Packet-level Overload Estimation in LTE Networks using Passive Measurements

Vivek Adarsh, Michael Nekrasov, Ellen Zegura* and Elizabeth Belding
Department of Computer Science, UC Santa Barbara
{vivek, mnekrasov, ebelding}@cs.ucsb.edu
*Georgia Institute of Technology
ewz@cc.gatech.edu

ABSTRACT

Over 87% of US mobile wireless subscriptions are currently held by LTE-capable devices [34]. However, prior work has demonstrated that connectivity may not equate 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 overload 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 overload 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 mobile network operator (MNO). Both approaches are resource and time intensive. In this work, we propose a novel method to estimate overload in LTE networks using only passive measurements, and without requiring provider cooperation. We use this method to analyze packet-level traces for three commercial LTE service providers, T-Mobile, Verizon and AT&T, 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 overload estimation through the analysis of unencrypted broadcast messages. We show that an upsurge in broadcast reject and cell barring messages can accurately detect an increase in network overload.

CCS CONCEPTS

• **Networks** → **Network performance analysis**; *Network measurement*.

KEYWORDS

LTE, Overload, Passive Measurements, Disaster Management

ACM Reference Format:
Vivek Adarsh, Michael Nekrasov, Ellen Zegura* and Elizabeth Belding. 2019. Packet-level Overload Estimation in LTE Networks using Passive Measurements. In *Internet Measurement Conference (IMC '19), October 21–23, 2019, Amsterdam, Netherlands.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3355369.3355574

IMC '19, October 21–23, 2019, Amsterdam, Netherlands

ACM ISBN 978-1-4503-6948-0/19/10…$15.00
https://doi.org/10.1145/3355369.3355574

1 INTRODUCTION

With 3 billion users and growing, LTE is set to become the leading mobile network technology worldwide in 2019 [24]. With this growth comes critical challenges in sustaining consistent, high-quality service to an increasing subscriber base [36]. In a well provisioned region, sudden escalation in traffic demand from user equipment (UEs) can occur during large gatherings (e.g., street festivals, protests). Similarly, after a disaster, damaged infrastructure and atypical volume of utilization can overwhelm a previously well-provisioned network. Prior work has also demonstrated that even in areas that cellular providers claim are well-covered, persistent over-usage due to insufficient capacity can exist [38].

As a specific example, in 2017, Hurricane Maria brought down 95% of cellular sites in Puerto Rico [21]. 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 [32, 43]. 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 indicated such statistics were grossly over-stated.

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, for instance [22], but the actual usability, due in part to overload, on active towers is difficult to assess without access to the internal network. 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 a cellular provider.

To address this critical need, we propose a novel solution to infer overload in LTE networks based on messages broadcast by the eNodeB. Through the analysis of multiple message types, we draw clear comparisons between instances of high network utilization and typical operating conditions for several eNodeBs. Our results indicate that eNodeBs demonstrate measurable performance differences indicative of overload conditions.

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 [12] on the broadcast channel can be passively collected and analyzed to estimate local overload, and hence 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 case. Consequently, we must use higher `waitTime` values, coupled with high rates of connection request denials, to reveal possible overload.

To test the operation of our system, we perform multiple measurement campaigns: two at events with unusually large crowd gatherings, and two at those same locations but during times of typical usage. Through these measurement campaigns, we collect and analyze over 3.2 million LTE frames. Our analysis indicates that overload on an eNodeB can be identified through an increase in reject messages and mean back-off time. Moreover, these events are often accompanied by a significant increase in cell barring signals. We show that overloaded cell towers frequently deny larger percentages of connection requests and issue higher `waitTime` as compared to typical utilization periods. Further, we observe an unusual number of barring signals prohibiting UEs from camping on their desired eNodeBs.

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` [20], for instance, can monitor control plane messages over 2G/3G but not LTE. `SCAT` [26] is a tool designed to detect problems in cellular networks, which although quite useful is limited to only active monitoring on Qualcomm and Samsung basebands. `QXDM` [31] is a tool developed to diagnose network statistics that is limited to only Qualcomm baseband and requires a paid license. While [28, 39, 40] offer very similar feature sets to the tools discussed above, they are not tailored to work with software defined radios for passive monitoring. Schmitt et al. [37, 38] 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 [19, 27, 30, 41], but little work has explored overload detection without involving an active monitoring aspect. Torres et al. [42] 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. [15] introduce LoadSense, which offers a measure of cellular load using channel sensing at the PHY layer. Similarly, [44] allows a client to efficiently monitor the LTE basestation'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 [45], 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.

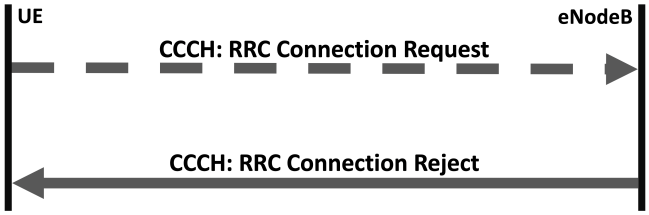


Figure 1: Flow diagram for connection reject message.

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

3 BACKGROUND

In our work we examine cellular transmissions using software-defined radios. While most of the transmissions on LTE are encrypted between the eNodeB (LTE base station) and UE (user equipment, such as a cellphone) [11], connection establishment messages are sent in the clear. We use these messages in order to determine overload, as described in the following sections.

3.1 Radio Resource Control (RRC)

The RRC protocol [5, 10] supports the transfer of common Non-Access Stratum (NAS) [4] information (which is applicable to all UEs) as well as dedicated NAS information (which is applicable only to a specific UE). Directed RRC messages (unicast to a single UE) are transferred across Signalling Radio Bearers (SRB)s, which are mapped onto logical channels [6, 7] – 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 `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

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

is not permitted to select/reselect this cell, not even for emergency calls [9]. In that case, the UE may connect to another cell.

3.2 Signalling Radio Bearers

A Signalling Radio Bearer (SRB) [8] carries CCCH signalling data. An SRB is used during connection establishment to establish the Radio Access Bearers (RABs) and to deliver 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*. In other words, SRB0 can 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 (solid arrow in Figure 1) 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 Managing Overload

Overload management 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 [29]. However, in a pure packet-based system such as LTE, the user bit rate is maintained at the MAC scheduler [17], which already provides a soft degradation of user throughput as the system load increases. Thus, if overload 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 [25] is used to restrict the number of UEs given access to the system, in order to provide acceptable QoS to admitted users.

4 IMPLEMENTATION

4.1 Experimental Setup

In our experimental setup, our receiver is comprised of an Ettus Research USRP B210 [33] SDR attached to a MPantenna SUPER-M ULTRA Mobile Antenna with a frequency range from 25MHz to 6GHz [13]. The USRP is connected to a Lenovo ThinkPad W550s laptop for data collection and post-processing. We use the `srsUE` mode in the open-source srsLTE software suite [23] to locate available cells in the vicinity by scanning all frequency bands. On the

Listing 1: Snapshot of a decoded DL - CCCH message showing RRCConectionReject.

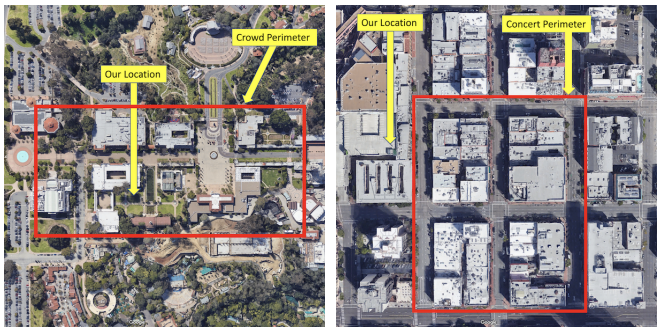
```
"user_dlt": "DLT: 147, Payload: lte-rrc.dl.ccch \
 (LTE Radio Resource Control (RRC) protocol)",
"lte-rrc.DL_CCCH_Message_element": {
 "per.choice_index": "0",
 "lte-rrc.message": "0",
 "lte-rrc.message_tree": {
  "per.choice_index": "2",
  "lte-rrc.c1": "2",
  "lte-rrc.c1_tree": {
   "lte-rrc.rrcConnectionReject_element" : {
    "per.choice_index": "0",
    "lte-rrc.criticalExtensions": "0",
    "lte-rrc.criticalExtensions_tree": {
     "per.choice_index": "0",
     "lte-rrc.c1": "0",
     "lte-rrc.c1_tree": {
      "lte-rrc.rrcConnectionReject_r8_element": {
       "per.optional_field_bit": "1",
       "lte-rrc.waitTime": "6"
       "lte-rrc.nonCriticalExtension_element": {
        "per.optional_field_bit": "1",
        "per.optional_field_bit": "1",
        "per.octet_string_length": "2048",
        "lte-rrc.lateNonCriticalExtension":
         "34:07:79:f0:2c:e7:90:00:28:07:63:48:31:b7:90:00:
          38:07:04:f0:22:67:81:08:30:87:9e:40:3f:37:60:70:
          20:27:82:00:21:17:4c:88:36:47:80:00:20:07:15:00:
          2a:97:90:00:28:17:95:30:2a:97:99:30:2c:87:82:00:
          21:07:4c:f0:36:77:85:b0:22:d7:82:30:21:07:82:40:
          21:27:9f:80:2f:d7:68:18:33:f7:84:00:32:07:23:80:
          21:d7:76:f0:2b:77:91:40:28:a7:81:00:30:97:42:00:
          21:17:88:70:24:27:96:00:2b:07:48:00:24:17:66:00:
          23:d7:93:c0:29:f7:94:00:3a:07:50:f0:38:77:68:80:
```

day of the event, we capture broadcast messages in the form of binary I/Q samples using srsLTE's UE `usrp_capture` utility.

4.2 LTE Packet Decoding

We start with converting binary I/Q samples to hexdumps. To investigate the extent of overload on eNodeBs, we then transform the hexdump into network traces using Wireshark's *text2pcap* command [1]. Next, we use *lte_rrc* lua dissectors to decode LTE RRC messages using *tshark* [16]. We employ *lte − rrc.dl.ccch* and *lte−rrc.ul.ccch* protocols to decode RRC messages on the downlink and uplink common control channel, respectively. Lastly, we use the *lte − rrc.bcch.dl.sch* protocol to decode downlink messages on the broadcast control channel.

Listing 1 shows a snapshot of the decoded RRC message on the downlink CCCH. We can see the *RRCConnectionReject* message tree along with additional options sent by the eNodeB during the RRC connection establishment phase. Embedded in this message is the *waitTime* parameter. While reject messages provide an indication of overload, we can use the value of the *waitTime* metric as a measure of the severity of overload. The value of *waitTime* is an integer in the range of 1–16, which defines how many seconds the UE should wait after reception of the *RRCConnectionReject* until a new connection can be attempted. According to 3GPP TS 23.401 [3], when rejecting an RRC connection request, the eNodeB indicates to the UE an appropriate timer value that limits further requests, relative to the severity of overload; the more the overload, the greater the *waitTime*. Upon receiving the *RRCConnectionReject* message, the UE starts timer T302 [10], with the timer value set to *waitTime*. The UE is not allowed to send another *RRCConnectionRequest* for



(a) SPD dataset. Balboa Park, San Diego, CA

(b) CSR dataset. Downtown San Diego, CA

Figure 2: Google aerial map of experimental datasets.

mobile originating calls, signalling, terminating access or circuit-switched fallback (CSFB) [2, 14] on the same cell until the expiry of T302. Note that in LTE, the *RRCConnectionReject* message does not contain a *RejectionCause*, therefore *waitTime*, in conjunction with reject messages, is a crucial parameter in assessing the level of overload.

4.3 Datasets

To test our proposed solution, we identify times and locations in which we anticipate cellular overload, 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. Overall, our dataset consists of over 3.2 million frames, with data collection that lasts for a cumulative duration of about 5.2 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.

St. Patrick's Day (SPD): We collect cellular traces during the 2019 St. Patrick's Day parade adjacent to Balboa Park in San Diego, CA [35]. The parade was held on Saturday March 16^{th}, beginning at 10:00AM and ending around noon, while the public fair lasted through 3:30PM. We physically positioned our data collection devices within the crowd to better assess the eNodeBs serving this particular region as shown in (Figure 2(a)). The total duration of data collection is about 76 minutes, which resulted in over 1.1 million LTE frames. We observe 27,349 `uniqueUeIDs`.

St. Patrick's Day Baseline (SPD_base): As a point of comparison for the SPD dataset, we again gather LTE traces from the same location, from 8pm to 9pm on Tuesday March 26^{th}. Collection in the evening on a weekday helped us to avoid unexpected large gatherings in the many venues of the park, while still capturing activity of local nightlife. Compared to the *SPD* dataset we expect this dataset to exhibit low levels of overload, acting as a baseline for the location. Indeed, we see about 6,992 `uniqueUeIDs`. We collect a little over 275K frames in 65 minutes.

ShamROCK Concert (CSR): We collect traces from the ShamROCK concert in the downtown area of San Diego [18] on March 16^{th}. The event started at 7:00 PM and lasted until midnight. We collect 113 minutes (~1.7 million frames) of traces during this time perioe. This event/location combination (as shown in Figure 2(b))100 was selected because we anticipated that the amount of cellular traffic during the event would well-exceed the typical traffic load. Because this location (city streets) does not typically have large crowds, we expect there to be network overload during a large event. This dataset contains 42,433 `uniqueUeIDs`.

ShamROCK Concert Baseline (CSR_base): As a baseline to the *CSR* dataset, we capture additional traces (~135K frames) in the same location from 9:30pm to 10:30pm on March 26^{th}, when the number of pedestrians and amount of vehicular traffic was more representative of normal operating hours. We detect only 3,338 `uniqueUeIDs` during this data capture.

5 EVALUATION

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 found that thirty-second bins were appropriate for our analysis because smaller time bins had little to no relative variation between the samples; however, we missed out on important data points when we used sixty-second or larger bins. In our evaluation, we observe that the rate of transmitted `RRCConnectionReject` messages is considerably higher in SPD and CSR than their respective baselines, in accordance with our initial hypothesis. Further, we discover an increase in *cellBarred* signals and *waitTime* values in overloaded datasets (i.e., SPD and CPR).

5.1 Rejects

According to[8], an eNodeB may send an `RRCConnectionReject` in response to the UE's `RRCConnectionRequest` 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

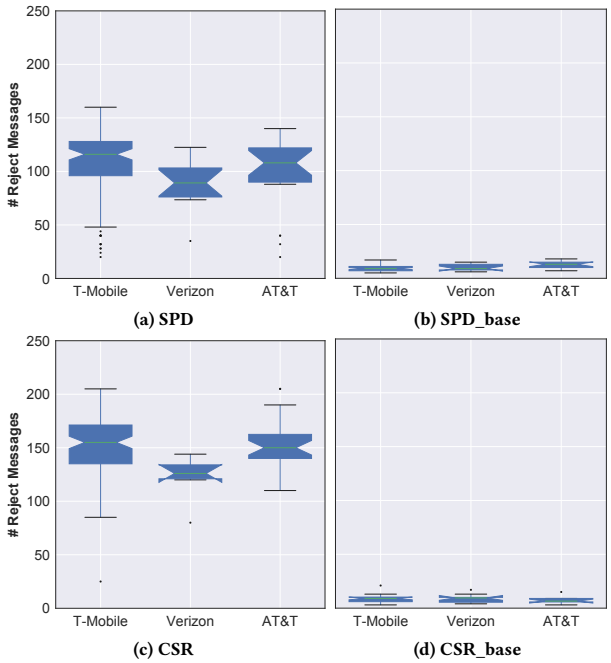


Figure 3: Number of `RRCConnectionReject` messages transmitted in thirty-second bins.

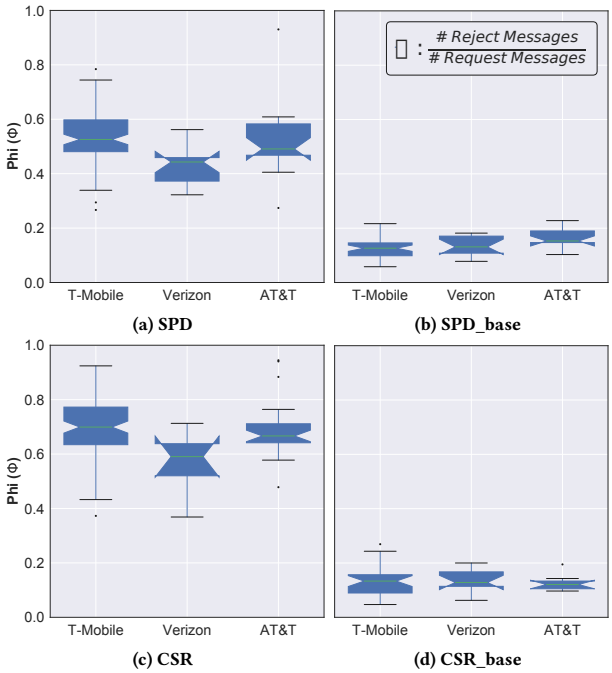


Figure 4: Phi (Φ) measure in thirty-second bins.

the average number of reject messages transmitted in thirty-second bins. As predicted, we notice significantly more reject messages in the SPD and CSR datasets. Figure 3(a) indicates that there are, on average, eight times more reject messages during the SPD parade compared to the SPD baseline (Figure 3(b)). Similarly, we observe a fifteen-fold increase in reject messages in Figure 3(c) as compared to Figure 3(d). This significant increase in reject messages is a clear indication of an increase in cellular network utilization.

5.2 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 `RRCConnectionReject` messages to the number of `RRCConnectionRequest` 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 wish 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. As expected, there is a considerable difference between overloaded datasets (i.e., SPG and CSR) and their respective baselines. Figure 4(a) shows that Phi is as much as three times that in Figure 4(b). This difference is even more pronounced in Figure 4(c), where Phi is more than seven times that in Figure 4(d). This trend is similar to what we observed in Section 5.1. It is also indicative of the relationship between the number of UEs (`# uniqueUeIDs`) to the tendency towards network overload, as is expected.

5.3 Average `waitTime`

When we compare the average `waitTime` across datasets in Figure 5, we observe that SPD and CSR have longer `waitTimes` than their baselines. We also see that AT&T performs worse in SPD, closely followed by T-Mobile. In CSR, T-Mobile appears to perform slightly worse than AT&T. Verizon, however, shows lower `waitTime` in all of the datasets. 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 SPD and CSR 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 infer overload. The comparison is noteworthy as it supports our earlier results where we compute `RRCConnectionReject` messages. Average `waitTime` serves as an additional indicator of overloaded eNodeBs.

5.4 Omega (Ω) Measure

In addition to the reject messages and their corresponding `waitTime`, `cellBarred` status is a crucial 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.

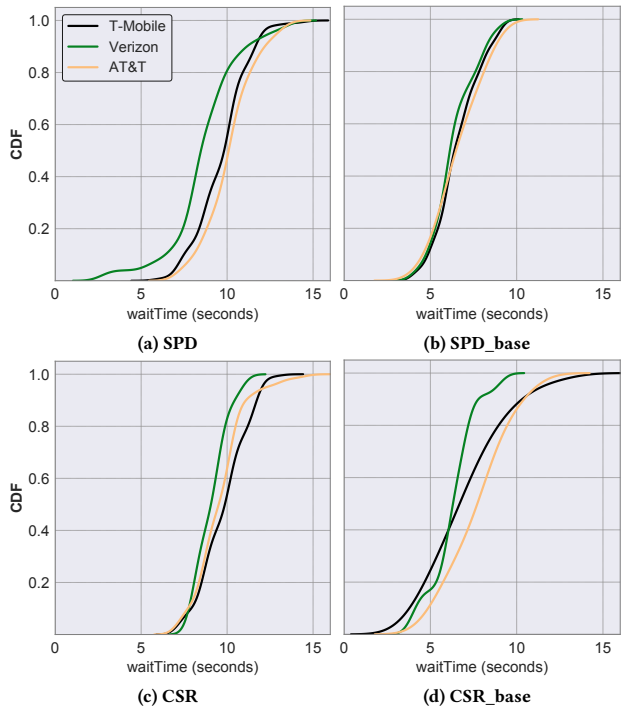


Figure 5: Distribution of average `waitTime`.

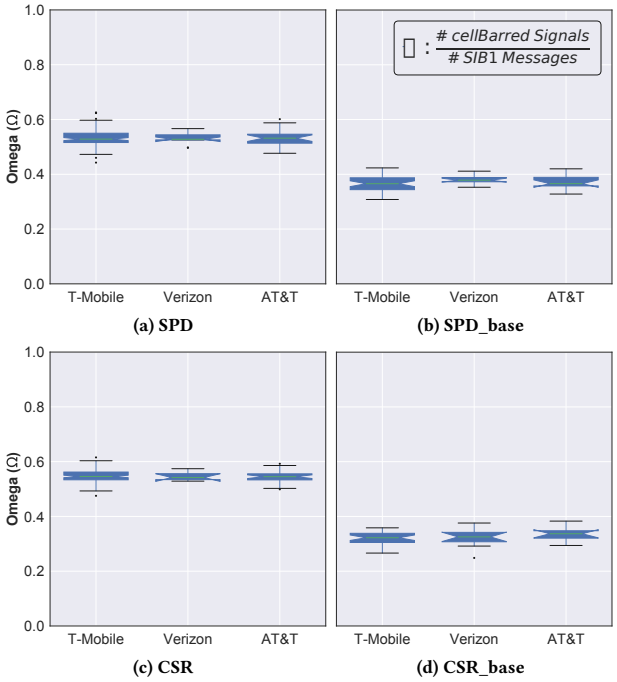


Figure 6: Omega (Ω) measure in thirty-second bins.

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 20% in SPD and CSR datasets over their respective baselines. This indicates a relationship between cell barring signals and overload, confirming our hypothesis. However, it is interesting to observe that each of the mobile network operators (i.e., T-Mobile, Verizon and AT&T) have comparable Omega values in SPD and CSR, even though they exhibit noticeably different trends in Figures 3 and 4. 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 operating in disparate 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 CONCLUSION

In this work, we propose a novel method to assess overload 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 cellular overload using passive monitoring. 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, independent of collaboration from a cellular provider. Such tools can be leveraged by regulators and policy makers and allow targeted deployment of alternative communication channels.

ACKNOWLEDGMENTS

We would like to thank our shepherd, Aaron Schulman and the anonymous IMC reviewers for their valuable feedback on the paper. We also wish to thank Sherri Lynn Conklin for assisting us in one of the data collection campaigns. This work was funded in part through NSF NeTS award 1563436.

REFERENCES

[1] 2014. Text2pcap - Generate a capture file from an ASCII hexdump of packets. https://www.wireshark.org/docs/man-pages/text2pcap.html. (2014).
[2] 3GPP TS 23.272. 2012. Circuit Switched (CS) fallback in Evolved Packet System (EPS); Stage 2 (Release 10). (March 2012).
[3] 3GPP TS 23.401. 2014. General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access. (September 2014).
[4] 3GPP TS 24.301. 2011. Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS). (June 2011).
[5] 3GPP TS 25.331. 2014. Radio Resource Control (RRC); Protocol specification. (October 2014).
[6] 3GPP TS 36.211. 2016. Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation. (June 2016).
[7] 3GPP TS 36.212. 2017. Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding. (April 2017).
[8] 3GPP TS 36.300. 201. Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2. (January 201).
[9] 3GPP TS 36.304. 2012. Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) procedures in idle mode. (January 2012).
[10] 3GPP TS 36.331. 2016. Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification. (January 2016).

[11] 3GPP TS 36.508. 2017. Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Packet Core (EPC); Common test environments for User Equipment (UE) conformance testing. (April 2017).

[12] 3GPP TR 36.802. 2016. Evolved Universal Terrestrial Radio Access (E-UTRA); NB-IOT; Technical Report for BS and UE radio transmission and reception. (February 2016).

[13] MP Antenna. 2014. SUPER-M ULTRA Mobile Antenna (25MHz–6GHz). (2014). https://www.mpantenna.com/product/mobile-ham-radio-antenna-scanner/

[14] Jose E Vargas Bautista, Salil Sawhney, Mutaz Shukair, Ishwinder Singh, Vishwanth K Govindaraju, and Sandip Sarkar. 2013. Performance of CS Fallback from LTE to UMTS. *IEEE Communications Magazine* 51, 9 (2013), 136–143.

[15] Abhijnan Chakraborty, Vishnu Navda, Venkata N Padmanabhan, and Ramachandran Ramjee. 2013. Coordinating Cellular Background Transfers using LoadSense. In *Proceedings of the 19th annual international conference on Mobile computing & networking*. ACM, 63–74.

[16] Gerald Combs. 2012. Tshark — Dump and Analyze Network Traffic. *Wireshark* (2012).

[17] Erik Dahlman, Stefan Parkvall, Johan Skold, and Per Beming. 2010. *3G evolution: HSPA and LTE for mobile broadband*. Academic press.

[18] ShamROCK San Diego. 2019. ShamROCK Concert. https://www.sandiegoshamrock.com/. (2019).

[19] Suyang Duan, Vahid Shah-Mansouri, Zehua Wang, and Vincent WS Wong. 2016. D-ACB: Adaptive Congestion Control Algorithm for Bursty M2M Traffic in LTE Networks. *IEEE Transactions on Vehicular Technology* 65, 12 (2016), 9847–9861.

[20] T Engel. 2013. Xgoldmon. https://github.com/2b-as/xgoldmon. (2013).

[21] FCC. 2017. Communications Status Report for Areas Impacted by Hurricane Maria - September 21, 2017. https://transition.fcc.gov/Daily_Releases/Daily_Business/2017/db0921/DOC-346840A1.pdf. (21 September 2017). (Accessed on 09/25/2017).

[22] FCC. 2017. Communications Status Report for Areas Impacted by Tropical Storm Harvey, September 1, 2017. http://transition.fcc.gov/Daily_Releases/Daily_Business/2017/db0901/DOC-346475A1.pdf. (1 September 2017).

[23] Ismael Gomez-Miguelez, Andres Garcia-Saavedra, Paul D Sutton, Pablo Serrano, Cristina Cano, and Doug J Leith. 2016. srsLTE: An Open-source Platform for LTE Evolution and Experimentation. In *ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation, and Characterization*.

[24] GSMA. 2018. The Mobile Economy Report. https://www.gsma.com/mobileeconomy/wp-content/uploads/2018/05/The-Mobile-Economy-2018.pdf. (2018).

[25] Harri Holma and Antti Toskala. 2007. *WCDMA for UMTS: HSPA evolution and LTE*. John Wiley & sons.

[26] Byeongdo Hong, Shinjo Park, Hongil Kim, Dongkwan Kim, Hyunwook Hong, Hyunwoo Choi, Jean-Pierre Seifert, Sung-Ju Lee, and Yongdae Kim. 2018. Peeking over the Cellular Walled Gardens-A Method for Closed Network Diagnosis. *IEEE Transactions on Mobile Computing* 17, 10 (2018), 2366–2380.

[27] Raymond Kwan, Rob Arnott, Riccardo Trivisonno, and Mitsuhiro Kubota. 2010. On Pre-emption and Congestion Control for LTE Systems. In *IEEE 72nd Vehicular Technology Conference-Fall*.

[28] Yuanjie Li, Chunyi Peng, Zengwen Yuan, Jiayao Li, Haotian Deng, and Tao Wang. 2016. MobileInsight: Extracting and Analyzing Cellular Network Information on Smartphones. In *ACM MobiCom*.

[29] Seung-Hun Oh and Young-Han Kim. 2006. 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*. IEEE, 153–155.

[30] Anup Kumar Paul, Hidehiko Kawakami, Atsuo Tachibana, and Teruyuki Hasegawa. 2016. An AQM based Congestion Control for eNB RLC in 4G/LTE Network. In *IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*.

[31] Qualcomm. 2012. QXDM. https://www.qualcomm.com/documents/qxdm-professional-qualcomm-extensible-diagnostic-monitor. (2012).

[32] Napa Valley Register. 2014. Earthquake calls flooded 911 dispatch center. (20 September 2014). http://napavalleyregister.com/news/local/earthquake-calls-flooded-dispatch-center/article_13f12614-2589-5013-b456-c48aff5663a2.html/

[33] Ettus Research. 2010. USRP B210. http://www.ettus.com/all-products/UB210-KIT/. (2010).

[34] Microsoft Research. 2018. The 2018 Microsoft Airband Initiative. (December 2018). https://blogs.microsoft.com/uploads/prod/sites/5/2018/12/MSFT-Airband_InteractivePDF_Final_12.3.18.pdf

[35] SanDiego.org. 2019. St. Patricks Day Parade. http://www.stpatsparade.org/parade-festival-schedule.html. (2019).

[36] Paul Schmitt and Elizabeth Belding. 2017. Low on Air: Inherent Wireless Channel Capacity Limitations. In *ACM Workshop on Computing Within Limits*.

[37] Paul Schmitt, Daniel Iland, and Elizabeth Belding. 2016. SmartCell: Small-scale Mobile Congestion Awareness. *IEEE Communications Magazine* 54, 7 (2016), 44–50.

[38] Paul Schmitt, Daniel Iland, Mariya Zheleva, and Elizabeth Belding. 2016. HybridCell: Cellular Connectivity on the Fringes with Demand-driven Local Cells. In *IEEE INFOCOM*.

[39] D. Spaar. 2014. Tracing LTE on the Phone. http://www.mirider.com/weblog/2013/08/index.html. (2014).

[40] SRLabs. 2014. SnoopSNitch. https://opensource.srlabs.de/projects/snoopsnitch. (2014).

[41] Morteza Tavana, Ali Rahmati, and Vahid Shah-Mansouri. 2018. Congestion Control with Adaptive Access Class Barring for LTE M2M Overload using Kalman Filters. *Computer Networks* 141 (2018), 222–233.

[42] Pedro Torres, Paulo Marques, Hugo Marques, Rogério Dionísio, Tiago Alves, Luis Pereira, and Jorge Ribeiro. 2017. Data Analytics for Forecasting Cell Congestion on LTE Networks. In *2017 Network Traffic Measurement and Analysis Conference (TMA)*. IEEE.

[43] Neal Ungerleider. 2013. Why Your Phone Doesn't Work During Disasters - and How to Fix it. https://www.fastcompany.com/3008458/why-your-phone-doesnt-work-during-disasters-and-how-fix-it/. (17 April 2013).

[44] Xiufeng Xie, Xinyu Zhang, Swarun Kumar, and Li Erran Li. 2015. pistream: Physical Layer Informed Adaptive Video Streaming over LTE. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. ACM, 413–425.

[45] Xiufeng Xie, Xinyu Zhang, and Shilin Zhu. 2017. Accelerating Mobile Web Loading using Cellular Link Information. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 427–439.