Speedtest-Like Measurements in 3G/4G Networks: The MONROE Experience

Speedtest-like Measurements in 3G/4G Networks: the MONROE Experience

Ali Safari Khatouni[1], Marco Mellia[1], Marco Ajmone Marsan[1,4],
Stefan Alfredsson[2], Jonas Karlsson[2], Anna Brunstrom[2],
Özgü Alay[3], Andra Lutu[3], Cise Midoglu[3], Vincenzo Mancuso[4]

[1] Politecnico di Torino, Italy [2] Karlstad University, Sweden

[3] Simula Research Laboratory, Norway [4] IMDEA Networks Institute, Spain

***Abstract*— Mobile Broadband (MBB) Networks are evolving at a fast pace, with technology enhancements that promise drastic improvements in capacity, connectivity, coverage, i.e., better performance in general. But how to measure the actual performance of a MBB solution? In this paper, we present our experience in running the simplest of the performance test: "speedtest-like" measurements to estimate the download speed offered by actual 3G/4G networks. Despite their simplicity, download speed measurements in MBB networks are much more complex than in wired networks, because of additional factors (e.g., mobility of users, physical impairments, diversity in technology, operator settings, mobile terminals diversity, etc.).**

We exploit the MONROE open platform, with hundreds of multihomed nodes scattered in 4 different countries, and explicitly designed with the goal of providing hardware and software solutions to run large scale experiments in MBB networks. We analyze datasets collected in 4 countries, over 11 operators, from about 50 nodes, for more than 2 months. After designing the experiment and instrumenting both the clients and the servers with active and passive monitoring tools, we dig into collected data, and provide insight to highlight the complexity of running even a simple speedtest. Results show interesting facts, like the occasional presence of NAT, and of Performance Enhancing Proxies (PEP), and pinpoint the impact of different network configurations that further complicate the picture. Our results will hopefully contribute to the debate about performance assessment in MBB networks, and to the definition of much needed benchmarks for performance comparisons of 3G, 4G and soon of 5G networks.

I. Introduction

The society's increased reliance on Mobile Broadband (MBB) networks has made provisioning ubiquitous coverage and providing high network performance and user quality of experience (QoE) the highest priority goal for mobile network operators. This motivates researchers and engineers to further enhance the capabilities of MBB networks, by designing new technologies to cater for a plethora of new applications and services, for the growth in traffic volume, and for a wide variety of user devices.

When coming to performance assessment, the picture is much more complicated in MBB networks than in wired networks. Even the simplest of the tests, i.e., a "speedtest-like" measurement of the single TCP bulk download speed using HTTP, may become complicated to interpret in MBB networks, due to the large number of factors that affect performance. Physical impairments, mobility, variety of devices, presence of Performance Enhancing Proxies (PEP) [1], different access network configurations, etc., all possibly impact the measurement results, and complicate the picture.

When facing performance assessments, a common approach is to rely on end users, and their devices, to run tests by visiting a website [2], or running a special application [3]. Federal Communications Commission (FCC) follows a similar crowdsourcing approach to measure MBB networks in the USA [4]. Network operators and independent agencies sometimes perform drive tests to identify coverage holes or performance problems. These tests are, however, expensive, do not scale well [5], and little information on methodology is given.

Here, we rely on the MONROE [6] open platform, that offers an independent, multihomed, large scale monitoring platform for MBB testing in Europe. It includes hundreds of mobile and stationary nodes, each equipped with three 3G/4G interfaces, and offers both hardware and software solutions to run experiments in a scalable manner. In this paper, we report our experience in designing, running, and analyzing speedtest experiments on MONROE nodes. After instrumenting both clients and servers with passive monitoring solutions that expose physical, network, and transport layer metrics, we instructed about 50 nodes to download a 40MB file from well-provisioned HTTP servers. We repeated the experiment every three hours, from each 3G/4G interface, and collected measurements for more than 2 months in 4 countries and on 11 different operators. By design, we tried to minimize randomness: all nodes have the same hardware, run the same software; only stationary nodes have been used; tests have been repeated multiple times, from multiple nodes connected in the same area, with the same operators and subscribed services. No interfering traffic was present on the terminal.

Despite the large dataset, and the scientific approach, we find that running even a simple speedtest-like experiment proves to be very complicated, with results that apparently vary on a large scale, with no obvious correlations, and sometimes in an unpredictable way. We observe the presence of NAT, and of transparent proxies, as well as different access network configurations, and roaming agreements, each adding complexity to the already complicated picture. Thanks to the MONROE platform, we design and run further experiments to corroborate our findings, and better understand the results.

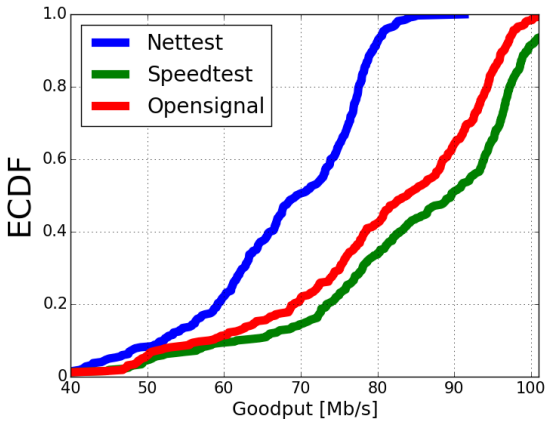

Fig. 1: ECDF of reported download rate for different tools in 4G

While preliminary, we present our finding (and make available all raw data) in the hope to shed some light into the debate about performance assessment in MBB environments. Indeed, since the issue is far from trivial, we believe there is a need to define benchmarking principles that allow to fairly compare performance in 3G/4G (and soon in 5G) networks.

The rest of this paper is organized as follows. In Section II we present the motivation of this work. In Section III we describe the MONROE platform and the measurement approach we use to collect and analyse the collected dataset. Our methodology is discussed in Section IV. In Section V we present our finding. In Section VI we briefly discuss the related work. Finally, in Section VII we conclude the paper and we discuss future research issues.

II. Motivation

To take a first look into speedtest measurements in commercial MBB networks, we conducted an initial measurement campaign, and measured different speedtest apps under the same conditions, using an Android phone as a regular user could do, from home. There are a number of crowdsourced apps for measuring MBB performance via end-user devices. Among them, we choose the most popular ones: *Speedtest* by Ookla [2], *OpenSignal* by OpenSignal [7], *RTR-Nettest* by Austrian Regulatory Authority for Broadcasting and Telecommunications (RTR) [8].

Typical performance measurements by such tools comprise Downlink (DL) and Uplink (UL) data rate, and latency. Here we focus on download speed only.

For our measurement campaign, we run speedtest measurements with Speedtest (v3.2.29), OpenSignal (v5.10), and Nettest (v2.2.9). To ensure the fair comparison of the tools, we execute the tools in rounds where each tool is run one after the other and in randomised order on a stationary measurement device located in Oslo, Norway, when connected to the same network in 4G.

We ran 320 batches of measurements in total. Fig. 1 shows the Empirical Cumulative Distribution Function (ECDF) of download rate values reported by the tools. Surprisingly, we observe a large variation in measurements, both within runs

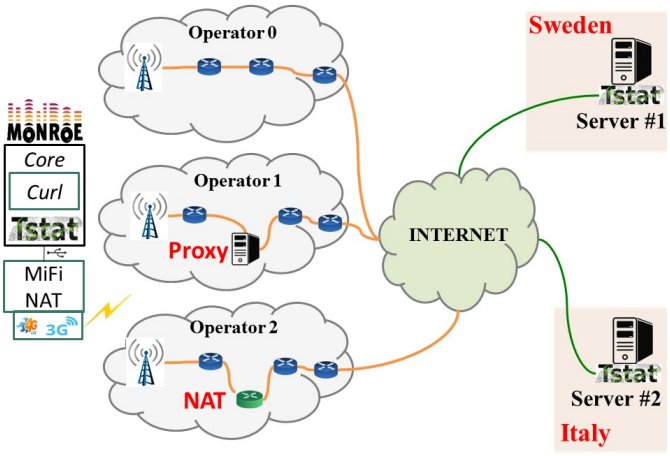

Fig. 2: Experiment setup

of the same tool (max-min variation of 60 Mb/s, see the Opensignal in Fig. 1), and between tools (max-max variation of 20 Mb/s range, see the difference between Nettest and Speedtest in Fig. 1).

These large differences indicate a significant variation in both measurement methodology and network condition, which we have confirmed through the reverse-analysis of traffic traces collected during measurements with different tools. Thus the natural question is "Can we reliably benchmark download speed in MBB networks?".

III. Measurement Setup

In this section, we briefly describe the MONROE platform and the collected dataset.

A. MONROE platform

The MONROE platform is available for researchers to run experiments on MBB networks in Europe. Nodes are deployed in 4 countries (Italy, Norway, Spain, and Sweden), and include both stationary and mobile nodes, the latter traveling on vehicles like buses, trains, trucks, etc.

MONROE offers an open MBB platform which enables users to run custom experiments by means of Docker [9] containers, and to schedule their experiments to collect data from operational MBB and WiFi networks, together with MONROE metadata[1], i.e., the full context information about the state of a node (e.g., signal strength, frequency, technology in use, cell-ID, etc.), and its location as from GPS. The MONROE node [6] is a multihomed system with 3 regular MBB subscriptions which are different in each country, some of which used abroad in roaming. All nodes are based on the same hardware – a dual core x86-based APU with 2GB of RAM – and connected to three MBB networks using three MiFi [10] cat.4 LTE modem (ZTE MF910 at the time of running the experiments in this paper).

Each node runs a stripped down version of Ubuntu Linux, with a Docker setup that allows experimenters to deploy their

[1] https://github.com/MONROE-PROJECT/data-exporter

TABLE I: The number of experiments in the dataset

country	City (sites)	Operator	# Nodes	# Experiments
Italy	Torino(4) Pisa(5)	op0	12	1995
		op1	14	2184
		op2	14	2316
Sweden	Karlstad(7)	op0	28	3029
		op1	28	2644
		op2	28	3117
Spain	Madrid(6) Leganes(5)	op0	18	4924
		op1	15	3502
		op2	7	1888
Norway	Fornebu(3) Oslo(4) Bergen(4)	op0	13	2437
		op1	12	2220
Total	8	11	73	30256

experiment by simply selecting the desired nodes and time to run their software on a centralized scheduler. The latter automates the Docker distribution on selected nodes, runs the experiment, and collects data and results, exposing the previously mentioned metadata about node status. The platform is also instrumented to regularly run baseline experiments (e.g., HTTP download, Ping, passive measurements, ...). All produced data is stored in the project database and available for researchers.

B. Basic HTTP test

Fig. 2 shows the experiment setup we consider in this paper. The leftmost element is the MONROE node. It contains the core components, with containers that run active experiments. Traffic generated by the applications passes through the selected MiFi modem where a NAT is in place, then goes through the ISP network, and the Internet, toward the selected server – on the rightmost part of figure. Each node runs also Tstat [11], a specialized passive sniffer. Tstat captures traffic on each MBB interface and extracts statistics by passively observing packets exchanged with the network. Another instance of Tstat runs on the server side, thus capturing and processing traffic at the other end of the path.

As previously mentioned, each MONROE node regularly runs a basic set of experiments. Among these, the HTTP download experiment uses single thread *curl* to download a 40 MB file for a maximum of 10 seconds from dedicated and not-congested servers in two countries, one in Italy, one in Sweden.[2] Network configuration may change from country to country, and from operator to operator as depicted in Fig. 2. Beside the NAT at the MiFi router, the ISP can provide a public IP address to the modem (e.g., Operator 0) and no other NAT or middlebox on the path. Alternatively, the ISP might use some kind of PEP (e.g., Operator 1), or it can use Carrier Grade NAT to do NAT/NAPT (e.g., Operator 2).

In this work, we consider measurements that were run during September and October 2016 in four countries and different sites. We consider only stationary nodes. The experiment ran every 3 hours in synchronized fashion. Table I reports the

[2]During the HTTP test no other experiment can run. The 3 h periodicity and 10 s limit are imposed to avoid booking the platform for long time. The 40 MB file size limits the total volume of data to less than 9.6 GB/month and avoids to erode the limited data quota of each subscription.

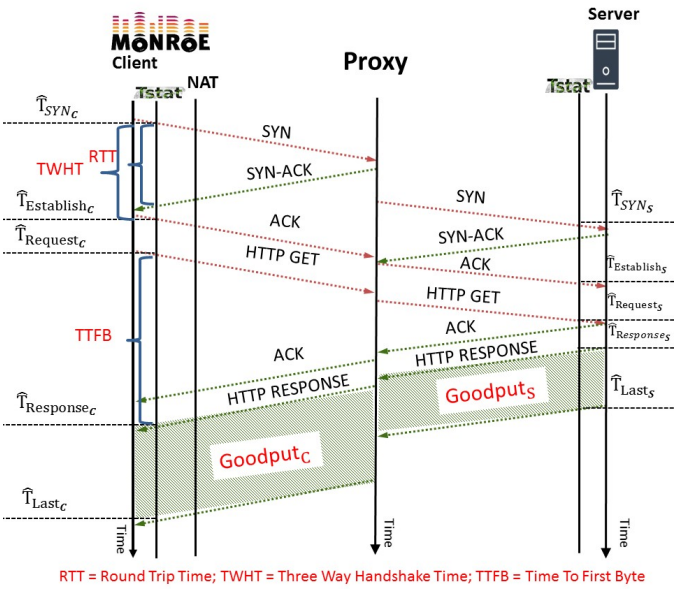

Fig. 3: Packet timeline in case of PEP in the path

total number of nodes and the number of experiments for each operator. Overall, we collected more than 30 000 experiments from 11 operators. ISPs were subjected to different numbers of experiments. The reason can be coverage holes, exhausted data quota on subscriptions, or rare failures inside the nodes. The name of the ISP is specified by a number, to avoid exposing the operator name – our goal is not to provide a ranking among ISPs but rather to observe if it would be possible to reliably measure performance. During experiments, all networks were in normal operating conditions (and unaware of our tests).

The active application and passive flow-level traces on the client and server sides cannot give us information about the technology and signal strength at the MBB channel during the experiment. Therefore, we use the metadata collected by the MONROE platform to augment the information about the access link status. The MONROE metadata are event-based data collected by passively monitoring the statistics exposed directly from the MiFi modems through their management interface. This data is transmitted and stored in the project database for analysis, and can be easily correlated to each node and interface.

C. Additional tests

To verify some of the hypotheses about the presence of NAT or PEP in the ISP network, we additionally instrumented a subset of nodes to run HTTP tests, but against HTTP servers running on different TCP ports. In particular, we checked possible HTTP-related ports (80, 8080), HTTPS port (443) and random ports (4981, 19563). Again, Tstat runs on both client and server, and lets us verify the presence of middleboxes by contrasting the measurements on both sides.

IV. Methodology

Here we detail the methodology we used to process the collected data. Let us first start describing in more details the available information at our disposal.

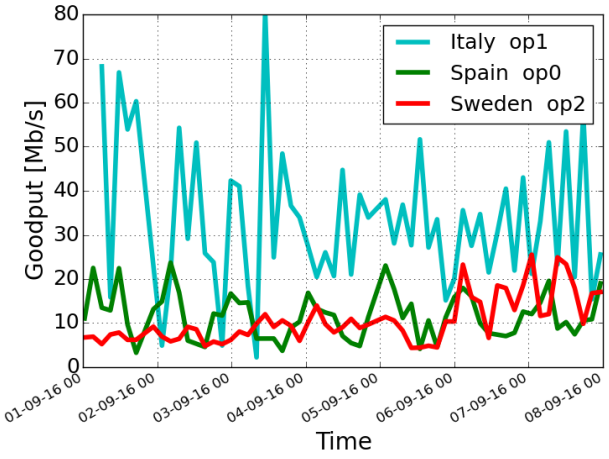

Fig. 4: Client-side goodput observed over one week for three operators

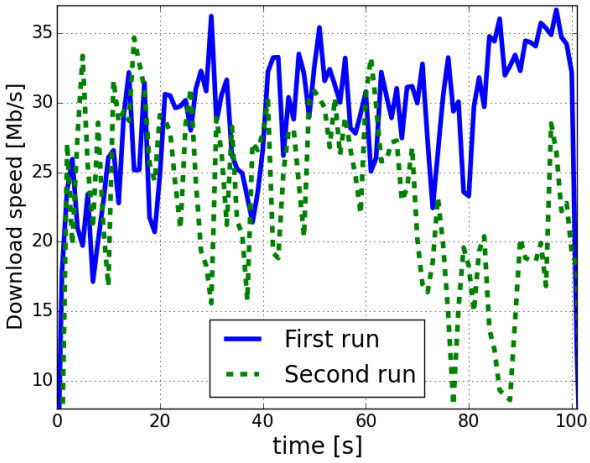

Fig. 5: Evolution over time of download speed in two simple run of 100 s on op2 in Italy

A. Measurement definition

Fig. 3 reports the possible setup during an experiment. The client (on the left) opens a TCP connection, and fetches the file via HTTP. Tstat on the client side sniffs packets, and extracts measurements by correlating the sent and received segments. For instance, it extracts the Round Trip Time (RTT) of each TCP segment/acknowledgement pair, the Time to complete the Three Way Handshake Time (TWHT), the Time To receive the First Byte from the server (TTFB), and the download speed. In the example, there is a PEP, which terminates the TCP connection from the client side, while opening another one toward the server. The second Tstat instance running on the server observes the segments being exchanged between the PEP and the server, and collects statistics that we can later contrast with those collected on the client side.

We now define the most important measurements we use in this work. We indicate measurements collected on the client side or server side with subscript C or S, respectively.

1) Goodput – $\widehat{G}$: $\widehat{G}$ is the most important measurement, and is defined as the average rate at which the client receives information at the application layer. Let $\hat{T}_{Response_C}$ and $\hat{T}_{Last_C}$ (see Fig. 3) be the timestamps of the first and the last data packet at the client side, and let D be the size of the application payload size sent by the server. We define the client-side goodput as:

$$\widehat{G}_C = \frac{D}{\hat{T}_{Last_C} - \hat{T}_{Response_C}}$$

Since Tstat is co-located at the client, this measurement is actually the same as the measure computed directly by the *curl* application.

2) Round Trip Time – RTT: Tstat measures the RTT by matching the data segment and the corresponding acknowledgement in a flow (as depicted in Fig. 3). For each segment/ack pair, Tstat obtains a RTT sample. It then computes the average, standard deviation, minimum and maximum among all RTT samples seen in the same TCP connection. On the client side, Tstat gets a reliable measurement of the RTT between the TCP client and the TCP server (or PEP) nodes. On the HTTP server, Tstat measures the RTT from the server to the client (or PEP).

3) Time To Live – TTL: For each packet, Tstat extracts the TTL values from IP packets, and tracks minimum, maximum, and average values seen in all packets of the same TCP flow. On the client side, we consider the maximum TTL observed in packets transmitted by the server (or PEP). This is linked to the number of hops that the packets in the flow have traversed before reaching their destination.

4) TCP options: For each TCP connection, Tstat logs information about TCP options such as Timestamps, Maximum Segment Size (MSS), and negotiated window scale factor [12]. In the MONROE platform, all nodes run the same software and hardware. Since we have also control on the server side, we know exactly which options are declared and supported by both endpoints. If the ISP does L4 mangling, or a PEP is present on the path, Tstat could observe different TCP options on the client side and server side.

5) Received Signal Strength Indicator – RSSI: Among the information the MONROE node collects from the modem, we use the RSSI reported in dBm (logarithmic scale) as indicator of the quality of the channel. The RSSI indicates the total received signal power and typically, -100 dBm and -60 dBm indicate low signal level and very strong signal level, respectively. Recall that all nodes use the same MiFi modems, so this information is measured consistently by the platform. We use the RSSI value reported at the time $\hat{T}_{SYN_C}$.

B. Joining client with server data

All connections go through at least the first NAT at the MONROE node. This implies that Tstat at the client side sees the client *private* IP address provided by the MiFi modem, while Tstat at the server would observe the client *public* IP address.[3] If there is a middle-box in the ISP network, it could further change the IP address, and the port numbers. Thus, matching the connection observed at the server side to the

[3]The MiFi does not change the TCP port number, but only the client IP address.

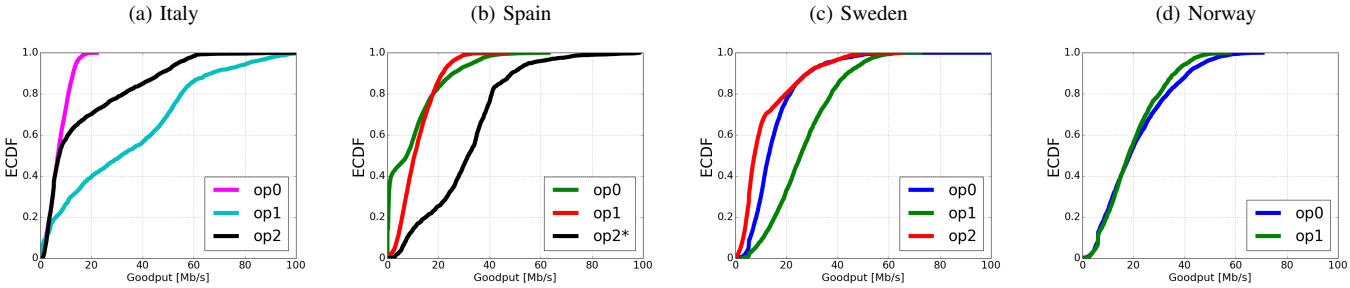

Fig. 6: ECDF of the download client-side goodput for the four considered countries

one seen at the client side is not trivial. The MONROE metadata exposes the actual IP address provided by the operator (either private or public) to the MiFi modem, so that we can use this to map connections on the client and server side. We call it "client IP" for simplicity in the following.

Let the client IP provided by operator to the MiFi modem at the node and seen by Tstat at the HTTP server side be indicated by IP_C and IP_S, respectively. Similarly, the client port at the node and HTTP server sides are denoted by $Port_C$ and $Port_S$, respectively.

In case of NAT, NAPT, or in presence of a PEP, $IP_C \neq IP_S$, and it becomes complicated to associate the flows seen in each single experiment (since we lose the information about the originating node). In this case, we associate the flow to the operator by resolving the IP_S address into its owner. We use the *MAXMIND* database [13], and, in case of a miss, we default to *whois* [14].

In more details, we match the *flow* associated with a certain experiment's TCP connection on the node side and HTTP server side if they start within a 1 second time window ($\hat{T}_{SYN_S} - \hat{T}_{SYN_C} < 1$ s), as follows:

1. If $IP_C = IP_S$ and $Port_C = Port_S$, we claim there is no NAT or PEP in the ISP network.
2. If $Port_C = Port_S$, $IP_C \neq IP_S$, and IP_C is a private IP address, we claim there is NAT in the ISP network. We can still associate each single flow by matching $Port_C$ to $Port_S$.
3. If $IP_C \neq IP_S$, $Port_C \neq Port_S$, we claim there is NAPT in the ISP network. We match the operator by looking at the IP_S as above.

Hence, we define a flow at the node and HTTP server sides when the connections start in a 1-second time window, have the same client IP address, the same server port number, and the same client port number (considering the port number is not changed by NAPT or PEP). If this is not possible, we simply assign data collected on the server side to the operator (but we cannot match the single flows). Our analysis shows that the first case can cover most of the operators.

C. $\widehat{G}$ mismatch

Given the i-th flow, let $\widehat{G}_C(i)$ and $\widehat{G}_S(i)$ be the goodput recorded by Tstat at the node and HTTP server, respectively. By comparing the observed values, we can show the existence of a PEP in the ISP network:

- $\widehat{G}_C(i) \sim \widehat{G}_S(i)$, illustrates the node experiences almost the same goodput as seen on the HTTP server. In this case, no PEP is present.[4]
- $\widehat{G}_C(i) < \widehat{G}_S(i)$, shows a *mismatch*. In this case, there is a PEP able to download the file from the server with considerably higher $\widehat{G}$ than the capacity on the path from the PEP to the client.

In case we cannot match the single flows, we can still compare statistics of $\{\widehat{G}_C(i)\}$ and $\{\widehat{G}_S(i)\}$ for all flows seen for a given operator.

V. Results

In this section we present the results obtained with the experiment setup described in the previous section.

A. Download goodput

As a first observation, Fig. 4 reports the goodput observed on three of the considered operators during a week, each point presenting the average $\widehat{G}_C$ of a set of experiments in a window of 1000 seconds, i.e., averaging all $\widehat{G}_C$ measurements for that operator during each run every 3 hours. This figure explains the complexity of speedtest-like experiments in MBB networks. Indeed, we observe quite different behaviors, such as i) a daily pattern (op0 in Spain), ii) a change of behavior over time (op2 in Sweden - see the last two days), iii) or unpredictable high variations (op1 in Italy). To check the impact of the duration of the test, and observe the fine grained variability of the capacity measurement, we also report the evolution over time of the download rate measured at the client, every second. Fig. 5 shows 2 runs, during which the client downloaded a 1 GB file in no more than 100 s. We observe a large variability, even during a relatively short test. This partly explains the variability observed in Fig. 4.

Fig. 6 shows the big picture of the client-side goodput observed over the eleven networks we tested in four European countries: Italy, Spain, Sweden, and Norway. Results report the ECDF of the client-side goodput computed from Tstat logs collected in our experiments. The x-axis in each chart of Fig. 6 gives the goodput ($\widehat{G}_C$) in Mb/s and the y-axis gives the probability of the goodput being less than the x-axis value. Variability is evident, and confirms the unpredictability seen in Fig. 1. Yet, some significant differences exist when comparing operators.

[4]We do not consider exact equality because some packets are in flight, and delay would make $\widehat{G}_S(i) > \widehat{G}_C(i)$ in general.

(a) Italy (b) Spain

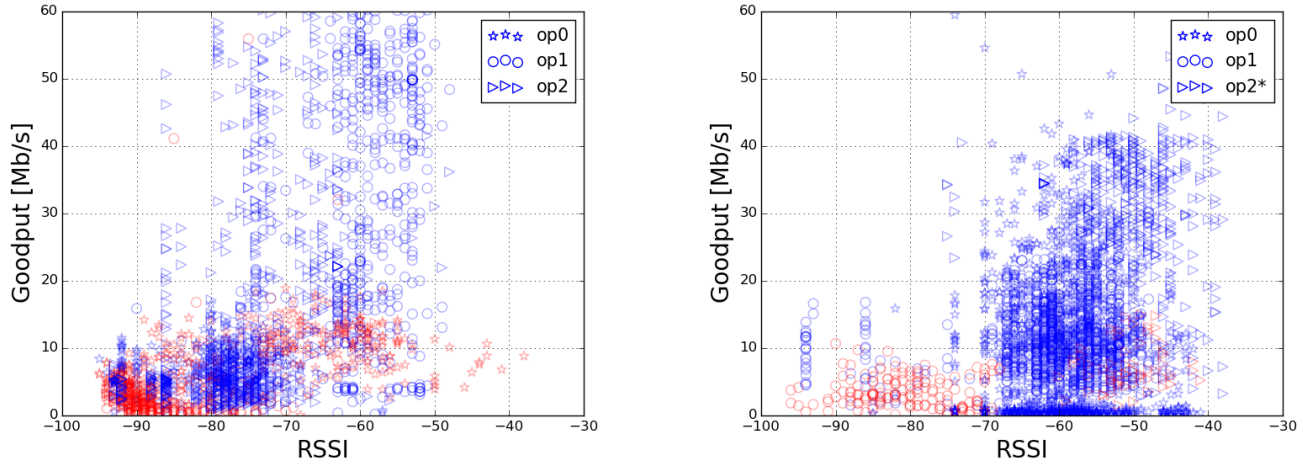

Fig. 7: RSSI and download client-side goodput for Italy and Spain. Blue and red markers indicate 4G and 3G, respectively. Pearson's correlation coefficients for Italy op0, op1, and op2 are 0.47, 0.61, and 0.50, respectively. Pearson's correlation coefficients for Spain op0, op1, and op2 are -0.008, 0.37, and -0.02, respectively

(a) Italy (b) Spain

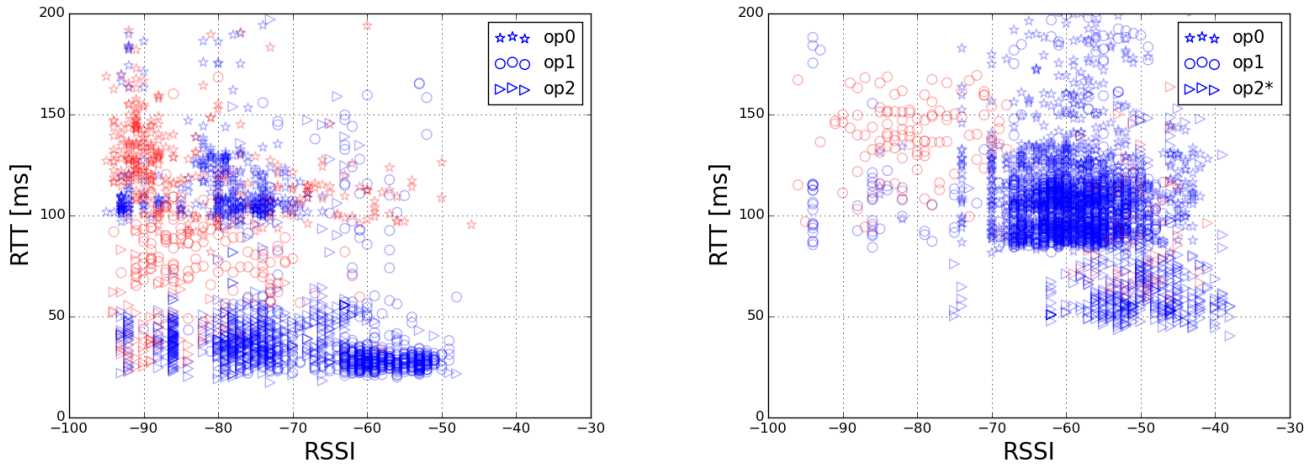

Fig. 8: RSSI and RTT for Italy and Spain. Blue and red markers indicate 4G and 3G, respectively. Pearson's correlation coefficients for Italy op0, op1, and op2 are 0.03, -0.49, and 0.39, respectively. Pearson's correlation coefficients for Spain op0, op1, and op2 are -0.009, -0.33, and -0.03, respectively

In Fig. 6d, we see that the two operators we considered in Norway provide similar values of the client-side goodput $\widehat{G}_C$.

On the contrary, the three operators that were measured in Italy gave quite different goodput results. In particular, op0 had a significantly high probability of providing low values of the client-side goodput $\widehat{G}_C$, in comparison to the other two operators. By looking at Fig. 8a, that we will discuss in detail later on, the red color of dots of op0 indicate that op0 mostly uses the 3G technology, and is configured so as to have higher RTT with respect to the other two operators. This explains the lower goodput values for op0.

In the case of Spain, we see that op0 in about 40% of the cases provided quite low values of the $\widehat{G}_C$. Our dataset indicates that, during peak times, the goodput provided by this operator is low, as can be seen in Fig. 4. We can clearly see that $\widehat{G}_C$ for op0 in Spain exhibits a daily pattern, probably due to throttling in periods of peak traffic. In addition, also by looking at the set of blue squares at the bottom of Fig.7b we observe a high percentage of low goodput experiments.

Fig. 7 plots for each experiment the values of $\widehat{G}_C$ on the x-axis, and the values of the RSSI on the y-axis. A first visual inspection indicates that the correlation between the RSSI and $\widehat{G}_C$ values is weak. Using Pearson's correlation coefficient [15] to quantitatively corroborate our impression, we obtain values up to 0.37 for Spain and up to 0.61 in Italy (the correlation coefficient takes values in the range [-1,1], with 1,-1, and 0 representing total positive correlation, total negative correlation, and no correlation, respectively). As generally expected, 4G (blue points) frequently outperforms 3G (red points), with some exceptions, which can be explained with the fact that RSSI it is not the only factor determining goodput in a mobile environment.

In Fig. 8 we plot for each experiment the average RTT value on the Y-axis, and the RSSI value on the x-axis. Interestingly, from Fig 8a, in the case of Italy we can observe two main intervals for RTT values, due to the fact that both op1 and op2

networks are configured so that RTT is mostly less than 50 ms, while op0 provides RTT values in the range of 100 ms. This can be the result of different network configuration choices. In the case of Spain, Fig 8b shows that op2*, largely using 4G technology, offers values of RTT in the range of 50 ms, which are lower than with other operators. Surprisingly, op2* in Spain is a roaming operator, that offers better performance with respect to the local operators.

B. Middle box detection

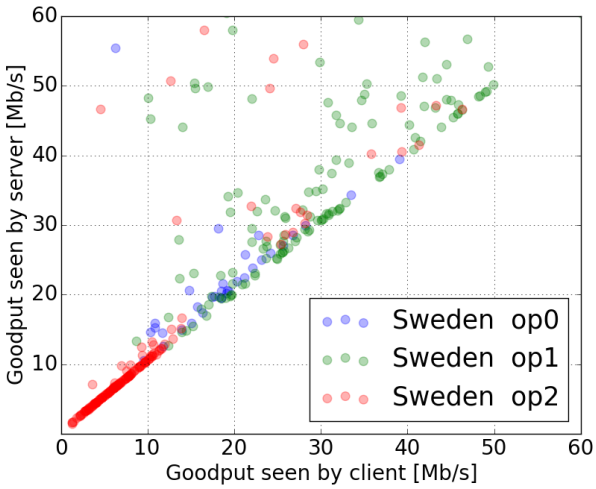

Fig. 9: Goodput experienced from client and server sides on Sweden operators

Fig. 9 shows the goodput in Mb/s experienced from the client-side (x-axis) and the server-side (y-axis), when $IP_C = IP_S$ and $Port_C = Port_S$ for operators in Sweden. If no PEP is present in the operator network, all points are expected to gather along the line $x = y$ in which $\widehat{G}_C \sim \widehat{G}_S$. While we see many points along this line, we also observe points where $\widehat{G}_C < \widehat{G}_S$, indicating the presence of a proxy. This is not surprising, since the use of PEP is becoming a common practice for mobile operators trying to improve end-users' Quality-of-Experience [16], [17], [18].

The MONROE platform allows us to gather detailed information about the operational state of the MBB networks in different countries. For example, we see that the operational setting of the Sweden operators are not static, and change over time. Indeed, the traffic of op2 in Sweden in some time periods crosses a PEP and in some others does not. Fig. 10 presents the server-side and client-side goodputs for this operator in the week when the traffic of op2 mostly crosses the PEP. The dashed line (server-side goodput) is often higher that the solid line (client-side goodput), but not always.

The volume of roaming traffic has been steadily increasing in Europe, and will increase even more after the reduction of the roaming surcharges, due to take place in June 2017. Operators have already started offering reduced tariffs for roaming, and exploiting international roaming agreements. In order to look at this aspect of MBB network performance, we considered op2* in Spain, which is the roaming network

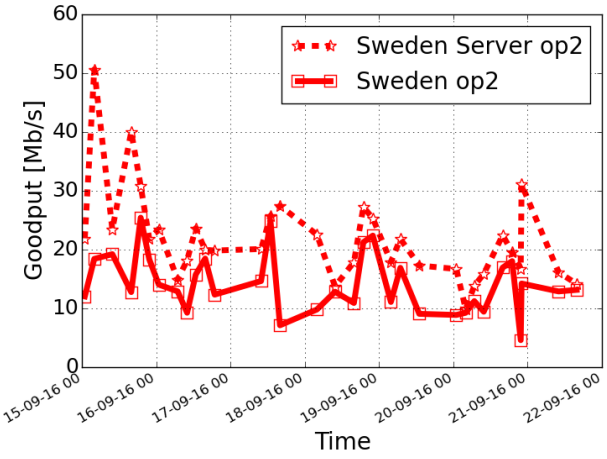

Fig. 10: Goodput experienced from client and server sides for op2 in Sweden during one week

for op2 in Italy. In other words, op2* in Spain is an Italian SIM used in roaming conditions in Spain. Quite surprisingly, Fig. 11 shows that the roaming SIM (op2* in Spain) obtains higher goodput than the corresponding SIMs at home (op2 in Italy), and that a PEP is in use in both cases.

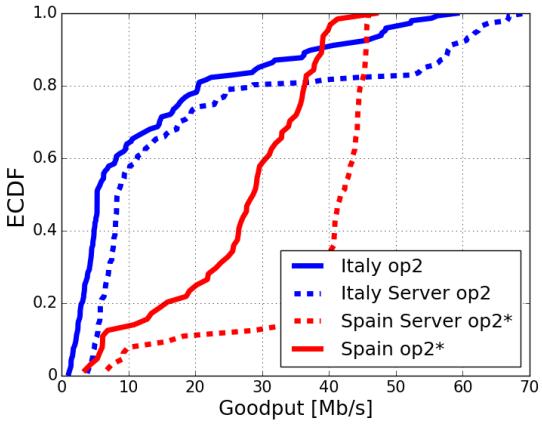

Fig. 11: Goodput experienced from the client and the server sides for the same operator SIM in Italy and Spain

Fig. 12 shows the values of the maximum segment size (MSS) and window scaling (WS) declared by the client to the server on port 80. The MONROE platform provides an equal setting at all clients with the default values of 1460 Bytes and 7 for MSS and WS. For visibility, the values in Fig. 12 are uniformly distributed around the observed value. Fig. 12a shows that Italian operators modify the client-declared TCP options. In order to see this, it is necessary to check more than one option, since, for instance, op1 does not change the MSS value, but changes the WS value. For other operators, the behavior varies. In Spain, both operators keep the WS value, but reduce the MSS value to 1400. In Sweden, operators again keep the WS value, but change the MSS to different values. In Norway, operators always change the MSS value, and sometimes also the WS value.

Finally, Table II shows a summary of the characteristics

(a) Italy (b) Spain (c) Sweden (d) Norway

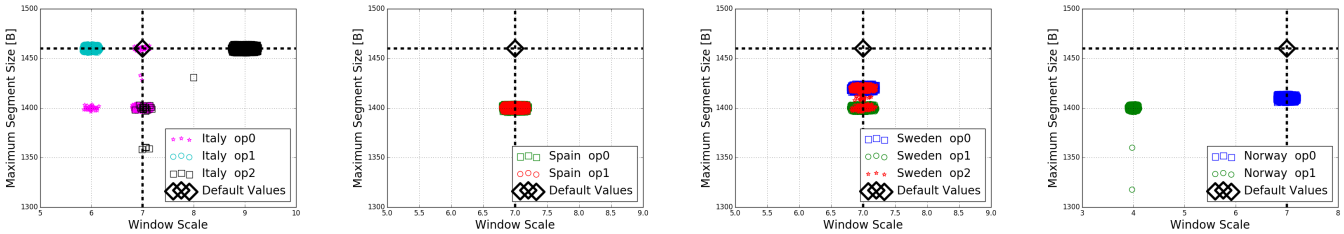

Fig. 12: WS and MSS values experienced at the server side on port 80, default values of MONROE nodes are 7 and 1460 Bytes, respectively

TABLE II: Summary of the operator settings

Country	Operator	Private IP & NAT	$\widehat{G}$ mismatch on port 80	L4 Mangling	Connection (percentage) Type
Italy	op0	Yes	Yes*	All	3G (0.46), 4G (0.54)
	op1	Yes*	Yes	All	3G (0.15), 4G (0.85)
	op2	No	Yes	All	2G (<0.01), 3G (0.08), 4G (0.92)
Sweden	op0	Yes*	Yes*	All	4G (100)
	op1	No	Yes	All	3G (<0.01), 4G (0.99)
	op2	No	Yes*	All	3G (0.37), 4G (0.63)
Spain	op0	Yes	No	All	4G (100)
	op1	Yes	No	All	3G (0.16), 4G (0.84)
	op2*	No	Yes	All	3G (0.07), 4G (0.93)
Norway	op0	No	Yes*	All	4G (100)
	op1	Yes*	Yes*	All	3G (0.08), 4G (0.92)

observed on the 11 European operators. The third column of the table indicates the usage of the NAT in the operator network. We see for example that in Italy op0 is always using NAT (Yes), while op1 sometimes uses it (Yes*), and op2 never uses it (No). Column 4 tells us that most of the operators use a PEP on port 80. The fifth column tells us that all operators do L4 mangling on all tested ports. Column 6 gives the fractions of observed 2G, 3G and 4G connections.

VI. Related work

The analysis of MBB network performance, and its prediction are on the research agenda of the networking community. There are mainly three approaches for measuring the performance of MBB networks: (i) crowd-sourced results from a large number of MBB users [19], [20] , (ii) measurements based on network-side data such as [21], [22], [23], and (iii) measurements collected using a dedicated infrastructure [24], [25], [6]. Network-side and active tests can be combined in the so-called "hybrid measurements" approach, as implemented, e.g., in [26]. In this paper, we collect data from a dedicated infrastructure in order to have full control over the measurement nodes, allowing us to systematically collect a rich and high quality dataset over a long period of time.

In the literature, some studies take it one step further and focus on the mobile infrastructure (e.g., presence of middleboxes) and its impact on performance. Performance enhancing middleboxes are widely deployed in the Internet and it is of great interest to measure and characterize the behavior of them especially in MBB networks where the resources are scarce. The impact of middleboxes on measurements was explored in [27] where the authors proposed a methodology for measurements in MBB networks. Farkas et al. [18] used numerical simulations to quantify the performance improvements of proxies in LTE networks. In [23], the authors analyzed LTE data collected in one city, to study the impact of protocol and application behaviors on network performance, mostly focusing on the utilization of TCP. Becker et al. [28] worked on analysis of application-level performance of LTE, and detected middle-boxes deployed on LTE networks, studying their impact on the measured performance. The most thorough analysis to characterize the behavior and performance impact of deployed proxies on MBB networks was carried out in [29] where the authors enumerate the detailed TCP-level behavior of MBB proxies for various network conditions and Web workloads. Although the common belief is that proxies provide performance benefits, Hui et al. [30] showed that they can actually hurt performance by revealing that direct server-client connections have lower retransmission rates and higher throughput. Wang et al. [31] showed how MBB middlebox settings can impact mobile device energy usage and how middleboxes can be used to attack or deny service to mobile devices. Taking a different route, Kaup et al. [32] studied the root causes of MBB network performance variability by means of measurements in one country, and showed that management and configuration decisions have a considerable impact on performance. We differentiate our work from these studies by focusing on different countries and operators. Furthermore, these studies consider a snapshot of the experiments which bound results to the measured ISP network and to the geographical location of the setup. On the contrary, our approach and experiments, by using the MONROE platform, allowed us to collect data through continuous experiments

over 4 countries and 11 operators. Our goal is to understand the mobile ecosystem and whether a simple speedtest can be run reliably over the current complex mobile networks, rather than measuring the performance of the mobile networks or the impact of middleboxes.

In closing, we remark that even performance measurements in wired networks can be a fairly complex task, because of user preferences, of the influence of users' home networks, of ISP traffic shaping policies, as noted by Sundaresan et al. in [33], who studied the performance of wired networks observed from home gateway devices, and observed counter-intuitive results.

VII. Conclusions

In this paper we discussed our experience in running "speedtest-like" measurements to estimate the download speed offered by actual 3G/4G networks. Our experiments were permitted by the availability of the MONROE open platform, with hundreds of multihomed nodes scattered in four different countries, and explicitly designed with the goal of providing hardware and software solutions to run large scale experiments in MBB networks. Our data were collected in 4 countries, over 11 operators, from about 50 nodes for more than 2 months.

Despite their simplicity, download speed measurements in MBB networks are much more complex than in wired networks, because of many factors which clutter the picture. The analysis of the results we obtained indicated how complex it is to draw conclusions, even from an extended and sophisticated measurement campaign.

As a result, the key conclusion of our work is that benchmarks for the performance assessment of MBB networks are badly needed, in order to avoid simplistic, superficial, wrong, or even biased studies, which are difficult to prove false.

Defining benchmarks that can provide reliable results is not easy, and requires preliminary investigation and experience, both being now possible thanks to the availability of an extensive Europe-wide platform like MONROE.

Acknowledgements

This work was funded by the European Union's Horizon 2020 research and innovation program under grant agreement No. 644399 (MONROE).

References

[1] J. Border, M. Kojo, J. Griner, G. Montenegro, and Z. Shelby, "Performance enhancing proxies intended to mitigate link-related degradations," 2001.

[2] OOKLA, "http://www.speedtest.net/,"

[3] C. Kreibich, N. Weaver, B. Nechaev, and V. Paxson, "Netalyzr: illuminating the edge network," in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pp. 246–259, ACM, 2010.

[4] FCC, "2013 Measuring Broadband America February Report," tech. rep., FCC's Office of Engineering and Technology and Consumer and Governmental Affairs Bureau, 2013.

[5] Tektronix, "Reduce Drive Test Costs and Increase Effectiveness of 3G Network Optimization," tech. rep., Tektronix Communications, 2009.

[6] O. Alay, A. Lutu, R. García, M. Peón-Quiròs, V. Mancuso, T. Hirsch, T. Dely, J. Werme, K. Evensen, A. Hansen, S. Alfredsson, J. Karlsson, A. Brunstrom, A. S. Khatouni, M. Mellia, M. A. Marsan, R. Monno, and H. Lonsethagen, "Measuring and assessing mobile broadband networks with monroe," in *2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 1–3, June 2016.

[7] "https://opensignal.com."

[8] "https://www.netztest.at/en/."

[9] DOCKER, "https://www.docker.com/,"

[10] ZTE, "Usb-based cat4 mf910 mifis, product specification:,"

[11] M. Trevisan, A. Finamore, M. Mellia, M. M. Munafo, and D. Rossi, "Traffic analysis with off-the-shelf hardware: Challenges and lessons learned," *IEEE Communications Magazine*, vol. 55, pp. 163–169, March 2017.

[12] B. B. J. V. Borman, D. and E. R. Scheffenegger, "Tcp extensions for high performance," 2014.

[13] MAXMIND, "https://www.maxmind.com/en/geoip2-isp-database,"

[14] WHOIS, "https://www.whois.net/,"

[15] J. Benesty, J. Chen, Y. Huang, and I. Cohen, *Pearson Correlation Coefficient*, pp. 1–4. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009.

[16] M. C. Necker, M. Scharf, and A. Weber, *Performance of Different Proxy Concepts in UMTS Networks*, pp. 36–51. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005.

[17] M. Ivanovich, P. W. Bickerdike, and J. C. Li, "On tcp performance enhancing proxies in a wireless environment," *IEEE Communications Magazine*, vol. 46, pp. 76–83, September 2008.

[18] V. Farkas, B. Héder, and S. Nováczki, *A Split Connection TCP Proxy in LTE Networks*, pp. 263–274. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[19] "MobiPerf." `http://www.mobiperf.com`, 2014.

[20] A. Nikravesh, D. R. Choffnes, E. Katz-Bassett, Z. M. Mao, and M. Welsh, "Mobile Network Performance from User Devices: A Longitudinal, Multidimensional Analysis," in *Procs. of PAM*, 2014.

[21] E. Halepovic, J. Pang, and O. Spatscheck, "Can you GET me now?: Estimating the time-to-first-byte of HTTP transactions with passive measurements.," in *Proc. of IMC*, 2012.

[22] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, S. Venkataraman, and J. Wang, "A first look at cellular network performance during crowded events," in *Proc. of SIGMETRICS*, 2013.

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

[24] Z. Koradia, G. Mannava, A. Raman, G. Aggarwal, V. Ribeiro, A. Seth, S. Ardon, A. Mahanti, and S. Triukose, "First Impressions on the State of Cellular Data Connectivity in India," in *Procs. of ACM DEV-4*, ACM DEV-4 '13, 2013.

[25] D. Baltrūnas, A. Elmokashfi, and A. Kvalbein, "Measuring the Reliability of Mobile Broadband Networks," in *Proc. of IMC*, 2014.

[26] M. Laner, P. Svoboda, P. Romirer-Maierhofer, N. Nikaein, F. Ricciato, and M. Rupp, "A comparison between one-way delays in operating hspa and lte networks," in *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2012 10th International Symposium on*, pp. 286–292, IEEE, 2012.

[27] A. Botta and A. Pescapé, "Monitoring and measuring wireless network performance in the presence of middleboxes," in *Wireless On-Demand Network Systems and Services (WONS), 2011 Eighth International Conference on*, pp. 146–149, IEEE, 2011.

[28] N. Becker, A. Rizk, and M. Fidler, "A measurement study on the application-level performance of lte," in *2014 IFIP Networking Conference*, pp. 1–9, June 2014.

[29] X. Xu, Y. Jiang, T. Flach, E. Katz-Bassett, D. Choffnes, and R. Govindan, "Investigating Transparent Web Proxies in Cellular Networks," in *Proc. of Passive and Active Measurement*, 2015.

[30] J. Hui, K. Lau, A. Jain, A. Terzis, and J. Smith, "How YouTube Performance is Improved in T-Mobile Network," in *Proc. of Velocity*, 2014.

[31] Z. Wang, Z. Qian, Q. Xu, Z. Mao, and M. Zhang, "An untold story of middleboxes in cellular networks," in *Proc. of SIGCOMM*, 2011.

[32] F. Kaup, F. Michelinakis, N. Bui, J. Widmer, K. Wac, and D. Hausheer, "Assessing the implications of cellular network performance on mobile content access," *IEEE Transactions on Network and Service Management*, vol. 13, pp. 168–180, June 2016.

[33] S. Sundaresan, W. de Donato, N. Feamster, R. Teixeira, S. Crawford, and A. Pescapè, "Broadband internet performance: A view from the gateway," *SIGCOMM Comput. Commun. Rev.*, vol. 41, pp. 134–145, Aug. 2011.