A Measurement Study on the Application-level Performance of LTE

Nico Becker, Amr Rizk, and Markus Fidler
Institute of Communications Technology
Leibniz Universität Hannover
{nico.becker, amr.rizk, markus.fidler}@ikt.uni-hannover.de

***Abstract*—Many of today's Internet applications such as mobile web browsing and live video streaming are delay and throughput sensitive. In face of the great success of cellular networking, especially with the advent of the high-speed LTE access technology, it is noteworthy that there is little consensus on the performance experienced by applications running over cellular networks. In this paper we present application-level performance results measured in a major commercial LTE network. We replicate measurements in a wired access network to provide a reference for the wireless results. We investigate the performance of common web application scenarios over LTE. In addition, we deploy controlled measurement nodes to discover transparent middleboxes in the LTE network. The introduction of middleboxes to LTE results in a faster connection establishment on the client side and notable performance gain for HTTP. However, this improvement comes at the price of ambiguity as some middlebox operations may introduce unnecessary timeouts. Further, we pinpoint LTE specific delays that arise from network signalling, energy saving algorithms and Hybrid Automatic Repeat reQuest (HARQ). Our analysis provides insights into the interaction between transport protocols and LTE.**

***Index Terms*—LTE, 4G, performance evaluation.**

I. Introduction

The evolution of mobile networking in the past few years has been driven by an ever increasing growth of mobile data traffic. This growth was substantiated by the introduction of convenient and powerful smartphones and tablets as well as flat rates for unlimited data amounts. In the sequel we refer to smartphones and tablets as user equipment (UE). To handle the imminent user demand for higher data rates as well as to support delay-sensitive mobile applications, 3GPP initiated a study [2] on the requirements of the long term evolution (LTE) of UMTS, i.e., the worldwide leading 3G system. The main targets were to increase the spectral efficiency, hence, the data rate and to decrease the latency for packets traveling through the mobile network core. 3G systems were known to have multiple 100 ms latency, which is not tolerable for voice communications [10]. Thus, mobile operators of 3G systems had two core networks, a circuit switched and a packet switched network for voice and data, respectively. The cost reduction through the core network aggregation and the deployment of voice over IP (VoIP) techniques is a vital aspect that motivated network operators to adopt LTE.

This work is supported by an ERC Starting Grant (UnIQue).

ISBN 978-3-901882-58-6

The rise of mobile applications that are heavily dependent on networking posed constraints on two performance metrics, i.e., data rate and latency. Modern UEs generate a data traffic mix that contains delay-sensitive real-time VoIP and video traffic, HTTP web browsing traffic, bulk file transfers such as music downloads and cloud services, as well as periodic refresh messages, e.g., for news applications. The theoretical design of LTE was required to provide data rates up to 100 Mbps in the downlink and 50 Mbps in the uplink [2]. Additionally, the design requires the possibility for radio-access network latency below 10 ms. However, assessing the performance gain for network applications running over LTE is a challenging task, as, for example, the nature of the wireless channel and the interaction of traffic patterns with LTE internal algorithms impact the experienced performance. Undoubtedly, LTE introduces a performance boost when compared to 3G systems.

In this work, we contribute an application-level performance analysis of LTE. We use active measurements to show greedy UDP and TCP throughput and infer limiting data rates in uplink and downlink directions for a stationary UE. We demonstrate the disparity of uplink and downlink paths, for example, the difference of the buffer sizes in both directions. Further, we address LTE specific MAC layer algorithms that influence packet latency, i.e., discontinuous reception (DRX) and Hybrid Automatic Repeat reQuest (HARQ), respectively. Finally, we show results on middlebox discovery and operation in the considered commercial LTE network. We devise HTTP measurements that enable us to discover the operation of proxies and highlight the deployed NAT policies. We compare results on the delay distributions for HTTP handshakes in LTE and wired access networks, respectively, to substantiate the performance improvement through the application of middleboxes in LTE.

The rest of this paper is structured as follows. In Sect. II we discuss related work on performance measurements in LTE and 3G networks. Sect. III covers the measurement setup providing details on the deployed software and hardware. Sect. IV presents measurement results for different transport layer protocols. In Sect. V we analyze the impact of LTE MAC-layer specific algorithms on packet latency. Sect. VI provides performance results for HTTP measurements in LTE and in wired access networks. The measurement results show application-level quality of service, the operation of

middleboxes and a comparison with wired access network performance. In Sect. VII we discuss the vantage point from which the measurements are made before concluding the paper in Sect. VIII.

II. Related Work

In this work we investigate the application-level performance of LTE in production networks. The related work comprises LTE measurement studies that investigate key performance characteristics such as throughput, packet delays and the energy consumption of user equipment.

Most of the measurement based comparisons of throughput and packet delays in 3G and LTE systems show the performance leap between the cellular generations [8], [12], [13], [17], [27]. Field trials [27] and lab emulations [28] that investigate the impact of the signal-to-noise ratio (SNR) and the channel conditions on the physical layer throughput conclude that LTE can achieve the theoretical throughput with a negligible margin.

The reasons for the performance improvement in LTE are manifold. LTE provides a higher bit rate due to, e.g., the OFDM transmission technology, higher coding and modulation schemes and multiple antenna configuration (MIMO) [10]. The lower latency compared to 3G networks is often attributed to the flat all-IP backhaul network architecture in LTE networks [13], [27]. In contrast, the work in [16] concludes from delay measurements in 3G and LTE systems that LTE is not yet generally preferable with respect to delay sensitive applications. The authors of [16] conjecture that for specific traffic patterns 3G is favorable due to its radio access scheme.

Modern LTE core networks comprise middleboxes for security and performance reasons. The authors of [25] present a tool to reveal cellular network NAT and firewall policies. The work in [25] includes an investigation of port mapping, IP spoofing and enforcing timeout on idle TCP connections. The authors argue that the connection disruption due to timeout enforcing causes additional radio signalling, that amounts to a substantial share of the standard power consumption per day.

Battery drain at the UE is a performance indicator that gained increasing attention due to its impact on the user experience [11], [21], [22]. Diverse transmission technologies as well as MAC-layer techniques utilize the available UE power in different manners. For example, the OFDM technology, that enables high bit rates in LTE, suffers from low power efficiency due to a high peak-to-average power ratio. The authors of [12] compare the power consumption characteristics of 3G and LTE systems to conclude that LTE is significantly less power efficient. They develop a power consumption model based on MAC-layer transmission specification and obtain power efficiency results from simulations with replayed real world traces. Closely related is the standardized discontinuous reception algorithm that is specifically devised for LTE networks to save power at the UE [3], [4], [6]. In DRX the UE periodically turns off the power consuming radio interface until paged that packet transmission is imminent. The work in [26] adapts the DRX cycle parameters to optimize power consumption for certain scenarios while comparing throughput versus power consumption.

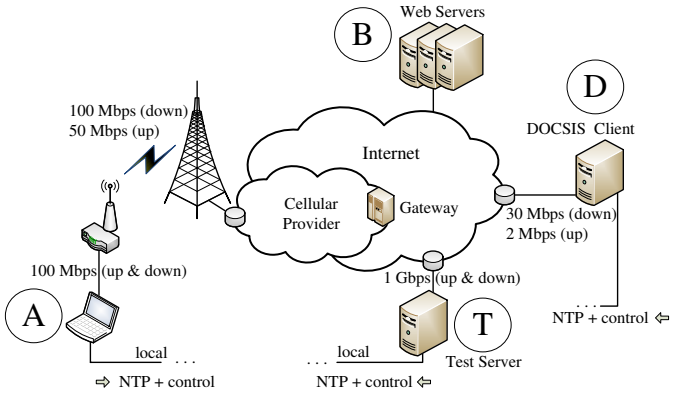


Fig. 1: Measurement topology for the LTE network.

The work in [13] reports on a massive real-world data collection campaign where the authors collected traffic in the core network of a US cellular network provider. The authors of [13] observe significantly lower uplink than downlink rates. Further, [13] uses a self-devised bandwidth estimation method to measure the bandwidth utilization of TCP in LTE. The authors find that TCP connections under-utilize the available bandwidth by 50% due to a limited TCP receive window and make a case for LTE-friendly transport protocols. In comparison, the work in [14] proposes, however, a dynamic throttling of the TCP receive window to encounter performance degradation due to large buffers in the mobile core network.

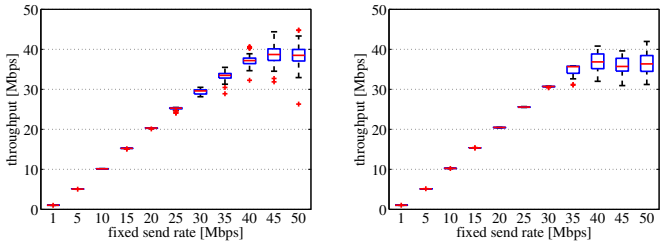
III. Measurement Setup

We perform measurements in a major German commercial LTE network and replicate relevant measurements in a wired DOCSIS access network to provide a reference for the LTE results. The nominal maximum downlink and uplink rates as stated by the LTE provider are 100 Mbps and 50 Mbps, respectively. Fig. 1 displays the main components of the measurement setup. We denote the relevant nodes, i.e., the LTE client, different world wide web servers, our test server and our wired client[1] as A, B, T, and D, respectively. In our measurements the LTE client (A) has a fixed position, hence we do not consider mobility. Also, we do not control the implementation of the LTE network components.

In the sequel, we analyze two different measurement settings, which we call the general and the controlled setting. In both settings client (A) is connected to the eNodeB, i.e., the LTE base station via a stationary category 3 LTE Teldat RS232j-4G modem [23]. Note that we have a separate control connection to client (A) via the local lab network.

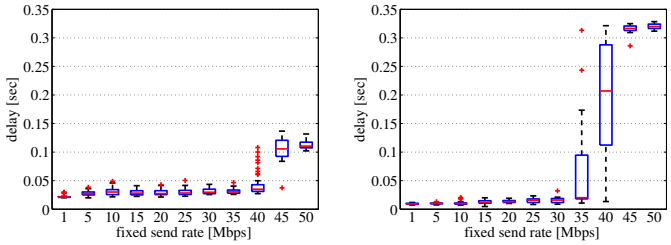
In the general setting we send HTTP requests from client (A) to distinct web servers (B) and capture the outgoing

[1]The server (T) runs Ubuntu 12.04 with kernel versions 3.5.0, while the clients (A) and (D) run Ubuntu 12.04 with kernel version 3.2.0.



(a) Uplink (b) Downlink

Fig. 2: Greedy throughput for different UDP traffic rates between client (A) and server (T).



(a) Uplink (b) Downlink

Fig. 3: Delay distributions for different UDP traffic rates between client (A) and server (T).

and incoming traffic using *libpcap*. In this setting we emulate the web surfing behavior of some LTE user.

In the controlled setting we perform measurements using various protocols between the client (A) and the test server (T), which resides at our lab. We replicate the measurements between the wired client (D) and the test server (T) to obtain wired reference results. In both cases we control both the client and the server and capture the traffic at both using *libpcap*. Note that the server (T) and the client (D) have public IP addresses and that our test server (T) is connected to the X-win network, i.e., the German national research network at a rate of 1 Gbps. We use the controlled setting, for example, for throughput and delay measurements, in which we analyze the send and receive timestamps on the client and the server side. Furthermore, controlling both ends of the connection enables detecting packet alterations that may arise due to transparent middleboxes.

To ensure reliable timestamps we employ time synchronization between the clients (A), (D) and the server (T) using the Network Time Protocol (NTP). We achieve a synchronization granularity of less than 1 ms with respect to a local NTP server. We automatize our measurements using the software *SSHLauncher* [7], which enables executing distributed network experiments remotely by using control connections on separate network interfaces (marked local in Fig. 1).

IV. Baseline Measurements

In this section, we provide measurement results on basic performance metrics in the considered LTE network, i.e., throughput, packet delay, and loss. Further, we show details on buffer sizes in uplink and downlink directions and highlight the increase in traffic burstiness over LTE and the occurrence of burst losses.

A. Maximum Throughput Measurements

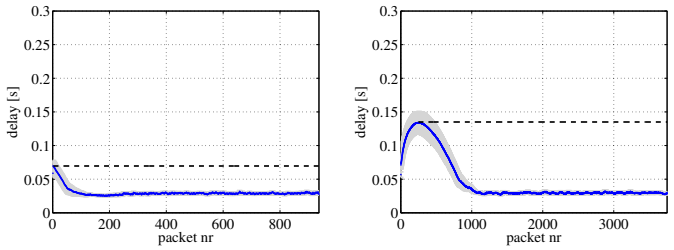
We perform our measurements in an LTE network with nominal rates up to 100 Mbps and 50 Mbps in downlink and uplink, respectively. These data rates are, however, calculated at the physical layer for idealized conditions [10]. In the following, we use greedy constant bit rate (CBR) UDP sources to measure the maximum data rates that an application perceives over LTE. We generate UDP traffic between the client (A) and the server (T) using the software *rude&crude* [15] and capture the data packets at both ends using *libpcap*. In a first measurement, we send UDP test packets from the client (A) in uplink direction to server (T). Examining the IP header information at the server reveals that the mobile network gateway performs network address translation (NAT). Hence, for downlink measurements we perform NAT *hole punching*. In essence, we send a test packet in uplink direction to the server, from which we extract the external IP address and port number. Thereupon, we send the downlink packets from (T) to (A) to the extracted IP / port combination.

Fig. 2a shows boxplots of the uplink throughput at server (T) versus the sending rate at the client (A) for CBR UDP traffic with a fixed packet size of 1400 Bytes. The median is given by the central mark and the borders of the box are the 0.25 and 0.75 percentiles. The lower and upper whiskers denote the range of data points that are not considered outliers, based on the 0.99 coverage of the Gaussian distribution. Outliers are marked individually. Each boxplot is calculated for 50 independent runs of two seconds per run. From Fig. 2a we find a limiting rate for the uplink of 40 Mbps. Interestingly, we observe approximately the same limiting rate for greedy UDP downlink measurements as shown in Fig. 2b. Averaged over 2 seconds, LTE provides stable data rates below its limiting rate in both directions, i.e., with nearly no variation. This is comparable to the stability of the uplink and downlink rates observed in wired access measurements between client (D) and server (T). For sending rates which are close or even larger than the limiting rate the achieved uplink and downlink rates fluctuate in a range of about 5 Mbps.

B. Packet Delays, Losses and Network Buffers in LTE

Next, we investigate buffering, packet delays, and losses in the LTE network using the same traffic as in Sect.IV-A. First, we analyze one way delay distributions for different sending rates in uplink and downlink directions.

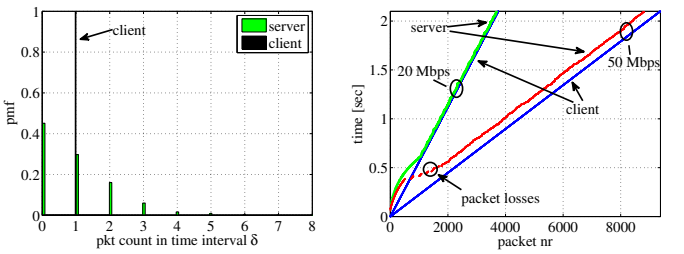
Figure 3a shows that the delay distributions for the uplink direction remains around a median of 30 ms for rates below the limiting rate of 40 Mbps. Higher rates cause a significant jump to about 110 ms in addition to packet losses. We estimate the maximum queue length to correspond to 80 ms in uplink direction. For an uplink rate of 40 Mbps and packets of



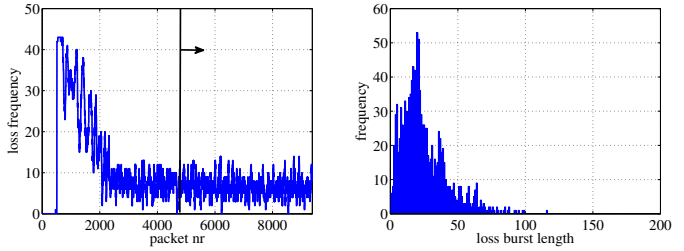
(a) 5 Mbps (b) 20 Mbps

(c) 40 Mbps (d) 50 Mbps

Fig. 4: One way delay distributions for different arrival rates in uplink direction. All measurements are 2 seconds long.



(a) Burstiness increase. (b) Sample paths of CBR traffic.



(c) Loss frequency. (d) Loss burst length distribution.

Fig. 5

1400 Bytes this dwell time would amount to a buffer size of around 280 packets. Comparing Fig. 3a to Fig. 3b, we see the discrepancy between the uplink and downlink in LTE. First, the one way delays differ strongly in downlink with a median of 10 ms. Second, the buffers in downlink direction correspond to 310 ms. This would amount to 1100 packets at the downlink limiting rate of approximately 40 Mbps and packets of 1400 Bytes. The existence of large buffers in LTE networks has pros and cons. On the one hand, it is reasonable to have a large buffer to deal with bad wireless channel conditions. On the other hand, oversized buffers may lead to severe performance degradation. For example, consider a user generating a traffic mix of delay-sensitive video chat data and large file transfers in parallel over this LTE network. In this case, it is highly probable that the user experiences sluggish performance of the video call.

Next, Fig. 4 shows one way delays for CBR packet indexes at different sending rates in uplink direction. We perform 50 independent runs and plot the average delay including shaded confidence intervals per packet number. We set all measurement durations to 2 seconds, thus the different number of packets on the abscissa. All measurement runs start from a dormant state, i.e., the UE has to establish a connection first before being able to send the first data packet. In all figures we observe a one way delay of the first packet of nearly 70 ms. With an increasing send rate in Fig. 4a to Fig. 4d we observe much higher delays during the connection establishment (horizontal dashed line) than the final stable delay. The oscillations of the mean values in Fig. 4c and 4d arise for data rates higher or equal to the limiting rate due to packet losses.

We further inspect the burstiness increase in traffic traversing LTE networks. Remember that we send CBR traffic from the client (A) to the server (T) with a given rate for 2 seconds. We denote the fixed inter-packet gap δ, which is given by the packet size divided by the send rate. In Fig. 5a we plot the empirical probability mass function (pmf) of the packets at client (A) for a fixed send rate of 40 Mbps and packet size of 1400 Bytes. The peak denotes the rigid structure of the send traffic as in each of the intervals δ there resides only one packet. However, at the server (T) we observe up to 6 packets in an interval of δ. A possible reason for this effect may be that radio blocks have to wait in the receive buffer until corrupted predecessor blocks are correctly received.

Next, we examine the service provided by the eNodeB to the UE. In Fig. 5b we show sample paths of sent and received uplink traffic for two different send rates of 20 and 50 Mbps, respectively. From the server timestamps we observe a high delay in the beginning of the trace due to the time for connection establishment in which the UE receives no service.

In the following, we turn our attention to packet losses. In Fig. 5b, we observe packet losses throughout the entire trace for the send rate of 50 Mbps, which exceeds the limiting rate. In particular, Fig. 5b depicts heavy packet losses at connection establishment that are probably due to the time with no or little service at the beginning of the trace. We also observe a clustering of packet losses throughout the traces. Figure 5c shows how frequent a packet of specific index is lost across 50 runs. Here the deterministic behavior at the beginning of the trace is apparent. If we only take the second half of the trace as illustrated in Fig. 5c, we find that most packet losses are burst losses as depicted in Fig. 5d. This may be attributed to the nature of the wireless channel and its temporal correlations.

C. TCP Throughput Measurements

In this subsection we report on TCP throughput measurements for LTE. Compared to UDP, the congestion control of

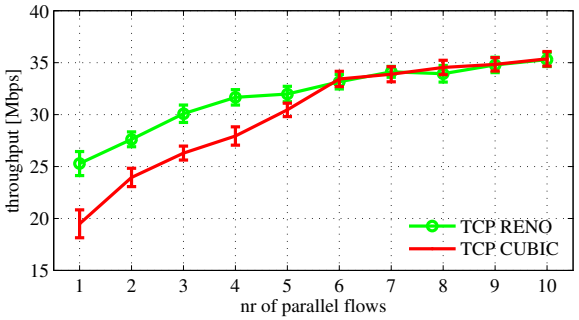


Fig. 6: Aggregate uplink TCP throughput for different congestion control algorithms and concurrent flows. Multiplexed flows make better use of the available bandwidth.

TCP may reduce the achievable data rate. We generate TCP traffic between client (A) and server (T) using the software *iperf* [1]. We perform measurements to infer the aggregate throughput with concurrent TCP flows in uplink and downlink directions. In the following, we show the average throughput over 35 runs each of 5 seconds duration. We omit the first second of the traces in the throughput calculation to avoid the slow start phase. We perform the measurements using both TCP congestion control algorithms Cubic and Reno. Note that TCP Cubic is the default congestion control algorithm in Linux starting from kernel 2.6.13.

For the downlink direction we observe a stable throughput around 30 Mbps with no significant difference between Cubic and Reno. We also find no evidence for a significantly different throughput when varying the number of concurrent flows. Fig. 6 shows that in uplink direction a user obtains a higher throughput from multiplexing TCP flows. In general, we do not observe a strong throughput difference between the algorithms Cubic and Reno.

In this section, we analyzed the interaction of different transport protocols with LTE. We highlighted different performance metrics such as throughput, packet delay, loss, and buffer sizes and revealed possible causes for the provided results.

V. Performance Implications of LTE MAC

A. Discontinuous Reception Mode

In the following, we investigate the impact of specific LTE MAC algorithms on packet latency. Specifically, we consider the LTE discontinuous reception mode, which aims at reducing the UE energy consumption [3], [4], [6]. In general, the UE monitors the Physical Downlink Control Channel (PDCCH) for incoming paging messages from the eNodeB. To save battery power the UE may enter DRX such that it monitors the PDCCH for control information only at certain time instances. In between, the UE turns off the radio circuitry to save power. This procedure is implemented in the so called DRX cycle [3], [6], which consists of a predefined repetition of a high power active time and a low power sleep mode. It is obvious that DRX introduces a trade-off between battery lifetime and

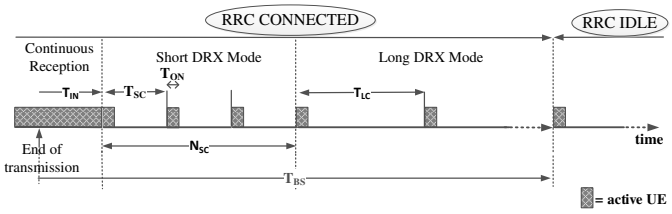


Fig. 7: A sample path of LTE DRX with RRC_CONNECTED and RRC_IDLE states.

packet delay. For example, during sleep mode the UE cannot be paged for incoming packets until it wakes up to monitor the PDCCH. Hence, a packet that is intended for a sleeping UE has to be buffered in between experiencing a higher delay.

In the following, we introduce the operation mode of DRX in LTE. Generally, a UE is in one of two radio resource control (RRC) states, i.e., RRC_IDLE or RRC_CONNECTED [6]. In RRC_IDLE the UE does not possess an RRC connection [6] and it monitors the PDCCH for paging information according to the DRX cycle. The default DRX cycle length in RRC_IDLE is specified by the eNodeB in the range of $0.32 - 2.56$ seconds [10]. For any data transmissions in up- or downlink direction the UE has to transit first to RRC_CONNECTED by establishing an RRC connection. This results into an exchange of 16 to 19 signalling messages in the mobile network core until data can be transmitted in uplink, respectively, downlink direction [10].

In RRC_CONNECTED the eNodeB provides the UE with the DRX parameters. An example of a UE DRX activity is given in Fig. 7. For each UE, the eNodeB possesses a so called user inactivity timer T_{BS} that is reset for each packet that is sent to, or received from the UE. The UE is internally in the so called "continuous reception" state as long as it sends or receives packets. When a packet transmission is finished the UE remains in the continuous reception state and starts the DRX inactivity timer T_{IN}. This timer is reset whenever a new packet is sent or received. First, the UE monitors the PDCCH until the expiry of T_{IN} which lies between 1 ms and 2.56 seconds [10]. If T_{IN} expires the UE starts the short DRX cycle of length T_{SC}, during which it monitors the PDCCH only for the duration T_{on}. After a predefined number of short DRX cycles N_{sc} with no activity, the UE changes to the long DRX cycle of length T_{LC}. If the timer T_{BS} expires at the eNodeB, the eNodeB initiates an RRC connection release to save resources. The eNodeB tears down its data connection to the UE which moves back to the RRC_IDLE state.

Next, we present measurement results showing the impact of DRX on packet delays. We measure packet round-trip times (RTT) for periodic ping packets from client (A) to the server (T). We vary the period length, i.e., the inter-packet gap and measure for each gap 5×10^3 corresponding RTTs. Figure 8 shows the increase in packet RTTs when the UE moves through the different DRX states. We plot the complementary cumulative distribution function (CCDF) of packet RTT for different inter-packet gaps. For packets with

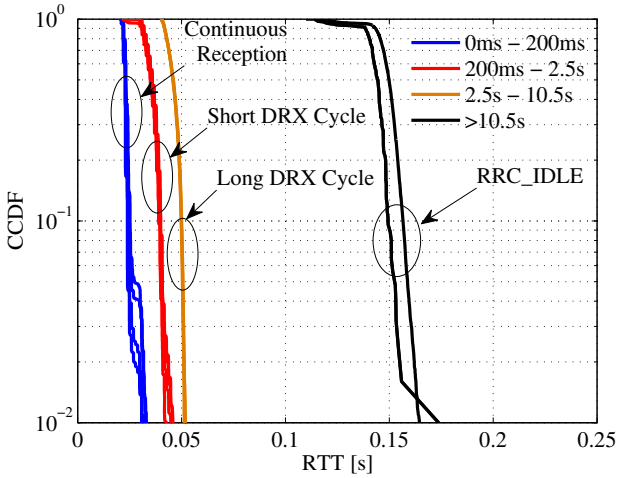


Fig. 8: CCDF of `ping` RTT for different inter-packet gaps.

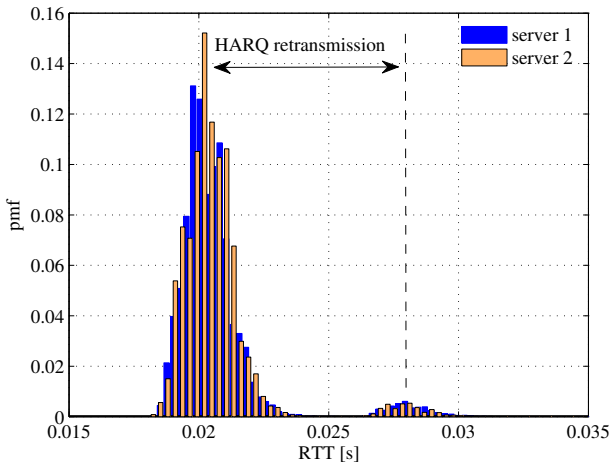


Fig. 9: PMF of the RTTs for TCP connection establishment handshakes.

an inter-packet gap that is less than 200 ms the UE stays in continuous reception. For inter-packet gaps between 200 ms and 2.5 s we observe a jump of the RTTs such that we infer that the UE switches from continuous reception into the short DRX mode. For an inter-packet gap of at least 2.5 s we observe a second jump of the RTT indicating that the UE entered long DRX. Finally, we infer the user inactivity timer of the eNodeB to be 10.5 s. Here, the RTT increases by 50 ms as the RRC connection has to be established first. Fig. 8 shows that for applications with periodic refresh messages, e.g., messaging and news applications, the choice of the refresh interval has a strong influence on both the battery lifetime and the perceived delay. Since a transition from `RRC_IDLE` to `RRC_CONNECTED` results into the exchange of at least 16 signalling messages in the network core it is apparent that applications with refresh intervals just above the user inactivity timer of the eNodeB may introduce an excessive load on the core network, which would impact the overall network performance [21]. Further, Fig. 8 helps to illustrate the difference between wired access networks and LTE with respect to a quality of service effect that we denote "instant browsing". Here, we observe that loading web pages of popular search engines over wired access networks requires less than 100 ms, which is perceived as *instant* interaction in human-machine interaction [18]. Fig. 8 shows that this interaction latency can hardly be achieved if the UE starts from `RRC_IDLE`.

B. MAC-layer Retransmissions

Next, we investigate the impact of the LTE MAC-layer data retransmission algorithm, i.e., Hybrid Automatic Repeat reQuest [4], [5] on packet delays. In a nutshell, HARQ detects erroneous data blocks through the included checksum and requests bad data blocks to be retransmitted. The receiver then combines the erroneous packet with the retransmitted copy to increase the likelihood of correct decoding. In the mean time, correctly received out-of-order blocks have to wait in the receiver buffer until predecessor blocks are retransmitted. In LTE a HARQ retransmission requires 8 ms [4].

We measure RTTs from the client (A) in continuous reception to two popular web servers (B). We measure 5×10^4 independent handshakes and plot the empirical pmf of the handshake duration in Fig. 9. We define a handshake by the time needed from sending a SYN packet to receiving the SYN/ACK reply. Typically a single handshake requires 20 ms. We find a characteristic second mode at about 28 ms that coincides with the additional 8 ms of a HARQ retransmission.

VI. Application Layer Interaction with LTE

In this section we characterize the interaction between application layer protocols and LTE. We focus on HTTP over TCP since it is currently the prevalent protocol in the Internet [20]. It serves a wide range of applications ranging from web browsing and web services to video streaming.

First, we perform measurements to assess the time needed to establish a connection using the TCP 3-way handshake over LTE. Further, we use HTTP HEAD requests to measure the *time to first Byte*, that is the time elapsed until the UE receives the first (useful) data Byte from the server. We replicate the measurements using the client (D) to provide a wired performance reference. Using the controlled setting we discover packet alteration in the mobile network that indicates the use of transparent middleboxes. Therein, we devise a measurement campaign to characterize the operation of the mobile network middlebox and its impact on the performance perceived by the UE.

A. HTTP/TCP Connection Establishment

Fig. 10 shows a sketch of the interaction between a client and a server for a typical HTTP HEAD request. For evaluation we use the points in time marked C {I, IIa, IIb, III} and S {I, II, III} in Fig. 10. The letters C and S denote the timestamps

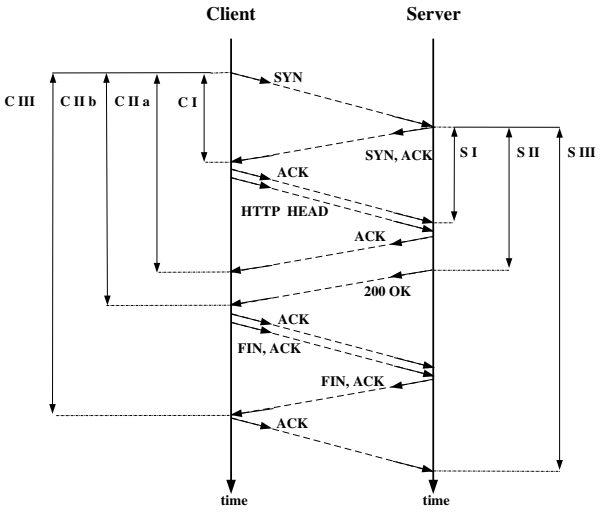


Fig. 10: The client/server interaction process for an HTTP HEAD request. The figure shows our timestamp evaluation points at the client (C I - C III) and the server (S I - S III).

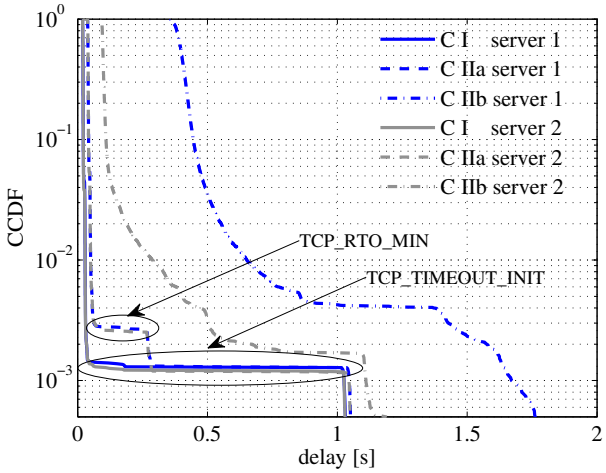


Fig. 11: Distributions for different measurement points (C I - C IIb) for HTTP HEAD requests to two different popular web servers. The requests originate from Client (A) over LTE.

which are taken at the client and (if possible) at the server, respectively. The timestamps are taken using *libpcap* upon packet arrival or departure according to the arrows in Fig. 10. The points C I and S I denote the time for the first handshake at the client, respectively, the server side. The point in time C IIa denotes the time to complete the second handshake at the client side. S II denotes the time until the first *useful* Byte is sent at the server, while C IIb denotes the *time to first Byte*, i.e., the time until the first Byte is received at the client. The points in time C III and S III denote the connection duration at the client and at the server, respectively.

We perform HTTP HEAD request measurements from the LTE client (A) and from the wired client (D). First, we send 5×10^4 HTTP HEAD requests from client (A) to popular web servers (B) and capture the traffic at the client. Fig. 11 shows

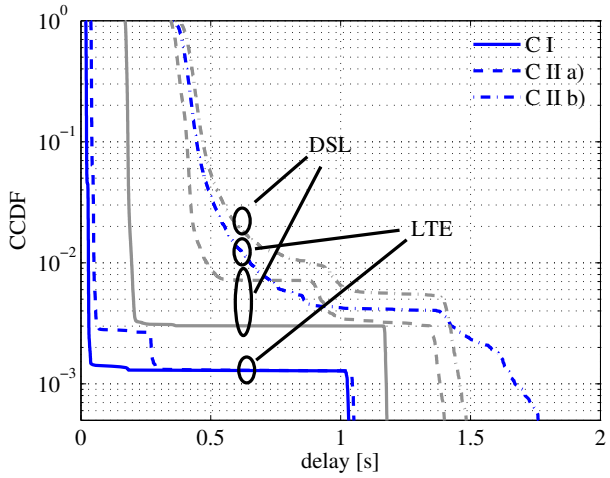


Fig. 12: Distributions for the client side measurement points of HTTP HEAD requests to server 1 from Client (A) over LTE vs. client (D) using wired access.

the CCDF of the delays C {I, IIa, IIb} for two different servers. For both servers we observe a remarkably identical behavior for the first two handshakes, i.e., C {I, IIa}. The points C {I, IIa} amount to 20 ms and 40 ms, respectively. We observe that with a probability around 10^{-3} a SYN or a SYN/ACK packet is lost such that we experience an extra delay of 1 second. This is due to the implementation of the initial TCP timeout in the Linux kernel to comply with RFC 6298 [19]. If a packet that belongs to the second handshake is lost we obtain an additional delay of 210 ms due to the TCP minimum retransmission timeout (RTO). Note that this value that is implemented in the Linux kernel does not comply with the minimum of 1 second that *should* be used [19]. Interestingly, we observe the first significant difference between the results for server 1 and server 2 at C IIb. In addition, the difference between C IIa and C IIb is remarkably large for both servers. In the following subsection, we will delve into the origin of this behavior.

B. Middlebox Discovery and Characterization

Next, we devise a measurement campaign to discover and characterize the operation of mobile network middleboxes. First, we replay the HTTP HEAD measurement campaign to the same servers (B) using the wired client (D). Figure 12 shows the CCDFs for the points C {I, IIa, IIb} for server 1 accessed from LTE, respectively from a wired network. Comparing LTE and the wired access, one observes that the gap between C IIa and C IIb is significantly different. Hence, we set up a dedicated web server in the controlled setting, i.e., test server (T) in Fig.1, and capture the data packets at both C {I, IIa, IIb, III} and S {I, II, III}. In addition, we insert a 250 ms artificial delay on each packet that is leaving the server (T) using `netem` [24]. First, we consider the client (A) that uses LTE. Fig. 13 shows the 250 ms artificial delay at the server points S {I, II, III}. However, the values for C {I, IIa} remain unchanged as given in Fig. 12 despite the 250 ms

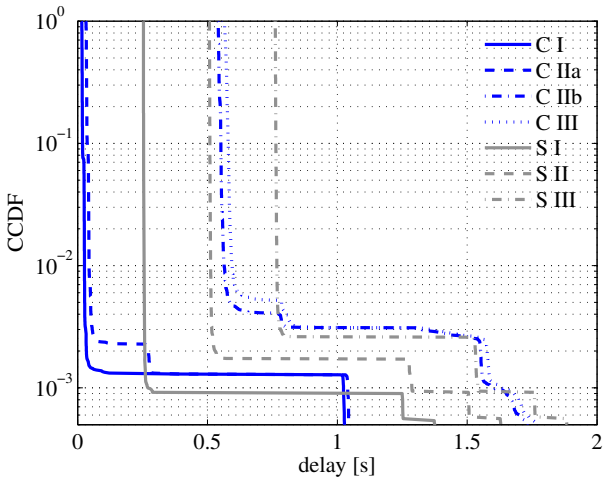


Fig. 13: Distributions for all measurement points at the LTE client (A) and the server (T). The server delays outgoing packets for 250 ms before sending.

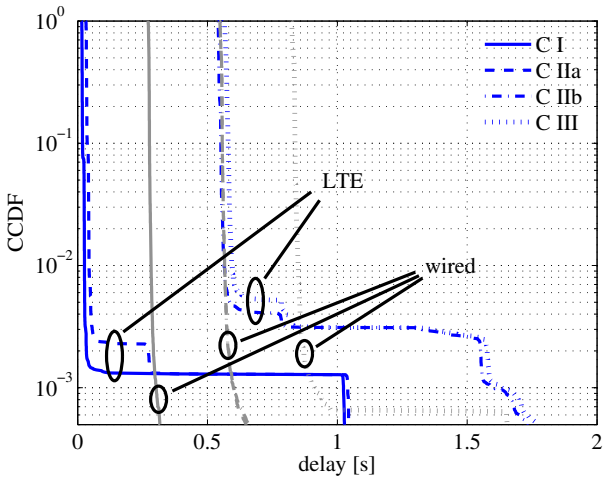


Fig. 14: A comparison of the distributions for the client side measurement points for LTE vs. wired access. The server delays outgoing packets for 250 ms before sending

delay at the server. From this, we evidence that the connection establishment, i.e., the first two handshakes in LTE are done between the client (A) and a transparent middlebox. Observe that the point C IIb correctly experiences the artificial delay twice, i.e., 500 ms, as it is the second handshake on the server side. Point C III again does not experience the additional delay. We conclude that the middlebox in LTE takes over the TCP connection establishment and termination. A quick comparison in Fig. 14 to the staggered delay distributions for the wired client (D) shows how the additional delay at the server (T) amounts 250 ms to every handshake.

Finally, a comparison of the packet timestamps at the LTE client (A) and the server (T) reveals more information on the operation of the considered middlebox. Figure 15 illustrates a measured sample path for one request. Figure 15 sketches the

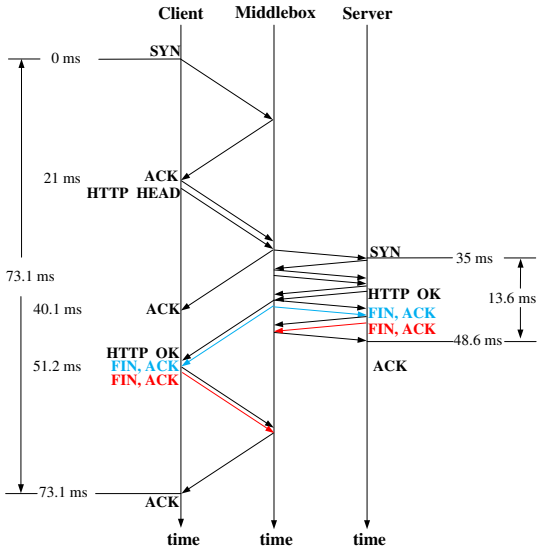


Fig. 15: A measured sample path of client/server interaction for an HTTP HEAD request that originated from the LTE client (A) to server (T).

inferred middlebox behavior from the measured timestamps at the client and the server. Observe that the middlebox performs the connection establishment and sends an ACK for the HTTP HEAD request without waiting for a SYN/ACK from the server.

To prove this, we shut down the webserver (T) to find that the middlebox establishes a TCP connection with the client (A) and sends an ACK for the HTTP HEAD request before sending a `Gateway Time-out` error message (code 504) after 10 seconds. This makes the middlebox resistent to flood type attacks as it throttles the error message rate. On the other hand, the middlebox has to allocate memory for this information for a time interval of 10 seconds. Further, we observe that in this trace the middlebox initiates the connection termination in both directions.

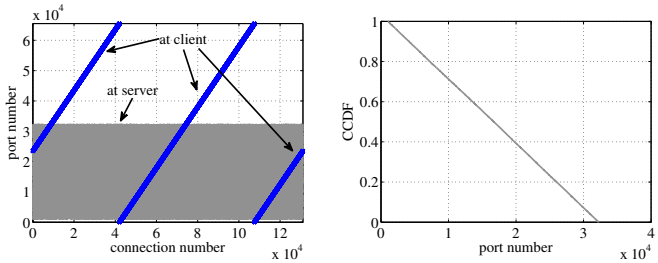
We note that we also deployed the middlebox detection tool *tracebox*, which like *traceroute* did not provide any results in the considered measurement scenarios. In our experiments, we observed filtering of ICMP TTL exceeded messages by the ISP.

To infer the middlebox port filtering rules we use SYN packets sent from client (A) to server (T). Packets that are filtered cause a `Gateway Time-out` error after 10 seconds. For non-filtered packets we receive RST packets from the server (T). Overall, we find that the middlebox filters 10 ports given in Tab. I. The ports are either well-known or registered and mainly used for HTTP, Email and remote login [9].

Further, we investigate the NAT operation mode of the middlebox. At the client (A) we set the ephemeral port range between 1 and $2^{16}-1$. At the server (T) we observe uniform randomly distributed external port numbers chosen from the range 1024 to $2^{15}-1$ as depicted in Fig. 16. The use of the uniform random distribution hinders port prediction, that

port #	description	port #	description
21	FTP	110	POP3
22	SSH	143	IMAP
25	SMTP	993	IMAPS
42	Name Server	8070	unassigned
80	HTTP	8080	alt. HTTP

TABLE I: List of TCP ports that are filtered by the middlebox.



(a) (b)

Fig. 16: (a) Collected TCP port numbers at the LTE client (A) and at the server (T) vs. the corresponding connection number. (b) Measured distribution of the external port numbers.

is an essential part for standard NAT traversal techniques. This configuration will probably lead to connection failure for P2P based applications such as legacy VoIP and video chat applications.

VII. Limitations of the Study

In this work we provided a snapshot of the performance perceived on the application level by a stationary LTE UE. The measurement results are bound to the measured ISP network and to the geographical location of the setup. Using a commercial LTE network we did not have control over the network components and configuration, hence we drew conclusions only from an end-to-end, application-level viewpoint.

VIII. Conclusion

In this paper we presented measurement results on the application-level performance of LTE in a major commercial network. We investigated achievable application-level data rates using different transport protocols. We showed the discrepancy between uplink and downlink in terms of one way delays and maximum allocated buffers. Further, we reported on phenomena such as burst packet losses, traffic burstiness increase and packet delay patterns that can be traced back to LTE specific MAC algorithms and wireless channel properties. Further, we showed the impact of LTE power saving algorithms on the delay performance. In addition, we dissected the operation of HTTP over LTE to discover the deployment of middleboxes in the considered LTE network. We characterized the operation of the detected middleboxes and showed their impact on the measured performance.

References

[1] Iperf. http://sourceforge.net/projects/iperf/. [accessed 06.12.2013].

[2] 3GPP specification TS 25.913. Requirements for Evolved UTRA (E-UTRA) and Evolved UTRAN (E-UTRAN), Jan. 2009. Release 8, version 8.0.

[3] 3GPP specification TS 36.304. Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) procedures in idle mode, June 2011. Release 8, version 8.10.

[4] 3GPP specification TS 36.321. Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specification, Mar. 2012. Release 8, version 8.12.

[5] 3GPP specification TS 36.322. Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Link Control (RLC) protocol specification, July 2010. Release 8, version 8.8.

[6] 3GPP specification TS 36.331. Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification, July 2013. Release 8, version 8.20.

[7] Z. Bozakov and M. Bredel. SSHLauncher.KOM - a tool for experiment automation in distributed environments. Technical Report KOM-TR-2008-11, TU-Darmstadt, Jul. 2008.

[8] Y.-C. Chen, Y.-s. Lim, R. J. Gibbens, E. M. Nahum, R. Khalili, and D. Towsley. A Measurement-based Study of MultiPath TCP Performance over Wireless Networks. In *Proc. of ACM IMC*, pages 455–468, 2013.

[9] M. Cotton, L. Eggert, J. Touch, M. Westerlund, and S. Cheshire. IETF RFC 6335: Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry, Aug. 2011.

[10] C. Cox. *An Introduction to LTE: LTE, LTE-Advanced, SAE and 4G Mobile Communications*. Wiley, 2012.

[11] S. Deng and H. Balakrishnan. Traffic-aware Techniques to Reduce 3G/LTE Wireless Energy Consumption. In *Proc. of ACM CoNEXT*, pages 181–192, 2012.

[12] J. Huang, F. Qian, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck. A close examination of performance and power characteristics of 4g LTE networks. In *Proc. of ACM MobiSys*, pages 225–238, 2012.

[13] 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. *ACM SIGCOMM Comput. Commun. Rev.*, 43(4):363–374, Aug. 2013.

[14] H. Jiang, Y. Wang, K. Lee, and I. Rhee. Tackling bufferbloat in 3G/4G networks. In *Proc. of ACM IMC*, pages 329–342, 2012.

[15] J. Laine, S. Saaristo, and R. Prior. rude & crude. http://rude.sourceforge.net/. [accessed 06.12.2013].

[16] 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 *Proc. of IEEE WiOpt*, pages 286–292, 2012.

[17] B. McWilliams, Y. Le Pezennec, and G. Collins. HSPA+ (2100 MHz) vs LTE (2600 MHz) spectral efficiency and latency comparison. In *Proc. of IEEE Telecommunications Network Strategy and Planning Symposium*, pages 1–6, 2012.

[18] R. B. Miller. Response time in man-computer conversational transactions. In *Proc. of ACM AFIPS*, pages 267–277, 1968.

[19] V. Paxson, M. Allman, J. Chu, and M. Sargent. IETF RFC 6298: Computing tcp's retransmission timer, June 2011.

[20] F. Schneider, B. Ager, G. Maier, A. Feldmann, and S. Uhlig. Pitfalls in HTTP Traffic Measurements and Analysis. In *Proc. of PAM*, pages 242–251. Springer, 2012.

[21] M. Siekkinen, M. A. Hoque, J. K. Nurminen, and M. Aalto. Streaming over 3G and LTE: How to Save Smartphone Energy in Radio Access Network-friendly Way. In *Proc. of ACM MoVid*, pages 13–18, 2013.

[22] T. Tirronen, A. Larmo, J. Sachs, B. Lindoff, and N. Wiberg. Reducing energy consumption of LTE devices for machine-to-machine communication. In *IEEE GLOBECOM*, pages 1650–1656, 2012.

[23] URL. http://www.teldat.de. [accessed 06.12.2013].

[24] URL. http://www.linuxfoundation.org /collaborate/workgroups/networking/netem. [accessed 06.12.2013].

[25] Z. Wang, Z. Qian, Q. Xu, Z. Mao, and M. Zhang. An untold story of middleboxes in cellular networks. *ACM SIGCOMM Comput. Commun. Rev.*, 41(4):374–385, 2011.

[26] J. Wigard, T. Kolding, L. Dalsgaard, and C. Coletti. On the user performance of LTE UE power savings schemes with discontinuous reception in LTE. In *Proc. of IEEE ICC*, pages 1–5, 2009.

[27] M. P. Wylie-Green and T. Svensson. Throughput, capacity, handover and latency performance in a 3GPP LTE FDD field trial. In *Proc. of IEEE GLOBECOM*, pages 1–6, 2010.

[28] L. Zhang, T. Okamawari, and T. Fujii. Performance Evaluation of End-to-End Communication Quality of LTE. In *Proc. of IEEE VTC*, pages 1–5, 2012.