On Uplink Measurement of Cellular Networks for Mobile Multi-Path TCP

Yung-Chih Chen[1], Erich M. Nahum[2], Don Towsley[1], and Chang Liu[1]

[1] Department of Computer Science, University of Massachusetts Amherst
{yungchih, towsley, cliu}@cs.umass.edu
[2] IBM Watson Research Center
nahum@us.ibm.com

Abstract. The popularity of social networks and the easy access to them have motivated people to share pictures and upload videos right after social events. Facilitated by the ubiquity of cellular data networks, uploading files from mobile devices to cellular networks has become an increasing trend. However, stalled connections and incomplete file transfers due to mobility remain problematic for regular TCP. Multi-path TCP, on the other hand, provides robust data transport by leveraging path diversity and can achieve this goal without terminating or splitting existing connections. It not only saves unnecessary cellular resource waste, but also improves throughput. In this paper, we first characterize the uplink of cellular data networks of two major US carriers in terms of throughput, round trip time, and loss rate. We then compare these uplink performance metrics to their downlink counterparts. Last, we show how one can benefit from running multi-path TCP for robust data transport in both static and mobile scenarios.

1 Introduction

With the rapid deployment of cellular data networks, it has been a growing trend that people transfer contents to social web sites on the go through their mobile devices [4]. For example, mobile users may post a series of pictures on Facebook right after a party, upload live videos to Youtube or make real-time video calls via Skype through various wireless media. Since the availability of Wi-Fi is usually limited within a certain wireless radio range, and cellular services might degrade or die from area to area, when uploading contents to a social site with traditional TCP, one may suffer stalled connections in mobile scenarios.

Multi-path TCP [6, 14], on the other hand, has the potential of mitigating those problems as it provides robust data transport by leveraging multiple paths available between both end hosts. When any of the paths is congested or broken, multi-path TCP performs load balancing and can dynamically offload traffic from the congested or broken paths to other working paths without stalling or terminating the existing connection. To showcase how multi-path TCP can provide robust uplink data transport in mobile scenarios, we first perform uplink measurements of cellular networks and compare them with our characterization of the performance of multi-path TCP in the downlink paths [3]. We then show how multi-path TCP can utilize these cellular paths to support mobile uploading.

The paper is organized as follows. Section 2 briefly describes the background knowledge of multi-path TCP. Section 3 presents our experiment setup of uplink measurement of cellular networks. Section 4 presents results and associated issues. Related work is discussed in Section 5 and Section 6 concludes this paper.

2 Background of Multi-path TCP

In a data transfer scenario where both end hosts are multi-homed [5] (i.e., mobile devices with Wi-Fi and 3G/4G, or smart phones with Wi-Fi and/or dual SIM cards with multiple antennas), multi-path TCP establishes a connection that utilizes all the end-to-end interface pairs. Each interface pair is called a *path*, and traffic on each path is called a *flow*. By leveraging available paths, the benefit of running multi-path TCP is three-fold. First, with additional paths, multi-path TCP improves the throughput of data transfer and still exhibits fairness to other TCP connections. Second, multi-path TCP performs load balancing across all paths and can dynamically offload traffic from congested paths to other working paths. Last, when a path breaks down or is temporarily unavailable, unlike traditional TCP, multi-path TCP deals with broken paths and continues data transfer via other working paths without stalling the connection. When the broken paths are fixed, traffic on the those paths of the multi-path TCP connection resumes and hence does not affect the operation of data transport. Follows we briefly describe the congestion control mechanisms in traditional Reno TCP and multi-path TCP.

Reno Congestion Control

The standard TCP Reno congestion controller increases and decreases the congestion window as follows:

- For each ACK on flow i: $w_i = w_i + \frac{1}{w_i}$
- For each loss on flow i: $w_i = \frac{w_i}{2}$.

When it comes to multi-path TCP, traditional TCP congestion control can not be employed due to its inability to treat other TCP connections fairly. Coupled congestion controller, on the other hand, binds the congestion windows of all working paths as a shared resource pool, and controls the congestion window as follows:

Coupled Congestion Control

- For each ACK on flow i: $w_i = w_i + \frac{1}{w}$
- For each loss on flow i: $w_i = \max(w_i - \frac{w}{2}, 0)$

A modified version of this coupled congestion controller is proposed by Wischik *et al*. [14] which takes into account paths of different RTTs and modifies the

window increase phase as follows:
– For each ACK on flow i: $w_i = w_i + \min(\frac{\alpha}{w}, \frac{1}{w_i})$
where α controls the aggressiveness of congestion window increase to compensate RTTs of different paths. The modified coupled congestion controller is implemented in the current multi-path TCP kernel [8] and is used in our multi-path TCP measurements.

Although multi-path TCP utilizes multiple paths and is designed to improve throughput, one should treat a multi-path TCP connection as a regular single-path TCP connection. When evaluating the performance of a multi-path TCP connection, one should compare it to that of the best single-path TCP connection running on any of the utilized paths, rather than to the aggregation of multiple independent single-path TCP connections for the purposes of fairness.

3 Measurement

We perform uplink measurements of two major cellular carriers in the US, Verizon and AT&T, in a geographical area of 150 square miles at Amherst in western Massachusetts. As of September 2012, Verizon wireless provides its 4G Long Term Evolution (LTE) service in part of that area, while other parts are covered by its 3G evolved High Rate Packet Data (eHRPD) service. AT&T, on the other hand, has not yet launched its 4G LTE service in western Massachusetts but has broadly upgraded its 3G High Speed Packet Access (HSPA) service to 3.5G HSPA+. In this paper, we focus on characterizing the uplink of these cellular network technologies, show how they can be utilized for multi-path TCP to provide robust uplink data transfer in mobile scenarios, and how it compares to what is possible in downlink data transfer.

3.1 Experiment Setup

The setting of our measurements consists of a wired host as receiver, residing at the University of Massachusetts Amherst (UMass) and a mobile sender. Our wired host is a Dell Precision T1600n workstation using a Intel Quad Core Xeon E3-1225 CPU (3.1 GHz processor, 8GB memory) and is configured as a multi-homed host [5], connected via 2 Intel Gigabit Ethernet interfaces to two subnets (LANs) of the UMass network. Each Ethernet interface is assigned with a public IP address and connected to the LAN via a 1 Gigabit Ethernet cable. The mobile sender is a Lenovo X220 (Intel Core i3-2350M 2.30GHz Processor with 4GB memory) with a built-in 802.11 a/b/g Wi-Fi interface. The mobile sender is connected to one or two additional cellular broadband data devices from Verizon and AT&T. Note that these devices can automatically switch between 3G and 4G service.

In all our experiments, we collect packet traces from all interfaces used in the mobile sender, and from both interfaces of the UMass receiver using *tcpdump*[12]. Collected traces are analyzed using *tcptrace*[13]. For each experiment, unless specifically mentioned, the mobile sender uploads a 40 MB file to the UMass server, and each result shown below is the average of (at least) 6 runs of any particular configuration in a day (morning/afternoon/evening).

We are interested in the following performance metrics:

- **Throughput**:
  The throughput is computed as the number of bytes sent divided by the elapsed time (in Mbps).
- **Round trip time (RTT)**:
  We measure RTT on a per-flow basis. Denote T_r as the mobile sender's receive time of an ACK packet for a packet sent by the sender at time T_s. The RTT is the time difference between these times, $RTT = T_r - T_s$, and only non-duplicate ACKs and non-retransmitted packets are considered.
- **Loss rate**:
  The loss rate is calculated on a per-flow basis, and it is computed as the total number of retransmitted data packets divided by the total number of data packets sent from the mobile sender over an interval of time.

3.2 Uplink Maximum Segment Size

Before presenting the measurement results, we first address a practical issue related to maximum segment size (MSS) when running multi-path TCP over a cellular network. In an end-to-end data transfer, throughput is typically an increasing function of MSS as it reduces the number of transmissions and interruptions needed to transfer a fixed amount of data. Most transmission links define a maximum transfer unit (MTU) for a packet: the maximum packet size that can pass through a specific media link. Currently Ethernet sets $MTU = 1,500$ bytes [10], consisting of 20 bytes of IP-header, 20 bytes of TCP-header, and the rest 1,460 bytes is its MSS. As multi-path TCP requires additional space in the option field, the current multi-path TCP kernel implementation [8] sets the default MSS to 1,400 bytes [2] over all paths. This value works in most scenarios [11], but does not function properly when uploading data to cellular networks.

From our observation, the default MSS of current multi-Path TCP implementation with 1,400 bytes works for downlink transmission of both Verizon and AT&T, and for Verizon 3G uplink ($MSS \leq 1428$). However, it does not work for Verizon 4G uplink ($MSS \leq 1372$), and AT&T 3.5G uplink ($MSS \leq 1357$). To ensure a successful multi-path TCP uplink transmission, the default MSS across all the paths should be set to the minimum MSS of all the involved carrier technologies. Therefore, the default MSS value in current multi-path TCP implementation [2] should be reconsidered for cellular uplink connections.

4 Results

4.1 Single-path TCP

In this section, we present measurement results of single-path TCP in both static and mobile scenarios and compare them with our previous study of downlink performance [3]. Table 1 lists the performance metrics of different technologies of Verizon and AT&T in static and mobile scenarios. For measurements of mobile scenarios, we drove around Amherst area with an average speed of 40 MPH.

From our observations, uplink connections of Verizon 4G and AT&T 3.5G have much smaller bandwidths, larger RTTs, and higher loss rates compared to their downlink counterparts. Verizon 3G uplink, although with a smaller bandwidth and larger RTTs than those of downlink, has a lower average loss rate than downlink. In terms of the performance impact of mobility, we do not observe performance degradation of these cellular technologies in mobile scenarios, and the performance comparison results are consistent in static and mobile scenarios.

In static scenarios, the average downlink bandwidths of Verizon 4G, AT&T 3.5G, and Verizon 3G are 12.1 Mbps, 6.49 Mbps, and 1.85 Mbps, respectively, and are much larger than their uplink bandwidths. The average downlink round trip times of Verizon 4G (77.31ms), AT&T 3.5G (115.32ms), and Verizon 3G (362.37ms) are also smaller than their uplink RTTs. Although the lost rates of uplink connections in Verizon 4G and AT&T 3.5G are much higher than their downlink counterparts (0.13% and 0.14%, respectively), Verizon 3G uplink, however, has a lower loss rate (0.03%) than its downlink loss rate (0.10%). This might be because Verizon 3G network tends to queue uplink packets for a much longer time (an average of 997.58 ms) to reduce potential packet losses.

Carrier	Scenario	Technology	BW (Mbps)	RTT (ms)	loss rate (%)
Verizon	**Static**	4G LTE	4.24 ±(1.98)	92.05 ±(24.80)	0.32 ±(0.38)
AT&T		3.5G HSPA+	0.89 ±(0.06)	287.63 ±(207.37)	0.45 ±(0.28)
Verizon		3G eHRPD	0.68 ±(0.08)	997.58 ±(373.86)	0.03 ±(0.01)
Verizon	**Mobile**	4G LTE	4.34 ±(1.44)	111.70 ±(55.37)	0.12 ±(0.07)
AT&T		3.5G HSPA+	0.91 ±(0.10)	276.30 ±(158.63)	0.60 ±(0.39)
Verizon		3G eHRPD	0.56 ±(0.06)	859.52 ±(574.62)	0.10 ±(0.07)

Table 1. Uplink measurements of single-path TCP

4.2 Uplink Flow Control and Queue Management

Figures 1-3 show the throughput (top figures) and round trip times (bottom figures) as a function of time when uploading 40 MB data via Verizon 4G, AT&T 3.5G, and Verizon 3G data networks. Note that the vertical lines are where packet retransmissions take place.

The throughput curve of a regular TCP Reno connection normally exhibits a prominent TCP sawtooth pattern as it reduces congestion window by one half when a packet loss occurs. We observe the pattern in Verizon 4G throughput curve but not in AT&T 3.5G and Verizon 3G. The throughput for the latter two is relatively stable with very little variance over time. At the bottom of Figure 2, we observe that the packet RTTs of AT&T 3.5G and Verizon 3G usually

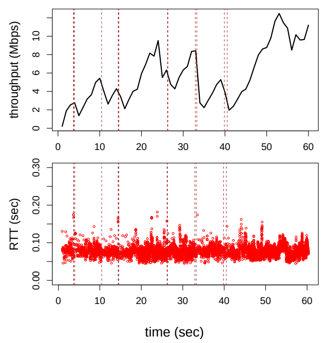

Fig. 1. Verizon 4G

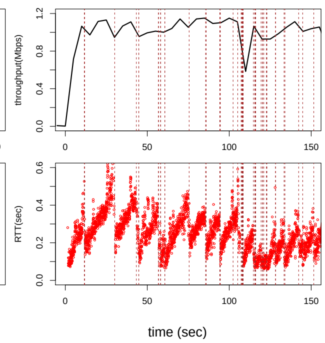

Fig. 2. AT&T 3.5G

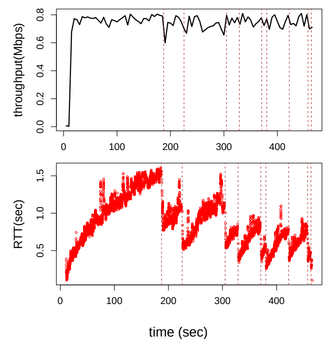

Fig. 3. Verizon 3G

increase over time until a packet retransmission occurs (due to packet loss or being dropped because of a full network queue).

As throughput equals *cwnd/RTT*, if the RTT increases at a similar rate as the congestion window (cwnd) does, the throughput can be maintained at a stable rate. In our previous downlink study [3], we did not observe this increase-then-drop RTT pattern in Verizon 4G and AT&T 3.5G (as do they in Figures2 and 3), and hence their downlink throughput curves have a more prominent sawtooth behavior.

To quantify throughput variability of each carrier, we introduce a statistical measure called *coefficient of variation (CV)*, a normalized measure of dispersion of a probability distribution. It is defined as $C_v = \sigma/\mu$ (where σ and μ are standard deviation and mean) and is used to compare two or more groups of data where their means are of different scales. When the standard deviation of the samples is close to the mean, its coefficient of variation approaches 1; if all the samples in a dataset are of the same value, its coefficient of variation is 0.

Carrier	Technology	Uplink		Downlink	
		Static	Mobile	Static	Mobile
Verizon	4G LTE	0.44 ±(0.16)	0.47 ±(0.09)	0.27 ±(0.09)	0.25 ±(0.08)
AT&T	3.5G HSPA+	0.17 ±(0.05)	0.13 ±(0.05)	0.18 ±(0.04)	0.18 ±(0.07)
Verizon	3G eHRPD	0.08 ±(0.02)	0.30 ±(0.08)	0.22 ±(0.10)	0.41 ±(0.05)

Table 2. Coefficient of variation: single-path TCP throughput

Table 2 lists the average coefficients of throughput variation of Verizon and AT&T cellular technologies[3], and we ignore the first five-second values as they

[3] Note that the throughput is smoothed by five seconds and the results are close to those smoothed by one second. Also, results in Table 1 are the mean and standard deviation of throughput means, and hence do not reflect the variation of each dataset.

contain slow-start speed ramp-up. In general, the results of throughput variability of AT&T 3.5G service is consistent in uplink and downlink, regardless of static or mobile scenarios. Verizon, on the other hand, has very extreme behaviors in its 3G and 4G uplink services. Verizon 4G has the highest throughput variability coefficient and Verizon 3G has the lowest one. The large coefficient value of Verizon 4G uplink, $C_v = 0.44$, is consistent with the TCP sawtooth behavior. This value is 4 times larger than Verizon 3G's average in static scenarios, and 2.5 times greater than the averages of AT&T's static and mobile scenarios. Moreover, our observations show that Verizon 3G seems to perform more stable in the static scenarios than in the mobile ones.

4.3 Multi-path TCP

We first present a scenario of running multi-path TCP with two interfaces (four flows). The mobile sender is connected with mobile broadband data dongles from Verizon and AT&T, which can automatically switch between 3G and 4G. As stated earlier, the UMass receiver is a multi-homed host with two interfaces connected to two different subnets. When the mobile sender initializes a multi-path TCP connection, there are four flows between the sender and the receiver: two from the Verizon device and two from the AT&T device, to each of the UMass receiver's interfaces.

Carrier	Technology	BW (Mbps)	RTT (ms)	loss rate (%)
Verizon-1	4G LTE	1.30 ±(0.51)	84.52 ±(34.93)	0.22 ±(0.15)
Verizon-2	4G LTE	3.17 ±(3.02)	139.67 ±(86.80)	0.31 ±(0.12)
AT&T-1	3.5 HSPA+	0.42 ±(0.08)	204.98 ±(102.10)	0.32 ±(0.14)
AT&T-2	3.5 HSPA+	0.40 ±(0.08)	206.87 ±(107.63)	0.31 ±(0.19)

Table 3. Uplink measurement of 4-flow Multi-path TCP connection

Table 3 presents the performance of each flow of the multi-path TCP connection. The average throughput of this 4-flow multi-path TCP connection is 5.29 Mbps, which is 25% higher than that of the best single-path TCP connection over Verizon (4.24 Mbps). In contrast, the average downlink throughput of 4-flow multi-path TCP at the same location is 8.5 Mbps, with a 7.5% performance gain compared to the best single-path TCP connection over Verizon (7.85 Mbps).

In the following, we present two mobile scenarios where a mobile sender has three interfaces (Verizon, AT&T, and Wi-Fi), but with intermittent Wi-Fi connectivity and occasional cellular service degradation.

Mobile Upload with Intermittent Wi-Fi

As there is free public Wi-Fi on UMass campus, when a mobile user uploads contents from a campus bus or a moving vehicle, he will have intermittent Wi-Fi connectivity. Figure 4 presents a scenario when a user uploads contents to the

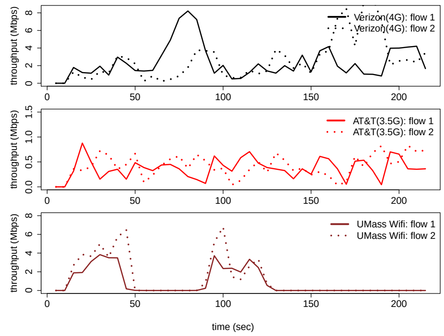

Fig. 4. Intermittent Wi-Fi

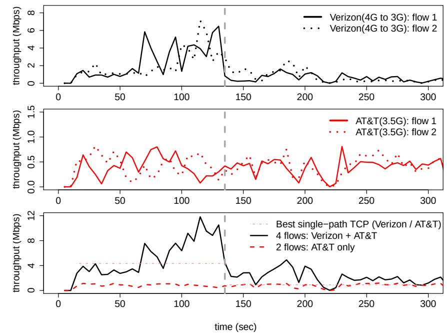

Fig. 5. Service downgrade: 4G to 3G

UMass host with an average moving speed of 20 MPH (campus speed limit). Two Wi-Fi flows are first established, together with two Verizon flows and two AT&T flows. Upon leaving the coverage of Wi-Fi hotspot, the two Wi-Fi flows soon break while the other four flows remain operational. When the mobile user reaches another Wi-Fi hotspot at time 85 seconds, the two Wi-Fi flows resume. Data transfer utilizes Wi-Fi before the connectivity is lost again at time 130 seconds. As Wi-Fi usually provides higher uplink bandwidth than cellular networks, in this case, multi-path TCP not only utilizes Wi-Fi's high bandwidth, but also dynamically offloads traffic from unavailable Wi-Fi links to other working paths without affecting the operation of existing connections.

Mobile Upload with Service Degradation

Since Verizon's 4G service only covers a part of Amherst in Massachusetts (mostly UMass campus area), Figure 5 illustrates the scenario where a mobile sender leaves UMass campus (with Verizon 4G service) and enters a Verizon 3G service zone. The top and middle figures present the throughput of two Verizon flows and two AT&T flows as a function of time, while the bottom shows the aggregate throughput of both AT&T flows, and the aggregate throughput of the entire multi-path TCP connection of 4 flows.

Upon leaving the Verizon 4G zone, the two Verizon flows soon stall for approximately five seconds before switching to 3G service with a much smaller throughput. For data transfer before switching from 4G to 3G at time 135 seconds (the dashed vertical lines in Figure 5), the average throughput of the multi-path TCP connection is 5.64 Mbps, with a roughly 30% performance gain comparing to the best mobile single-path TCP connection over Verizon 4G at 4.34 Mbps (the first horizontal dashed line at the bottom of Figure 5). After switching to 3G at time 300 seconds, the average throughput of the multi-path TCP connection is 2.09 Mbps, which is twice as much as that of the best mobile single-path TCP connection over AT&T 3.5G at 0.91 Mbps (the second horizontal dashed line at the bottom of Figure 5). In this scenario, multi-path TCP not only provides robust data transport when certain paths are temporarily unavailable or suffering service degradation, but also improves the performance of the entire data connection. We also conducted experiments of moving from a 3G zone to a 4G zone, but the Verizon device tends to stay with 3G until the end of the data transfer. Due to the lack of space, we do not present the results here.

5 Related Work

Previous works on multi-path TCP mostly focus on improving throughput and connection robustness for wired Internet [14, 11]. As transferring contents from mobile devices has become an increasing trend [4] and uplink usually has lower bandwidth than downlink, it is therefore critical to provide robust mobile data connection to reduce the waste of cellular uplink resources. One approach is to augment 3G with Wi-Fi by offloading delay-tolerant data to Wi-Fi [1]. However, this approach splits existing connection into multiple short and delayed ones, and hence introduces additional resource waste in cellular networks due to additional connection establishments and communication overhead. Raiciu *et al*. [9] proposed a similar scheme as [1] with multi-path TCP, but focus on the responsiveness of applications and resource allocation of multi-path TCP.

Our work differs from theirs in that we focus on supporting mobile uploading with multi-path TCP to reduce the waste of precious uplink cellular resources. Although recent study [7] also performed measurements on cellular networks, their results were passively collected by unknown users running mobile APPs. Our measurements focus more on the performance metrics of connections with enough traffic riding on them, such as throughput, round trip time, and loss rate. In our previous study [3], we investigated at the performance of multi-path TCP in downlink paths of cellular networks. In this paper, we not only characterize the uplink of these cellular network technologies, but also focus on the performance comparison of uplink and downlink. Moreover, we also show how these cellular technologies can be utilized for multi-path TCP to provide robust uplink data transfer in mobile scenarios. To our knowledge, this is the first work that characterizes the uplink of cellular networks for mobile uploading with multi-path TCP.

6 Conclusion

The ubiquity of cellular networks has enabled the possibility of uploading data on the go. However, cellular services might die or downgrade in different areas, and incomplete data transfer and stalled connections can cause huge waste of cellular uplink resources. As multi-path TCP can provide robust data transport, in this paper, we seek to support mobile uploading with multi-path TCP.

We first perform uplink measurements of single-path TCP of two major US cellular network carriers, with a focus on throughput, round trip times, and loss rates, and compare them with downlink performance. As uplink resources are more precious than downlink (in terms of above performance metrics), we observe that some carriers perform rate control over their uplink channels for more stable throughput performance and lower loss rates. To further reduce uplink resource waste, we show that running multi-path TCP for uplink data transfer is a promising solution. It not only provides great performance improvement, but also mitigates the impact from intermittent Wi-Fi connectivity or cellular service degradation, and thereby can provide robust uplink data transport in mobile scenarios.

Acknowledgments

This research was sponsored by US Army Research laboratory and the UK Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the US Army Research Laboratory, the U.S. Government, the UK Ministry of Defense, or the UK Government. The US and UK Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon. This material is also based upon work supported by the National Science Foundation under Grant No. CNS-1040781.

References

1. A. Balasubramanian, R. Mahajan, and A. Venkataramani. Augmenting mobile 3G using WiFi. In MobiSys'10.
2. S. Barré, C. Paasch, and O. Bonaventure. Multipath TCP: from theory to practice. In NETWORKING'11.
3. Y.-C. Chen, Y.-s. Lim, E. M. Nahum, R. J. Gibbens, and D. Towsley. Characterizing 4G and 3G networks: Supporting mobility with multi-path tcp. Technical report, CS Department, UMass Amherst (UM-CS-2012-022), 2012.
4. J. Erman, A. Gerber, K. K. Ramadrishnan, S. Sen, and O. Spatscheck. Over the top video: the gorilla in cellular networks. In IMC '11.
5. A. Ford, C. Raiciu, H. M., and O. Bonaventure. TCP extensions for multipath operation with multiple addresses. draft ietf-mptcp-multiaddressed-07, 2012.
6. H. Han, S. Shakkottai, C. V. Hollot, R. Srikant, and D. Towsley. Multi-path TCP: a joint congestion control and routing scheme to exploit path diversity in the internet. *IEEE/ACM Trans. Netw.*, 14:1260–1271, December 2006.
7. J. Huang, Q. Feng, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck. A close examination of performance and power characteristics of 4G LTE networks. In MobiSys'12.
8. MultiPath TCP Linux Kernel implementation. http://mptcp.info.ucl.ac.be/.
9. C. Paasch, G. Detal, F. Duchene, C. Raiciu, and O. Bonaventure. Exploring mobile/wifi handover with multipath TCP. In CellNet'12.
10. RFC 791 Internet Protocol. http://www.rfc-editor.org/rfc/rfc791.txt.
11. C. Raiciu, C. Paasch, S. Barre, A. Ford, M. Honda, F. Duchene, O. Bonaventure, and M. Handley. How hard can it be? designing and implementing a deployable multipath TCP. In NSDI'12.
12. tcpdump. http://www.tcpdump.org.
13. tcptrace. http://www.tcptrace.org.
14. D. Wischik, C. Raiciu, A. Greenhalgh, and M. Handley. Design, implementation and evaluation of congestion control for multipath TCP. In NSDI'11.