NORTHWESTERN UNIVERSITY

Adopting a Gateway Centric View for Cellular Network Content Delivery

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree

DOCTOR OF PHILOSOPHY

Field of Computer Science

By

John Paul Rula

EVANSTON, ILLINOIS

June 2017

Committee

Fabián Bustamante (Chair), *Northwestern University*

Aleksandar Kuzmanovic, *Northwestern University*

Peter Dinda, *Northwestern University*

Jon Crowcroft, *University of Cambridge*

ABSTRACT

Adopting a Gateway Centric View for Cellular Network Content Delivery

John Paul Rula

Mobile traffic is expected to grow tenfold by 2019, topping 24 exabytes of monthly traffic and accounting for nearly half of all Internet traffic. This growth is driven by the increasing number of smart phones and tablets, and the data demands of high bandwidth services enabled by next-generation cellular networks such as LTE/5G. As in the wired Internet, network usage is dominated by content consumption, with the vast majority served through content delivery networks (CDNs).

CDNs host and replicate popular content across thousands of servers worldwide, directing users to "nearby" servers. This replica selection is a key determinant of client performance, yet replica selection for cellular clients has previously been overlooked, due to high radio latency, inconsistent throughput, and a limited number of ingress locations which dominated end-to-end latency. NGCNs and their improved performance place a renewed emphasis on replica selection policies for cellular clients. We find that the performance of existing replica selection systems in cellular networks is hindered by their opacity, the dynamic assignment of clients to infrastructure components, the emergence of centralized DNS within cellular networks, and the growth of public DNS in global mobile operators. This opacity prohibits network probes from entering these networks, rendering existing monitoring and measurement systems ineffective.

In this dissertation, I argue for the centrality of cellular network packet gateways (PGWs), and that this centrality has critical implications on the architecture, characterization, and performance of cellular networks. PGWs separate the interior mobile network from external data networks, and define the independent network partitions which compose modern cellular networks. I posit that understanding the locations of PGWs and their allocation of clients constitutes sufficient network topology coverage. The presence of PGW's on all routes to

and from cellular clients make them ideal proxies of client latency for network services. I demonstrate techniques for characterizing cellular networks which allow both the discovery of PGW locations and their assignments of mobile clients. I designed and implemented two live systems which utilize these techniques to characterize cellular infrastructure: TILLER which uses instrumented mobile devices to characterize cellular networks, and MACHETE which uses traces from external vantage points to accomplish this characterization at a global scale. I introduce a novel method of content replica selection which chooses cellular client servers based on the location of a client's PGW, called Gateway-Based Replica Selection (GBRS), and show this achieves near optimal replica selection for cellular clients.

Acknowledgements

I would like to thank all of those who have believed in, inspired, and supported me throughout this journey. I would not have made it this far without the gracious love and encouragement of my friends and family. To my parents, thank you for for all of your love and unending support. Even at my lowest points, you never gave up on me. To my sister, you've always had my back and helped me to see beyond life's trivialities.

I am eternally grateful to my advisor, Fabian Bustamante, for first giving me the opportunity to begin this journey as an undergraduate, and for taking the chance on an unqualified misfit with little computer science experience. I am forever grateful for you seeing the potential in me, and for tirelessly helping me to fully develop it.

I want to thank my committee for their guidance throughout this process. To Aleksander Kuzmanovic, thank you for his straightforward feedback and bluntness, Peter Dinda for his thoroughness, and Jon Crowcroft for his insights.

I have had the opportunity to work with great researchers through my summer internships. I would like to thank my collaborators at Microsoft Research India, Vishnu Navda, Ranjita Bhagwan and Saikat Guha, for granting me a unique and amazing experience, and showing me how fulfilling pure research can be. I would like to thank those I worked with at Akamai, Marcelo Torres and Pablo Alvarez, for trusting me with the freedom to pursue my work.

To my lab mates for keeping me sane throughout this process: Maciej Swiech, Mario Sanchez, James Newman, and Zach Bischoff. Your continued antics kept my spirits high and got me through many hard times. To Kyle Hale, for being an incredible friend and roommate, for always being supportive, and for the many late night discussions exploring the universe. To John Otto for being a great mentor and close friend; and to Marcel Flores, for showing me what hard work looks like.

Last, I want to thank Chenault Taylor for being my guiding light through this journey. Your unending love and support have allowed me to grow beyond what I ever thought was possible. You have my eternal love.

List of Abbreviations

CDN	Content Delivery Network
GGSN	Gateway GPRS Serving Node
MNO	Mobile Network Operator
PGW	Packet Gateway
SGSN	Serving Gateway Support Node
SGW	Service Gateway
UE	User Equipment

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Mobile traffic is expected to grow tenfold by 2019, topping 24 exabytes of monthly traffic and accounting for nearly half of all Internet traffic [35]. This growth is driven by the increasing number of smart phones and tablets, and the data demands of high bandwidth services enabled by next-generation cellular networks such as LTE [1]/5G. As in the wired Internet, network usage is dominated by content consumption, with the vast majority served through content delivery networks (CDNs) [85].

CDNs host and replicate popular content across thousands of servers worldwide. Key to their performance is accurate client localization, which allows CDNs to redirect users to "nearby" replica servers. This redirection is commonly based on heuristics such as the location of a user's DNS resolver [67, 75, 109] or IP address [27]. Independent of the chosen heuristic, CDNs conduct large numbers of both active and passive measurements to determine the relative distance and performance of replica servers to and from clients. Despite its role in end-usernperformance, the relative effectiveness of existing approaches has been overlooked given the high radio latency, inconsistent throughput and limited number of network ingress locations which previously dominated path latencies.

Improvements in radio technology and the expanded core network infrastructure of NGCNs places a renewed emphasis on replica selection policies for cellular clients. For

[1] Although already deployed in certain operators, LTE accounted for only 15% of mobile subscriptions in 2015 [45].

example, clients in LTE networks average 69.5 ms of end-to-end latency [61], compared to the nearly 400 ms of average latency experienced in 3G networks [63]. This transition to LTE has also seen a growth in the numbers of packet gateways (PGWs) for cellular operators, from the 4-6 reported for U.S. MNOs in 2011 [117] to the up to 40 we have found for these same networks (§ 5).

In this work, we find the effectiveness of existing replica selection systems in cellular networks is hindered by these networks' opacity, growing adoption of public DNS usage, and the prevalence of centralized resolver architectures. The lack of visibility into these networks, caused by the NAT and firewall policies of cellular operators, prohibits probes from entering and measuring to common network landmarks, such as clients' DNS servers, depriving CDNs of fine-grained path information for cellular clients. Concurrently, the rise of centralized resolver architectures within MNOs, and the increased reliance of public DNS within cellular networks, both adversely affect existing DNS-based replica selection [85].

1.1 Thesis Statement

In this dissertation, I argue for the centrality of cellular network packet gateways (PGWs), and that this centrality has critical implications the architecture, characterization, and performance of cellular networks. PGWs separate the interior mobile network from external data networks, and define the independent network partitions which compose modern cellular networks. We posit that understanding the locations of PGWs and their allocation of clients constitutes sufficient network topology coverage. The presence of PGW's on all routes to and from cellular clients make them ideal proxies of client latency for network services. We demonstrate techniques for characterizing cellular networks which allow both the discovery of PGW locations and their assignments of mobile clients. We designed and implemented two live systems which utilize these techniques to characterize cellular infrastructure: TILLER which uses instrumented mobile devices to characterize cellular networks, and MACHETE

which uses traces from external vantage points to accomplish this characterization at a global scale. We introduce a novel method of content replica selection which chooses cellular client servers based on the location of a client's PGW, called Gateway-Based Replica Selection (GBRS), and show this achieves near optimal replica selection for cellular clients.

One of the challenges when studying cellular networks and their underlying infrastructure is the lack of available information about these networks, their topologies and their policies. The aforementioned opacity limits the effectiveness of existing tools and measurement techniques. As part of this dissertation, we developed a mobile platform, ALICE , for the measurement and characterization of cellular network infrastructure, and deployed it across four public mobile applications over a three year period. This tool provides programmable network experiments run from volunteer mobile devices. We used ALICE equipped clients to conduct an in-depth characterization of global mobile operators, looking specifically at (*i*) DNS infrastructure, (*ii*) PGW allocation and assignment dynamics, and (*iii*) inter-domain routing policies.

Our characterization of cellular DNS infrastructure reveals an opaque and indirect resolver structure, with inconsistent client-to-resolver mappings. Since DNS impacts nearly all network services on mobile devices, and plays a critical role in CDN replica selection, this dynamic behavior adversely impacts these services' performance. We discovered instances where CDNs were strictly mapping replica servers to individual resolvers, meaning changes in client resolvers resulted in entirely independent, and distant, replica server sets.

We utilize the longitudinal measurements collected by ALICE clients to look into the spatial and temporal assignment patterns of mobile clients to PGWs. A client's PGW plays an integral role in determining paths to and from network services. Knowing the locations of these PGWs, and more importantly, determining a client's assigned PGW location are critical for successful replica selection in cellular networks. We find dynamic assignment occurs between clients and cellular infrastructure, often with little spatial and temporal

locality. This means clients may be assigned to different, and distant, PGWs over relatively short time scales.

We use a combination of over 10 million mobile client traceroutes, directed towards large content providers as well as other mobile clients within their own, and across other mobile operators, to characterize the AS-level structure and connectivity of cellular networks. We find that the structure of MNOs is often composed of multiple cooperating ASes, each performing a individual *logical* function such as housing cellular clients, or interconnecting multiple PGWs. By combining these multiple components into a single *logical domain*, we more accurately are able to characterize inter-domain interactions of these networks.

Based on the results from our characterization, and the centrality of network PGWs, we propose a novel approach for content replica selection, which chooses servers based on the location of a client's PGW as opposed to existing heuristic. We find this Gateway Based Replica Selection (GBRS) provides near optimal replica selection for cellular clients, providing equal or better performance in nearly all instances compared to existing replica selection systems, and upwards of 60% improvement in over 20% of cases.

The challenge of this approach comes not just from discovering the number and locations of network PGWs, but accurately determining clients' current PGW. We develop a methodology which both discovers cellular network PGW locations as well as their current mobile clients. Our techniques exploit the relative stability of IP address pools at PGWs. Once mapped, we are able to identify network PGWs by client IP addresses. We introduce two separate systems which utilize this methodology to characterize cellular network infrastructure. We design and implemented a system – TILLER – which characterizes cellular networks through the use of distributed, instrumented mobile clients. TILLER is able to provide highly accurate maps of cellular network infrastructure, yet suffers from coverage problems endemic to mobile systems. To overcome this limitation, we created MACHETE, system for characterizing cellular networks at a global scale. MACHETE uses traces from a

number of distributed external vantage towards cellular IP space to cluster cellular clients by detected PGWs. Both TILLER and MACHETE are able to partition cellular networks with over 98% accuracy.

1.2 Summary of Major Contributions

The primary contributions of this theses are as follows:

- We show that existing approaches for client localization are ineffective for cellular networks, and that this causes suboptimal replica server selection for CDNs.
- We present a tool for mobile end-hosts to measure and characterize next generation cellular networks. We developed a mobile experimentation platform, ALICE , designed for exploring cellular network infrastructure and its interconnection with content delivery networks.
- Using three years of data from over 1900 volunteer mobile clients, we characterize the DNS infrastructure, inter-domain connectivity and network assignment dynamics of global cellular networks.
- We propose an alternative approach for CDN replica selection based on a client's assigned packet gateway, called Gateway-Based Replica Selection (GBRS). We develop measurement techniques which allow PGW discovery and partitioning of cellular networks.
- We demonstrate this measurement methodology through two separate systems. The first system, TILLER , utilizes instrumented mobile clients to collects and communicate cellular network location information to content delivery networks. The second, MACHETE , uses traces from external vantage points towards cellular IP addresses to map IP addresses to assigned PGWs.

1.3 Roadmap

This dissertation is divided into the following chapters. In Chapter 2, we describe relevant background information and related work for cellular networks, cellular network performance, content delivery networks. Chapter 3 expands on the factors motivating this dissertation, including the rise of mobile traffic and the ineffectiveness of existing replica selection systems in cellular networks. I describe the causes of this poor performance, including the opacity of these networks, and the rise of both centralized resolvers and public DNS services which limit the effectiveness of existing resolver based systems.

In Chapter 4, we make the case for PGW centrality in content replica selection, and introduce GBRS. I present the characterization of LTE network infrastructure in Chapter 5, first by investigating the cellular DNS infrastructure and its impact on existing replica selection systems, and then to characterize deployments of network PGWs and their client assignment behavior. Chapter 6 presents the results of our network-level characterization of their intra, and inter-domain connectivity.

In Chapter 7, we introduce our system for efficient mobile network characterization, TILLER . We motivate TILLER's design from an in depth study of coverage potential from individual mobile vantage points, and use these findings to inform TILLER's adaptive probing technique. We present MACHETE , a scalable solution to cellular network PGW identification and client mapping in Chapter 8. MACHETE uses traces towards cellular clients to partition cellular networks based on trace behavior.

I present an overview of my contributions, and conclude this dissertation in Chapter 9. I describe the design and implementation of ALICE, our tool for the exploration and characterization of cellular networks in Appendix A.

Chapter 2

Background and Related Work

This dissertation draws from multiple areas of prior research, including current and future cellular network infrastructure, network performance over wireless links, network topology discovery, as well as content delivery network design and operation. We provide provide relevant background information for each, and discuss and place our contributions in the context of prior research endeavors.

2.1 Next-Generation Cellular Networks

Next-generation cellular networks, first coined in 2006 from the Next Generation Mobile Network Alliance [78], describe data-centric mobile connections which enable wide area mobile broadband connections. LTE networks, initially deployed in 2009, are considered the first technology to meet these NGCN standards. LTE networks are expected to serve the majority of North American devices by 2018 [34]. LTE allow speeds up 150/75 Mbps of downstream/upstream throughput, over an order of magnitude faster than 3G networks [61]. The newly proposed successor of LTE, known as 5G networks, propose several orders of magnitude improvements in throughput, promising speeds up to 100 Gbps, and single millisecond access latencies [78].

LTE introduces several changes to its core network architecture which positions it primarily as a data-first network, compared to focus on voice calls in prior network technologies. The core networks in LTE contain substantial changes, transitioning from

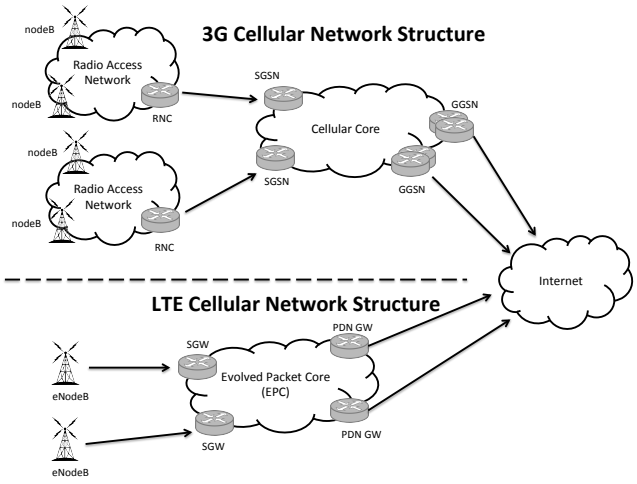


Figure 2.1: Network architecture changes cellular networks between 2/3G and LTE networks. LTE introduces a simpler, flatter network structure and an all-IP network.

the circuit-based data connectivity of prior technologies to an all-IP network [33]. This eases management of cellular core networks, thus allowing quicker expansion to meet growing demand. LTE networks also implement a flatter core architecture, removing the previous Radio Access Network (RAN) layer, combining its functionality into each eNodeB tower. This simplifies the cellular core, and reduces latencies between clients and network services. The key infrastructure components in LTE networks are the serving gateway (SGW), the packet gateway (PGW) and the mobility management entity (MME). The SGW acts as the mobility anchor for mobile clients. The PGW bounds the cellular core network from external data networks (e.g. the Internet). All traffic from mobile clients routes through a PGW instance before reaching the Internet. The MME maintains state information on mobile client locations through the operator's network. These architectural components for LTE and prior network technologies are shown in Figure 2.1.

Operationally, modern cellular networks are divided into independent partitions based on PGW instances. These partitions exist as *logical* partitions, in contrast to spatial

partitioning, of network resources. From the perspective of cellular data traffic, all client traffic must first traverse the region's PGW before routing to other destinations. This is true even for clients within the same including clients in the same partition. Additionally, clients are isolated within their current partition, with no visibility into other network partitions.

2.2 Mobile Network Performance

Performance over cellular networks is determined by a large number of interconnected factors which span multiple layers of the networking stack. These include radio latencies determined by physical and MAC layer interactions, the performance of existing transport protocols such as TCP over unpredictable and lossy wireless links, and path latencies determined by core network latencies and distances to content replicas. In this section we describe these different sources of cellular network performance degradation.

2.2.1 Components of Cellular Network Latency

Cellular device latencies are composed of both of control plane and user plane latencies [77]. Control plane latencies involve handsets negotiating with the core network, attaching themselves to cellular networks and reserving resources within that network. We measure control plane latency as the latency needed for the UE to begin sending or receiving data. User plane latencies represent the time it takes for a data packet to travel from a UE to its destination server.

Radio Access Latency. The control plane latency measures the time to transition a UE from an idle state to one which is capable of sending and receiving data traffic. Control plane latency consists of a series of interactions between the UE, the radio tower, and core network management systems (i.e. MME).

For the UE radio there is a delay to switch the cellular radio from an idle to an active state. These multiple states are known as radio resource states (RRC), and govern the activity and power usage of device radios. Cellular radios consist of multiple radio states,

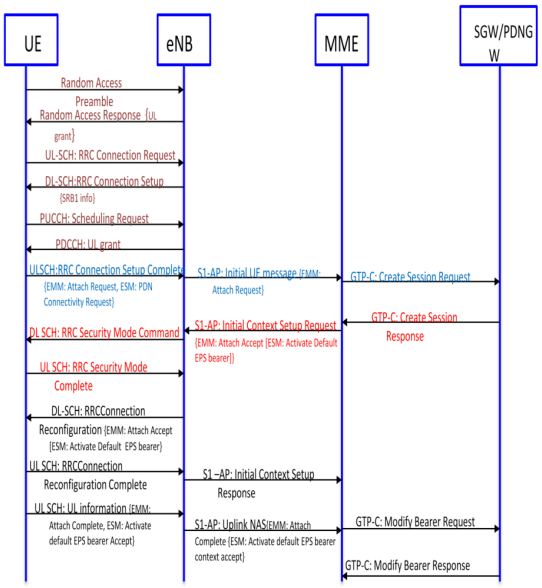


Figure 2.2: Call setup procedure for LTE. Adapted from Mohan et al. [77].

with 3 for 3G radios and 2 for LTE radios. The time taken to transition from an idle state to a transmission state is known as the promotion delay. For 3G networks this was measured to average 582 ms, and in LTE networks averages 260 ms [62].

Carrier dependant state machines govern the transitions and set the length of timeouts to transition from one phase to the other. These state machines have been documented previously by Quian et al. [89] for 3G network and Huang et al. [62] for LTE networks, finding idle timeouts (known as tail times) vary between carriers from 11-16 seconds for investigated 3G carriers and averaging 11.5 seconds in measured LTE carriers.

Once a UE is ready to transmit, it must then negotiate with the cellular network itself, first to request to transmit over controlled spectrum, then to request attachment to the network. This negotiation procedure for network attachment is displayed in Figure 2.2, featuring 20 independent messages between UE, radio tower, and core network components such as the MME and SGW/PGW – all required before client data can be transmitted. All of this interaction results in what is known as PDP context, a state vector for that

cellular devices which consists of identifiers such as device IP addresses, and radio bearer information.

Core Network Latency. User plane latencies represent the time it takes for a data packet to travel from a UE to its destination server. User plane latencies are composed of core network latencies as well as traditional Internet path latencies [16]. Core network latencies result from the distance between mobile devices and packet gateways, as well as congestion over the radio link.

Internet Latency. In the context of content consumption, Internet path latencies are largely determined by the distance between content servers and cellular PGWs. As we show throughout this dissertation, Internet path latencies are comprising ever greater fractions of end-to-end latencies, driven by the improvements in cellular radio and core network technologies.

2.2.2 Transport Performance Over Wireless Links

The quickly changing bandwidth, and lossy environment of cellular networks [73] adversely affect general purpose transport protocols such as TCP. In this section we discuss the prior work diagnosing poor transport performance over wireless links, and their proposed solutions.

A large body of work has diagnosed the problems facing TCP performance within cellular networks [24, 25]. The work by Chan et al. [26] was one of the first to diagnose the poor performance of TCP over the variable link quality, and non-congestive loss rates of TCP. Since this early work, solutions have come in the form of MAC-level retransmissions, and large packet queues at radio towers. These changes effectively mask radio losses from TCP state machines.

These early solutions have themselves created performance problems stemming from excessive queue lengths, known as bufferbloat [64]. Recent work has attempted to address these additional challenges, designing clean-slate transport protocols to replace TCP for

wireless connections. The Sprout protocol by Winstein et al. [115] optimizes cellular network performance through stochastic forecasts of available bandwidth. Similarly, the Versus protocol by Zaki et al. [119], uses adaptive congestion control for the quickly changing link properties of cellular connections.

Another response to the heterogeneous environment of wireless links is to split TCP connections; with one for wired Internet paths, and another for the wireless link. Gonzalez et al. [57] was one of the first to advocate for the use of these performance-enhancing-proxies (PEPs), to compensate for the high latency and complex MAC-layer behavior of cellular technologies. Recently, the use of these PEPs has been studied and characterized in various cellular networks. [55, 94, 118].

Our work is complementary to these approaches since, replica placement plays an important role in overall client performance. Improved replica selection reduces overall latency, which will only improve overall end-to-end performance.

2.3 Network Topology Discovery

This work investigates the physical infrastructure cellular networks, the locations of key infrastructure components such as network PGWs, and peering relationships. My work builds off the vast literature on Internet measurement and topology mapping compiled over the last 15 years. Below we highlight some of the most influential work of Internet mapping and measurement, and look at recent efforts to extend this towards cellular networks.

2.3.1 Wired Network Exploration

Early efforts at network mapping include Rocketfuel from Spring et al. [106]. Rocketfuel performed efficient probing from traceroute and looking glass servers within several large ISPs, mapping points-of-presence (PoPs) and router connectivity. Rocketfuel performed the dual task of both identifying ISP points of presence (PoPs), and geolocating them through techniques developed to decode location hints in router hostnames. More recently Durairajan

et al. [43, 44] showed that physical layer maps of long-haul fiber infrastructure provided valuable information about Internet topology. These techniques have become standard for Internet mapping, and we transfer these techniques to our exporation of cellular networks.

As large ISPs implemented more complicated traffic engineering policies such as MPLS, the effectiveness of traceroute probes were hindered by the multiple paths taken by packets, and the network tunneling procedures which failed to decrement TTL, becoming "invisible" to these probes. In response, Augustin et al. [12] created Paris Traceroute as a tool to view the multiple paths. Paris Traceroute sends out packets with different combinations of header options since common traffic engineering policies attempt to keep flow-level routing consistent. The shift of operators to more opaque network management solutions such as MPLS motivated Sherwood et al. [102, 103] extended router level mapping utilizing combinations of TTL based traceroute along with the IP's route record option to map router level paths including those hidden by MPLS network tunnels. We face similar struggles with network topology discovery in cellulr networks, which widely use tunneling and sub-layer 3 routing within their core networks.

2.3.2 Measuring Cellular Networks

While these approaches have proven themselves in numerous Internet mapping and topology discovery efforts, they are less effective for exploring cellular networks due to the opacity of these networks (§ 3.4), and the lack of measurement vantage points.

Cellular network exploration suffers from the widespread opacity of cellular networks, which prevent common approaches such as those described in the previous section. For instance, many MNOs prevent probes launched from external vantage points from reaching any of their clients or infrastructure. While the externalities of network opaqueness have previously been studied in edge networks by Casado et al. [22], these were focused mainly on the obfuscation of multiple home devices by broadband router NATs. This impact of cellular

network opaqueness is orders of magnitude larger than this, as tens of thousands of clients sit behind operator carrier grade NATs (CGNs) [113]. Efforts to explore the infrastructure and policies of cellular networks therefore require measurement from instrumented mobile devices, typically deployed as volunteer applications [47, 61, 63, 111, 113].

Cellular networks lack common measurement vantage points, such as Looking Glass servers [18], the Archipelago (Ark) project [23], and crowdsourced efforts such as RIPE Atlas [93]. This coupled with the aforementioned opacity make cellular network measurements difficult to obtain. The power and computational limitations from mobile devices also exclude several common volunteer end-host monitoring systems such as Dasu [96] and News [31]. Further complicating the issue is the independence of cellular network partitions, which prohibit internal vantage points from accessing other network regions, requiring greater and coordinated vantage point deployment.

These challenges in mobile network measurement motivated the creation of our mobile experiment engine, ALICE (App. A) and its supporting delivery applications.

2.3.3 Prior Cellular Network Characterization Efforts

Despite these challenges, there have been several efforts to characterize cellular networks, their infrastructure and policies. This work builds on the multiple discoveries of this prior work

The importance of gateways in cellular networks has previously been recognized, motivating several early efforts for their characterization. The work by Xu et al. [117] which characterized the cellular 2G and 3G infrastructure of the four largest U.S. MNOs. The authors cluster cellular clients based on their DNS servers, locating GGSN instances by client's recorded locations. Noting the importance of client gateways in content delivery, the authors recommended placing content replicas near cellular packet gateways. Similar efforts were conducted for earlier 4G network deployments by Zarifis et al. [120]. Here gateways

were identified using the first public IP address in an outbound mobile client traceroute, with gateways located based on applicable hostname location hints, or lacking those, the centroid of client locations.

Other work has looked at policies governing network identifiers in cellular networks such as IP addresses. Balakrishnan et al. [14] investigated the IP assignment to cellular clients, measuring the consistency and stickiness of cellular IP addresses, finding both the potentially rapid assignment of IP addresses to clients, and consequently the ineffectiveness of existing IP geolocation services.

Our work builds off of these earlier efforts. Our contributions differ from these early works in three main areas: (*i*) a focus on replica selection rather than replica placement (*ii*) a more principled methodology for gateway identification and localization which covers greater heterogeneity in MNO configuration (*iii*) longitudinal measurements of MNO policies such as IP and gateway assignment.

2.4 Content Delivery Networks

Since their emergence in the late 1990s, content delivery networks (CDNs) have become the primary vehicle for serving Internet content. CDNs replicate content across geographically distributed sets of servers and redirect clients to nearby replicas to reduce access time to a web site [70]. There is extensive literature studying CDNs, investigating their performance [4, 59,60,65,67], management [68,112] and architectures [66,83]. Our work seeks optimal replica selection for cellular clients.

Krishnamurthy et al. [67] performed one of the earliest studies of content delivery networks and evaluated the effectiveness of replica selection for client performance. The effectiveness of DNS servers for content replica selection has been extensively explored before (e.g., [75, 91, 100]). In addition, many research efforts on future name service architectures

have incorporated replica selection into its core services, including the DONAR [114] and Auspice [101].

Content replica selection, also known as request routing, is the process of directing client requests to a "nearby" replica server, and is one of the key components of any content delivery network. Effective replica selection is critical for achieving high client performance, as well as load balancing Internet demand across CDNs' networks. CDNs employ several different mechanisms for replica selection, including DNS-based server selection [41], end-user mapping [27], and anycast routing [6, 7]. We provide an overview of each method below.

DNS-Based Selection. DNS-based selection uses the location of a client's local DNS resolver as a proxy location for server selection, and is also the most popular method in use today. DNS-based server selection provides the most control over client requests at the expense of system complexity.

A client requesting a website through DNS-based selection first performs a DNS resolution of the websites hostname through their local resolver. Though one or more recursive resolutions, the clients local resolver eventually contacts the authoritative DNS of the CDN, which chooses a replica IP to return based on the location of the requesting resolver. A common approach for this redirection is through CNAME aliases. CDNs typically attain greater control over these assignments by using short TTLs for addresses [109].

Despite their popularity, the effectiveness of DNS-based server selection faces several challenges. The first is that local resolvers are not always good approximations of user locations. Early work from Mao et al. [75] found that 36% of clients use DNS servers not in their same AS. Another is the recent growth of public DNS services such as GoogleDNS [58] and OpenDNS [84], and the deployment of centralized and hierarchical nameserver architectures within ISPs [8, 99]. These remote resolvers further obfuscate the distance between clients and their local resolvers, resulting in poorly located replica severs [85].

End-User Mapping. A proposed solution to the inaccuracies of DNS-based server selection is to select replica servers based on the requesting client IP address. Known as the EDNS-Client-Subnet Extension (ECS) [38, 108], this approach extends existing DNS protocols to include client IP addresses in requests from local resolvers. While this end-user mapping [27] potentially providing improved client localization, it greatly increases the complexity of CDN systems.

Anycast Selection. Anycast routing utilizes BGP routing protocols to direct clients to their closest replica server. In an anycast CDN, multiple servers advertise the same IP address, allowing BGP to calculate the shortest paths between clients and servers. While anycast selection suffers from its own set of problems, namely the lack of control over client routes and slow update propagation, it can greatly reduce the complexity of selection systems [49].

Our work is the first to investigate the effectiveness of DNS-based server selection on cellular clients. Our analysis supported by our detailed characterization of global cellular infrastructure, and our performance analysis from instrumented clients, highlights its poor performance. We build on prior work improving DNS-based selection, end-user mapping, and CDN analysis to create new approaches adapted to the unique constraints of cellular networks.

Chapter 3

Motivation

3.1 Overview

This dissertation is motivated by the emergence of multiple growing trends, which together are increasing the impact of replica selection on mobile client performance. These include (*i*) the improvements in cellular radio technology and expansion of cellular infrastructure, (*ii*) the poor performance of existing replica selection systems, (*iii*) the difficulty locating cellular clients, and (*iv*) the growing volume of mobile traffic.

The rise of NGCNs have brought vast improvements in cellular radio technology which have quickly, and greatly, increased demand from mobile devices. When compared with 3G networks, 4G LTE presents a significantly different network, with a radically larger number of ingress points, and offers much lower radio access latency and variance. We show that these changes make accurate content replica selection critical to the performance of end users in cellular networks.

We find that existing replica selection systems perform poorly in cellular networks caused by poor client localization. We show that this poor localization is caused by the opacity, increasing use of public DNS services, and centralized resolver structures of cellular networks.

We discuss how the growth of mobile data means CDNs' poor performance will affect an increasing fraction of clients. We also highlight how the poor accuracy of existing systems threatens to disrupt effective load balancing.

We discuss how upcoming advancements in NGCNs, which will greatly improve the latency and throughput of these connections, will impact the contributions of this work. We posit that the improved latencies in the upcoming 5G specification, and critical application which require them, will only increase the impact of server selection.

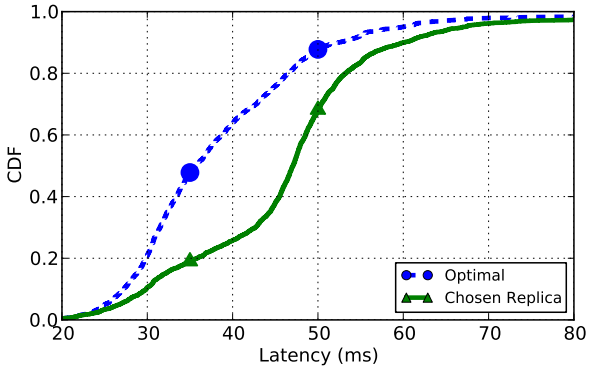
3.2 Advancements of NGCNs

Recent advances in cellular technology have greatly improved the performance of the radio link, and the growing demand of cellular traffic has brought rapid expansion to cell network infrastructure. This has resulted in the location of selected replicas becoming a larger fraction of end-to-end latency, and with a growing number of network ingress points, CDNs have a greater chance for selecting incorrectly.

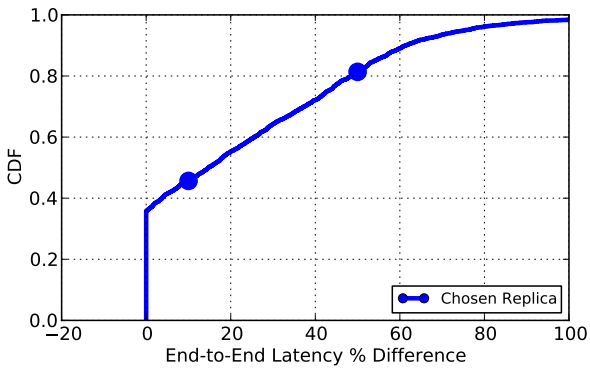
In 3G and previous cellular technologies, the significant fraction of radio access latencies dominated end-to-end latencies, and were measured to be 400 ms at median [63]. From the perspective of content delivery, these high latencies dilute any of the benefits of closer content replicas, meaning CDNs had little control over the end-to-end latency of mobile clients.

The current and future improvements of NGCNs introduce radio technology with greatly reduced access latencies with more stable performance. LTE networks, for instance, offer access latencies close to 10s of milliseconds,an order of magnitude lower than in 2G/3G [62], resulting in end-to-end latencies less than 70 ms [61] at median, 5.7 times lower than prior generations.

The expanding cellular infrastructure, and the increasing number of cellular gateways, increases the difficulty for selecting a client's optimal replica. Over the past 5 years, the number of gateways per U.S. operator has increased from the 4-6 reported by Xu et al. [117] in 2011, to the 10-20 reported by Zarifis et al. [120] in 2014. Our own results in Chapter 5 find between 25-40 PGWs in US operators. The significantly larger number of ingress points,



(a) Performance of existing CDN redirection vs. optimal.



(b) Relative performance of existing CDN redirections compared to optimal replica.

Figure 3.1: Performance of current replica selection systems of a large CDN for clients in a large U.S. mobile network operator. The CDN has agreements with network operator with replicas located within the network. In addition, the CDN receives periodic information regarding LDNS and client locations. Even in this "ideal" case, optimal selection is only achieved in less than 40% of measured cases, and over 20% of clients directed to replicas 1.5x further away.

a trend clear in Zarifis et al. [120] and in our own results (Sec. 5), means that CDNs have more options for placing and choosing content caches.

These architectural changes and the radical improvements in radio access technology, suggest it is time to revisit the effectiveness of content delivery and the impact of existing server selection policies in cellular networks.

3.3 Ineffectiveness of Existing Replica Selection

Existing CDN replica selection systems are challenged by cellular network structure and policies. We show that current CDN redirection policies choose poorly performing replicas even in best case scenarios, where the CDN has partnerships with mobile operators.

We highlight these inefficiencies by measuring the performance to a likely set of caches for mobile operators. For each MNO, we selected a subset of a CDN's replicas which were either located *within* the operator's network in addition to replicas located at major PoPs and Internet exchanges near MNO coverage areas. This server information was provided

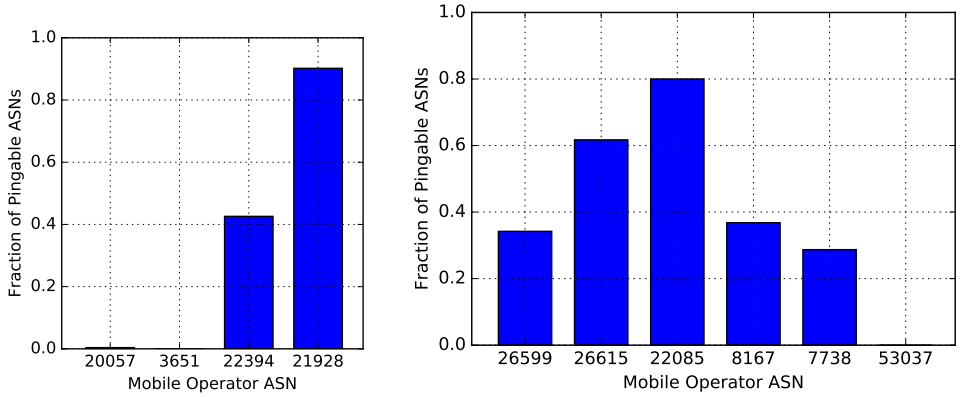
by a large CDN. Using measurements from the mobile clients running the ALICE library (App. A), we measured network latency to each replica in the each MNO's set or servers, and recorded the minimum round-trip-time to each replica. We compared the latencies seen in each experiment to the replica chosen by the CDN.

Figure 3.1 shows the results of these experiments for clients in a large U.S. mobile operator. The relatively poor performance of existing replica selection is shown both by the overall latency difference between assigned and optimal replicas (Fig 3.1a) as well as the relative performance differential of the assigned replica over the optimal (Fig. 3.1b). Clients were assigned to their optimal replica less than 40% of the time, and in 20% of cases, clients were assigned to servers 50% further away than optimal.

The performance differential is even more impressive when one considers that the chosen CDN and operator have a deep partnership consisting of shared information of network topology and collocated replicas within that operator's network. This indicates the problem with cellular request routing stems not just from inaccessibility from cellular operator's networks, but from the ineffectiveness of existing replica selection policies, such as DNS-based selection, in cellular networks.

3.4 Problems Locating Cellular Clients

Our investigations into the causes of this poor performance revealed that (*i*) the opacity of cellular network infrastructure and clients, (*ii*) a rise in public DNS use for cellular clients, and (*iii*) the use of centralized resolver structures in cellular networks all degrade the performance of existing replica selection systems. Each of these in their own way obscure the locations of clients from CDNS, resulting in poor client localization and suboptimal replica selection.



(a) US MNOs (b) BR MNOs

Figure 3.2: Resolver opacity for U.S. and Brazilian operators. While entirely dependent on operator policy, many MNOs such as Sprint and AT&T prohibit all probing of cellular LDNS resolvers.

3.4.1 Cellular Network Opacity

Opacity, in this context, is the inability of externally launched probes (e.g. ping, traceroute) to penetrate a particular network. For cellular networks, this opacity is due to firewall and NAT policies intended to protect mobile users, which without would be open to several attacks including data quota drain, DoS flooding and battery drain [71]. While the effects of opacity have previously been studied by Casado et al. [22] looking at the impact of NAT usage in edge networks, cellular networks implement this at a much greater scale, presenting network-wide opacity.

Without the ability to probe these networks, CDNs cannot measure latencies between replica servers and client landmarks. They are forced to make assumptions about these black-box networks, including hypothesizing the locations of clients and their DNS resolvers.

Opacity policies differ in the extent of their impermeability between operators, challenging any one method as a general solution for all networks. We exemplify this problem by showing the number of LDNS servers responding to ping probes in the top 4 U.S. and top 5 Brazilian MNOs. Using the set of LDNS resolvers discovered from the logs of a large content

delivery network, we attempted to probe each resolver from a computer on our university network using both ping and traceroute utilities. Figure 3.2 displays the fraction of resolvers which responded to ping probes for each MNO organized by country.

The figure displays the disparity in opacity between operators. While we can ping 90% of T-Mobile's resolvers, we reach just over 40% of Verizon's resolvers, and less than 1% of both Sprint and AT%T's. The problem is not specific to the U.S., but common worldwide. As another example we show the reachability of the 5 largest Brazilian MNOs. We see a range of resolvers responding to pings: as low as 0% and no higher than 80%. While ping reachability varies across operators, not a single traceroute from the entire set of resolvers in the two countries was completed successfully.

These opacity policies even differ within operators, varying which probes are allowed, and where they can be used. For instance, while T-Mobile allows over 90% of their resolvers to respond to pings, they completely disallows pings to clients and prohibit all traceroutes. These individualized policies complicate operations for CDNs by requiring per operator tuning of their systems.

3.4.2 Public DNS Usage for Cellular Clients

Public DNS services such as Google DNS and OpenDNS provide users fast, reliable and secure DNS resolution service from third party resolvers. With the rise of DNS based censorship [10], these services have also become a popular tool to circumvent common censorship practices. Yet public DNS services affect the quality of selected replicas, since they increase the distance between clients their local resolvers [85].

While public resolver usage has been shown to be increasing problematic in broadband networks [27], in cellular networks it was not believed to be as prevalent due to the difficulty in changing DNS resolvers on mobile handsets. Contrary to this assumption, we find that many MNOs across the globe rely heavily on public DNS resolvers.

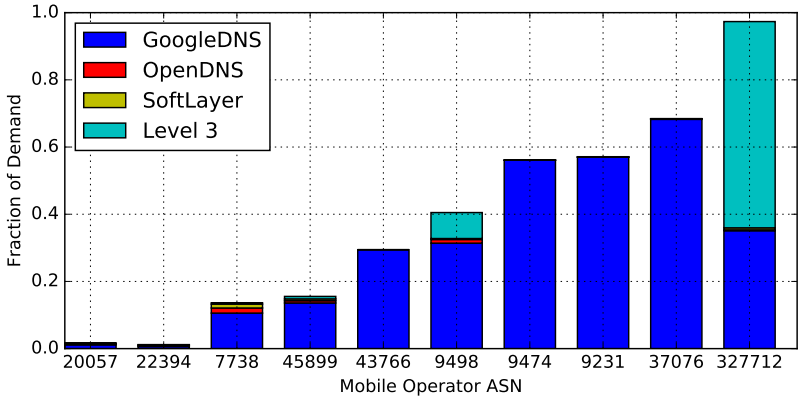
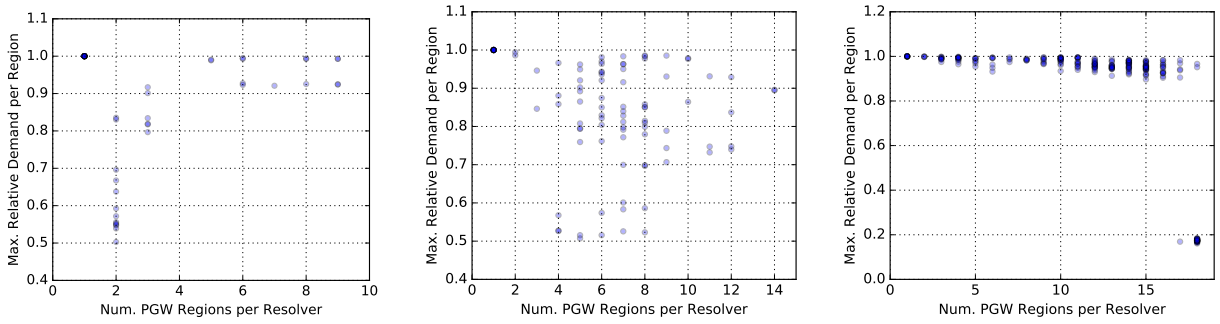


Figure 3.3: Public resolver usage across a subset of global mobile networks. In contrast to current assumptions, many MNOs rely heavily on public DNS services such as Google DNS and Level 3. While the percentage varies widely across operators and countries (e.g. no U.S. MNO has $> 3\%$ public resolver usage), certain MNOs have more than 97% requests coming from public resolvers.

We calculate the percentage of requests coming from public DNS resolvers using request logs from a large CDN. This log data was processed to map client IP addresses with their requesting DNS resolvers, and calculate a normalized fraction of requests from each resolver observed, aggregating by /24 subnet for IPv4 and /48 subnet for IPv6. Using these logs we are able to calculate the relative request demand from cellular clients across different public resolvers.

Our results discovered a wide range of public DNS usage in cellular operators, ranging from less than 2% to upwards of 90% between operators. From this analysis, we selected 11 large MNOs around the globe exemplifying different fractions of public resolver usage, displayed in Figure 3.3. These include AT&T Mobility (US,20057), Verizon Wireless (US,22394), Nextel (BR,7738), Bharti Airtel (IN,9498), and Telecom Algiers (DZ,327712). The figure shows the fraction of client requests sent through the four most popular public DNS services in our dataset: Google DNS [58], OpenDNS [84], SoftLayer [105] and Level 3 [72].



(a) AT&T. (b) Sprint. (c) T-Mobile.

Figure 3.4: Characterization of cellular resolver requests from three large U.S. MNOs. The x-axis represents the number of independent PGW regions observed using each resolver. The y-axis represents the largest fraction of requests coming from a single dominant PGW region. Resolvers located lower and further to the right indicate greater fractions of centralization.

Although public DNS usage in the U.S. is quite low, we found large numbers of global MNOs reliant on public DNS infrastructure. For U.S. operators, both AT&T and Verizon see less than 2% of requests sent through public resolvers. However, for Bharti Airtel in India (ASN 9498), we see public resolver use in nearly 40% of cases. Honk Kong operators SmarTone (ASN 9474) and China Mobile Hong Kong (ASN 9231) both use public resolvers for upwards of 55% of requests. In the extreme case, we see 97% of request demand coming through public DNS resolvers in Telecom Algiers (ASN 327712).

The prevalence of public DNS usage in operators in countries such as India, China and Brazil is possibly due to the larger fraction of non-handset devices connected to cellular networks – either connected directly or tethered through mobile hotspot. The presence of two Hong Kong operators may indicate users wishing to bypass Chinese censorship efforts through these public DNS services.

3.4.3 Centralized Resolver Structure

In addition to public DNS usage, LDNS based redirection effectiveness is also adversely affected by centralized resolver structures of ISPs [85]. Centralized resolver architectures result from ISPs coalescing their distributed resolver infrastructure to a more central location.

These configurations present many of the same hurdles to CDNs as public DNS, by increasing the distance from clients to resolvers [3]. We show that existing cellular networks also utilize centralized resolvers which further hinders the effectiveness of existing replica selection policies.

Due to their strict partitioning [121], the use of centralized resolver architectures in cellular networks manifests as resolvers that are shared by clients from multiple PGWs. This leads to ambiguity when attempting to locate clients based on DNS locations, since these resolvers do not map to a specific PGW.

In order to determine PGW-to-resolver mappings, We utilize ground-truth information from two large mobile operators in the U.S. to map client IP addresses to assigned PGW regions. This ground truth information gives the allocation of client IP addresses to the nearest metropolitan area of the PGW region. For the same time period, we use nameserver logs from a large content delivery network to associate client IP CIDRs to their local DNS resolvers. These logs give the aggregated nameserver demand for each nameserver observed from a requesting client, for each client /24 and /48 subnet for IPv4 and IPv6.

Figure 3.4 displays the mapping of cellular resolvers to PGW regions for three large U.S. operators. We first counted the number of client /24 subnets observed using each resolver, mapping those to PGW regions using our ground-truth information. This is represented on the x-axis. We then looked at the relative demand for each of those PGW regions, taking the largest, and plotting on the y-axis. The figures display the effectiveness of each operator's resolver as a client locator. Resolvers near the top left represent highly localized resolvers since they are mapped dominantly to a single PGW region, while those further towards the lower right corner indicate higher levels of centralization, since they are shared among multiple PGW regions, with less dominant regions.

In AT&T (Fig. 3.4a), we see that while a large fraction of resolvers use a single dominant region, a large fraction appear to evenly share resolving duties across 2-5 PGW locations.

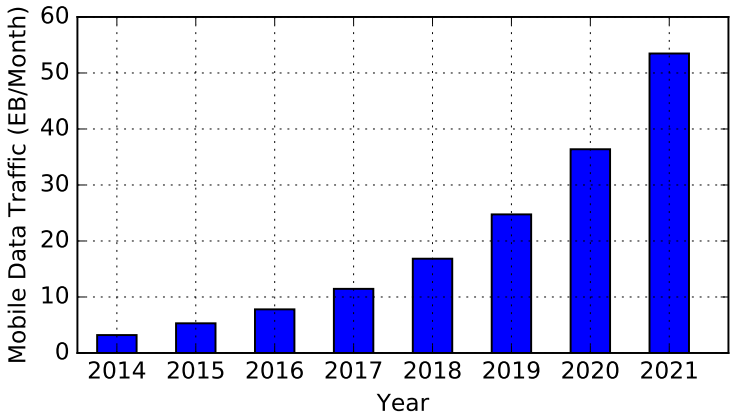


Figure 3.5: Global monthly data traffic and forecast 2014-2021. Monthly mobile data traffic is grow at a 45% CAGR, exceeding 53 Exabytes by 2021 [45]

In T-Mobile (Fig. 3.4c), we see extreme cases where resolvers are shared across nearly all 21 PGW locations. This analysis reveals that even in cases where only local resolvers are used by cellular clients, that these can act as poor indicators of a client's PGW.

3.5 Rise of Mobile Traffic

The inefficiencies in of current replica selection systems are magnified by the explosive growth of mobile traffic over the past several years. The continued growth of mobile subscriptions, coupled with the capacity improvements in cellular radio technology, have vastly increased the traffic demand from mobile devices.

Traffic from mobile devices such as smart phones and tablets has quickly grown over the past decade. In 2015, the number of active mobile subscriptions topped 7.2 billion, meaning there are more mobile subscriptions than the global population [35]. For many, mobile devices are the primary access medium for the Internet. In Vietnam for instance, while only 34% of citizens own a computer, 82% own a mobile device [86]. That same report projects the total number of mobile subscriptions exceed 11.5 billion by 2019.

The latest report from Ericsson predicts a 45% CAGR for mobile data over the next five years [45], illustrated in Figure 3.5. The continued deployment of NGCNs worldwide will

further the spread of bandwidth intensive and latency sensitive applications such as video chat and user generated live streaming. It is already observed that users in LTE networks consume nearly 10 times the traffic as their 3G counterparts [35]. This traffic growth rate is only expected to increase, with the release of the next-generation 5G wireless, which support data rates two orders of magnitude greater that cellular technology today.

This increased demand will require an improved effort by CDNs to load balance cellular traffic in order to not overload individual replica servers. Effective load balancing is achieved through accurate and fine-grained client localization, meaning the problems we have outlined in this chapter caused by coarse and inaccurate information will only continue to hinder CDNs and the clients they serve.

3.6 A Look to the Future

Future improvements to cellular technology will only increase the impact of server selection on overall performance. The next generation of cellular technology, 5G, is expected to support one millisecond access latencies with data rates of 100 Gb/s.

This ubiquitous and high speed connectivity is expected to transform both consumer network services, as well as critical infrastructure. The extremely low latencies are proposed to power real-time interactive interfaces over cellular connections. Examples of this include... In addition critical systems are set to use this high speed data link to connect to cloud services. For instance, many autonomous vehicles systems are designed to implement cloud-based control – a system which *requires* latencies in the single milliseconds.

We posit that the improved radio latencies in the upcoming 5G specification, and critical application which require them, will only increase the need for accurate server selection. Such low latencies are only achieved by geographically proximal servers.

Chapter 4

Approach

In this chapter we introduce our approach for representing cellular clients by their assigned PGW. We begin by making the case for this client proxy, showing that it is fully captures network distances, and greatly reduces the noise inherent to cellular measurements. We describe our methodology for the discovery of and client assignment to PGWs from instrumented mobile vantage points. We next apply this methodology from the data collected from ALICE clients and present initial results of this characterization of global cellular network PGW infrastructure. Last, we apply this principle to content replica selection, selecting servers based on client PGWs. We highlight the potential benefits of this Gateway-Based Replica Selection (GBRS) using data collected from ALICE clients in cooperation with a large CDN in experimental simulations. We find that GBRS provides near optimal server selection for cellular clients, providing equal or improved performance over DNS-based selection in *all* cases.

4.1 A Case for PGW Representation of Cellular Clients

Cellular PGWs make ideal proxies for cellular client location. PGWs play an essential role in cellular networks: among their many responsibilities, they act as ingress/egress points for cellular data networks, manage client billing data, and perform quality of service (QoS) adjustments. In existing cellular networks, all client traffic must traverse the same PGW for the duration of a client's network session, which can last several days in many cases. PGWs

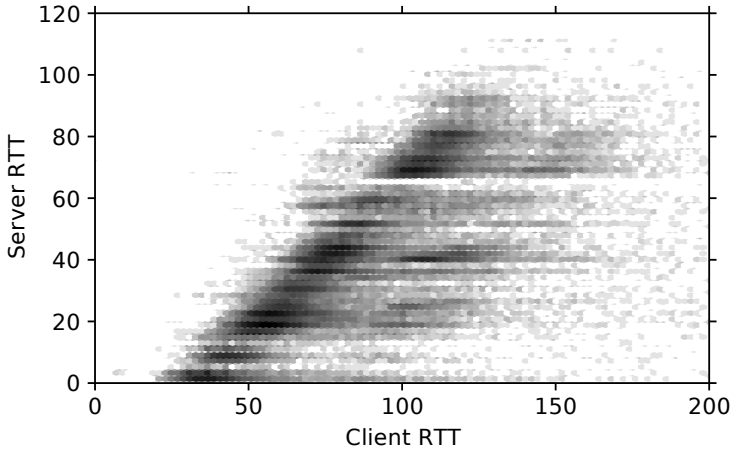


Figure 4.1: Server latency to AT&T client PGWs compared to client end-to-end latency. We see a strong correlation between each latency, denoted by the sharp diagonal boundary formed in the plot. The range of values along the x-axis is due to the larger in cellular radio latency.

are located at the boundary between the internal cellular network and the public Internet, making them the closest point of access for network services to cellular clients. We find that a client's assigned PGW accurately represents network distances to cellular clients, reduce variance in network path measurements, and determines inter and intra-network locality.

PGWs Determine Network Latency. Their presence on all paths to and from cellular clients allow PGWs to be used as proxies for client network location. We validate this assertion through active measurements from mobile clients and content servers. We performed symmetric latency measurements between ALICE clients in the U.S. and a set of approximately 30 content replica severs from a large CDN. While each mobile client measured the latency between itself and each server, the servers simultaneously performed a traceroute to that client's IP address. We use the latency to the last responding hop as that client's PGW latency.

We found server latencies to PGWs correlates strongly to the end-to-end latency between cellular clients and servers. Figure 4.1 plots this relationship between these two values as a heat map. The linear boundary formed on the left side of the point cloud is evidence of this

strong correlation between server latency to clients' gateways and overall end-to-end latency. This shows that distances to client gateways are a reliable predictor of overall end-to-end latency. We utilize this property through our select content replicas based on PGW distance alone.

The figure also shows the different properties of cellular latency, between radio and core network latencies, and Internet path latencies. The wide range of values along the x-axis is evidence of the large variance cellular connections, which do not exist on server's PGW paths. We can use the x-intercept of the boundary line to approximate radio and core network latencies, finding this to be approximately 25 ms for AT&T in the figure. This greater variance leads to problems for popular methods of end-user performance measurements such as Javascript-based measurements [20] or commercial measurement platforms [1]. PGWs, in contrast, currently provide a more predictable marker for performance than these client measurements.

We calculated the path variance of individual server-to-PGW paths, as well as for client end-to-end paths. For each set of latency measurements between an individual server and clients in the same /24 prefix, we measured the standard deviation of each distribution. We find server paths average 5.19 ms of standard deviation, while end-to-end paths average 60.4 ms of standard deviation. Active measurements to PGWs actually improve distance accuracies since they lack the variable cellular core and radio links.

PGWs Determine Network Locality. Client network locality is important to understand from the perspective of network services. CDNs use it to find nearby servers as well as low cost paths to clients. Peer-to-peer services rely on this information to pair nearby hosts for improved overall performance.

Network locality in cellular networks is entirely derived from PGW location. Users which are geographically close to each other may be distant in the network due the locations of

[1] http://www.dynatrace.com , https://www.appneta.com

clients' PGWs. For instance, we discovered that Cricket Wireless clients in Boston, MA were assigned to PGWs in California. A Cricket customer in Boston Skyping between himself and another Boston resident would incur a network round-trip distance of 12,516 miles! While this represents an extreme case, our analysis of client to PGW assignment in Chapter 5 found multiple instances which violate the spatial locality of PGW assignment. Representing clients by PGWs rather than alternative metrics like geographic locality allows for more accurate and better performing mobile peer-to-peer services.

In the following section, we present our methodology for using PGWs to represent cellular client location.

4.2 Gateway Representation of Clients

We posit that representing mobile clients solely by their assigned PGW will greatly simplify mobile network architecture, and acts as optimal network representations of cellular clients. The challenges involved in gateway representation are twofold. The first is the accurately identifying the number and locations of network gateways, and the second is correctly mapping cellular clients to their *assigned* gateway. In this section we introduce our methodology which achieves this dual objective, simultaneously exploring cellular networks and partitioning clients across found PGWs.

4.2.1 Gateway Clustering Methodology

Our approach leverages the stability of IP-to-PGW mappings, and the coalescence of all paths at PGWs. Cellular clients are dynamically assigned addresses from the pools of IP addresses at each PGW. These pools are allocated through carrier-grade NATs (CGNs), typically collocated at PGW instances [92, 113]. We utilize the aggregation of paths at cellular PGWs, since all traffic must traverse a client's assigned PGW. We use these relatively static mappings of IPs to PGWs, along with heuristically detected PGW router addresses to cluster cellular networks into PGW partitions.

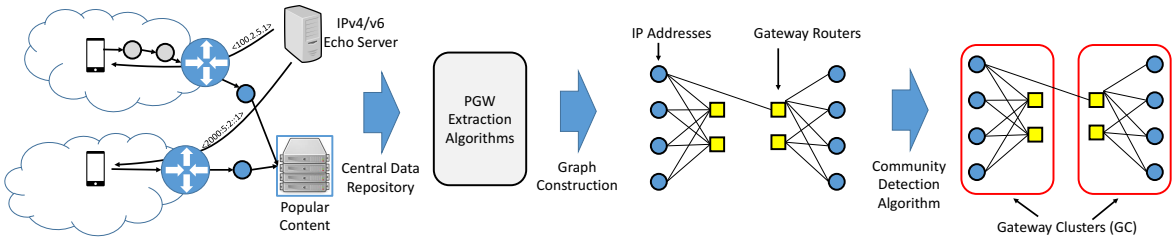


Figure 4.2: Gateway clustering methodology. Cellular client IP addresses are clustered to cellular gateway routers through community detection algorithms.

Our clustering methods utilize active measurements from instrumented mobile devices, capturing the relationship between client IP addresses and detected PGW routers, and partitioning them using graph clustering algorithms. An overview of this process is illustrated in Figure 4.2. We describe our clustering methodology in detail below.

1. **Client Data Collection.** From instrumented mobile clients connected to cellular interfaces, we perform active measurements to collect public IP addresses and traceroute data. Public IP address are recorded from a public IP echo service for IPv4 (and IPv6 addresses when available). Clients perform traceroutes to popular content destinations (both IPv4 and IPv6 when available) from large content delivery networks or content providers. Clients report this tuple of $(IP_4, IP_6, traceroute_4, traceroute_6)$ to a centralized data repository.

2. **Gateway Router Extraction.** We then extract gateway router IP addresses from the traceroutes collected from mobile clients. These router IP addresses are identified using a set of heuristics detailed below.

   Step 1. Determine provider primary ASN. Each provider's ASN is determined by calculating the ASN with the largest fraction of reported client IP addresses for that provider from our entire dataset.

Step 2. Filter measurements by provider ASN. Discard measurements where client IP addresses do not match the provider ASN.

Step 3. From client traceroutes, determine the last hop which is *within* the MNO. This includes private address space, addresses allocated for CGN use (e.g. 100.64.0.0./10), and addresses within each provider's ASN. These final interior hops are marked as that trace's gate router. In cases of "mixed ASes", where an AS encompasses cellular clients, broadband customers and possible transit routers – as is the case with T-Mobile in Germany (AS3320) – we remove the provider ASN from the set of suitable gateway router addresses.

3. **Graph Construction.** Next, we construct a bipartite graph between cellular IP addresses one one side of the graph, and gateway router IP addresses on the other for each cellular provider. Edges connect co-occurrences of cellular IP, and gateway router IP addresses in each client reported tuple. Edges are weighted by the total number of such occurrences in our dataset.

4. **Gateway Clustering.** We cluster these operator graphs into independent gateway regions, consisting of sets of client IP addresses and their associated gateway routers. Unfortunately, due to the noise inherent in mobile measurements from network mobility and standard traceroute noise, simple connected component clustering is unsuitable for this domain. Instead, we use community detection algorithms from Clauset et al. [36] to generate these clusters. These community detection algorithms generate group nodes together by their modularity, or interconnectedness, and automatically select the appropriate number of clusters for each operator.

The algorithm produces for each mobile operator a set of clusters, composed of unique sets of client IP addresses, and gateway router addresses. In many operators, multiple PGWs are collocated within the same facility. Each cluster represents these collocated sets of PGWs.

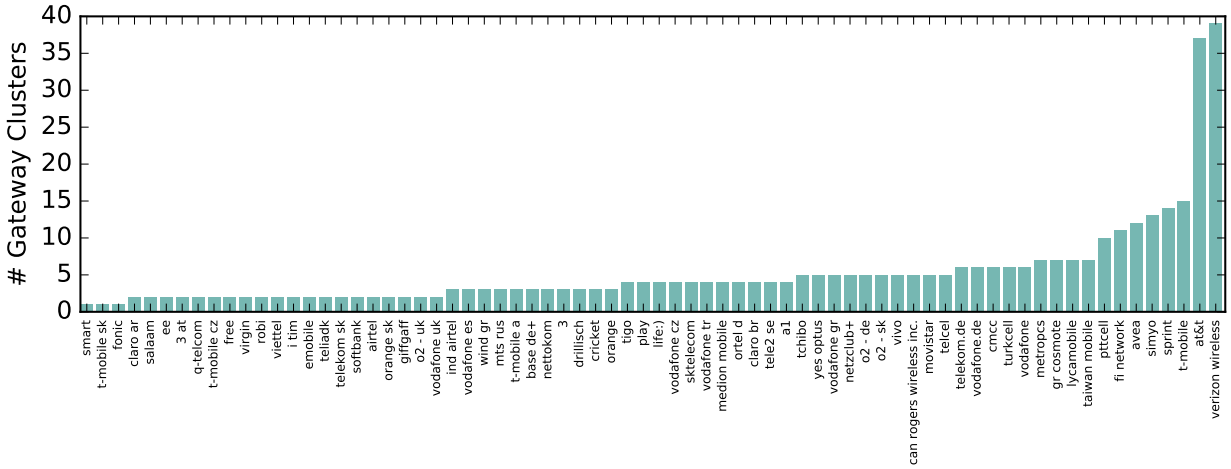
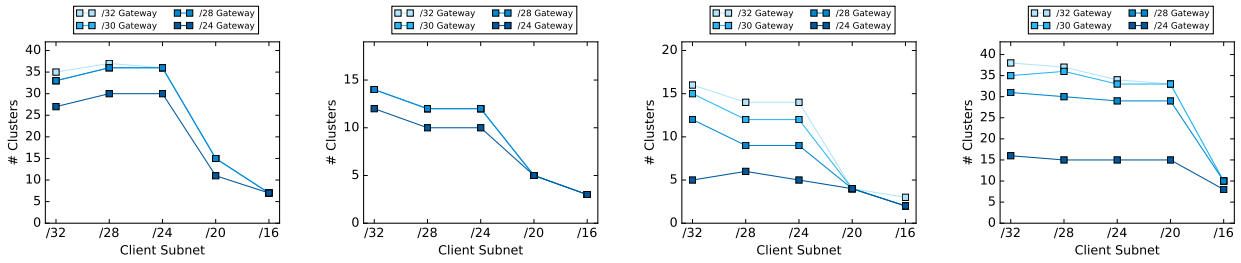


Figure 4.3: Number of detected gateway clusters from our global measurements

We apply our clustering methodology to over two years of mobile data collected from ALICE clients. Our data contains measurements in 94 mobile operators (both MNO and MVNO) around the world, with a bias towards U.S. MNOs. We use this data to perform a preliminary characterization of global MNO infrastructure, looking at the number of PGWs per MNO. Due to the reliance on instrumented mobile clients, and the independence of PGW partitions requiring vantage points in each for detection, our measurements do not represent complete coverage of these networks. Instead these indicate a far lower bound for PGW instances in these operators.

Figure 8.8 displays the detected number of PGW clusters from our methodology for these 94 mobile operators, ordered by detected PGWs. We discovered a total of 394 unique PGWs across all measured operators, yet with a skewed distribution. We found less than five PGWs for over two thirds of the measured MNOs, and ten or less in all but 7 MNOs. The MNOs with the two largest numbers of detected PGWs, AT&T with 37 and Verizon with 39, account for 20% of total detected gateways.



(a) AT&T (b) Sprint (c) T-Mobile (d) Verizon

Figure 4.4: Exploration of clustering parameters for various subnet aggregations for both client IP, and gateway router subnets.

4.2.2 Clustering Sensitivity

We next explore the sensitivity of our clustering across different aggregation levels of both client IP and gateway router subnets. For the four largest U.S. MNOs, we compare our clustering algorithm across this parameter space. Larger aggregate subnets trade potential losses in accuracy for greater network coverage.

We explore this parameter space by performing our clustering algorithm across different subnet values applied to our input measurements. In each iteration, client IP addresses were added to operator graphs according to different subnet masks. Similarly gateway routers were added with the chosen PGW mask. The results are plotted in Figure 4.4, which displays the number of clusters detected for client subnets ranging from /32 to /24, and PGW subnets across that same range.

The figure shows that at least up to /24 subnets, aggregating client prefixes results in little change to detected clusters. It is unlikely that MNOs would allocate IP pools smaller than /24 prefixes, which is reflected in our results.

We find that gateway router subnets can have a large impact on the numbers of detected gateway clusters. We find that the aggregation of gateway routers to subnets larger than /30s pose too large a risk for inaccuracy.

Operator	Client /24	Client /26	Client /28	Client /30
Sprint	0.96	0.94	0.95	**0.99**
T-Mobile	0.76	0.72	0.87	**0.90**
AT&T	0.94	0.94	0.95	**0.95**

Table 4.1: F1 scores for gateway clustering compared to ground-truth allocations for three large U.S. operators. We display the scores across different client subnet sizes used during clustering. We find that smaller client IP subnets result in more accurate clusterings.

4.2.3 Clustering Accuracy

We evaluate the accuracy of our gateway clustering algorithm using ground truth information obtained from two large U.S. MNO: Sprint, T-Mobile, and AT&T. The ground truth we obtained contains IP CIDRs labeled with the location of their currently assigned gateway. Unfortunately, the output of our clustering is unlabeled, consisting only of cluster memberships. For each of our clusters, we assign labels to each entire cluster based on the largest fraction of ground-truth labels for contained IP addresses.

Using these labeled clusterings, we calculate the accuracy of our clustering using the F1 score, a common metric for measuring classifier accuracy. We first calculate the precision, $precision = \frac{tp}{tp+fp}$, and recall, $recall = \frac{tp}{tp+fn}$, for each cluster. The F1 score is calculated as shown below.

$$F1 = 2 * \frac{precision * recall}{precision + recall} \tag{4.1}$$

For each operator, we compared the accuracy of our gateway clusters to these ground truth regions. Table 4.1 displays the F1 scores for our clusters in each of the three operators. The results highlight the high overall accuracy of our clustering method, achieving F1 scores of 0.99, 0.90, and 0.95 for Sprint, T-Mobile and AT&T respectively. These high F1 scores verify that our method is accurate across different and heterogeneous MNOs, and can be considered a general method for clustering all MNOs.

We performed multiple clusterings for these three operators, varying the size of client subnets used during graph creation, choosing client subnets between /24 and /30 in length. Table 4.1 displays the results for each of these efforts. While we had assumed that larger client subnet aggregation would increase cluster accuracy by overcoming the noise of many mobile traceroute measurements, we in fact found the opposite. It appears that measurement noise has the effect of joining disparate clusters at greater aggregation, rather than becoming less significant. While there are more total incorrectly mapped subnets with /30 aggregation, it is much smaller relative to the total overall numbers.

4.3 Gateway-Based Replica Selection

To highlight the benefits of PGW representation for cellular clients, we developed a system for content replica selection which selects servers based on the location of a client's assigned gateway. We call this approach Gateway-Based Replica Selection (GBRS). For the purposes of content distribution and replica selection, GBRS clusters cellular clients into ideal load partitions across IP space, and provides near-optimal network landmarks for each cluster. We evaluate the potential benefits of GBRS with experimental simulations using ALICE clients in cooperation with a large CDN. Our results show that GBRS improves upon existing methods in nearly all cases, and can decrease latency to selected replicas by over 60% in certain operators.

In GBRS, we first cluster cellular clients by their assigned PGWs using the methods described in the prior section. In addition to the list of IP CIDRs contained within them, each cluster contains an IP address which is representative of each cluster's PGW location. This IP address is used as a network landmark for active measurement (e.g. ping and traceroute) by CDNs. All cellular clients GBRS are therefore represented by their cluster landmark.

GBRS holds many advantages over existing replica selection systems such as DNS-based redirection. GBRS clustering is ideal for existing cellular network architecture, which divides clients among a number of isolated partitions These partitions are bounded by PGWs, meaning that all inbound and outbound paths of clients in the same partition must traverse the same PGW, regardless of server or client location. PGWs also represent the closest point of access to cellular clients; one can do no better than selecting a content replica adjacent to a client's gateway.

In cooperation with a large CDN, we show the effectiveness of GBRS using measurements from live mobile clients distributed across the U.S., and compare its performance to existing DNS-based approaches. We performed active measurement experiments from ALICE clients within the four largest U.S. MNOs. With data obtained from the CDN, we preselected a set of approximately 30 replica servers at geographically diverse locations throughout the U.S. or each of the four MNOs. In each case, preference was given to servers located within operator networks or at operator points-of-presence (PoPs). ALICE clients measured the network latency to each of these replica servers approximately every hour. We additionally performed a DNS resolution to a popular website hosted by this CDN, and measured its network latency to act as our baseline for existing state-of-the-art systems. During the same time period, we directed all the chosen replica servers to measure the latency between themselves and each PGW cluster's landmark IP address.

Using these set of experiments we simulated GBRS replica assignment and compare its performance to deployed systems. GBRS assignments were chosen based on the lowest average latency between replica servers and cluster landmark IP addresses. We compare the latency of GBRS selected replicas to those selected by DNS-based approaches. Figure 4.5 plots the performance of GBRS against existing systems for clients in AT&T, Verizon Wireless and T-Mobile.

The left side of each figure plots the performance of existing CDN assignments, "Chosen Replica", GBRS assignments, "Tiller", and the minimum RTT measured to any server during each experiment "Optimal". The figure highlights the benefits of GBRS, showing improved performance to existing systems in nearly all cases. Servers selected by existing DNS-based approaches are consistently 5-10 ms more distant than GBRS selections. These seemingly small latency differences between replica choices are accumulated throughout the hundreds of round trip times needed for popular web pages.

Similarly, the right side of the figures plots the relative performance of server selection, plotting the percent difference in latency between selected servers for each method and the "Optimal" seen. GBRS provides optimal mapping in nearly 80% of cases, compared to the 40% of those achieved by DNS-based selection. In the top 20th percentile of cases, GBRS selected replica servers which were upwards of 60% closer.

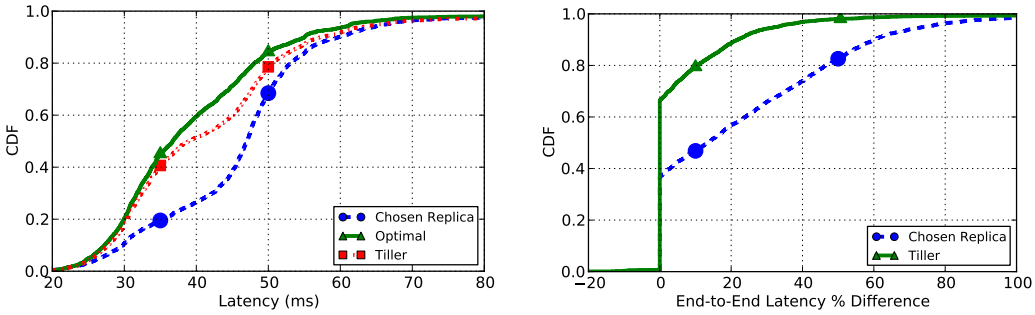
4.4 Summary and Contributions

In this chapter we made the following contributions:

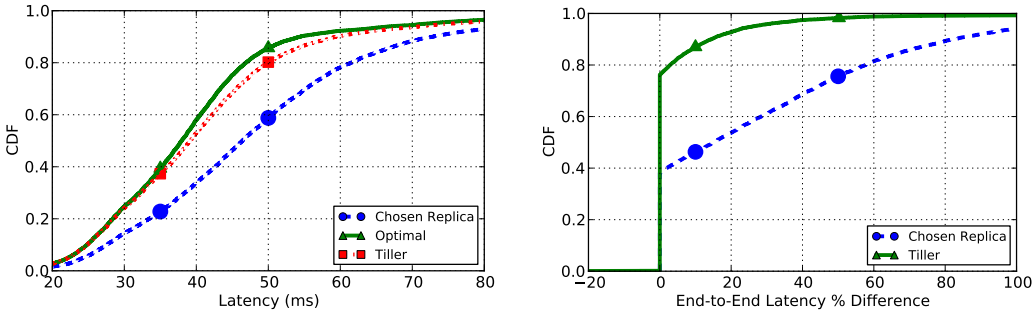
- we introduced our approach for representing cellular clients by their assigned PGW.
- We presented our case for this client proxy, showing that client PGWs are reliable predictors of end-to-end latency, and this representation greatly simplifies locality for network services.
- We describe our methodology for the discovery of and client assignment to PGWs from instrumented mobile vantage points. We next apply this methodology from the data collected from ALICE clients and present initial results of this characterization of global cellular network PGW infrastructure.
- We apply this principle to content replica selection, selecting servers based on client PGWs. We highlight the potential benefits of this Gateway-Based Replica Selection

(GBRS) using data collected from ALICE clients in cooperation with a large CDN in experimental simulations. We find that GBRS provides near optimal server selection for cellular clients, providing equal or improved performance over DNS-based selection in *all* cases.

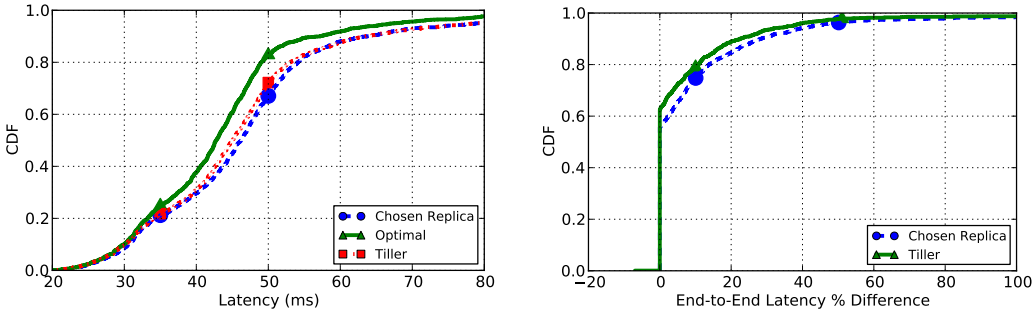
We implement this approach through TILLER , our system for cellular network exploration from mobile vantage points. We detail the design and implementation of TILLER in Chapter 7.



(a) AT&T.



(b) Verizon.



(c) T-Mobile.

Figure 4.5: Absolute and relative network latency to Akamai replica servers for the three largest U.S. MNOs. The figure plots the given replica performance against the measured optimal replica, and the mapping provided by GBRS. GBRS provides equal or better performance in nearly all cases, and a 60% improvement in latency in 20% of cases.

Chapter 5

Cellular Infrastructure Characterization

5.1 Overview

This chapter presents a characterization of existing cellular infrastructure, with a particular focus on key structural components – DNS and packet gateways – since they have the largest impact to CDNs. We characterized existing cellular DNS infrastructure with experiments from ALICE clients. Our findings reveal cellular DNS to be a poor proxy for client location, resulting in suboptimal selected replica servers. Cellular operator's use of (*i*) indirect resolution methods, (*ii*) inconsistent client-to-resolver assignment, and (*iii*) opaque resolver infrastructure, all challenge existing DNS-based replica selection systems.

We argue that instead, packet gateways should be used as location proxies for cellular clients. Since they route all traffic to and from assigned cellular clients, PGWs represent the closest point of access between clients and the greater Internet. We present our techniques for identifying and locating cellular PGWs, and determining a client's assigned PGW based on their public IP address. We present our characterization of PGW instances for the four largest U.S. MNOs showing the continued expansion of cellular infrastructure. We motivate our approach through our longitudinal results, which highlight the loose locality of client assignment to packet gateways, for which our approach is the only accurate way to determine optimal performance.

The contributions of this chapter are summarized as follows:

- We present results from our exploration of cellular LDNS infrastructure (§ 5.2). We find that all investigated operators utilize indirect resolution methods which challenge existing DNS-based replica selection. We show client-resolver mappings to be inconsistent among cellular clients, and highlight its impact on replica mappings. We investigate the resolution performance and distance of public DNS services to cellular clients, longer resolution times than operator resolvers, but with much shorter tail performance.

- We present results from our investigation into PGW locations for the four largest U.S. MNOs (§ 5.3). We introduce our techniques of mapping clients to these instances based on their public IP address, and show the heterogeneity in PGW subnet allocation across operators.

- We present our longitudinal results from ALICE clients looking at the dynamics of network assignment over time (§ 5.4). We find that certain operators employ weak locality between clients and PGWs.

5.2 Cellular DNS

The domain name service (DNS) is a critical infrastructure service for nearly all networked activity on mobile devices, translating human-readable domain names into network locatable IP addresses. DNS is also the most commonly relied upon method for client localization in content replica selection [75]. An understanding of cellular DNS services, and its performance for both hostname resolution as well as client localization for replica selection, are important for evaluating the effectiveness of existing content delivery systems.

Our investigation into cellular DNS revealed several differences from assumed DNS behaviors, each of which impact the effectiveness as a content replica landmark. We found that operators utilize resolution methods which increase the distance between clients and their local DNS resolvers, and that these client-to-resolver mappings are inconsistent over

Provider	Client	External	Pairs	Consistency %
Sprint	19	22	31	64.1
Verizon	27	27	27	100
T-Mobile	3	32	32	7.3
AT&T	5	43	43	12
SK Telecom	2	24	24	12
LG U+	5	80	80	6.2

Table 5.1: Number of LDNS Pairs seen by our mobile clients. Network structure and configuration varies by network in both the number of client facing and external facing resolvers, as well as the consistency of their pairings.

time. We investigate the use of public DNS in cellular networks, finding both its resolution performance, and distance to clients inferior to client's local DNS options.

5.2.1 Operator DNS Characterization

In this section we present the results of our characterization of the DNS infrastructure of four large U.S., and two large South Korean MNOs. To determine the resolver structure, we analyze both the DNS server configured on the client, which we call the Client Resolver, and the DNS server seen by a authoritative DNS (ADNS) run by our research group, which we call the External Resolver. ALICE clients periodically resolved hostnames through our ADNS to obtain their current external resolver, and performed latency measurements to both client and external resolvers.

We recorded the grouping of observed client and external resolvers to understand the configuration and dynamics of cellular infrastructure and their DNS resolvers. We refer to each grouping as an *LDNS Pair*. We calculate the *consistency* of these resolver pairings as the percentage of our measurements in which the client and external resolvers are paired. The consistency of pairings captures the stability of mappings between clients, their locally configured resolver, and the external facing resolver. For example, a client resolver equally load balanced between two external resolvers would have a consistency of 50%. A summary of each operator's DNS infrastructure is given in Table 5.1.

Within each operator, we detected the use of *indirect resolution* in cases where the client resolver and external resolver are different. An indirect resolver structure typically indicates greater distances between a client and the resolver seen by the CDN [99]. We detected indirect resolution techniques in all of the operators investigated. These take the form of (*i*) anycast resolvers, (*ii*) LDNS pools, and (*iii*) a tiered resolver hierarchy.

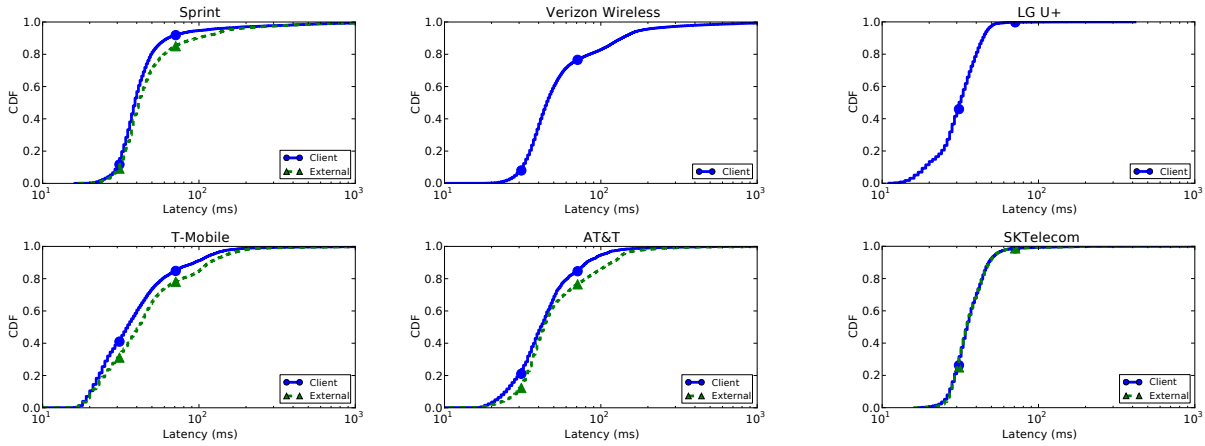
With anycast resolvers, all clients are assigned the same DNS resolver IP address regardless of their location. DNS queries are directed toward nearby DNS resolvers within the cellular network through anycast routing.

We found the use of anycast DNS within AT&T's and T-Mobile's networks. Both carriers showed a limited number of configured DNS resolver addresses on client devices with a significantly larger number of publicly visible addresses indicating the use of IP anycast for resolvers. For example, a single AT&T address (172.26.38.1) in our measurements shows mapping to 40 external resolver addresses.

LDNS pools, as previously described by Alzoubi et al. [8], consist of a collection of servers which load balance DNS requests within themselves. Unlike Alzoubi et al., who detected the presence of LDNS pools by seeing different resolvers for consecutive queries responding to a CNAME entry, we were able to identify LDNS pools by directly comparing the configured resolver on the mobile device with the IP address seen by our ADNS.

We discovered the presence of LDNS pools within both South Korean operators. In each of these cases, all resolvers are public IP addresses, and all have pairs in which a client facing resolver is observed paired with multiple external resolver addresses. For SK Telecom and LG U+, we observed 2 and 5 client configured LDNS resolver addresses and 24 and 89 publicly visible addresses, respectively. For these carriers, each client and external pair are contained within the same /24 prefix.

Tiered DNS servers exist as two separate public IP addresses, yet with one client resolver and one external facing resolver. These paired resolvers also differ in latency and traceroute



(a) US Carriers (b) SK Carriers

Figure 5.1: Client latency to internal and external resolver locations. Ping latencies in Sprint, T-Mobile and AT&T reveal resolvers which are located in separate locations, with external resolvers located further away from clients. Although no external resolvers in either Verizon's or LG U+'s networks responded to probes, client and external resolvers exist in separate ASes in the case of Verizon.

hops from client probes. Tiered resolvers may indicate a hierarchy of DNS resolvers within that operator's network, or a centralized resolver structure, however, we are only able to observe the end points from our experiments.

We observed the tiered resolvers in Sprint's and Verizon's networks. In the case of Sprint, each resolver maintains a fairly consistent mapping between client and external resolvers, with consistent pairs 64% of the time. Verizon was the only cellular operator which maintained a 100% consistency between client and external facing resolvers. While both resolver locations were public IP addresses, we were unable to measure the distance between these resolver pairs due to unresponsive probes to external resolvers. However, each LDNS pair within Verizon exists in different ASes: 22394 for client facing resolvers and 6167 for external facing resolvers.

5.2.2 Cellular Resolver Distance

An important aspect of DNS in cellular networks is the network distance between clients and their corresponding resolver infrastructure. As shown in the previous section, cellular DNS often consists of multiple, hierarchical resolvers. Our vantage point at the mobile client, as well as the ADNS, allows us to capture characteristics of both the client resolver configured on each device, as well as the external resolver visible from our ADNS.

We measure the latency between these two resolvers by directing clients to ping both sets of resolvers during each experiment. Distance to client facing resolvers is important for resolution performance, while distance to external facing resolvers has implications on content replica selection [85]. Figure 5.1 plots the cumulative distribution of latencies to clients' configured client facing resolver and external facing resolvers.

We find evidence of hierarchical resolver infrastructure in three of the four U.S. operators. Resolvers in T-Mobile, Sprint and AT&T showed signs of distance between resolvers, with external resolvers up to 20 ms further away in many instances. We see cases where both resolvers have nearly equal latencies indicating either identical machines or collocated resolvers, as is the case with SK Telecom. External resolvers for both Verizon and LG U+ failed to respond to client measurements, preventing their characterization. While we were unable to determine structural properties from latency measurements in Verizon, we nonetheless infer a DNS hierarchy within Verizon since client and external resolvers reside in separate ASes.

5.2.3 Cellular DNS Performance

We now look at the resolution performance of each mobile client's DNS provided from their cellular operator. We find DNS performance under LTE to be relatively consistent and comparable to DNS performance on wired broadband.

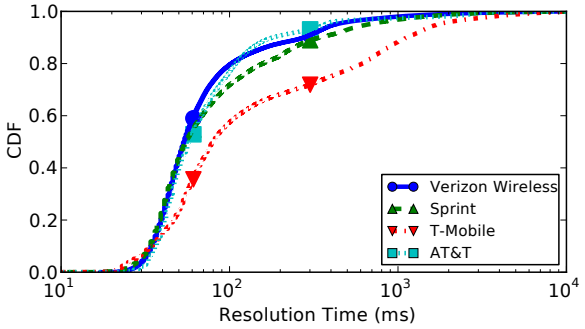


Figure 5.2: DNS resolution time for US carriers measured from client devices for the 4 major US cellular providers.

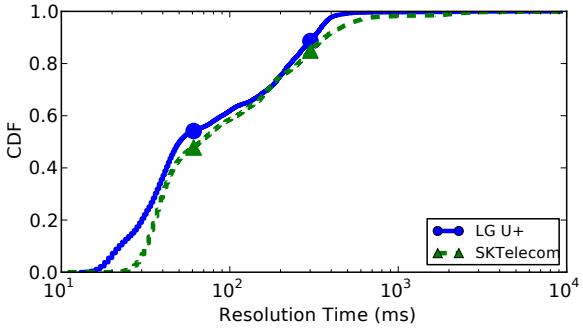


Figure 5.3: DNS resolution time for South Korean carriers measured from client devices for 2 major cellular providers.

Figures 5.2 and 5.3 present, respectively, CDFs of resolution time for each of the four US carriers, and for the two South Korean carriers we studied. The figures show resolution performance times between 30 and 50 ms for carriers in both markets. These numbers are comparable to resolution times within the wired Internet [3] for the lower 50th percentile.

Both South Korean carriers and T-Mobile exhibit bimodal behavior above their 50th percentile, and the remaining operators show a long tail of resolution times above the 80th percentile. To determine measure the impact of resolver cache on resolution time tails, we conducted back to back queries, measuring the difference between the first and second DNS queries. The results, presented in Figure 5.4, show cache misses accounting for additional delays approximately 20% of the time, similar to the bimodal behavior seen in Figures 5.2 and 5.3.

5.2.4 Cellular Resolver Opacity

Unlike related studies characterizing the behavior and structure of wired networks DNS resolvers, measurement analysis of cellular DNS resolvers can only be carried from clients within their networks. This is because most cellular operators employ NAT and firewall policies that prohibit externally generated traffic from their network [113].

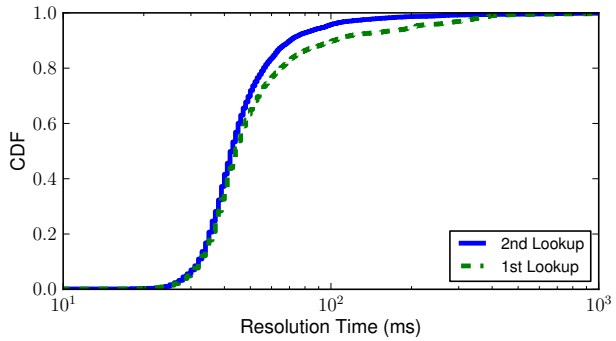


Figure 5.4: Cache performance for clients local DNS resolvers combined for each of the four US carriers. Although the hostnames we looked up were very popular, we see DNS cache misses for nearly 20% of DNS requests on cellular. This is a product of the short TTLs used by CDNs, and explains the bimodal distribution, and long tails of resolution times seen in Figure 5.2.

Provider	**Total**	**Ping**	**Traceroute**
Sprint	20	0	0
Verizon	34	32	0
AT&T	47	3	0
T-Mobile	40	40	0
SKTelecom	24	0	0
LG U+	80	0	0

Table 5.2: Number of external DNS resolvers able to be reached externally by either ping or traceroute probes.

We tested the external reachability of cellular network DNS resolvers by launching ping and traceroute probes from our university network to the observed external resolvers (Sec. 5.2.1). Table 5.2 presents a summary of our results. Of the six major cellular carriers we profiled, only Verizon and T-Mobile resolvers responded to a majority of ping requests. In the case of Sprint and the two South Korean carriers we studied, no resolvers responded to any of our `ping` probes. No traceroutes were successfully completed to *any* of the resolvers we investigated.

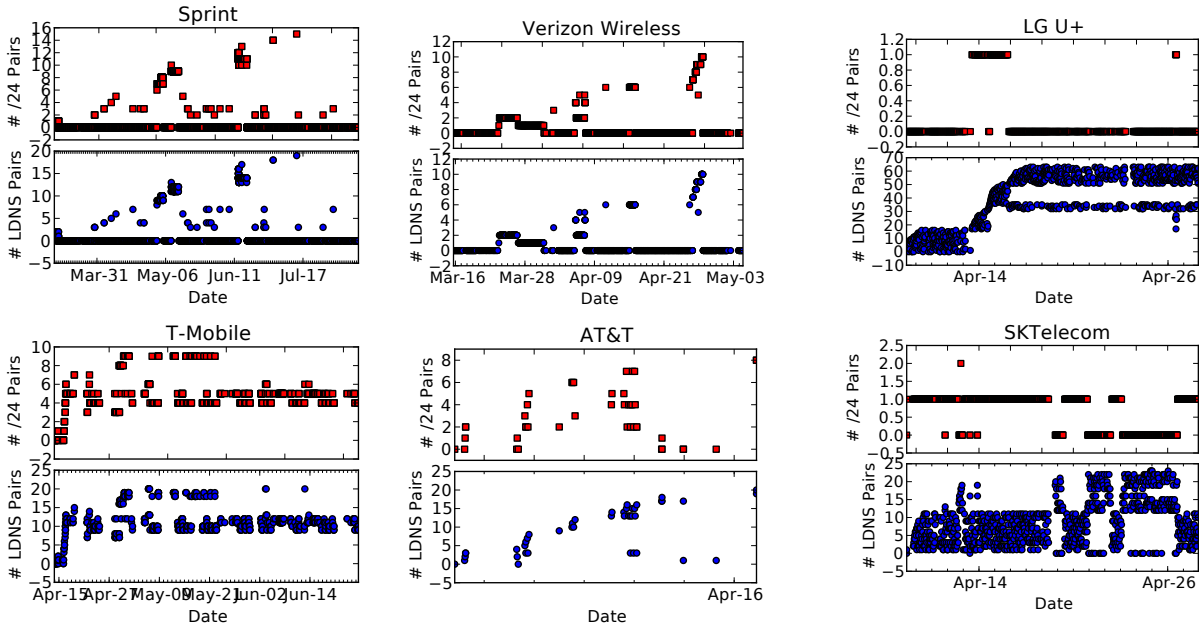
In contrast, all the probes launched by our mobile clients were able to measure some if not all of the DNS infrastructure of these carriers. Clearly the known opaqueness of cellular networks extends to the cellular DNS infrastructure and, thus, any analysis of such infrastructure requires the participation of devices within each cellular network.

5.2.5 Client resolver inconsistency

In this section we analyze the consistency LDNS resolver assignment for clients in each cellular provider. The consistency (or stickiness) of a device's LDNS resolver can significantly impact the effectiveness of CDNs, due to the reliance of a client's LDNS resolver as a client locator. We recorded the occurrence of LDNS Pairs, consisting of pairings of (client resolver, external resolver), seen by clients over time. Our analysis revealed inconsistency between a client-to-resolver mappings in all cellular providers investigated.

Figure 5.5 shows the LDNS pairs observed by each device over time, enumerated based on the order of appearance in our measurements. The figure displays that the type of inconsistency of client-to-resolver mappings varies between operators. The consistency of mapping for Sprint and Verizon clients, for instance, show relatively stable mappings while the mappings for the remaining carriers appeared to be very unstable.

Unstable mappings are not all the same, however, as can be seen when contrasting T-Mobile, AT&T and the two South Korean carriers. In the first two, changes in resolver



(a) US Carriers

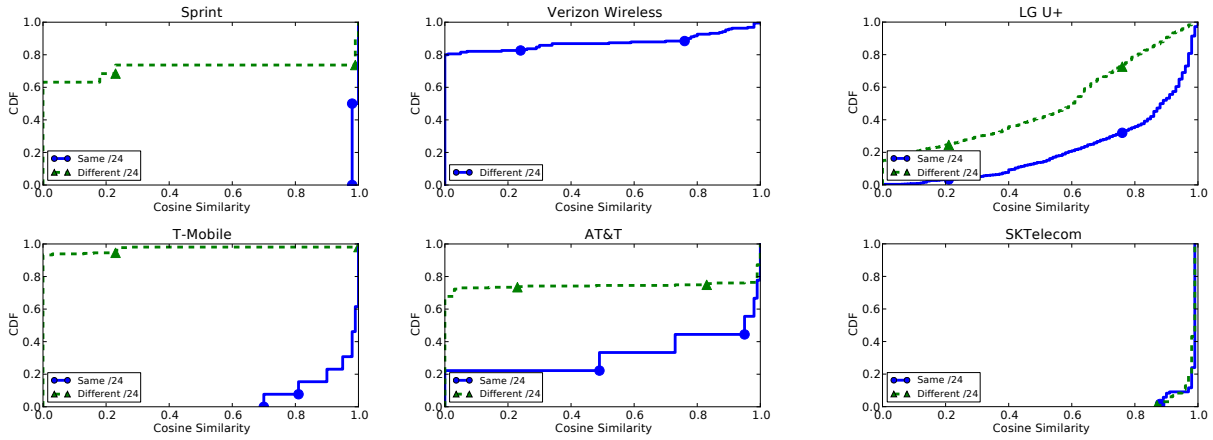
(b) SK Carriers

Figure 5.5: Number of external resolvers observed by a client in each of the networks we looked at. Bottom: number of external resolver IP addresses. Top: number of unique /24 prefixes observed by resolvers. Client DNS resolvers change not just within localized clusters, but span multiple /24 prefixes over time.

IP addresses are typically accompanied by changes in the resolvers /24 prefix. In contrast, while clients in the two South Korean carriers experience more frequent changes in resolver IP addresses, the alternative resolvers are contained within one or two /24 prefixes. For example, a client within LG U+'s network witnessed over 65 external resolver IP addresses within a two week period, all of which were within only 2 /24 prefixes.

5.2.6 Impact on CDN Replica Selection

We now explore the impact of inconsistent resolver mappings on replica server selection. For each mobile website in our study, we look at the number of replica IP addresses returned, and how often each replica is chosen, for each resolver /24 subnet in each carrier. For the selected DNS resolver, we construct a map of $< replicaIP, ratio >$ pairs capturing, for each replica server, the server IP address and the fraction of time that replica was used:



(a) US Carriers (b) SK Carriers

Figure 5.6: Cosine similarity of replica servers for buzzfeed.com between resolvers within the same /24 prefix, and those in separate prefixes. Resolvers within the same /24 prefix see very similar sets of replicas (cosine similarity values close to one), and those in separate prefixes see high set independence (values close to zero). Clients changing resolver /24 prefixes are directed towards completely different sets of replica servers.

$$replica\_map =< (ip_1, \frac{ip_1 seen}{total\_seen}), ..., (ip_n, \frac{ip_n seen}{total\_seen}) >$$

We then us *cosine similarity* [29,109] to quantify the similarity of replica servers mapped to each DNS resolver. The cosine similarity between two vectors A and B quantifies the degree of overlap between two vectors by computing the dot product of the vectors and dividing by the product of their lengths:

$$cos\_sim = \frac{A \cdot B}{\|A\|\|B\|}$$

Given our vectors of non-negative probabilities, cos_sim ranges from 0 to 1. When $cos\_sim = 0$, the sets of redirections have no clusters in common. Values greater than 0 indicate that some clusters are seen in both sets; $cos\_sim = 1$ means that the sets of clusters seen are equivalent.

We highlight the significance of resolver inconsistency by measuring the similarity between replicas sets assigned to resolvers. We computed the cosine similarity of replica vectors for every resolver in a carrier's network.

Figure 5.6 shows the cosine similarity (overlap) between replica sets for DNS resolvers in the same /24 prefix, and those in different prefixes. We see large degrees of independence between sets from differing /24 prefixes, with over 60% of sets having a cosine similarity of 0, meaning there is no overlap at all between replica vectors. This high degree of replica set independence becomes a significant issue since, as we showed in Section 5.2.5, cellular clients change LDNS resolvers frequently and across /24 prefixes potentially leading to large performance variability.

The relatively small numbers of replica servers mapped to each cellular DNS resolver, particularly when compared to the diversity of typical CDN-resolver mappings in wired networks [109], may be a product of cellular network opacity (Sec. 5.2.4). CDNs typically aggregate client resolvers behind traceroute divergence points and map clients based on active measurements to these points [74]. The opacity of cellular networks, which restricts the reach of traceroutes, calls into question the effectiveness of this approach. Looking at the replica maps for each cellular operator and comparing cosine similarities, it appears that CDNs are strictly mapping replicas to resolver /24 prefix. The inconsistency of resolver mappings means clients are sent to entirely different replicas for each resolver assigned.

5.2.7 Public DNS in Mobile Networks

In this section we investigate the resolution performance of public DNS services such as GoogleDNS and OpenDNS, as well their impact on CDN replica selection. Despite the fact that some cellular operators prohibit customers from configuring different DNS resolvers,[1]

[1]Mobile devices must be "rooted" in order to change these settings; and, while no longer *illegal* in the United States, rooting voids the device's warranty in most cases.

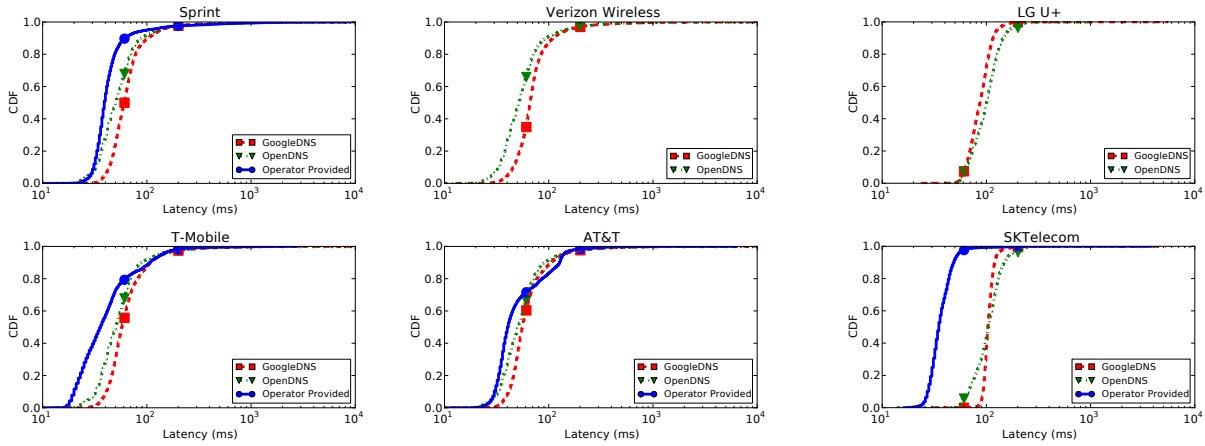
Provider	Local	GoogleDNS	OpenDNS
Sprint (all IPs)	24	122	38
Verizon (all IPs)	37	135	41
T-Mobile (all IPs)	38	151	49
AT&T (all IPs)	47	160	38
SKTelecom	25	33	7
LG U+	80	47	6
Sprint (/24)	16	21	9
Verizon (/24)	37	13	7
T-Mobile (/24)	21	15	8
AT&T (/24)	27	15	6
SKTelecom (/24)	4	5	2
LG U+ (/24)	3	6	1

Table 5.3: Total number of DNS resolver IP addresses seen from our ADNS for each provider and resolver group. While public resolvers have more total IP addresses, most are located within the same /24 block. In addition we see more /24 blocks for local resolvers than public ones, with the exception of Sprint.

our characterization of public DNS services provides a valuable benchmark against which to compare the performance and localization effectiveness of different cellular operators DNS.

We characterize the number of resolvers seen from clients within each mobile provider. Table 5.3 shows the number of unique resolver IPs seen from our clients on each mobile provider. We see that the anycast public DNS resolvers give significantly higher numbers of unique IP addresses to clients (GoogleDNS has over 4 times the IP addresses than the cellular DNS providers for US carriers). This is partly due to the architecture of these public DNS resolvers. For instance, according to their public documentation, GoogleDNS consists of 30 geographically distributed /24 subnetworks [58].

Accounting for the clustered nature of these public DNS resolvers, the bottom half of Table 5.3 shows the number of unique /24 subnetworks seen for each resolver. By aggregating by /24 subnetworks, we see relatively equal numbers for all three resolver types across each cellular provider, especially when compared to the large disparity in IP addresses shown above.



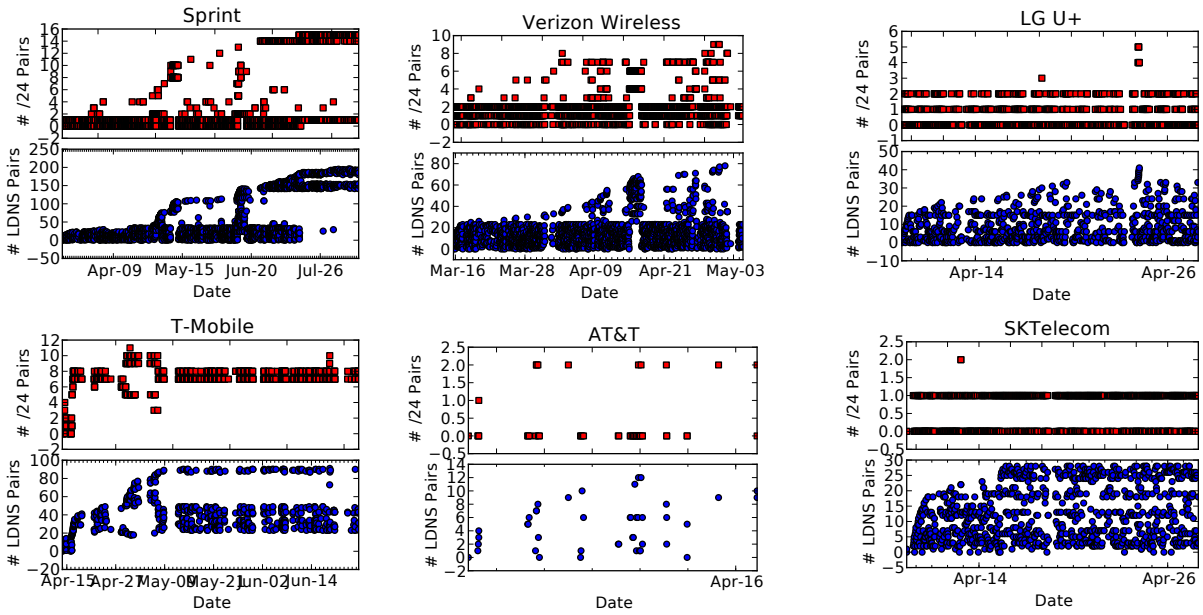
(a) US Carriers (b) SK Carriers

Figure 5.7: Client latency to public DNS resolvers, GoogleDNS and OpenDNS, compared to their local operator provided DNS resolvers.

As in wired networks, the increased distance to public DNS resolvers could significantly impact the web performance experienced by clients [85]. Using the methods described in Section 5.2.1, we measure the distance to both public DNS resolvers, and compare it with that of the cellular operator provided DNS. Figure 5.7 present CDFs of these measured latencies for the different carriers in ours study.

The figure shows that for the carriers whose resolvers responded to our probes, the cell DNS is commonly closer to clients than the public resolver. This is not surprising since all public DNS resolvers are outside of cellular networks, and resolution requests would have to leave the cellular network to complete. For the US carriers, the cell DNS resolvers is, at median, 10-25 ms closer than the best public DNS resolver. For South Korean operators, public DNS resolvers taken nearly twice for resolution as long at the median. On the other hand, the figure also shows the greater performance consistency of public DNS services and a long tail of resolution times from cell DNS. In the case of T-Mobile, public DNS resolvers performed equal or better over 40% of the time.

In addition to analyzing the observed structure of public DNS and its relative performance, we also explore the consistency of client-resolver mappings. We focus on Google

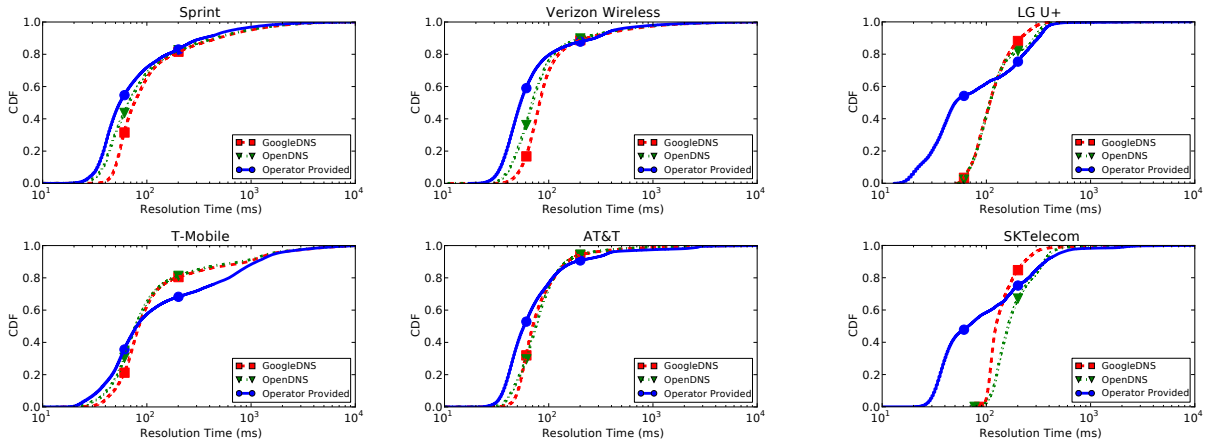


(a) US Carriers (b) SK Carriers

Figure 5.8: Resolver consistency for GoogleDNS for users in each carrier. It is interesting to note that even though GoogleDNS's IP address (8.8.8.8) is anycast, users see large variability in the /24s they are sent to. Each /24 for GoogleDNS represents one of thirty distinct geographic locations for their services.

Public DNS, comparing the external resolver IP addresses assigned to a single client over time, in each of the carriers in our study. Figure 5.8 presents this both for DNS resolvers and their /24 prefix. As the figure shows, despite relying on anycast, Google users are directed toward multiple /24 blocks of resolvers at different geographic locations, given that each /24 block represents one of the 30 geographically distinct resolver clusters. This inconsistency could be the results of the widespread use of tunneling (e.g., via MPLS).

Figure 5.9 shows domain resolution times for the device's locally configured DNS along with public DNS resolvers GoogleDNS and OpenDNS. Our results show that in a majority of cases, the locally configured resolver provides faster domain name resolutions. While the name resolution times are greater on average for public DNS resolvers, they exhibit lower variance in response times and have a shorter tail than all cellular operators we investigated.



(a) US Carriers (b) SK Carriers

Figure 5.9: Domain resolution times for the cellular operator's provided DNS compared with public DNS resolvers GoogleDNS and OpenDNS. Cellular operator DNS offers lower resolution times when compared to public DNS services.

In general, our results are consistent with those previously reported in [3, 85], where public DNS resolvers were located further away from clients, and therefore incurred longer domain name resolution times due to the larger round trip times to the resolvers themselves.

5.3 Cellular Gateways

Cellular packet gateways (PGWs) are critical infrastructure components for cellular networks and their clients. Since they route all traffic to and from assigned cellular clients, PGWs represent the closest point of access between clients and the greater Internet.

We argue that PGWs should be used as location proxies for cellular clients in network services. Understanding the locations of cellular clients is useful for (*i*) content replica selection, (*ii*) performance debugging of cellular clients, (*iii*) peer-to-peer locality. As previously shown in Section 3.4, common heuristics like a client's LDNS server are less effective locators in cellular networks, and often are shared by clients in multiple PGWs.

Utilizing PGWs for network services faces two main challenges: (*i*) identifying PGW locations in cellular infrastructure and (*ii*) determining PGW assignments of cellular clients. We develop a clustering technique which solves each of these challenges, improving the

accuracy of identification and geo-location, while simultaneously mapping clients to these PGWs.

We present our techniques for identifying and locating cellular PGWs. We posit that cellular IP addresses can be used to accurately identify PGWs, and present a novel clustering method which groups client, and PGW IP addresses at collocated facilities. We show that this clustering improves accuracy and reduces noise in mobile measurements, while simultaneously discovering IP assignments to each PGW.

Using the results from this clustering, we present our characterization of PGW instances for the four largest U.S. MNOs. We find large differences PGW configurations and allocation patterns between operators, both in the number and size of subnets, as well as spatial proximity and locality of user assignment.

5.3.1 Identifying Cellular Gateways

Discovering PGW locations is challenged by the opacity of cellular networks, and the growing cellular infrastructure. We expand upon prior work to detected PGW addresses, from work using the first public IP hop of client traceroutes [98, 117], to methods tuned for specific operators [120]. We extend these simple heuristics, finding a general method to account for the large variety of cellular network topologies and policies.

We identified PGW IP addresses using outbound traceroutes from ALICE clients, using heuristics to identify PGW locations. We directed instrumented clients to perform traceroutes towards major content delivery networks such as Akamai and Edgecast, as well as large content providers such as Google and Facebook. This dataset contains only IPv4 traceroutes since Android did not have native IPv6 traceroute capability until version 4.4. However, client PGW assignment remain the same for both IPv4 and IPv6 networks.

For outbound traceroute, we look for changes in ASes to denote PGW instances. Specifically we look for changes in IP addresses from the first responding hop, either with

Operator	**Num. PGW IPs**	**Num. Client /24s**
AT&T	99	42
Sprint	89	36
T-Mobile	199	70
Verizon Wireless	162	625

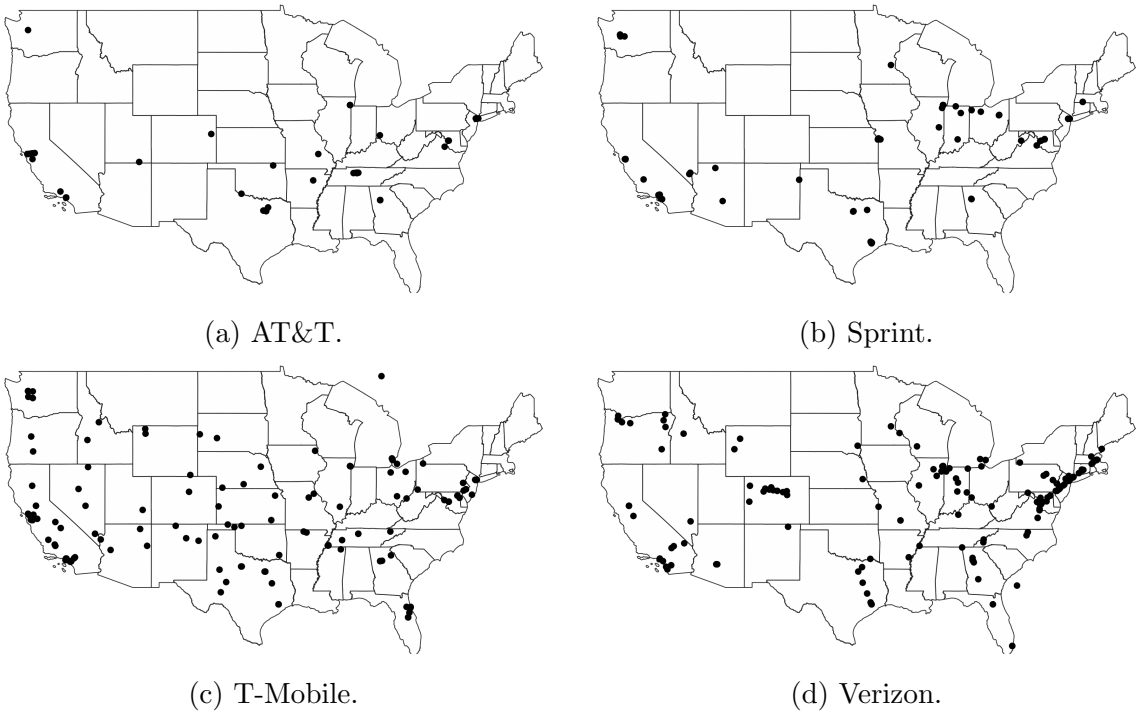
Table 5.4: Results from our initial exploration of cellular PGWs.

differences in the ASN, changes between private and public addresses space, or changes within private address space (e.g. shifting from 10.0.0.0/8 to 172.16.0.0/16 addresses). We take the final hop in the initial network to be the PGW. For example, if initial network hops are in private address space, we take the final private hop as the PGW. We find that this general heuristic in fact captures a wide range of MNO operator configuration.

Using the heuristics described above, we first discovered the overall number of detected PGWs in the four largest U.S. MNOs. We then present our attempts to locate each instance using state-of-the-art IP-geolocation databases.

In total, we discovered 89 gateway IP addresses within Sprint's network, 162 within Verizon Wireless, 99 within AT&T, and 199 in T-Mobile's network. These results are summarized in Table 5.4. A large number of these addresses were private IP addresses. For instance, in the AT&T case, 77 of the 99 (77%) gateways detected were private addresses. We found similar high fractions for Sprint, where 49 out of 89 gateways (55%) were private, as were all of T-Mobile's 199 gateways. Verizon showed a slightly different configuration with no private IP addresses in its infrastructure, with all internal hops within ASN 6167.

Many of these PGW IP addresses are located within the same data center, therefore we initially tried group by geographic location. To determine the locations of these PGW instances, we attempted to geo-locate each of the gateway IP addresses identified by our traces. Since many of our detected PGW IP addresses are in private address space, we used geographic information from adjacent hops in these cases. For each gateway IP address and



(a) AT&T. (b) Sprint.

(c) T-Mobile. (d) Verizon.

Figure 5.10: Determined PGW locations from geoIP databases. Simply geolocating detected PGW addresses is too inaccurate to determine the numbers, and locations of PGW instances.

its adjacent hops, we averaged the latitude and longitude given using Akamai's Edgescape [5] geo-location database.

We found that many of these address sets have poor and inconsistent geographic information. We illustrate this by plotting the average geographic coordinates of each detected PGW, displayed in Figure 5.10. The large number of unique locations of PGW addresses, and their high variance, highlight the coarse and inconsistent location information in existing databases. Both T-Mobile and Verizon show large numbers of PGW locations evenly distributed throughout the country. These inexact locations make it difficult to determine the numbers of distinct PGW instances. In Verizon, for instance, the Northeast region of the U.S. appears as a single large cluster of points.

In order to overcome the limitations of this naïve mapping effort, and to simplify the localization problem, we attempt to cluster together traces from client at collocated PGW instances.

Clustering PGWs. Exploiting the deployments of carrier grade NATs in cellular networks, and the relatively stable mappings of client IP subnets to these locations, we cluster collocated PGW instances by client IP address. We outline our clustering methodology below:

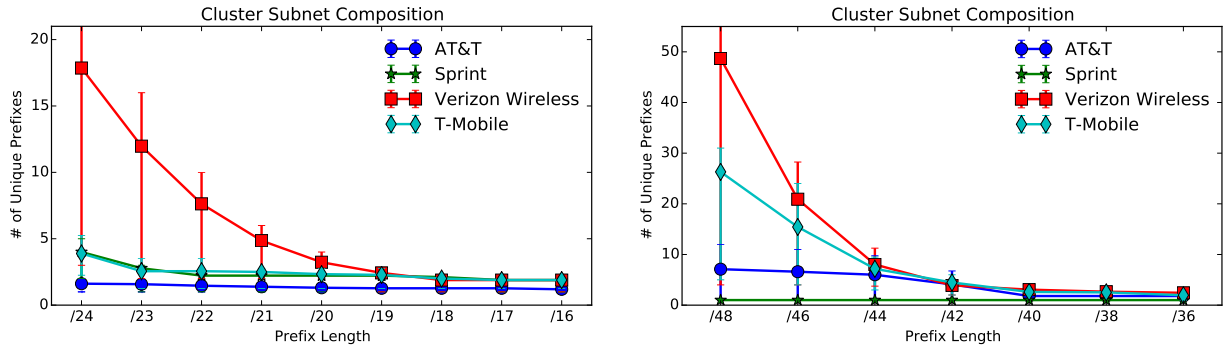
1. Mobile clients record their public IP address from an IP echo server, and perform traceroutes to their public IP address, yielding their assigned PGW address. The latter sends traceroutes which terminate at the client's current PGW.

2. We aggregate these measurements from all clients, and construct a bipartite graph for each operators with client IP addresses on one side and PGW IP addresses on the other. The client discovered pairs of $< client\ ip, pgw\ ip >$ form the edges between nodes.

3. We cluster each graph using the greedy community detection algorithm from Clauset et al. [36], generating clusters of PGW regions, and the client IP addresses contained within them. While connected components initially seem to fit this need, we found small amounts of noise in our data cause connected components to greatly underestimate the numbers of clusters.

Operator PGW Characterization. We analyze the results of our PGW clustering. We look at the total number of clusters discovered, as well as the composition of IP pools.

We first look at the composition of subnets within each detected cluster across different subnet sizes. From this analysis, several patterns emerge to the number and size of subnets assigned to each PGW. Figure 5.11 displays the average number of unique prefixes observed in each Tiller detected PGW cluster, across varying prefix lengths.



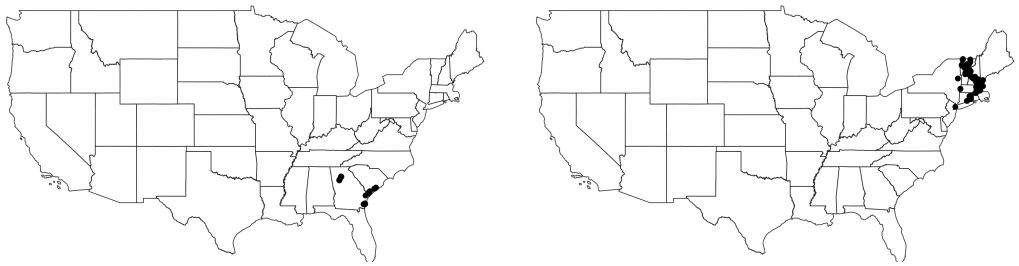
(a) IPv4 Cluster Prefixes.

(b) IPv6 Cluster Prefixes.

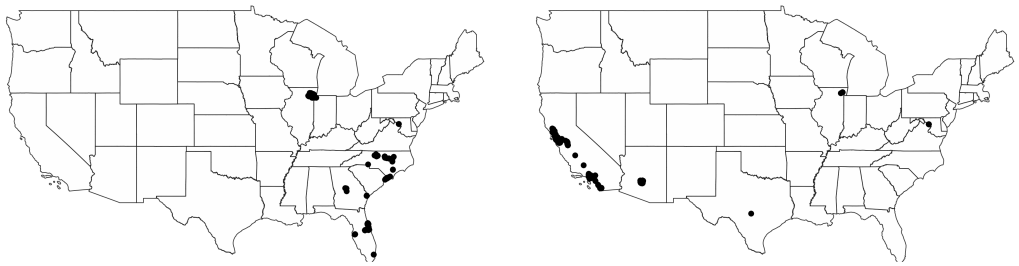
Figure 5.11: Size of cellular network clusters across different subnet prefix lengths for both IPv4 and IPv6. Points represent the average number of unique prefixes in each cluster, error bars represent the 25th and 75th percentile values. The allocation of a single /24 prefix per cluster of AT&T largely contrasts the over 30 observed in certain clusters in Verizon's network.

We find that certain mobile operators allocate a relatively small number of IP addresses to each PGW. For instance, AT&T maintains a nearly one-to-one allocation of /24 prefixes to PGWs. Both Sprint and T-Mobile also allocate small numbers of /24 subnets to each cluster – four /24s subnets per cluster on average. Of the four operators we profiled, we only found clusters with more than 10 /24 prefixes in Verizon, which averaged slightly above 20 /24s per cluster, and as high as 30 prefixes in some clusters. The implications of such a small number of prefixes at each GWP is a very high rate of IP reuse among active cellular clients. For instance, with 126 million subscribers in Q3 2015 [107], even with 100 GWP clusters across U.S. operators, that would still translate in over 1 million subscribers per /24 prefix, assuming an even distribution.

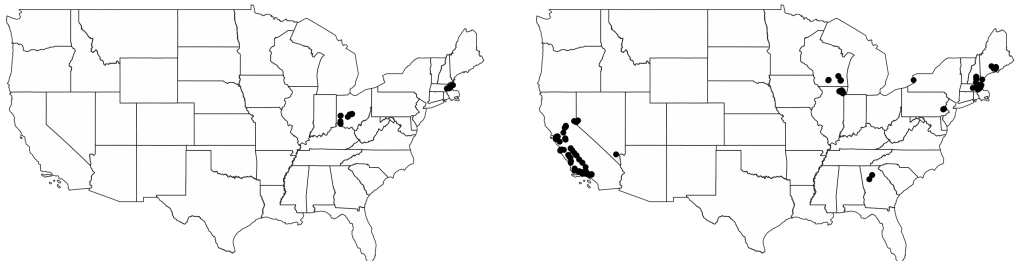
Spatial PGW Assignment. We next view the spatial locality of clients to PGWs. Using GPS measurements from clients, we plot the locations of ALICE clients in selected clusters for each of the four operators in Figure 5.12. In the figure, we can see the different spatial localities employed between operators.



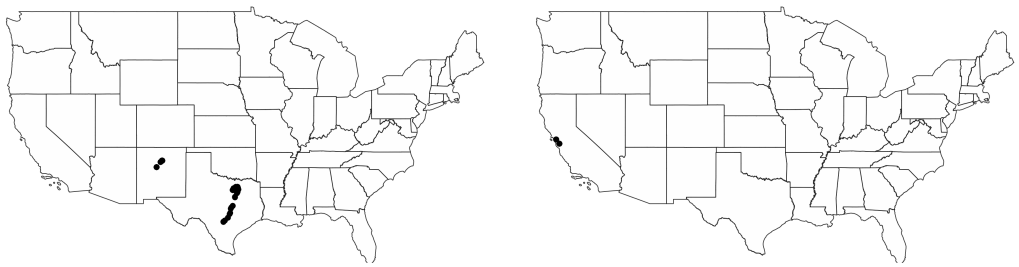
(a) Verizon Wireless.



(b) T-Mobile.



(c) AT&T.



(d) Sprint.

Figure 5.12: Locations of clients assigned to a PGW cluster. Each map represents the geographic coordinates of users assigned to that PGW for Verizon Wireless (top), T-Mobile, AT&T and Sprint (bottom). The differences in geographic locality of PGW assignment are clear between the close geographic proximity of Verizon assignments compared to the large geographic bounds of T-Mobile. While the large geographic range of assignment for T-Mobile clients increases core network latency, the optimal replica remains the same.

In all, we found Verizon and Sprint to practice the greatest spatial locality, signified by the tight clusters of user locations in the figure. In contrast, we see clients in AT&T and T-Mobile are assigned to PGWs from very large geographic regions, many spanning the country. We show in the following section (§ 5.4) that T-Mobile in certain instances maintains little if any spatial locality with PGW assignment.

In contrast to existing widsom of LTE PGW assignment, spatial locality cannot be assumed.

5.3.2 Validation of PGW Localization

Using ground truth information from three U.S. MNOs, we evaluate the accuracy of our client IP clustering. We compare the composition of our IP clusters to ground-truth obtained from two large US carriers containing both the number and locations of network PGWs, along with the allocation of IP addresses to each.

We calculate the pairwise community membership accuracy for each set of detected clusters. This common metric captures the accuracy of membership assignments by calculating the percentage of membership violations – an entity assigned to the wrong community – between detected communities and ground truth data.

Sprint. In Sprint we found a high level of accuracy in our clustering achieving a near one-to-one mapping between our detected clusters and ground-truth PGW regions. The pairwise accuracy for the case of Sprint's clusters was 99.3%. In all 9 discovered clusters in Sprint, each of the /24 prefixes of the same ground-truth region were mapped to the same cluster. In addition, all but one of the partitions mapped to unique Sprint's gateway partitions, with one region split across two clusters. We believe this clustering issue was caused by a relatively low number of measurements for that region, which was insufficient to join multiple PGW IP addresses.

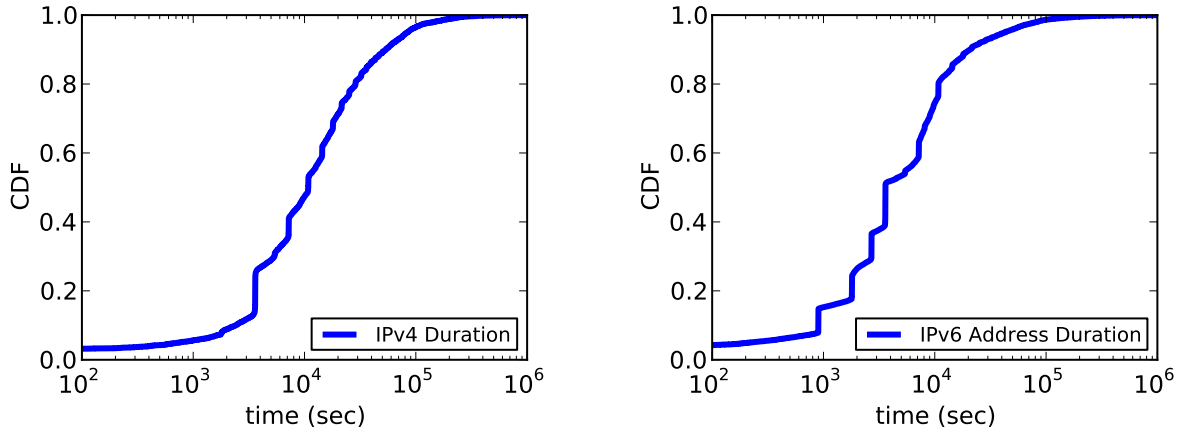
T-Mobile. The results of our clustering of T-Mobile's PGWs yielded a pairwise accuracy of 88.9% compared to ground-truth. We detected 18 unique PGW clusters from in T-Mobile, comprised of 74 /24 prefixes. The inaccuracy stemmed from the combination of nearby PGWs into single clusters. There were four cases where geographically adjacent PGWs regions were combined into single clusters. It is interesting to note the geographic locality of these combinations since our clustering algorithm uses no geographic information – only client and detected PGW IP addresses – for clustering. We believe this is due to the internal routing present in T-Mobile's network, which challenges our trace-based approach.

5.4 Mobile Client Dynamics

In this section present the results from our longitudinal study of client network dynamics. These network dynamics are separate from other sources of change in cellular networks such as tower hand-off and radio control states, and refer to assignment of data network resources such as IP addresses, DNS servers and packet gateways. While mobile devices change to active network interfaces frequently (e.g. WiFi, cellular), we find that network operator policies play a large role in determining a device's network context.

We investigate the length of network sessions from our longitudinal study of ALICE clients. We determine the length of individual network sessions by measuring the length of IP assignment. We next utilize the PGW clusters we introduced in the previous section to investigate inter-PGW dynamics. Our results reveal that although spatial locality is generally maintained, mobile operators do assign stationary clients to multiple, often distant PGW regions.

Client IP Assignment. We are able to detect the length of a mobile client's network session by the length of their IP address assignment. Due to its allocation during network attach procedures, a client's IP address duration can be used as a proxy for understanding each device's network session duration. This session length is driven by both operator policies



(a) IPv4.

(b) IPv6.

Figure 5.13: Duration of public IP addresses for measured mobile clients in the four major U.S. mobile carriers. Mobile clients see IP durations ranging from 1-3 hours at median, indicating high churn of assigned IP addresses. Jumps in distribution are an artifact of periodic measurements which occurred approximately every hour.

and handset behavior. Operators set the value of network idle timeouts, which detach inactive clients from the cellular network to free up network resources. As long as a client maintains some network activity within the timeout range, its session and current network context will remain. Handset behavior, therefore, is a large determinant in these session lengths.

We plot the distribution of network session times using the assignment length of cellular IP addresses. Our instrumented clients obtained their public IP address by contacting an IP echo service and reporting the results approximately every hour. We calculated the time between changes in reported cellular IP addresses. We filtered our data to only use IP addresses recorded over the cellular interface, and restricted the set to contiguous measurements with no gaps in collected measurement longer than 6 hours.

Figure 5.13 displays the distribution of time between cellular IP address changes for our instrumented clients in the four large U.S. MNOs for IPv4 (Fig. 5.13a) and IPv6 (Fig. 5.13b) addresses. We observed all four operators exhibiting similar patterns of IP assignment,

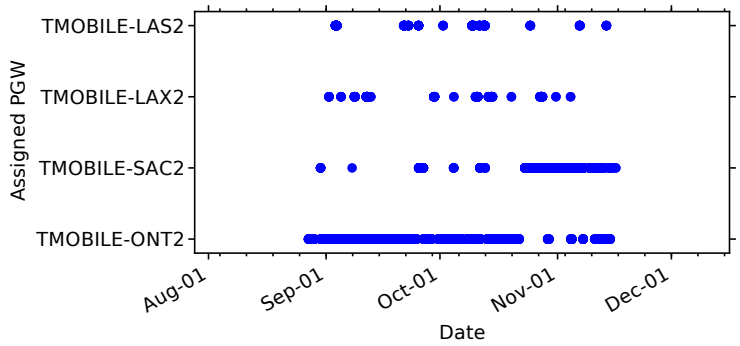
meaning either policies across operators is similar, or that length of network session is mostly dependent on user behavior.

For IPv4 addresses, we observed a median time between address changes ranging from 1-3 hours depending on the operator, yet the longest observed IP address lengths were for nearly 7 days. We found similar distributions of assignment lengths for IPv6 addresses to that of IPv4. This implies that mobile operators are also pooling IPv6 addresses in addition to IPv4 addresses, even though address reuse for IPv6 is not necessary due to address exhaustion. This confirms the prior work by Plonka et al. looking at IPv6 address usage [88].

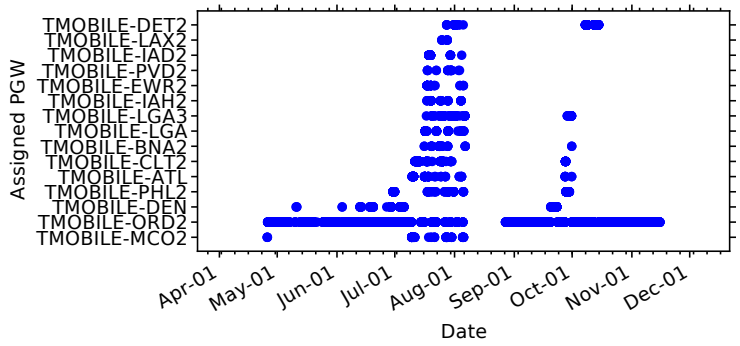
Client-Gateway Assignment. We now present our results detailing the length of time between PGW assignment changes for the four largest U.S. mobile operators. We detect the active PGW of cellular clients from their IP address, and the PGW clusters introduced in the previous section. PGW changes can occur for several reasons, the most obvious being client mobility. However, we find that in contrast to common assumptions of PGW locality, clients may switch PGW regions solely by operator policies for purposed such as load balancing. Since clients are assigned a new network context during each cellular network attachment event, there is the possibility of new PGW assignment.

For cases of multiple PGW assignments, we discover two types: one which assigns clients to nearby alternative PGWs, and the other which disregards geographic locality entirely, assigning clients to PGWs in an apparent random fashion. Our results highlight that geographic locality of client PGWs cannot be assumed, and more over, that instability exists in client PGW mappings. Examples of each type are illustrated in Figure 5.14.

Clients with multiple PGWs in the same geographic region can be load balanced across depending on individual region load. Since each PGW has a maximum capacity for simultaneous users and limits on its available bandwidth, it is reasonable to assume that certain PGWs may be overloaded, and more distant PGWs would need to be assigned to clients. We observed this behavior in all four U.S. MNOs.



(a) Geographically local PGW assignment.



(b) Chaotic PGW assignment.

Figure 5.14: PGW assignment for two separate clients in T-Mobile's network displaying two distinct patterns of PGW assignment, one load balancing clients to nearby PGW instances, and the other exhibiting almost random assignment behavior. Each pattern is evidence of non-mobility based PGW assignment, and shows how knowing client's current PGW assignment is invaluable for understanding cellular client performance.

Figure 5.14a shows this case for a single T-Mobile client in southern California, which is assigned to multiple, relatively nearby PGW regions, including 2 in the Los Angeles, CA area, one in Sacramento, CA and one in Las Vegas, NV. We verified from our measurements that this client remained within the greater Los Angeles area during this time interval. It is interesting to see that in addition to two different PGWs in Los Angeles, this client is also sent to Las Vegas and Sacramento during certain periods, the latter of which is over 450 miles away. As the number of PGWs continues to increase in cellular networks, we expect this behavior to become even more common, as cellular clients continue to greatly expand their traffic demands.

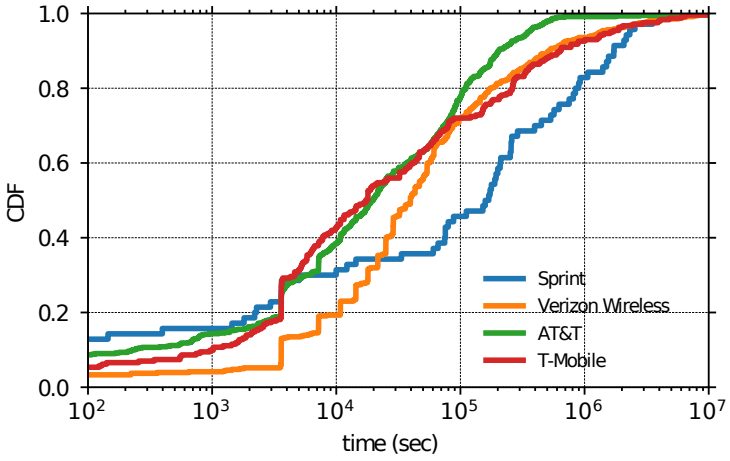


Figure 5.15: Interval between PGW changes measured from instrumented clients.

In contrast to the geographic locality described above, we also observed more chaotic PGW assignment to clients. In these cases, PGWs are assigned little regard for client location, and seem to follow either random or round-robin selection methods. This case of assignment is displayed in Figure 5.14b. In the figure, for the period between July 1 and August 15, the client located in Chicago is cycled through 14 separate PGW locations located as distant as Orlando, FL, Los Angeles, CA and Providence, RI.

Figure 5.15 displays the cumulative distribution of measured intervals between client PGW changes for the four U.S. MNOs. For this we limited measurements to clients which contained over 24 hours of contiguous measurements. The figure displays a wide range of PGW sessions, with some lasting from 10-100 days, and others lasting mere hours (or less). Surprisingly we find that for Verizon, AT&T and T-mobile approximately 70% of PGW sessions lasted less than 20 hours.

5.5 Summary and Contributions

The contributions of this chapter are summarized as follows:

- We presented results from our exploration of cellular LDNS infrastructure (§ 5.2). We find that all investigated operators utilize indirect resolution methods which

challenge existing DNS-based replica selection. We show client-resolver mappings to be inconsistent among cellular clients, and highlight its impact on replica mappings. We investigate the resolution performance and distance of public DNS services to cellular clients, longer resolution times than operator resolvers, but with much shorter tail performance.

- We presented results from our investigation into PGW locations for the four largest U.S. MNOs (§ 5.3). We introduce our techniques of mapping clients to these instances based on their public IP address, and show the heterogeneity in PGW subnet allocation across operators.

- We presented our longitudinal results from ALICE clients looking at the dynamics of network assignment over time (§ 5.4). We find that certain operators employ weak locality between clients and PGWs.

Chapter 6

A Network-Level View of Mobile Networks

6.1 Overview

While traditionally mobile devices have been primarily traffic consumers, new applications are turning them into major producers of Internet traffic. Mobile-to-mobile applications such as user-generated live streaming services, IP telephony, and video chat applications, as well as the growing number of cellular connected sensors promise to vastly increase the generation of traffic over cellular networks. The transition to VoIP voice communication (e.g. VoLTE) mean that these critical services are now reliant on Internet inter-domain routing. Unfortunately we lack a thorough understanding of the network-level structure of many cellular networks, including their AS composition, and connectivity.

We argue that new abstractions are needed to simplify the heterogeneity of mobile networks, while still capturing relevant differences in their structures. Towards that goal, we present the concept of a *Logical Domain* (LD) for mobile networks, which contains all ASes (network components) used within each MNO for their main functions, including (*i*) cellular core networks hosting client IP addresses, (*ii*) intra-MNO paths interconnecting multiple cellular core instances, and (*iii*) Internet connectivity. We develop empirical techniques to determine the ASes which compose a MNO's LD, and show the benefits of this construct for understanding MNO behavior.

6.2 MNO Organization

In this section we describe the major components of MNOs, and decompose each MNO into their underlying functions. We break each network into its *functional components*. These components represent the functions necessary to all eyeball networks, and include (*i*) hosting client IP addresses (*ii*) provide routing between customers within its network and (*iii*) provide Internet connectivity to its clients.

This decomposition is motivated by our in-depth characterization of 125 global MNOs, presented in detail in Section 6.4. We outline these three AS responsibilities for MNOs below.

- **Cellular Core Network.** This hosts the IP addresses of mobile clients, as well as infrastructure components such as PGWs. We define the cellular core network as the AS hosting cellular clients, as all routes from cellular clients must originate and terminate in this AS. We identified these cellular core ASes by looking at the IP-to-AS mapping of public IP addresses of our instrumented mobile clients. We further characterize our investigation of cellular core networks in Section 6.4.1.

- **Internal Transit.** The AS used to transfer packets between separate PGW instances. The nature of existing cellular networks means that packet transfers must travel between PGW instances through some network. This can be accomplished either entirely through private networks, co-owned backbone networks, or independent transit providers. Further analysis of intra-MNO paths is given in Section 6.4.2.

- **Internet Transit.** Provide Internet connectivity to cellular networks. We profile the Internet and inter-domain connectivity in Section 6.5.

In practice we find a wide variety of mobile network configurations, with network functions disbursed across different ASes in nearly all possible combinations. For instance,

StarHub in Singapore hosts client IP addresses in one of three different ASes, all interconnected through the same network. In another instance, Cricket Wireless in the U.S. utilizes three separate, third-party ASes to interconnect its cellular core networks.

Part of this heterogeneity stems from the large variety of mobile network operators. Aside from base MNOs, which typically own the radio and core network infrastructure, mobile service is provided by virtual operators (MVNOs), virtual aggregators (MVNAs) which rent radio and infrastructure resources from base MNOs. Further complicating this landscape are roaming agreements between operators which are based on complex business arrangements. In practice, we find that this complicated landscape further obscures paths to and from cellular clients.

6.2.1 Logical MNO Domains

We argue for the creation of *Logical Domains* for MNOs, which capture the full set of networks used within a single MNO. The existence of multiple, and geographically disjoint ASes hosting cellular clients, and more importantly that intra-AS traffic must traverse one or more independent ASes, significantly distances cellular AS topology from other eyeball ASes.

In cellular networks, disjoint networks are the norm. The use of third-party transit providers between these AS instances breaks many of the assumptions long held about the efficacy of intra-AS routing, especially for P2P applications [30, 97]. We argue that the AS composition of MNOs must include not just the cellular core networks (and their ASes), but also the networks interconnecting them. The reason is that only this composition of networks fulfills all of the functions of traditional eyeball networks. We call this collection of an MNO's ASes the **Logical Domain** of a particular MNO.

6.2.2 MNO Motifs

From the set of LDs discovered from our dataset, we identify four motifs of MNO design. We describe each of the four motifs below, and show examples of each in Figure 6.1.

- **Monolithic AS.** These MNOs operate within a single AS which hosts cellular clients, cellular infrastructure, and provides intra/inter network routing. Example: T-Mobile in Germany resides within Deutsche Telekom's ASN 3320, which in addition to hosting cellular and broadband customers, also serves as a large Tier-1 transit network.

- **Self Contained MNO.** These networks host all client addresses and supply internal network transit, fully containing the itself within a single AS. In many cases, these networks exist as entirely private address space. Example: Vodafone Espana, ASN 12357.

- **Combined Internal/Internet Transit.** These MNOs utilize the same provider network both to provide Internet connectivity to their customers, as well as interconnect their cellular core instances. Example: AT&T Wireless uses its national backbone ASN 7018 to connect instances of its cellular core networks (ASN 20057).

- **Functionally Independent ASes.** MNOs which delegate the cellular core, internal network transit and Internet transit into entirely separate ASes. Similar to the previous case, traffic between cellular users passes through a separate AS to reach independent cellular core instances, yet in these cases this internal transit AS only connects cellular core instances, traversing through external Internet transit to reach external destinations. Example: Verizon Wireless connects its cellular core (ASN 22394) through its internal transit (ASN 6167), and utilizes multiple Internet transit providers to connect to external destinations.

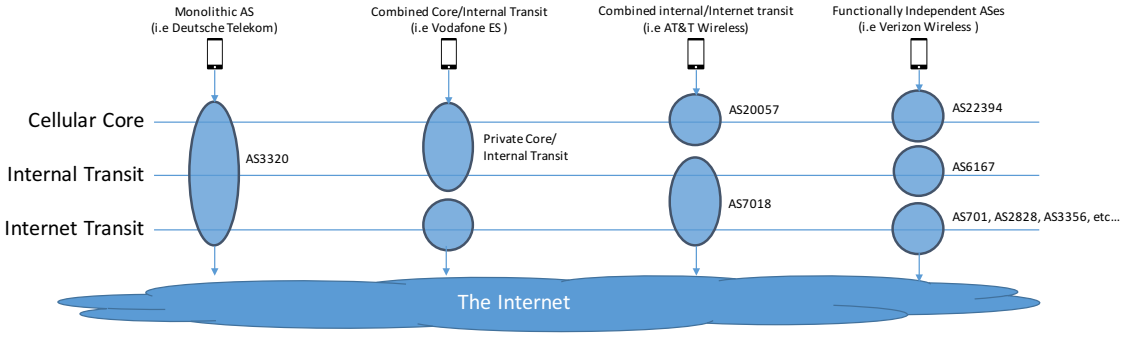


Figure 6.1: Cellular AS structure. We discovered operators which configured their network for each combination of AS-level structure possible.

6.3 Data Collection

Our analysis leverages a large collection of mobile measurements collected from over 1,900 cell users worldwide. In the following paragraphs we describe our data collection process.

6.3.1 Data Sources

Instrumented Mobile Devices.

We gathered mobile client data using ALICE clients (Chp. A). We launched traceroutes to top content providers (taken from Quantcast) We collected a range of contextual information, such as network connectivity state and public IP addresses.

- **Traceroutes.** We directed mobile clients to traceroute websites hosted by large CDNs and popular content providers, in light of recent work showing a majority of Internet traffic is directed towards a small number of large content providers [69]. In addition we directed clients to traceroute IP addresses of other mobile clients both within their own MNO and towards clients in other MNOs. In all we recorded over 8.25 million traceroutes from our instrumented mobile clients.

- **Public IPv4 address.** Clients recorded their public IPv4 address, IP address outside of cellular network NATs, as returned from an IPv4 echo service [1].

- **Device network interface trace.** Each device recorded its network connectivity state (Connected, Disconnected), active network interface (Mobile, WiFi), and network technology (LTE, HSPA, UTMS).

Server Traces to Mobile Clients. We performed traceroutes from servers of a large content delivery network to each /24 IPv4 prefix in the four largest U.S. MNOs (AT&T, Sprint, T-Mobile and Verizon Wireless). Traces were performed approximately once every hour to each subnet, and were collected during a 5 month period between August 1, 2015 and December 31, 2015.

We pair these traces with mobile end-host traces directed at the same set of servers during the same collection period. Traces were matched to server traces performed within the same hour bucket as the mobile client trace, based on *(client /24, server IP, date, hour)*, where the client /24 subnet is based on the /24 subnet of the public IP addresses of the mobile client, described above.

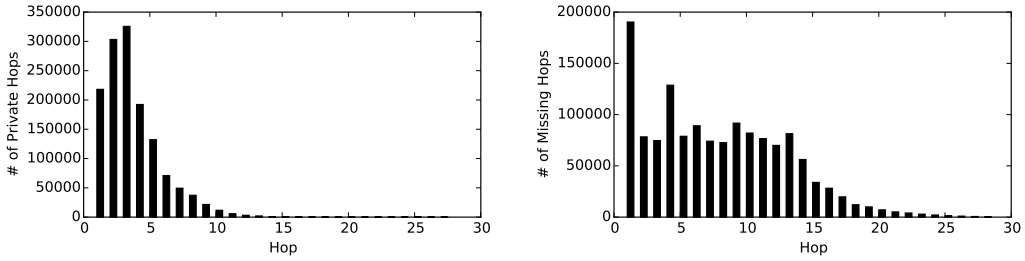
These paired client and server traceroutes allow us to explore the path symmetry between cellular clients and content. In all, we collected over 205,000 symmetric paths from the four U.S. carriers.

6.3.2 Mobile Traceroute Processing

The next paragraphs outline the procedures used to process mobile client traceroutes.

Raw Traceroute Processing While prior work has developed techniques for processing end-host traces, mobile end-host traces contain significant amounts of noise unique to their collection. The mobility of users across networks, that is between cellular and WiFi, means a path could completely change source networks in the middle of a traceroute. In certain

[1] `http://whatismyip.com`



(a) Private Hops. (b) Missing Hops.

Figure 6.2: Mobile traceroutes experience large numbers of missing and private hops.

cases, if the TTLs are high enough, may appear to be a valid, especially if the target network or its provider has been reached.

We develop a series of techniques to sanitize mobile end-host traces to accounts for the variety of confounding factors unique to them. We outline this method below.

1. **Interface Filtering.** As we are only interested in traces over cellular networks, we filter out any traces where the active interface polled by the operating system was not *cellular*. In addition, used the connectivity trace for each user to identify periods of connectivity change. We filtered out any measurement which was conducted during a connecivity change, for both network interface changes as well as network technologies (e.g. shifting from LTE to 3G).

2. **Private Addresses.** MNOs utilize private addresses at much higher rates than other eyeball networks. While previous approaches typically discard hops in private addresses, the large percentage of traces which reside in private address space require these to be maintained during processing. In the extreme case, T-Mobile in the U.S. utilizes private address space for both their cellular core network as well as their nationwide internal transit. The fraction of all traceroute hops which are private are shown in Figure 6.2a.

3. **Missing Hops.** Similarly to the private addresses of cellular core networks, there exist much large percentages of missing hops within cellular infrastructure. The fraction of all traceroute hops in our dataset with missing hops is shown in Figure 6.2b.

4. **Unannounced Infrastructure IP addresses**. We found it common that many MNOs would use addresses for infrastructure routers which were not listed in global routing tables. For these addresses we attempt a reverse DNS lookup, and if available, match the domain name to a set of known MNO domains. For instance, unlisted addresses in AT&T Wireless were identified by their reverse DNS names which resolved to a `cingular.net` domain.

5. **Source AS Filtering.** We filtered our traces where client IPs were not in that operator's source AS. We define source AS as the AS in each mobile operator with the largest fraction of recorded client IP records. While there are instances where mobile operators utilize multiple ASes for clients, we found the vast majority of operators to use only a single client AS.

6. **Prior procedures on inter-domain path analysis.** We follow the procedures of Chen et al. [28] and Mao et al. [76] in determining AS-paths from traceroutes.

6.3.3 Logical Domain Discovery

We directed mobile clients to traceroute IP addresses *within* their own operator. The ASes crossed on paths between addresses in the same cellular core AS constitute the intra-network connectivity of an MNO. We use these intra-MNO paths to discover the *internal transit* component of MNOs.

Instrumented clients were directed to periodically trace a random set of IP addresses, taken from the same set of reported mobile IP addresses used in the previous section, which match the AS from the requesting client. We analyze the over 208,000 traces between MNO

peers recorded from 22 separate MNO source ASes, collected between March 1, 2016 and June 1, 2016.

6.4 An AS-level look at MNOs

In this section we characterize the AS structure of MNOs, and present our methodology for grouping ASes into logical MNO domains. In order to discover the organization of each component, we perform separate experiments to discover the structure of (*i*) the cellular core AS (*ii*) internal network transit of MNOs. Our experimental results reveal that MNOs often utilize multiple ASes, separated by the different needs of the operator.

6.4.1 Cellular Core ASes

We define the cellular core network as the AS hosting cellular clients, as all routes from cellular clients must originate and terminate in this AS. We identified these cellular core ASes by looking at the IP-to-AS mapping of public IP addresses of our instrumented mobile clients.

We polled the mobile operating system to find both the device's mobile provider name in addition to the active network interface. In cases where the active interface was a mobile network, we identified public cellular IP addresses through an IPv4 echo service. Each IP address returned was mapped to autonomous system using IP-to-AS mapping using pWhoIs data.

We find the vast majority of MNOs utilize a single AS for their mobile clients. Within each operator we often fine large numbers of ASes which represent a very small (<1%) percentage of client IP addresses. These additional ASes seem to fall into three main categories: other MNOs which have roaming agreements with the target carrier, VPN services, and measurement errors.

We discovered several cases where the mobile operator name provided by the operating system was paired with IP addresses in competing mobile carriers. We believe these represent

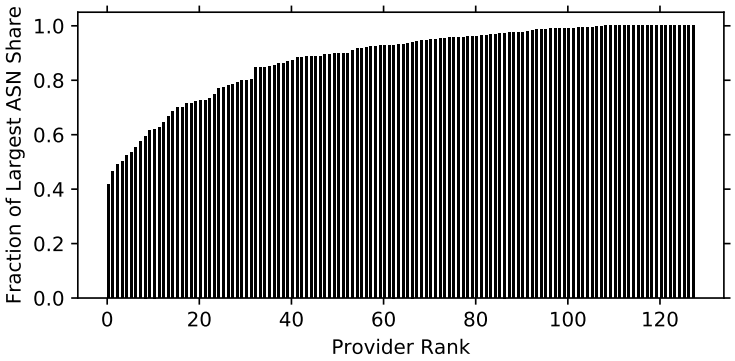


Figure 6.3: Fraction of largest ASN share for each provider, ranked by percentage. The vast majority of MNOs utilize a single AS for mobile clients in our dataset. A few cases exist where providers utilize 2 or more ASes, signified by those providers on the far left of the graph, but these isolated cases are the exception.

cases of certain roaming agreements, known as *local breakout*, where traffic from a roaming user is directed out through that visiting networks' PGWs. This is in contrast to *home routed traffic* where traffic is forwarded back to the home operators' networks. In the former case, the visiting networks' AS hosts the roaming user, in the latter case the home networks' AS hosts the roaming user. These roaming scenarios While the relations between mobile operators, and its impact on inter-domain routing is interesting, we leave that as an exercise for future work.

Overall, our measurements data encompasses 237 unique MNOs from around the globe. These consist of full MNOs, both light and heavy MVNOs, and MVNAs (i.e. Google Fi). Of these providers. We focus on the top 128 mobile operators, where we have at least 10 unique mobile IP addresses for each. Keep in mind this is not just 10 unique measurements, since PDP contexts, and consequently, IP assignment to mobile devices can last up to several days [116].

For each of the 128 mobile operators, we took the fraction of ASes mapped to reported IP addresses for each operator, displayed in Figure 6.3. For each operator, we took the AS representing the largest share. In the vast majority of cases, this AS represented over 90%

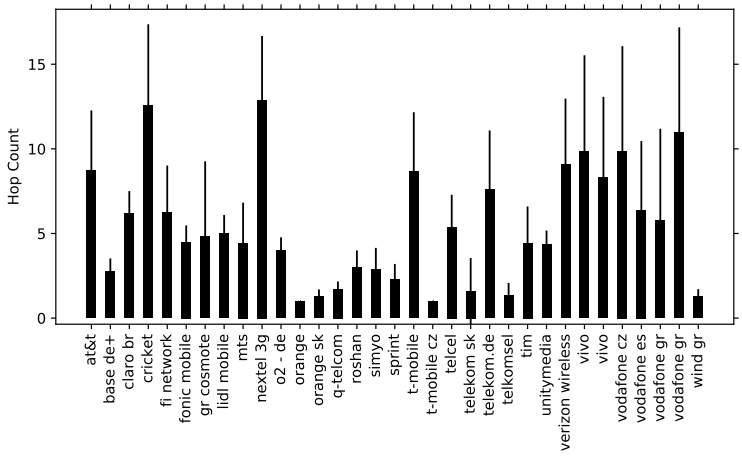


Figure 6.4: Number of hops between intra-MNO peers. The bars represent the average traceroute hop count between peers, with the error bars displaying the standard deviation of the distribution.

of reported IP addresses, with most substantial outliers coming from the aforementioned roaming agreements. Notable exceptions operators which utilize multiple ASes for clients, including the MVNA Google Fi, which in the U.S. switches between Sprint (AS 3651) and T-Mobile (AS 21928), Starhub in Singapore which uses 3 separate ASes for its mobile clients (AS 9874, AS 38861, AS 4657), Smart in the Phillipines (AS 132199, AS 10139). In another case, MNO *3* operates separate services in multiple European countries, each with their own AS, yet the provider name used is the same.

With the exception of these cases, we take the AS with the largest share for each provider as a cellular source AS. For the remainder of our analysis, we limit our further study to the 88 ASes discovered above. Our inter-domain analysis uses these 88 ASes as cellular sources.

6.4.2 Intra-Network Connectivity

We next investigate the intra-network connectivity of cellular operators. Figure 6.4 displays the average traceroute hop length between MNO peers, with error bars representing the standard deviation. Hop length encompasses both private and public IP addresses. The figure shows the differences between network size for the different MNOs, with some certain

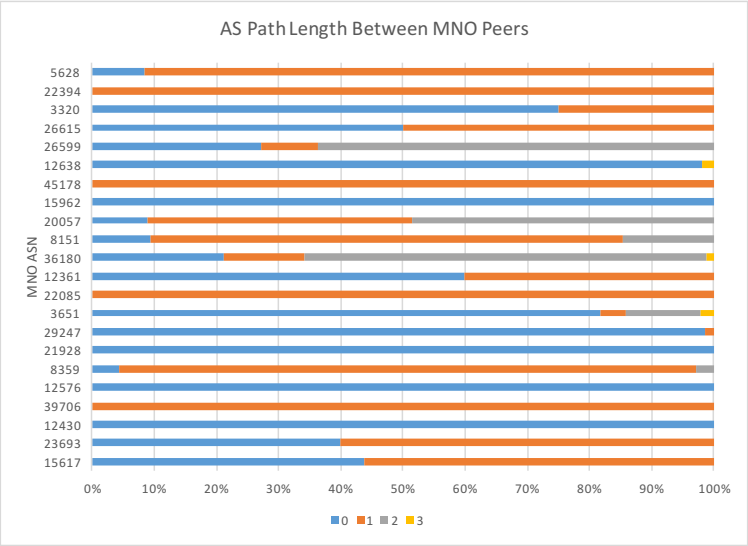


Figure 6.5: Fraction of AS path lengths for each cellular ASN studied. We find that traces *within* MNOs can cross up to three independent ASes. The fraction of paths taken by each is dependent on the number of paths to clients in the same, or distant PGWs.

MNOs such as Nextel in Brazil averaging over 13 hops between MNO peers, while others such as Orange in Germany average only a single hop.

The differences between operators, as well as within operators, is dependent on the number of gateway instances and the distance between each instance. For example, traces between clients behind the same PGW only contain a single hop – the gateway itself. Much of this has to do with the opacity of cellular core networks, and the structure of cellular networks, where public IP addresses do not exist for clients until their gateway instance is reached. The path length between PGWs therefore correlates roughly with their geographic distance. It is unsurprising that operators in large geographic regions such as the U.S. have large hop counts between peers. Yet even smaller geographic regions, such as Vodafone in Greece average some of the largest hop counts between peers in our dataset.

Following the procedure outlined in the previous section, we generate AS paths for these intra-MNO traces. From analysis of the intra-MNO AS paths, we discover that AS paths of intra-MNO routing can cross anywhere from 0 to 3 separate ASes to interconnect disjoint

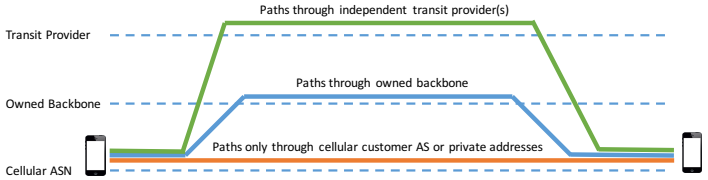


Figure 6.6: Different patterns of intra-MNO AS routing.

core networks. In this case, 0 ASes refer to paths entirely of private addresses. Figure 6.4 displays the number of AS-hops encountered on intra-MNO traces, aggregated by source cellular core ASN.

The figure shows that operator diversity also extends to the AS structure of MNOs in addition to overall network size. For each path, we combine private addresses with cellular core ASes. We find that traces MNOs can cross up to three independent ASes. The fraction of paths taken by each is dependent on the number of paths to clients in the same, or distant PGWs.

Looking at the AS path of intra-MNO paths, we find several patterns emerge in intra-MNO routing, displayed in Figure 6.6.

- **Private/Core Cellular Network Only.** Cellular traffic is contained either entirely within private address space, or only within the same core cellular ASN.
- **Owned Backbone Network.** Traffic between core network instances traverses one or more separate ASes, yet ones that are owned by the same organization as the MNO.
- **Independent Transit Network.** Traffic between core network instances travels over one or more ASes *not* owned by the MNO. In practice, several heavy MVNOs utilize this model.

Figure 6.7 shows a mobile traceroute for a client in Cricket Wireless, a heavy MVNO which runs atop AT&T's mobile network, yet operates their own AS and cellular core

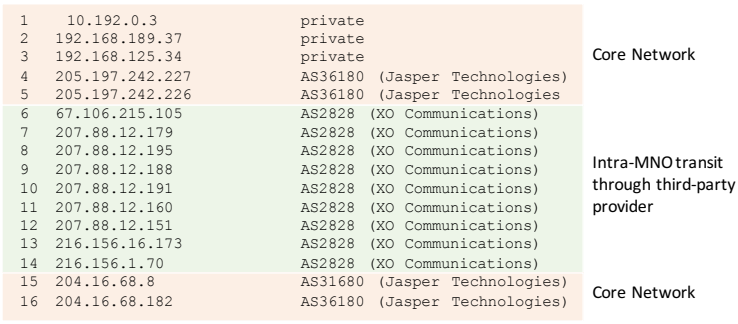


Figure 6.7: Intra-MNO traceroute for Cricket Wireless client. Cricket Wireless is a heavy MVNO which utilizes third-party transit providers to route packets between core network instances.

network. Routes between Cricket clients traverse one of several independent transit providers, AS 2828 or AS 6461, between its cellular core instances.

As we will see next, understanding these motifs of cellular network structure are essential for understanding their Internet connectivity.

6.5 Cellular Internet Connectivity

In this section, we present our results on the inter-domain connectivity of MNOs. We demonstrate that when analyzing MNOs, it is imperative to treat MNOs according to their logical domain in order to fully capture this inter-domain connectivity and routing behavior of MNOs.

6.5.1 Traceroute AS-Connectivity

The AS connectivity degree of MNOs appears vastly different when looking only at the cellular core ASN compared to the MNO's logical domain.

We chose to look at the AS connectivity of MNOs as determined by our active client traces rather than BGP, since we found BGP derived connectivity to over represent the connectivity of many MNOs. As our traces were directed at top content providers, as well as other MNOs, we feel these traces to be more representative of common paths of MNO

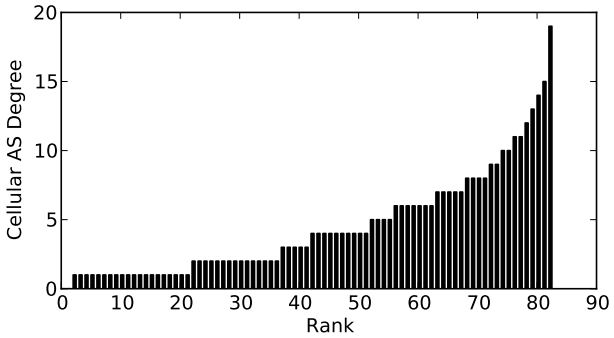


Figure 6.8: Traceroute determined AS-connectivity.

clients, now and in the future. When looking at the BGP connectivity from common sources such as RouteViews, we found several cases where the AS connectivity was overrepresented in BGP by small fractions of routes published through ASes which we never observed on any actual path to or from cellular clients.

Figure 6.8 shows the AS connectivity degree of the 88 cellular source ASes determined in § 6.4.1. From these cellular core ASes alone, the connectivity degree appears quite low, with one third of cellular networks connecting to two or fewer ASes. Looking at these core ASes leads one to believe that most cellular networks are poorly connected, to the larger Internet.

When considering the logical domain of an MNO, however, we find that these networks become much better connected. Table 6.1 displays the AS connectivity change between an MNO's cellular core AS and its logical domain for a subset of our dataset. An MNO's logical domain can greatly increase the measured connectivity of an MNO, and give a more accurate view of its actual AS connectivity. As an extreme example, the cellular core AS of AT&T Wireless is only connected to a single provider, yet its logical domain has a connectivity of 27, an increase of 2600%.

6.5.2 Path Symmetry to Content

We now look at the symmetry of paths between mobile clients and content servers from a large content delivery network. Our analysis of path symmetry reveals two main findings.

MNO	Cellular ASN	Degree	Combined ASNs	LD-Degree	Difference
MTS (RU)	8359	6	[8359,13174]	7	+1
Verizon Wireles	22394	3	[22394,6167]	23	+20
Vivo (BR)	26599	1	[26599,27699]	3	+2
AT&T Mobility	20057	1	[20057,7018]	28	+27
Sprint	3651	1	[3651,10057,1239]	23	+22
Cricket Wireless	36180	3	[36180,2828,6461,7018]	56	+53
UniNet (MX)	8151	10	[8151,13591]	10	0

Table 6.1: Logical domains can greatly increase the connectivity degree of MNOs.

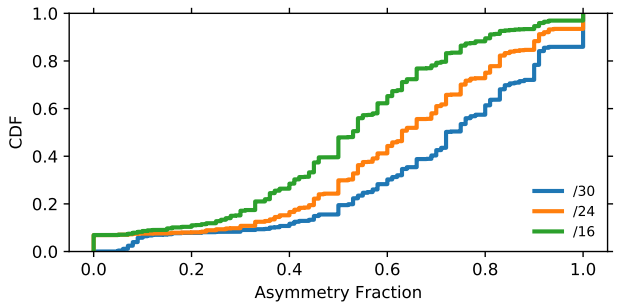


Figure 6.9: Fraction of path asymmetry for the four largest U.S. MNOs and a large content provider.

First, the link-level paths between cellular users and content are highly asymmetric, as are the AS paths. Second, AS paths only match source and destination ASes when considering the MNO's logical domain.

We calculated path asymmetry by calculating the Levenshtein distance, often known as the edit distance, between client and server paths. The Levenshtein distance is commonly used to measure the difference between two sequences, and calculates the number of "edits" needed to make the sequences match, including insertion, deletion and substitution operations. We calculate the normalized path asymmetry as the Levenshtein distance between the client path and the reversed server path, divided by the length of the longest sequence.

Overall, we find link-level paths between cellular clients and servers to be highly asymmetric. Figure 6.9 displays the normalized path asymmetry for paths, aggregated by different length subnets. Less than 10% of paths are 30% or less asymmetric, meaning

there are very few paths with highly similar link-level paths – even at /24 subnets. This is confirmed by the more than 70% of /24 paths which have greater than 50% path asymmetry. For /32 paths, over 50% of /32 paths have 100% asymmetry!

Prior work by Sanchez et al. [96] looked at the path asymmetry between edge networks and PlanetLab, finding /24 link-level paths to exhibit near 40% normalized asymmetry at median, compared to the 60% experienced by our mobile clients.

We also find AS paths between clients and content to experience similarly elevated levels of asymmetry. Figure 6.10 displays the normalized path asymmetry for AS paths between cellular clients and content servers. While nearly 40% of AS paths are completely symmetric, another 40% have 25% asymmetry. Again this, is nearly 50% higher than prior studies to broadband networks, which found only 40% of paths to exhibit any AS path asymmetry.

Part of the issue with asymmetry comes from the isolation and opaqueness of dedicated cellular network ASes. Of the over 205,000 reverse paths we collected, in only 107,995 (52.5%)reverse paths do the client traces begin, and server traces end, in the same AS. Much of this is due by client traces often not encountering their own cellular AS on their paths, with only 22,949 (11.1%) client paths containing the client's cellular AS, versus 59,316 (28.8%) server paths which do. This disparity is partially explainable on the client side by client traces encountering the private interface of cellular network gateway routers, and, on the server side by the overall opacity of the cellular network, where the entire cellular AS is unreachable in many locations.

In many ways, this motivates the need for MNO logical domains, for understanding path information to/from cellular clients. In each of the instances above, client and server paths become fully symmetric regarding source and destination when we aggregate cellular networks by the logical MNO domains

After aggregating AS paths by logical domain, we find greater path symmetry, shown as the dashed line in Figure 6.10. We also find significantly higher source and destination

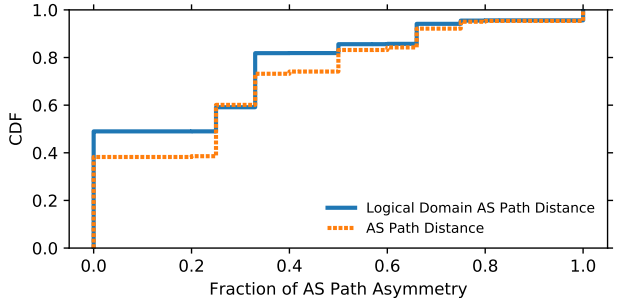


Figure 6.10: Fraction of AS-path asymmetry for the four largest U.S. MNOs and a large content provider.

symmetry, with 86.5% of joined paths containing source and destination AS symmetry. Similarly the MNO logical domain is much more frequently once aggregated by logical domain, with 69.2% of client traces and 67.2% of server traces passing through the MNO logical domain.

Identification of MNO's logical domain from traces is important largely due to the current opacity of MNO's networks, since the majority of traces to cellular end-users will not reach their destination. Without a reliable way to detect an MNO's network, these traces are largely meaningless, since it cannot be determined at what point the trace terminated, either at the entrance to cellular core AS, or elsewhere along the path.

6.6 Related Work

A large body of work has looked at modelling and observing the Internet's AS topology [39, 48, 51]. Dhamdhere et al. [39] categorize ASes based on their connectivity and position. In contrast to the author's work, we showed throughout this paper that MNOs blur the lines between traditional eyeball networks and large transit providers.

The work by Gill et al. [53] investigated AS connectivity from edge networks in relation to large content providers, finding a much flatter topology due to expansion of content providers' networks, and close relationships with eyeball networks. The work by Anwar et al. [11] looked at inter-domain routing policies using traces from eyeball network vantage points toward large

content providers. Prior work has looked AS path discovery from end-user vantage points. Chen et al. [28] expanded the Internet AS topology map from crowdsourced end-user traces, developing methods for processing noisy end-user traces for AS link discovery. Our work naturally follows from these by expanding the investigation into eyeball networks to include cellular networks, which have been the fastest growing eyeball networks in recent years with regard to traffic volume.

Several projects have utilized instrumented mobile devices to perform network measurements of MNOs [61, 95, 111]. Xu et al. [117] performed extensive traces from mobile clients to characterize the infrastructure of the four largest MNOs in the U.S. The authors were mainly concerned with the clustering of users to GGSNs, and in fraction of latency to popular content within the cellular network versus the Internet. The authors do not look into the AS-level topology or routing of these MNOs, choosing instead to investigate the infrastructure. Our work is the first examine the AS topology and routing of MNOs, and its implications for content placement.

Closest to our work is that from Faggiani et al. [47] which attempts to use traceroutes from mobile devices to explore missing links not found from BGP announcements. Their method involves large numbers of traceroutes from mobile devices, leveraging the mobility of users to connect to multiple networks from a single vantage point, thus multiplying its effectiveness. While the author's work discovers connectivity of Internet links, there is no discussion specific to MNO AS connectivity or routing. Our work focuses solely on MNO connectivity, and develops novel techniques to process mobile end-host traces to handle the increased noise and variability of the mobile environment.

6.7 Summary and Conclusions

In this chapter, we presented the largest investigation into the network level composition and structure of mobile network operators worldwide. Using over 8 million traceroutes collected

from 1900 volunteer mobile devices, we investigated the AS structure and connectivity of over 125 MNOs. We found that there exists a large variety of MNO structures, and categorize them into four structural motifs. We introduced the concept of Logical Domains for mobile operators which contain each of the functional components of these networks. We showed that Logical Domains can aid the understanding of mobile networks.

Chapter 7

TILLER: An End-Host Solution for Cellular Exploration

7.1 Overview

Cellular networks are the fastest growing sector of Internet traffic. Cellular operators have quickly been expanding their networks and technologies in order to meet this quickly rising demand. With mobile devices becoming the dominant vehicle for user content consumption [35], cellular networks are expected to grow at a 45% CAGR over the next 5 years [45].

Measuring cellular infrastructure is challenged by to the lack of visibility from external vantage points, limiting the set of vantage points (VPs) to those within cellular operator networks. The nearly universal deployment of NATs and restrictive firewall policies of cellular operators typically prohibit common network probes such as ping or traceroute from penetrating their networks. Attempts at mobile network exploration have therefore had to rely measurements from instrumented mobile clients. While gaining visibility inside these networks, mobile measurement platforms have their own unique constraints, including (*i*) recruiting sufficient numbers of diverse vantage points and (*ii*) computational, network and power resource constraints. These limit the coverage of these platforms as well as the measurement capabilities of individual devices.

While accurate network topology information, ranging from AS-connectivity to the geolocation of routers and end-hosts, is widely available for most of the Internet, this

information is largely nonexistent for cellular networks. While topology exploration and mapping has been studied and largely solved for wired networks, cellular networks render common approaches largely ineffective.

The concept of *coverage* within cellular networks is an open question. Existing cellular measurement efforts a vague "more is better" notion of coverage, with no objective metrics for comparison. This problem is exacerbated by the lack of cellular network ground-truth information available for evaluation. We attempt to take a first step towards effective cellular network measurement by that objective metrics for network coverage are the first step toward efficient and effective cellular network measurement.

Towards this goal, we propose a new abstraction for cellular network topology, the Gateway Cluster (GC) which represents pools of cellular clients assigned to the same cellular packet gateway (PGW). The use of GCs in cellular networks is based on the coalesce of all paths at client PGWs, and the natural partitioning of cellular client IP space among PGW instances. We argue that cellular network topology coverage can be based on the fraction of GCs observed from measurement vantage points. Since the number of GCs for a particular cellular operator is often unknown beforehand, as is the allocation of IP addresses to each GC, we can approximate this GC coverage by measuring the fraction of cellular IP space observed by vantage points.

We present the results of our analysis of into the IP allocation and usage of the four largest U.S. cellular operators. By analyzing both BGP announcements and request logs from a large CDN, we develop an objective methodology for deriving a baseline set of IP addresses to represent network coverage. We use this baseline coverage to evaluate the coverage results of over 2.5 years of mobile measurement traffic from the ALICE measurement platform. Looking at the individual and combined coverage of ALICE VPs, we find that a relatively small number of VPs (< 100) can provide near complete coverage of mobile networks according to our coverage definitions.

Using the insights learned from this analysis, we present the design and implementation of TILLER, a mobile distributed measurement system for cellular networks. Using the concept of GCs as its metric for coverage, TILLER is able to provide efficient topology characterization through adaptive probing of cellular networks, reducing the numbers of redundant measurements from mobile vantage points. TILLER combines local cellular topology views into a single global oracle which is distributed periodically between clients to minimize global probing.

In this chapter we make the following contributions:

- We introduce a new abstraction for cellular topology based on network PGWs and network IP space allocated to each, the Gateway Cluster (GC).
- We objectively define topology coverage for cellular networks based on the fraction network GCs covered by mobile vantage points. Since the number of GCs is not known beforehand, we approximate this coverage by the IP space observed from vantage points.
- We present a data-driven methodology for deriving a *baseline coverage* of cellular IP space using BGP announcements and request traffic from a large content delivery network.
- Using this new metric of we present coverage analysis from 2.5 years of longitudinal mobile measurements.
- We present the design and implementation of TILLER, a mobile distributed measurement system which uses adaptive probing to efficiently characterize and monitor cellular networks.

7.2 Cellular Network Coverage

Finding a general metric for network coverage is challenging due to the heterogeneity of cellular networks, their infrastructure and policies, and their overall opacity. Prior efforts at network exploration and mapping have utilized various metrics for coverage, including network routers observed [17, 106], and edges in Internet AS topology [28]. Unfortunately these previous metrics are ineffective for cellular topology, since much of cellular paths are either invisible, tunneled between cellular end-points, or routed through private address space. For instance, many cellular ASes contain no visible routing infrastructure, and instead only exist to provide IP addresses for cellular clients.

We propose to define cellular network coverage based on the fraction of GCs observed from mobile vantage points. Since the number of GCs is not known ahead of time, we can approximate this coverage by the measuring the observed client IP-space from mobile vantage points within cellular ASes. We present our method of obtaining this reprsentative IP space later in this section.

7.2.1 Cellular Gateway Clusters

We next introduce our concept of Cellular Gateway Cluster (GCs), and show that they provide the ideal abstraction for cellular network topology. Each Gateway Cluster is composed of a set of IP subnets, dynamically assigned to mobile clients behind that PGW. Formally we define each GC g_i as the set of all IP addresses, s, routed through the same PGW p_i, $g_i = \{s \in S : s \; routed \; through \; p_i\}$

As we argued earlier in this dissertation (Chp. 4), cellular PGWs largely determine client's network position, influence distances to content replicas, their network locality and network performance. From a network exploration perspective, GCs represent a natural partitioning of cellular IP space into discreet network and physical locations. For cellular topology and

Operator	# VPs	# GCs
AT&T	54	37
Sprint	18	14
T-Mobile	61	15
Verizon Wireless	89	39

Table 7.1: Summary of gateway clusters (GCs) determined for four U.S. MNOs. For each operator, we list the number of mobile vantage points used for measurements, and the number of GCs detected within each.

infrastructure discovery, we propose that metrics for network coverage be derived from the GCs in each network.

Unfortunately, due to the lack of ground truth information about cellular operators, the total number of GCs in each operator is unknown. We therefore lack an accurate target for coverage evaluation. Since each GC exists as a collection of client IP subnets, we can approximate GC coverage by measuring the fraction of client IP space observed from mobile vantage points. We use this coverage as our baseline to view the marginal utility of adding new vantage points. We also look at the marginal utility of continued measurements from individual vantage points over time.

Preliminary Results of Gateway Clustering

Using the approach presented in Chapter 4, we present our analysis of discovered GCs in the four largest U.S. operators. Our measurements come from 222 volunteer mobile clients spread across the four largest U.S. MNOs: 54 in AT&T, 18 in Sprint, 61 in T-Mobile and 89 in Verizon Wireless. For each operator, we apply graph clustering to construct a set of GCs for IPv4 addresses in these networks [1].

Table 7.1 displays a summary of detected gateway clusters in each of the four operators, which range from the 14 detected in Sprint to the 39 discovered in Verizon Wireless.

[1]While each of these mobile operators have transitioned to IPv6 as of January 2016, we currently only perform this for IPv4 addresses and gateway router addresses due to the bulk of our data being IPv4, since IPv6 traceroutes were not available in stock Android until version 4.4

Operator (ASN)	BGP (/24)	BGP (/48)	CDN (/24)	CDN (/48)	CDN % (/24, /48)
AT&T (20057)	17,152	46,220	6,674	9,227	(38.9, 19.9)
Sprint (3651)	45,565	4,061	38,924	2,048	(85.4, 50.4)
T-Mobile (21928)	42,112	65,537	9,327	5,929	(22.1, 9.0)
Verizon (22394)	38,107	1,181,088	34,096	3,394	(89.4, 0.2)

Table 7.2: Differences in IP space visibility from BGP announcements and from a large CDN. We find BGP announcements to be greatly over announce both IPv4 and IPv6 address space.

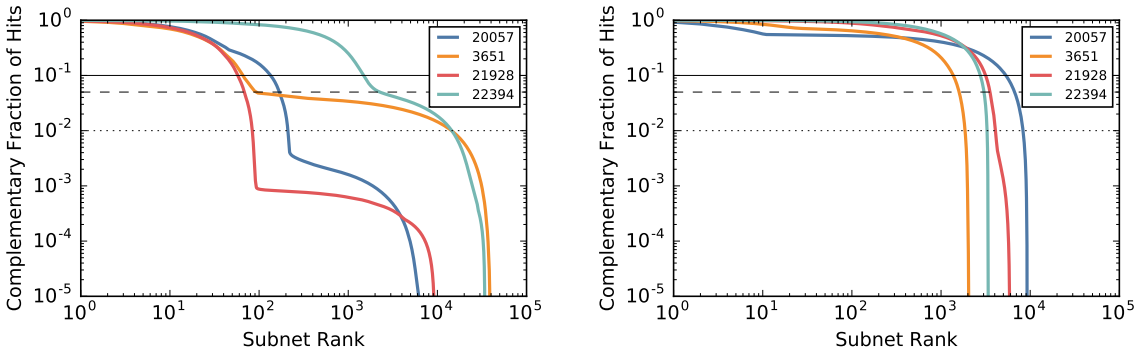
Unfortunately we make no assertions of network coverage from these results, or of the fraction of GCs we have discovered within each network. Lacking ground truth information from each operator – as is often the case – we have no way of knowing the total number of target GCs, or our measured coverage fraction. This is one of the challenges with mobile network measurements, a lack of an objective metric for calculating coverage.

In the following sections we address this problem of coverage, using observed cellular IP space as a proxy for gateway cluster coverage. Using multiple data sources, we derive metrics for determining IP coverage of cellular networks.

7.2.2 Baseline IP Coverage for Cellular Networks

In this section, we investigate cellular IP space to develop a baseline coverage for cellular network IP space. We compare two potential data sources: BGP announcements for cellular ASes, and the request traffic observed from a large content delivery network for these same ASes. For this analysis, as well as that on mobile network coverage, we look at the four largest U.S. MNOs: AT&T (AS20057), Sprint (AS3651), T-Mobile (AS21928) and Verizon Wireless (AS22394).

Each dataset captures records from January, 2016. The BGP announcements were obtained through logs from the large CDN for January 1, 2016. Request traffic was obtained from logs of that CDN's real-user monitoring system (RUM), which collects data from a Javascript beacon embedded within participating customers. This RUM data encompasses all of January 2016. Table 7.2 summarizes the results obtained from each dataset, and



(a) IPv4 /24 Subnet Requests.

(b) IPv6 /48 Subnet Requests.

Figure 7.1: CCDF of /24 and /48 subnet requests, ranked by total number of requests by subnet descending. While /24 subnets observe a bimodal distribution where a small fraction of addresses account for the vast majority ($> 95\%$) of request traffic, /48 subnet requests are distributed more evenly.

displays the total number of IPv4 /24 and IPv6 /48 subnets observed from both BGP announcements and CDN traffic, as well as the overlap of subnets from these two datasets.

We find that as a source of cellular IP space, BGP announcements are too coarse to be effective estimators for actual cellular IP usage. BGP announcements overestimate the numbers of both IPv4 and IPv6 addresses. We found the overlap between IPv4 datasets to be range from 22% in T-Mobile to the nearly 90% in Verizon. We found much greater variance in coverage from IPv6 addresses, which announce large blocks of IPv6 addresses that greatly over announce this space. In T-Mobile, which announces a single large /32, of the 65,537 /48 prefixes only 5,929 (9%) of /48 subnets were observed from client traffic. Even more extreme is Verizon Wireless which announces over 1.1 million /48 subnets, yet only 3,394 were observed – a total of 0.2%. It is apparent that BGP announcements are not accurate portrayals of cellular client IP space, especially for IPv6.

In light of BGP's over announcement of cellular IP space, we derive our baseline coverage on the IPs observed from CDN logs. We investigate the distribution of traffic across cellular IP space by counting the total number of hits seen for each /24 and /48 subnet seen in our CDN dataset. We find that while IPv6 addresses appear to exhibit relatively even load

distributed, in the case of IPv4 addresses, there exists a large disparity in request volume from for different /24 subnets. We plot the ranked cumulative request total in Figure 7.1.

Figure 7.1a plots the CCDF of /24 subnet requests, ranked by the number of requests descending. The horizontal lines in the figure denote the 90th percentile (solid), 95th percentile (dashed) and 99th percentile (dotted) of total requests. The figure displays a bimodal distribution of request traffic across subnets, with small fractions of observed IP space responsible for the vast majority of traffic. In the extreme case with T-Mobile, only 92 of the 9,327 /24 subnets (0.9%) account for over 99.9% of requests. Similar patterns are observed across the other three MNOs, with distributions shifting at the 95th percentile for all operators. IPv6 address usage (Fig. 7.1b) shows a much more even distribution of load across /48 subnets. Although there does appear to be very slight bimodal behavior, it is much less pronounced than the IPv4 cases.

We posit the cause of this distribution for /24 subnets is the frequent use of Carrier Grade NATs (CGNs) within cellular networks, which recent work by Richter et al. [92] found deployed in 92% of measured cellular networks. Subnets allocated to CGN pools would potentially account for large fractions of cellular traffic. If this is the case, the long tailed addresses would arise from would arise from traffic generated from untranslated addresses, such as customers using static IP addresses. For IPv6 addresses, recent work by Plonka et al. [88] found that cellular operators also dynamically assign IPv6 addresses across clients. This dynamic assignment helps explain the more even distribution of load across IPv6 subnets.

In light of this skewed distribution, we select a baseline set of IP addresses from those comprising the 95th percentile of traffic for both /24 and /48 subnets for each MNO. We consider these set of IP addresses to be the *baseline coverage* for these chosen cellular operators. The selected number of subnets in each baseline coverage is shown in Table 7.3.

Operator (ASN)	/24 (CDN%) [BGP%]	/48 (CDN%) [BGP%]
AT&T (20057)	168 (2.5) [0.9]	6,680 (92.3) [14.4]
Sprint (3651)	92 (0.2) [0.2]	1595 (77.8) [39.2]
T-Mobile (21928)	68 (0.7) [0.1]	3539 (59.6) [5.4]
Verizon (22394)	2257 (6.6) [5.9]	3032 (89.3) [0.2]

Table 7.3: Number of /24 and /48 subnets accounting for the 95th percentile of cellular requests. The table shows the 95th percentile of subnets, and their corresponding fraction of all observed prefixes for each operator. We use this subnet of IP subnets as a baseline for cellular network coverage.

Using this baseline coverage for each operator greatly reduces the scope of network coverage, since relatively few numbers of possible subnets account for the vast majority of demand. For three out of the four MNOs, we see less than 170 /24 subnets comprise 95% of mobile traffic, and that these make up 2.5%, 0.7% and 0.2% of *observed* subnets for AT&T, T-Mobile and Sprint respectively! This greatly reduces the sample space for exploration of cellular networks by at least 2-3 orders of magnitude depending on the operator. The implications of this are that it greatly simplifies the task of cellular network exploration from mobile devices, and presents tractable goals for measurement coverage. In the following section, we analyze the results of our longitudinal study looking at the coverage capabilities of individual mobile vantage points.

7.3 Mobile Vantage Point Coverage

In this section, we utilize our newly defined metrics for cellular network coverage to analyze the results of the ALICE mobile measurement platform. Using over 2.5 years of measurements across 222 mobile vantage points, we analyze the coverage of each vantage point in order to evaluate roughly the numbers of mobile vantage points necessary for mobile network coverage. Specifically, we attempt to answer for the first time the following questions:

- What is the possible coverage of individual vantage points?
- Are there any benefits to continuous measurements from mobile vantage points?

- How many vantage points are needed for complete coverage in mobile networks?

ALICE measurements are crowdsourced from volunteer devices, and represent real-world conditions for network behavior and device mobility. Our measurements come from 222 volunteer mobile clients spread across the four largest U.S. MNOs: 54 in AT&T, 18 in Sprint, 61 in T-Mobile and 89 in Verizon Wireless.

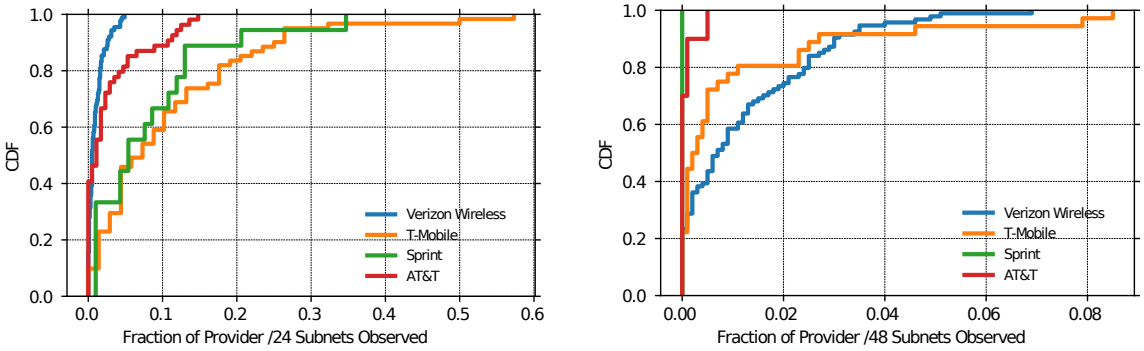
7.3.1 Coverage of Individual Vantage Points

Mobile devices have the ability to act as multiple traditional network vantage points within cellular networks. The physical mobility inherent with these devices allow single devices to measure multiple parts of the cellular network. This coupled with the dynamics of network operation, which can assign users to separate gateway clusters based on load/policies, mean that instrumented mobile devices are able to expand their visibility over time.

Using our newly available metrics of coverage, we look at the cumulative network coverage of individual vantage points from ALICE . Since ALICE crowdsources network measurements from volunteer clients, our analysis covers real device behavior in the wild, allowing us to view the impact of actual device mobility and network assignment policies.

We calculated the overall network coverage of each vantage point as the fraction of cellular IP space assigned to clients for the duration of their measurements. As part of their network characterization, ALICE clients periodically obtain their public IP addresses by contacting an IP echo service which returns the IPv4 and IPv6 address seen at the server. We aggregated client IP addresses into /24 and /48 subnets for IPv4 and IPv6 addresses respectively, as is common practice in network analysis. From the set of unique subnets observed, we take the intersection of this set with the baseline set of subnets determined in the previous section, to determine a network coverage fraction for each VP.

Figure 7.2 displays the cumulative distribution of VP coverage for each operator for both IPv4 addresses (Fig. 7.2a) and IPv6 addresses (Fig. 7.2b). While the range of coverage varies



(a) IPv4. (b) IPv6.

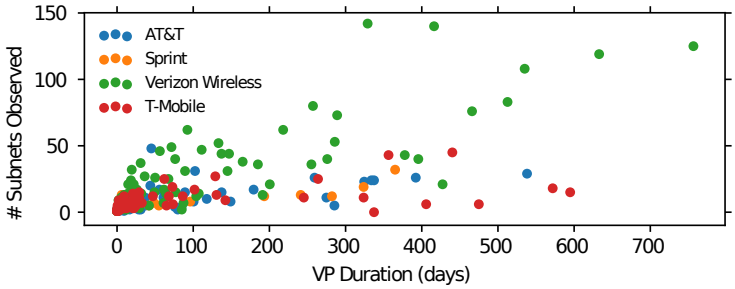
Figure 7.2: Coverage from individual ALICE clients for /24 and /48 subnets.

between VPs, we observe two trends in this coverage. The first is that a large fraction of VPs, around 40% in AT&T and Verizon, observe less than 1% of both IPv4 and IPv6 subnets. This is potentially due to retention issues with volunteers inherent to crowdsourced measurements. The other is that for IPv4 addresses, a small fraction of vantage points (5-10%) were able to observe upwards of 25% of operator networks, and in the case of 2 T-Mobile VPs, upwards of 50%. These vantage points, either through erratic network assignment policies, or frequent device mobility, are able to cover large portions of their operator's networks.

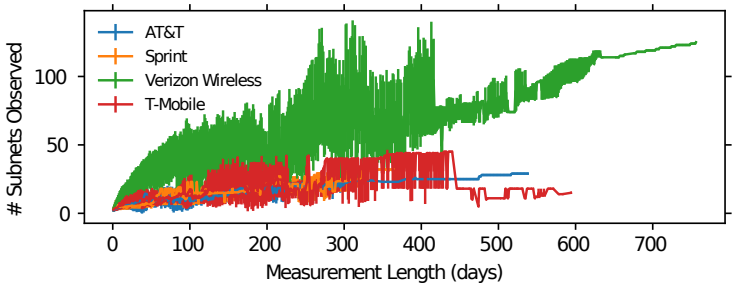
While we find that certain vantage points are potentially able to observe large fractions of operator networks, the amount of time needed for this, and the rate of exploration are unknown. We explore these temporal qualities of vantage point behavior in the following section.

7.3.2 Vantage Point Temporal Dynamics

We next investigate the coverage of individual vantage points over time. The extent to which mobile devices under normal conditions experience different areas of cellular networks, either through physical mobility or network assignment policies, is unknown. We utilize our 2.5 years of mobile measurements to explore the limits of individual device visibility over time. Figure 7.3 displays the /24 subnet coverage over time for each of the four profiled operators.



(a) Subnets per day.



(b) Subnets per day.

Figure 7.3: Subnets discovered by ALICE clients over time plotted against their measurement duration.

Figure 7.3a plots the total unique /24 subnets observed from mobile vantage points, plotted against the total length of VP observation. The figure shows an overall linear correlation with discovered subnets and time. The rate of discovery is dependent on the overall size of IP space deployed by each operator. For example, several VPs in Verizon observe nearly 150 unique /24 subnets, yet this accounts for less than 7% of Verizon's IP space at the /24 level. The nearly 50 /24s seen for certain T-Mobile clients, in contrast, accounts for over 70% of network IP space.

While this shows the potential of individual vantage points to become exposed to significant portions of cellular networks over large enough time scales, the figure also shows that the challenge of crowdsourced VPs come from retention. A large majority of VPs have less than 50 days of continuous measurements, and many have only a single recorded measurement.

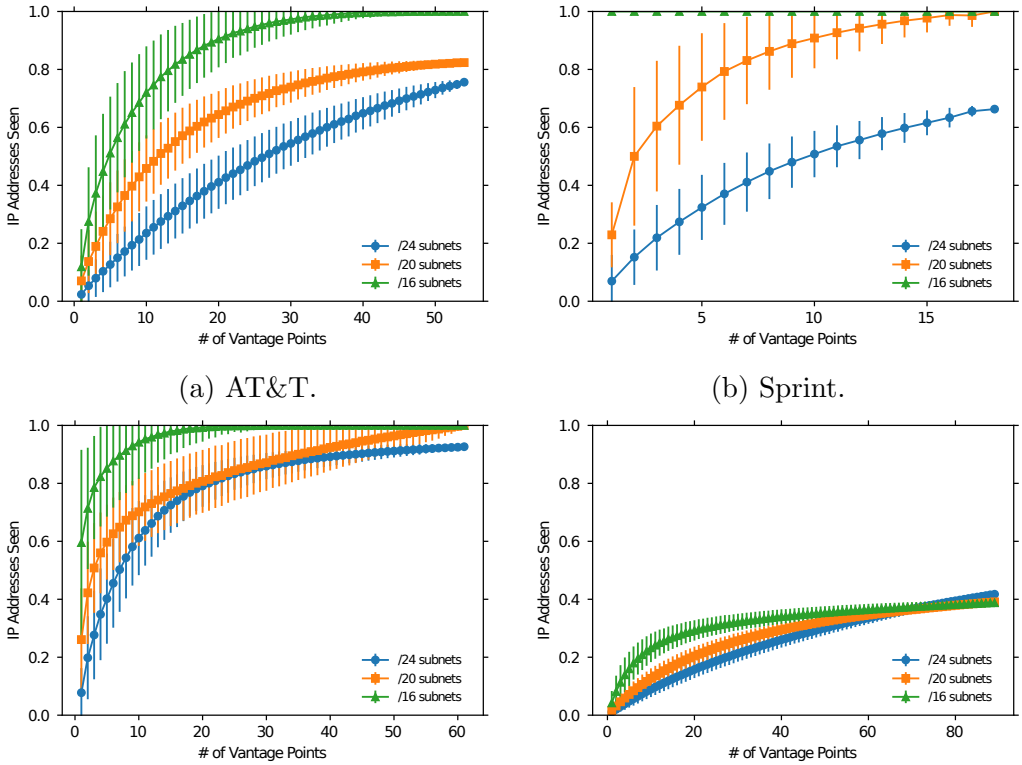
We find that the early portion of VP measurements (< 100 days) are the most informative for network investigation. Figure 7.3 displays the number of unique /24s observed from VPs each day after the start of measurements, averaged for each operator. The figure shows the information gain from each vantage point diminished over time. In particular there exists an inflection point on each operator's curves within the first 100 days where the rate of subnet discovery slows, yet does not disappear. This is beneficial for crowdsourced platforms since it means that the potentially short durations of many vantage point measurements are the most informative for each VP.

Reasons for this drop-off in VP information gain are that VPs often remain within the same PGW region, due to the (mostly) spatial locality of PGW assignment. The small numbers of IP addresses assigned to each PGW, and high rates of IP assignment from these pools, mean that measurements from users are often redundant beyond these first few days. We can utilize this with adaptive probing to limit the numbers of probes launched from individual VPs when we detect a previously observed area of the network.

7.3.3 How Many Vantage Points?

We utilize the data collected from ALICE clients to investigate the coverage achieved from different numbers of vantage points in each network. We evaluate this accumulated coverage looking at the overall coverage achieved by increasing the number of VPs, as well as the marginal utility of additional VPs in mobile measurement.

We measure the accumulated coverage of VPs as the union of observed IP sets from each VP across various aggregate subnet lengths. To calculate the coverage fraction, we take the intersection of these subnets with the baseline subnet set derived in the previous section, and divide the length of that by the length of the baseline set. In order to account for differences in vantage point mobility and measurement time in our crowdsourced platform, we take the average length of these unions from all possible combinations of VPs in our dataset. From



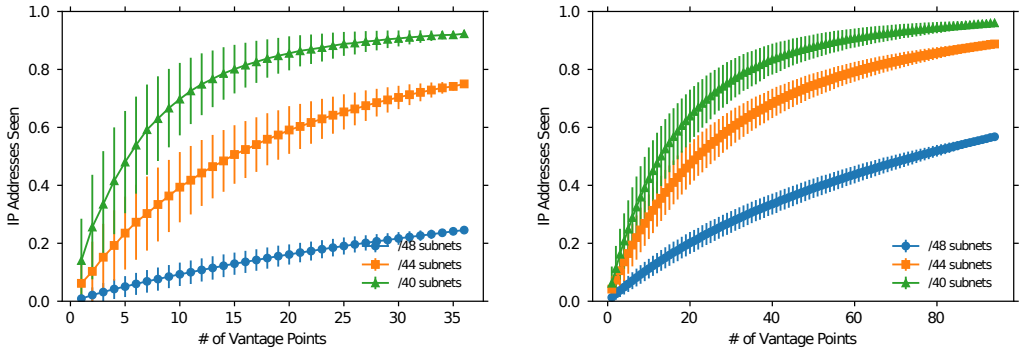
(a) AT&T. (b) Sprint.

(c) T-Mobile. (d) Verizon.

Figure 7.4: IPv4 Coverage for increasing numbers of mobile vantage points. Coverage is displayed across multiple subnet aggregations of observed IP addresses. Markers denote the average coverage from all possible combinations of VPs, and error bars show the standard deviation of these sets.

the set of vantage points v in each operator, and for each set size $s = 1...v$, we take the average of the number of unique subnets seen across each combinations of VPs, $\binom{v}{s}$.

Figures 7.4 and 7.5 display the average coverage fraction for combinations of different vantage points of increasing size for IPv4 and IPv6 addresses. In Figure 7.4, we plot the coverage fraction for different subnet sizes, /24, /20 and /16, since subnet boundaries for GCs are not known ahead of time. For AT&T and T-Mobile, we see high levels of coverage even from the limited numbers of VPs available, reaching over 75% of /24 coverage in AT&T and over 90% /24 coverage in T-Mobile. Interestingly, Verizon coverage approaches a limit just above 40% for all subnet sizes. This is surprising especially since we had the largest



(a) T-Mobile. (b) Verizon.

Figure 7.5: IPv6 Coverage for increasing numbers of mobile vantage points. Coverage is displayed across multiple subnet aggregations of observed IP addresses. Markers denote the average coverage from all possible combinations of VPs, and error bars show the standard deviation of these sets.

number of vantage points in Verizon's network, 89, and the number of determined GCs roughly matched our expectations for a network of its size and deployment. The fact that all subnets approach the same limit lends us to believe that Verizon is partitioning its overall IPv4 space, and that significant portions of traffic are coming from addresses allocated to entities other than Verizon Wireless customers.

Figure 7.5 displays the same coverage analysis for IPv6 addresess. We display only T-Mobile and Verizon's networks since we lack sufficient IPv6 measurements for the remaining two operators. For T-Mobile, the significantly larger IP space for IPv6 exhibits lower overall coverage of /48 subnets, and a shallower growth curve for large prefix sizes. In contrast to its IPv4 addresses, VPs in Verizon approach 60% of coverage for /48 subnets, and close to 100% for /40 subnets.

We can also measure the marginal utility of additional mobile vantage points by analyzing the curvature of subnet coverage paths. The marginal utility of additional VPs in our instance depends a variety of factors, including the on the distribution of VPs across different GCs, the propensity of client assignment to distant GCs, and the mobility of individual devices

across network boundaries. The utility of additional vantage points also varies by operator, determined by the size of IP pools allocated to each GC.

These differences can be observed in Figure 7.4 by the knee in each of the coverage curves. For instance, T-Mobile (Fig. 7.4c), with only 68 /24 subnets and high observed rates of remote GC assignment, has a steep growth rate until it reaches approximately 20 VPs where it then decreases rapidly. AT&T in contrast (Fig. 7.4a) with 168 /24 subnets and a high consistency with /24 subnet assignment sees a near linear growth rate across all vantage points. This highlights the difficulty in obtaining complete mobile network coverage, since it depends largely on vantage point location and behavior, which is difficult to know a priori.

7.4 Tiller – An End-Host System for GBRS

We build upon these observations to design and implement TILLER, a mobile distributed measurement system. TILLER is a mobile distributed system based on our concept of gateway clusters for cellular network coverage. gateway-based replica selection for cellular networks. TILLER relies on instrumented mobile clients to discover key pieces of cellular network infrastructure through active probing. Using this infrastructure information along with device's assigned client IP addresses, TILLER then accurately clusters cell networks into GCs. In this section we present the deign and implementation of TILLER.

7.4.1 Tiller Architecture

TILLER's design builds on several features of modern cellular networks to accurately and efficiently characterize mobile networks. There are two main components in TILLER's design: (*i*) cellular network atlas globally aggregated and periodically distributed to TILLER clients, (*ii*) adaptive network probing which detects local context and adapts probing rate based on preexisting coverage.

Cellular Network Atlas

In this section we describe TILLER's internal representation of cellular networks, how it aggregates the views of multiple mobile clients into a global view, and how it clusters this view network GCs.

Each TILLER instance maintains a historical mapping of its current gateway routers and public IP addresses described in the previous section. TILLER represents this information in an undirected weighted graph $G = (V, E, w)$, where both client IP addresses and gateway routers are nodes, the relation between the two constitutes edges, with weights corresponding to the number of occurrences recorded by the client. TILLER maintains separate graphs for IPv4 and IPv6 addresses and partition identifiers.

To further reduce overhead, and enhance user privacy, we aggregate client IP addresses by /24 subnets for IPv4 addresses and /48 subnets for IPv6 addresses. For each mobile operator we investigated, we found no smaller allocations of subnets to GWP, so assume these aggregations sufficiently fine-grained for GWP identification.

TILLER utilizes these networks maps to (*i*) represent cellular network topology, (*ii*) determine device network locality, and (*iii*) monitor for network changes or reconfigurations.

Global Vantage Point. TILLER aggregates the views of all clients in a particular operator's network to create a global view of each operator's infrastructure. TILLER's *global network oracle* receives periodic updates from its clients in the form of an adjacency list of each client's weighted graph. These reports are combined into a single weighted graph, by combining the edge weights from each client's individual graphs.

A global view is necessary for full visibility into mobile operator networks, due to the isolation of each GWP, and the spatial and temporal locality of assignment. No single mobile vantage point could practically measure an entire cellular network spanning a large geographic areas such as the U.S.

IPv4 and IPv6 Aggregation. The deployment of dual-stack networks in cellular networks, which allow both IPv4 and IPv6 networks simultaneously for clients [56]. TILLER combines the subnet clusters at collocated GCs to better understand cellular topology. TILLER maintains a separate graph containing IPv4 and IPv6 addresses observed by devices, with weighted edges connecting co-occurrences of addresses. These are clustered using the same community detection algorithms as GCs, to obtain connected IPv4 and IPv6 GCs.

Combining the views of IPv4 and IPv6 networks is useful since often these networks reveal different information about their underlying infrastructure. For instance, as we observed with Verizon in the prior section, it appears that not all of their IPv4 address space is allocated to Verizon Wireless customers, yet their IPv6 addresses are. Combining these views can help derive coverage for these IPv4 addresses in this instance.

Adaptive Probing

TILLER periodically runs active probes from mobile devices to determine assigned partition and public client IP addresses. In order to ensure cellular connectivity, Tiller queries the device's operating system to determine its active network interface.

TILLER adapts both its measurement rate, as well as the types of probes used based on the availability of network information existent, and based on the results of initial exploratory probes. In cases where new network topology is discovered, TILLER performs additional probes. In addition to topology exploration, TILLER monitors cellular network expansion and reconfiguration. In addition to the presence of new IP space or gateway router addresses seen in network expansion, reconfigurations occur when IP addresses are reallocated to different GCs. These are detected when new links form between IP addresses and existing gateway routers.

The rate at which TILLER launches active probes is determined by utility of a device's existing network location. TILLER determines this network location in the following way:

1. Obtain device's IPv4 and IPv6 (if applicable) from echo server.
2. Compare each address to local network map:
    - If new IP address detected: perform traceroutes to popular content. Gateway routers are extracted from traceroute and added network map.
    - If new gateway router is detected (e.g. new GC): perform additional traceroutes for both IPv4 and IPv6 to additional destinations.
3. Perform community detection on network graph. If the number of or composition of clustered GCs differs from previous, inform global atlas.

7.4.2 Tiller Implementation

We implemented TILLER on the Android operating system. Since October 2014, Tiller has been run on over 1900 unique volunteer mobile devices, across 96 unique mobile operators and across 5 continents. TILLER clients performed network exploration measurements approximately every hour, contacting our IP echo services and performing outbound traceroutes to popular content destinations. We highlight the performance of TILLER from the over 260 TILLER clients in the four largest U.S. MNOs.

The operation of TILLER's network clustering can be seen in Figure 7.6, which displays the detected number of GWPs by TILLER's aggregate view since its launch in October 2014. New clusters appear over time as mobile clients are assigned to new gateways, discover new partition identifiers, or relocate (or join) to new cellular network regions. Over time, clusters may coalesce as shared IP space is discovered through continued measurement. These are shown as dips in the cumulative cluster count in the figure.

To illustrate the limitations of individual vantage points, we calculate the number of GWP clusters running locally on each device compared to the global aggregate view. Figure 7.7

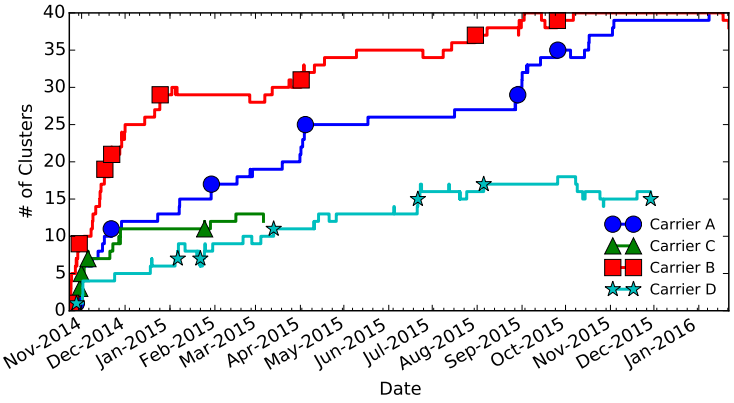


Figure 7.6: Tiller detected GCs for the 4 major U.S. operators over time. Spikes in number of detected clusters are due to recruitment efforts of our mobile system, allowing Tiller to discover new, previously unknown partitions. Dips represent coalescing of clusters when from Tiller's ongoing community detection algorithms.

illustrates the number of detected GWPs against the average number of detected clusters for individual vantage points.

As seen in the figure, individual vantage points are able to detect multiple GWP clusters, likely due to user mobility, physically relocating into different network regions, as well as changes to operator assignment. While users average close to 5 GWPs, we observed several clients with over 10 detected clusters. Yet this is still well below the aggregate total of nearly 40 PGW clusters observed from all vantage points.

7.5 Summary and Contributions

In this chapter, we made the following contributions:

- We introduced a new abstraction for cellular topology based on network PGWs and network IP space allocated to each, the Gateway Cluster (GC).
- We objectively defined topology coverage for cellular networks based on the fraction network GCs covered by mobile vantage points, and approximated this coverage by the IP space observed from vantage points.

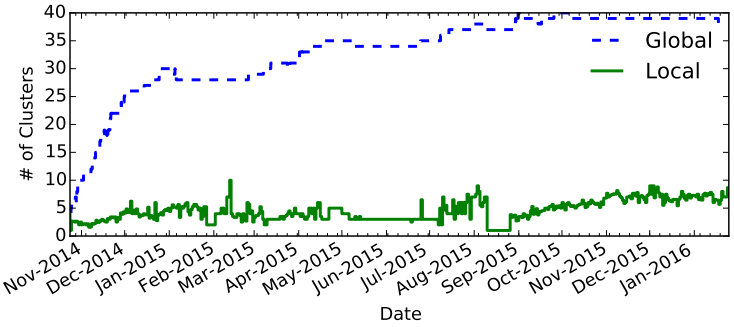


Figure 7.7: Number of GCs clusters detected by TILLER in its global vantage point (blue) versus each individual vantage point (green) over time a large U.S. carrier. While individual vantage points can detect more than a single cluster due to user mobility and operator assignment, an aggregate view allows much greater visibility into network infrastructure.

- We derived a *baseline coverage* of cellular IP space using BGP announcements and request traffic from a large content delivery network.
- Using this new metric of we presented coverage analysis from 2.5 years of longitudinal mobile measurements.
- We presented the design and implementation of TILLER, a mobile distributed measurement system which uses adaptive probing to efficiently characterize and monitor cellular networks.

Chapter 8

Trace-based Clustering of Cellular End-Users

In this chapter, I examine clustering cellular end-users entirely from external vantage points. I first present assess the feasibility of external characterization from multiple, longitudinal, traceroute campaigns towards 9 large mobile network operators in the United States and Brazil. Building off this initial investigation, I present the design and implementation of MACHETE , a system for characterizing cellular networks from external vantage points. MACHETE uses traceroutes toward cellular end-users to cluster cellular address space to network PGWs. MACHETE is currently deployed by a large content delivery network to identify and locate groups of cellular end-uesrs. In the final part of this chapter, I discuss some of the challenges encountered during MACHETE's global deployment.

8.1 Scaling Cellular Network Exploration

Network services, such as content delivery networks, have invested heavily in both hardware and software infrastructure for monitoring Internet conditions. Unfortunately, as has been expressed multiple times throughout this dissertation (e.g Chp. 3), systems which rely on common tools such as `ping` and `traceroute` are ineffective in light of the opacity of modern cellular networks.

To overcome this, several approaches have attempted to utilize instrumented mobile handsets to identify cellular clients' PGWs [98, 117, 120], as well as TILLER presented in the previous Chapter. While these approaches have shown to accurately identify mobile client's

PGWs (see § 4.2), they suffer from coverage and scalability restrictions due to challenges in recruiting, retaining and positioning mobile vantage points.

Internet-scale services require full coverage and reliable measurement vantage points, yet neither of these can be practically achieved through instrumented handsets. The dynamic management of cellular network resources means that handsets cannot guarantee which PGW they are assigned, therefore attempts at full and continuous coverage would need to over provision mobile vantage points across all global MNOs.

This chapter investigates techniques to perform cellular network characterizations – PGW discovery and mobile client assignments – from *external* vantage points. Towards this goal, it presents the results from a 11 month traceroute study of 9 large cellular operators: the 4 largest U.S. MNOs and the 5 largest Brazilian MNOs. Through our experiments we conducted hourly traceroutes toward over 100,000 cellular client targets over an 11 month period, collecting over 2.2 billion traceroutes.

I find that traceroutes from external vantage points are sufficient to detect cellular PGWs and map cellular client IPs to their assigned PGW. We find that in many operators, inbound traces are able to reach nearly to client's PGWs. Our findings also show these PGW mappings to be stable on the order of months in a majority of cases.

Building on these preliminary findings, I present the design and implementation of MACHETE , a distributed system for cellular network characterization. MACHETE is designed for CDNs and other Internet-scale services, and allows a reliable measurement system capable of global coverage of cellular networks. MACHETE operates by conducting, and processing large numbers of traceroutes directed towards cellular IP addresses to create clusters of cellular end-users by their assigned PGW. Using ground truth information from two large MNOs in the U.S., MACHETE is able to successfully map client IP addresses to their assigned PGW instance with 95% accuracy.

This chapter contains the following contributions:

- An 11 month study tracing large fractions of cellular end-users in 9 large cellular networks across 2 continents.
  - Analysis of the external reachability of cellular clients.
  - Path symmetry between clients and external vantage points.
  - Temporal stability of paths to cellular client addresses.
- Novel trace-based clustering of cellular end-users that achieves over 95% accuracy.
- Presents the design and implementation details of MACHETE, and discusses the experiences from its global deployment with a large content delivery network.

8.2 Data Collection

In this section we describe the dataset and methodology of our 11 months of measurements periodically tracing towards cellular clients from a large number of distributed vantage points. We profile the 4 largest U.S. and 5 largest Brazilian MNOs, collecting a total of over 2.2 billion traceroutes towards 91,000 IPv4 and 8,600 IPv6 cellular targets. These carriers were chosen for their large subscriber base and user demand, as well as their large geographic coverage area. Table 8.1 lists the target ASN used for each mobile operator. While MNOs may use multiple ASNs for client IP addresses, we discovered that in the operators investigated, the vast majority of IP addresses were housed within a single ASN. We used only this primary ASN for our target analysis for simplicity.

We selected targets using the the request logs of a large content delivery network. For each cellular ASN investigated, we selected a single IP address for every unique /24 subnet for IPv4 and /48 subnet for IPv6 from the CDN's logs. The scale of the CDN's deployment ensures that we are selecting the addresses of active cellular clients. Targets were selected for the U.S. MNOs using log data from July 2015, targets for Brazilian operators were selected

Operator Name	Country	Primary ASN	No. /24s	No. /48s
AT&T	US	20057	2750	186
Sprint	US	3651	7917	1650
T-Mobile	US	21928	303	3532
Verizon Wireless	US	22394	2292	3297
Claro	BR	22085	1352	0
Tim Cellular	BR	26615	26594	4
Vivo	BR	26599	42845	15
Oi	BR	8167	2950	1
Nextel	BR	53037	4217	0

Table 8.1: Mobile network operators used for our study.

using logs from December 2015. The number of subnets targeted for each operator are summarized in Table 8.1.

Data collection occurred in two separate phases. The first phase traced U.S. MNOs from a set of 76 vantage points geographically distributed within the U.S., with each vantage point launching a traceroute toward each target approximately every hour. These measurements were performed between July 2015 and December 2015. The second phase traced toward Brazilian MNOs from a set of 115 globally distributed vantage points, half within Brazil and half outside, with probes launched towards approximately once every three hours from each vantage point. These measurements were collected between January and June 2016. In all we performed 2.29 *billion* traceroutes, 614 million to U.S. cellular clients and 1.67 billion to Brazilian cellular clients.

8.3 Path Characterization of Cellular Networks

The opacity of cellular network has been well established in this (Chp. 3) and prior work [117, 120], the extent of the reach, visibility and symmetry of paths to cellular clients has yet to be fully understood. In this section we present results of our trace-based exploration of a select group of 9 large cellular networks in the U.S. and Brazil. Specifically we look into (*i*) the reachability of cellular clients, (*ii*) the temporal stability of paths to cellular clients and (*iii*) the path symmetry between mobile clients and content providers.

8.3.1 Traceroute Characterization

We now investigate the traceroute paths toward cellular clients and cellular IP space.

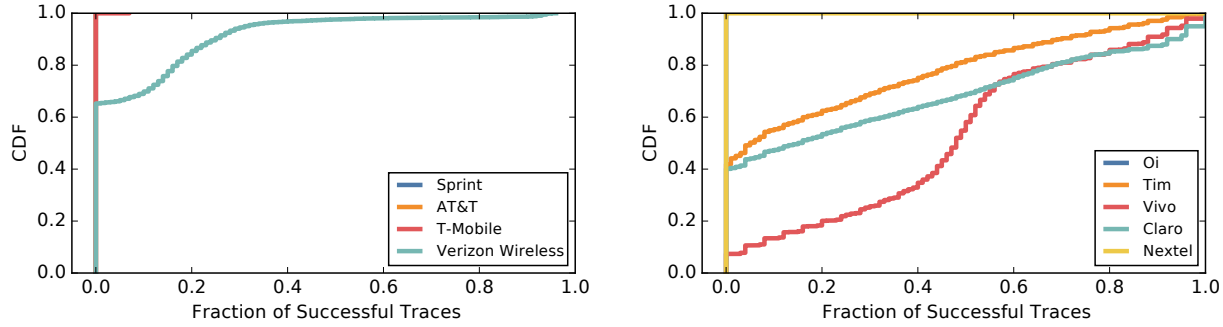
Reachability of Cellular IP Addresses

In this section we look at the reachability of cellular IP addresses. We define reachability to be the fraction of traceroutes which reach their destination, that is, the final hop in the traceroute is the destination IP address. While it is widely assumed that traces towards cellular users terminate at, or well before, client PGWs [95, 117], our analysis revealed that cellular IP reachability depends on individual operator policies. Operators either completely restrict client access, or those provide partially reachable clients.

From the entire set of traces, we calculated each target IP address's reachability. The distribution of each cellular operator's IP address reachability is shown in Figure 8.1. The figure highlights the differences in reachability across MNOs, with many networks completely prohibiting client access all of their addresses, and others allowing traceroute access for a certain fraction of addresses with varying frequency.

For U.S. MNOs (Fig. 8.1a), three out of four operators have practically all their IP addresses with zero reachability. Verizon is the only U.S. carrier to have significant reachable addresses, with nearly 30% of addresses successful at least part of the time. These results are roughly consistent with the prior work looking at these same, which showed largely opaque cellular networks.

Brazilian MNOs (Fig. 8.1b), on the other hand, showed much more accessible networks. While Nextel and Oi disallowed all client probes, the other three operators showed much greater fractions of reachable clients – traces reached 60% of Tim Cellular's and Claro's clients, and nearly 95% of clients in Vivo's network. We validated that these traces were indeed reaching cellular targets by looking at the final hop latencies for these successful traces. In these cases, last hop latencies ranged from several hundreds of milliseconds to



(a) U.S. MNOs

(b) Brazilian MNOs.

Figure 8.1: Fraction of traceroutes which reach their targets in U.S. and Brazilian MNOs. While the addresses of many mobile operators are entirely unreachable, other operators allow reachable cellular addresses with varying frequency.

several seconds in the extreme cases, leading us to conclude that these traces were successfully reaching cellular connected devices.

While overall the fraction of successful traces followed a diurnal pattern, further investigation into specific IP addresses yielded no discernible patterns of access. It is unknown why certain addresses are partially reachable to begin with, or what causes the temporal differences in reachability are unknown. In the end we find these access of individual addresses are largely bursty and random.

To illustrate this random behavior, we selected two partially reachable IP addresses at random from Verizon Wireless, and plotted the number of successful traces every hour for a 30 day period, shown in Figure 8.2. The figure shows the seemingly random patterns of access for individual IP addresses. This leads us to hypothesize that reachability is a combination of provider policies allowing reachability, and mobile device behavior.

Path Symmetry between Cellular Client and Servers

We investigated the path symmetry between mobile clients and our trace servers to ensure that server traces were able to capture the network locations of cellular IP addresses, and are directed towards the correct PGW region and egress location . Using over 200 ALICE

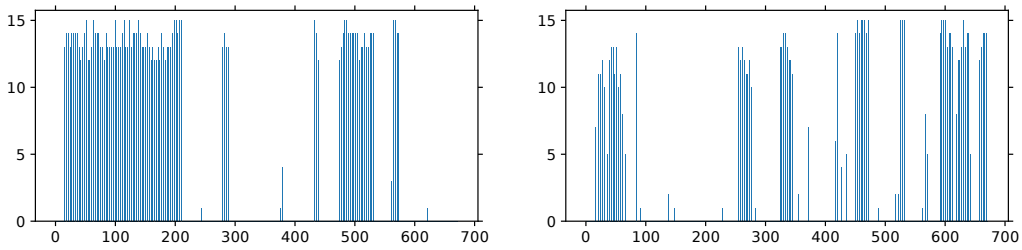


Figure 8.2: Histogram of successful traces per day for two random partially reachable Verizon Wireless addresses. Patterns of reachability appear random for individual IP addresses though overall patterns exhibit a minor diurnal pattern.

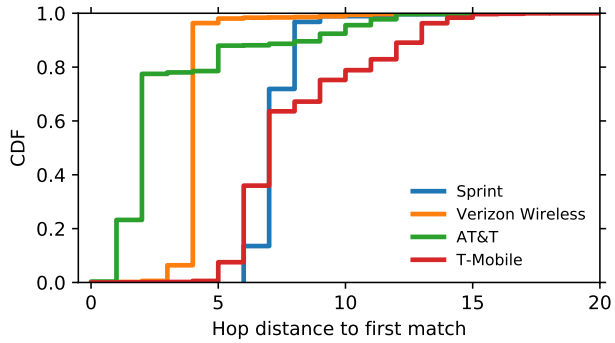


Figure 8.3: Distance between client and server traces, matched on /28 subnets.

clients in the U.S., we performed traceroutes towards selected content servers involved in tracing mobile clients from each instrumented handset. We then compared the paths from mobile clients toward content servers, matching cellular client IP addresses by /24 subnet, pairing with server traces performed in the matching day and hour.

While overall symmetry is important, we look at the visibility difference between outbound mobile client traces and inbound server traces. We wish to see how close to cellular infrastructure server traces are able to penetrate. We count the number of public IP addresses that exist in the mobile trace of symmetric traceroutes to approximate lost visibility from server traces.

Figure 8.3 displays the cumulative distribution of hop distances for the four U.S. MNOs. The figure displays the different level of visibility in each network across the four operators, with AT&T allowing the most visibility, followed by Verizon, and then Sprint and T-Mobile.

While AT&T and Verizon have moderate losses in visibility, 2 and 4 public hops respectively, Sprint averages 7 public IP hops and T-Mobile averages a distance of nearly 9 public IP hops.

The tails of the distribution hint at larger path asymmetries. Further investigation into T-Mobile found that outbound packets are actually routed through a nationwide private transit network before exiting to the Internet, meaning outbound packets may leave from a number of different PoPs, depending on the destination AS. This may explain the large discrepancy between outbound and inbound paths.

8.3.2 Representing Traces through Sink-Vectors

In light of the limited reachability of cellular IP space, as well as the temporally instability of reachable targets, we propose representing cellular IP addresses based on the termination point of traceroute probes.

As shown in the previous section, traces to cellular targets often fail to reach their destination in some or nearly all cases depending on the operator. In these cases, traces terminate at arbitrary routers along the path, which change based on when and where traces are launched. Often these routers are in different ASes than the destination, making it difficult to know whether a trace should be filtered out or can provide useful information.

We find that the combined termination points of these traces can act as an identifying set of coordinates for cellular IPs within an operator. From the set of traces toward a particular IP address, we represent each IP address by its *sink vector*, that is the set of, and frequency of terminating routers of traceroutes. For instance, a cellular target, t, can be represented by the vector of terminating routers, r_i, and their relative frequencies, f_i.

$$t = \{r_0 \rightarrow 0.75, r_1 \rightarrow 0.2, r_3 \rightarrow 0.05\} \tag{8.1}$$

We can use these trace vectors to calculate various statistics between IP addresses. We can compute the similarity between these vectors using vector distances such as Cosine

Operator	In/Out Similarity
AT&T	0.893
Sprint	0.926
T-Mobile	0.004
Verizon Wireless	0.775

Table 8.2: Cosine similarity between trace vectors created from vantage points located inside target operator's networks, and those sent from external vantage points. All U.S. MNOs except for T-Mobile display high degrees of similarity between internal and external vantage points.

Similarity.

$$cos\_sim = \frac{A \cdot B}{\|A\|\|B\|}$$

We now use our newly defined trace vectors to measure characteristics of cellular IP addresses and cellular IP space. Specifically, we look at (i) the visibility differences for different vantage points, and (ii) the temporal stability of cellular IP addresses. The former is important for selecting vantage points for cellular network exploration and characterization. The latter is important for understanding the stability and lifespan of cellular IP address assignments to cellular gateways.

Differences in Vantage Point Visibility

We now investigate the differences in vantage point visibility into cellular networks. Vantage point selection is an important component of any active Internet measurement system. For our purposes, we seek vantage points with the furthest visibility into cellular networks, meaning the highest probability of observing PGWs (or nearby routers) of cellular networks. In this context, visibility refers to the termination point of cellular inbound traceroutes, and their distance from cellular end-hosts.

We analyze the differences between different vantage point traces by looking at vantage points positioned *within* mobile operator's networks. The cooperation of the large CDN, and access to CDN replica servers as tracing vantage points, many of our traces were able to

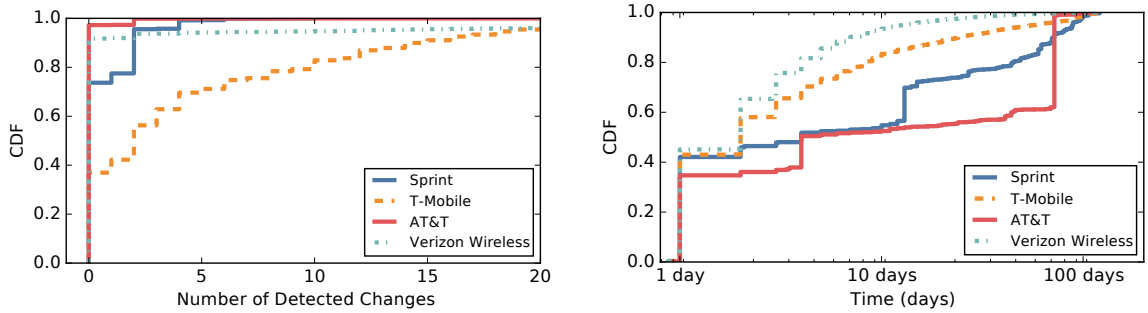
obtain vantage points within cellular operator networks. We consider vantage points to be in-network when the VP is provisioned within an MNO's facility, resides in the same ASN as the MNO or its parent provider. This relationship information is kept by the CDN. We assume for this portion, that VPs within an MNOs network will *always* have the greatest visibility.

To compare the visibility of in-network VPs to out-of-network VPs, we construct a trace vector from each set of in-network VPs, sv_i, and each set of out-of-network VPs, sv_o, for each target IP address. We calculate the Cosine Similarity of each vector to determine the trace similarity between the two vantage point sets. Table 8.2 displays the average cosine similarity between trace vectors from in-network and out-of-network vantage points.

All U.S. MNOs except for T-Mobile display high degrees of similarity between internal and external vantage points. For these three networks, the value of having in-network VPs is minimal compared to their out-of-network counterparts. T-Mobile, on the other hand, has entirely different visibility from its in-network VPs than out-of-network VPs. This indicates that out-of-network VPs have much less visibility into its network. Further investigation revealed this to be the case, with T-Mobile routing Internet traffic to a small number of Points of Presence (PoPs), less than their total number of PGWs, and then routing traffic internally to each client's PGW. The end result of this is that traces from out-of-network VPs see no further than these PoPs, which often have little locality to a client's PGW.

Temporal Stability of Vectors

In order to understand how stable traces to cellular IP addresses are, we calculated the sink-vectors for each day of traces between August 1, 2015 and December 1, 2015. This allows us to track the transience of cellular IP addresses, and can be used to detect changes in PGW assignment for cellular IP addresses. In addition its usefulness in furthering our understanding of cellular networks in general, determining the rate of change of cellular



(a) Change count for U.S. operators.

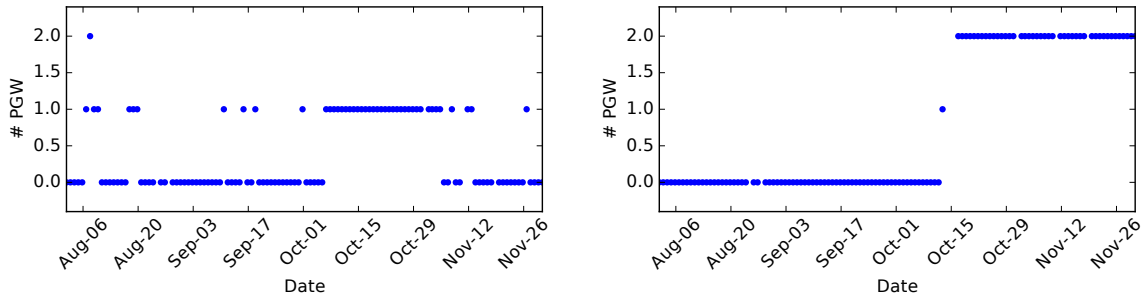
(b) Time between detected changes.

Figure 8.4: Distribution of detected sink-vector changes per IP address for U.S. operators. A significant change was detected when the cosine similarity between consecutive days was less than 0.5.

targets helps to set measurement frequency for cellular topology measurement. For each trace target, we calculated the cosine similarity between each day d_n and the prior day d_{n-1}. For cases where the Cosine Similarity between consecutive days was less than 0.5, we mark that as a change in trace vectors.

Figure 8.4a displays the distribution of detected changes per IP address grouped by operator. The figure shows that for Verizon and AT&T, trace vectors remained highly stationary, with no detected changes for most of its IP addresses during the 4 month period. With Sprint, we see that around 25% of addresses experienced between 1 and 4 path changes during the measurement period. While T-Mobile appears to exhibit significant changes, both in the number of transient addresses and the number of changes, however keep in mind the internal routing inherent to T-Mobile's network.

For each trace vector change, we calculated the time between each change to see how quickly each path change occurred. Figure 8.4b plots the time between trace vector changes for each U.S. MNO. We find trace vectors exhibit two types of change behavior: transient and stable changes. Transient changes due to the trace-vector's sensitivity to traceroutes which intermittently reach their destination. We found that nearly all cases where with a change time of < 10 days was due to these types of changes. Figure 8.5a displays an example



(a) Oscillating path change (Verizon).

(b) Stable path change (AT&T).

Figure 8.5: Two types of path changes observed. The oscillatory changes switch between two main states, and in this case reflect changes in path reachability of the cellular client. The stable path change indicates a reassignment of an IP address to another PGW.

case of this behavior. Stable changes occur when operators reassign IP addresses to different PGW locations. These changes often occur over larger time periods. Figure 8.5b displays this type of behavior.

8.4 Clustering Approach

In this section we outline our approach for clustering cellular end-users using the trace vectors described in the previous section. The goal of our clustering is to group of end-users together in a way which accurately represents similar network location and performance, and reduces the complexity of the system. Cellular networks by their design already partition users in a way which largely determines network location by PGW allocation. Our clustering attempts to determine the number of PGW instances within each MNO and the allocation of IP addresses to each. PGW clustering faces several challenges, including determining an unknown number of PGW instances of unknown size and location.

8.4.1 Similarity of Trace Vectors

We analyze the similarity of each IP's trace vector to the entirety of an MNO's targets. If each vector is entirely unique (e.g a cosine similarity or 0), then trace vectors would provide

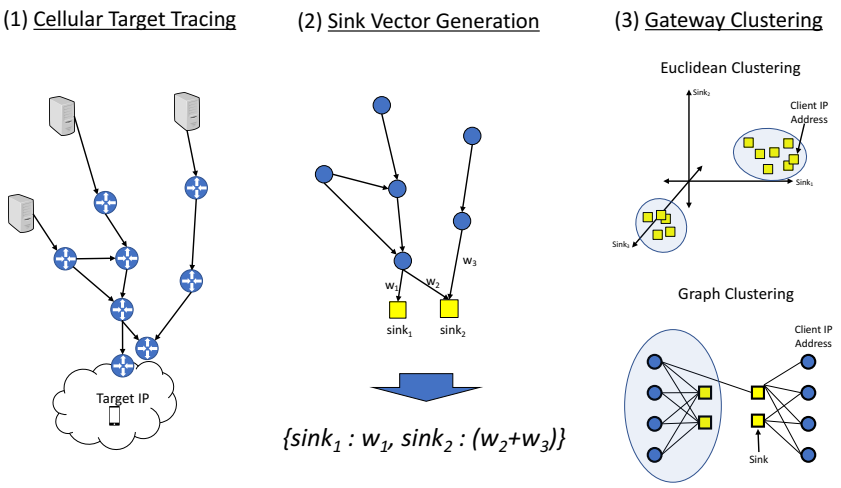
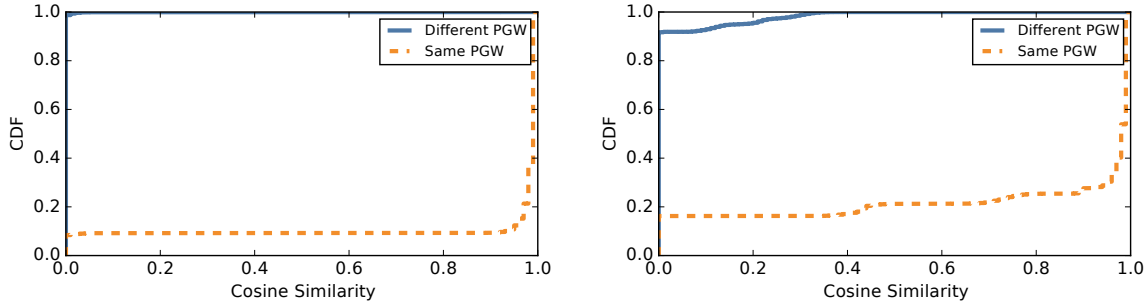


Figure 8.6: Trace-based clustering involves three steps: (*i*) traceroutes to cellular IP addresses, (*ii*) generation of trace vectors for each target, and (*iii*) clustering through either Euclidean or graph-based methods.

little information to group similar clients together. On the other hand, if all vectors are similar, then there will be difficulty clustering IP addresses into distinct partitions.

To estimate the effectiveness of client trace vectors, we calculated the cosine similarity between trace vectors of all pairs of targeted subnets within each operators. To illustrate the use of Cosine Similarities between trace vectors, and its potential usefulness for clustering, we calculated the similarities between trace vectors for IP addresses assigned the same PGW, and those assigned to different PGWs. This PGW membership information was taken from ground truth data obtained from Sprint and AT&T.

Figure 8.7 plots the cumulative distribution of similarity values of those trace vectors assigned the same PGW, and those assigned to different PGWs, for AT&T (Fig. 8.7a) and Sprint (Fig. 8.7b). For both operators, the figure displays the high Cosine Similarities between those trace vectors for IPs in the same PGW – nearly 1 for 60-80% of IPs – as well as the low similarity between trace vectors for IPs in different PGWs. Over 90% of different PGW similarities were 0 in Sprint, and nearly 100% of different PGW similarities in AT&T were 0. These hyperbolic similarity values indicate the utility of Cosine Similarities and



(a) AT&T. (b) Sprint.

Figure 8.7: Cosine Similarities for trace vectors for IP addresses within the same PGW, and those in separate PGWs. High Cosine Similarity corresponds to PGW membership, making it a useful distance metric for clustering.

trace vectors for inferring same PGW membership. In the following section we utilize this fact to cluster cellular IP addresses by assigned PGW.

8.4.2 Clustering Methods

For clustering, we place each cellular IP address in a high dimensional space, with each dimension representing one of the set of all routers existent in trace vectors for all probed IP addresses within an operator. We evaluated several Euclidean clustering methods, including K-Means [15] of various cluster sizes, MeanShift [37], and DBScan [46]. Since K-Means requires a specified number of clusters, we selected 32, 64 and 128 for cluster sizes for U.S. MNOs, and 8 and 16 for Brazilian MNOs.

In addition to these Euclidean clustering methods, we also experimented with graph-based community detection for clustering. From the set of trace vectors for an operator, we create a bipartite graph consisting of cellular IP addresses on one side and the set of sink routers on the other. The edges of the graph are weighted by their frequency from each IP's trace vector. We utilized a common community detection approach introduced by Clauset et al. [36] which maximizes the modularity of potential clustering within the graph.

Clustering Results

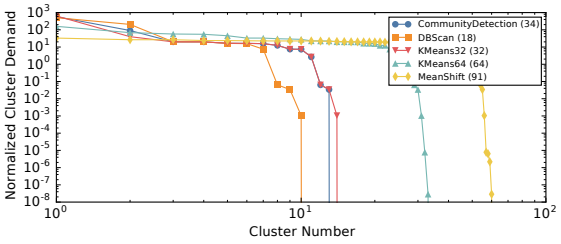
In this section we present the results of our clustering of 9 different cellular networks in the U.S. and Brazil. We present the results of our clustering attempts of cellular IP addresses, utilizing different methods of Euclidean and graph clustering.

We plot the results for all 4 U.S. and 5 Brazilian MNOs in Figure 8.8. In the figure, each x-axis item represents a determined cluster, the y-axis represents the CDN's traffic demand contained within that cluster. This demand is represented as a normalized fraction of the CDN's entire traffic volume. The cluster demand helps add context to the clusters generated by each algorithm, since other metrics such as the number of subnets little value due to the great inequality of demand across IP addresses in cellular networks (e.g. small numbers of IP addresses contain the vast majority of demand due to NATting).

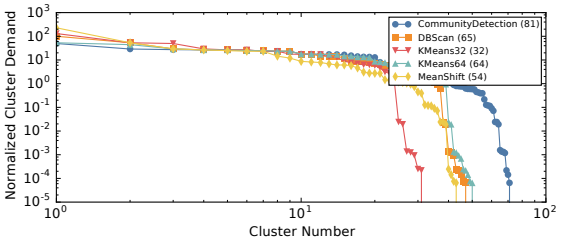
Comparing between the two countries, we unsurprisingly find U.S. MNOs have significantly larger numbers of clusters than the Brazilian MNOs. Much of this is due to the larger geographic areas covered by U.S. MNOs, the deployment of dual stack networks which create duplicate clusters for IPv4 and IPv6 addresses at the same location, and finally the penetration of LTE networks which are typically accompanied by greater numbers of PGWs.

8.4.3 Ground-Truth Evaluation

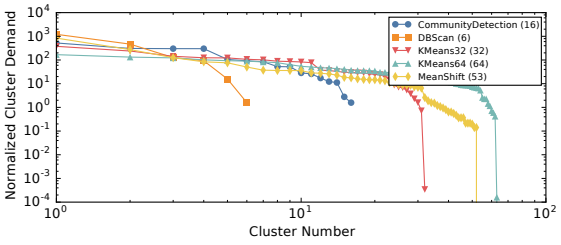
In this section we present our evaluation of the proposed clustering algorithms, using ground truth from two large U.S. MNOs, Sprint and AT&T. The output from our approach is a set of clusters, which contain membership information for cellular IP addresses which we believe are assigned to the same PGW. This set, however, is unlabeled since we do not attempt to determine the location of client's PGWs from the clustering. To evaluate, we use the F1 score, a common metric for measuring classifier accuracy. We generate labels for our clusters based on the majority ground truth label within each cluster, which can be used by the F1 metric.



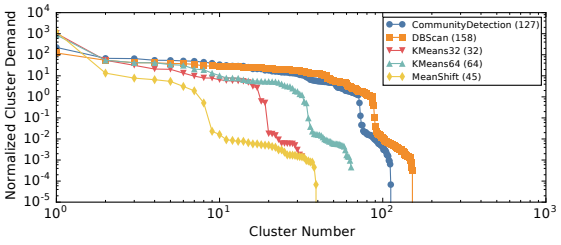
(a) AT&T.



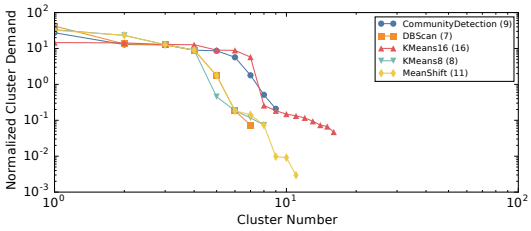
(b) Sprint.



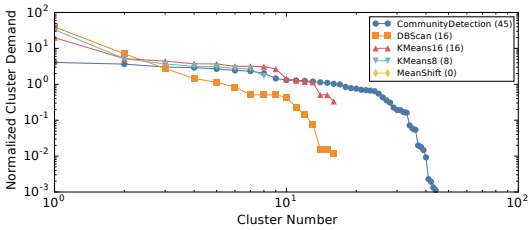
(c) T-Mobile.



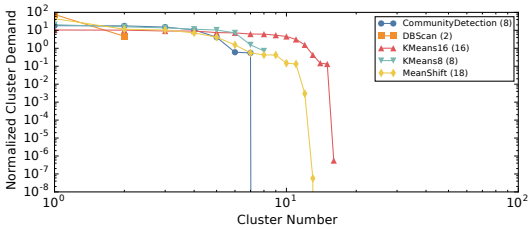
(d) Verizon Wireless.



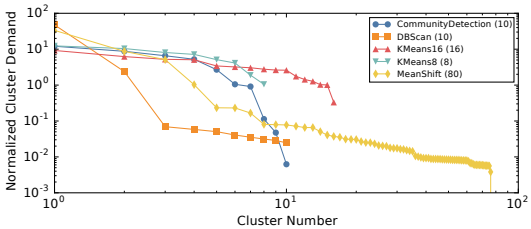
(e) Claro.



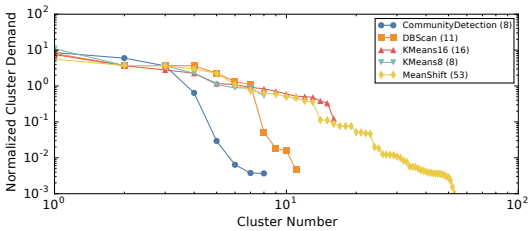
(f) Vivo.



(g) Tim.



(h) Oi.



(i) Nextel.

Figure 8.8: Results of multiple clustering algorithms across each MNO. Each point represents the normalized traffic demand of each cluster, as seen from the CDN, in descending cluster size.

Algorithm	AT&T	Sprint
DBScan	0.602	0.895
MeanShift	0.929	0.596
K-Means (32)	0.839	0.585
K-Means (64)	0.873	0.917
K-Means (128)	**0.929**	0.975
Community Detection	0.843	**0.985**

Table 8.3: F1-scores from labeled set of clusters.

$$F1 = \frac{precision * recall}{precision + recall} \tag{8.2}$$

Table 8.3 displays the calculated F1 scores of each clustering algorithm for the two U.S. operators. The table reveals that when viewed with calculated labels, our clustering is able to achieve very high classification accuracy, with the largest scores reaching 0.929 and 0.985 F1 scores for AT&T and Sprint respectively.

These results reveal that our clustering approach can determine with a high accuracy the assignment of cellular clients' PGWs from external trace data alone. Building off of these results, we designed and implemented a system for the automatic data collection and clustering of cellular clients at a global scale, MACHETE . In the following section we discuss the design of MACHETE and our experiences deploying on a large CDN.

8.5 MACHETE

We now present the design of MACHETE , our system for trace based clustering of cellular end-users. At a high-level, MACHETE contains 5 main components, used for selecting, tracing and clustering cellular end-users.

- **Cellular Target Selector.** Processes CDN request logs to obtain a set of cellular IP addresses, that is, IP addresses for end-hosts traversing cellular access links.

- **Vantage Point Selector.** Selects the set of trace vantage points which are determined to have the maximum visibility into each target network.
- **Trace Scheduler.** Dispatches trace assignments to selected vantage points. Trace frequency is set per network, and is determined by the temporal stability of each network operator.
- **Trace ingest.** Aggregates, and pre-processes completed traces. Results are stored in a single location HDFS cluster for later processing.
- **Gateway Clustering.** Clusters network IP addresses into groups corresponding to clients' assigned PGW instances.

Cellular Target Selector

The Cellular Target Selector utilizes the log from a large CDN, specifically those from the CDN's Real User Monitoring (RUM) system to determine the set of IP addresses which are cellular. In order to find which addresses traverse cellular access links, we utilize the Network Information API [1], a Javascript API which reveals the last-mile connectivity of the end-user. These logs are aggregated by /24 subnet for IPv4 addresses and /48 subnet for IPv6 addresses, common address aggregation levels which have been shown to exhibit common network properties [19, 50].

Cellular addresses are derived from a series of heuristics related to the ratio of "cellular" to "non-cellular" hits obtained within the logs. The derivation of these heuristics is outside the scope of this dissertation, but internal studies of this methodology have been shown highly accurate. From this set derived cellular subnets, we select a single *active* IP address. An address is active if it has completed an HTTP request during the desired time period. For our purposes, we selected addresses active in the prior month.

Vantage Point Selection

In order to ensure the greatest visibility of cellular networks, we select a custom set of vantage points for each cellular ASN. These tracing vantage points double as CDN replica servers, therefore many are deployed deeply into operator networks. As we showed previously in Section 8.3.2, certain vantage points, especially those within cellular operator's networks, can have significantly greater visibility towards cellular end-users.

For each ASN, we select 20 (40 for the U.S.) geographically distributed vantage points from this set, using the following criteria for selection.

1. **Network Locality.** The CDN MACHETE is deployed on has many relationships with network operators. We preference tracing regions which are either located within cellular operator's networks, or have prearranged network partnerships with the CDN.

2. **Geographic Locality (Country).** Aside from network locality, we place the highest preference on vantage points in the same country as the target cellular ASN.

3. **Geographic Locality (Continent).** If not enough vantage points are available in the target's country, we relax the geographic restrictions to the contintent level, ordering by geographic distance to the target country.

Trace Scheduling

To handle the large numbers of traceroutes to cellular end-users, the Trace Scheduler stages tracesroutes towards each cellular network, and dispatches individual trace jobs to the appropriate vantage points. A key part of trace scheduling is to determine the frequency of tracing, over both near-term and long-term time scales.

For near-term time scales, we found that tracing a single target 3 times from each assigned vantage point, approximately 8 hours apart struck a balance between trace quantity and temporal coverage. For long-term time scales, we adjusted the periodic rate of tracing based

on the average time between IP gateway reassignments (§ 8.3.2). Given that the majority of IP addresses were stable during our 5 months of tracing, we found tracing each operator once per month is more than sufficient.

Trace Ingest

Completed traceroutes from vantage points are sent to the Trace Ingest process for data storage. The Trace Ingest process receives the successful traceroute messages from tracing vantage points, batches them on its local disk, and periodically uploads the information to a large HDFS cluster for storage and later processing.

Gateway Clustering

The Gateway Clustering performs procedure outlined in the previous section (§ 8.4). We found that the graph-based community detection performed the best when balancing the accuracy and speed of clustering algorithms. More importantly, our experience found that the ability of community detection algorithms to determine the appropriate number of clusters greatly exceeded that of similar Euclidean clustering algorithms, namely DBScan and MeanShift. We found these algorithms frequently either over-fit, vastly overestimating the numbers of clusters, or under-fit, creating a low number of very large clusters. While prior work has attempted to position Internet users in Euclidean space [87], we leave work on alternative projections for clustering to future work.

8.6 System Implementation

MACHETE is implemented through the existing system of a large CDN, and was written in a combination of C++ for existing software modules, and Python scripts for map reduce and data processing. MACHETE has been deployed on a live CDN since January 2017. The clusters generated by MACHETE are fed into the existing CDN's request routing system to improve the quality of cellular network request routing. MACHETE is currently used to characterize and cluster cellular infrastructure for over 660 global mobile network operators.

8.6.1 Deployment Experiences

Throughout its deployment, we have encountered several obstacles related to the characterization of global cellular networks. At a high-level these relate to overcoming the opacity of cellular networks, and the great variety of network configurations and operation policies. Below, we highlight some of the most valuable lessons learned through our deployment across the global cellular infrastructure.

Internal network routing.

Many cellular networks choose to route user traffic using their own networks. While can lead to greater control over client traffic, we find that it extends the opaqueness of cellular networks from external visibility. In large networks (such as T-Mobile), we find that traces often route to the nearest point-of-presence for that particular MNO rather than near the PGW. Internal routing appears in our data with trace vectors from different vantage points having very dissimilar values, as traces are often routed to the nearest PoP for a particular cellular network and then disappear into that operator's internal network.

We developed a simple heuristic for detecting internal routing called the *VP Similarity*. We calculate the VP Similarity by creating a individual trace vector for every VP and IP address pair, and calculate the cross product of Cosine Similarities across. We then average these values. Networks which internally route traffic typically have a very low VP Similarity value of less than 0.2.

Figure 8.9 displays the cumulative distribution of VP Similarity values for all 660 global MNOs currently characterized by MACHETE. We determined that operators with a VP Similarity less than 0.2 have a combination of significant internal routing and geographic diversity where our trace-based clustering is ineffective. In these instances, we have worked with mobile operators to find additional topology information as well as incorporated alternative measurement information, such as passive network statistics, to try and cluster

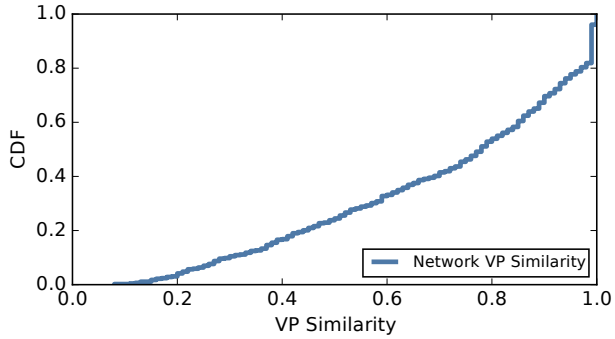


Figure 8.9: VP Similarity for the 660 global MNOs currently tracked by MACHETE. In our experiments, we found that networks with region similarity below 0.2 have problematic internal routing for our trace-based clustering.

cellular users. Development of these additional data sources is ongoing, and combining it with our trace-based approach is part of future work.

Varying reachability of cellular clients.

Our original approach was based on the observed opacity of U.S. cellular networks. However, in cases where operators allow trace access to clients forced us to modify our original algorithm to detect this. The challenge with reachable cellular IP addresses in our methodology is that our method assumes that trace termination points are near PGWs. Traces which reach their destination even some of the time have sinks (themselves) which are not shared by any other target. One of the consequences of this in our clustering is an increase of singleton clusters.

To combat this, we introduced an extra preprocessing step to the traceroutes before sending them for clustering. We first compile a list of common sink routers from the sets of traces terminating before their target IP (e.g. at or near network PGWs). We truncate traces which reach their destination at the last router which is contained within this common sink router set. This heuristic works well for operators with smaller fractions of reachable IP addresses (e.g. Verizon), however, as seen in this chapter, certain mobile operators

8.7 Summary and Contributions

In this chapter we presented our characterization of 9 large cellular networks in the U.S. and Brazil. Our trace analysis revealed the stability of cellular IP addresses to PGWs, and the distinct groupings of IP trace vectors motivate the trace-based clustering of cellular networks. We introduce MACHETE , a scalable solution for gateway-based replica selection. MACHETE works by clustering PGW IP addresses by trace vectors. We showed that MACHETE is able to accurately characterize cellular networks, and do so at a global scale.

Chapter 9

Contributions and Conclusion

This dissertation argued for the centrality of packet gateways (PGWs) in cellular networks. PGWs play a significant role in all cellular network functions, ranging from network ingress/egress to client billing, to network localization, as well as being the closest point of access for network services and content delivery networks. This dissertation's view is that by characterizing only these PGWs, one is able to capture all the necessary information about these networks to functionally incorporate them into existing Internet models.

From an in depth characterization of next generation cellular networks (NGCNs), we observed that network PGWs play a role in all aspects of cellular network's architecture and client performance, and affects how they should be measured and characterized.

We developed a novel method of representing cellular networks as gateway clusters (GCs), which partition cellular network IP space across their assigned PGW instances. We utilize these gateway clusters to create a new definition of topological network coverage for cellular networks.

We designed and implemented two systems for cellular network characterization, TILLER and MACHETE, which discover and partition cellular networks into their respective gateway clusters. We showed through live deployments of each system their effectiveness and accuracy.

9.1 Summary and Contributions

The contributions of this dissertation are as follows:

- We successfully argued for the centrality of cellular network packet gateways, showing that they impact all aspects of cellular networks' architecture, measurements and client performance.

- We showed that existing approaches for client localization are ineffective for cellular networks, and that this causes suboptimal replica server selection for CDNs.

- We presented a tool for mobile end-hosts to measure and characterize next generation cellular networks. We developed a mobile experimentation platform, ALICE , designed for exploring cellular network infrastructure and its interconnection with content delivery networks.

- We characterized the DNS infrastructure, inter-domain connectivity and network assignment dynamics of next generation cellular networks using three years of data from over 1900 volunteer mobile clients.

- We proposed an alternative approach for CDN replica selection based on a client's assigned packet gateway, called Gateway-Based Replica Selection (GBRS). We showed that GBRS performs near optimal replica selection for cellular clients. We demonstrated measurement techniques which allow PGW discovery and partitioning of cellular networks.

- We designed and implemented two systems, TILLER and MACHETE , which perform this cellular network partitioning with high accuracy.

9.2 Future Work

Combined Network Vantage Points. The obvious next steps for this work involves combining the view points of both TILLER and MACHETE . Each system works as a perfect complement for the other. TILLER has greater visibility into cellular networks

and thus greater accuracy in its characterization, yet requires sufficient and continued mobile vantage point coverage in all networks. From its position within the network core, MACHETE overcomes these scalability issues, yet suffers from a lack of visibility and obscured measurements due to its external position. Exploring how even partial TILLER presence can improve the accuracy of all server side measurements is an area of open exploration.

Deploying GBRS. One of the issues in fully deploying our proposed Gateway Based Replica Selection system is the lack of an effective client location signal. Cellular networks, and the stability of IP addresses to PGW instances, seem a perfect fit for end-user mapping, assuming one knows the allocation and location of network PGWs which TILLER and MACHETE produce. The problem is that to our knowledge, no cellular operator supports the EDNS(0) client subnet extension (ECS), and most mobile devices are prevented from changing their DNS resolvers. We are exploring options for stub resolver on mobile devices to implement GBRS from the client's device.

Appendix A

Flexible Experimentation for Mobile Networks

In this chapter, I introduce ALICE , **A** **L**ightweight **I**nterface for **C**ontrolled **E**xperiments, a platform for mobile experimentation, and describe its design and implementation. I motivate the need for ALICE despite the large body of research in mobile network measurement and network experimentation.

A.1 A Case for More Robust Mobile Experimentation

End-host measurements are critical to understanding the performance of cellular networks. Many MNOs prohibit external probes from reaching mobile clients or network infrastructure such as DNS servers. Probes from mobile end-hosts, on the other hand, enjoy much greater visibility into cellular infrastructure. In addition, in light of the large impact the radio link has on end-to-end performance, mobile end-hosts are the only party able to capture the impact of their context on measurements.

More than other types of connectivity, cellular performance is heavily influenced by last-mile effects of the radio, and is many times a function of its surrounding context. The influences on the radio interface include contention for the wireless channel, device radio power states [62], external radio interference, device mobility and orientation [52], to name a few. Attempts at measuring cellular network performance from server side measurements (such as TCP round-trip-times) can be difficult to interpret, since performance degradation caused by the client context alone may be indistinguishable from other sources (e.g. radio

allocation, RRC configuration, etc.). These connections may originate from TCP splitters or performance enhancing proxies (PEPs) [118] and therefore lack the ability to capture full end-to-end performance.

In addition to the challenges faced by wired network experimentation platforms, mobile platforms face the following unique constraints.

- **Context**: As previously discussed, device context has a large role in the overall performance of cellular networks [52]. All measurements of cellular devices need to take this context into account, either through measuring device context, or through direct comparison in approximately the same context.

- **Resource Limitations**: Mobile devices have more restrictions on both power and network usage than other end-hosts. A mobile experimentation platform must be both lightweight to use a few resources as possible, as well as the ability to audit and temper its resource usage.

- **Coverage**: Mobile networks require extensive geographic coverage, due to the high spatial variance in radio signal infrastructure deployments. In more traditional networks, such as broadband access networks, coverage is related to the number of vantage points in each network, and temporal changes in network conditions. The additional spatial changes brought upon by mobile users greatly increases the number of measurements needed to characterize cellular performance.

- **Security and Privacy**: Mobile devices present more avenues for privacy intrusion and data leakage than wired machines. Some of this information such as a device's location, is crucial for understanding the underlying network measurement. A platform for mobile experimentation must balance between measurement utility and user privacy.

In the following section we outline how these constraints existing network experimentation platforms fail to account for these constraints.

A.1.1 Existing Network Experimentation Platforms

Mobile experimentation platforms for wired networks have shown the value of open and flexible vantage points for network experiments. Successful efforts of these platforms include general purpose measurement and system platforms such as PlanetLab [32] and Seattle [21]. These platforms create sandboxed environments on participating nodes which allow the execution of, sometimes limited, but arbitrary code on these hosts.

Other systems are more focused on network measurements rather than general purpose use. These can be implemented either through software on participating hosts such as DipZoom [90] and Dasu [96], or through dedicated hardware in end-user networks such as BisMark [110], SamKnows [2] and Ripe Atlas [13]. These systems present a restricted API, generally presenting common network probe functionality such as `ping`, `traceroute`, `HTTP GET`, and `DNS` resolution to name a few.

Each system differs along a few axes of design trends in network experimentation platforms: programmability, security and adoption. For example, PlanetLab allows near complete access to the assigned virtual machine, yet requires participants to provide dedicated hardware in order to join the collective.

Numerous research projects have collected mobile performance data from mobile end-hosts [61, 63, 79]. These projects, using specialized mobile applications on end-host devices, can be grouped into either *one-off experiments* or *platforms*.

The one-off experiments capture measurements through a single mobile application designed with a specific purpose. These experiments must all follow the same process of (1) application development where experiments are designed and a individual mobile application is developed and (2) application dissemination where that experiment application

is distributed to volunteers under different incentives. These incentives range from free mobile phone plans [104], to paid studies, to alternative incentives such as a speed testing application [61, 63].

To avoid repeating the same procedure for each experiment, several projects have developed a general purpose mobile experiment platform for network performance monitoring. These platforms expose an API for common network measurement actions, such as `ping` or `traceroute`, for experimentation. Existing mobile platforms include MobiPerf (now Mobilyzer [79]) and Mitate [54]. Closest to this work is the Mobilyzer project [79], which attempts to become a centralized platform for mobile experimentation. While the authors have attempted to solve many of the same challenges we previously described, the experimentation semantics is limiting, in most instances allowing only a collection of independent measurements.

While our experience has shown that greater amounts of flexibility and programmability are While flexibility and programmability are always appreciated for wired network experiments, mobile devices devices demand more flexible experimentation due to their constantly changing connectivity and context. Effective network experiments need the ability adapt to changing context, to respond to results from previous probes, and conduct conditional experimentation depending network changes. The design of ALICE allows for this adaptability, offering significantly greater experiment flexibility, which is necessary for more complex experimentation in mobile environments. In the following section, we outline the design of `Alice` to provide these capabilities.

A.2 ALICE Mobile Experimentation Platform

To address these challenges, as well as provide an interface for powerful mobile network experimentation, I designed and implemented Alice as a library for the Android mobile operating system.

A.2.1 Alice Design Principles

In `Alice`, we address each of the challenges outlined in the previous section. Below we outline our major design decisions in `Alice` which differentiate if from previous iterations of network experiment platforms.

- **Flexible experimentation through data manipulation.** Alice balances capability and security by allowing fully programmable access to the result data, but not full program execution. Alice experiment scripts are interpreted Javascript programs, which execute in a sandboxed environment where only a small number of API calls are allowed to access larger system resources, yet, we allow full ability to inspect, manipulate and develop logical elements to experiment scripts within the sandbox environment. We call this approach *data manipulation*, and believe it represents a balanced compromise to future experimentation platform designs.

- **Multiplexed distribution as application library.** Since Alice is built as a library, it can easily included in multiple applications, thus greatly increasing its avenues of distribution.

- **Lightweight resource usage.** Alice's execution engine is designed to audit, and limit the amount of resources used over different time scales. These cover to rate of network probing, overall network traffic and device uptime. Limits apply to per experiment quotas, as well as overall device usage. These are meant less to protect devices from malicious experimenters, and more to prevent failures from experimenter mistakes.

A.2.2 System Architecture

In this section we outline the `Alice` architecture, illustrated in Figure A.1. `Alice` operates both on the mobile device, and through a centralized cloud service. The cloud service orchestrates the distribution of experiment scripts to clients, ensuring scripts meet the

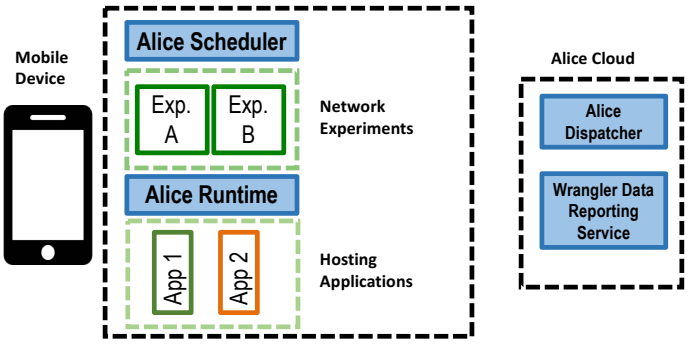


Figure A.1: Architecture diagram for ALICE mobile experiment platform.

capabilities of each client and across heterogeneous version of the `Alice` client library. The `Alice` client library handles the scheduling of experiment scripts, execution of experiment code, and reporting of experiment data.

The client library is distributed as a add-on library to existing Android applications. By packaging our experiment platform as a mobile library instead of a stand-alone application, we are able to multiplex distribution across multiple distribution channels, increase adoption chances. While this model would benefit any such experimentation platform, it is especially required for a mobile experiment platform, since the half-life of mobile apps is significantly lower than desktop applications, as users are more conscious of device memory and running processes. In fact, a recent report [9] showed that 77% of daily active users are lost after 3 days, and 95% are lost after 90 days! Therefore a platform which is able to adapt to different to the preferences of different users is necessary.

- **Experiment programming.** The key tenant in the design of ALICE is to lower the barrier to complex network experimentation through ease of experiment construction. Based on our experience operating the Dasu [96] experimentation platform, we found a

Probe Name	Type	Description
DNS	Active	DNS resolution with functionality similar to dig.
HTTP GET	Active	Performs an HTTP Get request for a specified URL.
IPerf	Active	Bandwidth testing tool with various control parameters.
Device Info	Active	Records device specific information such as the device unique identifier and available radios and sensors.
NDT	Active	Network Diagnostic Tool is an existing robust network performance testing suite implemented in Java.
Network Info	Active	Records the active and available network interfaces (i.e. WiFI, 3g, 4g).
Wifi Scan	Active	Performs a WiFi access point scan.
Ping	Active	A ping probe which measures the RTT for a specified IP address.
Traceroute	Active	A traceroute probe to a specified IP address.
Traffic Stats	Active	Records the bytes used by each registered application, similar to running /proc/stat/net.
Location	Passive	Records a users current location at a specified granularity.
Cellular Signal Strength	Passive	Records the measured cellular signal from the device for a specified time interval.

Table A.1: Overview of available probes in Alice platform. Active probes are those which are launched and can return a value. Passive probes are recorded in the background for a specified period of time.

large learning curve when constructing experiments in the event-driven JBOSS Drools language [42]. Instead, ALICE follows the example of the Fathom [40] project and has experimenters construct network experiments through Javascript. In addition to being one of the most common programming languages today, Javascript allows for complicated parsing and processing of data along with loop and conditional control structures.

Unique to `Alice` is the ability to create functions *within* the experiment itself, looping through data structures for multiple, repetitive probes, adjust experiment to different conditions and dynamically generate experiment probes. Figure A.2 displays the experiment script code for parsing the results of a DNS query, returning the IP address from any"A" records in the response.

This ability to pull information from previous experiment probes, such as the client's current IP address from an external web service, and to construct dynamic experiment

```
function get_a_record_ip(dnsresponse) {
  if (dnsresponse.answers.length > 0) {
    for (var ind=0;ind < dnsresponse.answers.length; ind++) {
      if (dnsresponse.answers[ind].type == "A") {
        return dnsresponse.answers[ind].address;
      }
    }
  }
  return null;
}
```

Figure A.2: Code for defining user-defined functions within experiments. The code above returns the IP address from any "A" records within a DNS response, if available.

```
for (var i=0; i < landmark_servers.length; i++) {
    var ret = AndroidMeasurementEngine.doJsDns(landmark_servers[i], "
        8.8.8.8");
    var dns_resp = JSON.parse(ret);
    var dest_ip = get_a_record_ip(dns_resp);

    if (dest_ip != null) {
        AndroidMeasurementEngine.getRemoteString("http://"+dest_ip+"
            :33000/tr?ip="+ip_str+"&trid="+tr_key);
        AndroidMeasurementEngine.doTraceroute("tr-"+landmark_servers[i]+"-
            "+dest_ip, dest_ip);
    }
}
```

Figure A.3: Sample code showing several Alice features, including loops, conditionals, and the ability to pass results from previous network probes into future probes. In the above code, Alice loops through a list of PlanetLab servers, launching bidirectional traceroutes between it and the mobile device.

probes (e.g. IP address and query string of the PlanetLab request), is a key feature of `Alice`. We show sample code showing this paradigm in Figure A.3.

- **Network probe modules.** The interface between the experiment process and executable system code are called network probes. These probes represent self contained functions with well defined execution and output. Network probes the functionality needed for network experimentation, while limiting the access to sockets and other low-level system functionality to maintain system security.

- **Synchronous and parallel execution.** Included in `Alice` are two sets of network probes, synchronous and parallel. By default all probes are considered independent and parallel, and thus are easier to schedule to minimize resources. Synchronous probes are also available, and used in cases where the results of a network probe are needed to determine program logic, or to construct an additional probe (e.g. to ping the IP address just resolved through a specific resolver). With synchronous probes, a JSON object is returned which can be read, manipulated, and passed on to other probes within the code.

- **Experiment dissemination.** Experiments are disseminated periodically through a centralized server interface. Each experiment contains its own scheduling information so end-devices do not need to make contact for each execution. Experiments are distributed depending on the particular version of Alice in use, since subsequent versions of Alice have introduced new or enhanced API calls.

- **Experiment execution.** Execution access in `Alice` spans a two-tiered system similar to Dasu [96]. In this two tiered system, access to low-level system commands such as network probes are only accessible through `Alice`'s API. `Alice` enhances flexibility by executing in a sandboxed environment, and allowing full language capability within that environment to institute program control, define functions or other complex language functions. The result is a execution environment which is able to perform complex network experiments while maintaining system and resource safety through its API.

- **Balancing security with programmability.** Within every experiment platform is a balance between the freedom and flexibility of the exposed interface, and the security of the hosting machine. For example, how can one ensure that an open, programmable network experiment platform won't be co-opted for malicious activity such as a DDOS

attack or a botnet. Other instances can include non-malicious intent, but nonetheless harmful actions to the hosting machine: such as a network experiment exceeding a user's data limits.

It is therefore imperative for a experiment platform to be able to ensure the safety of its host. `Alice` accomplishes this through its two-tiered experiment interface, which allows full programmability and manipulation of probe data, while only exposing a limited interface for actual system commands. This approach yields nearly all the programmability achieved by native code in the context of network experimentation, while maintaining strict isolation of experiment code, and auditing of each probe interface to prevent excessive resource usage.

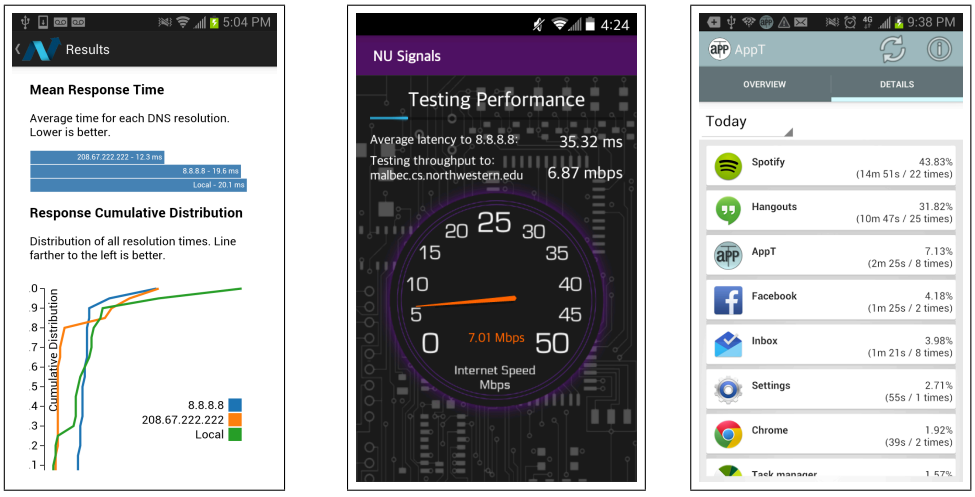
A.3 Alice Implementation

`Alice` is implemented for the Android Mobile Operating System as a library, in 8864 SLOC in Java. It runs as a background service on the device. as a . `Alice` requires minimal integration into hosting applications. Applications need only to add a *single* line of code into their main application class in order to start the service. In addition, 6 lines of XML need to be added into the application's Android manifest file to declare the service and its receivers. In total, these changes only amount to 7 single lines of code modification.

A.3.1 Alice Deployments

As of December 2016, the `Alice` library has already been run on over 2100 unique mobile devices. It has been included in three production Android apps: NU Signals [82], Namehelp Mobile [81], and Application Time (AppT) [80]. Figure A.4 displays screenshots from each existing hosting application.

During the deployment of `Alice`, I encountered several challenges running and maintaining a crowdsourced mobile experiment platform for mobile. One of the largest issues stems from user acquisition and retention. We attempted several marketing efforts from the

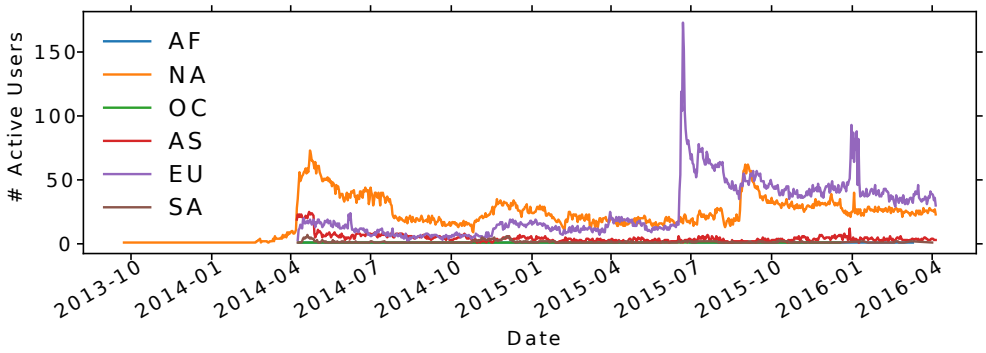


(a) Namehelp Mobile (b) NUX NU Signals (c) Application Time

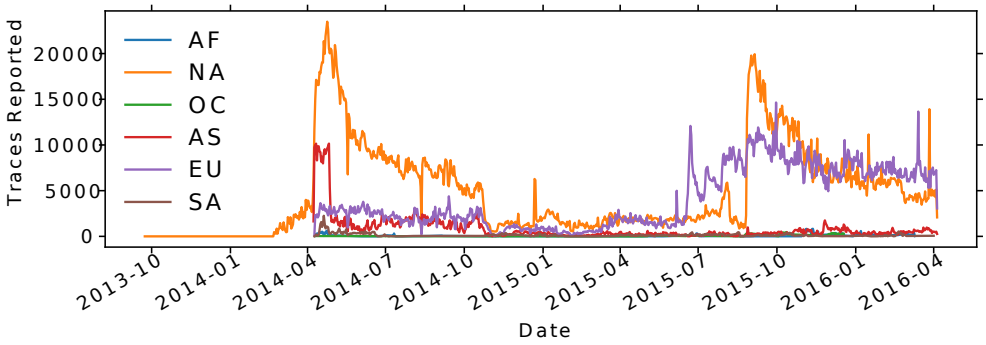
Figure A.4: Screenshots from each of the three Android applications which are using the Alice experiment library.

created hosting applications, such as advertising through our University's press team, and promoting on several large news aggregators (e.g. Slashdot, HackerNews). More challenging than user recruitment was user retention, ensuring that users kept our mobile applications installed over a long period of time. As with all mobile applications, user retention is a problem, with typical mobile applications losing 77% of daily active users after 3 days, and 95% after 3 months [9]. This is a large problem for our platform since we are interested in longitudinal measurements from mobile clients.

Figure A.5 displays deployment statistics for `Alice` from October 2013 to May 2016. The number of active users (Fig. A.5a) displays the number of unique devices which reported measurements each day. The spikes in reported users correspond to different application releases which were picked up by news outlets. The first spike in North American users around April 2014 represents when *Namehelp Mobile*, our first hosting application, was published in our University's monthly magazine. The second spike, in June 2015, of European users comes from *Application Time*, another of our developed hosting applications, which



(a) Daily active users of Alice.



(b) Daily traceroutes conducted by Alice.

Figure A.5: Daily statistics for Alice platform aggregated by continent.

was picked up by several European online news outlets. Finally, the final spike in August 2015 comes from internal advertising within a large Internet company.

References

[1] Network information api.

[2] Samknows. `http://www.samknows.com`.

[3] Bernhard Ager, Wolfgang Mühlbauer, Georgios Smaragdakis, and Steve Uhlig. Comparing DNS resolvers in the wild. In *Proc. IMC*, 2010.

[4] Akamai NetSession. `http://www.akamai.com/client`.

[5] Akamai. Edgescape. `http://goo.gl/qCHPh1`.

[6] Zakaria Al-Qudah, Seungjoon Lee, Michael Rabinovich, Oliver Spatscheck, and Jacobus Van der Merwe. Anycast-aware transport for content delivery networks. In *Proc. WWW*, pages 301–310, 2009.

[7] Hussein A Alzoubi, Seungjoon Lee, Michael Rabinovich, Oliver Spatscheck, and Jacobus Van der Merwe. Anycast cdns revisited. In *Proc. WWW*, 2008.

[8] Hussein A. Alzoubi, Michael Rabinovich, and Oliver Spatscheck. The anatomy of LDNS clusters: Findings and implications for web content delivery. In *Proc. WWW*, 2013.

[9] Andrew Chen. New data shows losing 80% of mobile users is normal, and why the best apps do better. `https://tinyurl.com/opladfz`.

[10] Anonymous. The collateral damage of internet censorship by dns injection. *SIGCOMM Comput. Commun. Rev.*, 42(3), June 2012.

[11] Ruwaifa Anwar, Haseeb Niaz, David Choffnes, Italo Cunha, Phillipa Gill, and Ethan Katz-Bassett. Investigating interdomain routing policies in the wild. In *Proceedings of the 2015 ACM Conference on Internet Measurement Conference*, pages 71–77. ACM, 2015.

[12] Brice Augustin, Xavier Cuvellier, Benjamin Orgogozo, Fabien Viger, Timur Friedman, Matthieu Latapy, Clémence Magnien, and Renata Teixeira. Avoiding traceroute anomalies with paris traceroute. In *Proc. IMC*. ACM, 2006.

[13] Vaibhav Bajpai and Jürgen Schönwälder. A survey on internet performance measurement platforms and related standardization efforts. *IEEE Communications Surveys & Tutorials*, 17(3):1313–1341.

[14] Mahesh Balakrishnan, Iqbal Mohomed, and Venugopalan Ramasubramanian. Where's that phone?: geolocating IP addresses on 3G networks. In *Proc. IMC*, 2009.

[15] Geoffrey H Ball and David J Hall. Isodata, a novel method of data analysis and pattern classification. Technical report, DTIC Document, 1965.

[16] Nimantha Baranasuriya, Vishnu Navda, Venkata N Padmanabhan, and Seth Gilbert. Qprobe: Locating the bottleneck in cellular communication. In *Proc. ACM CoNEXT*, 2015.

[17] Paul Barford, Azer Bestavros, John Byers, and Mark Crovella. On the marginal utility of network topology measurements. In *Proc. ACM SIGCOMM Workshop on Internet Measurement*, 2001.

[18] BGP4.as. BGP Looking Glasses for IPv4/IPv6, Traceroute and BGP Route Servers. `http://www.bgp4.as/looking-glasses`.

[19] Xue Cai and John Heidemann. Understanding block-level address usage in the visible internet. In *ACM SIGCOMM Computer Communication Review*, volume 40, pages 99–110. ACM, 2010.

[20] Matt Calder, Ashley Flavel, Ethan Katz-Bassett, Ratul Mahajan, and Jitendra Padhye. Analyzing the performance of an anycast cdn. In *Proc. IMC*, 2015.

[21] Justin Cappos, Ivan Beschastnikh, Arvind Krishnamurthy, and Tom Anderson. Seattle: a platform for educational cloud computing. In *ACM SIGCSE Bulletin*, volume 41, pages 111–115. ACM, 2009.

[22] Martin Casado and Michael J. Freedman. Peering through the shroud: The effect of edge opacity on ip-based client identification. In *Proc. USENIX NSDI*, 2007.

[23] Center for Applied Internet Data Analysis. Archipelago (Ark) Measurement Infrastructure. `http://www.caida.org/projects/ark/`.

[24] Rajiv Chakravorty, Suman Banerjee, Pablo Rodriguez, Julian Chesterfield, and Ian Pratt. Performance optimizations for wireless wide-area networks: Comparative study and experimental evaluation. In *Proc. of MobiCom*, 2004.

[25] Rajiv Chakravorty, Andrew Clark, and Ian Pratt. Optimizing web delivery over wireless links: design, implementation, and experiences. *Selected Areas in Communications, IEEE Journal on*, 23(2):402–416, 2005.

[26] Mun Choon Chan and Ramachandran Ramjee. Tcp/ip performance over 3g wireless links with rate and delay variation. *Wireless Networks*, 11(1-2):81–97, 2005.

[27] Fangfei Chen, Ramesh K Sitaraman, and Marcelo Torres. End-user mapping: Next generation request routing for content delivery. In *Proc. ACM SIGCOMM*, 2015.

[28] Kai Chen, David R Choffnes, Rahul Potharaju, Yan Chen, Fabian E Bustamante, Dan Pei, and Yao Zhao. Where the sidewalk ends: Extending the internet as graph using traceroutes from p2p users. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, pages 217–228. ACM, 2009.

[29] David R Choffnes and Fabián E Bustamante. Taming the torrent: a practical approach to reducing cross-isp traffic in peer-to-peer systems. *ACM SIGCOMM Computer Communication Review*, 38(4):363–374, 2008.

[30] David R Choffnes and Fabián E Bustamante. Taming the torrent: a practical approach to reducing cross-isp traffic in peer-to-peer systems. In *ACM SIGCOMM Computer Communication Review*, volume 38, pages 363–374. ACM, 2008.

[31] David R Choffnes, Fabián E Bustamante, and Zihui Ge. Crowdsourcing service-level network event monitoring. In *Proc. ACM SIGCOMM*, 2010.

[32] Brent Chun, David Culler, Timothy Roscoe, Andy Bavier, Larry Peterson, Mike Wawrzoniak, and Mic Bowman. Planetlab: an overlay testbed for broad-coverage services. *ACM SIGCOMM Computer Communication Review*, 33(3):3–12, 2003.

[33] CISCO. Architectural Considerations for Backhaul of 2G/3G and Long Term Evolution Networks. `http://www.cisco.com/c/en/us/solutions/collateral/service-provider/mobile-internet/white_paper_c11-613002.pdf`.

[34] CISCO. CISCO visual networking index: Global mobile data traffic forecast update 2013-2018. Technical report, CISCO Systems Inc., 2014.

[35] CISCO. CISCO global mobile data traffic forecast update, 2014-2019 white paper, 2015.

[36] Aaron Clauset, Mark EJ Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical review E*, 70(6):066111, 2004.

[37] Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 24(5):603–619, 2002.

[38] Joao Damas, Michael Graff, and Paul Vixie. Extension mechanisms for dns (edns (0)). 2013.

[39] Amogh Dhamdhere and Constantine Dovrolis. Ten years in the evolution of the internet ecosystem. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*, pages 183–196. ACM, 2008.

[40] Mohan Dhawan, Justin Samuel, Renata Teixeira, Christian Kreibich, Mark Allman, Nicholas Weaver, and Vern Paxson. Fathom: A browser-based network measurement platform. In *Proc. IMC*, 2012.

[41] John Dilley, Bruce Maggs, Jay Parikh, Harald Prokop, Ramesh Sitaraman, and Bill Weihl. Globally distributed content delivery. *Internet Computing, IEEE*, 6(5):50–58, 2002.

[42] Drools. Drools - Business Rules Management System. `http://www.drools.org/`.

[43] Ramakrishnan Durairajan, Paul Barford, Joel Sommers, and Walter Willinger. Intertubes: a study of the us long-haul fiber-optic infrastructure. In *Proc. ACM SIGCOMM*. ACM, 2015.

[44] Ramakrishnan Durairajan, Joel Sommers, and Paul Barford. Layer 1-informed internet topology measurement. In *Proc. IMC*. ACM, 2014.

[45] Ericsson. Ericsson Mobility Report: June 2016. Technical report.

[46] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, 1996.

[47] Adriano Faggiani, Enrico Gregori, Luciano Lenzini, and Valerio Luconi. Measuring the internet topology with smartphones. In *Proceedings of the 2015 ACM SIGCOMM Workshop on Crowdsourcing and Crowdsharing of Big (Internet) Data*, pages 45–50. ACM, 2015.

[48] Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. On power-law relationships of the internet topology. In *ACM SIGCOMM computer communication review*, volume 29, pages 251–262. ACM, 1999.

[49] Ashley Flavel, Pradeepkumar Mani, David Maltz, Nick Holt, Jie Liu, Yingying Chen, and Oleg Surmachev. Fastroute: A scalable load-aware anycast routing architecture for modern cdns. In *Proc. USENIX NSDI*, 2015.

[50] Pawel Foremski, David Plonka, and Arthur Berger. Entropy/ip: Uncovering structure in ipv6 addresses. In *Proceedings of the 2016 ACM on Internet Measurement Conference*, pages 167–181. ACM, 2016.

[51] Lixin Gao. On inferring autonomous system relationships in the internet. *IEEE/ACM Transactions on Networking (ToN)*, 9(6):733–745, 2001.

[52] Aaron Gember, Aditya Akella, Jeffrey Pang, Alexander Varshavsky, and Ramon Caceres. Obtaining in-context measurements of cellular network performance. In *Proc. IMC*, 2012.

[53] Phillipa Gill, Martin Arlitt, Zongpeng Li, and Anirban Mahanti. The flattening internet topology: Natural evolution, unsightly barnacles or contrived collapse? In *Passive and Active Network Measurement*, pages 1–10. Springer, 2008.

[54] Utkarsh Goel, Ajay Miyyapuram, Mike P Wittie, and Qing Yang. Mitate: mobile internet testbed for application traffic experimentation. In *International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*, pages 224–236. Springer, 2013.

[55] Utkarsh Goel, Moritz Steiner, Mike P Wittie, Martin Flack, and Stephen Ludin. Detecting cellular middleboxes using passive measurement techniques. In *Proc. PAM*, 2016.

[56] Utkarsh Goel, Moritz Steiner, Mike P Wittie, Erik Nygren, Martin Flack, and Stephen Ludin. A case for faster mobile web in cellular ipv6 networks. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, pages 176–188. ACM, 2016.

[57] Carles Gomez, Marisa Catalan, David Viamonte, Josep Paradells, and Anna Calveras. Web browsing optimization over 2.5 g and 3g: end-to-end mechanisms vs. usage of performance enhancing proxies. *Wireless Communications and Mobile Computing*, 8(2):213–230, 2008.

[58] Google. Frequently asked questions - Public DNS – Google Developers. `https://developers.google.com/speed/public-dns/faq`.

[59] Cheng Huang, Angela Wang, Jin Li, and Keith W Ross. Measuring and evaluating large-scale CDNs. In *Proc. IMC*, 2008.

[60] Cheng Huang, Angela Wang, Jin Li, and Keith W. Ross. Understanding hybrid CDN-P2P: why Limelight needs its own Red Swoosh. In *Proc. ACM NOSSDAV*, 2008.

[61] Junxian Huang, Feng Qian, Alexandre Gerber, Z. Morley Mao, Subhabrata Sen, and Oliver Spatscheck. A close examination of performance and power characteristics of 4g lte networks. In *Proc. of MobiSys*, 2012.

[62] Junxian Huang, Feng Qian, Alexandre Gerber, Z Morley Mao, Subhabrata Sen, and Oliver Spatscheck. A close examination of performance and power characteristics of 4g lte networks. In *Proc. of MobiSys*, 2012.

[63] Junxian Huang, Qiang Xu, Birjodh Tiwana, Z. Morley Mao, Ming Zhang, and Paramvir Bahl. Anatomizing application performance differences on smartphones. In *Proc. of MobiSys*, 2010.

[64] Haiqing Jiang, Zeyu Liu, Yaogong Wang, Kyunghan Lee, and Injong Rhee. Understanding bufferbloat in cellular networks. In *Proceedings of the 2012 ACM SIGCOMM workshop on Cellular networks: operations, challenges, and future design*. ACM, 2012.

[65] Kirk L. Johnson, John F. Carr, Mark S. Day, and M. Frans Kaashoek. The measured performance of content distribution networks. *Computer Communications*, 24, 2001.

[66] David Karger, Eric Lehman, Tom Leighton, Rina Panigrahy, Matthew Levine, and Daniel Lewin. Consistent hashing and random trees: Distributed caching protocols for relieving hot spots on the world wide web. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*. ACM, 1997.

[67] Balachander Krishnamurthy, Craig Wills, and Yin Zhang. On the use and performance of content distribution networks. In *Proc. IMC*, 2001.

[68] Rupa Krishnan, Harsha V. Madhyastha, Sridhar Srinivasan, Sushant Jain, Arvind Krishnamurthy, Thomas Anderson, and Jie Gao. Moving beyond end-to-end path information to optimize CDN performance. In *Proc. IMC*, 2009.

[69] Craig Labovitz, Scott Iekel-Johnson, Danny McPherson, Jon Oberheide, and Farnam Jahanian. Internet inter-domain traffic. *ACM SIGCOMM Computer Communication Review*, 41(4):75–86, 2011.

[70] Tom Leighton. Improving performance on the Internet. *CACM*, 52, February 2009.

[71] Wai Kay Leong, Aditya Kulkarni, Yin Xu, and Ben Leong. Unveiling the hidden dangers of public ip addresses in 4g/lte cellular data networks. In *Proc. of HotMobile*, 2014.

[72] Level 3. Level 3 Communications. `http://www.level3.com/`.

[73] Xin Liu, Ashwin Sridharan, Sridhar Machiraju, Mukund Seshadri, and Hui Zang. Experiences in a 3g network: Interplay between the wireless channel and applications. In *Proc. of MobiCom*, 2008.

[74] Bruce Maggs. Challenges in engineering the world's largest content delivery networkstate of the union address. 2008. Keynote Address.

[75] Z. Morley Mao, Charles D. Cranor, Fred Douglis, Michael Rabinovich, Oliver Spatscheck, and Jia Wang. A precise and efficient evaluation of the proximity between web clients and their local DNS servers. In *Proc. USENIX ATC*, 2002.

[76] Zhuoqing Morley Mao, Jennifer Rexford, Jia Wang, and Randy H Katz. Towards an accurate as-level traceroute tool. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 365–378. ACM, 2003.

[77] Siddharth Mohan, Rohit Kapoor, and Bibhu Mohanty. Latency in hspa data networks.

[78] NGMN Alliance. NGMN 5G White Paper. Technical report, Next Generation Mobile Network Alliance, 2016.

[79] Ashkan Nikravesh, Hongyi Yao, Shichang Xu, David Choffnes, and Z Morley Mao. Mobilyzer: An open platform for controllable mobile network measurements. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, pages 389–404. ACM, 2015.

[80] Northwestern Aqualab. Application Time (AppT). `http://www.aqualab.cs.northwestern.edu/projects/283-appt`.

[81] Northwestern Aqualab. Namehelp Mobile. `http://aqualab.cs.northwestern.edu/projects/237-namehelp-mobile`.

[82] Northwestern Undergraduate User Experience and Mobile Development Team. NU Signals v2. `https://nux.northwestern.edu/projects/nu-signals-v2`.

[83] Erik Nygren, Ramesh K Sitaraman, and Jennifer Sun. The akamai network: a platform for high-performance internet applications. *ACM SIGOPS Operating Systems Review*, 44(3):2–19, 2010.

[84] OpenDNS. Cloud Delivered Enterprise Security with OpenDNS. `https://www.opendns.com`.

[85] John S. Otto, Mario A. Sánchez, John P. Rula, and Fabián E. Bustamante. Content delivery and the natural evolution of DNS: Remote DNS trends, performance issues and alternative solutions. In *Proc. IMC*, 2012.

[86] Pew Research Center. Internet Seen as Positive Influence on Education but Negative on Morality in Emerging and developing Nations. Technical report, Pew Research Center, 2015.

[87] Marcelo Pias, Jon Crowcroft, Steve Wilbur, Tim Harris, and Saleem Bhatti. Lighthouses for scalable distributed location. *Peer-to-Peer Systems II*, pages 278–291, 2003.

[88] David Plonka and Arthur Berger. Temporal and spatial classification of active ipv6 addresses. In *Proc. IMC*, 2015.

[89] Feng Qian, Zhaoguang Wang, Alexandre Gerber, Zhuoqing Morley Mao, Subhabrata Sen, and Oliver Spatscheck. Characterizing radio resource allocation for 3g networks. In *Proc. IMC*, 2010.

[90] Michael Rabinovich, Sipat Triukose, Zhihua Wen, and Limin Wang. Dipzoom: The internet measurements marketplace. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1–6. IEEE, 2006.

[91] Sylvia Ratnasamy, Mark Handley, Richard Karp, and Scott Shenker. Topologically-aware overlay construction and server selection. In *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1190–1199. IEEE, 2002.

[92] Philipp Richter, Florian Wohlfart, Narseo Vallina-Rodriguez, Mark Allman, Randy Bush, Anja Feldmann, Christian Kreibich, Nicholas Weaver, and Vern Paxson. A multi-perspective analysis of carrier-grade nat deployment. In *Proc. IMC*, 2016.

[93] RIPE NCC. RIPE Atlas. `https://atlas.ripe.net`.

[94] Pablo Rodriguez and Vitali Fridman. Performance of peps in cellular wireless networks. In *Web content caching and distribution*, pages 19–38. Springer, 2004.

[95] John P. Rula and Fabian E. Bustamante. Behind the curtain: Cellular dns and content replica selection. In *Proc. IMC*, 2014.

[96] Mario A Sánchez, John S Otto, Zachary S Bischof, David R Choffnes, Fabián E Bustamante, Balachander Krishnamurthy, and Walter Willinger. Dasu: Pushing experiments to the internet's edge. In *NSDI*, pages 487–499, 2013.

[97] Ryan Saunders, Junguk Cho, Arijit Banerjee, Frederico Rocha, and Jacobus Van der Merwe. P2p offloading in mobile networks using sdn. In *Proc. SOSR*, 2016.

[98] Paul Schmitt, Morgan Vigil, and Elizabeth Belding. A study of mvno data paths and performance. In *Proc. PAM*, 2016.

[99] Kyle Schomp, Tom Callahan, Michael Rabinovich, and Mark Allman. On measuring the client-side dns infrastructure. In *Proc. IMC*, 2013.

[100] Anees Shaikh, Renu Tewari, and Mukesh Agrawal. On the effectiveness of DNS-based server selection. In *Proc. IEEE INFOCOM*.

[101] Abhigyan Sharma, Xiaozheng Tie, Hardeep Uppal, Arun Venkataramani, David Westbrook, and Aditya Yadav. A global name service for a highly mobile internetwork. In *Proc. ACM SIGCOMM*, 2014.

[102] Rob Sherwood, Adam Bender, and Neil Spring. Discarte: a disjunctive internet cartographer. 2008.

[103] Rob Sherwood and Neil Spring. Touring the internet in a tcp sidecar. In *Proc. IMC*. ACM, 2006.

[104] Jinghao Shi, Edwin Santos, and Geoffrey Challen. Why and how to use phonelab. *GetMobile: Mobile Comp. and Comm.*, pages 32–38, 2016.

[105] SoftLayer. SoftLayer. `http://www.softlayer.com`.

[106] Neil Spring, Ratul Mahajan, and David Wetherall. Measuring isp topologies with rocketfuel. In *ACM SIGCOMM Computer Communication Review*, volume 32, pages 133–145. ACM, 2002.

[107] Statista. Wireless subscriber united states by carrier 2013-2015. `http://www.statista.com/statistics/283507/subscribers-to-top-wireless-carriers-in-the-us/`.

[108] Florian Streibelt, Jan Böttger, Nikolaos Chatzis, Georgios Smaragdakis, and Anja Feldmann. Exploring edns-client-subnet adopters in your free time. In *Proc. IMC*, 2013.

[109] Ao-Jan Su, David R Choffnes, Aleksandar Kuzmanovic, and Fabián E Bustamante. Drafting behind akamai (travelocity-based detouring). *ACM SIGCOMM Computer Communication Review*, 36(4):435–446, 2006.

[110] Srikanth Sundaresan, Walter De Donato, Nick Feamster, Renata Teixeira, Sam Crawford, and Antonio Pescapè. Broadband internet performance: a view from the gateway. In *ACM SIGCOMM computer communication review*, volume 41, pages 134–145. ACM, 2011.

[111] Narseo Vallina-Rodriguez, Srikanth Sundaresan, Christian Kreibich, Nicholas Weaver, and Vern Paxson. Beyond the radio: Illuminating the higher layers of mobile networks. In *Proc. of MobiSys*, 2015.

[112] Limin Wang, Vivek Pai, and Larry Peterson. The effectiveness of request redirection on CDN robustness. In *Proc. USENIX OSDI*, 2002.

[113] Zhaoguang Wang, Zhiyun Qian, Qiang Xu, Zhuoqing Mao, and Ming Zhang. An untold story of middleboxes in cellular networks. 2011.

[114] Patrick Wendell, Joe Wenjie Jiang, Michael J Freedman, and Jennifer Rexford. Donar: decentralized server selection for cloud services. *ACM SIGCOMM Computer Communication Review*, 41(4):231–242, 2011.

[115] Keith Winstein, Anirudh Sivaraman, and Hari Balakrishnan. Stochastic forecasts achieve high throughput and low delay over cellular networks. In *Proc. USENIX NSDI*, 2013.

[116] Qiang Xu, Alexandre Gerber, Zhuoqing Morley Mao, and Jeffrey Pang. Acculoc: practical localization of performance measurements in 3g networks. In *Proceedings of*

the 9th international conference on Mobile systems, applications, and services, pages 183–196. ACM, 2011.

[117] Qiang Xu, Junxian Huang, Zhaoguang Wang, Feng Qian, and Alexandre Gerber Z. Morley Mao. Cellular data network infrastructure characterization and implication on mobile content placement. In *Proc. ACM SIGMETRICS*, 2011.

[118] Xing Xu, Yurong Jiang, Tobias Flach, Ethan Katz-Bassett, David Choffnes, and Ramesh Govindan. Investigating Transparent Web Proxies in Cellular Networks. Technical Report 14-944, University of Southern California, April 2014.

[119] Yasir Zaki, Thomas Pötsch, Jay Chen, Lakshminarayanan Subramanian, and Carmelita Görg. Adaptive congestion control for unpredictable cellular networks. In *Proc. ACM SIGCOMM*, 2015.

[120] Kyriakos Zarifis, Tobias Flach, Srikanth Nori, David Choffnes, Ramesh Govindan, Ethan Katz-Bassett, Z. Morley Mao, and Matt Welsh. Diagnosing Path Inflation of Mobile Client Traffic. In *Proc. PAM*, 2014.

[121] J.C. Zuniga, C.J. Bernardos, A. De La Oliva, T. Melia, R. Costa, and A. Reznik. Distributed mobility management: A standards landscape. *Communications Magazine, IEEE*, 51(3):80–87, March 2013.