# Hypothesis-based Comparison of IPv6 and IPv4 Path Distances

David Hasselquist[1], Christian Wahl[1,2], Otto Bergdal[1], and Niklas Carlsson[1]

[1]Linköping University, Sweden
[2]Technische Universität München, Germany

**Abstract.** Short end-to-end path lengths and faster round-trip times (RTTs) are important for good client performance. While prior measurement studies related to IPv6 primarily focus on various adoption aspects, much less work have focused on performance metrics such as these. In this paper, we compare the relative end-to-end path distances and RTTs when using IPv6 and IPv4 between PlanetLab nodes in Europe and different subsets of popular domains. In addition to providing access to multiple measurement nodes, the use of PlanetLab also provides a use-case driven report of running IPv6 experiments on this previously prosperous experimental platform for academic research. In particular, the study provides a first report on performing IPv6 experiments on PlanetLab, highlights the lack of IP support among PlanetLab nodes and limitations of state-of-the-art traceroute tools used for IPv6 measurements, and provides a statistical methodology that uses hypothesis testing to derive insights while accounting for such testbed and traceroute shortcomings. Our performance analysis shows (among other things) that the relative RTTs of the IPv6 paths are currently faster than the corresponding IPv4 paths, and that the fraction of pairings for which this is the case is quickly increasing across a wide range of domain popularities and domain categories. These findings suggest that there is incentive to use IPv6, which may impact the rate of further IPv6 deployment.

**Keywords:** IPv4 vs IPv6 · Path distances · Traceroute · PlanetLab

## 1 Introduction

After a long, slow initial adoption period, IPv6 has finally started to see significant usage. For example, over the past ten years, the fraction of IPv6 connections to Google servers has increased from 0.25% (Jan. 2011) to 5% in Jan. 2015 and to 29-34% in Sept. 2020 [18]. A diminishing pool of IPv4 addresses and various flag days may have been contributing factors.

With increasing use of IPv6, it is becoming increasingly important to understand the performance that clients observe when accessing the web using IPv6. However, it has long been understood that the IPv6 and IPv4 routing topologies are non-overlapping (e.g., with IPv6 having a less connected core [16]) and that IPv6 tunnels can negatively impact the performance. In this paper, we use

traceroute measurements from a number of European locations to measure the end-to-end path distances between these locations and different sets of popular web domains, perform statistical tests on the collected datasets, and report on similarities and differences in the relative distance differences observed when using IPv6 and IPv4, respectively. Of particular interest are the observed IP-hop counts, AS-hop counts, and end-to-end round-trip times (RTTs). These metrics are important to understand to what extent end-to-end routing differences (e.g., due to differences in connectivity and the use of tunnels) still cause significant differences in the end-to-end path distances observed with IPv6 and IPv4, and to what extent such differences impact the end-to-end RTT performance.

While many measurement studies have focused on IPv6 adoption [12, 13, 16, 21, 23, 30, 31], less work has focused on the relative end-to-end performance of IPv6 and IPv4 connections and how it may be affected by the lack of full end-to-end adoption. Most closely related our work (Section 5), Giotsas et al. [16] showed in 2015 that the performance of IPv6 paths can be significantly hurt by IPv6 tunnels and a less connected transit-free clique. However, since then, further adoption has taken place, and one would expect that IPv6 paths and their performance would improve over time as tunneled paths are replaced with native paths and IPv6 AS relationships mature.

Motivated by the observations above and shortcomings of the basic traceroute tool (which has been shown to not capture the actual paths taken [3]), the popular Paris traceroute alternative (which we find does not work well for IPv6), and traceroutes in general (today often having many missing entries), in this paper we develop a methodology that combines basic traceroute measurements and statistical methods to determine whether there are statistically significant differences in paths lengths and RTTs between IPv6 and IPv4 paths, while accounting for the limitations of existing traceroute tools.

In contrast to prior work (Section 5), we focus on the relative end-to-end distances when connecting to different categories of domains when using IPv6 and IPv4, and how the RTT performance may be affected by the lack of full end-to-end adoption. Implementing our data collection on PlanetLab using different traceroute tools, we first provide some methodological insights regarding challenges (such as lack of IPv6 support among PlanetLab nodes and state-of-the-art traceroute tools) that complicate traceroute-based monitoring of IPv6 paths from PlanetLab. Second, we present a data collection (Section 3) and pairwise analysis (Section 4.1) methodology that allows head-to-head comparisons of the distances observed when using IPv6 and IPv4 from example locations in Europe. For data collection, we employ two different *traceroute* tools [2, 3] on the full (but small) set of PlanetLab nodes located in Europe that run IPv6, and measure the network routes to popular website domains [1, 29].

While Paris traceroute (used for a four-week measurement campaign) in theory should better capture the actual paths taken than the basic traceroute tool, we find the success rates of this tool unacceptably low (22%), and instead mostly focus our analysis on the datasets collected in May 2019 and Sept. 2019 using parallel instances of the basic traceroute tool (74-78% success rate). Using these

datasets, we evaluate the differences and changes observed across different sets of domains (grouped based on popularity rank or domain category) and measurement locations, with regards to the measured RTTs and the number of IP and Autonomous Systems (AS) hops along the paths. For our quantitative analysis, we used three different statistics (mean, median, and 95% confidence tests) to determine which protocol version has the shorter distance for each end-to-end pairing, between PlanetLab node and domain, and then summarized the results on a per-category basis.

Our analysis provides a quantitative snapshot into the relative differences in the distances observed when using IPv6 and IPv4 to connect to different domain classes, and how these differences are changing. For example, there still appears to be significant use of IP tunnels (e.g., much lower IP and AS hop counts), the relative RTTs of the IPv6 paths (compared to the corresponding IPv4 paths) improved notably between the datasets (May 2019 and Sept. 2019) across all five rank categories considered (based on Alexa ranks) and across almost all 16 domain categories considered (each represented using the 50 top-ranked domains of that category). Overall, our findings are encouraging, since IPv6 already appears to outperform IPv4 and these advantages are increasing. This may further incentivize IPv6 deployment.

The remainder of the paper is organized as follows. Section 2 presents background and challenges comparing IPv6 and IPv4 paths from PlanetLab. The following sections present our collection methodology (Section 3), analysis methodology and results (Section 4), discussion of related works (Section 5), and conclusions (Section 6).

## 2 Background and Challenges

**PlanetLab (status and challenges):** At one point in time, PlanetLab provided an excellent testbed for running large-scale network experiments. However, today many PlanetLab nodes are old, out of date, and often not even reachable. Among the 295 PlanetLab Europe nodes that we had access to, only 66 nodes responded to at least one ping during an eight-day measurement (May 2019) in which we sent one ping every 10 minutes. Out of the 66 responding nodes, only 45 nodes responded every time and we could only login to 39 nodes using ssh. To make things worse, the current implementation of the virtual machine system used in PlanetLab lacks IPv6 support, preventing us from running IPv6 traceroutes from within our Planetlab slice even when a node was supporting IPv6. After contacting PlanetLab support, we found out that only nine (!) nodes on PlanetLab Europe support IPv6 in some way. In addition to limited maintenance, this shows that few members have upgraded their machines to support IPv6.

The lack of IPv6 support among the existing PlanetLab nodes captures a general inertia in deploying IPv6. While some PlanetLab participation requirements (e.g., that nodes should be placed in a DMZ, outside the local firewall, and typically need to be isolated from the institutions regular network) may be

**Table 1.** PlanetLab nodes used for experiments

| **Location** | **Node ID** |
|---|---|
| Université Pierre et Marie Curie, Paris, France | **ple3**.planet-lab.eu (not "Sept 2019"), **ple42**.planet-lab.eu, **ple44**.planet-lab.eu, **nuc1**.planet-lab.eu |
| Univ. of Rostock, Germany | **pl1**.uni-rostock.de, **pl2**.uni-rostock.de |
| Univ. of Göttingen, Germany | **planetlab2**.informatik.uni-goettingen.de |
| CESNET, Prague, Czech Republic | **ple1**.cesnet.cz (not "Paris"), **ple2**.cesnet.cz (only "Sept. 2019") |

a contributing factor to the low IPv6 support, one may still expect that research institutes such as those participating in PlanetLab would be among the "early" adopters of such technology. (Whatever "early" adopter means in the context of IPv6 deployment!) Fortunately, PlanetLab's excellent support team gave us direct access to all host machines that had IPv6 support and installed the necessary software on these machines. This allowed us to run our experiments (using both IPv6 and IPv4) from multiple locations in Europe.

While the IPv6 study presented in Sections 3 and 4 would have benefited from more PlanetLab nodes in Europe deploying IPv6, these nine nodes provided us with access to multiple geographically diverse measurement locations and allowed us to demonstrate the use of our hypothesis-based methodology. In this regard, PlanetLab still offers some benefits over running experiments from only local machines.

Table 1 summarizes the nine machines that had support for IPv6. Already at this time, we note that one of the machines was not accessible during our May 2019 experiments and one machine was not accessible during our Sept. 2019 experiments. To allow comparison over time, we excluded measurements from these machines from the analysis in Section 4.

**Traceroute (versions, limitations, and challenges):** Traceroute tools typically use a sequence of probe messages with increasing time-to-live (TTL) values, and leverage that ICMP "Time exceeded" messages are returned by the router at which the TTL value reaches zero to learn where each probe ended and measure the RTTs to each such router/node. Due to route changes and load balancing, for example, the end-to-end path during such sequence of probes may change over time. It is therefore important to note that such basic implementation does not necessarily return a specific route taken by a packet and may suggest false links (between unconnected routers). Furthermore, some routers do not respond with ICMP packets and/or make different decisions based on packet type, leaving holes in the path information [20]. These challenges have motivated the implementation of many traceroute versions and many designs allow different packet types to be used for the probes (e.g., UDP/DNS, TCP SYN, and ICMP Echo packets).

Augustin et al. [3] recognized that per-flow load balancing is often used to to ensure end-to-end stability, and proposed the *Paris* traceroute tool as a means to mitigate network topology mapping anomalies that can occur due to such load balancing. With per-flow load balancing, packets from the same flow (defined

as a *five-tuple* consisting of source IP, destination IP, source port, destination port, and protocol) are forwarded over the same route, while packets associated with other flows (same IP-pair) may be routed on different paths. This causes problems for the standard traceroute tool, since it uses randomized values for some IP header fields (e.g., ports) to distinguish responses from different probes. Paris traceroute tries to amend this problem by identifying probes using only IP header fields not used by per-flow load balancing.

Paris traceroute has been shown to provide more accurate paths than basic traceroute. However, due to a bug in the current implementation, simultaneous traceroutes are not possible with Paris traceroute (as routes gets mixed). The use of this tool therefore significantly limits the number of (accurate) traceroutes that can be performed within a time window. We have contacted the developers of the tool and a fix is expected. However, as of the writing of this paper, the authors only have a fix for IPv4 (on a separate version), not IPv6, and we are not able to install any such tools on the PlanetLab nodes ourselves. We therefore limit our study to using the basic traceroute tool and using Paris traceroute strictly sequentially. Paris traceroute was used for a four-week long single-threaded campaign, while we used the basic traceroute tool for two separate one-week long campaigns.

Naturally, the data collected with Paris traceroute should in theory enable somewhat deeper analysis, as the paths collected with this tool are more likely to correspond to actual paths. However, due to lack of parallelism and much lower success rates (Section 3.2), these datasets are much smaller in size and only capture paths to a smaller subset of the domains. For most of our analysis we instead focus on the end-to-end path lengths and RTTs reported by basic traceroute, and note that these metrics still are representative of the actual distances observed to these domains. For the analysis presented in this paper, this tool therefore provides sufficient accuracy.

**Heterogeneous environments with competing load:** The run times of example measurements differ substantially between PlanetLab nodes and can vary over time as the mix of competing loads on the nodes change. For example, during the initial measurement campaign (see Section 3) the run times of a large batch of traceroutes (to a fixed set of sample domains) differed by more than six times, and the fastest nodes in these experiments were among the slowest in experiments we ran three months later. To account for these speed differences, while trying to capture potential time varying traceroute effects, we carefully scheduled traceoute measurements to account for the run times on each individual node (Section 3) and perform all analysis on a pairwise basis, allowing us to account for different source-destination pairs having more/less measurement samples.

**Domain dependent path distances:** The route lengths and RTTs can differ substantially depending on the popularity of the sites. For example, the routes to popular services are typically shorter than the routes to less popular services [10], with the route lengths being closely related to the amount of traffic that they forward, and routes often differing both regionally and within the

same AS. To compare path lengths of IPv6 and IPv4 routes, it is therefore important to measure the paths to domains associated with different popularity classes and service categories. For popularity-based domain selection, we leverage the commonly used Alexa top-million list [29] and 16 per-category top lists [1]. A subtle challenge we address when using the Alexa top-1M list, is how to downsample the list in easily reproducible way that result in the exact same sample set [29]. For this purpose, we present a simple, deterministic domain sampling technique that we applied on lists available in public repositories [29].

## 3 Collection Methodology

### 3.1 Measurement framework

**Overview:** All campaigns run repeated traceroutes to the IP addresses returned by each PlanetLab's local DNS resolver for a pre-determined selection of domains. At the start of each campaign, we first distribute this domain list to every European PlanetLab node supporting IPv6. Throughout the duration of the measurement campaign, we then schedule multiple traceroute "batches" on each such node, where a "batch" includes a series of traceroutes to all IPv6 and IPv4 addresses that the local DNS resolver has returned for each domain. Domain-to-IP mappings are refreshed on a daily basis, with each batch job always starting off by checking whether new mappings have been obtained that day. If not, new mappings are obtained using the local DNS resolver of the node. At the end of each campaign, we run reverse DNS lookups and perform AS lookups to obtain additional information about all unique IP addresses observed. Finally, the data is downloaded from each node, merged into a single database, and analyzed. We next highlight some of the details associated with these steps and how they address the challenges discussed in Section 2.

**Domain sampling:** Using the *Alexa 1M Global* list [29] from May 13, 2019, we selected the first 100 domains (ranks 1-100) and the last 100 domains from each additional magnitude sample (i.e., ranks 901-1,000, 9,901-10,000, 99,901-100,000, and 999,901-1,000,000), as well as the top-50 domains from the Alexa top sites of 16 top-category lists [1]: Adult, Shopping, Arts, Society, Business, Health, Computers, Home, Games, Kids & Teens, Reference, News, Regional, Recreation, Science, Sports. Ignoring a very small number of duplicates, this results in a list of 1,300 domains. The smaller sample set allows us to run traceroutes for each domain and location multiple times per day, and the diversity in sample classes allows us to compare routes to domains across both domain popularities and domain categories. Finally, we again note that this sampling method allows others to easily rerun the experiments (Section 4.1) with the exact same sample set.

**Node selection:** As described in Section 2, PlanetLab's support team provided us with accounts and installed the necessary tools on the small, but full, set of European PlanetLab nodes supporting IPv6. No further sampling criteria were used.

**Daily DNS resolution and traceroute scheduling within a batch:** In the case that the local DNS server returns several IP addresses, all returned addresses are stored and used. In the case that the DNS server cannot resolve an IP address corresponding to the domain name, a prefix of "www" is added to the domain name before repeating the DNS lookup procedure. If this secondary DNS query also yields no results, we discarded the domain from the study. In total, only 5 out of the 1,300 sampled domains needed to be discarded. Finally, to keep unknown duplication to a minimum, we did not include CNAME pointer records in our dataset. Given the set of IP addresses, traceroutes within a batch of traceroutes were scheduled one domain at a time. For each domain, we first scheduled traceroutes to IPv4 addresses and then to IPv6 addresses. This ensures that IPv6 and IPv4 traceroutes to the same domain runs relatively nearby in time (typically within less than a minute). We leverage this in our analysis, as path lengths always are compared for the same source-destination pair (for which we then have many nearby sample pairs).

**Batch scheduling:** Due to time-varying loads and significant differences between the processing times on the different PlanetLab nodes, we decided to schedule batches at different intervals. In the first campaign ("May 2019"), we pre-scheduled periodic batch jobs (on 1, 2, 3 or 6 hour intervals) based on the maximum run lengths that we had observed at each location the days leading up to the actual collection period, and interrupted the batch jobs that did not fully complete within one such interval. This pruning resulted in some missing data for one of the eight PlanetLab nodes available during the first campaign. When planning the next campaign, we observed substantially different run times for the different nodes, prompting us to improve the methodology somewhat for the final two campaigns. In particular, we schedule new batch jobs to start at the top of the next even hour (as measured locally) following the ending of the previous batch jobs. For most of the locations, this results in batch jobs starting every 2, 4, or 6 hours with the basic traceroute tool ("Sept 2019"). However, for the second campaign (single threaded with Paris traceroute) we observed 12-hour intervals.

**Post campaign lookups:** At the end of each campaign, we collected the reverse DNS entry for each observed IP address as well as the AS number (as provided by RIPEstat [27]). To reduce the number of calls to the later API and to speed up the IP-to-AS mapping, we (i) converted the unique set of IP addresses into their binary form, (ii) cached looked-up entries, and (iii) used the AS number of cached entries whenever there already exists an entry in the cache for which the IP address fitted within its IP network mask. The choice to only run the lookups once per campaign is motivated by the high resource usage during these lookups (that otherwise would impact the data collection itself). Also, note that such mappings change much less frequently than the IP routes themselves, and that this choice is expected to have negligible impact on our results.

**Table 2.** Summary of measurement campaigns

| Short name | Duration | Dates (all 2019) | Method | Nodes | Traceroutes | Success |
|---|---|---|---|---|---|---|
| May 2019 | 1 week | May 14-20 | Baseline | 8 | 1,966,793 | 74% |
| Paris | 4 weeks | Aug. 11 - Sept. 8 | Paris | 6 | 265,206 | 22% |
| Sept. 2019 | 1 week | Sept. 18 - 24 | Baseline | 8 | 1,773,553 | 78% |

### 3.2 Overview of datasets

Table 2 summarizes some key differences and characteristics for the three measurement campaigns analyzed in this paper. Again, all three campaigns use the same sample list of domains (see "Domain sampling" above), are based on the same high-level methodology, and only differ by the adjustments made to account for differences in the traceroute tools (impacting parallelism) and how the run duration of individual samples differ across PlanetLab nodes and vary over time. For example, the "Paris" campaign lasted for four weeks (but only used one thread per node), while the other two campaigns used the basic traceroute tool and lasted a week each. Despite having much shorter collection duration, the two campaigns performed with the basic traceroute tool were able to perform many more traceroutes and resulted in 3.4-3.5 times higher success rates than when using Paris traceroute.

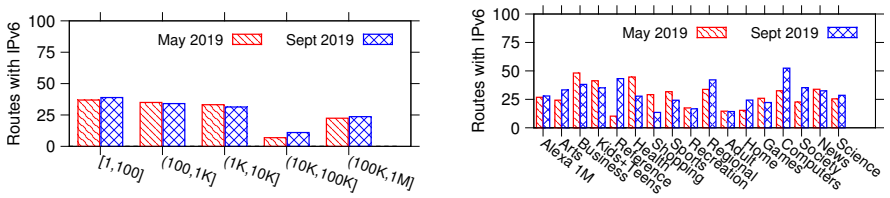
Given the far lower Paris traceroute success rate, we also ran Paris traceroute with alternative configurations. However, such alternative configurations resulted in even worse success rates. For example, in a 12 hour experiment using the 9 active PlanetLab nodes at that time, we obtained the following success rates with the main configuration options: 20% (5,966/30,372) when using the default option (i.e., UDP probes with default destination port 33457), which also is the option we used in the above experiments, 5% (1,606/30,392) when using UDP probes with destination port 53, 0.6% (178/30,349) when using ICMP probes, and 0.14% (42/30,345) when using TCP probes.

Due to the small success rate of Paris traceroutes, as observed from our example locations, in the following, we focus only on the two datasets using the baseline traceroute implementation (i.e., "May 2019" and "Sept. 2019"). Again, note that this tool allows use of parallel traceroutes and provided much higher success rates, but that we cannot trust the exact paths reported. While this limits any accurate analysis to comparing distances rather than the exact "paths" reported by the tool, it should be emphasized that the type of statistics analysis we present here are designed for the purpose of looking only at path distances and RTTs, not the exact paths.

## 4 Evaluation Method and Results

### 4.1 High-level analysis methodology

For the evaluation presented here, we use the measurements collected from the seven IPv6 enabled PlanetLab nodes that were active both in May 2019 and




(a) Alexa top-1M popularity ranking (b) Domain category

**Fig. 1.** Fraction of end-to-end pairings evaluated in different domain categories that were IPv6 enabled in May 2019 and Sept. 2019

Sept. 2019. By excluding the measurements for the two additional nodes, we ensure fairer longitudinal analysis of the changes that have happened over the four-month period between the datasets. First, note that this fixed set of PlanetLab nodes combined with using a fixed set of sample domains (see "Domain sampling" in Section 3) ensures that we use the same set of node-domain pairings for all the datasets. Second, to account for differences in the number of samples observed (to each domain) at each PlanetLab node, we apply a pairwise analysis and report summary statistics for sets of node-domain pairs. In particular, for most comparisons, we calculate the fraction of node-domain pairings of a subset of such pairings for which either IPv4 or IPv6 is deemed the "winner", as calculated using different pairwise statistics.

In addition to minimizing the effects of different nodes allowing for different number of measurements (e.g., due to load differences and their relative network speeds when performing the measurements) our pairwise analysis methodology also minimizes the effects of differences in the number of measurements per domain from a particular node (e.g., due to multiple IPs for some domains and one of the nodes not fully completing all of its batches in the first dataset). We also stress that the two dataset we focus on both were collected using the same traceroute tool, and only use Paris measurements for complementing analysis. Finally, while we only use a limited number of measurement nodes, we note that these nodes are public and therefore allow others to with some work (and extra help from the PlanetLab group) repeat our measurements and analysis (assuming those nodes remains active).

### 4.2 IPv6 deployment

Much prior work has considered the IPv6 adoption from different perspectives. While this is not the focus of this paper, to provide some context of end-to-end paths that we evaluate, here, we briefly (i) note that all end-to-end pairings that we observed were IPv6 enabled also had corresponding IPv4 paths, and (ii) report the fraction of pairings that we observed were IPv6 enabled. Figure 1 summarizes these results.

Across the domain categories and locations that we used for the evaluation, the number of IPv6 enabled paths were typically below 50% (for each category). In May 2019, 27.8% of all observed pairings were IPv6 enabled and in Sept. 2019,

**Table 3.** Summary table of pairwise distance comparisons

| | | Median winner (%) | | | Average winner (%) | | | 95% conf. win. (%) | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Metric | v.4 | v.6 | tie | v.4 | v.6 | tie | v.4 | v.6 | none |
| May'19 | IP hops | 15.4 | **77.5** | 7.0 | 21.1 | **78.7** | 0.2 | 19.9 | **77.5** | 2.6 |
| | AS hops | 14.3 | **59.3** | 26.4 | 17.1 | **79.6** | 3.3 | 16.0 | **78.0** | 6.0 |
| | RTTs | 46.0 | **54.0** | 0.0 | 47.2 | **52.8** | 0.0 | 33.1 | **44.7** | 22.2 |
| Sep'19 | IP hops | 14.4 | **77.6** | 8.0 | 20.2 | **79.8** | 0.0 | 19.4 | **79.0** | 1.6 |
| | AS hops | 10.3 | **55.4** | 34.3 | 15.4 | **81.5** | 3.1 | 13.3 | **78.7** | 8.1 |
| | RTTs | 36.2 | **63.8** | 0.0 | 31.3 | **68.7** | 0.0 | 25.7 | **59.0** | 15.3 |

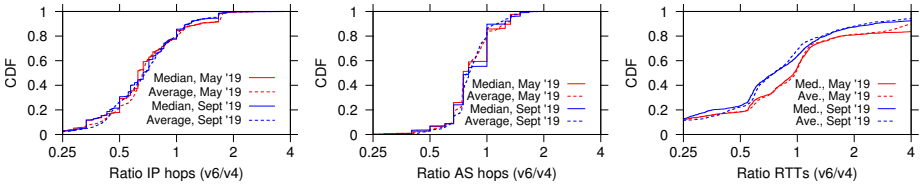
29.2% of all observed pairings were IPv6 enabled, suggesting a small increase by 1.44% over this period. Within the Alexa top-1M dataset (Figure 1(a)), the fraction of IPv6 enabled paths is highest for the subsets with domains ranked in the top-10K subset (i.e., [1,100], (100, 1K], and (1K,10K]). When instead considering the statistics for the top-50 domains of different domain categories (Figure 1(b)), we observed individual categories with above ("Computers") or close to 50% IPv6 enabled pairings, including two categories ("References" and "Computers") for which the fraction of IPv6 enabled paths increased substantially between May 2019 and Sept. 2019: 323% (10.2%→43.2%) and 61% (32.5%→52.4%), respectively.

### 4.3 High-level distance comparisons

Table 3 summarizes the percent of pairings for which IPv4 and IPv6 are deemed the "winner" in the two datasets ("May 2019" and "Sept. 2019"), using three distance metrics (IP hops, AS hops, and RTTs) and three statistics (median, average, and a 95% confidence test on the average difference).

**Statistics:** For each pair, the median and average statistics are trivially calculated over all observations. Here, we simply report the percent of pairings for which these statistics are lower (i.e., fewer hops or shorter RTTs) for IPv4 and IPv6, respectively, as the fraction of "winners". In the case that the statistics are the same, we report a "tie" for that pairing. With the exception for the median number of AS hops, ties are relatively rare for these metrics.

For the 95% confidence tests, we use one-sided t-tests for the paths associated with each of the two protocol versions and report the percent of cases where the null-hypothesis that the metrics are the same can be rejected in favour of the alternative hypothesis that the (average) paths associated with that particular protocol are shorter with a confidence level of 95%. In the case that both tests fail, we list the pairings under the "none" column (indicating that neither is a significant winner). Note that the fraction of "none" entries always should be greater than the fraction of "ties" for the average statistic and that the 95% confidence test in general provides greater statistical insights into which differences are significant than the other two statistics. The average statistics provide insights regarding in which direction the "none" cases are leaning, and the median statistics (typically considered more robust to outliers than averages)




(a) IP hops (b) AS hops (c) RTTs

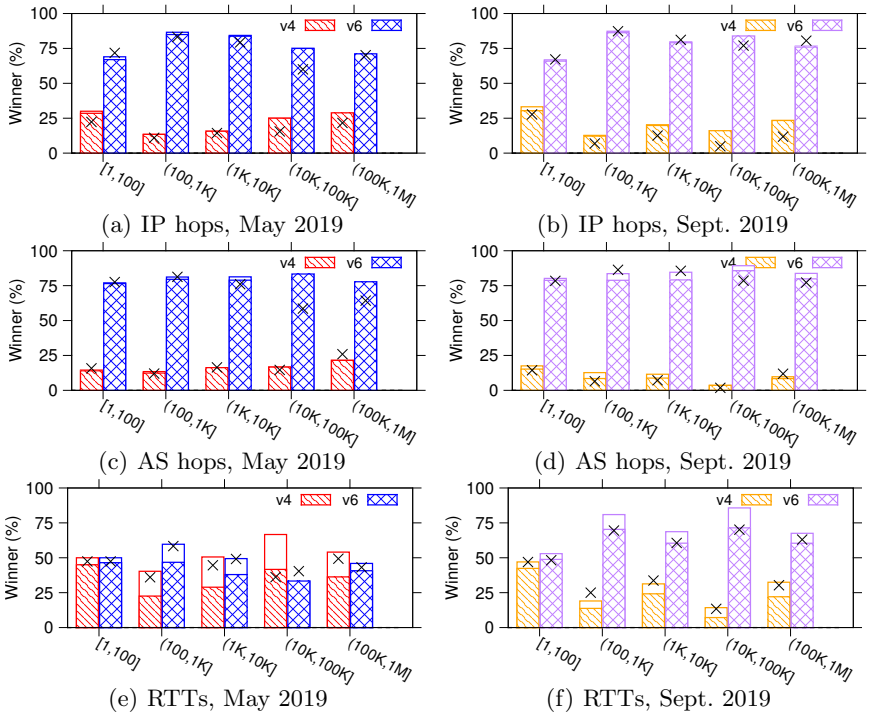
**Fig. 2.** CDFs of the ratio of the pairwise distances using IPv6 and IPv4, as measured using (a) IP hops, (b) AS hops, and (c) RTTs

provides a complementing perspective into which set of paths are shorter more robust to outliers.

**IPv6 most frequent winner in all cases:** The IPv6 paths have the largest fraction of pairwise winners across all three metrics, all three statistics, and for both datasets. To highlight this, the table uses bold text to indicate the set of paths with the most winners for each of the 18 cases (3×3×2). Furthermore, while we observe an increase in the fraction of IPv6 winners in 8 out of 9 cases (median AS hops being the exception), the differences between the datasets are only substantial for the three RTT cases, for which we see the following increases: 54.0%→63.8%, 52.8%→68.7%, and 44.7%→59.0%. We next analyze each distance metric separately.

**IP and AS hops:** Both the number of IP hops and AS hops are significantly shorter (95% confidence) in a 77-79% of the pairwise cases observed. The shorter hop-count lengths can also be observed when considering the ratios of the pairwise IP hop counts (Figure 2(a)) and AS hop counts (Figure 2(b)) using IPv6 and IPv4. Here, smaller ratios mean that the IPv6 paths has less visible hops. We also note that the values reported for the median and average statistics in Table 3 simply refers to the point at which the curves in Figure 2 have a ratio of 1. These figures not only emphasize that IPv6 paths are shorter, but that they in some cases are substantially shorter. For example, in nearly 20% of the cases the number of IP hops are half of what was observed using IPv4. Again, we expect that these cases often correspond to cases where IP tunnels have been used; something we have manually validated for some cases. Furthermore, we note that the median and average curves follow each other relatively closely (with the average curves being smoother) across the two datasets.

**RTTs:** Interestingly, when considering the RTTs, we see a relatively lower but increasing fraction of paths for which IPv6 is deemed the winner (Table 3). For example, using the median statistic the fraction increases from 54.0% to 63.8%; with the average statistic the fraction increases from 52.8% to 68.7%, and with the 95% confidence test statistic the fraction increases from 44.7% to 59.0%. The improving IPv6 RTTs are perhaps even more visible in Figure 2(c), as they result in in a clear shift of the relative RTT ratios. Comparing this with the (typically higher, but stable fractions) for the IP and AS hop metrics (which remained relatively stable), the IPv6 paths' relative RTTs improve significantly.




(a) IP hops, May 2019 (b) IP hops, Sept. 2019

(c) AS hops, May 2019 (d) AS hops, Sept. 2019

(e) RTTs, May 2019 (f) RTTs, Sept. 2019

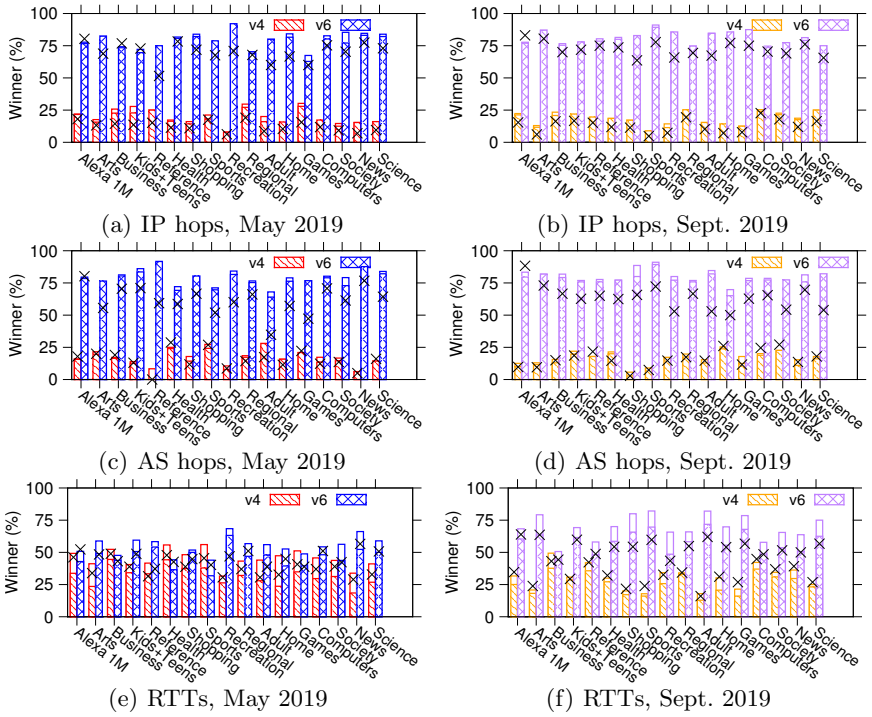
**Fig. 3.** Fraction of instances that the IPv6 path (blue/purple) and IPv4 path (red/orange) is the winner, as broken down per the rank of the top-1M domains

Overall, the lower IPv6 RTTs (skew towards lower ratios observed in Figure 2(c)) and further improvements of the IPv6 RTTs (slight shift to the left of curves) suggest that the IPv6 paths perform very well and that current deployment examples are encouraging.

### 4.4 Rank-based distance comparisons

Figure 3 shows the fraction of pairings that the IPv6 path (blue/purple) and IPv4 path (red/orange) is the winner for domains with different ranking. The full bars show the results using the average statistic, the filled region of the bars shows the winners using the 95% confidence tests, and the crosses (×) show the results for the median statistic. Here, we include a pair of plots for each of the three metrics: IP path lengths ((a) and (b)), AS path lengths ((c) and (d)), and RTTs ((e) and (f)). Changes over time are seen by comparing the left (May 2019) and right (Sept. 2019) figure.

In general, our previous observations are consistent across the different domain ranks. First, the IP hop counts and AS hop counts are the clear winner in most cases, and these ratios do not change much over time. Second, when considering the RTTs, with exception of the top-100 ranked domains (for which




(a) IP hops, May 2019 (b) IP hops, Sept. 2019

(c) AS hops, May 2019 (d) AS hops, Sept. 2019

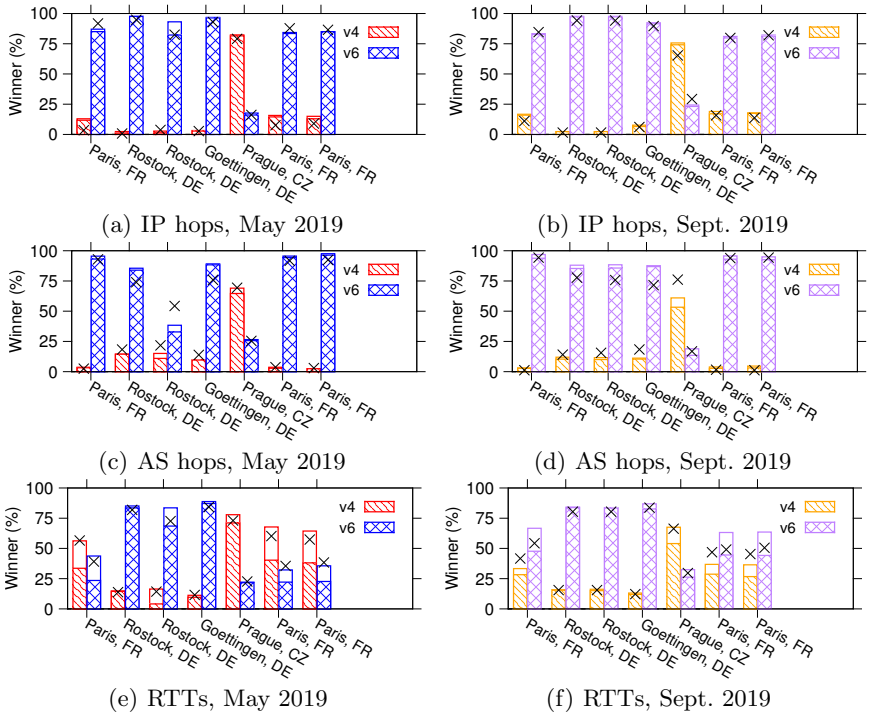
(e) RTTs, May 2019 (f) RTTs, Sept. 2019

**Fig. 4.** Fraction of instances that the IPv6 path (blue/purple) and IPv4 path (red/orange) is the winner, as broken down per the domain category

there are small differences), we observe a significant increase in pairings for which IPv6 is the winner.

### 4.5 Category-based distance comparisons

The corresponding results for the 50 top ranked domains of 16 Alexa domain categories (plus the top-1M set itself) are shown in Figure 4. Here, we again break down the results per dataset (column), metric (row), and statistic (bars, filled bars, and crosses). Similar as for the rank-based results, our main observations are relatively consistent across the different categories. First, the fraction of paths for which the IPv6 paths have shorter IP and AS hop counts than the corresponding IPv4 hop counts are consistently (much) higher for all categories than the fraction of pairs that IPv4 would be the winner, and the differences remained relatively consistent between the two snapshots. Second, across the domain categories, we observe an increasing fraction of pairings for which the IPv6 RTTs are lower than the corresponding IPv4 RTTs. This has also resulted in an increase in the number of categories for which there are more IPv6 winners than IPv4 winners. For example, IPv6 has gone from having more winners in 13 out of the 17 categories (May 2019) to having more winners in all of the categories (Sept. 2019), regardless of statistic.




(a) IP hops, May 2019

(b) IP hops, Sept. 2019

(c) AS hops, May 2019

(d) AS hops, Sept. 2019

(e) RTTs, May 2019

(f) RTTs, Sept. 2019

**Fig. 5.** Differences in the fraction of instances that the IPv6 path (blue/purple) and IPv4 path (red/orange) is the winner, as broken down per PlanetLab node

### 4.6 Impact of PlanetLab node

To better understand the impact of the selection of origin sources, we next discuss to what degree the results also were consistent across the PlanetLab nodes used for our analysis.

While we did observe some significant differences in the fraction of IPv6 winners at each node, the observations were relatively consistent. For example, for the hop-count based results, IPv6 had a larger fraction of winners for 6 out of 7 of the nodes in both May 2019 and Sept. 2019, whereas the fraction of nodes for which IPv6 had more winners when using the RTT as the metric increased from 3/7 to 6/7. The node that stood out the most was `ple1.cesnet.cz` (located in Prague, CZ), which consistently had IPv4 as the winner across all metrics and datasets. However, also for this node, the relative changes in the fraction of IPv6 winners when using the RTTs metric increased noticeably from May 2019 to Sept. 2019. Figure 5 shows a per-node breakdown.

## 5 Related Work

Over the years, many measurement studies have focused on IPv6 adoption [12, 13, 16, 21, 23, 30, 31]. Although many such studies are a few years old, they com-

bine for a good picture of the slow IPv6 adoption across geographic regions, user equipment technologies, edge networks, ISPs, content providers, and other entities in the end-to-end ecosystem. However, less work has focused on the relative end-to-end IPv6 performance and how it may be affected by the lack of full end-to-end adoption.

As discussed in the introduction, Giotsas et al. [16] showed in 2015 that the performance of IPv6 paths can be significantly hurt by IPv6 tunnels and a less connected transit-free clique. However, since then, further adoption has taken place. As validated by our study, the IPv6 paths and their performance are therefore expected to improve over time as tunneled paths are replaced with native paths and IPv6 AS relationships mature. Giotsas et al. [16] also showed that Hurricane Electric (HE), a prominent provider of this technique, already at that time significantly contributed to the AS connectivity and that the use of their peerings was quickly increasing.

Bajpai et al. [4] investigated the time it takes to complete the initial TCP handshake over an IPv6 and IPv4 network to the top-100 dual-stack websites. Using measurements from both residential and research networks, they identified several cases where CDN caches were present at the edge networks for IPv4, but not for IPv6. This resulted in relatively higher connection establishment times with IPv6. While they observed some improvements in IPv6 connection times over time, more recent TCP connection establishment measurements towards YouTube media servers suggest that both TCP connection establishment times and startup delays are higher over IPv6 [5].

In other recent work, Goel et al. [17] measure RTTs, DNS lookup times, and page load times using the Akamai monitoring system, and show that IPv6 performs better than IPv4 in US cellular networks. Pujol et al. [26] use DNS and flow-level statistics from an ISP to show that RTTs observed in the backbone are similar for IPv6 and IPv4 (e.g., 80% of RTTs within 10ms of corresponding IPv4 RTTs).

Other, somewhat older studies, have evaluated the IPv6 performance using HTTP requests to experimental Google web service hostnames [11], using pings from three US locations to globally distributed dual-stack name servers [6], by performing download speed tests [24], and by measuring page load times [13] of popular websites. Although older, we note that the two later studies (i.e., [13,24]) made some interesting observations that suggest that IPv6 performance typically was comparable to IPv4 performance when AS-level forwarding paths were the same, but that the performance typically was much worse otherwise. Based on the adoption trends observed by Giotas et al. [16], which suggest that the IPv6 AS-level topology (at least in 2015) slowly is converging towards the IPv4 topology, we would therefore expect that IPv6 performance will improve relative to IPv4 performance over time. This is also supported by taking a (close to) 10-year perspective and comparing our average, median, and 95%-ile results with those obtained by Berger [6] in 2010. While the IPv4 ping times that they measured between their three US-based locations and dual-stack name servers associated with different geographic regions were almost always faster than the correspond-

ing IPv6 ping times, we observe IPv6 to be the winner in most of cases today (from our measurement locations). While some of the above papers give a glimpse into the relative performance that clients may see when using IPv6 rather than IPv4, most of these works are old, and none of the works capture, compare, or contrast the status observed for different domain categories. Furthermore, none of the previous works use pairwise hypothesis testing to quantify the number of paths for which IPv6 (or IPv4) is the winner.

Finally, we note that yet others have developed techniques to scan the IPv6 address space [14, 15, 28], to study the stability of IPv6 in the control/data planes [22], to discover the IPv6 topology [9] or address space [25], and to pair addresses of dual-stacked DNS resolvers or IP end points [7,8]. We consider these works orthogonal to ours.

## 6 Conclusions

In this paper we presented the methodological challenges and results from a measurement study in which we compared the relative end-to-end distances when using IPv6 and IPv4 between PlanetLab nodes in Europe and selected domain sets. The paper provides a use-case driven report of running IPv6 experiments on PlanetLab, highlights the lack of IP support among PlanetLab nodes, and provides a statistical methodology that uses hypothesis testing and pairwise comparisons to provide insights into the current IPv6 paths performance, relative to that of IPv4, while accounting for current testbed and traceroute limitations. Our analysis shows (among other things) that despite significant use of IP tunnels (e.g., much shorter IP and AS hop counts), the RTTs of the IPv6 paths are now relatively faster than the corresponding IPv4 paths in the majority of cases when they are available, and the fraction of pairings for which this is the case has increased notably between May 2019 and Sept. 2019 across all five rank categories and almost all 16 domain categories considered. These findings suggest that the IPv6 end-to-end path performance is continuing to improve and most often already outperform IPv4 path performance. These performance improvements may be due to careful deployment by individual operators, but may also help incentivize further deployment by others.

The limited IPv6 deployment among PlanetLab nodes in Europe restricted us to a smaller number of measurement locations. Interesting future work include applying the pairwise hypothesis-based methodology presented here on similar datasets collected from other locations and/or collected at different points in time. An interesting measurement effort worth mentioning here is an online tool created by Geoff Huston [19], which provides country-by-country statistics and data that potentially could be used for such analysis.

**Acknowledgement**

We thank Burim Ljuma at PlanetLab for helping us setting up PlanetLab with IPv6 connectivity. We also thank the developers of Paris traceroute for their continuing efforts to improve the tool.

## References


1. Alexa - top sites by category: Top, https://www.alexa.com/topsites/category, visited on 2019-03-07
2. traceroute(8) - Linux manual page, http://man7.org/linux/man-pages/man8/traceroute.8.html, visited on 2019-02-14
3. Augustin, B., Cuvellier, X., Orgogozo, B., Viger, F., Friedman, T., Latapy, M., Magnien, C., Teixeira, R.: Avoiding traceroute anomalies with Paris traceroute. In: Proc. IMC (2006)
4. Bajpai, V., Schönwälder, J.: IPv4 versus IPv6–who connects faster? In: Proc. IFIP Networking (2015)
5. Bajpai, V., Ahsan, S., Schönwälder, J., Ott, J.: Measuring YouTube content delivery over IPv6. ACM CCR (Oct 2017)
6. Berger, A.: Working paper on comparison of performance over IPv6 versus IPv4. Tech. rep., Akamai Technologies (2011)
7. Berger, A., Weaver, N., Beverly, R., Campbell, L.: Internet nameserver IPv4 and IPv6 address relationships. In: Proc. IMC (2013)
8. Beverly, R., Berger, A.: Server siblings: Identifying shared IPv4/IPv6 infrastructure via active fingerprinting. In: Proc. PAM (2015)
9. Beverly, R., Durairajan, R., Plonka, D., Rohrer, J.P.: In the IP of the beholder: Strategies for active IPv6 topology discovery. In: Proc. IMC (2018)
10. Chiu, Y.C., Schlinker, B., Radhakrishnan, A.B., Katz-bassett, E., Govindan, R.: Are we one hop away from a better internet? In: Proc. IMC (2015)
11. Colitti, L., Gunderson, S.H., Kline, E., Refice, T.: Evaluating IPv6 adoption in the internet. In: Proc. PAM (2010)
12. Czyz, J., Allman, M., Zhang, J., Iekel-Johnson, S., Osterweil, E., Bailey, M.: Measuring IPv6 adoption. In: Proc. ACM SIGCOMM (2014)
13. Dhamdhere, A., Luckie, M., Huffaker, B., kc claffy, Elmokashfi, A., Aben, E.: Measuring the deployment of IPv6: topology, routing and performance. In: Proc. IMC (2012)
14. Fukuda, K., Heidemann, J.: Who knocks at the IPv6 door?: Detecting IPv6 scanning. In: Proc. IMC (2018)
15. Gasser, O., Scheitle, Q., Gebhard, S., Carle, G.: Scanning the IPv6 internet: Towards a comprehensive hitlist. In: Proc. IFIP TMA (2016)
16. Giotsas, V., Luckie, M., Huffaker, B., kc Claffy: IPv6 AS relationships, cliques, and congruence. In: Proc. PAM (2015)
17. Goel, U., Steiner, M., Wittie, M.P., Flack, M., Ludin, S.: A case for faster mobile web in cellular IPv6 networks. In: Proc. ACM MobiCom (2016)
18. Google: Google IPv6 adoption statistics, http://www.google.com/intl/en/ipv6/statistics.html, visited on 2020-10-01
19. Huston, G.: V6/v4 RTT comparison by country (ms) (2020), https://stats.labs.apnic.net/v6perf, visited on 2020-10-01
20. Jobst, M.E.: Traceroute anomalies. In: Seminar Future Internet (2013)
21. Karir, M., Huston, G., Michaelson, G., Bailey, M.: Understanding IPv6 populations in the wild. In: Proc. PAM (2013)
22. Livadariu, I., Elmokashfi, A., Dhamdhere, A.: Characterizing IPv6 control and data plane stability. In: Proc. IEEE INFOCOM (2016)
23. Nikkhah, M., Guerin, R., Nikkhah, M.: Migrating the internet to IPv6: An exploration of the when and why. IEEE/ACM Trans. on Networking (Aug 2016)

24. Nikkhah, M., Guérin, R., Lee, Y., Woundy, R.: Assessing IPv6 through web access a measurement study and its findings. In: Proc. ACM CoNEXT (2011)
25. Plonka, D., Berger, A.: Temporal and spatial classification of active IPv6 addresses. In: Proc. IMC (2015)
26. Pujol, E., Richter, P., Feldmann, A.: Understanding the share of IPv6 traffic in a dual-stack ISP. In: Proc. PAM (2017)
27. RIPE NCC: RIPEstat data API, https://stat.ripe.net/docs/data_api#network-info, visited on 2019-05-03
28. Rohrer, J.P., LaFever, B., Beverly, R.: Empirical study of router IPv6 interface address distributions. IEEE Internet Comput. (Jul/Aug 2016)
29. Scheitle, Q., Hohlfeld, O., Gamba, J., Jelten, J., Zimmermann, T., Strowes, S.D., Vallina-Rodriguez, N.: A long way to the top: Significance, structure, and stability of internet top lists. In: Proc. IMC (2018)
30. Zander, S., Andrew, L.L., Armitage, G., Huston, G., Michaelson, G.: Investigating the IPv6 teredo tunnelling capability and performance of internet clients. ACM CCR (Oct 2012)
31. Zander, S., Andrew, L.L., Armitage, G., Huston, G., Michaelson, G.: Mitigating sampling error when measuring internet client IPv6 capabilities. In: Proc. IMC (2012)