Peering vs. Transit: Performance Comparison of Peering and Transit Interconnections

Adnan Ahmed and Zubair Shafiq
The University of Iowa

Harkeerat Bedi and Amir Khakpour
Verizon Digital Media Services

***Abstract*—The economic aspects of peering and transit interconnections between ISPs have been extensively studied in prior literature. Prior research primarily focuses on the economic issues associated with establishing peering and transit connectivity among ISPs to model interconnection strategies. Performance analysis, on the other hand, while understood intuitively, has not been empirically quantified and incorporated in such models. To fill this gap, we conduct a large scale measurement based performance comparison of peering and transit interconnection strategies. We use JavaScript to conduct application layer latency measurements between 510K clients in 900 access ISPs and multi-homed CDN servers located at 33 IXPs around the world. Overall, we find that peering paths outperformed transit paths for 91% Autonomous Systems (ASes) in our data. Peering paths have smaller propagation delays as compared to transit paths for more than 95% ASes. Peering paths outperform transit paths in terms of propagation delay due to shorter path lengths. Peering paths also have smaller queueing delays as compared to transit paths for more than 50% ASes.**

I. Introduction

Background. The traffic dynamics and the revenue models of different stakeholders in the Internet content delivery ecosystem have profoundly changed over the last decade [36]. On one hand, a few large content providers and content delivery networks (CDNs), particularly the ones serving video, now originate most of the Internet traffic [6]. For example, Netflix alone accounts for more than 35% of downstream Internet traffic during peak period in North America [7]. On the other hand, a few large access ISPs (or "eyeball networks") provide Internet access to a majority of consumers. For example, Comcast alone serves more than half of the U.S. broadband market [39]. Internet's topology has changed from a hierarchical structure to a flatter structure in order to accommodate these changes [23], [19]. Unlike the hierarchical tiered Internet, where access ISPs acquired global Internet connectivity from transit providers, more and more ISPs are engaging in peering relationships with bilateral traffic exchange agreements [29], [41]. This transition from a transit hierarchy to a peering mesh is facilitated by Internet Exchange Points (IXPs), which allow a large number of Autonomous Systems (ASes) to interconnect with each other [9].

Motivation. Economic and performance benefits have played a key role in the widespread adoption of peering [36]. From an economic perspective, peering reduces upstream transit costs for access ISPs. From a performance perspective, peering is expected to improve network quality-of-service (QoS) due to shorter paths. Since content providers primarily generate revenue through advertisements and subscriptions, peering with access ISPs to improve user experience makes sense for them. Prior literature assumes that peering improves performance based on anecdotal evidence. Peering and transit relationships between ASes are modeled in light of these assumed performance improvements [35], [33], [15]. However, to the best of our knowledge, an empirical performance comparison of peering and transit interconnections is lacking in prior literature.

Problem Statement. In this paper, our goal is to measure, characterize, and compare the performance of peering and transit interconnections. Specifically, we aim to empirically answer the following questions about peering and transit paths between content providers and access ISPs. (1) Are peering paths better in terms of performance than transit paths? (2) If yes, how much performance improvement can be expected when a content provider peers with an access ISP? (3) Is there more to peering, apart from shorter paths, that contributes to improved performance?

Technical Challenges. We need to overcome several technical challenges to empirically compare and contrast performance of peering and transit. First, we need to conduct simultaneous measurements over peering and transit paths in order to compare their performance. Second, we need geographically distributed vantage points to analyze peering relationships at different IXPs around the world. Third, we need to be able to scale our measurements to a large number of ASes across the Internet. Finally, we need our measurements to be lightweight so they do not overwhelm the underlying network and negatively impact user experience.

Proposed Approach. To overcome these challenges, we collaborate with a commercial content delivery network (CDN) that has a large geographic footprint. The CDN has presence at more than 30 different IXPs where it has peering and transit connectivity with hundreds of different ASes. We use JavaScript to conduct simultaneous performance measurements between end-users in access ISPs and CDN servers at IXPs over peering and transit paths. More specifically, we embed our JavaScript code in webpages requested by end-users which, when loaded, prompt the browser to conduct end-to-end performance measurements to a CDN server at an IXP via both peering and transit paths. To keep our performance

978-1-5090-6501-1/17/$31.00

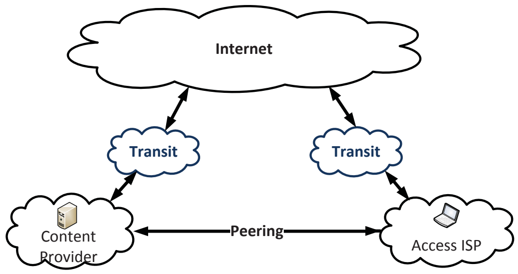


Fig. 1. Illustration of peering and transit interconnection strategies

measurements lightweight, we calculate the download time of a 1x1 transparent pixel tag from the CDN server to end-user. Since the CDN serves content that is requested by a large number of end-users, we are able to scale our measurement study to hundreds of thousands of end-users located in different ASes around the world.

Key Contributions. We summarize our key contributions and findings as follows.

• In collaboration with a large commercial CDN, we conduct simultaneous performance measurements between 510K clients in 920 access ISPs and CDN servers at 33 IXPs via peering and transit paths. Overall, our measurements show that peering paths substantially outperform transit paths. More specifically, we note at least 5% improvement in end-to-end latency over peering paths as compared to transit paths for more than 91% ASes. In contrast, we find that only 2% ASes get at least 5% improvement in end-to-end latency over transit paths as compared to peering paths.

• We estimate propagation and queueing delays on peering and transit paths by analyzing temporal variations in end-to-end latency measurements. We note that peering paths exhibit at least 5% improvement in propagation delay as compared to transit paths for 95% ASes. We observe more than 10 milliseconds improvement in propagation delays on peering paths as compared to transit paths for 68% ASes. To further analyze differences in propagation delays, we conduct traceroute measurements from CDN servers at IXPs to access ISPs via peering and transit paths. As expected, we note that peering paths are shorter than transit paths. More specifically, median path length between CDN servers and end-users in access ISPs over peering paths is 6 IP hops, which increases to 8 IP hops for transit paths. Similarly, median path length is 2 AS hops over peering paths and 3 AS hops over transit paths. The disparity in path length explains why peering paths have smaller propagation delays as compared to transit paths.

• We note that peering paths have at least 5% less queueing delays than transit paths for 57% ASes. In particular, we note up to 450 milliseconds improvement in queueing delays over peering paths as compared to transit paths for 20% ASes. However, unlike propagation delays which are equal or better on peering paths as compared to transit paths, we note at least 100 milliseconds degradation in queueing delays on peering paths as compared to transit paths for 2% ASes. However, a vast majority of ASes still get better overall performance over peering paths than transit paths.

II. Background

Content publishers often rely on CDNs to optimize the delivery of their content to end-users. CDNs leverage their geographically distributed network of cache servers to bring content close to end-users. They redirect content requests to a suitable cache server based on geographic proximity through DNS redirection or IP anycast. Two common strategies for CDN cache server deployment are *enter-deep* and *bring-home* [22]. In the enter-deep strategy, CDNs deploy small sets of cache servers inside a large number of ISPs to maximize their geographic footprint. For example, Akamai, operates more than 200K servers across more than 1,400 ISPs around the world [1]. In the bring-home strategy, CDNs deploy large clusters of cache servers at a few locations (commonly near IXPs), where they interconnect with a large number of ISPs. For example, Limelight operates thousands of servers at a few dozen locations [22]. Note that some CDNs deploy a hybrid infrastructure by combining both strategies. For example, content providers such as Google and Netflix not only operate dozens of bring-home data centers but also operate thousands of enter-deep servers such as Google Global Cache [2] and Netflix Open Connect [4] inside access ISPs.

Content providers and CDNs use different interconnection strategies to send content from cache servers to end-users. To this end, they either buy *transit* connectivity or directly interconnect with access ISPs using *peering* arrangements.

A transit provider is responsible for carrying the traffic between their customer AS and any other AS on the Internet. As illustrated in Figure 1, both access ISPs and content providers use transit providers to carry their traffic. Internet transit prices have steadily declined year-over-year, reaching $\approx$ \$0.45 per Mbps [40]. However, global Internet traffic volume has increased by 40-50% year-over-year as well [6]. Even though transit costs per traffic unit have been decreasing historically, Internet transit bills have increased [36]. Therefore, to reduce their transit costs, CDNs and access ISPs are increasingly engaging in peering relationships.

In peering, as illustrated in Figure 1, ASes directly interconnect with each other for bilateral traffic exchange. While peering relationships are often settlement-free (i.e., cost-free), content providers and access ISPs also sometimes engage in paid peering. This is because some access ISPs impose specific requirements for settlement-free peering with other networks, e.g., minimum traffic volumes, 2:1 traffic ratios, presence at certain IXPs, etc. These requirements have non-trivial operational and economic overheads; for example, a lot of content providers and CDNs cannot satisfy the 2:1 traffic ratio requirements with access ISPs due to the heavily asymmetric nature of video content.

The choice between peering and transit is not only dependent on economic factors but also on potential performance gains. The perception is that peering should improve performance due to shorter paths between cache servers and end-users. However, there is a lack of clear understanding about *whether* peering always outperforms transit and *how much* performance gain can be expected.

III. Proposed Approach

Our goal is to conduct performance comparison of peering and transit between CDN servers and end-users at scale. To this end, we can use throughput measurements that rely on bulk data transfer. However, bulk data transfer would not scale to a large number of users because it can overload the very network infrastructure that we are trying to measure. We can also use latency measurements which are lightweight because they rely on sending a few packets. Latency measurements are generally conducted by sending an ICMP ping echo request packet to the destination and computing the time it takes for the corresponding ICMP echo reply packet to arrive at the source. ICMP measurements are unfeasible for CDNs because: (a) CDNs cannot remotely initiate ICMP measurements from a client's web browser; and (b) a large number of ICMP measurements initiated from CDN's infrastructure can trigger ICMP rate limiting at ISPs [5], [16].

We propose to conduct application layer latency measurements between CDN servers and client browsers. Since CDNs typically embed pixel tags in client-requested web pages for analytics purposes, we can piggyback on these pixel tags to conduct application layer latency measurements. However, it is non-trivial to conduct browser-based application layer latency measurements over peering and transit paths. First, we need to be able to conduct simultaneous latency measurements via both peering and transit paths. This is challenging because we cannot control network layer routing configuration at the application layer in web browsers. Second, we need to ensure that our browser-based application layer latency measurements are accurate. Since browser-based network measurements can incur non-trivial overheads [27], we need to validate the accuracy of application layer latency measurements.

In collaboration with a commercial CDN, we embed IFrames containing our measurement JavaScript in client-requested webpages to carry out end-to-end latency measurements via peering and transit paths. Our measurement script conducts application layer latency measurements between clients and dedicated measurement servers at IXPs. These measurement servers are multi-homed so they can send content to clients via peering and transit paths. Clients requesting content from CDN's peering IP address get response over peering paths. Similarly, clients requesting content from CDN's transit IP address get response over transit paths. Figure 2 provides an overview of our measurement methodology.

- When a client requests a web page from a CDN server, we embed an IFrame in the requested web page. The IFrame contains our measurement JavaScript, which includes the IP addresses of multi-homed measurement servers at different CDN PoPs (points of presence) in IXPs around the world. We probabilistically embed the IFrame in client-requested web pages to avoid overwhelming our measurement servers.
- To figure out the nearest PoP, the measurement JavaScript prompts the client to issue an XHR HTTP GET request for a pixel tag to the nearest PoP. The CDN uses an anycast IP address to host content across all the PoPs. BGP configuration ensures that client requests are redirected to their nearest PoP. The HTTP response headers contain the name of the nearest PoP. Using the PoP name, the client can identify IP addresses of all network interfaces (peering and multiple transit providers) of the multi-homed measurement server.

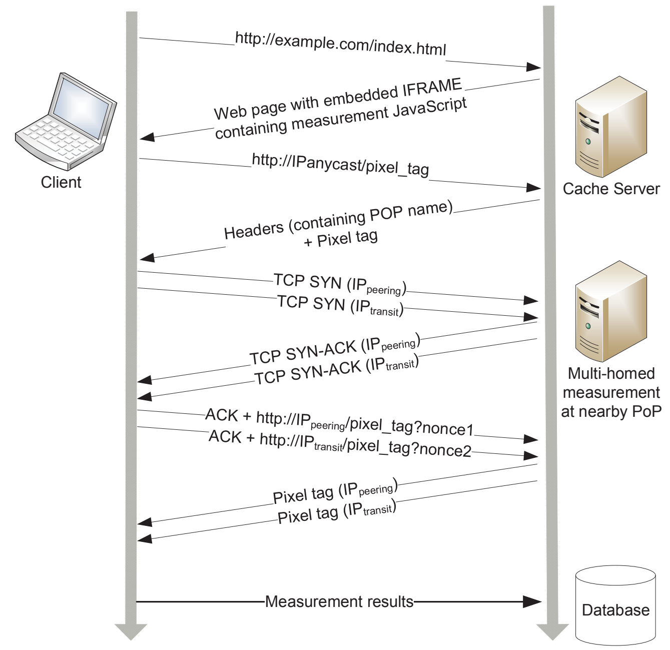


Fig. 2. Measurement methodology to compare performance of peering and transit paths

- The measurement JavaScript is now ready to perform latency measurements via peering and transit paths. The measurement JavaScript inserts multiple new DOM elements to the page using the `<img>` tag, which points to the URLs to download the pixel tag from the multi-homed measurement server via peering and transit paths. Note that IP addresses are hardcoded in the URLs to avoid DNS lookups.
- The client establishes TCP connections over the peering and transit interfaces of the measurement server. This step involves the exchange of SYN and SYN-ACK packets over a time period of 1 RTT. Note that the peering and transit interfaces of the CDN measurement server are part of the same subnet. Therefore, TCP SYN and subsequent packets sent to IP addresses of both peering and transit interfaces are expected to traverse the same path through the network. However, on the way back, CDN routes packets via either transit or peering interconnections based on the IP address of the measurement server's network interface.
- The client sends the HTTP request to the server along with the ACK of the three-way TCP handshake. We add a nonce to the HTTP request to ensure that the tag is served by the CDN measurement server rather than the local browser cache. Moreover, the size of the pixel tag is set

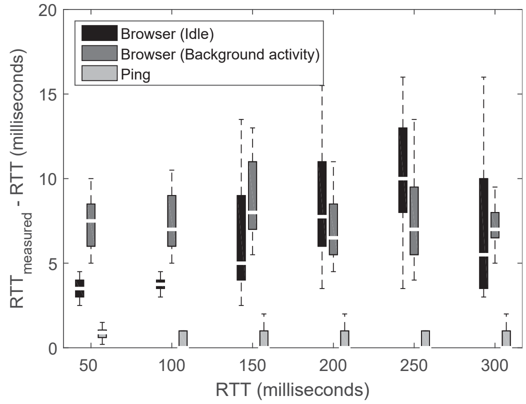


Fig. 3. Comparison of latency measurement between a client browser (with and without background activity) and web server using JavaScript and ping. The x-axis represents the actual RTT set between the client and server while the y-axis plots various percentiles of the difference between the measured RTT and actual RTT values. Note that the difference between JavaScript based measurement and actual RTT is relatively small even on browser with background activity across all RTT values.

to be smaller than the server's TCP congestion window to ensure that the pixel tag is downloaded in approximately one round trip time.

- Finally, the pixel tag is downloaded over peering and transit paths. Note that the total download time is approximately two round trip times (one for TCP handshake and another for HTTP request-response). The JavaScript uploads the pixel tag download time for peering ($RTT_{Peering}$) and transit ($RTT_{Transit}$) paths to a database server.

It is important that the same pixel tag is downloaded *simultaneously* via peering and transit paths—this allows us to quantify the relative performance difference between peering and transit paths instantaneously. To this end, as noted in Figure 2, we send simultaneous HTTP requests over both peering and transit paths.

Next, we validate the accuracy of our latency measurements to mitigate any concerns about browser overheads due to the JavaScript engine [27], [25]. Specifically, we compare application layer JavaScript-based latency measurements and network layer ping measurements in a controlled testbed containing a web server and a client browser set up on two different hosts. We conduct our JavaScript-based latency measurements without background activity as well as with background activity in the Chrome web browser. Specifically, we vary link delays between the web server and browser from 50 milliseconds to 300 milliseconds using the Linux traffic controller and measure RTT using both JavaScript-based methodology and ping. We repeat our measurements 50 times for each link delay and plot the comparison between application layer JavaScript-based latency measurements and network layer ping measurements in Figure 3. The x-axis represents the actual RTT set on the link between client and server, and the y-axis represents the difference between the measured RTT and the actual RTT set on the link. We note that the RTT measured using JavaScript is, at maximum, ≈ 15 milliseconds more than that measured using ping. The median difference between the RTT measured using JavaScript and actual RTT stays below 10 milliseconds across all RTT values. We conclude that our application-layer JavaScript-based latency measurements can provide a reliable latency estimate of the actual RTT between client and server.

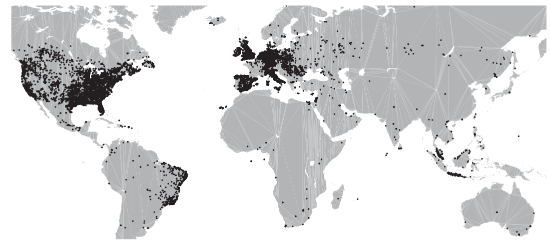


Fig. 4. World map with clients.

PoP Name	# of IP addresses	# of /24 prefixes	# of ASes
Madrid	154,781 (28.5%)	16,094 (5.5%)	25 (1.4%)
New York	36,454 (6.7%)	22,972 (7.9%)	90 (5.0%)
Chicago	34,726 (6.4%)	25,631 (8.8%)	81 (4.4%)
Los Angeles	33,086 (6.1%)	18,902 (6.5%)	71 (3.9%)
London 1	28,883 (5.3%)	23,171 (8.0%)	226 (12.5%)
London 2	28,833 (5.3%)	23,500 (8.1%)	221 (12.2%)
Miami	28,608 (5.3%)	19,555 (6.7%)	33 (1.8%)
Fort Lauderdale	28,605 (5.3%)	19,525 (6.7%)	35 (1.9%)
Everett	28,074 (5.2%)	17,523 (6.0%)	88 (4.9%)
San Jose	25,274 (4.7%)	17,054 (5.9%)	45 (2.5%)

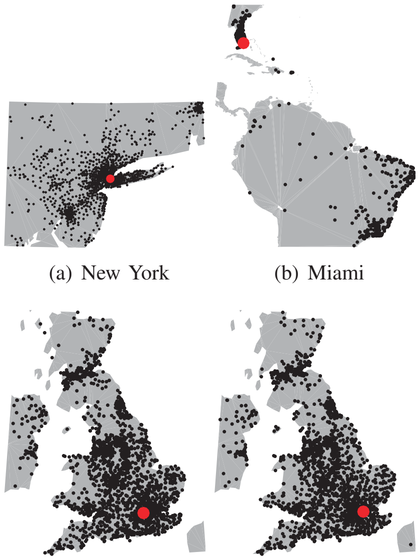
TABLE I
IP ADDRESS STATISTICS FOR THE TOP 10 MOST POPULAR POPS IN OUR DATASET.

IV. DATA

To conduct measurements, we deploy measurement servers at CDN PoPs across 33 different IXP locations. We embed IFrames containing our measurement script in publisher websites that are served by the CDN. We collect 1,132,110 measurements over the course of approximately two years (from January 1st, 2015 to October 11, 2016) from more than 500K clients, which are spread over approximately 300K /24 prefixes and span more than 900 ASes that are peering with the CDN and are also reachable via different transit providers. For each measurement, we record its timestamp, client IP address, PoP name, and RTT over peering ($RTT_{Peering}$) and transit ($RTT_{Transit}$) paths.

We use MaxMind IP geolocation database to locate client IP addresses in Figure 4. Black dots represent the location of all client IP addresses in our measurements. Note that IP geolocation databases are reasonably accurate at the country-level [38]. While we observe IP addresses from more than 100 different countries, a vast majority of the IP addresses are located in the United States and Europe. For instance, 54% of the clients in our measurements are located in the United States, 12% in the United Kingdom, 6% in Spain, and 5% in Germany.

Next, we analyze measurements with respect to PoP locations. Table I shows the number of IP addresses, /24 prefixes, and ASes observed across popular PoPs in our data. Overall, we note that the majority of users (IP addresses and /24



(a) New York (b) Miami

(c) London 1 (d) London 2

Fig. 5. Geographical location of PoPs and clients. The red dots show the location of the PoP while the black dots identify the location of the IP prefixes connecting to the PoP.

prefixes) connect to the PoPs in the United States. Specifically, we note that 7 out of the 10 PoPs with most connected clients are located in the United States. We observe that some /24 prefixes and ASes host more clients conducting active end-to-end latency measurements compared to others. For example, we note that around 28.5% of clients connect to the PoP located in Madrid while the PoP in New York serves around 6.7% of the clients. However, the clients connected to New York PoP are distributed among more /24 IP prefixes and ASes relative to the clients connected to Madrid PoP. More specifically, the clients connected to New York PoP originate from 22,972 different /24 IP prefixes and 90 different ASes while the clients connected to Madrid PoP originate from 16,904 different /24 IP prefixes and only 25 ASes.

Figure 5 plots the geographical footprint of some PoPs in the United States and Europe. Red dots represent the locations of PoPs and black dots represent the locations of clients connecting to these PoPs. As expected, we note that clients are typically located in geographical proximity to the PoP. However, since there is no PoP in South America, Figure 5(b) shows that clients in Brazil are being served by the PoP in Miami. The CDN has two PoPs in London for load-balancing purposes. In Figures 5(c) and 5(d), it is interesting to note that the geographical distribution of clients connecting to both London PoPs is very similar.

V. Results

In this section, we compare the performance of peering and transit interconnections based on our measurements.

Recall that we perform simultaneous latency measurements between a client and a CDN measurement server via peering and transit paths ($RTT_{Peering}$ and $RTT_{Transit}$).[1] For each

[1] Note that the CDN uses three major transit providers (NTT, TELIA, and DTAG [Deutsche Telekom]) at most PoPs. We conduct our measurements from PoPs which are multi-homed to all three transit providers.

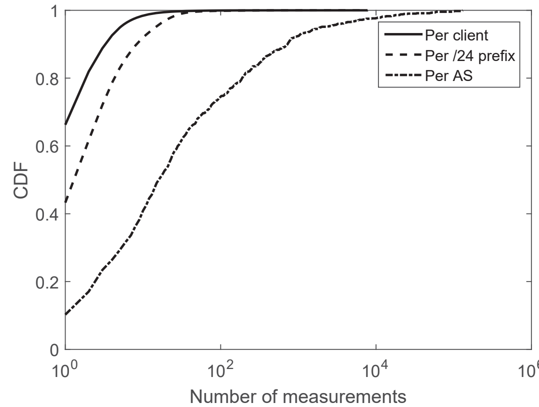


Fig. 6. CDF of number of measurements.

measurement, we calculate the normalized latency difference between peering and transit paths as:

$$\frac{RTT_{Peering} - RTT_{Transit}}{max(RTT_{Peering}, RTT_{Transit})}.$$

We use the maximum of both latencies in the denominator to limit the values between $[-100\%, 100\%]$. Since measurements are sparsely distributed across clients over time, we want to identify a suitable level of aggregation to conduct statistically significant analysis. First, we aggregate measurements for every client. Figure 6 shows that a majority of clients conduct only a few measurements. Specifically, we note that more than 65% of clients in our dataset conduct only one measurement. Furthermore, 99% clients conduct less than 13 measurements. Therefore, we would filter out a vast majority of clients from our analysis even if we set a moderate threshold on their measurement count. Second, we aggregate clients based on /24 prefixes. Figure 6 shows that about 40% of /24 prefixes have only one measurement. Therefore, filtering out /24 prefixes based on their measurement count would again filter out a substantial portion of /24 prefixes. Third, we aggregate clients based on their AS affiliation. In Figure 6, we note that about 25.8% of ASes have at least 100 measurements. On the other hand, we note that 82.4% ASes in our dataset have all of their IP addresses geolocated in one country. Thus, we conclude that AS-level aggregation is sufficient to address sparsity of our measurements while providing reasonably consistent geographic footprint. Therefore, in the rest of this paper, we compare the peering and transit paths based on the measurements aggregated over ASes.

Figure 7 plots the normalized latency difference between peering and three different transit paths. The y-axis represents the average relative performance difference between peering and transit paths. For each AS on the x-axis, we average the normalized latency difference between peering and transit paths over the entire duration of our data set. The horizontal lines represent the points on y-axis where performance difference is greater or less than 5% and -5% respectively. The ASes are sorted with respect to their average normalized latency difference over peering and transit paths independently for each transit provider. ASes on the left have

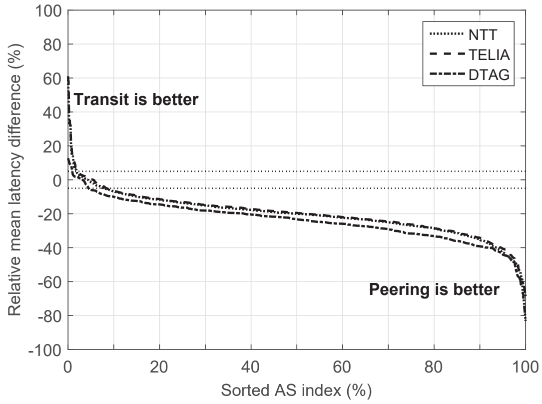


Fig. 7. Performance difference between peering and various transit paths per AS. Note that the x-axis is normalized to 100% based on the number of ASes for various transits.

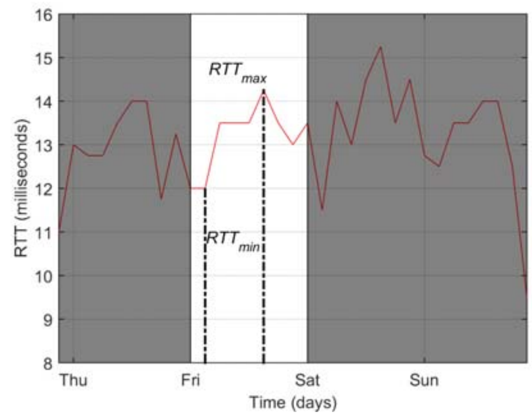


Fig. 8. Timeseries of measured RTT for a popular access ISP.

better performance over transit paths as compared to peering paths and those on the right have better performance over peering paths as compared to transit paths. In Figure 7, we note that majority of ASes experience better latency via peering paths as compared to transit paths. For example, considering NTT, we note that only 2% of the ASes get substantial performance benefit from transit paths as compared to peering paths. This pattern is consistent for other transit providers as well. More specifically, only 2% and 0.7% of ASes get better performance via transit paths compared to peering paths for TELIA and DTAG transit providers. On the other hand, we note that approximately 92%, 91% and 95% of the ASes get at least more than 5% improvement in latency from peering paths compared to the transit paths for NTT, TELIA and DTAG respectively. Our aggregate analysis shows that peering paths significantly outperform different transit paths for a vast majority of ASes. Since there is not much difference across different transit providers, we limit our analysis to NTT transit provider in the rest of this paper.

Why do peering paths outperform transit paths for a vast majority of ASes? Intuitively, we would expect peering paths to perform better due to the following reasons. First, peering paths typically only carry the bidirectional traffic between the peering ASes. Therefore, peering links likely experience less congestion as compared to transit paths which are responsible to handle traffic from multiple transit customers. Second, peering paths are likely less circuitous as compared to transit paths. Therefore, peering paths are expected to provide better end-to-end latency compared to transit paths even in the absence of congestion.

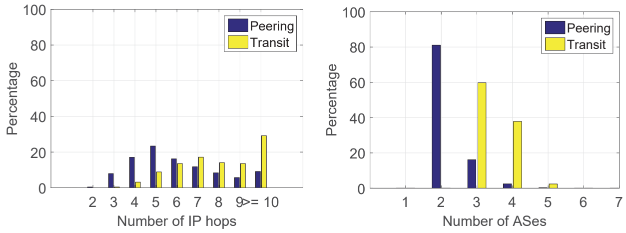
We further investigate the observed performance differences between peering and transit paths by decomposing the measured delay into its constituent components. End-to-end network delays consist of the following components: (1) processing, (2) transmission, (3) propagation, and (4) queueing. However, processing and transmission delays are negligible as compared to propagation and queueing delays. Therefore, our latency measurements are primarily composed of propagation and queueing delays.

Figure 8 plots a representative example of the average latency timeseries for a large access ISP in our data set. As reported in prior literature [32], we observe diurnal variations in RTT measurements. We analyze diurnal variations in RTT to estimate propagation and queueing components of the measured RTT. We note that minimum measured RTT (RTT_{min}) in a 24-hour time interval captures the end-to-end delay when queue buildup at routers is at a minimum. We use RTT_{min} as the estimated upper bound of *propagation delay* experienced by our probe packets.[2] We also note that since end-to-end paths over the Internet tend to stay relatively constant over time, propagation delay (RTT_{min}) stays constant over a relatively large timescale. This assumption allows us to use the difference between maximum measured RTT (RTT_{max}) and minimum measured RTT (RTT_{min}) in a 24-hour time interval as the upper bound estimate of maximum queueing delay experienced by our probe packets [32].

We next leverage the estimated propagation and queueing delays to further analyze the performance difference between peering and transit paths.

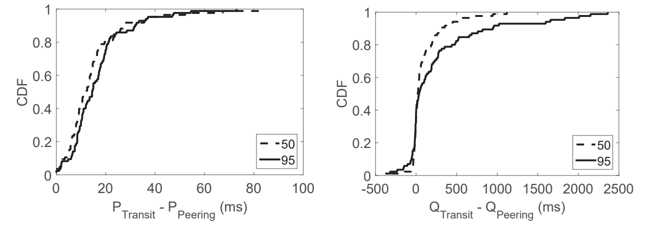
Propagation Delays. We expect transit paths to have larger propagation delays as compared to peering paths because they span multiple ASes other than client and CDN ASes. To verify this, we conduct traceroute measurements from our CDN measurement servers located at different PoPs to clients in access ISPs around the world via transit and peering paths. For every AS-PoP pair, we identify client subnets in access ISPs that conduct most measurements in our dataset. We then use MTR [3] to conduct traceroute and ping measurements for identifying hops on transit and peering paths between our CDN measurement servers and the clients in these subnets. First, we analyze the path length between CDN measurement servers at PoPs and clients in access ISPs from the traceroute measurements. We calculate the path length as the number of IP hops in traceroute measurements. We further analyze AS-level path information by mapping IP addresses to ASes. We use standard practices (see [14] for more details) to recover traceroutes with unresponsive hops.

[2] RTT_{min} is an upper bound estimate of propagation delay because a path may still have non-zero queue buildup when RTT is minimum.



(a) IP hops (b) ASes

Fig. 9. Distribution of path lengths in terms of IP hops and AS hops.



(a) Propagation Delay (b) Queueing Delay

Fig. 10. Distribution of ASes in terms of delay comparison at 50th and 95th percentile for peering and transit paths.

Figure 9 plots the distribution of number of IP hops and number of ASes between CDN measurement servers and clients in access ISPs. As expected, we observe that peering paths are considerably shorter than transit paths. Specifically, Figure 9(a) shows that median path length of peering paths is 6 whereas that of transit paths is 8. Moreover, from Figure 9(b), we note that peering paths span over less number of ASes compared to transit paths. As expected, we note that more than 80% of the peering paths comprise of only CDN and client ASes. It is interesting to note that more than two ASes for some peering paths. Our further investigation revealed that the intermediate hop typically belongs to the IXP possibly due to the use of public peering fabric or other interconnection infrastructure provided by the IXP. In contrast, more than 80% of the transit paths comprise of 3 or more ASes. Apart from CDN and client ASes, the traceroute measurements include one or more transit ASes. We conclude that transit paths are longer than peering paths both in terms of IP hops and number of intermediate ASes.

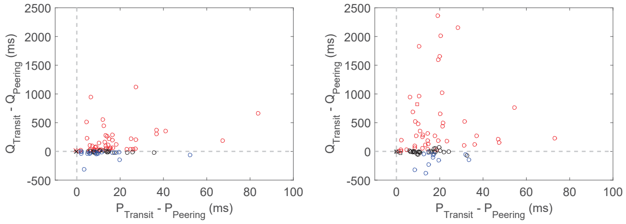
We expect longer paths to lead to larger propagation delays. Figure 10(a) plots the 50th and 95th percentiles of the difference in propagation delays between peering and transit paths across client ASes. $P_{Peering}$ and $P_{Transit}$ refer to the propagation delays of peering and transit paths to an AS, respectively. The x-axis represents the difference in propagation delays between peering and transit paths. We observe that more than 95% of the ASes in the dataset experience better propagation delay via peering paths as compared to transit paths. Our observation is consistent across CDF of both 50th and 95th percentiles. More specifically, half of the ASes experience up to 12-15 milliseconds shorter propagation delays over peering paths as compared to transit paths.

Queueing Delays. We next compare and contrast queueing delays of peering and transit paths. The queueing delay is an indicator of potential congestion along the end-to-end paths. Our analysis of the estimated queueing delays via peering and transit paths can indicate congestion due to underprovisioning. Transit providers optimize resource usage by selling more aggregate bandwidth to their customers than what their infrastructure can handle [36]. The key idea of underprovisioning is that since customer traffic demand is elastic it would be lower than the peak demand for most of the time. Such underprovisioning can cause congestion when multiple transit customers send more traffic than what was anticipated through the transit provider's network. In contrast, we do not expect this issue for peering interconnections because they are between an access ISP and the CDN. Overall, we expect higher queueing delays for transit paths as compared to peering paths.

Figure 10(b) plots the 50th and 95th percentiles of the difference in queueing delays between peering and transit paths across client ASes. $Q_{Peering}$ and $Q_{Transit}$ refer to the queueing delays of peering and transit paths to an AS, respectively. The x-axis represents the difference in queueing delays between peering and transit paths. It is noteworthy that our estimate of queueing delay is an *upper bound* on the maximum queueing delay over 24-hour time intervals. Therefore, some of the high queueing delays in Figure 10(b) are possibly inflated due to retransmissions and rerouting events. We cannot account for these issues due to our lack of visibility at the transport and network layers.

Figure 10(b) shows that peering paths generally outperform transit paths in terms of queueing delays. For instance, more than 60% ASes experience at least 20 milliseconds improvement in queueing delays via peering paths as compared to transit paths for the 50th percentile comparison. Furthermore, the difference in queueing delays exhibits a long tailed distribution for the 95th percentile comparison. For instance, more than 20% ASes experience at least 450 milliseconds improvement in queueing delays via peering paths for 95th percentile comparison. It is interesting to note that some ASes experience better queueing delays via transit paths compared to peering paths. This trend is highlighted by ASes that lie in the negative x-axis region in Figure 10(b). For instance, we note that around 18% of ASes in our dataset experience at least 20 milliseconds improvement in queueing delays from transit paths as compared to peering paths. This trend is consistent across the 50th and 95th percentile comparison distribution in Figure 10(b). Overall, however, we conclude that peering paths provide better queueing delays as compared to transit paths for a majority of ASes in our dataset.

Next, we analyze the impact of number of ASes in transit paths on queueing delays. To this end, we compare queueing delays for transit paths with different AS path lengths. We find that the difference in queueing delays between peering and transit paths increases as the number of transit ASes increases. For example, the median difference in queuing delays increases from 24 milliseconds for one transit AS to 40 milliseconds for three transit ASes. This shows that transit paths traversing more transit ASes perform significantly worse in terms of queueing delays as compared to transit paths



(a) 50th percentile (b) 95th percentile

Fig. 11. Scatter plots for the comparison of peering and transit path delays in terms of propagation and queueing delays for ASes at 50th and 95th percentiles.

traversing fewer transit ASes.

Combined Delay Analysis. We next jointly analyze the performance difference between peering and transit paths in terms of both propagation and queueing delays. In Figure 11, the x-axis represents the difference in propagation delays between transit and peering paths ($P_{Transit} - P_{Peering}$) whereas the y-axis represents the difference in queueing delays between transit and peering paths ($Q_{Transit} - Q_{Peering}$). Each marker in Figure 11 represents an AS. The marker types of circle, cross, and square indicate whether the normalized difference between propagation delays of peering and transit paths is more than 5%, less than -5%, or between -5% to 5%, respectively. The marker colors of red, blue and black indicates whether the normalized difference between queueing delays of peering and transit paths is more than 5%, less than -5%, or between -5% to 5%, respectively. We compare propagation and queueing delays of both peering and transit paths at the 50th and 95th percentiles for the ASes.

We further summarize the performance comparison of peering and transit paths in terms of propagation and queueing delays in Table II. We populate Table II by calculating the percentage difference between $P_{Peering}$, $P_{Transit}$ and $Q_{Peering}$, $Q_{Transit}$ for each AS at the 50th and 95th percentiles. $P_{Transit} > P_{Peering}$ implies that $P_{Transit}$ is at least 5% more than $P_{Peering}$. $P_{Transit} \approx P_{Peering}$ implies that $P_{Transit}$ and $P_{Peering}$ are within 5% of each other. Using this notation, we categorize each AS in one of the nine categories in Table II.

From Figure 11 and Table II, we note that peering consistently outperforms transit in terms of propagation delay ($P_{Transit} > P_{Peering}$) and queueing delays ($Q_{Transit} > Q_{Peering}$). Specifically, we note that peering paths provide at least 5% improvement in both propagation and queueing delays for at least 44 ASes at the 50th and 95th percentiles. These ASes appear as red circles in the first quadrant for all percentile comparisons in Figure 11. We do observe that queueing delays for peering paths are comparable to ($Q_{Transit} \approx Q_{Peering}$) queueing delays for transit paths for many ASes. Specifically, around 20 ASes experience comparable queueing delay performance from both peering and transit paths. We note that these ASes appear as black markers in Figure 11.

From Figure 11 and Table II, we note that some ASes experience worse queueing delays over peering paths as compared to transit paths ($Q_{Transit} < Q_{Peering}$). Specifically, we note that 11-17 ASes appear with blue markers in Figure 11. These ASes experience better propagation delay but suffer from worse queueing delay performance over peering paths. For these ASes, the degradation in queueing delay dominates the improvement in propagation delay; leading to poor overall performance over the peering paths. Such ASes that get lower latencies over transit paths as compared to peering paths lie on the left end of the aggregate performance curve in Figure 7. Overall, we conclude that peering paths provide better propagation and queueing delays with the exception of a few ASes.

VI. Limitations

We discuss the limitations of our measurement methodology, data, and analysis below.

Accuracy. Our application layer latency measurement approach can potentially overestimate RTTs (as compared to network layer latency measurements) due to added delays of browser and kernel scheduling. Further note that application layer latency measurements are susceptible to overestimation due to packet losses because retransmissions at the transport layer are not visible at the application layer. Therefore, packet loss would cause the measured RTT to be greater than actual end-to-end RTT. To mitigate these concerns, we compared our application and network layer latency measurements in a controlled testbed under different scenarios in Section III. We found that our application layer latency measurements closely follow network layer latency measurements. We also found that the variations in application layer latency measurements become relatively insignificant for large RTT values.

Sparsity. Our measurements are crowdsourced from clients that visit a set of webpages hosted by the CDN. From the CDN's perspective, we do not have any control over the sparsity of measurements from specific clients or ASes. We observed that measurements from specific clients are sparse across time. We mitigate the sparsity issue by grouping measurements by clients based on their AS affiliation. To further mitigate the sparsity issue for propagation and queueing delay analysis, we only consider days with a large number of RTT measurements.

Localization. Our application layer measurements do not allow us to estimate propagation delays of individual links and queueing delay experienced by packets traversing different routers. Therefore, we cannot localize performance differences between peering and transit paths. However, for the scope of this study, we are interested in characterizing *end-to-end* path performance in terms of both propagation and queueing delays. We leave further analysis of performance differences, including localization, for future work. To this end, we plan to complement our application layer latency measurements with a manageable number of network layer ICMP measurements.

Impact of path changes. Recall that we use the temporal variations over a period of 24 hours to estimate propagation and queueing delays. In case of a path change, our propagation delay estimate reflects the minimum end-to-end propagation delay observed during the time period. Since we measure

	$Q_{Transit} > Q_{Peering}$	$Q_{Transit} \approx Q_{Peering}$	$Q_{Transit} < Q_{Peering}$
$P_{Transit} > P_{Peering}$	48/44	17/26	17/11
$P_{Transit} \approx P_{Peering}$	0/1	0/2	0/0
$P_{Transit} < P_{Peering}$	1/0	2/1	0/0

TABLE II

AS-BREAKDOWN IN TERMS OF PROPAGATION AND QUEUEING DELAYS AT 50/95TH PERCENTILE ACROSS PEERING AND TRANSIT PATHS

queueing delay as the difference between the minimum and maximum RTT observed during the day, we potentially overestimate queueing delay in case of path changes. It is noteworthy that our observed queueing delays are generally at least an order of magnitude greater than propagation delays. Therefore, we argue that our estimate of queueing delays only incur minor overestimation.

Traceroute measurements. First, some of the in-path routers reply with IP address of a different interface than the incoming interface. This causes the estimated number of ASes on peering and transit paths to be inflated. Figure 9(b) shows that these cases represent a small fraction of all traceroute measurements. Second, we note that routers sometimes incorrectly send peering traffic over transit due to misconfiguration. However, such misconfigurations are rare and do not have a significant impact on our results in Figure 9. Finally, we are unable to map some IP addresses to ASes. We try to resolve such situations using standard practices [14] and discard the unresolved traceroutes.

Having in mind the caveats presented here, we believe that our first look at the performance comparison of peering and transit interconnections provides valuable insights to researchers studying the Internet's topology using analytical and empirical methods.

VII. Related Work

Internet peering ecosystem has been widely studied to understand its technical, logistical, economic, and political constraints. Norton [36] provides a detailed analysis about the common (and uncommon! [24], [13], [43]) interconnection practices that are used by ISPs. A major portion of prior literature is focussed on studying the prevalence of peering ecosystem by conducting active and passive experiments. Augustin et al. [10] used traceroute to detect 223 IXPs and identify IXP-related peering relationships that were not present in the AS maps of the Internet then. Labovitz et al. [26] examined commercial inter-domain traffic of more than 3K peering routes to show that the majority of inter-domain traffic flows directly over peering links between content providers and access ISPs. Lodhi et al. [31] analyzed PeeringDB and BGP data to study the Internet peering ecosystem. Dhamdhere et al. [17], [20] used RIPE and RouteViews repositories to study the evolution of Internet topology at AS level over the course of a decade. Researchers have studied the role of IXPs in facilitating peering interconnections between ISPs. Ager et al. [9] studied a rich peering ecosystem at an European IXP and found that close to 400 members at the IXP have established more than 50K peering interconnections. Castro et al. [12] studied an emerging phenomenon of remote peering. Generally, peering has become a major interconnection strategy among ISPs in this vast Internet ecosystem. Therefore, in this study, we empirically quantify the performance benefits of peering as compared to transit.

Researchers have also extensively studied economic aspects of peering and transit selection strategies using game-theoretic models [11], [8]. Dhamdhere et al. [18] proposed dynamic agent-based models to study the impact of economic decisions such as provider and peer selection on the evolution of Internet topology in "steady state". They also developed an agent-based network formation tool, ITER, to model the "flattening" Internet topology as a consequence of provider and peer selection strategies [19]. Lodhi et al. [28], [29] studied the myopic adoption of open peering strategies among transit providers as a result of peer pressure which degrades their economic utility. Lodhi et al. [30] further noted that a lack of traffic information and network topology information limits the abilities of a tier-2 ISP to accurately forecast the impact of its peering decisions. Ma et al. [34] observed that for peering interconnections between content providers and access ISPs, settlement-free and paid are optimal pricing models for symmetric and asymmetric traffic patterns respectively. However, we note that these papers primarily consider the connectivity costs associated with transit and peering interconnections to model utility of the service providers, and not consider the effect of performance benefits on the utility of service providers. Prior research has shown that better performance such as better end-to-end latency and throughput, leads to better user engagement, which improves service providers' revenues [42], [37]. To this end, Ma et al. further incorporated the content providers' characteristics (quality requirements, traffic patterns) and access providers' characteristics (QoS guarantees, price) to model transit and peer selection strategies [35]. Furthermore, in [33] Ma characterized the benefits of paid peering to service providers by considering various parameters like end-user stickiness and market shares of the service providers. Courcoubetis et al. [15] used service profitability from advertising, user/subscriber loyalty to derive a pricing model for premium peering relationships between content and access providers using Nash bargaining solutions. We note that the prior literature relies on intuitive understanding of performance benefits of peering paths over transit paths especially for the peering interconnections between content providers and other ISPs [21]. Our results can complement these analytical studies.

VIII. Conclusion

In this paper, we empirically compare the performance of peering and transit interconnections by conducting a large scale measurement study. We deploy our measurement

JavaScript on multi-homed CDN servers located at 33 IXPs around the world and conduct measurements from 510K clients. Overall, we find that peering paths outperform transit paths. We find that peering paths almost always outperform transit paths in terms of propagation delays because peering paths are shorter. While peering paths often outperform transit paths in terms of queueing delays, we do observe higher queueing delays on peering paths as compared to transit paths for a few ASes in our data.

To the best of our knowledge, we present the first large-scale empirical performance comparison of peering and transit interconnections in the wild. Prior research on modeling peer and transit interconnection strategies either does not incorporate performance as a key factor or makes arbitrary assumptions. We expect our empirical results to inform future research on Internet topology modeling. Our results also establish baseline from which future performance measurements of peering and transit can be studied. In future, we plan to study performance differences between public and private peering as well as study the performance impact of different traffic engineering decisions.

Acknowledgements

We thank our shepherd, Ignacio Castro, and the anonymous reviewers for their useful feedback on this paper. This work is supported in part by the National Science Foundation under grant numbers CNS-1464110 and CNS-1617288.

REFERENCES

[1] Akamai Facts & Figures. https://www.akamai.com/us/en/about/facts-figures.jsp.
[2] Google Caching Overview. https://peering.google.com/about/ggc.html.
[3] MTR - My Traceroute. http://www.bitwizard.nl/mtr.
[4] Open Connect Overview. https://openconnect.netflix.com/Open-Connect-Overview.pdf.
[5] ICMP Rate Limiting. http://www-iepm.slac.stanford.edu/monitoring/limit/limiting.html, December 1999.
[6] Cisco Visual Networking Index: Forecast and Methodology, 2014–2019, 2015.
[7] Sandvine: Global Internet Phenomena Report. https://www.sandvine.com/trends/global-internet-phenomena/, May 2015.
[8] G. Accongiagioco, E. Altman, E. Gregori, and L. Lenzini. Peering vs Transit: a Game Theoretical model for Autonomous Systems connectivity. In *IFIP Networking Conference*, 2014.
[9] B. Ager, N. Chatzis, A. Feldmann, N. Sarrar, S. Uhlig, and W. Willinger. Anatomy of a Large European IXP. In *ACM SIGCOMM*, 2012.
[10] B. Augustin, B. Krishnamurthy, and W. Willinger. IXPs: Mapped? In *ACM IMC*, 2009.
[11] N. Badasyan and S. Chakrabarti. A simple game-theoretic analysis of peering and transit contracting among Internet service providers. In *Telecommunications Policy*, 2008.
[12] I. Castro, J. C. Cardona, S. Gorinsky, and P. Francois. Remote Peering: More Peering without Internet Flattening. In *ACM CoNEXT*, 2014.
[13] I. Castro and S. Gorinsky. T4P: Hybrid Interconnection for Cost Reduction. In *IEEE INFOCOM Workshops*, 2012.
[14] Y.-C. Chiu, B. Schlinker, A. B. Radhakrishnan, E. Katz-Bassett, and R. Govindan. Are We One Hop Away from a Better Internet? In *IMC*, 2015.
[15] C. Courcoubetis, L. Gyarmati, N. Laoutaris, P. Rodriguez, and K. Sdrolias. Negotiating Premium Peering Prices: A Quantitative Model with Applications. In *ACM Transactions on Internet Technology*, 2016.
[16] R. A. Deal. Cisco Router Firewall Security: DoS Protection. http://www.ciscopress.com/articles/article.asp?p=345618&seqNum=5, October 2004.
[17] A. Dhamdhere and C. Dovrolis. Ten Years in the Evolution of the Internet Ecosystem. In *IMC*, 2008.
[18] A. Dhamdhere and C. Dovrolis. An agent-based model for the evolution of the internet ecosystem. In *COMSNETS*, 2009.
[19] A. Dhamdhere and C. Dovrolis. The Internet is Flat: Modeling the Transition from a Transit Hierarchy to a Peering Mesh. In *ACM CoNEXT*, 2010.
[20] A. Dhamdhere and C. Dovrolis. Twelve Years in the Evolution of the Internet Ecosystem. In *IEEE/ACM Transactions on Networking*, 2011.
[21] A. Dhamdhere, C. Dovrolis, and P. Francois. A Value-based Framework for Internet Peering Agreements. In *Teletraffic Congress (ITC)*, 2010.
[22] B. Frank, I. Poese, G. Smaragdakis, A. Feldmann, B. M. Maggs, S. Uhlig, V. Aggarwal, and F. Schneider. *Collaboration Opportunities for Content Delivery and Network Infrastructures*. Book Chapter. ACM SIGCOMM ebook on Recent Advances in Networking, 2013.
[23] P. Gill, M. Arlitt, Zongpeng, and A. Mahanti. The Flattening Internet Topology: Natural Evolution, Unsightly Barnacles or Contrived Collapse? In *PAM*, 2008.
[24] V. Giotsas, M. Luckie, B. Huffaker, and kc claffy. Inferring Complex AS Relationships. In *IMC*, 2014.
[25] A. Jain, T. Reifsteck, J. Mann, Z. Wang, and A. Quach. Resource Timing Level 1. W3C Working Draft, April 2016.
[26] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet Inter-Domain Traffic. In *ACM SIGCOMM*, 2010.
[27] W. Li, R. K. P. Mok, R. K. C. Chang, and W. W. T. Fok. Appraising the Delay Accuracy in Browser-based Network Measurement. In *ACM Internet Measurement Conference (IMC)*, 2013.
[28] A. Lodhi, A. Dhamdhere, and C. Dovrolis. Peering Strategy Adoption by Transit Providers in the Internet: A Game Theoretic Approach. *ACM SIGMETRICS Performance Evaluation Review*, 40(2):38–41, 2012.
[29] A. Lodhi, A. Dhamdhere, and C. Dovrolis. Open Peering by Internet Transit Providers: Peer Preference or Peer Pressure? In *IEEE INFOCOM*, 2014.
[30] A. Lodhi, N. Laoutaris, A. Dhamdhere, and C. Dovrolis. Complexities in Internet Peering: Understanding the “Black” in the “Black Art”. In *IEEE INFOCOM*, 2015.
[31] A. Lodhi, N. Larson, A. Dhamdhere, C. Dovrolis, and kc claffy. Using PeeringDB to Understand the Peering Ecosystem. In *ACM SIGCOMM Computer Communication Review*, 2014.
[32] M. Luckie, A. Dhamdhere, D. Clark, B. Huffaker, and kc claffy. Challenges in Inferring Internet Interdomain Congestion. In *ACM IMC*, 2014.
[33] R. T. B. Ma. Pay or Perish: The Economics of Premium Peering. In *IEEE Journal on Selected Areas in Communications*, 2017.
[34] R. T. B. Ma, D. M. Chiu, J. C. S. Lui, V. Misra, and D. Rubenstein. On Cooperative Settlement Between Content, Transit, and Eyeball Internet Service Providers. In *IEEE/ACM Transactions on Networking*, 2011.
[35] R. T. B. Ma, J. C. S. Lui, and V. Misra. On the Evolution of the Internet Economic Ecosystem. In *WWW*, 2013.
[36] W. B. Norton. *The 2014 Internet Peering Playbook: Connecting to the Core of the Internet*. DrPeering Press, 2014.
[37] O. Palmer. How Does Page Load Time Impact Engagement? https://blog.optimizely.com/2016/07/13/how-does-page-load-time-impact-engagement/, July 2016.
[38] I. Poese, S. Uhlig, M. A. Kaafar, B. Donnet, and B. Gueye. IP Geolocation Databases: Unreliable? In *ACM SIGCOMM Computer Communication Review*, 2011.
[39] S. Ramachandran. New FCC Broadband Benchmark Lifts Comcast’s Share to Nearly 60%. http://blogs.wsj.com/corporate-intelligence/2015/01/29/comcast-bulks-up-on-broadband, January 2015.
[40] D. Rayburn. Heres The Latest North American Transit Pricing, Down 10Year-Over-Year. http://blog.streamingmedia.com/2016/05/north-american-transit-pricing.html, May 2016.
[41] P. Richter, G. Smaragdakis, A. Feldmann, N. Chatzis, J. Boettger, and W. Willinger. Peering at Peerings: On the Role of IXP Route Servers. In *IMC*, 2014.
[42] N. Shalom. Amazon found every 100ms of latency cost them 1% in sales. https://blog.gigaspaces.com/amazon-found-every-100ms-of-latency-cost-them-1-in-sales/, August 2008.
[43] V. Valancius, C. Lumezanu, N. Feamster, R. Johari, and V. V. Vazirani. How Many Tiers? Pricing in the Internet Transit Market. In *SIGCOMM*, 2011.