3G Meets the Internet: Understanding the Performance of Hierarchical Routing in 3G Networks

Wei Dong
Department of Computer Science
University of Texas, Austin, TX
wdong86@cs.utexas.edu

Zihui Ge
AT&T Labs - Research
Florham Park, NJ
gezihui@research.att.com

Seungjoon Lee
AT&T Labs - Research
Florham Park, NJ
slee@research.att.com

***Abstract*—The volume of Internet traffic over 3G wireless networks is sharply rising. In contrast to many Internet services utilizing replicated resources, such as Content Distribution Networks (CDN), the current 3G standard architecture employs hierarchical routing, where all user data traffic goes through a small number of aggregation points using logical tunnels. In this paper, we study the performance implications of the interplay when 3G users access Internet services.**

We first identify a number of scenarios in which 3G users' service performance can be affected under hierarchical routing in comparison to an idealized *flat* routing. We then quantify this service impact by analyzing trace data obtained from a large-scale 3G network and a CDN provider. We find that the performance difference between hierarchical routing and flat routing increases when a 3G user accesses highly replicated service, and can further aggravates when the DNS caching is not properly managed under vertical handoff. For example, in our data analysis, the detour under hierarchical routing can cause a packet to travel extra distance by up to 1627km on the average case, which can lead to around 45.4% increase in round-trip latency. We also perform a measurement study to demonstrate that user mobility and web applications can lead to unexpected performance-impacting interactions, which can degrade the download throughput by up to an order of magnitude (0.9Mbps vs. 10.8Mbps).

I. Introduction

The past few years have witnessed a booming growth in cellular data network technologies (most-commonly available as 3G networks); smartphones and akin advanced portable devices (e.g., iPad), and a wide-variety of mobile telecommunication applications (such as mobile web, video conferencing, e-banking, online social networking, online gaming, e-commerce, etc.). Technology advances in these areas—network, device, and application—form a virtuous circle that further stimulates more technical innovation in them and drives popularity in the use of mobile applications even higher. 3G wireless communication has increasingly become an integral part of daily life. Rising together with the ever maturing technologies is users' expectation of the 3G service—users are looking beyond basic service availability and starting to demand higher service performance.

In this paper, we focus on a fundamental aspect of the 3G architecture—the routing, particularly in the 3G core network. As shown in Figure 1, the 3G core network consists of SGSNs (Serving GPRS Support Nodes) and GGSNs (Gateway GPRS Support Nodes), which connects network elements in the RAN (Radio Access Network). Service access of user device over the air is managed through the RAN, which consists

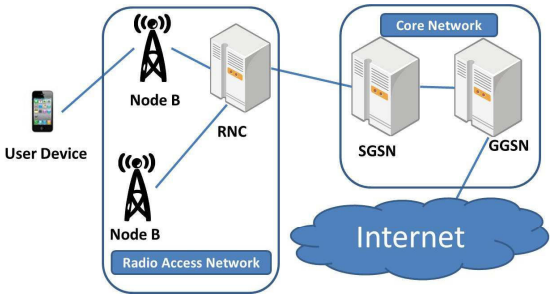


Fig. 1. Simplified 3G architecture

of NodeBs and RNCs (Radio Network Controllers). The 3G core network use a *hierarchical* structure for both routing traffic toward internal servers (e.g., for visual voice-mail) and channeling traffic to reach external networks (including the Internet). Specifically, to work with TCP/IP and other external networks, all data traffic uses GTP (GPRS Tunneling Protocol) tunnel and goes through GGSN. By using GGSN as a gateway to external networks, the 3G core network becomes fully transparent to the end user, and at the same time the cellular service provider retains full control of the key service management functions (e.g., access authorization, traffic accounting, mobility management). Hence, even in more recent cellular architecture, i.e., Enhanced Packet Core for LTE (Long Term Evolution, commercially branded as 4G), which uses fewer intermediate entities along the data path, all data packets still go through GGSN-equivalent gateways before reaching the Internet. (See Section VI for more details.)

While NodeBs and RNCs are typically widely spread in order to provide good radio coverage, typical 3G service providers have a small number of GGSN locations (e.g., less than 10 for a U.S. carrier), each covering a greater geographical region [13]. This is in contrast to the network architecture for High-Speed Internet service, which typically has a much larger number of equivalent sites (e.g., several tens) [12]. Such a design of 3G networks is based on various aspects such as service autonomy, security, and deployment and operational cost (to gradually keep up with increasing 3G service demand) as well as service performance. As a result, the hierarchical routing through few aggregations points (i.e., GGSNs) can lead to additional performance penalty when compared to the case with a *flat* routing structure. For example, the additional delay due to the SGSN and GGSN relay can affect user's perceived performance in interactive applications (such as gaming and

978-0-9836283-0-9

This paper was peer reviewed by subject matter experts for publication in the Proceedings of ITC 2011

e-trading). The increased round trip latency can also impact the TCP throughput on user's data transfers (such as web browsing and video streaming). Although such performance degradation might be tolerable in the past and even now(e.g., with little Internet traffic, larger delay on radio links), it will become increasingly unacceptable as users' demand and service performance expectation rises with advanced radio technology.

Mobility is another new aspect that 3G users bring to Internet service infrastructure. To support constant, yet high-performance connectivity, many 3G mobile devices have multiple wireless interfaces and automatically switch between different technologies (called *vertical handoff* [28]), as available technologies change over time (e.g., a smartphone switching from 3G to WiFi). This may cause some unexpected interactions with different system components or applications that do not take mobility into account. In this paper, we present an illustrating scenario and show how an interaction between vertical handoff and DNS (Domain Name System) caching can lead to suboptimal web download performance in practice.

We make the following contributions in this paper. (1) We analyze various scenarios where the hierarchical structure can affect the performance of Internet service by a 3G user. We first compare the *hierarchical* routing and *flat* routing to understand the inherent performance impact. We also consider scenarios where a 3G user accesses replicated contents (e.g., via CDN service) and show how the 3G architecture interacts with replicated Internet service. (2) We quantify the performance impact through detailed analysis of data obtained from a commercial large-scale 3G network and evaluate the potential performance gain with additional service resource deployment or alternative network structures. We also perform a measurement study to validate the performance impact in real settings. We have found that the hierarchical routing through relatively small number of GGSN locations does not compare favorably with *flat* routing, and the relative performance degrades further as 3G users access highly replicated services such as CDN, or when DNS caching in mobile applications is not properly managed under vertical handoff.

The rest of the paper is organized as follows: Section II provides some background on 3G architecture and CDN. In Section III, we identify a number of aspects in which the hierarchical routing of 3G data traffic affects system efficiency and service performance. In Section IV and Section V we present the result of our trace-driven analysis and measurement study. Section VI discusses how our findings apply to future wireless networks. We review related work in Section VII and conclude in Section VIII.

II. Background

A. 3G Architecture

The 3G mobile telecommunications services are embodied in two types of systems – the UMTS system (standardized by 3GPP) and the CDMA2000 system (standardized by 3GPP2). While our focus in this study is on the UMTS system [17], our findings should be equally applicable to the CDMA2000 system as well as other networks or services that use a small number of network aggregation points for Internet traffic (e.g., corporate Virtual Private Networks). As depicted in Figure 1, a UMTS network consists of three components – the User Equipment (UE), the Radio Access Network (RAN), and the Core Network. Based on the radio interface technology, the UMTS system can be further categorized as (1) the WCDMA (Wideband Code Division Multiple Access), which is the original system with largest deployment, (2) the newer TD-SCDMA (Time Division Synchronous CDMA), and (3) the latest HSPA (High Speed Packet Access) [7], which supports a much higher peak data rate and system capacity and is sometimes referred to as the 3.5G. These UMTS systems share the same core network architecture, which is based on GSM network (2G) with GPRS (General Packet Radio Service). We can further divide UMTS system into circuit switched and packet switched domains. The focus of this study is on the mobile data (Internet) service that utilizes the packet switched core network.

The UMTS packet switch core network consists of SGSN and GGSN. The main function of the packet core is to provide packet routing, traffic management (e.g., load balancing), session authentication and application layer management, and traffic accounting for billing. In the case of uplink traffic (from user devices), all Internet traffic must go through GGSN using GTP based encapsulation. While GTP can run on top of TCP/IP, connectionless messages such as IP datagrams typically use UDP/IP.

In an illustrative example in Figure 2, suppose that a 3G user needs to communicate with a web server C1. The data packet exchanges between the user and C1 would take *R1* in the figure (RNC and SGSN are omitted for brevity). Leaving out the signaling details of session setup, a user data IP packet would go through the corresponding NodeB and RNC (using a different encapsulation protocol called PDCP (Packet Data Convergence Protocol)). At the RNC, the data packet is encapsulated using GTP, or more specifically GTP-U (User plane), and routed through the 3G provider's internal network to a designated SGSN. At the SGSN, the packet switches to a different GTP tunnel and is routed through the 3G provider's network to a designated GGSN. At the GGSN, the GTP header is decapsulated, and the original IP packet is forwarded toward C1 in the wide-area network (the Internet). The return traffic from C1 is also first routed to the GGSN (e.g., via BGP routing) and then follows the reverse path/process to reach the UE.

B. CDN and DNS-based Request Routing

Our study investigates the interaction between 3G network and CDN service, which currently serves a large portion of web content. At a high level, CDN service improves user's web performance (e.g., reducing latency, increasing data throughput) by bringing the content closer to the users (eyeballs). A CDN provider typically replicates the web contents on behalf of content providers across multiple diversely-connected and geographically-distributed servers, and use *request routing* to direct users' web requests to *appropriate* servers (e.g., based on proximity) to improve service performance [10], [23]. The

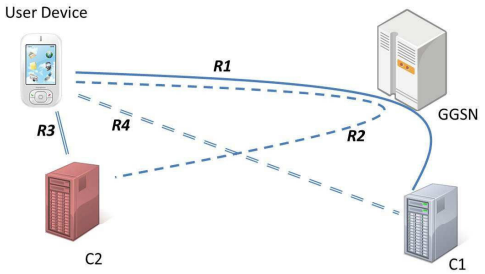


Fig. 2. Various scenarios for hierarchical and flat routing. *R1* and *R2* correspond to hierarchical routing, and *R3* and *R4* correspond to flat routing.

most widely adopted approach to such request routing is through DNS lookup [23]. By examining the source IP address of a received DNS lookup request, which is the IP address of the local DNS server used by the user, a CDN provider can respond with a web server that is close (from the network routing perspective) to the IP address. This is based on the assumption that a client host would typically set up the DNS server in its local network and hence a web server being close to the local DNS server would also be close to the user.

Note that not all web requests involve a DNS lookup — DNS lookup results can be cached in the local DNS servers, the client host's operating system, or the user's web browser. To ensure more dynamic request routing, most CDNs set the TTL (Time-to-Live), i.e., the expiration time, for DNS entries to small values.

III. Performance Implications of Hierarchical Routing

In this section, we illustrate how hierarchical routing in 3G may potentially affect service performance.

A. Hierarchical Routing vs. Flat Routing

As mentioned in the previous section, all 3G traffic must go through GGSN, which is an endpoint of a GTP tunnel. Compared to the direct shortest path from the UE and the destination, such routing may cause the traffic to travel extra distance, which can result in higher network resource usage and lower user performance (such as longer delay and lower throughput). Figure 2 illustrates the scenario. A user tries to communicate with destination C1. The actual route *R1* for 3G traffic can potentially be significantly longer than a shortest path *R4*, which flat routing without any detour points would take. We quantify this extra distance and its service impact in Section IV-B.

B. Hierarchical Routing and Replicated Service

The impact of a detour on service performance can become even worse when a service is provided through replicated and widely-distributed resources. This includes web requests that can be served by multiple replicated servers from a CDN provider, video editing applications (e.g., YouTube) that can be serviced by any server in a distributed infrastructure, or gaming that is hosted at distributed game servers. Network latency to the end users is often a crucial factor for the service performance, which has motivated the service replication in the first place. For example, whether one can connect to a close-by first-shooter game server can significantly impact the gaming experience. In the rest of the paper, we will use CDN as a representative for this class of service for the convenience of presentation, while our findings are applicable to all services in this class.

In Figure 2, suppose that the user is requesting a web page that is also replicated in C2, which would be the best server for the user without the detour. Note that the performance that the 3G user experiences is likely to improve over the non-replication case, since she can still use a better server between C1 and C2 rather than a fixed server in a non-replicated scenario. However, when compared to the flat routing case, the relative performance penalty due to the detour can potentially increase with more CDN locations (e.g., using *R1* instead of *R3*). We evaluate this aspect in more detail in Section IV-C.

C. Interaction with Application Layer

Suppose a 3G user visits a website replicated at multiple CDN servers. When the user device initiates a DNS lookup using its local DNS server, it typically finds a server (C1) close to the GGSN that the user device is connected to. It is because the local DNS server is usually co-located with GGSN, and the CDN's DNS server returns a server close to the local DNS [23]. As previously explained, the traffic will take path *R1*. Later when WiFi becomes available, the user device performs a vertical handoff and switches to WiFi. After the switching, the user device should be able to utilize a better located server (C2), which is close to the new local DNS server for the WiFi network and the user's traffic can take the direct route *R3*. However, it is possible that the user device keeps using the old server address in its DNS cache and hence experiences sub-optimal performance using route *R4*. Similarly, such sub-optimal web access can occur when the user switches from WiFi to 3G (i.e., using *R2* instead of *R1*), for which case we omit the detail for brevity.

Although in practice, CDNs usually use small TTL values for DNS entries in order to keep users from using stale information, we find that many web browsers ignore TTL values and continue to use the old DNS entries for a certain time period [20], which can cause significant performance degradation. We present our findings from a measurement study in more detail in Section V.

IV. Trace-driven Analysis

In this section we quantify the performance implications described in Section III, using traces obtained from a large-scale 3G network operator and a CDN provider in the United States. We first describe data sources and performance metrics used in our study.

A. Metrics and Data Sets

We quantify the difference between routing schemes due to the two types of detour described in the previous section. In addition to the current routing strategy where all traffic goes through a GGSN (called EAG (Exit-at-GGSN)), we also study the performance when we allow traffic to exit at

corresponding SGSN or RNC (called EAS and EAR, respectively).[1] Since it is difficult to arrange these unconventional routing schemes and perform a large-scale active measurement study, we mainly focus on analysis using location information and traffic volume data. We complement our analysis with a measurement study in Section V.

In our evaluation, we mainly use the straight-line distance between two points (called *air mile* [15]) as a metric to quantify the difference between different routing strategies. In case of detour, we use the sum of distances for multiple line segments. For instance, for EAG, we consider three line segments: RNC to SGSN, SGSN to GGSN, and GGSN to server. (We ignore the distance between NodeB and RNC, which is relatively small.) It is well accepted that air mile has strong correlation with actual performance (e.g., delay) and is one of the primary performance metrics for CDN providers [10]. In addition, using that metric allows isolating the performance comparison from other external factors such as underlying topologies, routing algorithms, and traffic engineering schemes. Still, we present the results that illustrate how the difference in air mile would affect the end-to-end performance later (Section IV-B2).

The data we use in this part include:

- Location information of 3G network entities (i.e., RNCs, SGSNs, GGSNs) and CDN data centers. In our study, we use thousands of RNCs, hundreds of SGSNs, tens of CDN data centers, and tens of GGSNs. [2] We also use a hierarchical topology that connects the 3G network entities.
- Packet count for upstream and downstream traffic at each RNC for a week. We use this as weight when computing the per-packet average air mile.
- Periodic probe data for end-to-end latency. This is obtained from more than 250 probe devices that contact remote servers. We also use the location information of probe devices and remote servers.

In our study, we assume that all CDN locations are equivalent and host the same set of content, unless otherwise stated. Depending on the exit point in different routing strategies (e.g., EAG, EAR), the traffic may go to different CDN locations. In our evaluation, unless otherwise stated, we use the closest CDN location (in terms of air mile) from the exit point. We define C(L) to be the CDN server chosen for location L. Since the amount of extra air mile is closely related to the number of GGSN locations, we further study how the performance changes with a varying number of GGSN locations.

B. Hierarchical Routing vs. Flat Routing

1) Air Mile vs. Number of GGSN locations: We first investigate the air mile that 3G user traffic needs to travel on average as a function of GGSN locations. More GGSN locations would make the system increasingly *flat*. We consider scenarios where 3G users access replicated Internet service (e.g., content server replicated by CDN service) in the next subsection. While there can be many aspects to consider in selecting GGSN locations (e.g., physical space, cost, peering connectivity), we focus on minimizing the total air mile between RNC and GGSN.

We model this problem as a k-median problem [22], where we consider all existing RNC locations as potential locations of GGSNs and use as metric the weighted air mile using the traffic volume at the RNC. Thus in the extreme case when there is one GGSN co-located with each RNC, the objective function will drop to 0, which is then equivalent to the EAR case. Finding an optimal solution to a k-median problem is NP-hard [22]. In our experiments, we use the following heuristics:

- *Greedy* algorithm tests all available locations and selects the one with maximal decrease in the objective function [22]. The procedure repeats until the required number of locations is reached.
- *k-means* [19] starts with randomly selected k centers. Then, each point is assigned to the closest center, and within each cluster, a new center is selected that minimizes the sum of costs in the cluster. The procedure is repeated until the objective function does not change.

We consider two cases for the greedy algorithm. The first one starts from the scratch without any prior GGSN location. The second one starts with 4 prior GGSN locations and then finds best locations for subsequent new GGSNs.[3] For k-means, the result varies significantly depending on the initial set of randomly selected centers. For each k, we run k-means 10 times with different initial centers and use the best result. We also compare the heuristic solutions against a *lower bound*, which we find by relaxing the integer solution constraint and allowing fractional solutions (often called linear-program relaxation).

We report the weighted average air mile in Figure 3. We observe that in all cases, having more GGSN locations reduces the air miles significantly. For example, the average air mile is more than 800km with 2 GGSN locations, while it is around 400km with 5 locations. The performance of k-means and greedy algorithm is highly close to the lower bound (within 7.5% and 15%), although k-means performs slightly better. While the performance of 4 initial GGSN locations is quite poor (e.g., 18% worse than k-means), it quickly converges to that of k-means as we add more GGSN locations. Interestingly, the benefit of adding more GGSN locations slows down after 7 or 8 locations. This in part supports the design of using a small number of GGSN locations.

2) Translating Air Mile to End-to-end Performance: In this subsection, we estimate the impact of air mile changes on end-to-end performance. Specifically, we study the correlation between air mile and round trip time (RTT) by analyzing probe data obtained from specialized devices scattered across 250 different locations in the continental US. Among these devices, approximately 70 use 3G and 180 use HSPA (High

[1] Existing products can realize these approaches [5].

[2] The exact numbers are proprietary information, which we do not provide here.

[3] We use the four most populated cities in the United States: New York, Los Angeles, Chicago, and Houston. We also experimented with different initial placements and observed a trend similar to Figure 3, even though the initial placements are worse than the one shown in the figure.

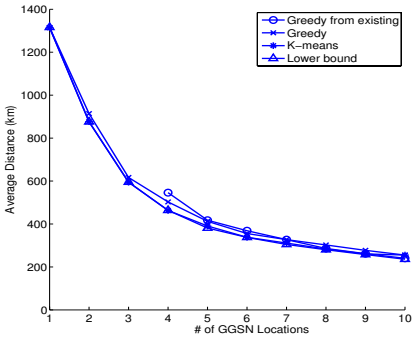
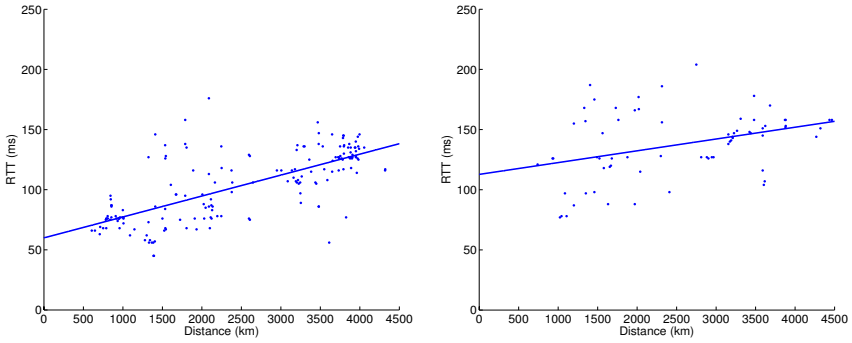


Fig. 3. Change in air mile when we vary the number of GGSN locations.



(a) HSPA (b) 3G

Fig. 4. Air mile - Latency Correlation. Dots: measurement points. Lines: best line fitting.

Speed Packet Access). Each probe device performs one or two ping tests per hour, and the ping destinations consist of both internal servers (maintained by the 3G operator) and external servers. The probe data contains the location information for the probe devices and internal servers. For the external servers, we obtain the location using a publicly available ip2location database [3]. We consider the detour routing via SGSN and GGSN and use the distance sum of the following three segments: probe-SGSN, SGSN-GGSN, and GGSN-destination. For each source-destination pair, we use the minimum RTT of all the measurements conducted during a day to minimize other external impact (e.g., link congestion). We have also analyzed the data for multiple days and found the result from the one-day data is representative. We have also examined the download throughput metric as a function of air mile distance and observed weaker correlation (which is not presented due to space limit), probably because the air interface is the main throughput bottleneck in current 3G systems.

Figure 4 shows the correlation between air mile and RTT. As expected, the latency of HSPA is mostly lower than that of 3G. We also observe positive correlation between air mile and latency for both HSPA and 3G. We performed a line fitting, and the least squares fitting for HSPA is $y = 0.017x + 59.9$ and for 3G $y = 0.010x + 112.7$. The fitting for the 3G is less accurate (the mean squared error for 3G is 684.8 vs. 400.4 for HSPA; the 95% confidence interval for the slope of the 3G fitting is [0.0039,0.0161] and for the HSPA fitting is [0.0144,0.0196]) in part because of the fewer data points. Note that the constant values for both cases roughly correspond to the average radio link delay values (75ms for HSPA and 125ms for 3G) reported in a recent measurement study [27]. We further discuss the trend and its implication of the routing independent factors in Section VI. The slope of the HSPA fitting is also consistent with the result in a previous study performed on the wired Internet users [9].

C. Hierarchical Routing and Replicated Service

Now we consider scenarios where 3G users access a replicated service and show that the difference between *flat* routing and hierarchical routing is much more significant in such a scenario.

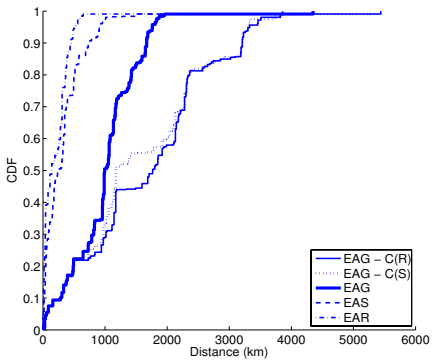


Fig. 5. CDF of air mile for different routing strategies

We first compare three schemes (EAG, EAS, and EAR) where the exit points are different. In our analysis, we use the nearest CDN server from the exit point (in terms of air mile). In Figure 5, the three leftmost lines correspond to the CDF of per-packet air mile for these schemes. (The other two lines are discussed in Section IV-D.) From this figure, we observe that bypassing GGSNs and exiting at an earlier network element such as RNC and SGSN can lead to significant benefit in terms of air mile reduction. Specifically, the difference in median between EAG and EAR is 851km, while the difference between EAG and EAS is 737km. The average of three schemes (EAG, EAS, and EAR) are 1009km, 338km and 226km, respectively. Using the result in Section IV-B2, we can estimate the impact due to longer air mile on end-to-end latency. For example, in HSPA, the difference in median between EAR and EAG translates to 14ms (or 23.7%) difference in RTT.

In the next set of experiments, we vary the number of CDN locations and evaluate the performance penalty due to detour. Given N, we select a subset of N CDN locations uniformly at random and calculate the air mile when users retrieve content only from those CDN locations. We run 10 experiments for each N and report the result in Figure 6. In this figure, we use the median of 10 median values from different runs while the error bar shows the maximum and minimum of the median values.

In Figure 6, we first observe that as the number of CDN

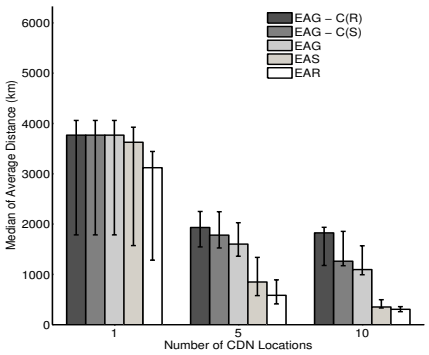


Fig. 6. Extra air mile vs. Number of CDN locations

locations increases, the air mile decreases in general. However, the penalty due to detour actually increases with more CDN locations. For example, when there is only one CDN location, the medians of EAG and EAR are 3769km and 3121km (or 20% difference), while with 10 locations, the numbers are 1096km and 306km (or a factor of 3+ difference). We also note that allowing early exits is quite effective in terms of reducing air mile and potentially end-to-end latency. Specifically, in Figure 6, the air mile performance of EAS at N=5 is mostly better than that of EAG at N=10 (849km vs. 1096km in the median case). This indicates that deploying more GGSNs or augmenting the 3G network communication architecture (e.g., allowing bypass [5]) is likely to be more beneficial to the web service performance of 3G users than deploying more CDN locations.

D. Interaction with DNS Caching

We now consider cases of using sub-optimal CDN servers due to vertical handoff and DNS caching (Section III-C). Specifically, we assume that due to cached DNS entries before a vertical handoff, web requests continue to go to a CDN server close to the RNC or the SGSN along the path (denoted by C(R) or C(S)) while the traffic exits at a GGSN.

The two rightmost lines in Figure 5 correspond to these cases. Not surprisingly, the resulting routes are even worse than that of EAG, and the air mile difference between EAR and EAG-C(R) in terms of median is over 1600 km. Based on our line fitting for HSPA in Section IV-B2, this translates to about 27ms difference (45.4%) in median RTT between EAR and EAG-C(R). The increase in the worst case is more pronounced. For example, in the 90th-percentile case, the difference is as large as 3211km. We also report the mean of the two rightmost lines, which are 1585km and 1676km, respectively. In Figure 6, the two leftmost bars for each N show the performance of the corresponding scenarios. We observe that the penalty due to using an incorrect CDN server (e.g., difference between EAG-C(R) and EAG) increases with more CDN locations, while there is no difference for N=1 as expected.

V. Measurement

We next use measurement experiments to validate and quantify the performance impact of using suboptimal CDN servers due to vertical handoff and DNS caching.

A. Measurement Setup

We have performed the measurement on a laptop running Microsoft Windows Vista. We use an internal WiFi interface on the laptop and an external USB 3G card by Sierra Wireless. In our measurement, we focus on the download throughput of web contents replicated by CDN providers. In particular, we identify a list of content providers using Akamai and Limelight and randomly select a subset of them to retrieve their web contents. As described in Section II, a CDN provider typically assigns one of the replicated servers to a web client when responding to a DNS lookup. We define C(W) to be the CDN server returned for a DNS lookup from the WiFi interface. We also define C(3G) similarly.

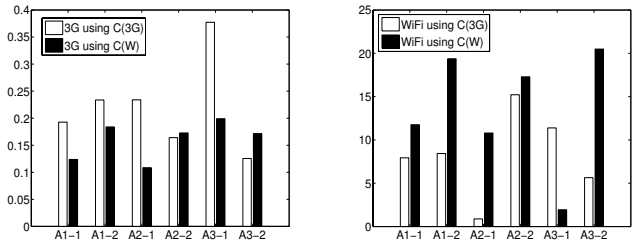
Although the TTL value of a DNS record stored in a DNS cache might be small, we observe that many web browsers have their own timeout value for their DNS cache entries, in an attempt to reduce the number of DNS requests. The default value can be up to 30 minutes depending on the browser.[4] To emulate vertical handoff, we manually switch between 3G interface and WiFi interface. Then, to emulate a user continuing to access the same content, we request web contents from the same CDN server using a web browser without closing it. To ensure downloading the content from a CDN server, we clear the web cache on the browser after a download is complete. We use *Wireshark* to capture all traffic and analyze the performance. All contents used in this study are relatively large (e.g., several-minute long video clips), with sizes ranging from several MB to 45MB. We stop a download if it takes more than 5 minutes. We performed all our measurement in August 2010.

We use the average download throughput as metric and compare the four cases: WiFi interface downloading from C(W), WiFi interface downloading from C(3G), 3G interface downloading from C(W), and 3G interface downloading from C(3G). In the rest of this section, we mainly report results using Microsoft Internet Explorer (IE, version 8.0), which has the largest market share [1]. We also briefly report the result using other web browsers.

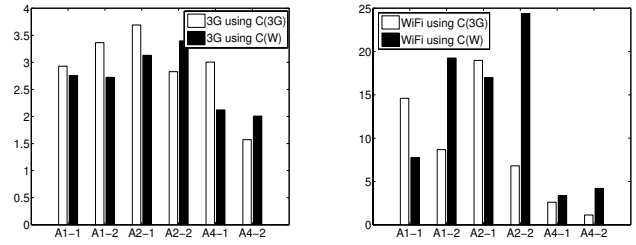
B. Measurement Result

We measured the throughput for four Akamai customers sites (A1, A2, A3, A4) from three different indoor locations: a corporate office in New Jersey, and two university campuses in Massachusetts and Texas. For a given customer, we perform the download twice and report the result of both instances. We use A1-1 and A1-2 to distinguish the two experiments for customer A1. The results are shown in Figure 7. Not surprisingly, the WiFi throughput is significantly higher than that of 3G. Also, when on the WiFi interface, using C(W) servers results in higher performance in most cases than using C(3G) based on an old DNS record obtained through the 3G interface. Specifically, in Figure 7(a), except for one case (A3-1), the WiFi throughput using C(W) servers is consistently higher than the case of using C(3G) servers, and the difference can be up to a factor of 13. We observe a similar trend in the other locations.

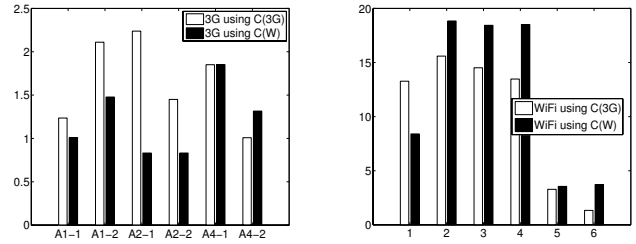
[4]http://support.microsoft.com/kb/263558



(a) New Jersey



(b) Massachusetts



(c) Texas

Fig. 7. Throughput Measurement on Akamai customers (Mbps)

In all three sets of measurement, we notice that the results are not always consistent, and using C(3G) servers sometimes results in higher throughput even when the user is on WiFi. There are multiple possible reasons. First, the download throughput is decided by many other factors besides the delay to CDN server, such as background traffic, interference, congestion and even channel state transtion [21]. Seccond, delay is not the only metric used for CDN server selection, but other factors such as load balancing can also affect the CDN server selection [23], in which case the selected server may not give the best performance.

When we use the 3G interface, the difference is less pronounced although using C(3G) servers results in slightly better performance. We believe that this is because in the current 3G networks, the radio link is the main bottleneck and dominates the performance, and thus the detour in the wired Internet has less impact on the end-to-end performance. We further discuss this aspect in Section VI.

We have also performed a similar measurement on Limelight CDN customers and found the difference quite marginal. Unlike Akamai, Limelight has fewer locations [4]. Instead, to achieve high performance, Limelight takes advantage of good peering connectivity of their network that connects their locations. Hence both the chance of hitting a sub-optimal CDN server (location) and the performance penalty of using a sub-optimal one become small as previously discussed in Section IV-C (Figure 6).

All the measurements reported above are done on IE. We also experimented with other popular browsers. We find that the behavior of Firefox (version 3.6.7) is similar to that of IE in that it keeps using the cached server after switching to another wireless interface, although its timeout value is shorter (1 minute). Safari (version 5.0) automatically does a new DNS look up using a new DNS server after a switch between interfaces.

Our results highlight the implication of interaction between vertical handoff and DNS caching on the real-world performance and show that the performance degradation can be an order of magnitude. Although the performance degradation is possible only during a relatively short window, other optimization techniques such as DNS prefetching [2] can possibly cause the issue to occur more frequently. In general, network system designs should consider such cross-layer interactions especially when more frequent vertical handoffs are possible in the future.

VI. Discussion

In this section we discuss how the performance issues identified in this paper can potentially have a larger impact on the service performance in future wireless networks.

In Section IV-C, we find that all traffic going through one of the GGSNs can result in up to an order of magnitude increase in air mile. Specifically, the median distance of EAR is 142km and that of EAG is 993km. However, based on the result in Section IV-B2, the difference in end-to-end latency is only modest (23.7% for HSPA), and the throughput difference is even smaller. This is because in the current 3G networks, air interface is the main bottleneck, contributing significant delay and limiting the throughput. However, as wireless technologies continue to improve, the performance difference due to traffic detour will likely grow larger. In fact, we can already see this trend in our results. For the same air mile, HSPA already shows significantly smaller end-to-end latency than 3G (Figure 4). As radio access technology further improves (e.g., $\sim$10ms in LTE [27]), the routing dependent delay would account for an even greater portion of the total delay. Also, we observe more noticeable performance degradation when a WiFi user gets stuck with a sub-optimal CDN server (Section V-B). Based on this evidence, we believe that as the wireless technology improves further, and the air link becomes less of a bottleneck, inefficient routing in wireless networks will have a larger impact on the overall performance.

The use of aggregation points (i.e., GGSN equivalents) for data traffic still applies to the recent cellular architecture such as EPC (Enhanced Packet Core) [6]. While the EPC is considered a *flat* architecture, and the network element in the cell tower (called eNodeB–enhanced NodeB) is IP-capable, the standard still requires all data traffic to go through gateway (called PDN Gateway). However, some operators may choose to use proprietary products in the existing architecture [5]. Also a new standard is being developed to allow Internet traffic to bypass the operator's core network [8]. Moreover, future wireless network architecture may allow wireless operators to

add GGSN-equivalent entities more easily. These scenarios will help reduce the amount of detour due to hierarchical routing, and the result in this work can provide a guideline to the design of such new systems.

VII. Related Work

Measurement on 3G Networks: Many measurement works on 3G networks have been done to obtain a better understanding of 3G networks and to identify possible performance problems. Ricciato et al. [26] investigate two approaches to infer the presence of a capacity bottleneck from passive measurements in a 3G mobile network. Tan et al. [29] find that the performance of 3G network varies widely across different operators. More recently, Huang et al. [16] perform a measurement study using smartphones on a number of 3G networks and present the application performance such as web browsing, VoIP, and video streaming. Our work is different in that we focus on the potential performance penalty due to the 3G standard architecture.

Optimizing 3G Networks: Another related area of research is to optimize 3G networks from the network planning perspective. Ricciato [24] describes how passive observation of network traffic and traffic-analysis methods can be used to optimize 3G networks. Ricciato et al. [25] perform a measurement study to understand an optimal assignment of base stations to SGSNs. Amaldi et al. [11] investigate the problem of optimizing the base station location and configuration in 3G networks. Wu and Pierre [30] propose a constraints-based model to find the best sites for RNCs. While we consider the problem of finding GGSN locations in Section IV, our work focuses on how the 3G packet core architecture interacts with Internet services such as CDNs.

WiFi Offloading: Recently, there has been huge interest in augmenting 3G capacity by using WiFi access (called *WiFi offloading*). Balasubramanian et al. [14] propose a prediction-based offloading system that can augment 3G capacity using WiFi. Lee et al. [18] collect data traces from around 100 smartphone users and provide quantitative analysis results about how beneficial WiFi offloading would be. Our work finds that the interaction at the application layer is also an important aspect in determining whether to switch between 3G and WiFi.

VIII. Conclusion

In this paper we have studied the performance impact of the current 3G standard architecture on Internet service access. Specifically, we have compared hierarchical routing in 3G networks with idealized *flat* routing. While many Internet services are provided through replicated and widely-distributed resources (such as CDNs), our trace-driven analysis results show that the relative 3G performance compared to the idealized case degrades when a 3G user accesses highly replicated service. We have also demonstrated that user mobility and web applications can lead to unexpected performance-impacting interactions in practice. Our findings suggest that future Internet services and applications for mobile broadband users should consider the network architecture and performance implications to provide high service quality.

References

[1] Browser Market Share. http://marketshare.hitslink.com.

[2] DNS Prefetching. http://www.chromium.org/developers/design-documents/dns-prefetching.

[3] IP2Location.com. http://www.ip2location.com.

[4] Limelight Networks Overview. http://www.limelightnetworks.com/resources/LLNW_Network_Overview.pdf.

[5] Mobile data offload solution. http://www.stoke.com.

[6] *Long Term Evolution (LTE):A Technical Overview*. LTE White paper, Motorola, 2006.

[7] *Technology of High Speed Packet Access (HSPA)*. White paper, Nomor Research, 2006.

[8] *Local IP Access and Selected IP Traffic Offload*. 3GPP TR 23.829, 2011.

[9] S. Agarwal and J. R. Lorch. Matchmaking for online games and other latency-sensitive P2P systems. In *Proceedings of ACM SIGCOMM*, 2009.

[10] H. A. Alzoubi, S. Lee, M. Rabinovich, O. Spatscheck, and J. Van der Merwe. Anycast CDN revisited. In *Proceedings of WWW*, pages 277–286, 2008.

[11] E. Amaldi, A. Capone, and F. Malucelli. Radio planning and coverage optimization of 3G cellular networks. *Wirel. Netw.*, 14(4):435–447, 2008.

[12] D. Applegate, A. Archer, V. Gopalakrishnan, S. Lee, and K. K. Ramakrishnan. Optimal content placement for a large-scale vod system. In *Proceedings of CoNext*, pages 4:1–4:12, 2010.

[13] M. Balakrishnan, I. Mohomed, and V. Ramasubramanian. Where's that phone?: geolocating IP addresses on 3G networks. In *Proceedings of Internet Measurement Conference (IMC)*. ACM, 2009.

[14] A. Balasubramanian, R. Mahajan, and A. Venkataramani. Augmenting mobile 3G using WiFi. In *Proc. of MobiSys*, 2010.

[15] J. Erman, A. Gerber, M. T. Hajiaghayi, D. Pei, and O. Spatscheck. Network-aware forward caching. In *Proceedings of WWW*, pages 291–300, 2009.

[16] J. Huang, Q. Xu, B. Tiwana, Z. M. Mao, M. Zhang, and P. Bahl. Anatomizing application performance differences on smartphones. In *Proceedings of MobiSys*, pages 165–178, 2010.

[17] H. Kaaranen, A. Ahtiainen, L. Laitinen, S. Naghian, and D. V. Niemi. *UMTS Networks: Architecture, Mobility and Services*. Wiley, 2nd edition, 2005.

[18] K. Lee, I. Rhee, J. Lee, Y. Yi, and S. Chong. Mobile data offloading: how much can wifi deliver? *SIGCOMM Comput. Commun. Rev.*, 40(4):425–426, 2010.

[19] S. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129 – 137, 1982.

[20] J. Pang, A. Akella, A. Shaikh, B. Krishnamurthy, and S. Seshan. On the responsiveness of dns-based network control. In *Proceedings of IMC*, pages 21–26, 2004.

[21] F. Qian, Z. Wang, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck. Profiling resource usage for mobile applications: a cross-layer approach. In *Proceedings of Mobisys*, 2011.

[22] L. Qiu, V. N. Padmanabhan, and G. M. Voelker. On the placement of web server replicas. In *Proc. of IEEE INFOCOM*, pages 1587–1596, 2001.

[23] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs. Cutting the electric bill for internet-scale systems. In *Proceedings of ACM SIGCOMM*, pages 123–134, 2009.

[24] F. Ricciato. Traffic monitoring and analysis for the optimization of a 3G network. *Wireless Communications, IEEE*, 13(6):42 –49, dec. 2006.

[25] F. Ricciato, R. Pilz, and E. Hasenleithner. Measurement-based optimization of a 3G core network: A case study. *IEEE Wireless Communications*, page 49, 2006.

[26] F. Ricciato, F. Vacirca, and P. Svoboda. Diagnosis of capacity bottlenecks via passive monitoring in 3G networks: an empirical analysis. *Computer Networks*, 57:1205–1231, 2007.

[27] C. Serrano, B. Garriga, J. Velasco, J. Urbano, S. Tenorio, and M. Sierra. Latency in broad-band mobile networks. In *Vehicular Technology Conference.*, pages 1 –7, 2009.

[28] M. Stemm and R. H. Katz. Vertical handoffs in wireless overlay networks. *Mob. Netw. Appl.*, 3(4):335–350, 1998.

[29] W. L. Tan, F. Lam, and W. C. Lau. An empirical study on the capacity and performance of 3G networks. *IEEE Transactions on Mobile Computing*, 7(6):737–750, 2008.

[30] Y. Wu and S. Pierre. Optimization of access network design in 3G networks. *Canadian Conference on Electrical and Computer Engineering*, 2:781 – 784 vol.2, may. 2003.