Obtaining In-Context Measurements of Cellular Network Performance

Aaron Gember, Aditya Akella
University of Wisconsin – Madison

Jeffrey Pang, Alexander Varshavsky,
Ramon Caceres
AT&T Labs – Research

ABSTRACT

Network service providers, and other parties, require an accurate understanding of the performance cellular networks deliver to users. In particular, they often seek a measure of the *network performance users experience solely when they are interacting with their device*—a measure we call *in-context*. Acquiring such measures is challenging due to the many factors, including time and physical context, that influence cellular network performance. This paper makes two contributions. First, we conduct a large scale measurement study, based on data collected from a large cellular provider and from hundreds of controlled experiments, to shed light on the issues underlying in-context measurements. Our novel observations show that measurements must be conducted on devices which (*i*) recently used the network as a result of user interaction with the device, (*ii*) remain in the same macro-environment (e.g., indoors and stationary), and in some cases the same micro-environment (e.g., in the user's hand), during the period between normal usage and a subsequent measurement, and (*iii*) are currently sending/ receiving little or no user-generated traffic. Second, we design and deploy a prototype active measurement service for Android phones based on these key insights. Our analysis of 1650 measurements gathered from 12 volunteer devices shows that the system is able to obtain average throughput measurements that accurately quantify the performance experienced during times of active device and network usage.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: Measurement techniques; C.2.3 [**Computer-Communication Networks**]: Network Operations—*Network monitoring*

General Terms

Measurement, Performance

IMC'12, November 14–16, 2012, Boston, Massachusetts, USA.
Copyright 2012 ACM 978-1-4503-XXXX-X/12/11 ...$15.00.

1. INTRODUCTION

Cellular data networks have become a main source of Internet access for hundreds of millions of users. Consequently, many parties require an accurate understanding of the performance that cellular networks deliver to users, e.g., the range of throughput, latency, and loss that users can experience in practice. Content providers need this to optimize application decisions; regulatory bodies need this to validate wireless broadband speeds; and network service providers need this for effective network management and efficient debugging.

Although many passive analysis techniques [17, 20] and mobile device tools for crowdsourcing active measurements [1, 2, 5, 6, 19, 33] have been developed, they mostly quantify the performance the cellular network can offer at random times, irrespective of whether users interact with their devices at these times and actually experience the consequences of the measured performance. For example, the speed of a background synchronization task conducted at night may not be relevant to most users, in contrast to the download speed of a movie a user is watching on-demand. We argue that in many important use-cases there is a need to obtain a measure of the *network performance users experience solely when they are interacting with their device*. Moreover, some parties may want to narrow their view of performance further, to focus on a specific scope of interest (e.g., morning commuters driving in Los Angeles). We refer to such measures of performance as *in-context measurements*, where *context* may refer to, e.g., whether or not a user is actively using their device, a device's position relative to the user (in pocket, in hand, etc.), or any of several other factors that may influence the cellular network performance a user experiences. §2 provides several example use-cases and discusses limitations of prior approaches in more detail.

Despite the need for in-context performance measures (as defined above), it has yet to be shown how one should go about capturing them, or even whether existing approaches are good enough. To this end, our paper makes two contributions: First, we use anonymized cellular network data from 20,000 subscribers of a large US-based cellular carrier and empirical results from 100s of controlled end-to-end experiments to examine the extent to which users' interactions with their devices quantitatively impact cellular network performance. Crucially, we find that measurements must be conducted in a manner that considers whether a user is actually using the device, what position a user puts the device in, and what activities a user is running on the device. Second, we leverage our empirical insights to design a crowdsourcing-based active measurement system for deriving in-context performance measures. We evaluate it using a real-world deployment spanning 12 volunteer users.

Empirical Study. (§3) Our empirical analysis addresses three questions that are central to obtaining in-context performance measurements: (i) How does performance differ when measured at the times users actually use their device as opposed to when devices are idle? (ii) What are the reasons for performance differences? In particular, to what extent do various contextual factors that manifest during active use, e.g., device position, motion, etc., impact measured performance? (iii) How should measurements be conducted to avoid conflating factors? In particular, what types of measurements and user traffic can overlap?

Active vs. Idle: Using network data from 20,000 subscribers, we find that, on average, latency is 16ms higher and loss rate is 6%-age points worse (~17% vs ~12%) near the times users actively use their device. This finding implies that a performance measurement scheme must take into account whether a user is actively using their device, otherwise measurements may overestimate network performance.

Impact of context: We conduct 100s of controlled end-to-end experiments in varying environments to identify the key contextual factors that affect measured performance. We observe that small changes in a device's position–e.g., moving a device from hand to pocket–can cause up to a 79% difference in measured throughput and, to a lesser extent, a difference in measured latency. We also confirm that changes in location by a few 100m and changes in whether a device is stationary or moving may cause >1Mbps difference in measured throughput and ~60ms difference in measured latency.

Measurement interference: Using our controlled end-to-end experiments, we quantify the impact of simultaneous device usage and measurement gathering—an important consideration for maximizing measurement opportunities. We observe that, despite the limited bandwidth and high latency of cellular links, web browsing or low-rate streaming can occur concurrently with measurements, while still obtaining accurate measurement results and suitable user experience.

Measurement System. (§4) A system for in-context measurement must decide *when* and on *which* devices to conduct measurements such that the results convey the cellular network performance users likely experience when they are interacting with their device. Our empirical study shows that measurements must be conducted only on devices which are active (for low-bandwidth measurements) or were recently-active (for measurements that may interfere with user actions). In the latter case, we must ensure that the device's context has not changed compared to its last active use.

To address these challenges, we design a measurement service for Android phones that leverages *crowdsourcing of active measurements*. We describe how to address key challenges, e.g., inferring user activity and monitoring the physical environment. We deployed the service across 12 volunteer cellular subscribers[1] over a three month period, gathering over 1650 measurements. Our analysis of these measurements shows that the system is able to obtain measures of mean throughput equivalent to the performance experienced during times of active device usage.

With a growing focus on the performance experienced by cellular subscribers from a variety of stakeholders, there is an urgent need for systems that provide the needed performance measures. Our work fills this gap both by showing what factors impact measurements to what degree, and by designing and evaluating an accurate measurement system.

2. TOWARD CROWDSOURCED ACTIVE MEASUREMENTS

In this section, we present several use cases for cellular network performance measurements taken from the perspective of mobile devices, and then discuss the requirements measurements must meet to serve these use cases. We argue that existing approaches for measuring end-to-end performance meet the needs of some use cases, but they lack the ability to accurately measure specific metrics in situations where users actually use their device.

2.1 Use Cases and Requirements

Network service providers, content providers, regulatory agencies, and researchers all have a vested interest in measuring cellular network performance; we refer to these parties as *measurement admins*.

Network management. When customers complain, the failures they experienced may be a result of any combination of the mobile device, the network, user behavior (e.g., mobility), etc. Understanding the performance experienced by a certain device in similar contexts can help network providers eliminate confounding factors and narrow down the root cause.

A variety of issues may lead to network alarms being raised, e.g., RTT estimates or radio link control failures crossing key thresholds. It is often unclear if the events impact users, because they manifest as soft failures rather than loss of connectivity. Understanding the performance impact across a range of users is important for prioritizing alarm analysis.

Network providers continuously upgrade their networks but have difficulty understanding the impact on performance because of complex interactions between protocol layers [12, 29]. They benefit from conducting before-and-after user experience measurements in the areas where changes occurred.

Evaluating network providers. Some third parties collect measurements to compare different network providers (e.g., [5, 6]). In addition, regulatory bodies require accurate network speed measurements to evaluate the ubiquity of access to broadband capacities [3, 4]. In both these cases, measurements must accurately represent network performance in order to provide a fair and unbiased view for the public.

Application decision making. Time-sensitive applications (e.g., video streaming, multiplayer games, etc.) require accurate estimates of end-to-end performance to tune parameters such as buffer sizes and streaming rates. Since performance characteristics like delay change over time and space [25], estimating performance on demand is essential.

Measurement Requirements. Several common requirements emerge from these use cases.

1. Measurements should convey performance experienced in environments, and on devices, where users actually send/receive traffic, and more specifically, when users are *interacting with their device* (as opposed to performance at random times or when a device is unused).
2. Admins should be able to control the measurement and obtain specific metrics of interest. Only characteristics

[1] The volunteers include both customers and employees of a large US-based cellular carrier.

of the cellular network which impact end-to-end performance should be depicted in measurement results.

3. Admins should be able to schedule measurements in a controlled scope of interest, e.g., devices used in an area where new infrastructure was deployed, or devices used indoors during peak times. Limiting measurements to controlled contexts helps admins identify the root cause of specific performance.

2.2 Need for a New Approach

Admins currently leverage a combination of passive analysis, field testing, and self-initiated reporting to understand cellular network performance. While these tools provide valuable views of performance, they do not entirely meet the requirements outlined above.

Network-based passive analysis. Network providers can instrument their infrastructure to passively estimate performance statistics based on user traffic [17, 20, 32]. However, it is difficult to determine or control what context mobile devices are in when they originate or receive the network traffic being measured, e.g., whether the device is indoors or outdoors. This limits the ability for administrators to focus their analysis on specific usage contexts. In addition, it can be difficult to eliminate the confounding factors of application variability [17]. Lastly, large scale passive analysis is only viable for admins with access to network infrastructure.

Field testing. Due to the manpower required and restrictions on access, it is impractical for field testing to cover the full range of physical contexts in which the network is used, e.g., to measure throughput in all buildings at many different times of day. Even if we crowdsource measurements from user devices for better coverage in time and space [33], we are unlikely to quantify the performance users experience when they are interacting with their device (§3.2, §3.3.1).

Self-initiated reporting. Admins also can rely on users to manually initiate active measurements and report performance anomalies [1, 2, 5, 6, 19]. However, these results will only capture a small subset of situations in which users interact with their device and the network: partially due to the need for manual initiation, and partially due to most users running these tools only when problems occur.

This paper advocates a new approach: *context-aware crowdsourcing of active measurements*. This measurement method enables a user's interactions with their device, along with the physical context of the device, to be considered when conducting measurements. This method also enables controlled metrics to be collected, without conflating factors external to the cellular network in our understanding of performance. In specific cases, passive analysis of application traffic may be an appropriate substitute for active measurements—e.g., it may be possible to use connections to Facebook to measure latency—but these measurements must be carefully filtered to only include passive observations from specific contexts.

To delineate the issues that determine the design of our approach, we first conduct a detailed measurement study of how users' interactions with their devices impact measured performance (§3). Then, we use the observations to design a measurement service prototype (§4).

3. EMPIRICAL STUDY

In this section, we examine the extent to which user's interactions with their device (or lack thereof), and other relevant factors (e.g., device position when not in use), impact in-context cellular network performance measurements. We analyze anonymized flow records and radio signaling data from 20,000 subscribers of a large US-based cellular carrier, and conduct 100s of end-to-end experiments from mobile devices under our control (§3.1). We start by looking at differences in performance between actual usage and idle periods (§3.2). We then investigate the impact of various contextual factors on the observed differences (§3.3). Finally, we examine the impact of overlap between measurement probes and user traffic (§3.4).

Identifier	*Release*	*Android Version*	*HSDPA/HSUPA 3G Speed (Mbps)*	*CPU (GHz)*	*RAM (MB)*
Phone A	May '11	2.3	Down 21, Up 3.6	1.2	512
Phone B	Jul '10	2.2	Down 7.2, Up 5.7	1.0	512
Phone C	Nov '10	2.1	Down 7.2, Up 2	0.8	512
Phone D	Jan '10	2.1	Down 7.2, Up 2	1.0	512

Table 1: Specifications for phones used in experiments

3.1 Datasets

Cellular Network Data. Our *network* dataset consists of anonymized flow records and radio resource control events from approximately 20,000 devices connected to a large US-based cellular provider. The devices are randomly sampled from a major metropolitan area, and only devices of a single device family are sampled (i.e., same OS and manufacturer). All device and user identifiers are anonymized to protect privacy without affecting the usefulness of our analysis.

Flow records for all traffic were collected on December 5–11, 2010 on the logical links between Serving GPRS Support Nodes (SGSNs) in the target metropolitan area and Gateway GPRS Support Nodes (GGSNs) in the UMTS core network. The data contains the following details, at one minute intervals, for each IP flow: initial cell sector, start and end timestamps, TCP header information, TCP handshake times [20], device identifiers, domain names, and application identifiers.[2]

Radio resource control (RRC) messages were collected on December 5–6, 2010 from the Radio Network Controllers (RNCs) in the target metropolitan area. RRC measurement and intra-frequency handover events were collected, with their relevant parameters. We post-process the data to extract: event timestamps, a list of cell sectors in the active set, and received signal code power (RSCP), i.e., signal strength.

Controlled Experiments. Our *context*[3] and *activity* datasets consist of latency and throughput measurements gathered using servers and devices under our control. Latency is measured using ping, with 1 KB (*context*) or 64 byte (*activity*) packets spaced 1 second apart. Downlink throughput is measured using iPerf, with observed bandwidth reported every 2 seconds (*context*) or 10 seconds (*activity*). All measurements are conducted on Android phones (Table 1) connected to the 3G UMTS network of the same carrier as above.

For the *context* dataset, latency and throughput measurements are conducted with the device in a variety of different positions and environments. The device positions in-

[2] Domain names are extracted from HTTP headers or DNS responses. Applications are identified using a combination of port information, HTTP headers, and other heuristics [15].

[3] The context dataset is available for download at: `http://cs.wisc.edu/~agember/go/imc2012dataset`

clude a user's hand, pocket, backpack, and desk/table. The environments vary in terms of geographic location, device movement speed (stationary, walking, city driving, highway driving), and place (indoors, outdoors, vehicle). For each environment, we conduct measurements on a single device model[4], and each measurement is run for either 1 or 5 minutes, depending on the environment. We conduct at least 5 measurements in each device position in each environment (in walking environments we only consider hand and pocket positions and in driving environments we do not consider desk/table), and we change device positions in a round-robin order to avoid bias due to temporal variation.

For the *activity* dataset, latency and throughput measurements are conducted on devices with and without user activities running simultaneously. The user activities include: web browsing, which loads 15 popular websites[5]; bulk downloads, which downloads a single large file 0.5-3MB in size; and streaming video, which models YouTube's streaming mechanism [9], sending a 4 minute video encoded at a specific bitrate. All measurements are conducted on devices in the same position and environment—stationary on a table indoors in a single geographic location. We conducted some measurements in a second location at different times, affirming that results from other environments are similar.

All figures in the paper identify the dataset used.

3.2 Active vs. Idle

The first step towards obtaining in-context cellular network performance measurements is to understand how performance differs between the times users actually use their devices versus the times the devices are unused. Specifically, we consider a device to be *active* when a user is interacting with the device (e.g., browsing the web, using a navigation application, playing music, etc.) and the activity requires sending/receiving data over the cellular network. A device is *idle* when the user is not interacting with the device (e.g., screen is off and no audio is playing), although background traffic (e.g., email sync) may still be present.

We compare the network performance of active devices to the performance of idle devices using the *network* dataset, accounting for the well-known effects of time [25], space [33], and resource allocation [28]. We find significant differences between the performance experienced by active and idle devices, implying that random measurements, conducted irrespective of whether a user is actually using their device, insufficiently capture an in-context view of performance.

3.2.1 Methodology

Estimating performance. To estimate network performance at different times and locations, we leverage the observation that each device in the *network* dataset has a TCP connection on which it sends and receives keep-alive messages approximately once every 30 minutes. This occurs whether the device is active or idle, and all devices connect to the same data center. We define each keep-alive event as a *measurement.*[6]

We estimate two performance metrics for each measurement: downlink loss rate and RTT. We estimate loss as the ratio of the number of bytes retransmitted over the number of bytes received: $(tcp\_datalen - seqno\_range)/seqno\_range$, where $tcp\_datalen$ is the total number of TCP payload bytes observed during the measurement and $seqno\_range$ is the difference between the minimum and maximum TCP sequence numbers observed. We estimate RTT using TCP handshake times when the measurement is preceded by a new TCP connection [20], as the connection is periodically reestablished by the device. There are approximately 2 million measurements in the network dataset.

Identifying active times. The network dataset does not give us direct indications of when a user is interacting with their device, so we infer a device is *active* based on the presence of network traffic that is likely to be triggered by user input. More specifically, we define an *active range* as the time range between the start and end times of flows that indicate user activity; active ranges are calculated on a per-user basis.

To differentiate flows that indicate user activity from background traffic flows (e.g., email polling and keep-alives), we analyze the periodicity of traffic for each <domain name, application> pair. Many applications send periodic traffic frequently while in active use (e.g., Facebook sends keep-alives every 1 minute), but the periodicity is much lower than background traffic (e.g., email polling every 10+ minutes). Based on manual inspection of the traffic patterns of the most used applications, we chose a periodicity threshold of 5 minutes to differentiate between the two categories of flows. If one of the top three periodicities for a <domain name, application> pair (excluding the keep-alive traffic described above) is longer than 5 minutes and appears as a spike in the frequency domain, we label the time ranges for flows belong to this pair as *unknown ranges*; this traffic is likely just background traffic but may indicate user activity. All other flows indicate active ranges.

We define any measurement that occurs within 10 minutes of an active range to be a *near active* measurement, and any measurement that occurs $>$ 30 minutes from all active and unknown ranges to be an *idle* measurement. We ignore measurements that overlap or occur within 17 seconds of an active or unknown range to avoid measuring the effects of self-interference and to ensure near active and idle measurements both start in the same cellular radio state [28]. While our definitions of idle and near active times are inexact, we conjecture that the near active measurements are more likely to overlap actual active times than the idle measurements. Thus, these definitions are sufficient to demonstrate differences in active and idle measurements on average.

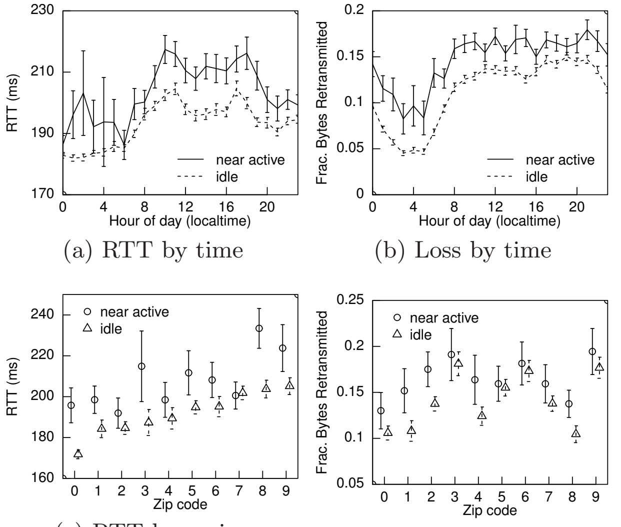
3.2.2 Results

Performance differences. We expect that performance will differ during different times of the day, as the traffic load on the network varies. So the first question we answer is: If we control for time of day, is the performance experienced by idle devices the same as active devices? Figures 1a and 1b show the mean latency and loss during each hour of the day for near active and idle devices; the error bars show 95% confidence intervals. We observe a clear difference in the

[4] We manually validated that multiple phones of the same model deliver similar performance under identical conditions.

[5] Based on Alexa, *http://www.alexa.com/topsites/countries/US.*

[6] While this connection is between the device and one of several different server IP addresses in the data center, the average performance of any subset of these connections should not be biased by the server selection algorithm. We statistically validated the server is chosen uniformly at random and all servers are co-located.



(a) RTT by time (b) Loss by time

(c) RTT by region (d) Loss by region

Figure 1: Distribution of latency and loss for active and idle devices in a large metropolitan area [$Network$]

near active and idle performance for most hours of the day. Measurements on idle devices have up to 16ms lower RTT; loss rates are lower by up to 6%-age points (or, 40% lower, relatively speaking) compared to near active measurements.

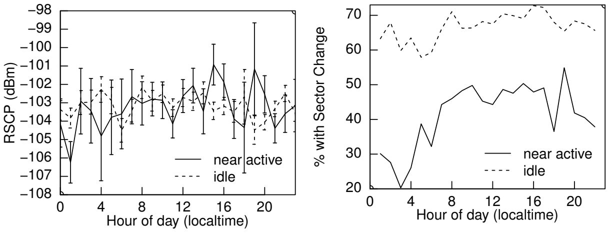
It has been shown that performance also differs between coarse-grain locations [33]. If we control for coarse-grain geographic location, is the performance experienced by idle devices the same as active devices? Figures 1c and 1d show the mean latency and loss for the 10 zipcodes with the most measurements for near active and idle devices. Again we see that measurements on idle devices have up to a 30ms lower RTT, and loss rates lower by up to 4%-age points (or, 26% on a relative scale) compared to near active measurements.

Thus, we conclude that the average performance of idle devices is *not* representative of the average performance of active devices, either at the same point in time, or within the same coarse geographic area. Including measurements from idle devices would overestimate performance, on average. Avoiding even such small overestimation is important, as minor differences in latency and loss can become magnified over the lifetime of a TCP connection. Moreover, studies have shown that even small increases in latency—as little as 100ms—can impact user retention [21].

Cause of differences. Two common causes of cellular performance variation—time of day and coarse geographic location—have already been accounted for in our examination of the performance differences between *active* and *idle* devices. We now consider three other possible causes.

First, we consider the duration of time a device has been idle. We calculate the Pearson's correlation coefficient between the idle duration and the RTT or loss. For both metrics, there is no correlation (0.0057 and -0.0045, respectively), indicating the duration of time since a device was last active does not impact its network performance.

Next, we consider signal strength. Figure 2a shows the mean signal strength across all near active and idle devices during each hour of the day; the error bars show 95% confidence intervals. We observe no significant difference in average signal strength for most hours of the day, indicating variations in signal strength are unlikely to cause the differences in latency and loss between active and idle devices. Our measurement system evaluation (§4.1) shows the effect of signal strength on measured throughput is also minimal.



(a) Average signal strength (b) Cell sector changes

Figure 2: Possible causes of performance differences between active and idle devices [$Network$]

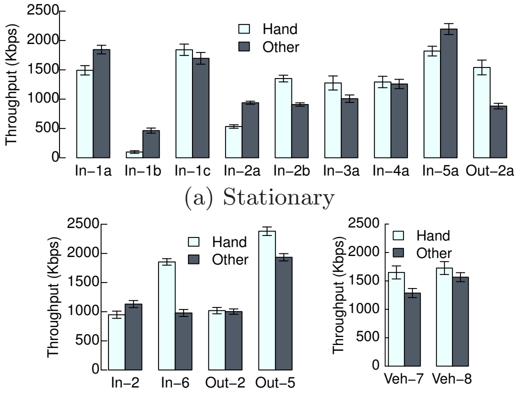
Finally, we consider differences between cell sectors as a possible cause of the performance differences. This consideration is prompted by Manweiler et al.'s observation that latency differs between cell sectors managed by different RNCs [25]. Figure 2b shows, for each hour of the day, the fraction of near active and idle devices which have changed cell sector since they were last active. We observe that a larger faction of idle devices are using different cell sectors, implying that differences between cell sectors—frequency, back-haul capacity, load, etc. [8]—are a possible cause of the performance differences between active and idle devices.

Summary. Our analysis indicates that the average network performance of active devices is worse than that of idle devices—the latency experienced by active devices is 16ms higher and the loss rate is 26% higher (∼17% loss vs ∼12% loss)—requiring measurements to be cognizant of whether devices are active to avoid overestimating network performance. Such differences in network performance cannot be attributed to idle duration or signal strength, but may be caused by differences between cell sectors. In the next section, we explore several other contextual factors, e.g., device position, that also likely cause the performance difference.

3.3 Impact of Physical Context

There are many factors that can contribute to differences in the end-to-end performance the network is capable of delivering to a device, including signal strength, signal fading and interference, handover rate, cell sector traffic load, and back-haul capacity. Several of these factors change over time as a result of what a user physically does with their mobile device—e.g., drives in a car with their device. In the previous section, we alluded to changes in these factors as possible explanations for the difference in achievable performance between active and idle devices. However, precisely accounting for changes in these low-level factors is extremely difficult, if not impossible.

We focus instead on the extent to which various aspects of a device's physical context—which influences these low-level factors—impacts measured performance. For example, changing a device's position from your hand to your pocket can change the signal characteristics, as can moving from indoors to outdoors. We use the *context* dataset (described in §3.1) to quantify the differences in performance between device positions—e.g., hand versus pocket—

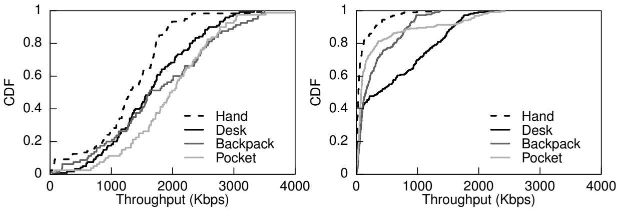


(a) Stationary

(b) Walking

(c) Driving

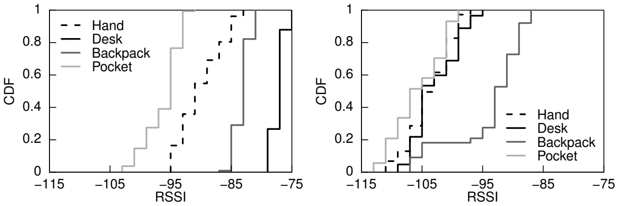
Figure 3: Difference in mean throughput (with 95% confidence intervals) between hand and all other positions for various environments. Environments are named by place (*In*doors, *Out*doors, *Veh*icle), geographic location (1-8), and stationary spot (a-c). [*Context*]



(a) In-1a

(b) In-1b

Figure 4: Throughput in two offices in a building [*Context*]



(a) In-1a

(b) In-1b

Figure 5: RSSI during throughput measurements [*Context*]

and validate prior observations on performance differences associated with some changes in environment—e.g., moving from one room to another. We find that a device's position, location, and movement significantly and consistently influence measured performance.

3.3.1 Impact of Device Position

Users typically hold their mobile device in their hand when they are actively using the device, whereas they may place it on their desk or in their pocket or backpack when not in use. The device's signal characteristics will vary with each position, potentially resulting in different network performance. We quantify the differences in performance between several positions—a user's hand, pocket, backpack, and desk—to determine what effect, if any, minor position variation has on measurements. We observe different trends with different types of measurements, so we discuss throughput and latency measurements separately.

Throughput. In most environments, we observe a significant difference in throughput between device positions. Figure 3 shows the mean throughput measured with a device in a user's hand versus in other positions across various environments; the error bars show 95% confidence intervals. Mean throughput differs by up to 875Kbps between positions in some environments, but it may differ by less than 50Kbps.

Differences in measured throughput between device positions occur across all types of environments. Stationary devices may exhibit 35-660Kbps differences in throughput between positions, and moving devices may exhibit 20-875Kbps differences between positions. Similar variation in throughput differences exists for indoor and outdoor environments. Furthermore, there is no consistent trend as to whether performance in hand is always better or worse: mean throughput can be up to 47% better when the device is in the user's hand, or it can be up to 79% worse. Hence, throughput measurements must be cognizant of device position.

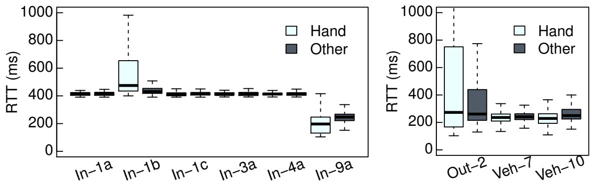
We look in detail at measured throughput across positions for a few of the environments to determine if we can attribute differences between positions to specific low-level factors. We consider two indoor stationary environments (*In-1a* and *In-1b*), which are different offices in the same building. However, similar trends in measurements and low-level factors hold for a user sitting at two different desks in a single office in a different building (indoor stationary environments *In-2a* and *In-2b*) and for users walking with their devices (environments *In-2*, *In-6*, *Out-2*, and *Out-5*).

Figure 4 shows CDFs of throughput across several measurements conducted in each device position in the two offices. In both offices, there are significant differences in the throughput distributions for each device position: the median throughput is 50% higher in pocket than in hand in In-1a and 917% higher on desk than in hand in In-1b. Furthermore, the ordering of throughput curves by device position is not consistent across the two offices.

We first consider signal strength as a possible explanation for the throughput differences between positions. Figure 5 shows the distribution of signal strength (RSSI) during our measurements in the two offices for each device position. There is a noticeable difference in median RSSI between positions: up to 18dB for In-1a and 12dB for In-1b. However, the ordering of the throughput curves does not match the ordering of the RSSI curves for either office. This suggests signal strength is not a primary factor.

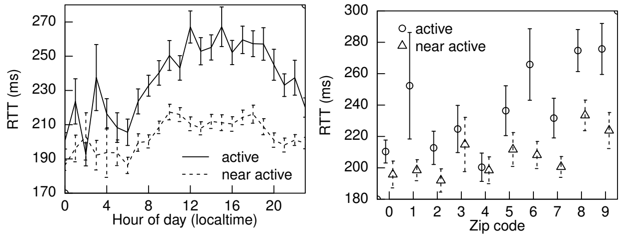
We also consider cell sector, and its associated low-level factors, as a possible cause of throughput differences. However, we observe that all measurements conducted in In-1a occur over a single cell sector, and most measurements conducted in In-1b occur over that same cell sector, along with one other cell sector. This implies that cell sector is not a primary cause of the throughput differences between positions.

Our hypothesis is that small scale fading is the primary low-level factor causing differences in measured throughput across positions. Due to constructive and destructive interference of a signal's reflections off different objects in the environment, the signal power can be time and space varying even over small regions of space. This means that a device being in a user's hand versus their pocket, when the user is stationary, can result in significant differences in network performance. We would expect that the small scale fading would average out in moving environments, but since



(a) Stationary (b) Moving

Figure 6: Difference in latency between hand and all other positions for various environments; boxes show 25th, 50th, and 75th %-tile, error bars show min and max [*Context*]



(a) RTT by time (b) RTT by region

Figure 7: RTTs with/without competing traffic [*Network*]

some objects do not move randomly—e.g., your pocket and a device within it moves in unison when you are walking—some regions of space may be consistently stuck in a null (destructive interference). Hence, we would still observe the effects of small scaling fading even when walking or driving.

Latency. In contrast to throughput measurements, latency measurements are less affected by device position. Figure 6 shows the median latency measured with a device in a user's hand versus in other positions across some of the same environments. We find that median latency tends to be equivalent across most environments, differing by more than 15ms in only one third of the environments. Furthermore, the worst-case (e.g., 75th percentile) measured latencies differ substantially between positions in only two environments.

Like throughput measurements, there is no consistent trend across environments as to when latency measurements will differ between device positions. However, our observation that median latency tends to be similar between device positions in the same environment agrees with previous findings [25] that latency is similar amongst mobile devices connected to the same cell sector. Overall, our findings suggest that median latency can be accurately measured irrespective of device position, but full distributions of latency require considering device positions.

3.3.2 Impact of Environment

In addition to changing the position of their mobile device, users may also change locations or transition between stationary and moving. These changes in environment have previously been shown to cause differences in end-to-end network performance [25, 33, 37]. We briefly discuss how our results compare with these findings.

Location. Measurements conducted using WiScape [33] showed that throughput differences of a few 100Kbps exist between locations as small as 250m apart. Similarly, Figure 3a shows a difference in measured throughput between three offices in the same building (environments *In-1a*, *In-1b*, and *In-1c*): the mean throughput of a device held in each of the three offices is 1491Kbps, 98Kbps, and 1842Kbps, respectively. All measurements are conducted in these offices within a timespan of about 2 hours and most measurements occur over the same cell sector, so these factors are unlikely to be major contributors to the measurement differences.

There also exists a difference, albeit less pronounced, in measured latency between locations (Figure 6a): median latency measured on a device held in each of the three offices is 416ms, 475ms, and 412ms, respectively. This discrepancy occurs despite most measurements occurring over the same cell sector, which others have shown to be the primary contributor of latency differences [25]. In summary, changes in location cause differences in measured performance.

Stationary versus moving. Tso et al. [37] showed that downlink throughput can decrease by over 1Mbps between a stationary environment and a public transportation environment (e.g., bus). Figure 3a shows a difference in measured throughput between stationary and moving devices in the same location (environment *2*): held devices moving at walking speeds indoors and outdoors have mean throughput of about 950Kbps, while stationary devices have mean throughput of 1540Kbps outdoors and a range of 533Kbps to 1351Kbps indoors. Thus, measurements conducted at even slow speeds can exhibit differences compared to measurements on stationary devices.

Summary. Our findings indicate that both device position and environment can impact measurement results. For example, an active device in a user's hand will experience different performance than an idle device in a user's pocket due to small scale fading. Additionally, differences in performance between environments can affect the accuracy of measurements if a device is typically active in some environments and idle in others. Therefore, *in-context measurements should not be conducted in positions or environments where devices are not active.*

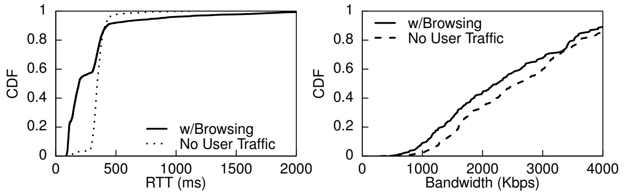
3.4 Measurement Interference

The final factor we consider in conducting in-context performance measurements is the allowable overlap in measurements and user traffic. Many use-cases require specific metrics and results that depict only how characteristics of the cellular network impact end-to-end performance (§2.1). Hence, we want to avoid the concomitant effects of, and on, other traffic sent/received by devices under consideration. Additionally, we want to maximize the opportunities for conducting in-context measurements, since our earlier observations already require limiting measurements to devices which are active and in certain physical contexts.

We use the *network* and *activity* datasets to examine how user traffic—traffic resulting from user interactions with the device—impacts measurement results. We also study how active measurements may impact a user's experience.

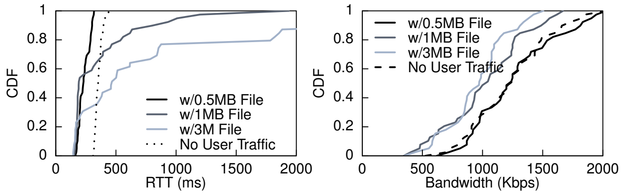
3.4.1 Impact on Measurement Results

Latency measurements. We first examine the effect of self-interference caused by real flows, which encompass a wide range of traffic triggered by user interactions. We use the *network* dataset and the same methodology described in §3.2.1. Figure 7 shows the mean RTT over time and in different regions for measurements that overlap user-initiated flows (active) and nearby measurements that do not overlap user flows (near active); the error bars show 95% confidence



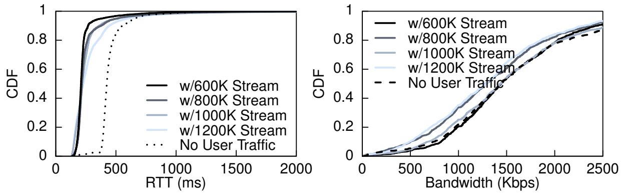
(a) Ping (b) iPerf

Figure 8: Measurements with web browsing [*Activity*]



(a) Ping (b) iPerf

Figure 9: Measurements with bulk downloads [*Activity*]



(a) Ping (b) iPerf

Figure 10: Measurements with video streaming [*Activity*]

intervals. We observe that mean RTT is about 20% (40ms) higher, during peak hours and in many regions, when measurements overlap with user traffic. This suggests that at least some types of concurrent user traffic will bias measurements.

We use the *activity* dataset to narrow the potential causes of this bias. We compare the RTTs reported by ping with and without three common types of user traffic: web browsing, bulk downloads, and video streaming (Figures 8a, 9a and 10a). We make three observations.

First, the lowest RTT measurements without user traffic are ≈150ms higher than the lowest RTTs reported during activities. This is due to effects from the radio resource control (RRC) state machine, which causes devices with low traffic volumes to stay in the *CELL_FACH* state, utilizing a low-bandwidth shared channel [29]. Operators need to adjust RTT measurements to account for this RRC effect.

Second, some user traffic significantly increases the tail of measured RTTs. In the presence of web browsing, 8% of measured RTTs are greater than 450ms, versus only 4% without web browsing (Figure 8a). The tail increase is even more pronounced for bulk downloads ≥ 1MB (Figure 9a).

Finally, the median RTT provides an accurate measure of latency only during short transfers and constant rate transfers. The median RTT with and without short web transfers is about 360ms (Figure 8a), as is the median RTT with and without video streaming (Figure 10a), due to the constant data transfer rate used after the initial video buffering period. In contrast, the median RTT measured during bulk downloads (Figure 9a) is only accurate for a small 0.5MB file, not the larger 1MB and 3MB downloads.

Throughput measurements. We also examine the bias user traffic introduces in throughput measurements. Figures 8b, 9b and 10b show that measured throughput distributions are affected by all three categories of user traffic. As the peak bandwidth used by the user interaction increases, the impact on throughput measurements increases. For example, 1MB and 3MB bulk downloads are able to achieve higher peak bandwidths due to an increase in connection duration and a corresponding opportunity for TCP to probe for more bandwidth. A similar, though less pronounced effect, occurs as video streaming rates increase.

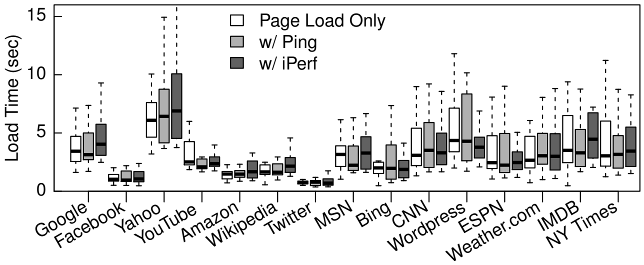
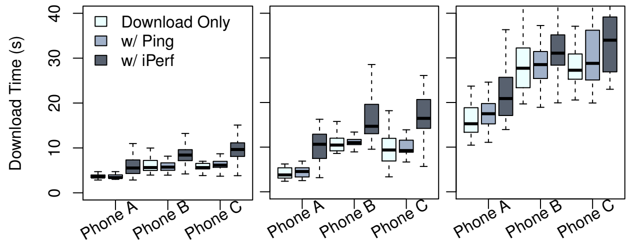


Figure 11: Page load times on *Phone A*; boxes show 25th, 50th, & 75th %-tile, error bars show min and max [*Activity*]



(a) 0.5MB (b) 1MB (c) 3MB

Figure 12: File download times; boxes show 25th, 50th, & 75th %-tile, error bars show min and max [*Activity*]

3.4.2 Impact on User Experience

We also consider the impact of active measurements on user experience, again using the *activity* dataset.

Web browsing. The median page load times—which includes both content retrieval and page rendering time—of 15 popular websites are unaffected by active latency or throughput measurements (Figure 11). We attribute the small variations (< 0.5s) for a few pages to transient network effects.

Bulk downloads. Applications relying on downloading data in bulk are affected by bandwidth-intensive throughput measurements. For files 0.5-3MB in size, throughput measurements cause download times to increase by 0.5s to 7s, with a 9% to 223% increase in variation (Figure 12). This time increase does not occur with web browsing because several small files (a few 100KB in total) are downloaded, keeping TCP congestion windows small and preventing the opportunity to achieve throughput levels that would be affected by co-occurring throughput measurements. Latency measurements have a much less pronounced impact on file download time: at most a 0.8s increase for a 1MB file.

Video streaming. Streaming is affected by both latency and throughput measurements. Table 2 shows the fraction of playback time spent buffering, which is a good indication of user experience [14]. Throughput measurements cause up to a 157% increase in the median buffering ratio, due to the high-bandwidth streaming requires. Latency measurements

	Median			Relative Increase	
Bit Rate	Only	w/Ping	w/iPerf	w/Ping	w/iPerf
600Kbps	0.00	0.00	0.00	0%	0%
800Kbps	0.00	0.01	0.09	11%	12%
1000Kbps	0.04	0.06	0.09	57%	114%
1200Kbps	0.07	0.09	0.19	24%	157%

Table 2: Streaming buffering ratios on *Phone B* [*Activity*]

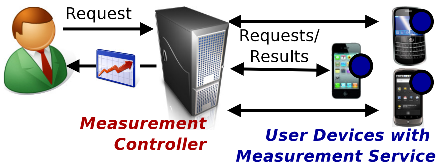


Figure 13: System architecture

cause up to a 57% increase in the median buffering ratio, as ping packets add jitter that impacts the ability to maintain a constant streaming ratio. Thus, any measurement activity could impact user experience for jitter-sensitive applications.

Summary. Our findings indicate the measurements conducted in the presence of low-bandwidth user activities (e.g., web browsing or low-rate streaming) accurately portray median latency and throughput, while high-bandwidth user traffic causes overestimates of worst-case latency and underestimates of throughput. Measurement systems (passive or active) must consider these effects for in-context results to be accurate. Additionally, active measurement systems should avoid high-bandwidth measurements during user activities that heavily use the network, and no active measurements should be conducted during jitter-sensitive activities.

3.5 Summary of Observations

With a goal of obtaining a measure of the *performance users experience solely when they are interacting with their device*, we have analyzed cellular network performance across active and idle devices, various physical contexts, and several combinations of measurements and user traffic. Our findings indicate that measurements must be conducted:

- Only on active devices, otherwise measurements may overestimate network performance.
- On devices which are in the same positions and environments where the devices are actively used, otherwise mean throughput and worst-case latency measurements will be inaccurate.
- At times when only low-bandwidth, non-jitter-sensitive user traffic is present, otherwise measurement results will be skewed and user experience will be degraded.

These findings inform the design of our system for context-aware crowdsourcing of active measurements, discussed next. However, these observations can also be used to filter passive measurements to quantify the performance the network delivers solely when users are interacting with their device.

4. MEASUREMENT SYSTEM DESIGN

We now present the detailed design of a cellular network measurement system capable of crowdsourcing in-context active measurements from user devices. Our system consists of a logically centralized measurement controller and users' mobile devices running a local measurement service (Figure 13).

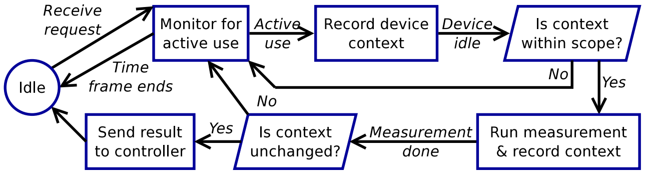


Figure 14: Measurement service decision process

Admins submit requests to the controller, specifying the metric of interest, the time frame and any constraints on which mobile devices to use. For instance, an administrator can request to measure latency on iPhones in Los Angeles during morning rush hour. The controller forwards the request to any viable participating mobile devices. After mobile devices conduct measurements, their results are received by the controller and aggregated for analysis by admins.

Users install our measurement service on their mobile device to conduct active measurements. The service (*i*) receives measurement requests, reports results, and sends coarse context updates to the controller, (*ii*) monitors network usage, screen state, and device sensors to identify opportunities to conduct an in-context measurement, and (*iii*) initiates the appropriate measurement when an opportunity arises. The service currently supports latency (ping), throughput (iPerf or bulk file download), streaming, and web page load time measurements, but it can be easily extended to gather other metrics. The local decision process used to initiate in-context measurements is shown in Figure 14.

Measurements are only conducted on devices which are *active* or were *recently-active*, to avoid overestimating network performance (§3.2). We consider a device to be active when a user is interacting with the device and the activity sends/receives data over the cellular network. Monitoring for active use is triggered by the receipt of a measurement request from the controller; we discuss the details below. When a device becomes idle, or if the device's usage of the network will not negatively impact the measurement (§3.4), we consider conducting a measurement.

A device's context is sensed and recorded while the device is active and while it is conducting a measurement. We check the context of an active device against the admin-specified scope of interest, conducting a measurement only if it matches. After the measurement finishes, we check if the position or environment changed in a way that impacts performance (§3.3), discarding the measurement if necessary; otherwise, the result is sent to the controller. When the specified measurement time frame ends, the service stops monitoring for measurement opportunities and returns to idle. We discuss the details of context monitoring below.

Our system conducts active measurements to avoid the confounding factors of application variability [17] and overlapping traffic (§3.4). However, our design could be easily modified to passively measure existing application traffic instead of conducting active measurements. The measurement decision process (Figure 14) would remain the same.

We now discuss how we address the challenges of active detection and context sensing in an energy-aware manner and describe their implementation on the Android platform.[7]

[7] Although our service is implemented for Android devices, we believe similar solutions could be ported to other mobile platforms.

Detecting Active. The Android platform generates an event when the screen turns on, which presumably means the user is interacting with their device. Subsequently, we can poll traffic counters for the cellular network interface to identify the presence of traffic from a user activity. However, some interactions may occur when the screen is off: listening to network-streamed audio or speaking into the microphone. Thus, we extend our monitoring to detect (using standard Android APIs) when the network is being used and either the screen is on, the microphone is on, or audio is playing.

Context Monitoring. Environment context can be obtained using existing solutions: cell sector identifier is often a sufficient indication of geo-location; mobility can be detected using cellular signals [34], GPS (used in our implementation), and accelerometers [31]; and indoors/outdoors detection can be based on the ability to obtain a GPS signal [30] or ambience fingerprinting [10].

Position changes are much more subtle. Existing accelerometer-based approaches for detecting a device's position on the human body are promising [11, 16], but no one has investigated the position changes we are interested in, e.g., moving a device from hand to backpack. We design a new approach that stems from a key observation: we do not need to identify the precise position of the device, only whether or not it has changed.

We detect changes in device position by monitoring for increases in the standard deviation of device acceleration. If the increase is significant enough, then significant device movement must have occurred—e.g., picking up the device from your desk—versus a slight device shift—e.g., tilting the device in your hand to see the screen better. Based on experimentation, we chose to sample the accelerometer at a rate of 4Hz, combine the acceleration along all three axis into a single acceleration vector, calculate the standard deviation over the last 15 samples, and signal changes in micro-environment when standard deviation exceeds $1m/s^2$.

To keep energy use low, we only sense the environment and position changes from the start of active use to the end of a corresponding measurement. If at any point we detect a change that affects whether a measurement is in-context (§3.3), we stop sensing until the next active use.

4.1 Evaluation

We evaluate our measurement system's accuracy in gathering in-context measurements, along with its position change detection accuracy and energy consumption. We show that measurements conducted under the constraints listed in §3.5 result in throughput metrics that closely match the performance users experience while interacting with their device. Furthermore, the mobile device service is able to detect position changes with a low false negative rate, and the energy burden imposed on devices is usually minimal.

4.1.1 In-context Measurement Accuracy

We deployed our measurement service prototype to the mobile devices of 12 volunteers over a three month period to evaluate how closely measurements gathered by our system match the cellular network performance these users experienced while interacting with their device. We focused on downlink TCP throughput, a common metric of interest which inherently captures other metrics (latency, loss, etc.).

Dataset. We measure downlink TCP throughput using a 20 second bulk file transfer from a server under our control. Throughput is calculated at one second intervals. Measurements were conducted (*i*) when the device's screen was on but no traffic was present, (*ii*) 20s after a device was active and the environment remained unchanged, and (*iii*) at random times regardless of current device usage or context. We refer to the three types of measurements as *ground truth*, *in-context*, and *random*, respectively.

We use measurements conducted under condition (*i*) as our *ground truth* for in-context performance, because times when the screen is on are times the user may engage in activities that use the network. We do not use passive measurements of actual application traffic because of the variability caused by application-specific factors (e.g., server location) [17] and the influence of cross-traffic from multiple applications (§3.4). Measurements conducted under condition (*ii*) are measurements we expect to be *in-context*, as the criteria for conducting these measurements satisfy the requirements outlined in §3.5. Hence, we compare these measurements against the ground-truth to verify our system can gather accurate in-context measurements of network performance.

We obtained 320 pairs of ground truth/in-context measurements and 607 random measurements. A few of these measurements (12 ground truth/in-context pairs, 14 random) were excluded because the measurement failed to start or complete. In 156 instances, the environment changed after a ground truth measurement was conducted; however, we still conducted a measurement in the idle period that followed (referred to as *environ-diff* measurements) to serve as a comparison to in-context measurements. Additionally, we obtained 284 ground truth measurements that did not have a corresponding in-context measurement, due to the lack of a sufficient idle period (<20s) between the device's screen turning off and then turning back on. These are likely caused by short power-saving screen timeouts. We use these *unpaired-active* measurements only for examining how various environment factors affect mean throughput.

Measurement Opportunities. Each of the volunteers contributed a different number of measurements to the dataset. Due to data plan constraints, the volunteers only ran our prototype for part of the three month period. The top of Figure 15 shows how many days each volunteer participated. Furthermore, the number of measurements contributed by a volunteer each day varies based on their device usage habits. The bars in Figure 15 show the average number of ground truth/in-context pairs, ground truth/environ-diff pairs, and random measurements conducted daily on each volunteer's device. The average number of measurement opportunities per day for each volunteer is the sum of the in-context and environ-diff counts, since these measurements are conducted when a user interacts with their device.

We make two important observations from Figure 15. First, the number of times per day volunteers interact with their device varies significantly. Some devices are used over 12 times per day, on average (e.g., volunteer 4), while other are used less than 3 times per day (e.g., volunteer 2).[8] Second, for most volunteers, 20% to 60% of measurements must be discarded due to changes in environment. This suggests

[8] This only includes instances when the device's screen is on for more than 10 seconds with no network traffic present; short device usage periods and background network usage also occur, but these are not adequate measurement opportunities.

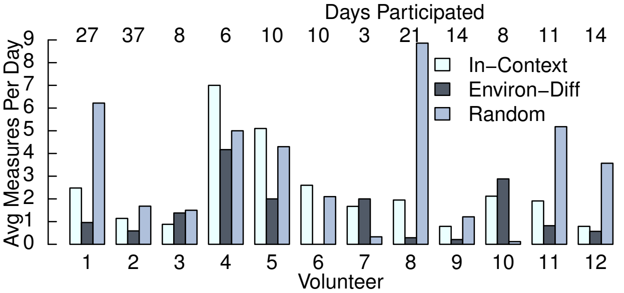


Figure 15: Measurement frequency by volunteer [*Eval*]

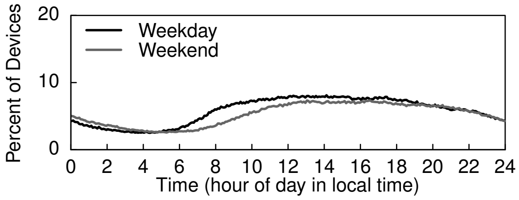


Figure 16: Measurement opportunities at scale [*Network*]

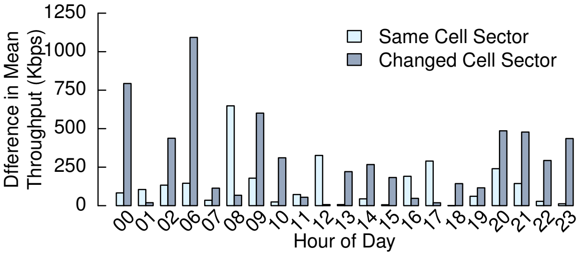


Figure 18: Effect of cell sector changes [*Eval*]

that analyses which do not account for context will include a significant number of results that are not in-context.

Since there are few measurement opportunities per-device per-day, we use the *network* dataset to examine whether there are sufficient opportunities across a large number of users. We consider a measurement opportunity to exist if a device has an active flow (according to our definition in §3.2.1) within a given 5-minute interval. As shown in Figure 16, measurement opportunities exist throughout the day on at least 2.5% of devices, with opportunities on as many as 8% of devices during peak hours on weekdays. Although these percentages are low, they translate to several tens of thousands of devices within a metro area.

Measurement Accuracy. We compare throughput performance across all volunteers for each hour of the day. Due to the sparsity of our collected dataset, we aggregate measurements from the entire three month evaluation period. Figure 17 shows the mean throughput, with 95% confidence intervals, for ground truth, in-context, and random measurements for each hour of the day. The mean throughput reported by ground truth and in-context measurements is approximately equivalent for 18 (75%) of the hours in a day. Two of the hours (4am, 5am) have zero or one measurement pairs, so a fair evaluation cannot be made. For three (11am, 1pm, 8pm) of the four hours when mean throughput differs between ground truth and in-context measurements, the difference can be attributed to a single measurement pair. The cause of these outlier pairs is unknown, but excluding them results in equivalent mean throughput between ground truth and in-context measurements.

Our data also confirms our earlier observation that random measurements provide inaccurate metric values (§3.2). The mean throughput reported by random measurements significantly differs from ground truth throughput for half of the hours. For several hours, the difference is more than 1 Mbps.

In practice, cellular networks offer a range of throughput performance, even within tightly controlled measurement conditions. We compare CDFs of throughput (graphs excluded for brevity) for the three types of measurements for each hour of the day to confirm that our system can accurately quantify this range of performance. We observe that, in addition to differences in mean throughput, the shape and range of the CDF curves for ground truth and in-context measurements are similar and the curves for in-context and random measurements are significantly different.

Environment Impact. We show the effect of changes in environment by comparing the mean throughput during ground truth measurements to both in-context and environ-diff measurements. Specifically, we use environ-diff measurements where the cell sector changes between the ground truth and environ-diff measurement. We aggregate the measurements by hour of day, and we exclude the three outlier ground truth/in-context measurement pairs discussed above. Figure 18 shows the absolute difference in average throughput between ground truth and in-context and between ground truth and environ-diff; the hours 3am-5am are excluded due to a lack of environ-diff measurements. We observe a significant inaccuracy in average throughput measurements for 15 of 21 hours (71%), confirming that changes in cell sector have an impact on measurement accuracy (§3.3). In several hours, the difference is over 0.5Mbps.

We further examine the impact of environment on measurement accuracy by examining which pieces of physical context are the most predictive of mean throughput. We consider nine factors: cell sector (LACID), location area (LAC), speed, indoors/outdoors, network connection type, signal strength, phone model, hour of day, and month. These factors are recorded for all measurements we obtained (including ground truth, in-context, environ-diff, and unpaired-active), along with the measured mean throughput.

We determine which factors are most influential by running the Reduced Error Pruning Tree (REPTree) algorithm in Weka [7]. REPTree builds a decision tree based on information gain, pruning the tree in a way that reduces decision errors. The accuracy of the tree is evaluated using 10-fold cross validation on our dataset. We limit the tree depth to 1, to determine the most predictive factor. Then, we exclude that factor from the dataset and re-run the REPTree algorithm to find the second most predictive factor; the process is repeated until all factors have been excluded.

We find that cell sector (LACID) is the most predictive of average throughput (with a raw absolute error of 619Kbps), indicating a change a cell sector has the biggest impact on measurement accuracy. The second most predictive factor is phone model, which we suspect is a result of differences in support for various 3G enhancements (e.g., HSDPA+) between devices. The ordering of the remaining seven factors from most predictive to least is: location area, hour of day, month, network connection type, indoors/outdoors, speed, and signal strength.

Summary. These findings indicate that our system is able

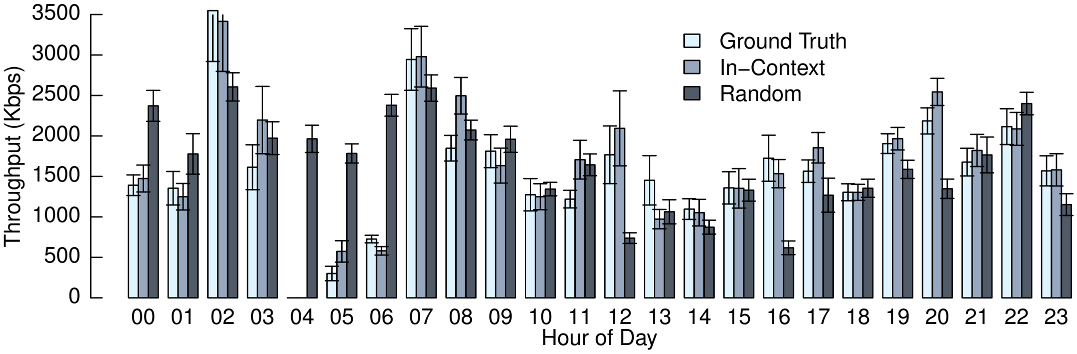


Figure 17: Mean throughput by hour of day; error bars show 95% confidence intervals [*Eval*]

Event	Correct	False Negatives	False Positives
Desk→Hand	7	0	-
Web browsing*	5	-	2
Hand→Pocket	7	0	-
In pocket*	7	-	0
Pocket→Hand	7	0	-
Hand→Desk	6	1	-

Table 3: Number of individuals for whom the service correctly/incorrectly detected position changes; *correct means not detecting any change in position

to accurately crowdsource in-context throughput measurements. Additionally, our findings on which environment factors are most predictive of throughput can be used to improve passive analysis techniques or other mobile applications seeking to approximate network performance, e.g., [25].

4.1.2 Measurement Service Benchmarks

Detecting Position Changes. To evaluate our accelerometer-based approach for detecting device position changes (§4), we asked 7 volunteers to perform a sequence of activities with their mobile devices. Users sitting at a desk were asked to pick up the device, browse to a website, then put the device in their pocket. After a minute, users were asked to remove the device from their pocket, make a phone call, and put the device back on the desk. Throughout the experiment, the service monitored for changes in position.

Table 3 shows the detection results from our user study. Our system correctly detected all but one position change, for a low false negative rate of 3%. However, some false positives occurred during the periods with no position change. No error in measurement accuracy occurs with false positives, but measurement opportunities are unnecessarily skipped.

Energy Usage. The exact energy used by the service depends on how frequently the device is active and how often measurements are requested. We broke down the service's process (Figure 14) into three pieces—idle, active monitoring, and context monitoring—and measured the energy consumed by each piece over a one minute interval. We ran the service on *Phone B* (Table 1 lists specifications), and read coarse-grained energy values from system counters every second. The device LCD was always off for consistency, and we subtracted the baseline energy the device consumes (20.5 Joules) when not running our service. We present results averaged over 30 one-minute runs for each function.

The energy consumed in one minute by each piece of the measurement service process is shown in Table 4. When the device is not active and no measurements are in progress, i.e., the service is idle, there is no noticeable energy consumption. The energy consumption slightly increases (to 0.44 Joules per minute) when the user interacts with the device and cellular interface traffic counters are monitored to detect network usage. Finally, energy consumption is the highest when using the GPS, accelerometer, and other events to detect environment changes. An optimized service could reduce this energy consumption significantly, by up to about 16 Joules, if movement speed and indoors/outdoors were detected without using the GPS.

Functionality	Energy Consumed in 1 Minute
Idle	0 Joules
Active Monitoring	0.44 Joules
Environment Monitoring (with GPS)	16.85 Joules
Environment Monitoring (no GPS)	0.15 Joules

Table 4: Energy consumed on *Phone B*

4.2 Additional Issues

Scope of interest. As discussed in §2, admins often seek performance measurements for a narrower set of users corresponding to a specific usage environment, device specification etc. Our measurement service already tracks usage environment, and device specifications can be easily obtained, enabling the device itself to determine if it is currently in-scope. However, this suffers from scaling issues, as having each device make this decision means the measurement controller must broadcast measurement requests. To avoid this, we can conduct a preliminary filtering of devices at the measurement controller. Elements in the cellular network are already aware of some environment context and device details (e.g., cell location), enabling them to inform this preliminary filtering. The remaining context (e.g., precise geo-location) and specifications still require on-device filtering. Exactly which parameters to filter at the controller and which to filter on device requires trade-offs in scalability, device overhead, and potentially missed measurement opportunities, a subject we leave for future work.

Security. Mechanisms must be in place to prevent devices from reporting phony measurements and to protect the system from being compromised. Since the threat model is not fundamentally different from existing active measurement testbeds, ideas can be borrowed from prior techniques [27, 35] to help address these issues.

Scheduling. An admin may want to run multiple concurrent experiments, so scheduling them such that they do not

interfere is important. The bottleneck and complex protocol interactions in cellular networks is typically in the "last mile" [8], so limiting geographic overlap may be sufficient.
Incentives for deployment. To lure users to adopt measurement crowdsourcing, admins may use application features or monetary incentives. Existing measurement tools have successfully recruited thousands of volunteers with these approaches [1, 5, 18]. Monetary incentives is a feasible approach for network service providers since they control both billing and measurements.

5. RELATED WORK

Our analysis and measurement system is related to work in cellular performance studies and measurement testbeds.
Cellular performance studies. Numerous measurement studies of cellular network performance have been conducted, including studies on delay [23, 25], TCP performance [24, 26], fairness and performance under load [8, 36], and application specific performance [19]. While these studies shed light on performance trends, cellular network performance is constantly changing as network providers upgrade and reconfigure their networks. Instead, we designed a system for measuring performance on demand as network infrastructure and user devices continuously evolve. Moreover, our analysis offers previously unrecognized guidelines for performing in-context measurement of cellular networks.
Measurement testbeds. Automatically initiating active measurements from vantage points of interest has received prior attention in both the wired and wireless domains. NIMI [27] and Scriptroute [35] pioneered the idea of using voluntary hosts on the Internet to conduct active network measurements and addressed many important issues regarding security and sharing of resources. DipZoom [38], ATMEN [22], and CEM [13] extended these wired testbeds with matchmaking services, event-based measurement triggers, and endpoint crowdsourcing. WiScape [33] uses measurements conducted by mobile devices to evaluate the performance of wide area networks across time and space. It focuses on collecting a small number of measurement samples to adequately characterize performance in a given epoch and zone. Similar to our approach of opportunistic crowdsourcing, these systems conduct measurements from voluntary endpoints. However, our system is the first to address the question of where and when those measurements should be conducted to obtain results that quantify the network performance users experience solely when they are interacting with their device. Thus, the techniques we designed in this paper complement these previous systems, which mostly focused on policy and scheduling issues to avoid interference and limit network disruption.

6. CONCLUSION

We argue that there are several important scenarios where there is a need to obtain in-context performance measures, i.e., the performance the network can deliver when users are interacting with their mobile devices. We conducted a large-scale empirical study using data collected across cell subscribers and controlled experiments, and concluded that such measurements must be conducted on devices that are (*i*) actively used during the measurement time frame, (*ii*) currently exchanging limited user traffic, and (*iii*) in the same position and usage environment since the device was last used. Approaches that don't take these guidelines into account are very likely to give skewed results. Based on our findings, we designed a crowdsourcing-based active measurement system that allows various parties to obtain in-context measurements in a controlled fashion. We showed, using a three month study of our system deployed across 12 users, that our measurement results are very accurate.

7. ACKNOWLEDGEMENTS

We would like to thank our 12 volunteers for running our measurement system on their mobile devices. We would also like to thank our shepherd Dina Papagiannaki and the anonymous reviewers for their insightful feedback. This work is partially supported by the National Science Foundation under grants CNS-1050170, CNS-1017545 and CNS-0746531.

8. REFERENCES

[1] AT&T mark the spot. `research.att.com/articles/featured_stories/2010_09/201009_MTS.html`.

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

[3] The national broadband plan. `http://broadband.gov`.

[4] Open internet. `http://openinternet.gov`.

[5] Root metrics. `http://rootmetrics.com`.

[6] Speedtest.net. `http://speedtest.net`.

[7] Weka. `http://www.cs.waikato.ac.nz/ml/weka`.

[8] V. Aggarwal, R. Jana, J. Pang, K. K. Ramakrishnan, and N. K. Shankaranarayanan. Characterizing fairness for 3g wireless networks. In *LANMAN*, 2011.

[9] S. Alcock and R. Nelson. Application flow control in YouTube video streams. *SIGCOMM CCR*, 41(2), 2011.

[10] M. Azizyan, I. Constandache, and R. Roy Choudhury. SurroundSense: mobile phone localization via ambience fingerprinting. In *MobiCom*, 2009.

[11] L. Bao and S. S. Intille. Activity recognition from user-annotated acceleration data. In *PERVASIVE*, 2004.

[12] M. C. Chan and R. Ramjee. TCP/IP performance over 3G wireless links with rate and delay variation. In *Mobicom*, 2002.

[13] D. R. Choffnes, F. E. Bustamante, and Z. Ge. Crowdsourcing service- level network event monitoring. *SIGCOMM CCR*, 40(4), 2010.

[14] F. Dobrian, A. Awan, D. Joseph, A. Ganjam, J. Zhan, V. Sekar, I. Stoica, and H. Zhang. Understanding the impact of video quality on user engagement. In *SIGCOMM*, 2011.

[15] J. Erman, A. Gerber, M. T. Hajiaghayi, D. Pei, and O. Spatscheck. Network-aware forward caching. In *WWW*, 2009.

[16] R. K. Ganti, P. Jayachandran, T. F. Abdelzaher, and J. A. Stankovic. Satire: a software architecture for smart attire. In *MobiSys*, 2006.

[17] A. Gerber, J. Pang, O. Spatscheck, and S. Venkataraman. Speed testing without speed tests: estimating achievable download speed from passive measurements. In *IMC*, 2010.

[18] D. Han and S. Seshan. A case for world-wide network measurements using smartphones and open marketplaces. Technical report, CMU, 2011.

[19] J. Huang, Q. Xu, B. Tiwana, Z. M. Mao, M. Zhang, and P. Bahl. Anatomizing application performance differences on smartphones. In *MobiSys*, 2010.

[20] H. Jiang and C. Dovrolis. Passive estimation of tcp round-trip times. *SIGCOMM CCR*, 32(3), 2002.

[21] R. Kohavi and R. Longbotham. Online experiments: Lessons learned. *Computer*, 40, 2007.

[22] B. Krishnamurthy, H. V. Madhyastha, and O. Spatscheck. Atmen: a triggered network measurement infrastructure. In *WWW*, 2005.

[23] M. Laner, P. Svoboda, E. Hasenleithner, and M. Rupp. Dissecting 3g uplink delay by measuring in an operational hspa network. In *PAM*, 2011.

[24] X. Liu, A. Sridharan, S. Machiraju, M. Seshadri, and H. Zang. Experiences in a 3g network: interplay between the wireless channel and applications. In *MobiCom*, 2008.

[25] J. Manweiler, S. Agarwal, M. Zhang, R. Roy Choudhury, and P. Bahl. Switchboard: a matchmaking system for multiplayer mobile games. In *MobiSys*, 2011.

[26] K. Mattar, A. Sridharan, H. Zang, I. Matta, and A. Bestavros. Tcp over cdma2000 networks: a cross-layer measurement study. In *PAM*, 2007.

[27] V. Paxson, J. Mahdavi, A. Adams, and M. Mathis. An architecture for large scale internet measurement. In *IEEE Comm. Magazine*, 1998.

[28] F. Qian, Z. Wang, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck. Characterizing radio resource allocation for 3g networks. In *IMC*, 2010.

[29] 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 *MobiSys*, 2011.

[30] L. Ravindranath, C. Newport, H. Balakrishnan, and S. Madden. Improving wireless network performance using sensor hints. In *NSDI*, 2011.

[31] S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, and M. Srivastava. Using mobile phones to determine transportation modes. *ACM Trans. Sensor Networks*, 6(2), 2010.

[32] F. Ricciato. Traffic monitoring and analysis for the optimization of a 3g network. *IEEE Wireless Communications*, 13(6), 2006.

[33] S. Sen, J. Yoon, J. Hare, J. Ormont, and S. Banerjee. Can they hear me now?: a case for a client-assisted approach to monitoring wide-area wireless networks. In *IMC*, 2011.

[34] T. Sohn, A. Varshavsky, A. Lamarca, M. Y. Chen, T. Choudhury, I. Smith, S. Consolvo, J. Hightower, W. G. Griswold, and E. D. Lara. Mobility detection using everyday gsm traces. In *Ubicomp*, 2006.

[35] N. Spring, D. Wetherall, and T. Anderson. Scriptroute: a public internet measurement facility. In *USITS*, 2003.

[36] W. L. Tan, F. Lam, and W. C. Lau. An empirical study on the capacity and performance of 3g networks. *IEEE Trans. on Mobile Computing*, 7(6), 2008.

[37] F. P. Tso, J. Teng, W. Jia, and D. Xuan. Mobility: a double-edged sword for hspa networks. In *MobiHoc*, 2010.

[38] Z. Wen, S. Triukose, and M. Rabinovich. Facilitating focused internet measurements. *SIGMETRICS Perform. Eval. Rev.*, 35(1), 2007.