

Estimation of Congestion from Cellular Walled Gardens using Passive Measurements

Vivek Adarsh , Member, IEEE, Michael Nekrasov , Member, IEEE,
Udit Paul , Member, IEEE, and Elizabeth Belding , Fellow, IEEE

Abstract—Despite widespread LTE deployment, coverage does not necessarily translate to usable service. Even in well-provisioned urban networks, unusually high usage (such as during a public event or after a natural disaster) can lead to congestion that makes the LTE service difficult, if not impossible, to use, even if the user is solidly within the coverage area. A typical approach to detect and quantify congestion on LTE networks is to secure the cooperation of the network provider for access to internal metrics. An alternative approach is to deploy multiple mobile devices with active subscriptions to each network operator. Both approaches are resource and time intensive. In this work, we propose a novel method to estimate congestion from overloaded LTE networks using only passive measurements, and without requiring provider cooperation. We analyze packet-level traces for four commercial LTE service providers, from several locations during both typical levels of usage and during public events that yield large, dense crowds. This study presents the first look at congestion detection through overload estimation by examining unencrypted broadcast messages. We show that an upsurge in broadcast reject and cell barring messages, leading to overload, can accurately detect an increase in network congestion.

Index Terms—LTE, Congestion, Overload, Network Measurement, Cellular Networks, QoE.



1 INTRODUCTION

LTE serves over 4 billion users and is slated to remain the dominant cellular network until 2025 [1], even while cellular providers across the globe claim rapid expansion of 5G services. Even as 5G coverage grows, coverage will likely remain spotty, particularly in rural and remote regions [2]. In the meantime, LTE usage has surged, yielding critical challenges in sustaining consistent, high-quality service to an increasing subscriber base. In a well-provisioned region, sudden escalation in traffic demand from user equipment (UE) can occur during large gatherings (e.g., street festivals, protests). Similarly, after a natural disaster, damaged infrastructure and atypical volume of utilization can overwhelm a previously well-provisioned network. Prior work has demonstrated that even in areas that cellular providers claim are well-covered, persistent over-usage due to insufficient capacity can exist [3].

As a specific example, in 2017, Hurricane Maria brought down 95% of cellular sites in Puerto Rico [4]. As a result, affected citizens on the ground were unable to request rescue from rising flood waters. In such disaster scenarios, call volume may overload capacity even when cellular towers remain functional, causing base stations to reject calls [5]. Unfortunately, cellular providers have incentive to state that damaged cellular services have been returned to an operational state. Indeed, after Hurricane Maria, statuspr.org soon reported that over 90% of cell towers were again operational; however, anecdotal evidence from responders

on the ground indicated such statistics were grossly overstated.

To remedy this disparity between reported coverage and actual usability, individual users, watchdog groups and government agencies need tools to verify whether a network is adequately serving customers. After a disaster, the FCC typically receives outage reports from telecoms [4], but the actual usability, due in part to overload, on active towers is difficult to assess without access to the internal network [6]. Ideally, public entities should be able to assess the overload and operational status/usability for a particular base station. Further, they should be able to accomplish this without relying on the cooperation of the cellular provider.

To address this critical need, we propose a novel solution¹ to infer overload and congestion² in LTE networks based on messages broadcast by the eNodeB. We develop Lumos, a data analysis platform that is capable of quantifying overload in eNodeBs. The design and implementation of Lumos is described in section 4. To validate the existence of congestion as detected by Lumos, we develop a network monitoring suite that automates the collection of Quality of Service (QoS) and Quality of Experience (QoE) metrics; this suite is described in section 5. Through the analysis of multiple message types, we draw clear comparisons between instances of high network utilization and typical operating conditions for several eNodeBs. Further, we evaluate performance

1. This study is an extension of [7], where the focus was on estimating overload. In this study we build upon our prior work to detect and quantify congestion as a result of the estimated overload on the network.

2. We consider overload as the state where user equipment is denied access to camp on an LTE base station (eNodeB) due to the current number of connections, whereas congestion is a state while a user device is connected that leads to performance degradation at the user end (e.g., slower downloads, poor video streaming quality, etc.). We explain these two terms in more detail in section §3.3.

- V. Adarsh, M. Nekrasov, U. Paul and E. Belding are with the Department of Computer Science, University of California, Santa Barbara.
- E-mail: {vivek, mnekrasov, u_paul, ebelding}@cs.ucsb.edu

differences incurred as a result of overload-driven congestion through assessment of QoS and QoE metrics. Our results indicate that eNodeBs demonstrate measurable performance differences indicative of overload conditions and network congestion.

Importantly, our solution works without the cooperation of the cellular provider. Using low-cost, readily usable off-the-shelf equipment, we demonstrate that unencrypted broadcast messages sent by the eNodeB on the broadcast channel can be passively collected and analyzed to estimate local overload. We concurrently collect measurements on active monitoring devices to draw parallels between overload and network congestion, and hence network usability.

We quantify our results by computing two normalized metrics, which are proportional to the number of connection reject messages and cell barring signals (`cellBarred`), respectively (cell barring signals prohibit UEs from camping on a particular cell). In addition, we evaluate the back-off timer (`waitTime`) encapsulated in each reject message. Note that in LTE, a connection reject message does not contain a rejection cause. Consequently, we must use higher `waitTime` values, coupled with high rates of connection request denials, to indicate possible overload. To validate our results, we use our network monitoring suite (§5) to demonstrate the corresponding performance degradation at the network (QoS) and user level (QoE). For instance, under high load, QoE for common applications such as Web browsing and video streaming can deteriorate to the point of unusability.

To test the operation of our system, we perform multiple measurement campaigns³: three at events with unusually large crowd gatherings, and three at those same locations but during times of typical usage. Through these measurement campaigns, we collect and analyze over 7 million LTE frames from four major telecom operators in the US (AT&T, Sprint, T-Mobile and Verizon).

Our key contributions and findings include:

- Overload on an eNodeB can be identified through an increase in reject messages and mean back-off time. We show that overloaded cell towers frequently deny $4\times$ larger percentages of connection requests, issue 35% higher `waitTimes`, and broadcast unavailability through 30% more barring signals than baseline measurements;
- Overload conditions are often accompanied by a significant increase in congestion as revealed through considerable drop in service usability at the user end. We observe at least $10\times$ lower throughput, $2\times$ higher latencies and $8\times$ higher packet losses in atypical utilization periods;
- Quality of experience significantly drops for video streaming applications: we note a minimum of $6\times$ higher start-up delay, $3\times$ lower video quality, $3\times$ higher stall ratio and over 30% decrease in buffer levels.

The remainder of this paper is organized as follows. In §2 we discuss prior work. §3 provides a background of Radio Resource Control protocols that form the basis of our analysis. §4 presents Lumos, our passive measurement tool, while §5 describes the active measurement suite that was used to validate Lumos and our dataset. In §6 we present

our research contributions and finally, we conclude the paper in §7.

2 RELATED WORK

Diagnostic methods in LTE networks are known to be cumbersome. This includes packet-level analysis to estimate overload, because messages transmitted after the connection establishment stage are invisible to a passive device. As a result, there is little prior work that leverages passive measurements to detect overload.

Previous work has led to the development of several network analysis tools. `xgoldmon` [8], for instance, can monitor control plane messages over 2G/3G but not LTE. SCAT [9] is a tool designed to detect problems in cellular networks, which although quite useful is limited to only active monitoring on Qualcomm and Samsung baseband chipsets. QXDM [10] is a tool developed to diagnose network statistics that is limited to only Qualcomm baseband chipset and requires a paid license. While [11] offers very similar feature sets to the tools discussed above, it is not tailored to work with software defined radios for passive monitoring. Schmitt et al. [3], [12] employ a comparable approach to ours, except their study is limited to GSM networks. We believe the biggest drawback of these prior tools is their inability to work with passive measurement devices, such as software-defined radios (SDRs).

Several prior works have studied various congestion control algorithms in LTE networks [13], [14], [15], but little work has explored overload detection without involving an active monitoring aspect. Torres et al. [16] use machine learning models to predict network congestion. However, their approach requires considerable historical data and is not suitable for urban sectors where eNodeBs are upgraded regularly to cater to increasing user bases, nor can it be used to assess current overload levels. Chakraborty et al. [17] introduce LoadSense, which offers a measure of cellular load using channel sensing at the PHY layer. Similarly, [18] allows a client to efficiently monitor the LTE base station's PHY-layer resource allocation, and then map such information to an estimation of available bandwidth. Cellular Link Aware Web loading (CLAW) is proposed in [19], which boosts mobile Web loading using a physical-layer informed transport protocol. Although the aforementioned tools can estimate whether the radio resources are fully allocated, they do not explicitly reveal whether the network is overloaded. Oussakel et al. [20] propose a supervised machine learning method to estimate QoS degradation in LTE-A in an emulated environment. Unlike Lumos, their approach requires a considerable amount of training data. Further, they do not evaluate their system with real-world traces. BurstTracker [21] is a tool to detect LTE downlink bottlenecks on the UE itself. However, this idea requires access to the modem's MAC-layer traces, which is facilitated only through QXDM([10], licensed) or MobileInsight ([11], root access).

Our method focuses primarily on analyzing messages broadcast *before* a connection is established, as these messages can be captured and analyzed by low-cost SDRs. Our approach is portable, scalable, independent of any proprietary

3. We plan to publicly release the dataset used for this study.

platform (e.g., Qualcomm, Samsung, etc.) and works with any cellular service.

3 BACKGROUND

We examine cellular transmissions using software-defined radios. While most transmissions on LTE are encrypted between the eNodeB (LTE base station) and UE (user equipment, such as a cellphone) [22], connection establishment messages are sent in the clear. We use these messages in order to determine overload, as described in the following sections. Note that at any given time a UE is connected to a sector on the eNodeB. There can be several sectors on the eNodeB, typically with different coverage areas. In this study, we refer to the connected sector as eNodeB for brevity and to avoid repetitions. However, note that our observations are applicable to said sector of the eNodeB only.

3.1 Radio Resource Control (RRC)

The RRC protocol [22] supports the transfer of common Non-Access Stratum (NAS) information (which is applicable to all UEs) as well as dedicated NAS information (which is applicable only to a specific UE). NAS is a set of protocols that is used to convey non-radio signalling between the UE and core network. Directed RRC messages (unicast to a single UE) are transferred across Signalling Radio Bearers (SRBs), which are mapped onto logical channels [23] – either the Common Control CHannel (CCCH) during connection establishment or a Dedicated Control CHannel (DCCH) if the UE is in an active connection state. Similarly, System Information (SI) messages are mapped to the Broadcast Control CHannel (BCCH). Since messages on DCCH are on a private channel, they cannot be decoded by passive monitoring devices.

Common Control CHannel (CCCH): This channel is used to deliver control information in both uplink and downlink directions when there is no confirmed association between a UE and the eNodeB – i.e. during connection establishment. Messages on this channel are transmitted in the clear and can be passively decoded. We leverage this knowledge to analyze signalling messages and estimate the overload level in an eNodeB.

Broadcast Control CHannel (BCCH): This is a downlink channel that is used to broadcast System Information (SI). It consists of the Master Information Block (MIB) and a number of System Information Blocks (SIBs). The MIB and SIBs are broadcast through Radio Resource Control (RRC) messages. SIB1 is carried by the `SystemInformationBlockType1` message. Though there are other SI messages, we focus on SIB1 for the purpose of this study. SIB1 contains the cell barring (`cellBarred`) status, which indicates whether or not a UE may choose the cell. When `cellBarred` status is indicated, the UE is not permitted to select/reselect this cell, not even for emergency calls [24]. In that case, the UE may connect to another cell, if one exists.

3.2 Signalling Radio Bearers

A Signalling Radio Bearer (SRB) carries CCCH signalling data. An SRB is used during connection establishment to establish the Radio Access Bearers (RABs) and to deliver

TABLE 1: SRB0 Summary

Channel Type	RLC Mode
CCCH	Transparent (Decodable from passive capture)
Direction	RRC Message
Downlink	RRC Connection Setup RRC Connection Reject
Uplink	RRC Connection Request

signalling while on the connection (for instance, to perform a handover, reconfiguration or release). There are three types of SRBs. SRB0 uses the CCCH channel with *transparent mode RLC* while SRB1 and SRB2 use the dedicated channel with *acknowledged mode RLC*. Transparent mode enables the SRB0 to be decoded by non-network equipment such as a software defined radio in the vicinity, while SRB1 and SRB2 cannot. Table 1 shows various signalling messages SRB0 carries.

For our study, we focus on `RRCConnectionReject` messages with corresponding `waitTime` (back-off time, before a UE can again initiate a connection) values; `ConnectionRequest` messages; and `cellBarred` signals (BCCH). We formulate two normalized metrics based on the percentage of reject messages per request sent and the ratio of `cellBarred` signals to the number of SIB1 messages transmitted over thirty-second time bins.

3.3 Congestion Control

While overload can potentially impair services at the eNodeBs, congestion can lead to severe performance degradation at the end-user. Congestion refers to the performance bottleneck experienced by users as a result of significantly higher traffic demand by the UEs. Overload is the scenario that causes the network to deny UEs in order to preserve the load balancing capabilities of the eNodeBs. An overload scenario will manifest as depletion of resources that are critical to the operation of the network.

Congestion control is invoked in order to unburden a cell to an acceptable level when overload is detected, for instance if the cell load remains above a threshold for some continuous period. An alternative strategy, such as that used by WCDMA, is to lower the bit rates of connected users until the load returns to an acceptable level [25]. However, in a pure packet-based system such as LTE, the user bit rate is maintained at the MAC scheduler [26], which already provides a soft degradation of user throughput as the system load increases. Thus, if congestion is detected in a cell the system must remove a subset of the connected bearers until the load is reduced to an acceptable level. Admission Control [26] is used to restrict the number of UEs given access to the system, in order to provide acceptable QoS to admitted users.

4 LUMOS: DETECTING OVERLOAD

To examine and quantify cell load on eNodeBs we develop Lumos. Lumos is based on the idea that third-party assessment tools should be accessible to the community and carry a low hardware footprint. Our design philosophy is driven by implementing comprehensible systems that are easy to understand and orchestrate.

to compare the performance of mobile broadband under varying network conditions.

5.1 Implementation

The monitoring suite’s functionality ranges from computing network level (throughput, latency and packet traces) to application level (on-demand video streaming (YouTube)) and page load time measurements. We measure cellular performance by tethering phones to laptops running the monitoring suite. We ensure that the cellular plans on all our devices have unlimited data and are hot-spot enabled to effectively achieve the same level of performance as we would on the mobile device. This tool was developed for Linux, keeping ease of deployment in mind. It is agnostic to network type and provides flexible deployment in either wired, Wi-Fi or cellular environments. Development of an integrated smartphone app was impractical as the level of unification achieved for various application-level measurements (YouTube, Skype, etc.) was simply not possible on smartphone operating systems, given the walled access to iOS ecosystem and recent restrictions introduced in Android APIs [32].

Latency: The monitoring suite’s `rtt_out` function automates the collection of round-trip times by initiating pings through Hping3 [33] to a server hosted on an AWS instance (Virginia). We configure Hping3 to use TCP packets instead of ICMP. The ping duration is capped at 120 seconds with one-second intervals between each ping. The average latency is then computed using two different sessions - one before the throughput tests (described below) and one after. This enables us to capture the latency variation introduced in the network, if any, after a throughput measurement session. We observe an average round-trip time of 61ms with a standard deviation of ± 3 ms, across all of the baseline measurements.

Throughput: To calculate the achieved throughput we initiate iPerf threads to download a 10 MB file from the same AWS instance as the latency test. The measurement is repeated 10 times and results are saved at the client side. We concurrently collect packet traces at the client to compute second-order metrics such as packet loss.

Page Load Time: Load times are initiated through the `plt_stream` function. We automate the loading of Web pages using Selenium [34]. For our measurements, we use the Tranco Top 25 list [35]. To evaluate load times, we log the performance timings of a Web page starting from `navigationStart` through the `loadEventEnd` event. These instances of event timings support fine-grained analysis of page load times. We set the monitoring suite to run `plt_stream` three times for better estimation of load times. The browser cache is automatically wiped after each Web page load to reflect true load time for the next iteration.

Video Streaming (YouTube): Examination of QoE metrics from on-demand video streaming services is a challenging problem, particularly because of encrypted traffic, as demonstrated by prior work [36], [37]. Because of the wide proliferation of video applications, user experience for streaming services is critical on mobile broadband. We built the `video_stream` function into the monitoring suite

to log QoE metrics from YouTube videos. To execute this experiment, we first automate the loading and playback of the YouTube video on the Chrome browser using Selenium [34]. The video resolution is set to auto. Then we use YouTube’s iframe API [38] to capture playback events reported by the video player. The API outputs a set of values that indicate player state (not started, paused, playing, completed, buffering) using the `getPlayerState()` function. The API also provides functions for accessing information about play time and the remaining buffer size. To ensure uniformity across all our datasets, we loop a 180-second video three times, for every location and cellular operator.

5.2 Datasets

We identify times and locations in which we anticipate cellular overload (§6.1); capture traces; and then compare network performance in those traces with baselines captured in the same location during normal operating conditions (when no network overload is likely to occur). We select spaces that are anticipated to have large gatherings but that are unlikely to be provisioned for large crowds (i.e. city streets as opposed to stadiums, which typically have sufficient network capacity to handle crowds).

Our hypothesis is that during large crowds we will observe higher numbers of `RRCCONNECTIONREJECT` messages than in times of regular operation. We demonstrate in [7] that first- and second-order metrics derived from `RRCCONNECTIONREJECT` messages can assess overload in nearby LTE eNodeBs. In this study we collect three new extensive datasets from several locations across California. Further, to establish the effect of overload on network congestion and user experience, we undertake a measurement study to synchronously aggregate QoS and QoE metrics. Overall, our dataset consists of over 7 million LTE frames for overload estimation, with data collection that lasts for a cumulative duration of about 10 hours. While it is not possible to compute the exact number of UEs in the vicinity due to the lack of international mobile subscriber identity (IMSI) number in broadcast messages for security reasons, measuring the number of temporary unique UE IDs (`uniqueUeID`) in RRC Connection Requests allows us to estimate the number of active UEs present nearby. Using the monitoring suite, we collect network level (throughput, latency and packet loss) and application level (YouTube streaming and page load times) measurements concurrently. To avoid reiterations, the monitoring suite is run alongside Lumos for all the datasets described below. Table 2 provides an overview of the datasets.

Adams Street Fair (ADM): We collected LTE traces during the 38th annual Adams Avenue Street Fair in the Normal Heights neighborhood of San Diego. The street fair was held on Sunday September 22nd 2019, beginning at 10:00AM and concluding at 6:00PM. We physically positioned our networking gear in a cafe on the same street as the fair (Adams Avenue) to better assess the eNodeBs serving this particular region, as shown in (Figure 2(a)). The total duration of data collection is 129 minutes, which resulted in over 1.63 million LTE frames. In addition, we observed 59,084 `uniqueUeIDs`.



(a) ADM dataset. Adams Avenue, San Diego, CA



(b) CWF dataset. Waterfront Park, San Diego, CA



(c) AIS dataset. Palace of Fine Arts, San Francisco, CA

Fig. 2: Google aerial map of experimental datasets.

Adams Street Fair Baseline (ADM_base): As a point of comparison for the ADM dataset, we gathered LTE traces from the same location, from 7:00AM to 9:00AM on Saturday, September 28th, 2019. Collection early in the morning on a weekend helped us to avoid unexpected large gatherings in the neighborhood, while still capturing activity of local residences and businesses. Compared to the ADM dataset we expect this dataset to exhibit low levels of overload, acting as a baseline for the location. Indeed, we see about 5,307 uniqueUeIDs. We collect a little over 316K frames in 57 minutes.

Concert WaterFront (CWF): We collected traces from the CRSSD music festival concert at the Waterfront Park in downtown San Diego on Sunday September 29th 2019. We monitored the event between 4:00PM and 7:00PM. In total, we collect 126 minutes (1.89 million frames) of traces during this time period. This event/location combination (as shown in Figure 2(b)) was selected because we anticipated that the amount of cellular traffic during the event would well-exceed the typical traffic load. Over this two day event, there were an estimated 15,000 attendees. Because the waterfront does not typically have large crowds, we expect there to be network overload during a large event. This dataset contains 69,728 uniqueUeIDs.

Concert WaterFront Baseline (CWF_base): As a baseline to the CWF dataset, we captured additional traces (442K frames) in the same location on Monday September 30th 2019, from 10 to 11am, when the number of pedestrians and amount of vehicular traffic was more representative of normal operating hours. We detect only 7,478 uniqueUeIDs during this data capture.

AI Summit (AIS): For our third congested dataset, we collected traces at the AI Summit, held on September 25th 2019 at the Palace of Fine Arts in San Francisco between 10:00AM to 3:00PM. The event attracted more than 6000 participants hosted within the confinements of the venue.

TABLE 2: Dataset Information

Locations	Duration	# LTE Frames	# uniqueUeID
ADM	129 mins	1.63M	59,084
CWF	126 mins	1.89M	67,728
AIS	149 mins	2.34M	111,404
ADM Base	57 mins	316K	5,307
CWF Base	62 mins	442K	7,478
AIS Base	65 mins	396K	6,089

Because of the size and tech-centric nature of the event, which requires participants to be digitally connected, we anticipated cellular congestion. Even though the venue provided Wi-Fi coverage, anecdotal evidence suggests that a major fraction of participants were on cellular service - a fact vibrantly exhibited through severe congestion on all of our test and personal mobile devices. This behavior could be attributed to the need to proactively login to Wi-Fi through the dedicated conference app, which would inevitably require downloading via cellular data. After parsing our dataset, we observe about 2.34 million LTE dataframes collected over a period of 149 minutes. In addition, we identify 111,404 uniqueUeIDs.

AI Summit Baseline (AIS_base): In order to establish a baseline for network performance in AIS, we ran our network monitoring suite at the same location. When our measurements were taken at 9:00PM on September 26th 2019, the venue was closed. Hence, we collect traces from the parking lot, roughly 30 meters away from our previous placement (i.e. AIS). In order to account for any disparity arising from slight change in location, we ensure that all our devices monitor and connect to the same cell towers that were present in AIS (this is achieved by matching the CellID parameter). The tests ran for over 65 minutes and collected over 396K frames. This dataset contains 6,089 uniqueUeIDs.

6 EVALUATION

We begin our analysis by studying the broadcast messages transmitted by eNodeBs. We observe that the rate of transmission for RRCConnectionReject messages can accurately indicate the state of network overload. Further, we evaluate QoS and QoE metrics to learn that severe congestion is introduced during network overload, which leads to degradation in user experience.

6.1 Overload Analysis

We analyze five RRC elements: (a) RRCConnectionReject, (b) waitTime, (c) RRCConnectionRequest, (d) cellBarred signal and (e) number of SIB1s transmitted (#SIB1). Collectively, we refer to this data as "RRC metrics". We plot the values of these RRC metrics over thirty-second bins. We find that thirty-second bins are appropriate for our analysis because smaller time bins have little to no relative variation between the samples; however, we miss important data points when we use sixty-second or larger bins. Our evaluation indicates that the rate of transmitted RRCConnectionReject messages in all of the locations is at least 4× higher than their respective baselines, in

accordance with our initial hypothesis. Further, we discover an increase of more than 30% in *cellBarred* signals and 35% higher *waitTime* values in overloaded datasets (i.e., ADM, CWF and AIS). For all of the following results, plots are color coded corresponding to their respective operator's logo for readers' convenience and easy understanding.

6.1.1 Rejects

According to [39], an eNodeB may send an `RRConnectionReject` in response to the UE's `RRConnectionRequest` for exactly one of the following three reasons: (i) the eNodeB is overloaded (e.g., severe increase in requesting UEs that the eNodeB cannot accommodate); (ii) the necessary radio resources for the connection setup cannot be provided (for instance, damaged equipment on eNodeB that results in limited access to the core network); or (iii) the Mobility Management Entity (MME) is overloaded. The MME is the key control-node for the LTE access network, which serves several eNodeBs. It is in charge of all the control plane functions related to subscriber and session management. Once the MME detects overload, it transmits an `overload start` message to the affected eNodeBs, signalling them to reject connection request messages that are for non-emergency and non-high priority mobile originated services.

Analysis of the reject messages sent over a fixed time interval can quantify the level of overload in the network. Figure 3 illustrates the average number of reject messages transmitted in thirty-second bins. As predicted, we notice significantly more reject messages in the overloaded datasets (ADM, CWF and AIS). Figure 3(a) indicates that, on average, Sprint and T-Mobile networks broadcast eight times more reject messages during ADM as compared to the ADM baseline (Figure 3(b)). We see that AT&T and Verizon are slightly less overloaded with about $4\times$ increase from their respective baselines. In CWF, we observe a similar ($\sim 5\times$) increase in rejects for all networks except T-Mobile, which reports a marginally lower increase (as shown in Figure 3(c)). Finally, Figure 3(e) (AIS) displays considerably more reject messages for all operators, as compared to their respective baselines in Figure 3(f). We posit that this trend is due to the dense presence of participants, as demonstrated by over 2 million captured LTE frames. Upon closer inspection of the AIS dataset, we detect $5.5\times$, $9\times$, $7\times$ and $6\times$ increase in `RRConnectionRejectmessages` for AT&T, Sprint, T-Mobile and Verizon, respectively. The significant increase in reject messages is a clear indication of an upsurge in network utilization.

6.1.2 Phi (Φ) Measure

To better understand how overload levels vary, we examine a normalized second-order metric. We define the Phi (Φ) measure as the ratio of the number of `RRConnectionReject` messages to the number of `RRConnectionRequest` messages. Once again, we choose a bin size of 30 seconds. The Phi measure provides an indication of the severity of overload, as it reflects the percentage of new users who were unable to connect to the network. In future studies, we plan to examine the temporal variation in Phi (or the number of new users that were rejected) in order to quantify the maximum acceptable load threshold in eNodeBs. The overall trend

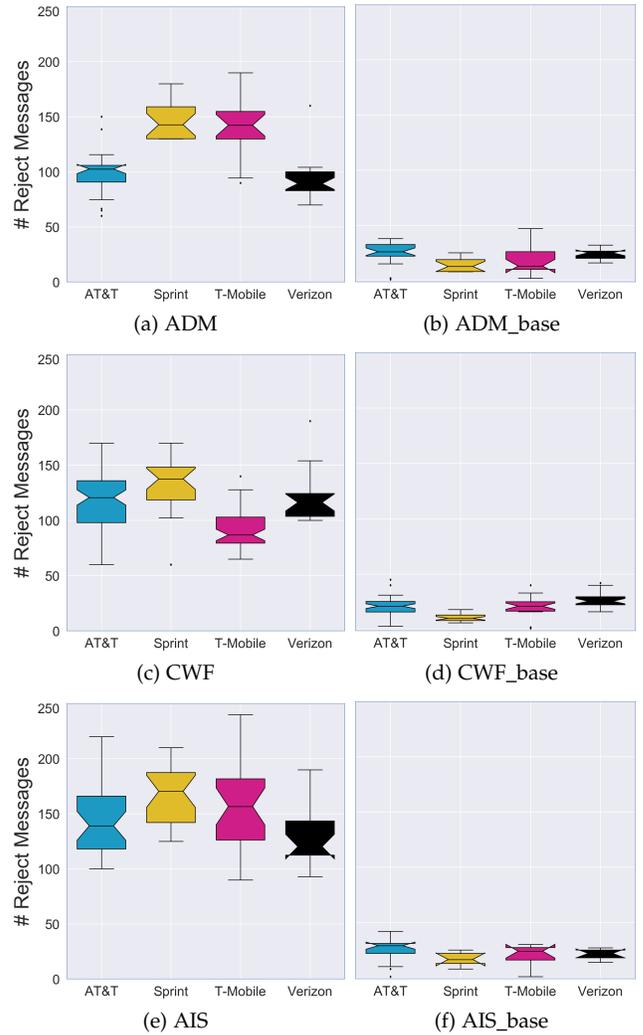


Fig. 3: Number of `RRConnectionRejectmessages` transmitted in thirty-second bins.

is similar to what we observed in Section 6.1.1. It is also indicative of the relationship between the number of UEs (`#uniqueUeIDs`) to the tendency towards network overload, as is expected.

Our examination reveals a remarkable difference between overloaded datasets (i.e., ADM, CWF and AIS) and their respective baselines. Figure 4(a) shows that Phi is more than three times that in Figure 4(b). This difference is even more pronounced in the CWF dataset. Figure 4(c) shows an increase of about $5.5\times$, $5.5\times$, $4.5\times$ and $3.5\times$ in Phi measure, respectively, as compared to Figure 4(d). Sprint underperforms in our evaluation of Phi, with over $8\times$ difference observed between AIS and AIS baseline. Further, we note a considerable variance in Sprint at ADM baseline as well. This result suggests that Sprint's infrastructure at Adams Avenue is under-provisioned for normal operating conditions when compared to other networks. Overall, Sprint's network appears to have the least ability to handle a sudden escalation in user demand.

6.1.3 Average waitTime

When we compare the average `waitTime` across datasets in Figure 5, we observe that overloaded datasets on the left have longer `waitTimes` than their baselines. An exception

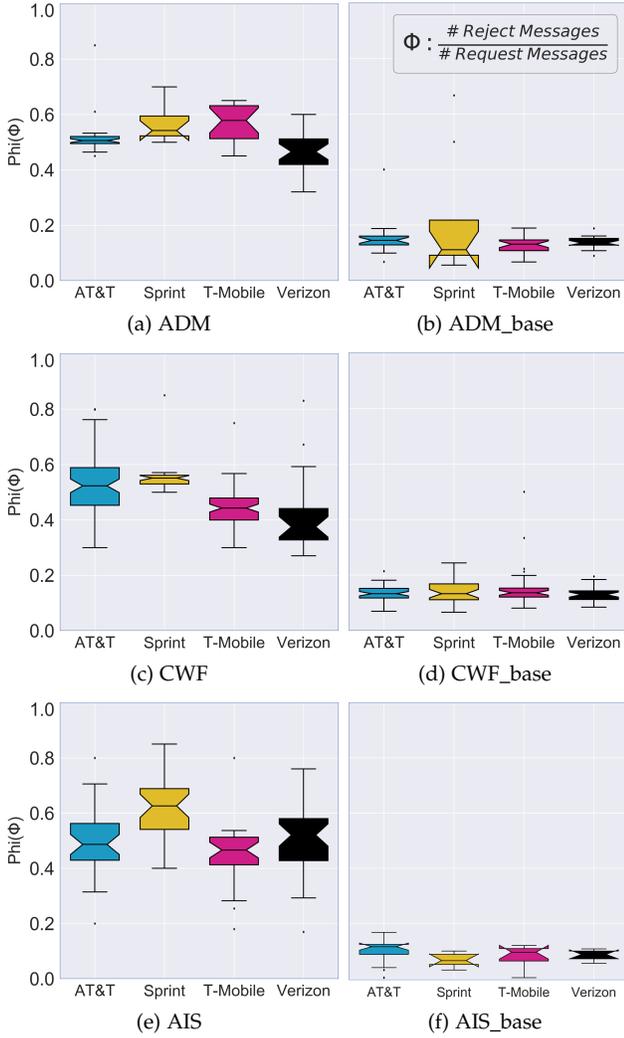


Fig. 4: Phi (Φ) measure in thirty-second bins.

occurs in ADM, where Sprint produces lower waitTimes during baseline measurements. This is likely indicative of lower loads, which would confirm our previous hypothesis that Sprint has far fewer subscribers (at least in ADM) and yet is still under-provisioned for local events such as the Adams Fair. Surprisingly, Verizon shows modestly higher waitTime in CWF despite reporting relatively lower Phi levels in Figure 4(c). In AIS, Sprint appears to perform slightly worse than others, with an average deviation of ~ 4 seconds from its baseline. Note that the sample sizes of these distributions are proportional to the number of reject messages, as shown in Figure 3. Nevertheless, all of the telecom providers transmit longer waitTimes during increases in traffic demand.

Longer waitTime in ADM, CWF and AIS is perhaps explained by the high proportion of UEs (# uniqueUeIDs) in the given location. If the magnitude of UEs is great enough to result in overload, eNodeBs start to curtail overload conditions by engaging proprietary mitigation schemes, one of which is transmitting longer waitTime. The overall result is a confirmation of our hypothesis that these messages and parameter values can be used to detect overload. The comparison supports our earlier results where we compute RRCConnectionReject messages. Average

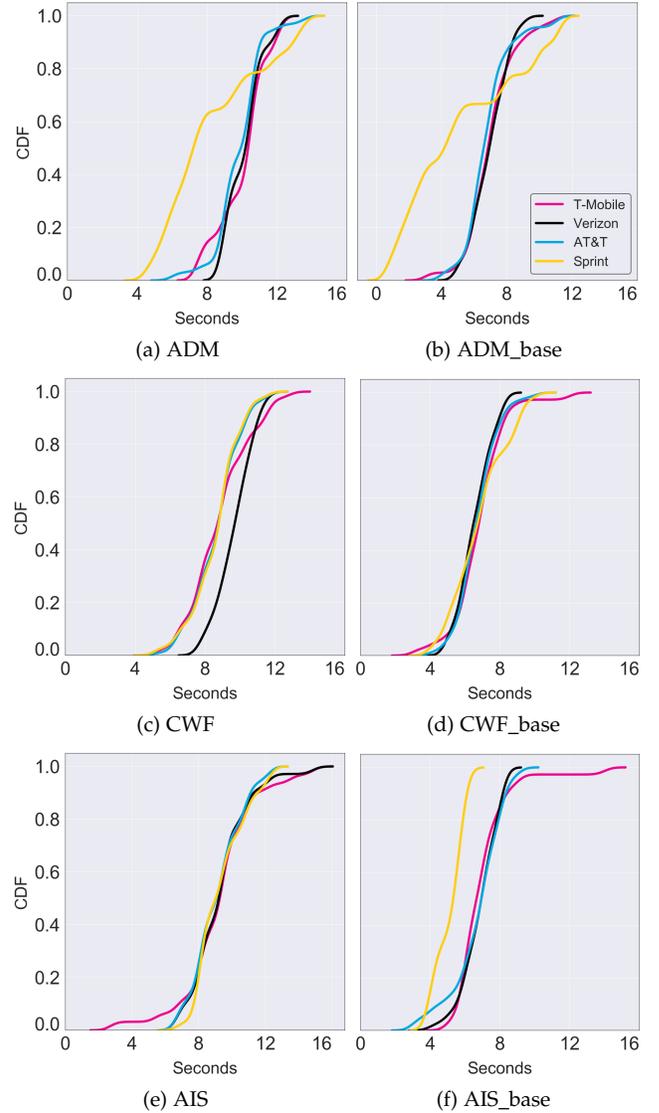


Fig. 5: Distribution of average waitTime.

waitTime serves as an additional indicator of overloaded eNodeBs.

6.1.4 Omega (Ω) Measure

In addition to the reject messages and their corresponding waitTime, cellBarred status is another parameter that can indicate overload in an eNodeB. The cellBarred status transmitted within a system information block 1 (SIB1) message indicates that the UE is not allowed to camp on a particular cell. We suspect that during overload conditions, cells can initiate load balancing by systematically preventing UEs from anchoring on them. In order to evaluate our theory, we analyze cellBarred messages to compare the percentage of these messages in our datasets.

The Omega (Ω) metric allows us to measure the ratio of cellBarred signals transmitted to the number of SIB1 frames received, in thirty-second bins. We use this second-order metric to establish a correlation between Omega and overload. Figure 6 depicts the variation in Omega across all datasets. We observe an increase of 30% in ADM and CWF datasets over their respective baselines and about 45% increase in AIS. This indicates a relationship between cell

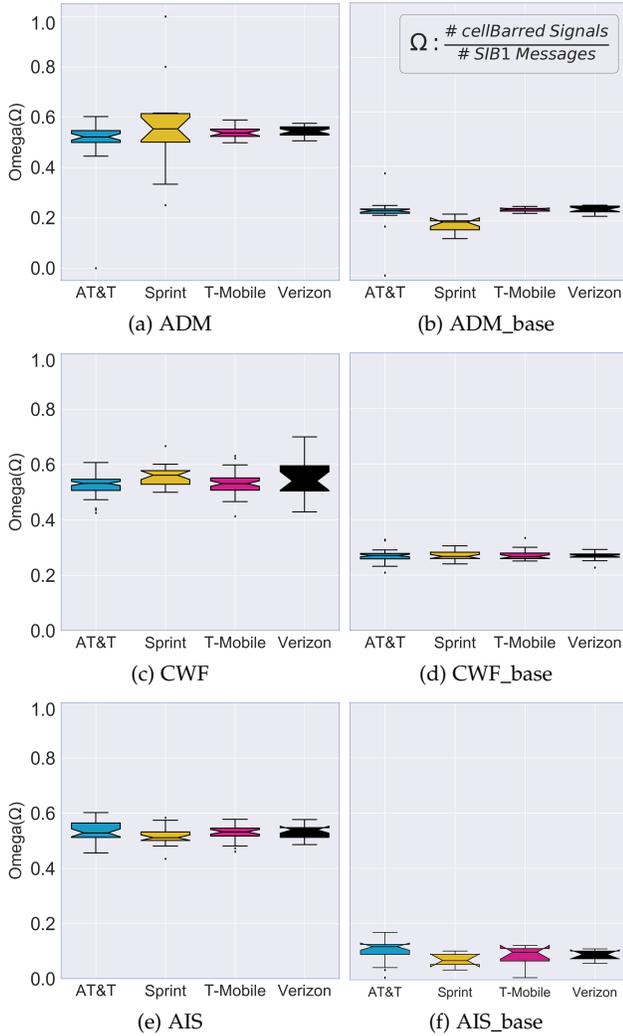


Fig. 6: Omega (Ω) measure in thirty-second bins.

barring signals and overload, confirming our hypothesis. However, it is interesting to observe that each of the mobile network operators have comparable Omega values in the overloaded datasets, even though they exhibit noticeably different trends in Figures 3 and 4. This is similar to the trend we discovered in [7]. Moreover, the AIS baseline has a 15% decrease in Omega measure as compared to other baselines. That is because the inherent load-handling capacity of eNodeBs, as well as the density of users served, apparently differs. This suggests that overloaded eNodeBs, even those that operate under different network conditions, prefer to consistently reject incoming connection requests rather than broadcast unavailability (through cell barring messages), regardless of their proprietary overload mitigation schemes.

6.2 Congestion Detection through Quality of Service

Given the current density of LTE deployments, the massive amount of multimedia traffic on these networks calls into question the Quality of Service (QoS) of these flows. The extent of QoS optimization required from a network management context depends on the type of application being used. For instance, delay-sensitive Internet traffic, such as live streaming video, voice over IP, and multimedia teleconferencing, requires low end-to-end delay in order to

maintain its interactive and/or live nature. On the other hand, on-demand gaming traffic is dependent on both end-to-end delay and achieved throughput. One of the primary barriers to achieving usable QoS in LTE networks is high network utilization, which can cause congestion. In this portion of our study, we collect three QoS metrics (throughput, latency, and packet loss). For throughput tests, we download a 10 MB file from an AWS instance; packet loss is computed from the gathered packet traces. For latency, we use Hping3 to collect average RTT from the same AWS instance. In this section, we evaluate those QoS metrics to study the effect of congestion that manifests as a result of overloaded LTE networks. Our analysis reveals stark differences in performance of congested and baseline measurements. Results show: (i) $10\times - 21\times$ lower throughput, (ii) $2\times - 12\times$ higher latencies and (iii) $8\times - 11\times$ higher packet losses in congested locations.

6.2.1 Throughput

Throughput on mobile broadband is a crucial parameter that reflects the health of the network. The comparison of throughput across an overloaded network to its baseline can reflect the extent of network congestion. Figure 7 shows the comparison of throughput measurements for the overloaded datasets (figures at the top) and their corresponding baselines (bottom figures). We clearly observe that throughput decreases substantially during heavy overload conditions, suggesting network congestion. Across all locations, AT&T and Verizon fare better than their competitors under normal operating conditions, which is consistent with results from independent studies across the industry [40], [41]. Not surprisingly, Sprint has the lowest throughput average across all our datasets, congested or otherwise. Network congestion leads to $24\times$ reduction in throughput measurements, as shown in Figure 7(c). In the congested ADM dataset, we observe higher variability associated with the Verizon and AT&T networks. This suggests that congestion mitigation schemes employed by these networks are marginally more effective as demonstrated by higher median throughput values in the Figure 7. AT&T and Verizon maintain steady rates across all baseline datasets despite serving a disproportionate fraction of users (more than 55% contribution in LTE frame captures).

6.2.2 Latency

With the advent of LTE and 5G networks, stringent requirements have been imposed on latency and reliability [42] with claims by some operators to introduce ultra-low latency on "advanced" LTE networks [43]. Thus, consistent low round-trip time latencies is an indication of a well functioning network [44]. We collect over 200 latency datapoints for each operator at every location (i.e. ADM, CWF and AIS). Table 3 shows the average round-trip times across congested and baseline measurements. We learn that during overload conditions at ADM, average RTT almost doubles, which is reflective of network congestion. In fact, we observe elevated levels of congestion in CWF, which reports latencies as high as $14\times$ (T-Mobile) its baseline measurements. Sprint's network appears to be exceedingly congested with average RTT $48\times$ higher than the baseline. Another notable observation is AIS, where even with the densely populated users and higher demand that lead to lowest average throughput (§6.2.1), the

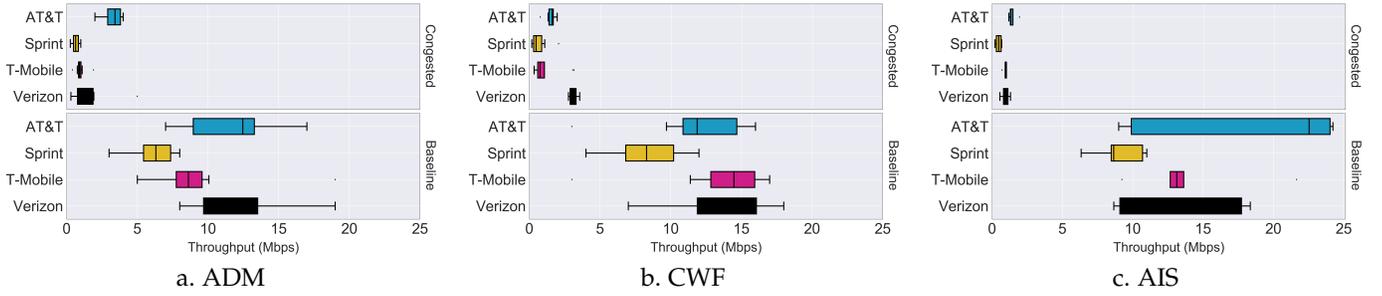


Fig. 7: Throughput measurements across all locations. Figures at the top are from congested datasets while figures at the bottom correspond to baseline measurements.

TABLE 3: Round-trip Times

Locations	ATT	Sprint	T-Mobile	Verizon
ADM	123 ms	112 ms	151 ms	138 ms
CWF	703 ms	3237 ms	857 ms	202 ms
AIS	191 ms	198 ms	134 ms	174 ms
ADM Base	62 ms	64 ms	62 ms	63 ms
CWF Base	63 ms	63 ms	61 ms	62 ms
AIS Base	59 ms	67 ms	63 ms	61 ms

TABLE 4: Packet Loss Rate

Locations	ATT	Sprint	T-Mobile	Verizon
ADM	0.61%	1.44%	1.92%	1.63%
CWF	1.92%	3.47%	2.69%	0.86%
AIS	2.14%	1.73%	1.38%	1.15%
ADM Base	0.15%	0.35%	0.17%	0.22%
CWF Base	0.26%	0.14%	0.39%	0.10%
AIS Base	0.21%	0.16%	0.20%	0.08%

average RTT for AT&T, Sprint and Verizon were about $3\times$ the baseline values. This level of congestion explains the meager average throughput achieved in Figure 7(c). These tests provide useful insights to understand the effect of overload in LTE networks, manifesting in the form of congestion and subsequent degradation in user experience.

6.2.3 Packet Loss

Packet loss in cellular networks is more prominent than wired networks [45]. Loss can happen due to network congestion and/or transmission errors [45]. We compare packet loss rate during crowded events with network overload (§6.1) and contrast these loss rates with those observed during instances of low network utilization. From Table 4 we infer that packet loss increases during overload conditions. We argue that this elevation in loss rate can be attributed primarily to congestion since our physical placement for data collection remains identical in ADM/ADM base and CWF/CWF base; in AIS base, we position ourselves ~ 30 meters from our original location. In addition, we compare the reference signal received power (RSRP) values logged during the tests, within each pairing of our datasets (congested and baseline), to ensure uniformity in radio measurements. We do not claim to have eliminated all aspects of the wireless medium that could contribute to loss rate. For instance, there could be temporary link failures or high bit-error rates. Rather, our focus is to eliminate obvious wireless channel discrepancies that could affect our measurements, such as differential RSRP values. We observe that Sprint’s network experiences $25\times$ more packet loss as compared to its baseline in CWF, whereas other networks have escalations between $6\times$ and $8\times$. In AIS, we see a mean increase of $10\times$ over the baseline, whereas ADM reports $6.5\times$ escalation in packet loss.

6.3 Congestion Detection through Quality of Experience

Quality of Experience (QoE) is one of the leading concepts for network management and performance evaluation in operational networks. Among the most relevant QoE-centric

services consumed by end customers in mobile networks, Web surfing and mobile video take the prime spots [46]. In particular, video now represents over three-quarters of the global IP traffic [46]. In this section we study the performance degradation introduced by congestion, as overload increases on LTE networks. Our results indicate: (i) $6\times - 38\times$ higher start-up delay, (ii) $3\times$ lower video quality, (iii) $3\times - 6\times$ higher stall ratio and (iv) $33\% - 56\%$ lower residual buffer levels in the congested dataset.

6.3.1 Video Streaming: YouTube

Start-up delay: Start-up delay is the time lag registered between user action to play video and video starting to play on the screen. This delay usually corresponds to how quickly the HTTP Adaptive Streaming (HAS) client is able to fill the threshold buffer required for playback. For instance, we observe diminishing throughput during congestion in Figure 7. Such scenarios would likely require additional time to download the same number of video chunks that go into the video buffer than in an uncongested network, leading to higher start-up delays. Here we note that start-up delay does not convey any information about the video resolution chosen for playback. Figure 8 reports the delay incurred during our measurement campaign. Upon examination, we observe a significant increase in delay as overload increases (figures at the top), signalling heavy congestion in the networks. We see that T-Mobile and Sprint have the most heavy-tailed distributions (outliers on either ends of the box), which indicate variable delay. This could be due to either variability in network throughput, or client fallback to lower resolution video, possibly after failed attempts to achieve higher bit-rates (or fetch higher resolution chunks).

Video Quality: With the proliferation of high resolution displays on smartphones and tablets in the past few years, it is now possible for users to take advantage of high-definition videos on their devices, with some mobile devices that offer 4K ultra high-definition capability. Prior studies have illustrated that drop in video resolution has a notable negative effect on user experience, such as sustained

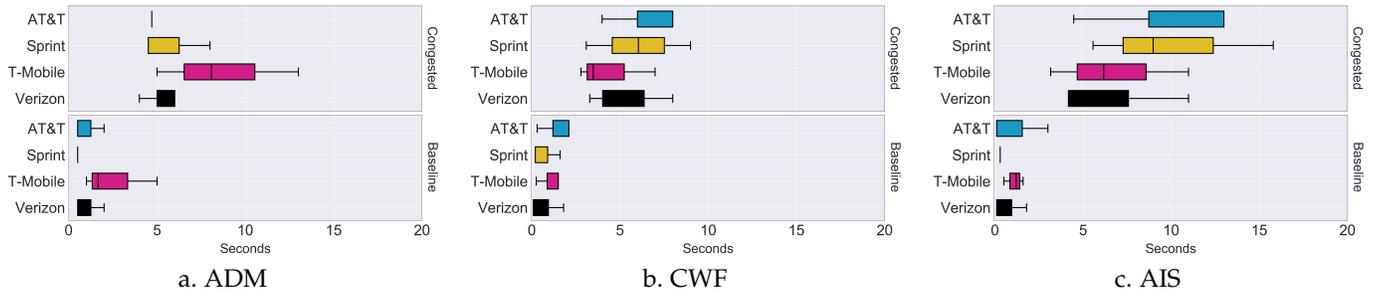


Fig. 8: Start-up delay during YouTube streaming. Lower is better.

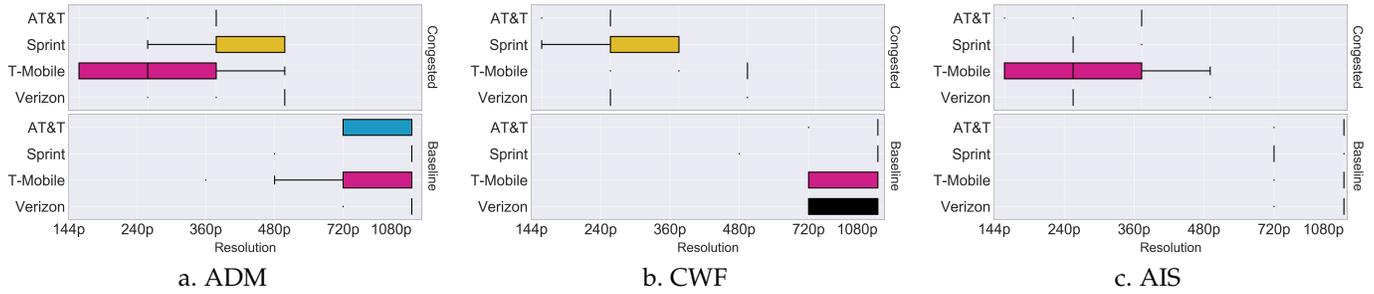


Fig. 9: Achieved video quality during YouTube streaming. Higher is better.

TABLE 5: YouTube stall ratio percentage.

Locations	ATT	Sprint	T-Mobile	Verizon
ADM	4.56%	0%	0%	0.85%
CWF	2.3%	0%	3.07%	18.53%
AIS	3.41%	0.29%	7.27%	2.49%
ADM Base	0%	0%	0%	0%
CWF Base	0.68%	0%	0.98%	0.3%
AIS Base	0%	0%	0.22%	0.44%

frustration [47]. Figure 9 depicts playback resolution of the YouTube video, sampled at one second granularity. During our measurements, we ensure that the video used for playback is uniform across all our datasets. However, the resolution is set to auto. Stated otherwise, final playback resolution and switches are dependent on network conditions and changes in congestion levels. While most of the baseline measurements (Figure 9, plots at the bottom) indicate near full-HD (1080p) rendering of the video, congested dataset (plots at the top) report severe drop in resolution. AT&T and Verizon have near consistent resolution rendering during congestion while T-Mobile and Sprint networks have wide variability. This is a significant finding. Variability in video resolution implies constant quality switches, which is usually perceived as an acute case of performance degradation in QoE [47], [48]. Overall, T-Mobile has the greatest number of quality switches across all locations (congested and baseline).

Stall Ratio: Re-buffering events on video streaming applications usually translate to unusable service [47]. Among other network artifacts, congestion can lead to an increase in re-buffering events while streaming online videos [49]. If re-buffering happens, the user notices interrupted video playback, commonly referred to as *stalling*. The stall ratio is the amount of time the video stalls during the playback expressed as a fraction of total playback time, shown in Table 5. Although not all telecoms report stalling across the three locations (i.e. ADM, CWF and AIS), those that do have a significantly higher ratio than their corresponding baselines.

For instance, we see 60× increase stall ratio on Verizon network in CWF. Similarly, our analysis reveals 30× and 3× increase on T-Mobile network at AIS and CWF, respectively. AT&T and Verizon report non-zero stall ratio across all of the congested datasets. Sprint, despite its poor performance in start-up delay and video quality, has the least stalled video with no stalls reported in either ADM and CWF locations.

Buffer Size: The streaming client employs a playout buffer or client buffer, whose maximum value is buffer capacity, to temporarily store chunks to absorb network variation. To ensure smooth playback and adequate buffer level, the client requests a video clip chunk by chunk using HTTP GET requests, and dynamically determines the resolution of the next chunk based on network condition and buffer status. When buffer level is below a low threshold, the client requests chunks as fast as the network can deliver them to increase buffer level. The playback stalls when the buffer is empty before the end of the playback is reached. From the perspective of YouTube video playback, a session can contain two exclusive regions: buffering and playing. The buffering region is defined as the period when the client is receiving data in its buffer, but video playback has not started or is stopped. The playing region is defined as the period when video playback is advancing regardless of buffer status. In the playing region, video state can be in either buffer increase, decay, or steady state. Figure 10 shows the distribution of buffer size captured during YouTube streaming sessions. Congested locations demonstrate lower buffer sizes than the baseline measurements. The median difference in ADM, CWF and AIS is 17.73 seconds, 23.51 seconds and 21.1 seconds, respectively. Among all the operators we evaluate, Verizon’s median difference is the lowest, at about 12.7 seconds. On the other hand, Sprint registers the widest variance with a median difference of more than 30 seconds.

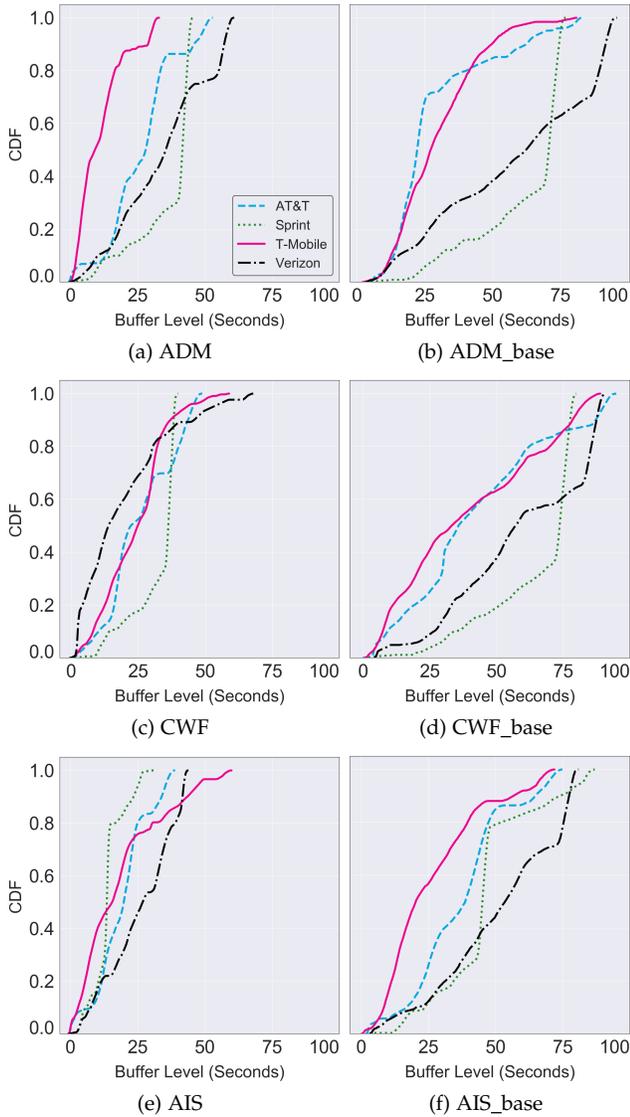


Fig. 10: Cumulative distribution of buffer size (seconds) during YouTube test.

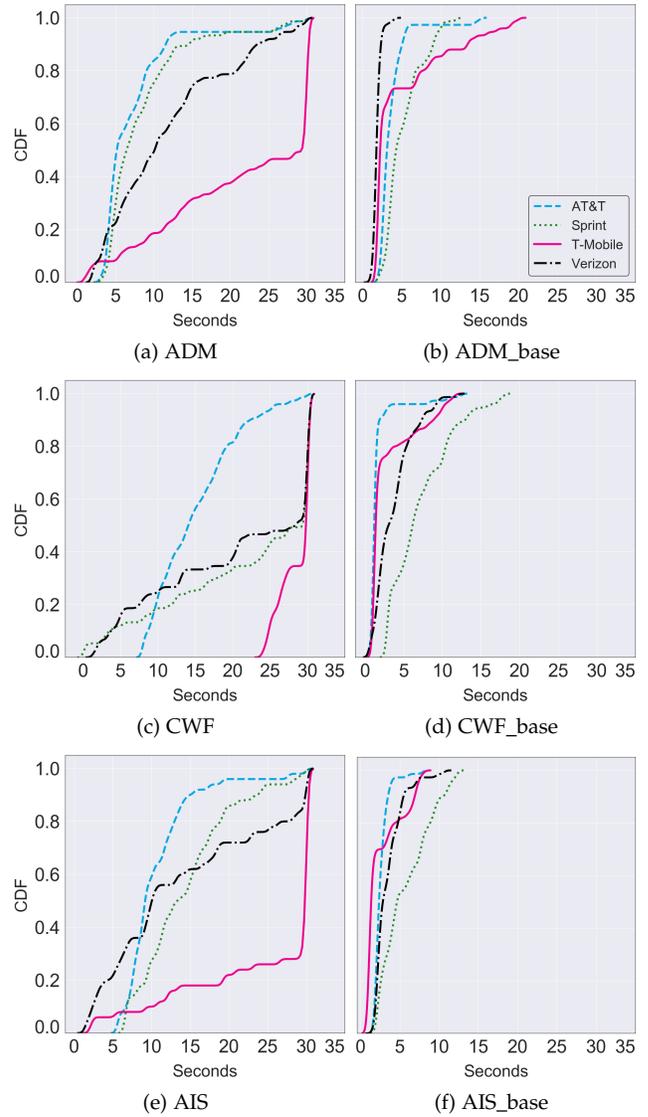


Fig. 11: Page load times of Tranco top 25 websites on all operators.

TABLE 6: Page load time-outs

Locations	ATT	Sprint	T-Mobile	Verizon
ADM	5.33%	6.67%	50.67%	8%
CWF	9.33%	51.89%	64%	42%
AIS	6.67%	5.33%	73.33%	10.67%
ADM Base	0%	0%	0%	0%
CWF Base	0%	0%	0%	0%
AIS Base	0%	0%	0%	0%

6.3.2 Page Load Time

Web performance has long been crucial to the Internet ecosystem since a significant fraction of Internet content is consumed as Web pages. A considerable share of applications such as Web, e-mail or just non-native social media access imply waiting times for their users, which is reflective of the responsiveness from the requested server. Responsiveness is also a function of network conditions, such as congestion [50]. Thus, end-user quality perception in the context of interactive data services is dominated by Web page loading times; the longer the wait, the lower the user satisfaction [50]. Moreover,

studies have shown that *perceived* time for users accessing the Web can be exceedingly magnified with respect to actual chronological time, thus degrading the *perceived* performance even further [51].

Page load times are depicted in Figure 11. From our evaluation we learn that overloaded eNodeBs experience higher congestion levels that leads to a stark contrast between the load times of ADM, CWF and AIS, and their respective baselines. T-Mobile stood out as the worst performing network across all of the congested datasets. AT&T, Sprint and Verizon exhibit deteriorating performance in CWF, which can be explained by inordinate round-trip times detected in Table 3. In the experimental setup, we set the timeout value of 30 seconds. Our choice for this timeout value is derived from Shaik et al. [52], who empirically found that users tend to get tired of wait times by terminating their Web sessions typically after 10-20 seconds. We present the results of Web page timeouts in Table 6. Not surprisingly, T-Mobile produces the highest number of timeouts in all of the congested datasets. Our examination reveals that the CWF dataset reflects poorer performance than other congested

datasets. In AIS, about three-quarters of websites on T-Mobile were either unreachable or could not load required objects before the timeout, indicating severe congestion across the network.

6.4 Discussion

We acknowledge that the addition of ground truth measurements from carriers would have provided another layer of validation. However, we were unable to obtain this data due to providers' strict policies against sharing client related data. Instead, we use our validation methodology (using active measurements to characterize performance and detect congestion) to provide validation from the users' perspective. We believe that our active measurements produce results similar to what would be demonstrated by carrier ground truth data. Further, Lumos requires prior baseline measurements to infer the network condition (congested/overloaded). This is despite the 15–20% reject rate observed across our datasets. Further investigation is needed to ascertain whether these baseline measurements are applicable to other locations and networks.

7 CONCLUSION

In this work, we propose a novel method to assess congestion in nearby LTE eNodeBs, utilizing off-the-shelf hardware and without requiring cooperation of the cellular provider. Our analysis offers convincing evidence that messages broadcast by the eNodeB can be used to detect network congestion by estimating cellular overload. In future work we will explore how passive overload inference can be leveraged in a system for automated overload mapping using ground-based data collection and Unmanned Aircraft Systems (UASs), independent of collaboration from a cellular provider. Software defined radios on UASs have been shown as effective tools for rapidly deployable LTE coverage mapping [53], and we are exploring expanding aerial capabilities to include overload estimation. Such tools can be leveraged by regulators and policy makers and allow targeted deployment of alternative communication channels.

ACKNOWLEDGMENTS

We wish to thank Tarun Mangla for his assistance during development of the monitoring suite. This work was funded in part through NSF S&CC award 1831698.

REFERENCES

- [1] GSMA, "The Mobile Economy Report 2020," <https://www.gsma.com/mobileeconomy/>.
- [2] CNET, "Why 5G is out of reach for more people than you think," 2018. [Online]. Available: <https://www.cnet.com/news/why-5gs-out-of-reach-for-more-people-than-you-think/>
- [3] Paul Schmitt, Daniel Iland, Mariya Zheleva, and Elizabeth Belding, "HybridCell: Cellular Connectivity on the Fringes with Demand-driven Local Cells," in *IEEE INFOCOM*. IEEE, 2016, pp. 1–9.
- [4] FCC, "Communications Status Report for Areas Impacted by Hurricane Maria - September 21, 2017." [Online]. Available: https://transition.fcc.gov/Daily_Releases/Daily_Business/2017/db0921/DOC-346840A1.pdf
- [5] Napa Valley Register, "Earthquake calls flooded 911 dispatch center - September 20, 2014." [Online]. Available: http://napavalleyregister.com/news/local/earthquake-calls-flooded-dispatch-center/article_13f12614-2589-5013-b456-c48aff5663a2.html/
- [6] Udit Paul, Alexander Ermakov, Michael Nekrasov, Vivek Adarsh, and Elizabeth Belding, "Outage: Detecting Power and Communication Outages from Social Networks," in *Proceedings of The Web Conference 2020*, ser. WWW '20, p. 1819–1829.
- [7] Vivek Adarsh, Michael Nekrasov, Ellen Zegura, and Elizabeth Belding, "Packet-Level Overload Estimation in LTE Networks Using Passive Measurements," in *Proceedings of the Internet Measurement Conference 2019*, ser. IMC '19, p. 158–164.
- [8] T Engel, "Xgoldmon," 2013. [Online]. Available: <https://github.com/2b-as/xgoldmon>
- [9] Byeongdo Hong, Shinjo Park, Hongil Kim, Dongkwan Kim, Hyunwook Hong, Hyunwoo Choi, Jean-Pierre Seifert, Sung-Ju Lee, and Yongdae Kim, "Peeking over the Cellular Walled Gardens-A Method for Closed Network Diagnosis," *IEEE Transactions on Mobile Computing*, vol. 17, no. 10, pp. 2366–2380.
- [10] Qualcomm, "QXDM," 2012. [Online]. Available: <https://www.qualcomm.com/documents/qxdm-professional-qualcomm-extensible-diagnostic-monitor>
- [11] Yuanjie Li, Chunyi Peng, Zengwen Yuan, Jiayao Li, Haotian Deng, and Tao Wang, "MobileInsight: Extracting and Analyzing Cellular Network Information on Smartphones," in *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking 2016*, ser. MobiCom '16, p. 202–215.
- [12] Paul Schmitt, Daniel Iland, and Elizabeth Belding, "SmartCell: Small-scale Mobile Congestion Awareness," *IEEE Communications Magazine*, vol. 54, no. 7, pp. 44–50.
- [13] Morteza Tavana, Ali Rahmati, and Vahid Shah-Mansouri, "Congestion Control with Adaptive Access Class Barring for LTE M2M Overload using Kalman Filters," *Computer Networks*, vol. 141, pp. 222–233, 2018.
- [14] Suyang Duan, Vahid Shah-Mansouri, Zehua Wang, and Vincent WS Wong, "D-ACB: Adaptive Congestion Control Algorithm for Bursty M2M Traffic in LTE Networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 12, pp. 9847–9861.
- [15] Raymond Kwan, Rob Arnott, Riccardo Trivisonno, and Mitsuhiro Kubota, "On Pre-emption and Congestion Control for LTE Systems," in *IEEE 72nd Vehicular Technology Conference-Fall 2010*, pp. 1–5.
- [16] Pedro Torres, Paulo Marques, Hugo Marques, Rogério Dionísio, Tiago Alves, Luis Pereira, and Jorge Ribeiro, "Data Analytics for Forecasting Cell Congestion on LTE Networks," in *Network Traffic Measurement and Analysis Conference (TMA) 2017*. IEEE, pp. 1–6.
- [17] Abhijnan Chakraborty, Vishnu Navda, Venkata N. Padmanabhan, and Ramachandran Ramjee, "Coordinating cellular background transfers using loadsense," in *Proceedings of the 19th Annual International Conference on Mobile Computing and Networking 2013*, ser. MobiCom '13, p. 63–74.
- [18] Xiufeng Xie, Xinyu Zhang, Swarun Kumar, and Li Erran Li, "piStream: Physical Layer Informed Adaptive Video Streaming over LTE," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '15, pp. 413–425.
- [19] Xiufeng Xie, Xinyu Zhang, and Shilin Zhu, "Accelerating Mobile Web Loading using Cellular Link Information," in *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '17, pp. 427–439.
- [20] Imane Oussakel, Philippe Owezarski, and Pascal Berthou, "Experimental Estimation of LTE-A Performance," in *15th International Conference on Network and Service Management (CNSM 2019)*.
- [21] Arjun Balasingam, Manu Bansal, Rakesh Misra, Kanthi Nagaraj, Rahul Tandra, Sachin Katti, and Aaron Schulman, "Detecting if LTE is the Bottleneck with BurstTracker," in *The 25th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '19, pp. 1–15.
- [22] 3GPP TS 36.331, "Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification." [Online]. Available: https://www.etsi.org/deliver/etsi_ts/136300_136399/136331/11.11.00_60/ts_136331v111100p.pdf
- [23] 3GPP TS 36.212, "Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding." [Online]. Available: https://www.etsi.org/deliver/etsi_ts/136200_136299/136212/15.02.01_60/ts_136212v150201p.pdf
- [24] 3GPP TS 36.304, "Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) procedures in idle mode."

- [Online]. Available: https://www.etsi.org/deliver/etsi_ts/136300_136399/136304/12.02.00_60/ts_136304v120200p.pdf
- [25] Seung-Hun Oh and Young-Han Kim, "Policy-based Congestion Control in WCDMA Wireless Access Networks for end-to-end QoS," in *COIN-NGNCON 2006-The Joint International Conference on Optical Internet and Next Generation Network*, pp. 153–155.
- [26] Erik Dahlman, Stefan Parkvall, Johan Skold, and Per Beming, *3G evolution: HSPA and LTE for mobile broadband*. Academic press, 2010.
- [27] Ettus Research, "USRP B210." [Online]. Available: <http://www.ettus.com/all-products/UB210-KIT/>
- [28] TAOGLAS, "TG.45.8113 Apex III Wideband 5G/4G Dipole Terminal Antenna." [Online]. Available: <https://cdn3.taoglas.com/datasheets/TG.45.8113.pdf>
- [29] Ismael Gomez-Miguel, Andres Garcia-Saavedra, Paul D Sutton, Pablo Serrano, Cristina Cano, and Doug J Leith, "srsLTE: An Open-Source Platform for LTE Evolution and Experimentation," in *ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation, and Characterization 2016*, p. 25–32.
- [30] 3GPP TS 23.272, "Circuit Switched (CS) fallback in Evolved Packet System (EPS); Stage 2 (Release 10)." [Online]. Available: https://www.etsi.org/deliver/etsi_ts/123200_123299/123272/12.06.00_60/ts_123272v120600p.pdf
- [31] 3GPP TS 38.331, "5G NR; Radio Resource Control (RRC); Protocol specification." [Online]. Available: https://www.etsi.org/deliver/etsi_ts/138300_138399/138331/15.03.00_60/ts_138331v150300p.pdf
- [32] Sai Teja Peddinti, Igor Bilogrevic, Nina Taft, Martin Pelikan, Ulfar Erlingsson, Pauline Anthonysamy, and Giles Hogben, "Reducing Permission Requests in Mobile Apps," ser. IMC '19.
- [33] Kali Tools, "Hping3 Package," 2016. [Online]. Available: <https://tools.kali.org/information-gathering/hping3>
- [34] Selenium, "The Selenium Browser Automation Project," 2019. [Online]. Available: <https://www.selenium.dev/documentation/en/>
- [35] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczyński, and Wouter Joosen, "Tranco: A Research-Oriented Top Sites Ranking Hardened against Manipulation," *arXiv preprint arXiv:1806.01156*, 2018. [Online]. Available: <https://arxiv.org/abs/1806.01156>
- [36] Tarun Mangla, Emir Halepovic, Mostafa Ammar, and Ellen Zegura, "eMIMIC: Estimating HTTP-based Video QoE Metrics from Encrypted Network Traffic," in *Network Traffic Measurement and Analysis Conference (TMA) 2018*, pp. 1–8.
- [37] Craig Gutterman, Katherine Guo, Sarthak Arora, Xiaoyang Wang, Les Wu, Ethan Katz-Bassett, and Gil Zussman, "Request: Real-time QoE Detection for Encrypted YouTube Traffic," in *Proceedings of the 10th ACM Multimedia Systems Conference 2019*, ser. MMSys '19.
- [38] YouTube, "YouTube Player API Reference for iframe Embeds," 2019. [Online]. Available: https://developers.google.com/youtube/iframe_api_reference
- [39] 3GPP TS 36.300, "Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description."
- [40] Ookla, "Mobile Speedtest Intelligence Data," 2019. [Online]. Available: <https://www.speedtest.net/reports/united-states/>
- [41] Tom's Guide, "Fastest Wireless Network 2020: It's Not Even Close," 2020. [Online]. Available: <https://www.tomsguide.com/us/best-mobile-network,review-2942.html>
- [42] Niklas A Johansson, Y-P Eric Wang, Erik Eriksson, and Martin Hessler, "Radio Access for Ultra-Reliable and Low-Latency 5G Communications," in *2015 IEEE International Conference on Communication Workshop (ICCW)*, pp. 1184–1189.
- [43] The Verge, "ATT Wireless on Fake 5G Complaints." [Online]. Available: <https://www.theverge.com/2019/1/9/18175566/att-communications-ceo-fake-5g-e-complaints-response-ces-2019>
- [44] Vivek Adarsh, Paul Schmitt, and Elizabeth Belding, "MPTCP Performance over Heterogenous Subpaths," in *2019 28th International Conference on Computer Communication and Networks (ICCCN)*, pp. 1–9.
- [45] Joel Sommers and Paul Barford, "Cell vs. Wi-Fi: On the Performance of Metro Area Mobile Connections," in *Proceedings of the 2012 ACM Conference on Internet Measurement Conference*, ser. IMC '12, pp. 301–314.
- [46] Cisco, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021 White Paper." [Online]. Available: <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf>
- [47] Tobias Hoßfeld, Michael Seufert, Matthias Hirth, Thomas Zinner, Phuoc Tran-Gia, and Raimund Schatz, "Quantification of YouTube QoE via Crowdsourcing," in *2011 IEEE International Symposium on Multimedia*, pp. 494–499.
- [48] Vengatanathan Krishnamoorthi, Niklas Carlsson, Emir Halepovic, and Eric Petajan, "BUFFEST: Predicting Buffer Conditions and Real-time Requirements of HTTP (S) Adaptive Streaming Clients," in *Proceedings of the 8th ACM on Multimedia Systems Conference 2017*, ser. MMSys '17, pp. 76–87.
- [49] Monia Ghobadi, Yuchung Cheng, Ankur Jain, and Matt Mathis, "Trickle: Rate Limiting YouTube Video Streaming," in *Proceedings of the 2012 USENIX Annual Technical Conference*, ser. USENIX ATC '12, pp. 191–196.
- [50] Tobias Flach, Nandita Dukkkipati, Andreas Terzis, Barath Raghavan, Neal Cardwell, Yuchung Cheng, Ankur Jain, Shuai Hao, Ethan Katz-Bassett, and Ramesh Govindan, "Reducing Web Latency: The Virtue of Gentle Aggression," in *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, ser. SIGCOMM '13, 2013, p. 159–170.
- [51] Raimund Schatz, Sebastian Egger, and Alexander Platzer, "Poor, Good Enough or Even Better? Bridging the Gap between Acceptability and QoE of Mobile Broadband Data Services," in *IEEE International Conference on Communications (ICC) 2011*, pp. 1–6.
- [52] Junaid Shaikh, Markus Fiedler, and Denis Collange, "Quality of Experience from User and Network Perspectives," *Annals of Telecommunications-Annales des Telecommunications*, vol. 65, no. 1-2, pp. 47–57.
- [53] Michael Nekrasov, Vivek Adarsh, Udit Paul, Esther Showalter, Ellen Zegura, Morgan Vigil-Hayes, and Elizabeth Belding, "Evaluating LTE Coverage and Quality from an Unmanned Aircraft System," in *Proceedings of the IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems (MASS) 2019*, pp. 171–179.



Vivek Adarsh received his bachelor's degree from the University of Pune in India. He is currently a PhD Candidate in the department of Computer Science at the University of California, Santa Barbara. His research interests include wireless networks with an emphasis on optimizing Quality of Experience on mobile broadband in challenged environments.



Michael Nekrasov received his PhD in Computer Science from UC Santa Barbara. Previously, he received a B.S. (cum laude) in Computer Science from UC San Diego. His research focuses on wireless aerial networks for environmental and disaster applications. His work applies evolving advancements in computer science to emerging social issues.



Udit Paul received his M.Sc in Electrical Engineering at the University of Cape Town in 2018. He is currently pursuing the Ph.D. degree in Computer Science at the University of California, Santa Barbara with the MOMENT lab. His primary research interests include wireless networks, big data analysis, machine learning and network measurement.



Elizabeth M. Belding is a Professor of Computer Science at UC Santa Barbara. Her research focuses on mobile and wireless networking, and information and communication technologies for development. She is particularly interested in improving Internet accessibility in resource-challenged communities worldwide. She is a Fellow of the ACM, AAAS and IEEE.