vehicle. A simple example is that, assuming a packet x_1 arrived at a mobile router at t_1, if its waiting time in queue is $t_{q\_i}^{x_1}(t_1)$, then, its sending duration is $t_{s\_i}^{x_1}(t_2) = S^{x_1}/B_i(t_2)$, where t_2 is the sending start time, $t_2 = t_1 + t_{q\_i}^{x_1}(t_1)$. After that, its one-way delay $t_{d\_i}^{x_1}(t_3) = D_i(t_3)$, where $t_3 = t_2 + t_{s\_i}^{x_1}(t_2)$. Thus, the packet's arrival time at the access router would be estimated as $b_i^{x_1} = t_1 + t_{q\_i}^{x_1}(t_1) + S^{x_1}/B_i(t_2) + D_i(t_3)$. This is reasonable because $B_i(t_1)$ may not be equal to $B_i(t_2)$, and $D_i(t_1)$ may not be equal to $D_i(t_3)$.

However, existing research [23] [37] estimate $B_i(t_2)$ and $D_i(t_3)$ by $B_i(t_1)$ and $D_i(t_1)$. As shown in [25] and our observations in Section III, these estimations are inaccurate, especially in high-speed mobile scenarios where $B_i(t)$ and $D_i(t)$ change over time. Our solution is to use the historical trace data. The reasonability of using history to predict the future has been proved by [24] and [25] using real-world usage. Furthermore, we also show its validity by our experiments. This is one of the significant differences between our algorithm and EDPF [23], which does not consider a variable t when calculating the delivery time of a packet.

After making a decision, the algorithm updates $t_{q\_i}^{x}(t)$ to $t_{q\_i}^{x}(t) + t_{s\_i}^{x}(t)$, i.e., the $(x+1)^{\text{th}}$ forwarding at path r can begin only after the x^{th} packet has been sent out.

Phase 2: Rescheduling when link status drifts away

There is a possibility that the estimated path parameters will drift away from the true values. In Phase 1 of the proposed algorithm, the path estimations are based on the tracing databases. If the estimations are inaccurate, the selected path r might not be the appropriate one to deliver the packet. Hence, it is reasonable to require some compensation methods to deal with the parameter drift.

There are two compensation methods dealing with the inaccurate estimation. They are as follows:

(1) The receiving buffer in an access router can reorder the out-of-order packets caused by random fluctuations of path parameters.

(2) The rescheduling mechanism can deal with the significant changes in link capabilities. For instance, new base stations could lead to a significant bandwidth growth and popular social events could lead to a longer path delay due to network congestion. To deal with unexpected parameter fluctuations, the rescheduling mechanism periodically monitors the current parameters $D_i^{mo}(t)$ and $B_i^{mo}(t)$ in step 10, and compares current parameters with historical parameters in step 11. Rescheduling will be performed if the forecast of $D_i(t)$ or $B_i(t)$ is obviously incorrect, i.e., ΔD and ΔB will be compensated to $D_i(t)$ and $B_i(t)$, respectively, in step 12.

After that, r re-calculation from step 13 to step 16 will be performed in order to deliver the delayed packets in queues earliest to the access router.

Note that the rescheduling is performed only when ΔD or ΔB exceeds a specified threshold D_{THR} or B_{THR}. Otherwise, the receiving buffer in the access router can work as the compensation method to reorder packets.

Due to the reason that the onboard users usually initiate the communications, such as sending an email or browsing websites, our explanation, so far, focuses only on the transmission from the mobile router to the access router. It is also fit for the case that the roles of the mobile router and the access router are interchanged.

V. Properties Analysis

In the subsequent sections, the effectiveness of the algorithm is analyzed, in terms of the length of sending queues, the size of receiving buffer for reordering packets, and the ability of bandwidth aggregation.

A. Difference between Length of Sending Queues

If the paths are of constant bandwidth and delay and if the packets are of the same size, we can easily derive from our algorithm that all the packets can arrive at the access router in order.

In contrast, if the paths are of variable bandwidth and delay, and (or) the packets are of variable size, it is crucial for the algorithm to schedule the packets among the paths appropriately. For any two paths i and j, we can say the algorithm performs a good scheduling if the maximum difference between the lengths of their sending queues $Q_i(t)$ and $Q_j(t)$ is not the function of the total number of packets.

In this analysis, we further define D_{max} as the maximal acceptable delay of the paths. If $D_i(t) > D_{max}$, we regard the path i as unavailable. No new packets will be scheduled on path i if $D_i(t) > D_{max}$ since path i is not suitable to transmit packets at this moment. We define B_{max} and B_{min} as the highest and lowest bandwidth of the paths, respectively. Since the bandwidth of any of the wireless links fluctuates in high-speed scenarios, we regard the path i as unavailable if $B_i(t) < B_{min}$. We define S^{max} as the largest packet size.

In the following, we first present a lemma that is helpful to get the properties of the algorithm. We just present the lemma here for better readability, while the detailed proof can be found in Appendix A.

Lemma 1. For any two paths i, j and any time t, $C_i(t)$ and $C_j(t)$ are the queueing time on the two paths, if $C_i(t) \le C_j(t)$, then, $C_j(t) - C_i(t) \le S^{max}/B_{min} + D_{max}$.

Next, we will show that the difference between the normalized lengths of sending queues is not related to the number of packets M, whereas it is a function of M for Round Robin scheduling algorithm and its variants.

Let n be the number of available paths, for any two paths i and j, the relation between normalized $Q_i(t)$ and normalized

$Q_j(t)$ is $Q_j(t)/w_j(t) - Q_i(t)/w_i(t) \le S^{max} + B_{min}D_{max}$, where $w_j(t) = B_j^{ave}\left(t, t + C_j(t)\right)/B_{min}$ is the normalize average bandwidth of link j between time t and $\left(t + C_j(t)\right)$.

Proof. From Lemma 1, we learn that $C_j(t) - C_i(t)$ cannot exceed $S^{max}/B_{min} + D_{max}$, i.e., $Q_j(t)/B_j^{ave}\left(t, t + C_j(t)\right) - Q_i(t)/B_i^{ave}\left(t, t + C_i(t)\right) \le S^{max}/B_{min} + D_{max}$.

Thus,

$$\left(Q_j(t)B_{min}\right)/B_j^{ave}\left(t, t + C_j(t)\right) - \left(Q_i(t)B_{min}\right)/B_i^{ave}\left(t, t + C_i(t)\right)$$
$$= Q_j(t)/w_j(t) - Q_i(t)/w_i(t) \le S^{max} + B_{min}D_{max}$$

In the following, we analyze the behavior of the scheduling schemes based on Round Robin.

Assuming there are n available paths $0, 1, \cdots, n-1$. At any time t, the total number of packets for transmission is M. Thus, any packet x, $x \in \{M \mid mod(M,n) == i\}$, will be scheduled on path i. The total size of packets in the sending queue on path i at time a^P is

$$Q_i(a^P) = \left(\sum_{x \in \{M \mid mod(M,n)==i\}} S^x\right) - a^P B_i^{ave}(0, a^P)$$

where a^P is the arrival time of packet P, and S^x is the size of packet x.

Then, we can conclude that, for any two paths i and j, $Q_j(a^P)/w_j(a^P) - Q_i(a^P)/w_i(a^P)$ is a function of M, and can grow without bound. To take a simple example, if the sizes of packets scheduled on two paths alternate between 1000bytes and 100bytes, while the two paths have the same bandwidth, the difference between the lengths of the two sending queues grows with the number of packets. For Weighted Round Robin, the variable packet size, variable delay, and variable bandwidth will make the difference between the queue length grow without bound. We evaluate the properties of the algorithms in Section VI through simulations.

B. *The Size of Receiving Buffer for Reordering Packets*

When a receiving buffer for reordering packets is used at the access router, let b^x be the arrival time of the x^{th} packet at the access router, and c^x be the leaving time of the x^{th} packet from the access router. Then, as shown in Fig. 12, $c^x = b^x + O^x$, where O^x is the buffering time of packet x at the access router.

For our algorithm, the buffer size needed at the access router for sending the packets in order out of the access router is at most $\sum_{i=1}^{n}\left(w_i(t)S^{max} + 2B_i^{ave}\left(t, t + \tau(m,i)\right)D_{max}\right)$.

Proof. At any time t, for any packet x in the sending queues at the mobile router, let the maximum delivery time to the access router be $b_m^y = max\{b^x\}$, where y is the last packet being delivered and the delivery path is m. If any packet arrived at the mobile router after t was delivered before b_m^y, it would need to be buffered at the access router. A packet that arrives after packet y may only be delivered before b_m^y on any path i, $i \ne m$. Let

$$\tau(m,i) = \left(C_m(a^y) + D_m\left(a^y + C_m(a^y)\right)\right) - \left(C_i(a^x) + D_i\left(a^x + C_i(a^x)\right)\right)$$

where a^x is the arrival time of the last packet in the queue on path i. Thus, on path i, a limited bits not exceeding $B_i^{ave}\left(t, t + \tau(m,i)\right)\tau(m,i)$ may be delivered before b_m^y. From Lemma 1,

$$\tau(m,i) \le S^{max}/B_{min} + D_{max} + D_m\left(a^y + C_m(a^y)\right) - D_i\left(a^x + C_i(a^x)\right) \le S^{max}/B_{min} + 2D_{max}$$

Thus, the total buffer size needed is

$$\sum_{i \ne m} B_i^{ave}\left(t, t + \tau(m,i)\right)\tau(m,i)$$
$$\le \sum_{i \ne m} B_i^{ave}\left(t, t + \tau(m,i)\right)\left(S^{max}/B_{min} + 2D_{max}\right)$$
$$= \sum_{i \ne m}\left(w_i(t)S^{max} + 2B_i^{ave}\left(t, t + \tau(m,i)\right)D_{max}\right)$$
$$\le \sum_{i=1}^{n}\left(w_i(t)S^{max} + 2B_i^{ave}\left(t, t + \tau(m,i)\right)D_{max}\right)$$

We can see that the buffer size for reordering packets is also not related to the total number of packets M, while Round Robin does not hold this property.

C. *The Ability to Aggregate Bandwidth*

A single link with the aggregate bandwidth $\sum_{i=1}^{n} B_i(t)$ is called Aggregated Single Link (ASL) if the available bandwidth of path i is $B_i(t)$. A scheduling algorithm would be better if its performance were closer to ASL. In this section, we will show that, when the delays of wireless links are in consideration and rescheduling is performed, the difference between total throughput served by the proposed algorithm and ASL in $(0, t)$ is upper bounded by $(S^{max} + B_{min}D_{max})\sum_{j=1}^{n} w_j(t)$ for any time t.

Proof. Assuming there are n available paths, for any path i the available bandwidth is $B_i(t)$. Assuming ASL finishes sending all the packets at time t, there are two possible cases:

First, one or more paths are idle at t. In this case, our algorithm gets the maximum difference with ASL if only one path l is idle. At the same time, path l should be with the lowest bandwidth among all the paths. From Lemma 1, we learn that $\left|C_j(t) - C_l(t)\right|$ cannot exceeds $S^{max}/B_{min} + D_{max}$ for any path $j, j \ne l$.

Thus, the upper bound is:

$$\sum_{j \ne l}\left[B_j^{ave}\left(t, t + C_j(t)\right)\left|C_j(t) - C_l(t)\right|\right]$$
$$\le \left(S^{max}/B_{min} + D_{max}\right)\sum_{j \ne l} B_j^{ave}\left(t, t + C_j(t)\right)$$
$$= \left(S^{max} + B_{min}D_{max}\right)\sum_{j \ne l}\left(B_j^{ave}\left(t, t + C_j(t)\right)/B_{min}\right)$$
$$\le \left(S^{max} + B_{min}D_{max}\right)\sum_{j=1}^{n}\left[B_j^{ave}\left(t, t + C_j(t)\right)/B_{min}\right]$$
$$= \left(S^{max} + B_{min}D_{max}\right)\sum_{j=1}^{n} w_j(t)$$

Second, no path is idle at t. In this case, let τ be the time when all the paths become active. After τ, all the paths are busy forwarding data packets. Thus, the difference of total bandwidth between the algorithm and ASL is derived from the period $(0, \tau)$. And, the first case bounds the difference with ASL at τ by $\left(S^{max} + B_{min}D_{max}\right)\sum_{j=1}^{n} w_j(t)$.

We can conclude that this upper bound is still not related to the total number of packets, whereas the scheduling algorithms based on Round Robin is.

VI. Simulation and Evaluation

In this section, we implement the multipath transmission scheme, and assess the performance advantages the proposed scheduling algorithm has by comparing with Aggregated Single Link, EDPF, and Weighted Round Robin scheduling algorithms.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2581020, IEEE Transactions on Vehicular Technology

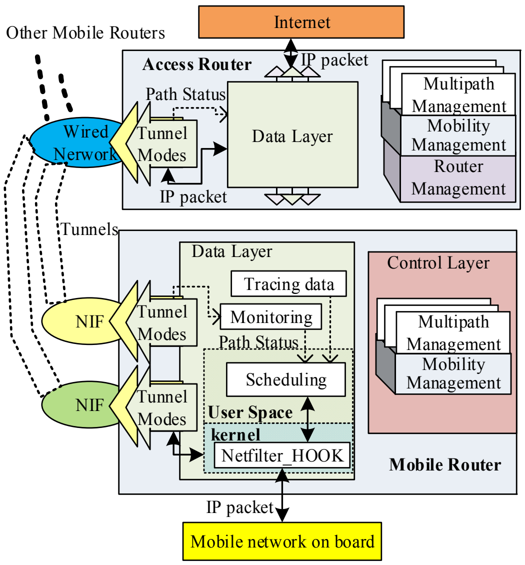

Fig. 13. Functional modules

A. Control Layer Implementation

Fig. 13 introduces our implementation. In the control layer, the mobility management module maintains the available IP addresses allocated by multiple wireless networks. In addition, a mobile router will inform the access router of the new assigned addresses whenever a handover occurs. When a PR-SCTP [27] tunnel is adopted, the access router can be informed of the new addresses by SCTP messages; while when a UDP or IP-in-IP tunnel is adopted, Mobile IP or a self-define message can be used to notify the access router about the changing of addresses.

The multipath management module is responsible for establishing logic tunnels on wireless links. It selects and changes the modes of logic tunnels according to different link conditions. It also changes the parameters of the logic tunnels, e.g., lifetime parameter for PR-SCTP, according to the path status. Four kinds of tunnel modes are supported: (a) IP-in-IP Tunnel, which is the simplest and commonest method to establish the logic tunnels. (b) PR-SCTP Tunnel, which can provide partial transmission reliability for data on an unstable wireless link. (c) UDP Tunnel, which can maximize the transmission of data without considering the loss of packets. When UDP tunnels are applied, the upper layer applications are responsible for the reliability of data. The advantage of a UDP tunnel is that it can maximize the survival of the tunnel in the atrocious wireless environments while providing some flexibilities that IP-in-IP Tunnel cannot achieve. (d) Mixed Mode, which establishes PR-SCTP tunnels on some links and UDP tunnels on the others so that the two kinds of tunnels can play to their strengths.

B. Data Layer Implementation

In the data layer, our scheduling algorithm delivers packets to different tunnels. Then, the packets are encapsulated in the logical tunnels and sent to the access router.

Although several modes of logic tunnel can be chosen, our scheduling algorithm works at the IP layer and is transparent for the logic tunnels.

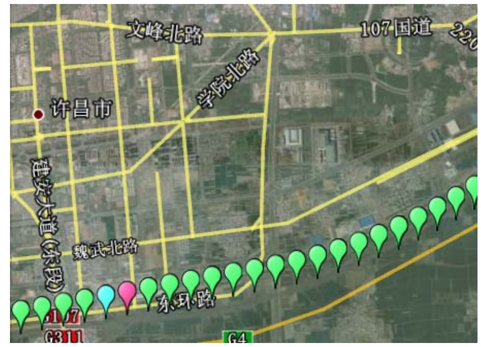

Fig. 14. Bandwidth showed by google earth

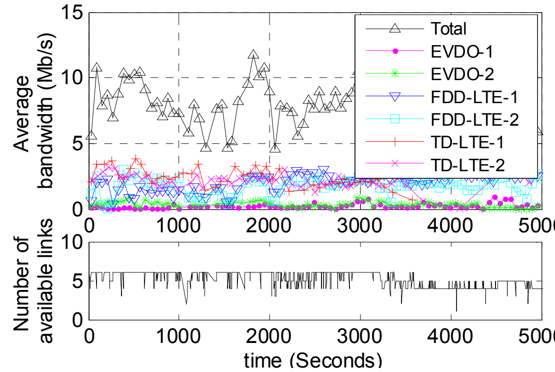

Fig. 15. Bandwidth aggregation on multiple paths

As shown in Fig. 13, the implementation of the proposed scheduling algorithm can be divided into two modules, one for the kernel processing functions, the other for the user space functions. In particular, netfilter queue is used in the Linux kernel to intercept IP packets by registering a HOOK function. Then, the scheduling module in the user space gets the IP packets, makes decisions according to the path status, and distributes the IP packets to the selected tunnels. Thus, our scheduling scheme is transparent to the protocol stack and easy to deploy.

C. Experiments on a High-Speed Train

We performed some experiments on a high-speed train to evaluate the proposed multipath scheme. In the experiments, the data was generated by real applications. All the data was captured and stored in a MySql database. In the experiment, two EV-DO cards, two FDD-LTE/HSPA+ cards and two TD-LTE cards were installed in a mobile router. The logic tunnels aggregated the available bandwidth of the six wireless links and provided the total bandwidth to the onboard devices.

Fig. 14 illustrates the bandwidth by google earth. The green point, the blue point and pink point represent the bandwidth over 5Mbps, between (2Mbps, 5Mbps), and between (0.1Mbps, 1Mbps), respectively.

The bandwidth of each of the six links and the total bandwidth are shown in Fig. 15. We can observe the number of available paths and the available bandwidth on each path at different times. We can conclude that our concurrent multipath transmission scheme works well in a practice environment, and can make full use of various existing infrastructures.

Further comparisons with other scheduling algorithms are presented in the next subsection.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2581020, IEEE Transactions on Vehicular Technology

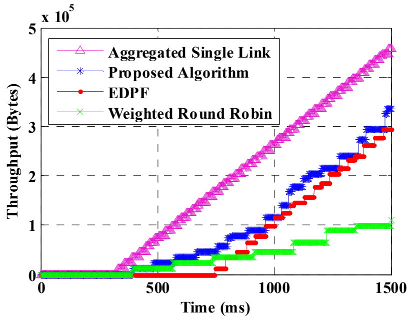

Fig. 16. The ability to aggregate bandwidth: receiving buffer size: 15000 bytes

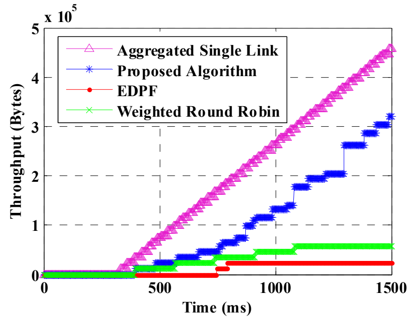

Fig. 17. The ability to aggregate bandwidth: receiving buffer size: 9000 bytes

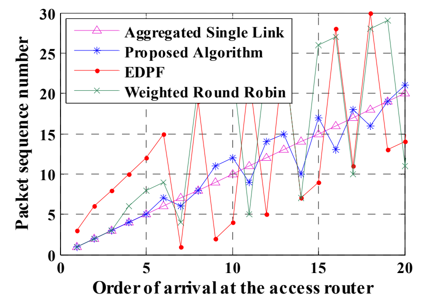

Fig. 18. Packet order of arrival at the access router for different algorithms

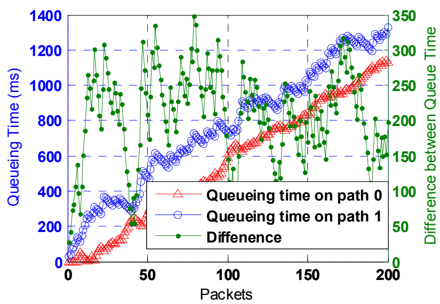

Fig. 19. Queueing times and their difference of our algorithm

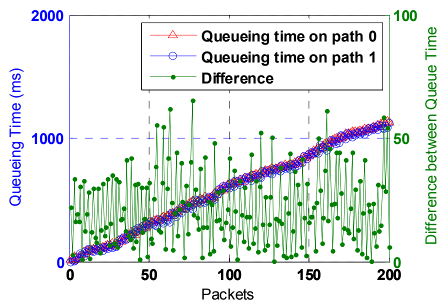

Fig. 20. Queueing times and their difference of EDPF

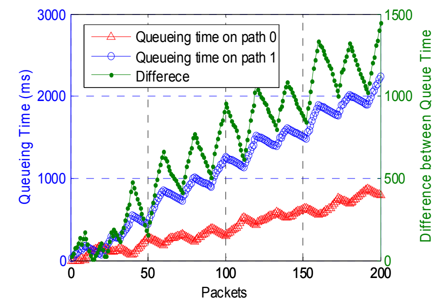

Fig. 21. Queueing times and their difference of WRR

D. *Simulations*

We evaluated the performance of the proposed algorithm to show its advantages. Some simulations were performed in Matlab to compare with Aggregated Single Link, EDPF, and Weighted Round Robin, in terms of the ability of bandwidth aggregation, the queueing time, and the length of sending queues.

The simulating network topology comprised a mobile router and an access router. Two disjointed paths were established between the two devices. To make the simulations more practical, real tracing databases were used as the bandwidth and delay parameters of the two paths. Hence, the bandwidth and delay varied over time just as the mobile router was under a high-speed mobility. For path 0, the bandwidth and delay were roughly in the range of 0.9-4Mb/s and 200-1500ms, respectively. For path 1, the bandwidth and delay were roughly in the range of 0.5-1.8Mb/s and 80-1300ms, respectively. The size of packets was randomly between 1000bytes and 1500bytes.

(a) The ability to aggregate bandwidth

First, the ability to aggregate bandwidth was evaluated. Aggregated Single Link is an ideal single link with the aggregated bandwidth $\sum_{i=1}^{n} B_i(t)$ where $B_i(t)$ is the bandwidth of path i. It represents the upper bound of aggregated bandwidth. The efficiency of a scheduling algorithm can be shown by comparing with Aggregated Single Link. For Weighted Round Robin, the packets were distributed to two different paths, depending on the ratio RTT_{path1}/RTT_{path2}, i.e., a path with longer RTT would be distributed less packets.

Fig. 16 and Fig. 17 show the ability to aggregate bandwidth by different algorithms with the receiving buffer size 15000 bytes and 9000 bytes, respectively. We can observe that our algorithm has a better performance than EDPF and WRR. In particular, the performances of EDPF and WRR become worse with the decrease of the receiving buffer size. The reason can be found in Fig. 18, which shows the packet order of arrival at the access router for different algorithms. For Aggregated Single Link, all the packets arrived at the access router in order since Aggregated Single Link is the ideal case. For other algorithms, packet disorder occurs, i.e., some packets with smaller sequence numbers arrive later than those with bigger sequence numbers. From Fig. 18 we can observe that our proposed algorithm generated less out-of-order packets than EDPF and WRR so that it required a smaller receiving buffer size, while the performances of EDPF and WRR decreased when there was not enough buffer size for them to reorder the out-of-order packets.

(b) Difference between queueing times

In order to evaluate the difference between the queueing times for our algorithm, EDPF and WRR, we traced the sending queues on both path 0 and path 1. Fig. 19, which is with double y axes, illustrates the queueing times on both path 0 and path 1 when our algorithm was adopted. Fig. 19 also shows the difference between queueing time on the two paths. Just as Lemma 1 shows, the time difference between the two queues is upper bounded.

Fig. 20 shows EDPF's queueing time difference on two paths, which is smaller than that of our algorithm. However, this is because EDPF ignores the variable path delay on wireless links, which leads to the disorder of packets at the access router.

Fig. 21 shows the queueing times of WRR. We can observe that the time difference between the two queues rises with the number of packets. The main reason is that the variable path

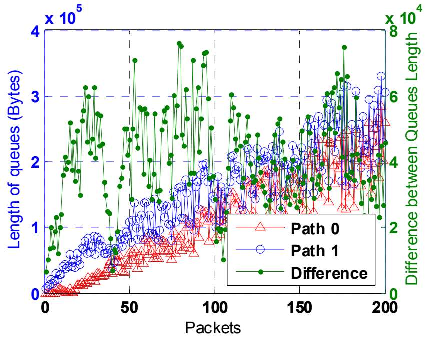

Fig. 22. Length of sending queues and their difference of our algorithm

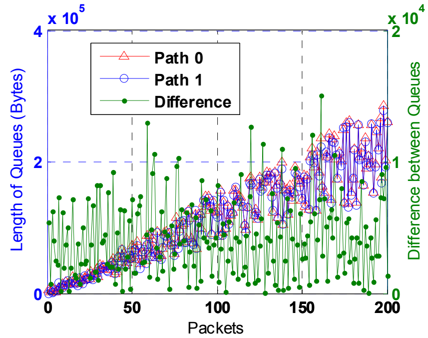

Fig. 23. Length of sending queues and their difference of EDPF

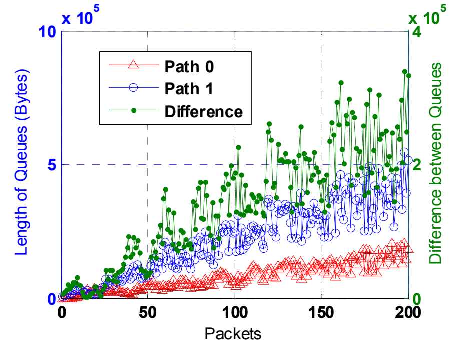

Fig. 24. Length of sending queues and their difference of WRR

parameters make it difficult to schedule packets well.

(c) Difference between the normalized queue lengths

We also compared the length of sending queues between our algorithm, EDPF and WRR. Fig. 22 shows the normalized queue lengths on path 0 and path 1, and the difference between the normalized queue lengths for our algorithm. Just as shown in Section IV, the difference between the normalized queue lengths on two paths is upper bounded.

Fig. 23 illustrates normalized queue lengths and their difference for EDPF. The difference between the normalized queue lengths is smaller than that of our algorithm. However, this is also because EDPF ignores the variable path delay on wireless links, which leads to more serious packet disorder at the access router than our algorithm.

Fig. 24 shows those with WRR. We can conclude that the difference of the queue lengths with WRR is with no upper bound. In fact, it is a function of the total packet number.

CONCLUSION

In this paper, some practical measurements of the existing 3G and 4G mobile networks are performed in a typical high-speed environment on a high-speed train. Then, a concurrent multipath transmission scheme, together with a network adaptive scheduling algorithm, is proposed, which provides transparent and effective Internet services for the onboard users by multiple available wireless networks. Some of the superior properties of our algorithm are analyzed, in terms of the length of sending queues, the buffer size for reordering and the ability to aggregate bandwidth. Simulations and experiments show that the scheme can improve QoS of Internet services on board by increasing the available bandwidth and avoiding the disorder of packets.

APPENDIX A

Lemma 1. For any two paths i, j and any time t, $C_i(t)$ and $C_j(t)$ are the queueing time on the two paths, if $C_i(t) \leq C_j(t)$, then, $C_j(t) - C_i(t) \leq S^{max}/B_{min} + D_{max}$.

Proof. The method of induction is used to prove this lemma. We first show that the lemma holds at the time $a^1 = 0$ when the first packet arrives. We then assume that the lemma holds when packet $1, 2, \cdots, x-1$ arrives, and further prove the lemma holds at a^x when the x^{th} packet arrives.

Basis. For the algorithm, the first packet will be scheduled on the path f that can deliver it the fastest to the access router. Thus, $C_f(a^1) - C_j(a^1) = S^1/B_f(a^1) \leq S^{max}/B_{min}$, where $C_f(a^1)$ is the queueing time on path f when the first packet just enters the queue, i.e., $C_f(a^1) = S^1/B_f(a^1)$. S^1 is the size of the first packet. At the same time, $C_j(a^1) = 0$ because the first packet does not enter the queue on path j $(j \neq f)$. Note that the sending queues on all paths are empty before the arriving of the first packet.

Thus, the lemma holds when the first packet just enters the queue on path f at time a^1. For any time t before the arrival of the second packet at a^2, $C_j(t) - C_i(t)$ decreases linearly with t because $C_k(t)$ is a linear function of t for any path k.

Inductive step. Assume that the lemma holds for packets $1, 2, \cdots, x-1$. Let r be the path chosen for transmission of packet x. According to Eq. (1), $b_r^x \leq b_j^x$, i.e.,

$$b_r^x = a^x + C_r(a^{x-}) + S^x/B_r\left(a^{x-} + C_r(a^{x-})\right) + D_r\left(a^x + C_r(a^x)\right)$$
$$\leq a^x + C_j(a^{x-}) + S^x/B_j\left(a^{x-} + C_j(a^{x-})\right) + D_j\left(a^x + C_j(a^x)\right)$$

where a^{x-} is the time just before a^x (at a^x packet x enters the queue). $C_r(a^{x-})$ is the queueing time just before packet x enters the queue. Thus,

$$C_r(a^{x-}) - C_j(a^{x-}) \leq \left(S^x/B_j\left(a^{x-} + C_j(a^{x-})\right) + D_j\left(a^x + C_j(a^x)\right)\right) - \left(S^x/B_r\left(a^{x-} + C_r(a^{x-})\right) + D_r\left(a^x + C_r(a^x)\right)\right)$$

Since

$$C_r(a^x) = C_r(a^{x-}) + S^x/B_r\left(a^{x-} + C_r(a^{x-})\right),\ C_j(a^x) = C_j(a^{x-}),$$

$$C_r(a^x) - C_j(a^x) \leq \left(S^x/B_j\left(a^{x-} + C_j(a^{x-})\right) + D_j\left(a^x + C_j(a^x)\right)\right) - D_r\left(a^x + C_r(a^x)\right) \quad (2)$$

Then two cases arise as follows:

Case 1. $C_r(a^x) > C_j(a^x)$. According to Eq. (2),
$C_r(a^x) - C_j(a^x) \leq S^{max}/B_{min} + D_{max}$

Case 2. $C_r(a^x) \leq C_j(a^x)$. Since the lemma holds at time a^{x-}, we have $C_j(a^{x-}) - C_r(a^{x-}) \leq S^{max}/B_{min} + D_{max}$. Since $C_r(a^{x-}) < C_r(a^x) \leq C_j(a^x) = C_j(a^{x-})$, from the above inequality we get $C_j(a^x) - C_r(a^x) \leq S^{max}/B_{min} + D_{max}$

Thus, the lemma holds at time a^x. As in the basis, $C_j(t) - C_i(t)$ decreases linearly with t at any time t before a^{x+1}. Thus, the lemma follows.

REFERENCES

[1] J. Wang, H. Zhu, and N. J. Gomes, "Distributed antenna systems for mobile communications in high speed trains," *IEEE J. Selected Areas in Commn.*, vol. 30, no. 4, pp. 675-683, May 2012.
[2] National Railway Administration of the People's Republic of China, "2014 railway statistical bulletin," http://www.nra.gov.cn/fwyd/zlzx/hytj/201504/t20150427_13281.htm, Apr. 2015.
[3] US High Speed Rail Association, "US high speed rail network map," http://www.ushsr.com/ushsrmap.html, 2015.
[4] China Internet Network Information Center, "35th China Internet network development state statistical report," https://www.cnnic.cn/hlwfzyj/hlwxzbg/201502/P020150203551802054676.pdf, Feb. 2015.
[5] Wi-Fi.org, "Total Wi-Fi® device shipments to surpass ten billion this month," http://www.wi-fi.org/news-events/newsroom/total-wi-fi-device-shipments-to-surpass-ten-billion-this-month, January 5, 2015.
[6] D. T. Fokum and V. S. Frost, "A survey on methods for broadband Internet access on trains," *IEEE Commun. Surveys Tuts.,* vol. 12, no. 2, pp. 171–185, Apr. 2010.
[7] C. Lee, M. Chuang, and et al, "Seamless handover for high-speed trains using femtocell-based multiple egress network Interfaces," *IEEE Trans. on Wireless Commun.* vol. 13, no. 12, pp. 6619-6628, 2014.
[8] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, "Network mobility (NEMO) basic support protocol," IETF RFC 3963, 2005.
[9] M. Biagi, S. Rinauro, S. Colonnese, G. Scarano, and R. Cusani, "Dynamic network selection in V2I systems," in *Proc. IEEE Vehicular Networking Conference (VNC)*, 2013, pp. 190-193.
[10] BBC News, "Wi-Fi May Tempt Travellers," http://http://news.bbc.co.uk/2/hi/technology/3729583.stm, 2004.
[11] X. Liang, F. L. C. Ong, P. M. Chan, R. E. Sheriff, and P. Conforto, "Mobile Internet access for high-speed trains via heterogeneous networks," in *Proc. 14th IEEE Proceedings on Personal, Indoor and Mobile Radio Commun., v*ol. 1, 2003, pp. 177-181.
[12] D. Ho and S. Valaee, "Information raining and optimal link-layer design for mobile hotspots," *IEEE Trans. Mobile Computing*, vol. 4, no. 3, pp. 271-284, 2005.
[13] D. Huang, P. Fan, and K. B. Letaief, "An optimal antenna assignment strategy for information raining," *IEEE Trans. Wireless Commun.*, vol. 7, no. 4, pp. 1134–1139, 2008.
[14] C. Sue, S. Sorour, Y. Yuk, and S. Valaee, "Network coded information raining over high-speed rail through IEEE 802.16j," in *Proc. IEEE PIMRC*, 2009, pp. 1138–1142.
[15] J. Zhang, Z. Tan, Z. Zhong, and Y. Kong, "A multi-mode multi-band and multi-system-based access architecture for high-speed railways," in *Proc. IEEE VTC-Fall*, Ottawa, Canada, Sep. 2010, pp. 1–5.
[16] R. Moskowitz and P. Nikander, "Host identity protocol (HIP) architecture," IETF RFC 4423, 2006.
[17] E. Nordmark and M. Bagnulo, "Shim6: level 3 multihoming shim protocol for IPv6," IETF RFC 5533, 2009.
[18] D. Farinacci, V. Fuller, D. Meyer and D. Lewis, "Locator/ID separation protocol (LISP)," IETF RFC 6830, 2013.
[19] D. Anipko, "Multiple Provisioning Domain Architecture," IETF RFC 7556, 2015.
[20] P. Rodriguez, R. Chakravorty, J. Chesterfield, I. Pratt, and S. Banerjee, "MAR: a commuter router infrastructure for the mobile Internet," in *Proc. 2nd international conference on Mobile systems, applications, and services*, 2004, pp. 217-230.
[21] D. S. Phatak, and T. Goff, "A novel mechanism for data streaming across multiple IP links for improving throughput and reliability in mobile environments," in *Proc. INFOCOM,* 2002, vol. 2, pp. 773-781.
[22] D. S. Phatak, T. Goff, and J. Plusquellic, "IP-in-IP tunneling to enable the simultaneous use of multiple IP interfaces for network level connection striping," *Computer Networks*, vol. 43, no.6, pp. 787-804, 2003.
[23] K. Chebrolu and R. R. Rao, "Bandwidth aggregation for real-time applications in heterogeneous wireless networks," *IEEE Trans. on Mobile Comput.,* vol. 5, no. 4, pp. 388-403, 2006.
[24] A. J. Nicholson and B. D. Noble, "Breadcrumbs: forecasting mobile connectivity," in *Proc. ACM MobiCom*, 2008, pp. 46-57.
[25] J. Yao, S. S. Kanhere, and M. Hassan, "Improving QoS in high-speed mobility using bandwidth maps," *IEEE Trans. on Mobile Computing*, vol. 11, no.4, 2012, pp.603-617.
[26] R. Stewart, "Stream control transmission protocol," IETF RFC 4960, 2007.
[27] R. Stewart, M. Ramalho, Q. Xie, and et al, "stream control transmission protocol (SCTP) partial reliability extension," IETF RFC 3758, 2004.
[28] C.-M. Huang and C.-H. Tsai, "WiMP-SCTP: Multi-path transmission using stream control transmission protocol (SCTP) in wireless networks,". in *Proc. AINA Workshops*, 2007, pp. 209–214.
[29] J. Liao, J. Wang, and X. Zhu, "cmpSCTP: An extension of SCTP to support concurrent multi-path transfer," in *Proc. ICC,* 2008, pp. 5762-5766.
[30] A. Ford, C. Raiciu, M. Handley and O. Bonaventure, "TCP extensions for multipath operation with multiple addresses," IETF RFC 6824, 2013.
[31] A. Ford, C. Raiciu, M. Handley, S. Barre and J. Iyengar, "Architectural guidelines for Multipath TCP development," IETF RFC 6182, 2011.
[32] C. Raiciu, S. Barre, C. Pluntke, A. Greenhalgh, D. Wischik, and Mark Handley, "Improving datacenter performance and robustness with multipath tcp," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4, 2011, pp. 266-277.
[33] C. Raiciu, C. Paasch, S. Barre, A. Ford, M. Honda, F. Duchene, and M. Handley, "How hard can it be? designing and implementing a deployable multipath TCP," in *Proc. 9th USENIX conference on Networked Systems Design and Implementation*, 2012, pp. 29-29.
[34] D. Zhou, W. Song, and M. Shi, "Goodput improvement for multipath TCP by congestion window adaptation in multi-radio devices," in *Proc. CCNC,* 2013, pp. 508-514.
[35] J. Zhao, C. Xu, J. Guan, and H. Zhang, "A fluid model of multipath TCP algorithm: Fairness design with congestion balancing, " in *Proc. ICC,* 2015, pp. 6965-6970.
[36] P. Amer, M. Becke, and et al, "Load Sharing for the Stream Control Transmission Protocol (SCTP)," draft-tuexen-tsvwg-sctp-multipath-11, 2015.
[37] N. Kuhn, E. Lochin, A. Mifdaoui, G. Sarwar, O. Mehani, and R. Boreli, "DAPS: Intelligent delay-aware packet scheduling for multipath transport," in *Proc. ICC,* 2014, pp. 1222-1227.
[38] P. Dong, X. Du, T. Zheng, H. Zhang, "Improving QoS on high-speed vehicle by multipath transmission based on practical experiment," in *Proc. 2015 IEEE Vehicular Networking Conference (VNC)*, 2015, pp. 32-35.
[39] iPerf/iPerf3, "iPerf - The network bandwidth measurement tool," https://iperf.fr/.

Ping Dong is currently an associate professor at Beijing Jiaotong University. Dr. Dong received his Ph.D. degree in Communications and Information System from Beijing Jiaotong University, Beijing, China, in 2009. He worked as a visiting scholar in Temple University, USA, in 2015. His research interests include mobile Internet, software-defined networking and network security. He has published more than 40 research papers in these areas.

Bin Song is currently a Professor in the School of Telecommunications Engineering at Xidian University. He is also the associate director of State Key Laboratory of Integrated Services Networks. He received the B.S., M.S., and Ph.D. degrees in communication and information systems from Xidian University, Xi'an, China, in 1996, 1999, and 2002, respectively. His research interests and areas of publication include wireless networks, computer networks, and video compression and transmission technologies. He has authored over 50 journal papers or conference papers and 30 patents.

Hongke Zhang is currently a professor in the School of Electronic and Information Engineering at Beijing Jiaotong University. He currently directs a National Engineering Laboratory on Next Generation Internet in China. Dr. Zhang received his Ph.D. degree in from University of Electronic Science and Technology of China, Chengdu, China, in 1992. His research has resulted in many research papers, books, patents, systems and equipment in the areas of communications, computer networks. He has served on the editorial board of several international journals.

Xiaojiang (James) Du is currently an associate professor in the Department of Computer and Information Sciences at Temple University. Dr. Du received his B.E. degree from Tsinghua University, China, in 1996, and his M.S. and Ph.D. degrees from the University of Maryland, College Park, in 2002 and 2003, respectively, all in electrical engineering. His research interests are security, systems, wireless networks, and computer networks. Dr. Du has published over 180 journal and conference papers in these areas, and has been awarded more than $5M in research grants from the U.S. National Science Foundation, Army Research Office, Air Force, NASA, and Amazon. He serves on the editorial board of three international journals.