Discovering Fine-grained RRC State Dynamics and Performance Impacts in Cellular Networks

Sanae Rosen, Haokun Luo,
Qi Alfred Chen, Z. Morley Mao
University of Michigan
{sanae, haokun, alfchen, zmao} @umich.edu

Jie Hui, Aaron Drake, Kevin Lau
T-Mobile USA Inc.*
{Jie.Hui, Aaron.Drake, Kevin.Lau}
@t-mobile.com

ABSTRACT

To conserve power while ensuring good performance on resource-constrained mobile devices, devices transition between different *Radio Resource Control* (RRC) states in response to network traffic and according to parameters specific to network operators. As RRC states significantly affect application power consumption and performance, it is important to understand how RRC state timers interact with network traffic patterns. In this paper, we show that the impact of RRC states on performance is significantly more complex and diverse than found in previous work. To do so, we introduce an open-source tool that allows the impact of RRC states on network and application performance to be measured in a robust and accurate manner on unmodified user devices, and deploy the tool in 23 countries around the world to test a broad range of cellular network technologies. We detect previously unknown performance problems which increase network latencies by up to several seconds and for LTE, can increase packet losses by an order of magnitude. Through an in-depth cross-layer analysis of several carriers, we examine the lower-layer causes of these problems. We determine that the highly complex state transitions of certain carriers, and in particular poor interactions between state demotions and network traffic, can lead to substantial, unexpected latencies.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: wireless communication; C.4 [**Performance of Systems**]: measurement techniques, performance attributes

Keywords

4G LTE; 3G; smartphones; RRC state machine; application QoE; cellular network performance

[1]The views presented here are as individuals and do not necessarily reflect any position of T-Mobile.

MobiCom'14, September 7-11, 2014, Maui, Hawaii, USA.

http://dx.doi.org/10.1145/2639108.2639115.

1. INTRODUCTION

Mobile clients' network traffic patterns cause cellular networks (such as 3G and 4G LTE) to transition between network states, known as *RRC (Radio Resource Control)* states. These states have different performance and energy consumption characteristics, and transitioning to a high-power state adds additional latencies. By using high-power RRC states only when necessary, and leveraging the temporal locality of network transmissions to avoid state promotion latencies, users can experience good network performance on resource-constrained mobile devices. Although the RRC states are largely defined by a set of specifications [7, 8], many aspects of the RRC state machine, such as timers for transitioning between states, are configured by the carrier.

Previous work [16, 22, 24, 23, 37] has measured RRC state configurations, either in controlled, in-lab experiments, or on specific device models which support logging RRC state transitions directly. They focus on measuring and inferring RRC state timers as they are implemented by the carrier, assuming that performance in those states fits a particular, consistent, ideal model. In this paper, we present a collection of measurement techniques and datasets that greatly improve our understanding of the impact of RRC state transitions, especially on application performance.

First, we present a tool that improves upon previous RRC inference approaches to make RRC measurements on real user devices *practical*, *scalable* and *robust*. This allows us to collect data on user-experienced performance directly, allowing us to capture non-ideal RRC state performance behavior, and uncover previous unknown performance problems.

To do so, we improve previous RRC measurement techniques to allow them to work effectively outside the lab and to run with no active user input needed. We account for interfering traffic and unrelated network congestion, and also measure the impact on application protocols such as HTTP and DNS requests directly. We present a snapshot of the data collected to date, covering 23 countries, and we are able to collect RRC state performance data on an ongoing, continuous basis. We make the tool available for carriers, researchers and other interested parties to use and adapt. Our method makes the ongoing monitoring of worldwide RRC state performance characteristics practical and eliminates the need for manual measurements or device-specific features.

Motivating the need for our global, systematic approach to measuring RRC state performance from a user perspective, we discovered a previously unknown cause of performance problems: *RRC state demotion delays*, not to be confused with the well-known *promotion* delays that occur when sending packets when the device is in a low-power state. The overhead of promotions has been explored in prior work, but the impact of the demotion process itself has been assumed to be non-critical. During state

demotions, when no data has been transmitted for several seconds, devices enter a lower power state. For some carriers, we discovered there can be a delay of up to several seconds when a packet is sent around the time the demotion process occurs (in addition to the delay incurred by the subsequent state promotion. For many carriers these demotion delays are often be the determining factor in the latency experienced by the user.

Next, to understand and verify the existence of the performance problems discovered, we perform an in-depth, cross-layer examination of the causes and application impact of state transitions. We examine the impact of RRC and RLC (Radio Link Control) messages on RRC state transition latencies for several carriers, using a tool called QxDM that logs these low-layer control and data plane messages on mobile devices [6]. We discover major differences between carriers in the implementation of RRC state changes and cell tower communication. In particular, the use of the optional FACH state for 3G networks as a performance optimization in many implementations actually leads to significant performance problems.

Additionally, we examine the impact of RRC states on higher-layer network protocols and Android applications. In addition to gathering the impact of RRC states on HTTP requests, DNS lookups and TCP connections, we also develop an application for in-lab testing of Android applications in order to systematically measure the impact of RRC states on user-perceived performance in major applications. In doing so, we demonstrate that RRC transition latencies — including the previously unknown demotion latencies — can have a substantial impact on user-perceived performance.

Maintaining up-to-date information on RRC state implementations and their impact on performance is especially valuable to carriers and app developers, such as those using tools like ARO [17] to allow them to optimize application performance and battery consumption. Furthermore, there has been interest recently by "power users" in understanding how issues such as RRC state implementations, as they differ among carriers, can affect performance [35, 18]. In this paper, we present a first dataset collected from this app over several months, and as this dataset continues to grow, we expect it to continue to be a useful resource for collecting data on RRC state performance.

We summarize our contributions as follows:

- We provide an open-source RRC inference framework that measures the impact of RRC state transitions on user performance. Unlike previous approaches, it can be run on any unmodified Android device and network type, allowing RRC state characterization to be easily crowdsourced.
- We survey how RRC states impact performance in carriers in 23 countries and provide an open data set of the results.
- We uncover previously unknown, severe latency problems that exist in many cellular network technologies and carriers around the world.
- Using cross-layer analysis, we investigate how unrelated RRC and other low-layer control plane messages and delays cause higher-layer state transition latencies.
- We measure the impact of RRC states on application latencies as a whole, demonstrating that RRC states, especially transitions, have a significant impact on application-level latency as well as individual packets.

We start by giving an overview of RRC state machines (§2) and related work in RRC state measurement and mobile measurements (§3). We then describe our measurement methodology (§4), including our inference approach for the global deployment and the approach used for cross-layer local experiments. We then discuss our global results (§5) followed by an in-depth examination and confirmation of results from several carriers (§6). Finally, we examine the impact on application performance (§7) and discuss the implications of our finding and future work (§8).

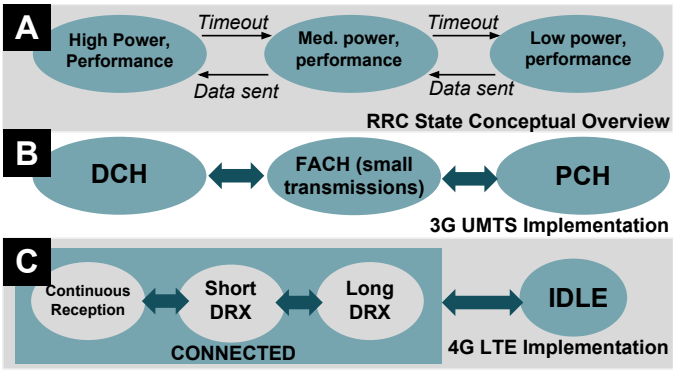

Figure 1: A: Overview of RRC state machine design. B-C: possible 3G and 4G state machines.

2. BACKGROUND

For cellular network protocols, there is a tradeoff between latency and battery consumption. Figure 1A gives a conceptual overview of how this tradeoff is managed. Mobile devices do not maintain a constant, active network connection due to their limited battery life, and switch to a high-power, active state to send data. Because this transition incurs additional latencies, the device remains in this state for several seconds, since network traffic often comes in bursts. There may also be an intermediate state where small amounts of data can be transmitted without the high power consumption of the fully active state.

These are known as *RRC States*, and are defined by 3GPP specifications [7, 8]. Carriers may configure their RRC state machine timers differently, subject to the constraints of the protocol specification. For 3G network technologies [7], there are two to three main states: DCH, which is high-power and high-bandwidth, FACH, an optional state which is low power and can only transmit a small amount of data before entering FACH, and PCH, where no transmission is possible. An example of a 3G state machine is shown in Figure 1B. For 4G LTE, as shown in Figure 1C, there are two main states: CONNECTED, a higher-power state, and IDLE, a lower-power state where no data is transmitted. CONNECTED often has sub-states where the device is active only at intervals of tens or hundreds of milliseconds after a few hundreds of milliseconds of idle time. This is known as *Discontinuous reception*, or DRX.

It is known that RRC timers can have a substantial impact on application performance and power consumption. In particular, periodic messages may be affected by long promotion latencies and lead to the device being in a high-power state longer than needed. The latter problem can be addressed in part through *fast dormancy* [19], where the device transitions to a low-power state early when no additional data transmissions are expected. We show in §5 that fast dormancy is rarely enabled in practice, perhaps due to the added complexity of implementing such a system and problems with certain implementations [29, 19].

In this paper, we explore how RRC timers impact performance in depth. It is known that state *promotions*—moving from a lower-power state to a higher-power state—involve additional latencies. We refer to these latencies as *promotion latencies*. We

Table 1: Summary of results in figures and tables

Section	Name	Key finding	Dataset
§4 (Methodology)	Table 2, Fig. 3	Validation of inferred RRC timers	CONTROLLED
§5 (Results)	Fig 5, 6, 7, 8	State transition delays for all network types can be substantial.	GLOBAL
§6 (Root causes)	Fig 9, 10	Causes of LTE transition delays.	COTNROLLED
§6 (Root causes)	Fig 11	Causes of 3G transition delays.	CONTROLLED
§7 (App impact)	Fig 12, 14, 13,	RRC states affect a variety of application and transport protocols.	GLOBAL (12), CONTROLLED

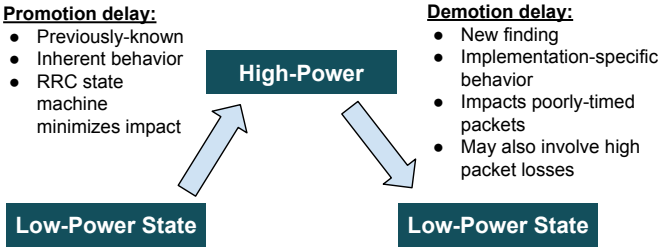

Figure 2: Comparison of promotion delays with newly-discovered demotion delays. Demotion delays occur when packets are delayed or lost when sent during the RRC state demotion process.

perform a cross-layer, experimental examination of variations in these latencies across carriers, and how implementation differences among carriers lead to these variations. We also discover that the impact of *demotions* on latency can also be quite substantial, which have previously been disregarded. Moving from a high-power to a low-power state in some cases takes several seconds, and for LTE may significantly increase the packet loss rate. We summarize the differences between these two delays in Figure 2.

3. RELATED WORK

Previous work examined power and performance characteristics of RRC state machines in both 3G [16] and 4G LTE networks [22, 24] in controlled environments, as well as specific features of those networks such as DRX [23]. Work by Souders [35] estimates RRC state machine performance through a web app, at a coarser granularity and without accounting for background network activity on phones. Recently, RILAnalyzer [37] leveraged chipset-specific functionality to monitor 3G RRC state transition events directly and measure how often applications cause excessive RRC state promotions. Unlike previous work, we focus on measuring and understanding how dynamic, non-ideal RRC state transition behavior varies and cause different performance trends on different carriers around the world.

Examining lower-layer control messages specifically, a Qualcomm whitepaper [27] explains how control plane messages in different network technologies are expected to lead to different promotion latencies. We also examine the demotion delays and take an experimental approach to determining the performance impact of RRC state transitions on different devices, uncovering new sources of delay. Work by Li *et al.* [33] examines the overhead of 3G transitions for one carrier, focusing on the number of signaling messages and on state promotions, rather than state demotions. Work by He *et al.* [21] investigates the impact of device type on low-layer control and data plane overhead. These papers demonstrate the importance of examining the impact of RRC control plane messages on performance.

Motivating our work, there has been a great deal of interest in understanding how applications can improve performance by accounting for RRC state timers, especially by temporally clustering network traffic. ARO [17] presents a tool for optimizing application performance, which accounts for poor interactions between applications and RRC state machines. TailTheft [20] prefetches or delays traffic to reduce the amount of traffic sent in high-latency states. Work by Lagar-Cavilla *et al.* [25] also proposes transmitting delay-tolerant traffic immediately after other data transmissions. Work by Evensen *et al.* [15] predicts when state promotions happen and schedules data accordingly to decrease latency. Work by Aucinas *et al.* [10] demonstrates the high costs of intermittent application transmissions. RadioJockey [31] investigates how to effectively trigger fast dormancy based on network traffic patterns, and TailEnder [28] and work by Deng *et al.* [34] propose a method of scheduling data transfers to minimize energy consumption without impacting user-perceived performance.

Our more accurate, per-carrier RRC state performance model would allow these tools to better determine the impact of RRC states on application traffic patterns and suggest performance improvements. In particular, we find that sending packets around state demotions should be avoided.

More generally, our work is related to efforts on measuring performance characteristics of cellular networks from the perspective of mobile devices. The Livelab project [12] also makes use of users running a measurement app on their phones. A wide range of findings on how users interact with mobile devices have been published, including measuring web usage in the wild [11]. Work by Halepovic *et al.* [14] presents a method of passively measuring HTTP transaction latency. Work by Gember *et al.* [9] determines how to accurately measure user-perceived performance on user devices. JamLogger [5] is an ongoing project to collect general performance and user activity on mobile devices. Unlike these projects, the mobile device measurement component of our work focuses on RRC performance.

4. MEASUREMENT METHODOLOGY

To understand RRC state performance, particularly the impact of RRC state transitions, we use three complementary approaches to develop a cross-layer understanding of RRC performance problems, their causes, and their impacts on application performance. First, we collect data on the impact of RRC state transitions on performance from carriers worldwide using an open-source cellular network testing tool for Android (§4.1). We then use local, controlled experiments to investigate the performance impact of RRC states. Starting at the RLC (Radio Link Control) layer, we examine control and data messages directly using a tool called QxDM (Qualcomm eXtensible diagnostic monitor) [6], in order to understand the causes of the observed transition delays (§4.2). Finally, we built an application controller tool in order to

test the impact of RRC states on user-perceived performance at the application layer (§4.3). We describe approach in turn below.

We produce two datasets to analyze RRC state transitions. One, which we call GLOBAL, is gathered from user devices worldwide. The other consists of data from various global experiments, which we call CONTROLLED.

4.1 Automated RRC Performance Measurement

Previous work has inferred RRC state timers by observing how packet latencies change as the time between packets increases [16, 22], and other work has looked at directly monitoring RRC state timers on certain chipsets [37]. Unlike previous work, we focus on *user-perceived performance* in addition to inferring the timers set by carriers. Measuring RRC state timers directly does not capture the performance impact of RRC transitions, and inferring RRC timers in a controlled environment does not allow RRC states to be measured on a large scale. We overcome these challenges, allowing us to deploy a RRC inference tool worldwide.

In a controlled environment, the standard technique used is as follows [16, 22]: first, a UDP packet is sent to ensure the device is in a high power state. Next, the device is left idle for a period of time before another UDP packet is sent and echoed back by the target server. The latency of the second packet can then be compared to the latency of the first packet; if there is a substantial increase, it implies that a state promotion has occurred between them, adding latency. By examining a range of inter-packet intervals, the time at which a state transition occurs after a packet is sent can be determined. To distinguish between PCH and FACH, where a transition only occurs for sufficiently large packets, we perform this test with empty packets and 1 KB packets.

We modify this technique so it can be automated and so we can crowdsource the measurement of RRC state parameters and performance globally, as several tools already do with latency and throughput [4, 1]. This allows us to detect global carrier-dependent effects, to capture probabilistically occurring problems (whose presence we then confirm through controlled experiments), and to detect any differences over time, among different device types, or different geographic regions.

There are two main challenges in supporting crowdsourced measurements: dealing with interfering traffic on the device, and dealing with unrelated network congestion and variable latencies in different geographic locations. First, knowledge of all network traffic on the device is needed. A Linux utility (*/proc/net/dev*) can be used to monitor the total traffic on the device. Tests were discarded and rescheduled when more traffic than expected was observed. Second, data collected on user devices is not as "clean" as that collected in the lab. Changing network congestion and highly lossy networks can lead to variations in performance that are unrelated to RRC states. We consider only networks that have at least 5 complete measurement results, and before identifying RRC state transitions, we eliminate outliers and consider the average latency for each inter-packet interval. Furthermore, all data shown in the paper is normalized by subtracting the baseline latency where no RRC state change occurs. The application also monitors how much data and power it consumes to avoid exceeding limits placed by the user, and can pause tests when the network type changes (such as when switching to WiFi).

To measure RRC states on a large scale, we added this method to MobiPerf [4], an open-source network measurement tool for Android devices, and released it to the public. This RRC test runs automatically in the background to allow ongoing monitoring with no user involvement, and sends results back to a server, along with information such as the carrier and signal strength at the time the measurements were taken. User identifiable data is anonymized. To observe the impact of RRC states on HTTP requests, DNS lookups, and TCP handshakes, we sent the request in question after a large UDP packet followed by a varying time interval, and measured the latency of that request. Furthermore, as we measure RRC states repeatedly over time, we are able to observe dynamically configured timer values varying from test to test, such as those resulting from fast dormancy. We refer to the dataset we produce using these experiments as GLOBAL throughout the paper.

	Demotion type	App	QxDM
3G	C1 DCH⇒FACH	3 ± 0.5 s	3.1 ± 0.1 s
3G	C1 FACH⇒PCH	6.5 ± 0.5 s	6.2 ± 0.8 s
3G	C2 DCH⇒Disconn.	10 ± 0.5 s	10.3 ± 0.1 s
	— fast dormancy	3 ± 0.5 s	3.2 ± 0.1 s
LTE	C1 Conn.⇒Idle	10 ± 0.5 s	10.5 ± 0.1 s
LTE	C2 Conn.⇒Idle	10 ± 0.5 s	10.2 ± 0.1 s

Table 2: Comparison of ground truth demotion timers from QxDM with values measured through the application.

4.2 Layer 2 Root Cause Analysis with QxDM

QxDM [6] is a debugging tool that can view all network data and signaling messages in the form of a pcap-like trace. Using this tool, we can map IP packets to RLC PDUs (Packet Data Units), which are layer 2 data-plane messages. QxDM produces detailed logs of all these events and the corresponding timestamps. We parse these logs and map lower layer messages to the UDP packets sent. By performing the test repeatedly we can eliminate messages that do not occur for each transition and calculate the average times between key events. We confirm this mapping by comparing bytes in the RLC PDUs and the packets sent.

Using these logs, we determine how RLC PDU delays and low-layer control-plane messages affect user-visible performance around RRC state transitions. We combine pcap traces with QxDM logs to determine what RLC events surround IP packet transmissions, using timestamps to match events at both layers. For 3G, we are able to map individual PDUs to IP packets as the contents of PDUs are logged.

To perform this analysis, we use a modified RRC state testing application which repeatedly cycles through inter-packet intervals in order to induce RRC state transitions. By analyzing the resulting trace, we can determine which of the control messages related to each RRC state transition result in substantial delays. We also determine if any non-RRC state transition related processes are interrupting state transitions. We analyzed traces from two different carriers with different RRC state implementations and different transition delay behavior, and performed some of the analysis on a third carrier. In §6, we break down the causes of various RRC state delays, and compare carriers with differing delay behavior in order to both validate the presence of the observed differences and previously unknown delays, in addition to understanding their causes.

A limitation of this approach, unlike the app-based approach, is that it cannot be performed on actively-used devices in the wild, as proprietary software, specially configured devices, and some external equipment is needed. It is complementary to the app-based approach, which allows for a broad survey of RRC state performance to be performed, covering many carriers, locations, and device types. This approach is more suitable for in-depth examination of specific performance problems, and is likely of

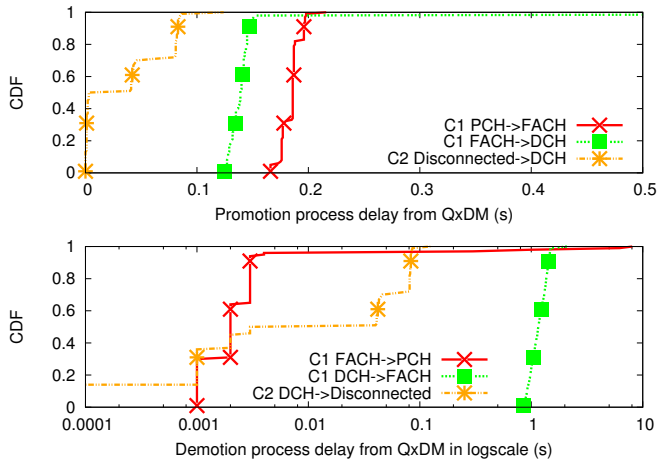

Figure 3: Measurement of demotion and promotion delays for two carriers in QxDM.

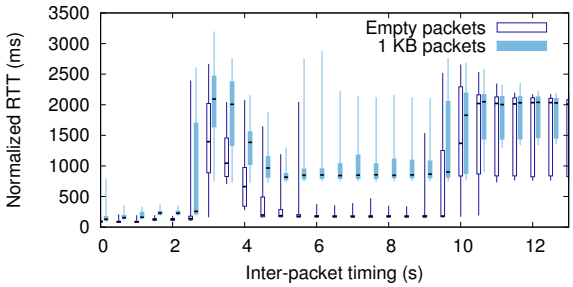

Figure 4: Example of observed round-trip times while transitioning through RRC states. Median, quartile and 5%/95% values shown. Data normalized by subtracting the baseline RTT in the highest power state.

most use to carriers who, having detected a performance problem, are interested in understanding how best to address it.

Finally, in order to validate the application-based RRC state measurement methodology, we use QxDM to determine a ground truth for RRC state timers. After determining timers from two carriers for RRC states, we then verify the values by comparing the inferred RRC timers with the ground truth values from QxDM, shown in Table 2. As we infer the RRC timers set by the carriers from changes in the measured performance, RRC demotion delays — which result in elevated and variable latencies during RRC demotions — often obscure the precise timer configured by the carrier, so these values can only be inferred to within about a second. This limitation applies only to inferring the *demotion timers*, not to be confused with the *demotion or promotion latency*, which we measure at the millisecond granularity. In this paper, our goal is primarily to measure how RRC states affect performance in practice, and since we found that RRC state demotions are not an instantaneous process, the impact of RRC states on packet latencies does not appear at the precise moment of the underlying state transition.

We also confirm the presence of the long demotion delays observed in our application measurements. We focus on two major carriers with over a hundred million subscribers each, which we refer to as C1 and C2. For some of our analysis, we also examined a third major carrier, C3. In Figure 3, we show C1 has a substantially longer RRC demotion process delay than C2, which prevents packets from being sent during that time and results in significantly longer transmission delays at the application layer. We evaluate this in §7. We refer to the dataset produced by these local, controlled experiments as CONTROLLED throughout the paper.

4.3 Application Controller

We developed an application controller which simulates user behavior on major Android applications such as Facebook. This controller enables us to systematically evaluate the impact of RRC state transition delays on user-perceived application performance in §7 through a cross-layer analysis framework. Built upon the Android Test Case framework [2], this controller programmatically triggers Android UI events such as clicking buttons and entering text. To measure user-perceived UI latency, the controller also logs UI events, such as the start and end time of the news feed loading. As measured by Android DDMS [3], an application performance profiling tool, our controller incurs a computational overhead of less than 2% and thus has minimal impact on the latency measurements. This also contributes to the CONTROLLED dataset.

5. GLOBAL PERFORMANCE MEASUREMENTS

Our RRC state measurement approach allows any Android device on any cellular network to measure the impact of RRC state and state transitions on user-perceived performance. In §4.1, we describe how our improved RRC inference methodology allows the tool to be deployed on uncontrolled user devices worldwide, and in this section we make use of data collected by that tool (the aforementioned GLOBAL dataset.) This tool measures network performance automatically and in the background on Android devices, allowing users to effortlessly monitor performance trends. Users can limit the amount of data consumed, and all data sent to the server is anonymized to protect the user's privacy. Promotion delay trends are roughly consistent with those in previous work [16, 22], although we have confirmed with the authors of the work that many of the timers for the carriers have changed. Our results are also consistent with more recent work that measures some RRC state changes directly, although we could not directly compare the anonymized carriers [37]. Our inference tool allows our knowledge of RRC timers to easily be kept up to date, and allows us to investigate a much larger dataset of carriers. In doing so, we uncover previously unknown demotion delays.

In Figure 4, we give an example of measurements from one device type and carrier (packet sizes do not include headers). The round-trip time for large packets is higher for inter-packet frequencies between 4–9 seconds, which is a characteristic of FACH. The round-trip time for both packets increases substantially for intervals greater than 10 seconds, which is characteristic of PCH. We also found that even in PCH, large packets still have a larger round-trip time than small packets, due to network delays. In this graph, as in all graphs of this type, we average values over several tests and then subtract the median latency in the highest power state in order to eliminate the effects of network latency.

We observe behavior inconsistent with the ideal model of RRC state transitions that has been discussed in the past. In Figure 4, there is a period of several seconds after a packet is sent, from about 2.5 to 4 seconds, when the next packet experiences unexpectedly high round-trip times. We refer to this delay as the *demotion delay*, and the period of time where it occurs as the *demotion period*. Where there are no demotion delays, we instead identify the interpacket interval at which the demotion occurs to be the

promotion period. For example, in Figure 4, a demotion also occurs between 9.5 and 10 seconds.

Dataset Overview: Our dataset consists of 650 000 sets of RRC tests in total at the time of writing, and we continue to collect more data. Each measurement set includes the results of a set of measurements with interpacket intervals from 0 to 15 s, increasing by half-second increments. We repeat this test with both empty and one kilobyte packets, and we collect the round-trip time, the number of lost packets, the signal strength, and metadata about each measurement, including the carrier, manufacturer, OS, and a coarse-grained location. We also measure the impact of RRC states on HTTP, DNS and TCP requests in a separate set of tests that occurs less frequently, as it is more data and power intensive. In this set of tests we also examine the effect of varying packet sizes on RRC state performance.

Due to lost packets, interrupted measurements and unrelated network delays, a single set of tests was usually insufficient. For our final results, we consider carriers with more than 5 complete tests only, so that transient network delays, unrelated to RRC, will not affect our results. We also excluded six carriers where network noise was so high that all RRC states were indistinguishable. After filtering out carriers with insufficient data for our analysis, we analyzed 44 carriers in 23 countries covering every continent. Data on 69 distinct device model types and seven distinct network types was collected, including 2G, 3G and 4G technologies. In this paper we focus on 3G and 4G, which have been adopted by most carriers. 7 carriers use LTE, 23 use HSPA+, 16 use HSPA, 25 use HSPDA, and 6 use EVDO_A, with many carriers supporting more than one technology. In particular, most carriers with LTE also provide 3G.

Carrier and device characteristics: Almost all carriers with LTE have a demotion timer to CONNECTED of 10 seconds, but with 3G technologies, the timers vary greatly, from 2 to 10 seconds. Carriers providing multiple 3G technologies generally use the same timers for each. About 2/3 of carriers with HSPA, HSPDA or HSPA+ have no FACH state, or at least no FACH state with a measurable performance impact. We only saw definitive evidence of fast dormancy — a demotion timer varying substantially from test to test — with one carrier. Two more carriers exhibited variations of about a second. As fast dormancy and other dynamic RRC state timer approaches become more prevalent, the ability to measure these variations will become increasingly valuable.

For 3G, we also examined the impact of packet sizes on RRC state transitions, by varying the packet payload size from 0 to 1000 bytes by increments of 200 bytes. Dramatic increases in latency, indicating the size threshold for a promotion from FACH, all occurred between 0 and 200 bytes. Given the small threshold for a promotion from FACH and the high associated demotion overheads, FACH may not provide much benefit. We also observed that RTTs increase steadily with packet size by as much as a few hundred milliseconds in all states.

Transition delays: Ideally, the overhead of acquiring radio resources to use the radio channel should be fairly constant, independent of the idle time of the device. We observed that when network transmissions are sent when the demotion timer expires and the device undergoes a state demotion, there is an unexpected and undesirable increase in the delay to promote back to a high power state. This problem occurs for a large number of carriers.

We start by describing detailed results from three major carriers, illustrative of three different observed behavior patterns, and then summarize results globally. Values are normalized by the average latency in the absence of any RRC state change. Round-trip times during state transitions are shown separately.

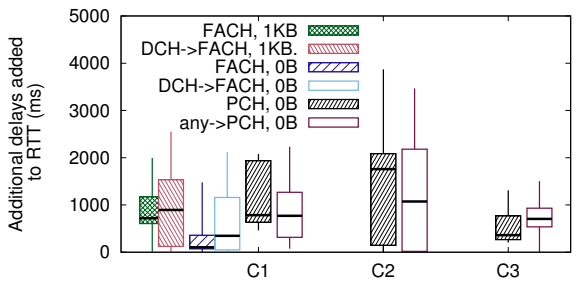

Figure 5: Variations in delays due to 3G states and state transitions, normalized against the DCH RTT. Median, quartile and 5th/95th% values shown.

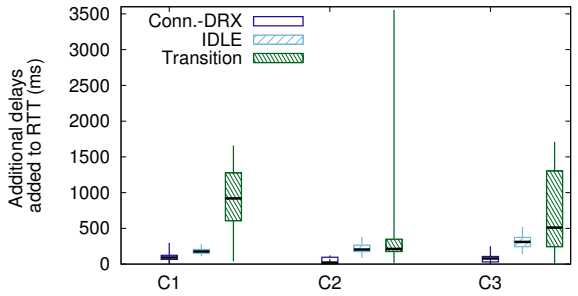

Figure 6: Delays due to LTE states and state demotions, normalized against the RTT of an empty packet sent in CONNECTED with no DRX.

In Figure 5 we show results from all 3G technologies for each carrier. As described above, we separate measurements into categories of RRC states and transitions based on observed, consistent changes in RRC states, such as those in Figure 4.

C1 makes use of FACH. When sending packets in FACH, latency becomes higher for larger packets, as expected. However, when sending packets during the transition from DCH to FACH, round-trip times are both higher and more variable. This is especially noticeable for small packets. Performance in PCH is also worse, but the demotion to PCH does not lead to additional latencies.

C2 does not implement FACH and does not always experience observable demotion delays, although when network performance is otherwise poor (including in some local experiments), demotion delays for this carrier appear. Network performance when sending data from low-power states is worse for this carrier than for others, although network performance for this carrier was generally poor. Finally, C3 is a CDMA network and thus does not implement FACH, but still experiences noticeable demotion delays.

In Figure 6, we compare LTE performance for the three carriers, comparing CONNECTED against IDLE and against the demotion period from CONNECTED to IDLE. LTE is supposed to perform better than 3G, but this is not necessarily true during demotions. For all three carriers the tail latency is substantially higher during state transitions, lasting potentially up to several seconds. For C1 and C3, the median values are substantially higher as well. In §6, we discover this difference is due to differences in how state transitions in these carriers are affected by control-plane activity. Different state change implementations, in this case, have both lead to performance problems.

For LTE, we also found packet loss rates during state transitions are higher than average, by up to an order of magnitude. To measure loss rates, we sent out ten empty packets simultaneously

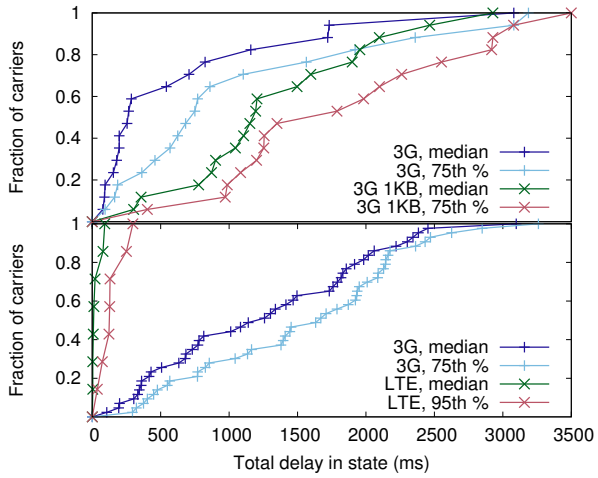

Figure 7: CDFs of promotion delays in each RRC state over all carriers, when promoting from FACH (upper graph) and from IDLE/PCH (lower graph).

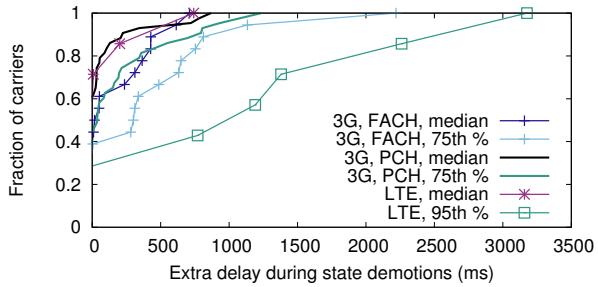

Figure 8: CDF over all carriers of additional latencies caused by transmissions during state demotions (minus promotion transition times in the new RRC state).

and counted how many were echoed back. C1, C2 and C3 experienced packet loss rates of 26%, 63% and 68% respectively. Normal loss rates were 1–3% depending on the network state (aside from C3 which experienced loss rates of up to 30% of packets).

We next examine trends across all carriers, starting by examining the impact of state *promotions* in Figure 7, which have been examined in previous work only for a small number of carriers. Promotions from FACH generally take several seconds, and are often triggered even when an empty packet is sent, meaning users get no performance benefit from FACH in these cases. Promotion times from PCH can also be long, and vary greatly from carrier to carrier. LTE promotion delays are generally no more than a few hundred milliseconds.

In Figure 8, we show the additional latencies added by attempting to send data near a state demotion, on top of the state promotion delays. We compare median values during state demotions with median values for state promotions only, and likewise for 75th and 95th percentile values. Ideally, no additional latency should be incurred, if the demotion is aborted, allowing the device to simply remain in the high-power state. This is not the case, especially for demotions to FACH. Eliminating FACH (as many carriers have) would likely reduce, though not completely eliminate, the prevalence of this demotion delay problem.

For LTE, median latencies are generally not affected by state promotions. It seems our local carriers explored above have somewhat atypical behavior, underscoring the need for a broad

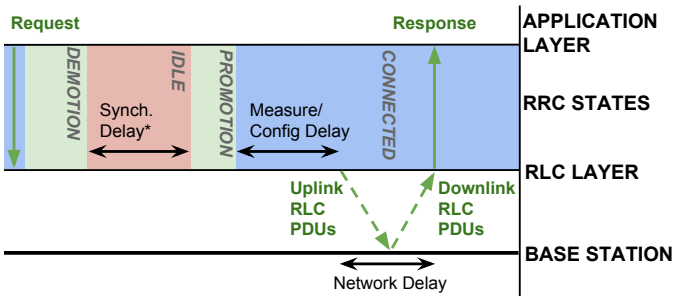

Figure 9: A cross-layer overview of RRC-related sources of transition delays for LTE. * indicates a delay present in only some carriers. The situation for 3G is similar, with longer demotions and promotions, and longer measurement delays.

survey of network performance. However, tail latencies are frequently affected. Note that we show 95th percentile latencies and not 75th percentile latencies. As we saw earlier, these tail latencies are substantially higher during demotions than any other time. Given the low network delays in LTE generally, these delays can have a major relative impact on user-perceived performance. As major web services go to great lengths to reduce tail latencies for 0.01% of users due to the potential revenue impacts [13], these latencies can be quite significant.

Finally, we investigated whether other factors lead to differences in RRC state timers among different carriers. We did not observe differences in RRC state configuration for the same carrier in different locations within a single country. In most cases, timer configurations were the same in different countries for international carriers, with the exception of two cases where subsidiaries of the same company operating in different countries had different timers in those countries. We did not detect differences in RRC state machines by device type, either. Unfortunately, using client measurements we could not directly confirm differences between different carrier equipment vendors.

Summary: State demotion delays are common worldwide, though not experienced by every carrier, and occur in both 3G and LTE. They can have a critical effect on performance, in some cases causing delays of several seconds; state demotion delays (and to a lesser extent, state promotion delays) can add additional latencies of up to several seconds on top of the normal state promotion latency. Additionally, in LTE, state demotions are associated with high packet loss rates. More generally, we have shown that running RRC state performance tests on user devices is an effective way of monitoring global RRC state performance trends.

6. BREAKDOWN OF RRC TRANSITION DELAY CAUSES

Through controlled, in-lab testing of RRC latencies, we examine the events that contribute to RRC state delays, using the methodology described in §4.2 to produce some of the CONTROLLED dataset. We collect measurements through QxDM, which provides detailed visibility of control messages related to RRC state transitions and RLC data messages. Consistent with previous work in this area [33, 21] we find promotion delays can be quite significant, although we focus on user-perceived latency rather than signaling overhead. We also determine that the overhead of state demotions can be significant, which has been overlooked. Although delays during state transitions occur in nearly all cellular network technologies, the causes of these delays (as well as their magnitudes) differ as shown in §5.

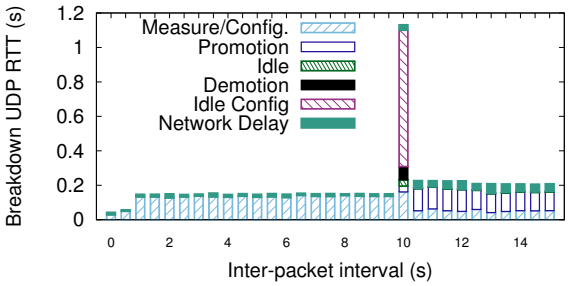

Figure 10: Median values of sources of LTE transmission delays for C1, using QxDM logs to determine the timing of layer 2 events. C2 is similar but lacks "Idle Config." The state demotion delay can be seen at 10 seconds.

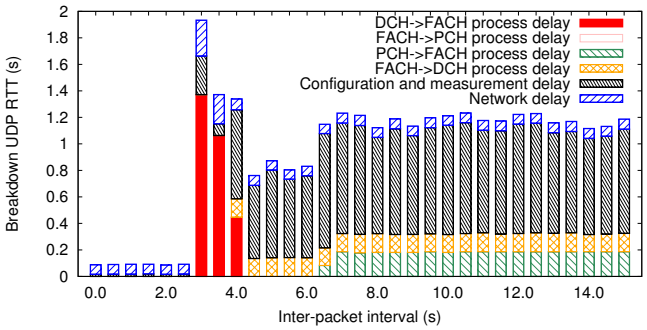

Figure 11: Breakdown of RTTs for varying inter-packet intervals, including a demotion to FACH at 3s and a demotion to PCH at 6.5s, based on QxDM logs. The demotion to FACH from DCH results in higher delays.

Breakdown of promotion delays in LTE: First, we examine state promotions that do not occur in the vicinity of state demotions. Although previous work has identified that state promotions cause delays [16, 22], the root causes of these delays and variations in these delays have yet to be examined. We summarize the median delays for different inter-packet intervals in Figure 10, and give an overview of the events involved in RRC state demotions and promotions in Figure 9. These include all delays that appear when sending a UDP packet, both delays in the network and lower-layer processing delays on the device.

We found that for LTE, state promotions from IDLE add a highly varying delay to the overall latency. This delay includes the effects of Discontinuous Reception (DRX), where devices will only send data during a small window of time, with a period of a few hundred milliseconds. In the messages logged by QxDM, the promotion process begins with a request to switch to a higher-bandwidth, more reliable channel. This message is sent on an unreliable network channel, so delays during this process due to poor network conditions can substantially increase the overall time to process a packet. This contributes significantly to the high variation in worst-case or tail network latency seen in Figure 6. A detailed description of some of the messages involved in state promotions (though not demotions) for LTE and 3G can be found in a white paper by Mohan et al. [27].

Breakdown of demotion delays in LTE: In Figure 10, it can be seen that packets sent during state demotions are accompanied by a large number of measurement and configuration messages before they occur, leading to higher latencies, although the state transition itself is a short process. Although there is a subsequent promotion after the demotion occurs, we found that it is generally quite short, since the promotion occurs immediately, without going through a DRX cycle.

We have isolated one set of message delays in particular that can add several seconds of delays, labeled "Idle config." These messages appear to be related to transmission synchronizations with the base station, although they are not well-documented. If, during a demotion, an IP packet is sent before this message appears, then the entire configuration process completes before the state transition process begins, leading to long delays. However, if an IP packet is sent after it appears, then this process is aborted and a state transition begins right away, so this delay only appears for a narrow window of inter-packet timings. This illustrates the dependencies of control messages on the data packet timing.

This problem appears to be implementation specific and not due to any problem with the LTE specification. These messages do not appear for C2, which explains the lower median latencies during demotions seen in Figure 6. Additionally, for all carriers, the device occasionally momentarily disconnects from the network before selecting a new cell tower, causing long delays. This appears to be responsible for the long 95th percentile latencies seen in Figure 6. This is likely unavoidable as user movement or poor network performance may necessitate this switch.

Breakdown of promotion delays in 3G: We summarize the breakdown of latency causes in Figure 11 as they vary by inter-packet interval. For 3G, promotion times are often longer — roughly 1200 ms on average where they occur. After a state promotion, additional control plane messages are sent, such as messages to measure channel conditions. These messages take up significantly more time than the state promotions themselves, and the messages seen can vary. One series of system information messages adds additional delays of hundreds of milliseconds where it occurs, leading to high latency variations. This set of messages occurs periodically, every few hundred milliseconds, not just during state transitions. Overall, state promotions in 3G are more complex and involve more messages being exchanged. 3G state promotions are already known to be slow for this reason [27, 33].

Breakdown of demotion delays in 3G: In Figure 11, it can be seen that state demotions have a substantial impact when the inter-packet interval is between 3 and 4 seconds. Unlike with LTE, it is simply the demotion process itself which can be slow, rather than other control plane messages which cause unexpected delays. This makes promotion latencies more common as well as affect a larger range of inter-packet intervals. It is also interesting to note that when a state demotion is interrupted by a packet being sent, there is often no subsequent promotion delay.

Interestingly, several carriers appear to lack these demotion delays altogether. We were able to examine one such carrier in depth using QxDM. This carrier's demotion process is substantially simpler, consisting of sending one message to the base station followed by a small amount of additional delay due to device configuration operations. As this adds a median time delay of 175 ms, it did not have a statistically significant effect on the user-experienced latency. This suggests that this carrier, and likely the others which lack demotion delays, are using a different RRC state transition implementation. This carrier also omits the FACH state, although we found in our global study of 36 carriers that not all carriers which omit FACH lack significant transition delays.

Summary: We have determined that, for both LTE and 3G, carrier-specific, RRC state related messaging and configuration delays can interact poorly with certain network state patterns. At least one carrier has been able to reduce these delays greatly. In general, LTE's state transition procedure ensures much better

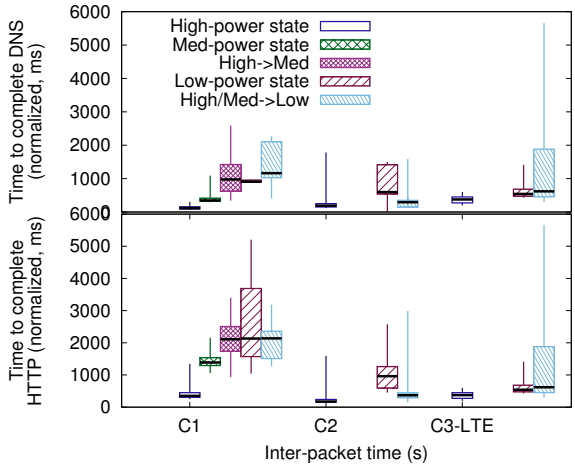

Figure 12: Performance of different carriers with different inter-packet timings, for DNS lookups and HTTP connections to a small website.

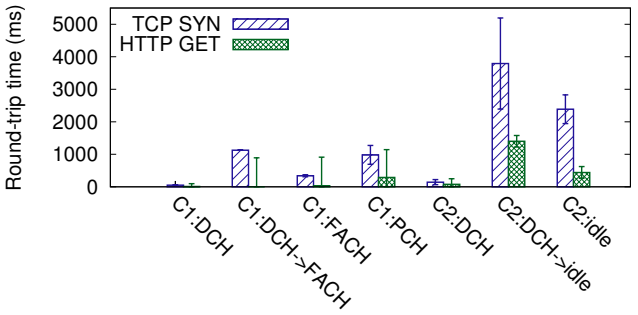

Figure 13: Effect of RRC states on TCP SYN RTTs and HTTP GET latencies.

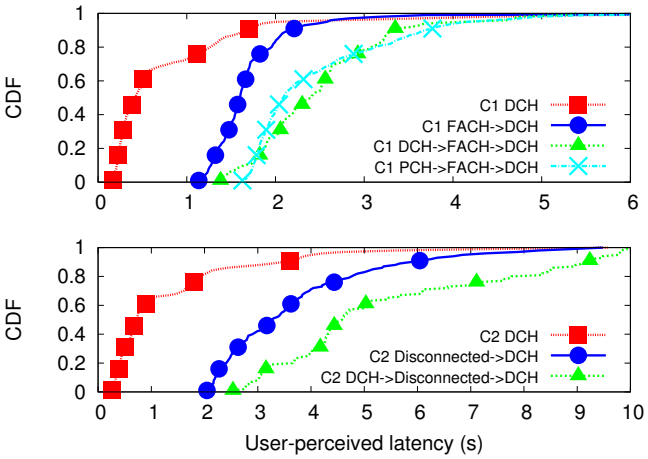

Figure 14: Effect of RRC states and transitions on user-perceived latency in web browsing experiments.

average performance than 3G's, largely due to a lower amount of control-plane signaling needed in order to transmit data or change RRC states. Delays in LTE are primarily due to poor interactions between certain control-plane messages and the state demotion process, affecting only a subset of requests (although it can add delays of several seconds). Delays in 3G, however, are generally due to issues with state demotion implementations. Additionally, while it was already known that state promotions can cause network delays, we experimentally quantify which components of the state promotion process lead to promotion delays. The overall observation is that RRC state transitions contribute significantly to tail latencies on mobile devices.

7. APPLICATION IMPACT

In this section, we explore how upper-layer protocols are affected by RRC state transitions, using both our globally deployed RRC state measurement tool and in-lab controlled experiments. We find that HTTP connections, DNS lookups, and mobile applications can all be significantly impacted by RRC state transitions.

7.1 HTTP and DNS Results from Global Deployment

In our public deployment (*i.e.,* the GLOBAL dataset), we measured the impact of RRC states on DNS and HTTP requests, which are more representative of real network traffic than individual UDP packets. Testing with UDP packets allows us to understand the impact of RRC states without being affected by network protocol features, but UDP is not representative of most network traffic. In Figure 12, we show the impact of RRC state on a DNS lookup and on the loading of a small web page in our global study, focusing on the carriers discussed in-depth before.

The behavior we observed for these tests was consistent with that we observed for UDP packets, exhibiting the same performance patterns. The completion times for DNS lookups and HTTP requests for C1 were strongly affected by state demotion delays in 3G, whereas C2 was less affected. For C1, FACH unsurprisingly performs worse than DCH. Data sent during the demotion to FACH performs comparably to the performance in PCH. Performance during the demotion to PCH is not significantly worse than in PCH proper. Also consistent with our UDP results, we found that for C2 there were no state demotion delays. For LTE, the tail results during state demotions are substantial, in one case lasting more than five seconds for a DNS lookup.

7.2 Controlled Web Browsing Experiments

To verify our findings, we also examined RRC state delays in different circumstances in controlled, in-lab experiments, contributing to our CONTROLLED dataset. We evaluated the page loading time in a browser for 10 major websites, including search, social networking, e-commerce, news, sports and finance websites. We varied the inter-request time from 1s to 11s, with a granularity of 0.1s. In total, we generated 3000 HTTP requests for both C1 and C2 over an entire day. In Figure 13, we show the TCP SYN RTTs and the HTTP GET request RTTs from TCP flows. RRC state demotion delays increase the SYN RTT substantially. As the HTTP GET request starts with a SYN request, it suffers from the same demotion delays.

We also evaluated the user-experienced network latency overall for these requests. We measured the latency from the first SYN packet until the last packet related to the HTTP request was received, which excludes Android UI and other system latencies. In Figure 14, we show the distribution of user-experienced latencies when browsing in various RRC states as well as during state transitions. Starting in a low-power state has a substantial performance impact, adding 0.5–3s to the user-perceived latency. C2's throughput is significantly worse than C1's at our location, so the overall network performance differs from that in our global study.

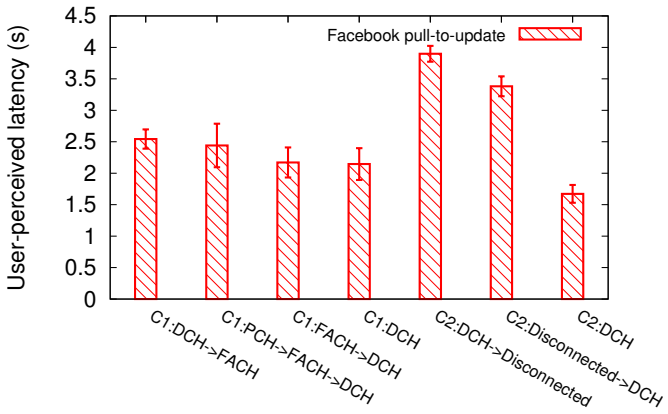

Figure 15: Impact of additional RRC state transition delays on Facebook's pull-to-update action.

Unlike C2, C1 has an intermediate FACH state and thus two demotions. As a result, there is a higher chance that users of C1 transmit data during a state demotion. In our controlled experiments, we found that for C2, 2.4% of HTTP GET requests experienced demotion delays, and for C1, 4.25% of requests were affected.

We also examined a 9 month user study trace of browsing traffic we collected, and we simulated the RRC state transitions for that carrier, counting how many packets would have fallen into the demotion period. For C1, which has relatively short timers and long transition periods, 3.2% of requests would have been affected. For C2, only 0.2% would have been affected. Since this carrier's timers are much longer, the chance of a packet being sent during the demotion period is much lower. However, this comes at the cost of higher energy consumption.

We briefly examined the impact of these RRC states on a major video streaming service as well, and found that packets are sent continuously, causing the device to remain in DCH and thus avoiding all transition problems (at the cost of high energy consumption). It has been proposed that streaming apps could save a substantial amount of energy by batching data [36, 17]. These sorts of optimizations would mean that streaming apps would then need to account for RRC state dynamics. In particular, interruptions in streaming video are highly undesirable, and so avoiding unexpected network delays is critical.

7.3 Case Study: Facebook Application

Through controlled experiments, we also examined the Facebook application, one of the most popular social networking apps [32]. A major Facebook feature is its news feed [30]. We examined the time to fetch new news feed content over the network in response to the user pulling down on the list (the "pull-to-update" action). As described in §4.3, to systematically and repeatedly measure the latency associated with app operations, we created a controller application that repeatedly performs this action, logging the timestamp when the action is initiated and when the news feed finishes loading.

We performed the experiment on two Android 4.2.2 Samsung Galaxy S3 devices. We created two Facebook accounts, A and B, which are friends with one another. One device with account A repeatedly uploaded two photos to generate news feed data. The other account performed a pull-to-update operation, varying the time intervals between each action. As shown in Figure 15, the DCH⇒FACH demotion process increases the user perceived latency by 398 ms for C1, and the DCH⇒Disconnected process introduces an additional 2.225s delay for C2. The results are consistent with the web browsing experiments. As with those experiments, the RRC state transition delay is worse for C2 due to exceptionally poor network performance where we performed the experiment.

Summary: We show that the problems we observed are not just limited to affecting individual packet latencies. RRC state transitions, especially RRC state demotions, can greatly impact the performance of network and application layer protocols, as well as the performance of web browsing and network applications directly, adding delays of up to five seconds in the worst case. The degree to which transitions impact application performance varies significantly by carrier, but can often add delays of over a second.

8. DISCUSSION

It is well known that there is a tradeoff between performance and battery consumption when setting RRC state timers. Our findings suggest that these tradeoff decisions have been made with an incomplete understanding of the impact on performance, as RRC state performance is more complex than previously believed. These findings have implications both for carriers and for developers.

Recommendations for carriers: In general, we have shown that the impact of RRC states on user-perceived performance is complex, unpredictable, and highly dependent on implementation details not defined in an official specification. Given these challenges, the tools we have introduced are valuable in helping carriers ensure their performance requirements are met. We next describe concretely how carriers can benefit from the measurement tools we discuss in this paper.

Given some power and performance goals a carrier wants to achieve with its RRC state configuration, the carrier can then use our RRC performance measurement tool to ensure that their state transition performance is as expected, and if not, make adjustments to their timers. For many carriers, our dataset of RRC state performance measurements already contains this information. If the carrier is not in our dataset it is easy to use our open-source application to collect this data. In addition to determining the performance impact of RRC states in practice, there are two major pitfalls they should watch out for in their RRC state implementation.

The first is the presence of RRC demotion delays or packet losses. These can be detected by our application, and a tool such as QxDM can confirm the existence of this problem, as well as pinpoint some possible causes. Then, the carrier can determine if this is a configuration or implementation bug that can be addressed. In general, we found that longer timers make these demotion problems less likely, as then traffic is less likely to be sent during a state demotion. For carriers with long RRC timers, the probability of traffic being impacted by bad RRC state demotions is low. However, carriers who value preserving battery life more should pay close attention to their RRC state implementations. Furthermore, at least one carrier appears to have simplified its RRC state demotion process, reducing demotion delays. LTE also has lower demotion delays due to a simpler demotion process.

The second potential problem is that often users do not in practice gain any performance benefit from FACH. For many carriers, demotions to FACH are particularly impacted by long demotion delays. FACH is supposed to provide a level of performance between DCH and IDLE, but in many cases, high state demotion delays mean that sending packets in FACH is worse than sending packets in IDLE for some traffic patterns. While we do not suggest eliminating FACH in every case — some carriers

which use FACH do achieve performance benefits — carriers which use FACH should investigate, through measurements on real user devices, whether FACH is beneficial in practice. Given the frequently marginal benefits gained by FACH, it seems that LTE's approach of eliminating the intermediate power state makes sense.

Finally, the carrier is likely also interested in the impact of the RRC state configuration on real, major applications. Our app can crowdsource small-scale HTTP and DNS performance measurements, that can then be confirmed using in-lab testing, as in § 7. Using these methods, carriers can be confident that their RRC state configuration has the performance properties they expect.

Recommendations for developers: Our findings also have implications on how application developers should design applications. We have found that the impact of certain inter-packet interval patterns can have an even worse performance impact than previously believed. This underscores the importance of batching data to ensure both good performance and power consumption.

Furthermore, our techniques for measuring the impact of RRC states on application QoE, both in-lab and through application deployments, would be valuable to developers. It is known that long inter-packet intervals lead to performance and power issues, but recent work [26] has argued that excessive batching and prefetching has a significant negative effect on data usage. An understanding of when application performance is and is not significantly impacted by RRC state transitions, which is carrier dependent, would assist developers in deciding whether or not batching data is necessary to achieve good performance. For instance, if the app developer is not overly concerned about battery life, but wants to conserve data and ensure good performance, it might not make sense to batch downloads too aggressively on carriers with long RRC state timers. Conversely, on carriers with high state transition delays, batching downloads might be more critical. It would even be possible for developers to use our RRC performance measurement method in their own apps, to determine if specific network requests of concern are substantially impacted by RRC state, and thus whether batching network traffic would reduce RRC state transition delays.

Furthermore, recent work has examined allowing applications to account for RRC state in order to reduce latency and save power during network transmissions [17, 20, 25, 15]. By using the dataset of RRC state performance gathered through our tool, these tools can update their model of RRC state implementations as they change over time and account for differences between carriers worldwide. Further development of libraries and frameworks to allow app developers to easily account for complex, varying network performance and power consumption behavior would be highly valuable.

Impact of battery consumption: In this paper, we have focused on measuring the impact of RRC state on application performance. However, RRC states also have a significant impact on battery consumption as well. Accurately measuring power consumption at a fine-grained level on unmodified user devices is inherently challenging. Latency can be measured directly, but accurately inferring power consumption on arbitrary user devices with no specialized equipment remains an open area of research. However, once a model of battery consumption in different power states has been developed for a specific model of device, battery consumption from cellular network usage could be estimated. Assuming that power in each state is constant or linear with time for a given signal strength, as in previous work [17] the power consumption of an application could be calculated from the inferred RRC state timers for each carrier, making network power consumption estimates more accurate.

9. CONCLUSION

In this paper, we examined the impact of RRC states on user-perceived performance in depth. We uncovered several previously unknown implementation artifacts that can lead to delays of up to several seconds, and have demonstrated the impact of RRC states on latency and packet loss for various network protocols and applications. We have investigated the root causes of these performance problems by examining RLC-layer messages in order to determine what configuration events and messages cause the delays observed. In doing so, we confirm the presence of these unexpected delays, and determine that, while they are partially unavoidable, they are exacerbated by complex, multi-stage state transitions and unexpected negative interactions with other control-plane configuration events. Furthermore, we discovered that some carriers have configured their RRC state machines to avoid many of these pitfalls, suggesting these problems are fixable.

In addition to identifying specific, previously unknown performance problems in networks around the world, this paper also motivates the need for continuous, long-term and global monitoring of cellular network configurations and the impact on performance, with a emphasis on uncovering unexpected and non-ideal behavior. As applications increasingly account for underlying cellular network implementation details to avoid excessive power consumption, data usage or latency, properly understanding how the underlying cellular network affects application performance in practice is crucial.

10. ACKNOWLEDGEMENTS

We would like to thank our anonymous reviewers and shepherd for their valuable comments. We would also like to thank Kranthi Sontineni, Randy Meyerson, Warren McNeel, and the Product Experience and QoE Lab team at T-Mobile for their assistance. This research was supported in part by the National Science Foundation under grants CNS-1059372, CNS-1039657, CNS-1345226, and CNS-0964545, as well as by an NSERC Canada PGS M scholarship.

11. REFERENCES

[1] 4GTest. `http://mobiperf.com/4g.html`.

[2] Android Activity Testing. `http://developer.android.com/tools/testing/activity_testing.html`.

[3] Android DDMS. `http://developer.android.com/tools/debugging/ddms.html`.

[4] MobiPerf. `http://mobiperf.com`.

[5] NU JamLogger: A Study of User Activity and System Performance on Mobile Architectures. `http://www.ece.northwestern.edu/microarchitecture/jamlogger/`, 2009.

[6] QxDM Professional Proven Diagnostic Tool for Evaluating Handset and Network Performance. `http://www.qualcomm.com/media/documents/files/qxdm-professional-qualcomm-extensible-diagnostic-monitor.pdf`, 2012.

[7] 3GPP TS 35.331: Radio Resource Control (RRC) - UMTS, 2013.

[8] 3GPP TS 36.331: Radio Resource Control (RRC) - LTE, 2013.

[9] A. Gember, A. Akella, J. Pang, A. Varshavsky, and R. Caceres. Obtaining Representative Measurements of Cellular Network Performance. In *Proc. ACM IMC*, 2012.

[10] A. Aucinas, N. Vallina-Rodriguez, Y. Grunenberger, V. Erramili, K. Papagiannaki, J. Crowcroft, and D. Whetheral. Staying Online While Mobile: The Hidden Costs. In *Proc. ACM CoNEXT*, 2013.

[11] C. C. Tossell, P. Kortum, A. Rahmati, C. Shepard, and L. Zhong. Characterizing web use on smartphones. In *ACM CHI*, 2012.

[12] C. Shepard, A. Rahmati, C. Tossell, L. Zhong, and P. Kortum. LiveLab: Measuring Wireless Networks and Smartphone Users in the Field. In *Proc. Hotmetrics*, 2010.

[13] G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. Lakshman, A. Pilchin, S. Sivasubramanian, P. Vosshall, and W. Vogels. Dynamo: Amazon's highly available key-value store. In *Proc. ACM SOSP*, 2007.

[14] E. Halepovic, J. Pang, and O. Spatscheck. Can you GET Me Now? Estimating the Time-to-First-Byte of HTTP Transactions with Passive Measurements. In *Proc. ACM IMC*, 2012.

[15] K. R. Evensen, D. Baltrünas, S. Ferlin-Oliveira, and A. Kvalbein. Preempting State Promotions to Improve Application Performance in Mobile broadband Networks. In *Proc. ACM MobiArch*, 2013.

[16] F. Qian, Z. Wang, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck. Characterizing Radio Resource Allocation for 3G Networks. In *Proc. ACM IMC*, 2010.

[17] F. Qian, Z. Wang, A. Gerber,Z. M. Mao, S. Sen, and O. Spatscheck. Profiling Resource Usage for Mobile Applications: A Cross-layer Approach. In *Proc. ACM MobiSys*, 2011.

[18] E. Griffin. Fast dormancy - save your battery from 3g drainage. `https://www.youtube.com/watch?v=O8L50sCY7CI`, 2012.

[19] GSMA. Fast dormancy best practices. `http://www.gsma.com/newsroom/wp-content/uploads/2013/08/TS18v1-0.pdf`, 2011.

[20] Y. Z. Hao Liu, Yaoxue Zhang. TailTheft: leveraging the wasted time for saving energy in cellular communications. In *Proc. ACM MobiArch*, 2011.

[21] X. He, P. P. C. Lee, L. Pan, C. He, and J. C. S. Lui. A panoramic view of 3g data/control-plane traffic: Mobile device perspective. In *IFIP*, 2012.

[22] J. Huang, F. Qian, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck. A Close Examination of Perfomance and Power Characteristics of 4G LTE Networks. In *Proc. ACM MobiSys*, 2012.

[23] J. Wigard, T. Kolding, L. Dalsgaard, and C. Coletti. On the User Performance of LTE UE Power Savings Schemes with Discontinuous Reception in LTE. In *Proc. International Conference on Communications*, 2009.

[24] L. Zhou, H. Xu, H. Tian, Y. Gao, L. Du, and L. Chen. Performance Analysis of Power Saving Mechanism with Adjustable DRX Cycles in 3GPP LTE. In *IEEE VTC*, 2008.

[25] H. A. Lagar-Cavilla, K. Joshi, A. Varshavsky, J. Bickford, and D. Parra. Traffic backfilling: subsidizing lunch for delay-tolerant applications in UMTS networks. In *Proc. ACM MobiHeld*, 2011.

[26] Lenin Ravindranath, Sharad Agarwal, Jitendra Padhye, Christopher Riederer. Procrastinator: pacing mobile apps' usage of the network. In *Proc. ACM MobiSys*, 2014.

[27] S. Mohan, R. Kapoor, and B. Mohanty. Latency in hspa data networks. Technical report, Qualcomm, 2011.

[28] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani. Energy Consumption in Mobile Phones: A Measurement Study and Implications for Network Applications. In *Proc. ACM IMC*, 2009.

[29] Nokia Siemens Networks. Understanding smartphone behavior in the network, 2011.

[30] J. Osofsky. More ways to drive traffic to news and publishing sites. `https://www.facebook.com/notes/facebook-media/more-ways-to-drive-traffic-to-news-and-publishing-sites/585971984771628`, 2013.

[31] P. K. Athivarapu, R. Bhagwan, S. Guha, V. Navda, R. Ramjee, D. Arora, V. N. Padmanabhan, and G. Varghese. RadioJockey: Mining Program Execution to Optimize Cellular Radio Usage. In *Proc. ACM MobiCom*, 2012.

[32] E. Protalinski. Facebook passes 1.23 billion monthly active users, 945 million mobile users, and 757 million daily users. `http://thenextweb.com/facebook/2014/01/29/facebook-passes-1-23-billion-monthly-active-users-945-million-mobile-users-757-million-daily-users`, 2014.

[33] L. Qian, E. W. Chan, P. P. Lee, and C. He. Characterization of 3G Control-Plane Signaling Overhead from a Data-Plane Perspective. In *MSWiM*, 2012.

[34] S. Deng, and H. Balakrishnan. Traffic-Aware Techniques to Reduce 3G/LTE Wireless Energy Consumption. In *Proc. ACM CoNEXT*, 2012.

[35] S. Souders. Making a mobile connection. `http://www.stevesouders.com/blog/2011/09/21/making-a-mobile-connection/`, 2011.

[36] A. Schulman, V. Navda, R. Ramjee, N. Spring, P. Deshpande, C. Grunewald, K. Jain, and V. N. Padmanabhan. Bartendr: A practical approach to energy-aware cellular data scheduling. In *Proc. ACM MobiCom*, 2013.

[37] N. Vallina-Rodriguez, A. Auçinas, M. Almeida, Y. Grunenberger, K. Papagiannaki, and J. Crowcroft. Rilanalyzer: A comprehensive 3g monitor on your phone. In *Proc. ACM IMC*, 2013.