Evaluating Performance and Inefficient Routing of an Anycast CDN

Conference Paper · June 2019

DOI: 10.1145/3326285.3329049

5 authors, including:

Weizhen Dang
Tsinghua University
5 PUBLICATIONS **18** CITATIONS
SEE PROFILE

Haibo Wang
Tsinghua University
10 PUBLICATIONS **24** CITATIONS
SEE PROFILE

Jilong Wang
Tsinghua University
54 PUBLICATIONS **200** CITATIONS
SEE PROFILE

Jessie Hui Wang
Tsinghua University
49 PUBLICATIONS **347** CITATIONS
SEE PROFILE

Evaluating Performance and Inefficient Routing of an Anycast CDN

Jing'an Xue
xuejingan@huawei.com
Huawei Technologies

Weizhen Dang, Haibo Wang
wang-hb15,dangwz16@mails.tsinghua.edu.cn
Tsinghua University

Jilong Wang, Hui Wang*
wjl,hwang@cernet.edu.cn
Tsinghua University

ABSTRACT

Anycast has been increasingly deployed for content delivery networks to map clients to their nearby replicas, which relies on the underlying routing. However, the simplicity of operation comes at cost of less precise client-mapping control. Although many works have measured anycast DNS, anycast CDNs, with different service goals and engineering, are still not fully understood. In this paper, we design novel methods and combine large-scale traceroute and HTTP measurement to evaluate the overall client-proximity and inefficient routing of the largest anycast CDN, Cloudflare. We find that 90% paths traverse only 2-4 ASes, which highlights its direct networks providers. By further identifying and characterizing direct providers at finer granularity of facilities, we quantitatively shows that Cloudflare unevenly uses few large transit providers to delivery the majority of contents. Inspired by the observations, we propose an anycast routing pathology and diagnosis methodology. Investigation reveals that few huge providers have outsized impact in that they are not only related to many inter-domain inflations, but also have path inflation inside their own networks, thus deserving priority focus when troubleshooting.

CCS CONCEPTS

• **Networks** → **Network measurement**.

KEYWORDS

content delivery, anycast, routing, measurement

1 INTRODUCTION

Content delivery networks (CDNs) provide low-latency and reliable services by employing widely distributed servers and mapping (scheduling) clients to available proximal ones. Client-mapping mechanism, as the key part of CDNs, primarily includes two popular ones: traditional DNS-based redirection [1, 2] and the rising anycast mechanism. The former offers fine-grained client-mapping by dynamically setting DNS resolution for different clients, but has disadvantages of costly control infrastructure and inaccurate client localization caused by local DNS (LDNS) not representative of clients [3]. Meanwhile, anycast offers clients with a single-address abstraction for distributed services [4]. In Internet IP anycast, a set of anycast nodes/replicas at different regions announce the same IP addresses via BGP, the de-facto inter-domain routing protocol. Traffic originated by clients are routed to the "closest" node in terms of metrics used by routing systems. This primitive provides several advantages, such as resilience to DDoS attack [5] and inherent server-client proximity, thus is widely used in critical Internet infrastructure services. It has been applied to DNS root servers [6–9] since the early 2000s. Recently, it is increasingly used for client mapping in CDNs [10, 11].

Anycast CDNs bring many benefits, *e.g.,* they do not need investment in control infrastructure, can avoid LDNS problem and have very short fail-over time. However, it also comes with some well-known challenges. First, it is not friendly to stateful services in that routing change can interrupt ongoing sessions. Nonetheless, this does not seem to be an issue, as evidenced by many operational anycast CDNs such as Cloudflare. Second, anycast essentially relies on the underlying routing, which however is not directly aware of performance quality such as latency or server load. Therefore, it is important to evaluate how well anycast performs for CDNs, and to investigate the characterization and causes of inefficient anycast routing.

Previous studies on anycast mainly include two aspects. One is to develop and improve methodologies to identify anycast nodes [7, 12–14], the other is to evaluate performance of critical anycast services, mainly DNS, in terms of client proximity [6, 8–10, 15], stability [10, 15, 16], deployment scheme [6, 9, 17], reliability [5] and load controllability [15, 18]. However, there are only a few works about operational anycast CDNs, whose performance and efficiency is still not fully understood. On one hand, the majority of state-of-art works focus on anycast DNS, which is stateless while CDNs are stateful. Operators may employ different peering and routing policies according to their service goals. For instance, there exist many local anycast DNS nodes, *i.e.,* announced with BGP "no-export" attribute, which intentionally limit where queries may come from, mainly for load management; rather, CDNs' priority is client proximity. On the other hand, there lacks of systematic routing efficiency analysis of anycast CDNs. Only recently, a measurement study about Bing [10] reveals its performance of proximity and stability, but does not show fundamental routing analysis.

In this paper, we conduct a large-scale measurement on the largest anycast CDN Cloudflare and present an in-depth anycast routing inefficiency analysis. Specifically, our main goals can be specified as: 1) evaluate the overall proximity performance that anycast can be tuned at global-scale CDNs with O(100) replicas; 2)

*Also affiliated with Beijing National Research Center for Information Science and Technology. This research is supported by National Key R&D Program of China (2016YFB0801301). Jilong Wang is the corresponding author.

IWQoS '19, June 24–25, 2019, Phoenix, AZ, USA

ACM ISBN 978-1-4503-6778-3/19/06…$15.00
https://doi.org/10.1145/3326285.3329049

estimate how much high latency is contributed by anycast inefficiency, *i.e.,* clients routed to a distant, high-latency replica over the close, low-latency one; 3) systematically diagnose the characteristics and causes of inefficient anycast routing.

Evaluating the performance and inefficient routing of anycast CDNs can help CDN providers effectively manage and improve CDN performance. However, it does not only require a significant amount of effort, but is also technically challenging. First, the evaluation target is not publicly straightforward as DNS, thus needs to be identified with specifically designed methods. Second, no single measurement platform can simultaneously satisfy function and capacity (both concurrent and total quota) demand, which requires a careful measurement design. Third, raw routing data analysis, especially Internet-wide root cause diagnosis, is notoriously challenging to analyze because practical routing policy is almost artificial and proprietary. This kind of research needs to explore routing characteristics under specific scenarios, *e.g.,* cellular network.

Contributions. We first exploit system features to identify fine-grained targets. We then novelly design and combine two kinds of active measurements: HTTP measurement from a large-scale proxy platform Luminati with 899062 proxies and traceroute measurement from 1632 RIPE Atlas probes. To the best our knowledge, we are among the first and largest ones that evaluate anycast CDNs.

We then evaluate the mapping quality in terms of overall geographical proximity, latency and anycast efficiency. Basically, our results show good overall client-proximity, about 81% of anycast replicas serve at least 85% of corresponding clients from the same continent the replicas are located in. But there exist a few exceptional replicas whose client distribution are very scattered and suboptimal. Further, the vast majority of anycast paths are performing well with normal latency and efficiency ratio, nearly consistent with geographical proximity. However, although anycast inefficiency only accounts for a small fraction of inflated paths, it makes latencies exhibit a 5-fold increase.

We next propose an diagnosis methodology for inefficient anycast routing at finer granularity of facilities. Specifically, we find ~90% paths traverse only 2-4 ASes and the direct ISPs may have several point of presences (PoP). We therefore propose an routing pathology of inter-domain inefficiency and intra-direct ISP inefficiency, and further design identification and verification method.

At last, we diagnose the characteristics and causes for inefficient anycast routing. We first demonstrate traditional models are not appropriate for anycast scenario. We then find that the scale of direct ISPs are extremely imbalanced and the first 3% ones, *e.g.,* AS10310, carry the majority paths, which inspires us to leverage metrics of direct ISPs to characterize inefficient routing. Investigation reveals that few direct Tier-1 providers have outsized impact in that they are not only related to the majority of inter-domain inflations, but also have path inflation inside their own networks, thereby deserving priority attention when troubleshooting.

2 BACKGROUND AND RELATED WORK

Many efforts have been devoted to measure and understand how anycast performs in critical infrastructure services. The first step to evaluate anycast is to uniquely identify the anycast nodes that clients are mapped to [7, 12–14], since all replicas share the same addresses. Most studies [7, 9] on DNS use a special DNS query type (**CHAOS type**); corresponding replies would include an unique server identifier conventionally configured by operators. For HTTP-based services such as CDNs, **custom HTTP headers** [10, 11] generally function in the same way, but identifiers are specific to service providers.

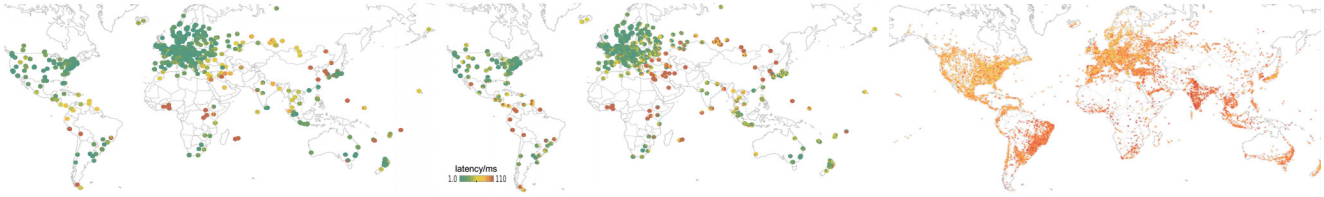
Most measurement works primarily characterize the following aspects of anycast. First, client-server *proximity* is the most basic and important feature, which is also our focus in this paper. It can be measured in geographic or network dimension. *Geographic metrics* include absolute distance or relative distance rank [8]. Some works also use **catchment** [7, 8], which means the clients (areas) each anycast node serves (covers). *Network metrics* generally include *latency* or *route path length* [9, 15]. To further evaluate the **efficiency** of anycast deployment, many works compare the latency between clients and **anycast-chosen nodes** to the lowest latency among all **potential nodes**, which *explore potential path diversity and could have been chosen.* To achieve this, each replica needs to be uniquely addressable to be directly and deterministically queried. In fact, each DNS replica usually is configured with both anycast and unicast IPs. Several studies [9, 10, 15] refer to the difference of anycast latency and lowest potential latency as **latency inflation** or stretch factor; Colitti *et al.* [6] refer to the ratio of them as **efficiency factor**. These metrics quantitatively measure how well the underlying routing system does in selecting the best node for a given client.

Stability is another metric [6, 8, 10, 15, 16]. While node switch/flap is of little importance for stateless services such as UDP-based DNS, it may pose problems for stateful services such as TCP-based CDN. Fortunately, measurement studies suggest it happens rarely. For instance, by passively analyzing the server log of K-root, Colitti *et al.* [6] show only 0.06% of queries had node switch and all come from 1.1% clients; similarly for anycast CDN Cachefly [19], only 0.017% TCP sessions had node switch during measurement. It has been proven not an issue in production environment since many companies [10] have been successfully operating anycast CDNs.

Deployment scheme [6, 9, 17] are often evaluated for DNS services, where each anycast node is either global or local. Local nodes intentionally limit where queries may come from by announcing BGP route with *no-export* or *no-advertise* attributes; global nodes are expected to have higher capacity to serve loads across the Internet, often announced with AS-path prepending. An anycast scheme is *flat* if it contains only global nodes, otherwise it is *hierarchical or hybrid.* Sarat *et al.* [17] show that hierarchical scheme has higher stability and availability while flat scheme achieves better proximity.

Additionally, *load controllability and reliability* [5] is also the concern of many researchers. Since the underlying routing is not aware of server load, operators need flexible means to adjust load distribution to ensure server capacities match demands. By using techniques such as AS-path prepending, studies[15, 18] prove that load is controllable. Moreover, anycast can effectively defend extreme burst load such as DDoS attack by inherently preventing distributed traffic from aggregating.

Only a few works are about operational anycast CDNs, including early Cachefly [19], Bing [10] and LinkedIn [11], whose scale range



(a) Latency to anycast targets (b) Latency to unicast targets (c) HTTP proxies

Figure 1: Geographic distribution of traceroute probes (a-b) and http proxies (c)

from several to dozens of nodes. This paper focuses on the largest anycast CDN Cloudflare by far. It has 117 nodes in 6 continents as of Oct 2017, among which 21 nodes are excluded from our measurement since they are not anycasted in China and operated by its partner Baidu. The remaining 96 nodes are distributed in 96 cities of 57 countries.

3 METHODOLOGY AND DATASET

We use anycast efficiency, as noted in Section 2, to evaluate whether anycast routing chooses the best path (**benchmark path**) among all potential paths and compare the chosen path with benchmark path to analyze anycast inefficiency. Potential path latency is deterministically obtained by measuring the unicast addresses of each nodes. Consequently, to answer the 3 key questions, our concrete measurement tasks can be specified as: (1) leverage custom HTTP header to obtain client-server mappings from extensive vantage points to approximate overall proximity; (2) use traceroute to measure paths and latencies to anycast-chosen nodes and potential replicas to evaluate anycast efficiency.

Achieving these is not straightforward because no single measurement platform can satisfy both function and capacity requirement at one time. In terms of function, there exits only a few proper platforms that can perform traceroute or HTTP measurement, with merely O(1k) vantage points. For correlation analysis, vantage points of the two functions are also expected to be aligned, which need a lot of test work to find overlap as much as possible. With regards to capacity, a probe needs to measure the anycast target and candidate targets at the same time for fair comparison. However, Atlas can only issue HTTP measurements to a few pre-defined targets and have expensive traceroute cost and strict concurrent limitation. We thus design and combine two kinds of active measurements: HTTP measurement from a large-scale proxy platform Luminati and traceroute measurement from RIPE Atlas probes. Next, we first propose methods to identify fine-grained measurements targets, *i.e.,* Cloudflare's anycast and unicast addresses. We then describe the detailed measurement methodology and datasets.

3.1 Identifying targets

Unlike DNS services, whose anycast and unicast addresses are publicly available, we need to exploit Cloudflare's system features to identify the targets as complete and fine-grained as possible. Specifically, we need to first identify its routable prefixes in the Internet, then confirm whether each prefix is anycast or unicast, and further associate it with a specific replica if it is unicast.

Routable prefixes. Cloudflare officially discloses its address space as continous IP blocks [20] and claims to announce same anycast prefixes from all replicas via AS13335 to its peering[1] Internet Service Providers (ISPs) [21], hereafter termed as **direct ISP**. Therefore, we obtain all routable prefixes originated by AS13335 in Routeveiws/RIPE NCC, the BGP control-plane monitoring systems.[2] In most cases, different monitors have consistent views, but there exist a very few prefixes observed with different lengths at different monitors due to the well-known complexity of distributed routing system. We use the most specific (longest) prefixes and finally get 548 routable prefixes.

Anycast targets. We first briefly describe the critical system features that can be leveraged to identify if addresses are unicast or anycast, and then explain how we accomplish the goal. When requesting a customer Web site delivered by Cloudflare, a client first gets resolved anycast IPs and then initiates a HTTP request to one of the IPs, which are theoretically equivalent since Cloudflare announces all anycast prefixes from all nodes identically. If the anycast-chosen replica has ever cached the requested content, it will reply immediately; otherwise, it will fetch the requested content with an **unicast IP** from the *origin Web site*, then cache the content and reply the client. Besides, for convenient debugging and monitoring, when replying clients and requesting *origin sites*, a replica would insert some custom HTTP headers [22], including *CF-Ray* to identify itself with IATA airport codes, *e.g.,* LAX stands for Los Angeles.

Note that anycast address is used for service side rather than request side, since the corresponding reply cannot be deterministically routed to the requester replica; instead, the responser's location decides which replica the reply lands. Therefore, an intuitive way to identify if a prefix is anycasted is to test if it is serving customer sites of Cloudflare. We thus identified a large quantity of customer sites and accumulate the corresponding IPs. Specifically, Web sites use DNS CNAME redirection or domain hosting to enable their content delivery services. We measure and record complete DNS resolution chains of Alexa Top 1M sites, and then match their CNAME or NS records with semantic regular expressions as prior works [1]. This results in 91556 customer sites, which accumulates 122 unique anycast prefixes.

[1] *Peering* loosely means any kind of connection between two ASes instead of commercial relationship.

[2] We also match publicly disclosed IP range with BGP control-plane announcements, which reveals three more ASes, *i.e.,* AS132892, AS202623 and AS395747. Whois information confirms they belong to Cloudflare, but this paper only focus on AS13335 since other ASes are not officially claimed and originate extremely few prefixes.

Later measurements need to send HTTP request to anycast prefixes to obtain client mappings. However, requests would be illegal if not setting the *Host* header, *i.e.,* requested domain. Therefore, we validate if there exists "binding" between anycast addresses and site domains, namely, if Cloudflare configures access control so that a customer site can only be accessed via its resolved IPs. Specifically, we get the correspondingly highest ranked site domain and one IP for each anycast prefixes, and then send "HTTP GET /" requests to each naked IPs with every domains as *Host* sequentially. The test shows that all requests are successful and replied with CF-Ray header. Thus, there is no "binding" and all customer sites are equivalent. We also test non-customer sites set as *Host*, which are forbidden by Cloudflare. To increase recall rate, we similarly test the remaining routable prefixes and finally get 149 anycast prefixes. Thus the other 399 prefixes are supposed to be unicasted, but need to be further validated and associated with a specific replica.

Unicast targets. Replicas use unicast addresses to fetch contents from *origin sites* and also insert *CF-Ray* header to identify itself. Therefore, we set up a Web server with domain name *example.me* as the *origin site* and host it on Cloudflare. We then use widely distributed HTTP proxies to request **non-existent** content under *example.me*, forcing replicas to contact the *origin site* under our control. Meanwhile, our Web server is configured to record the remote address, *i.e.,* unicast address, and *CF-Ray* header for each request, which provides us with unicast addresses and their associated replicas.

Since each replica has its own client catchment, vantage points need to be highly diversified to *enumerate* all replicas. This goal is consistent with the one to evaluate client proximity of all replicas, so we merge them in the same measurement. The details are given in next part of HTTP proxy measurement. Note that unicast addresses used to fetch contents cannot be controlled by a third party like us. As the 20-day measurement continues, the accumulation of new unicast addresses shows a diminishing return. We finally collect 7973 unicast IPs, covering all 96 replicas and 227 prefixes among the 399 remaining ones. Unexpectedly, all of their city locations from geolocation database Maxmind are the same with their associated IATA identifiers, which is a good cross-validation. 88% of replicas have less than 4 unicast prefixes and each replica has at least one (the average is 2.4), we therefore use these 227 prefixes as the deterministic representatives for each replica.

3.2 HTTP measurement for client-server mappings

To achieve task (1) and simultaneously obtain each replica's deterministic unicast addresses, we use a large-scale P2P-based HTTP proxy platform Luminati (advertised with O(1M) proxies) to perform HTTP request. Luminati allows users to select proxies at the level of proxy country, city or ASN instead of appointing proxies directly for security. Peer proxies also join and exit dynamically. We therefore scan CAIDA's complete ASN list [23] as HTTP sources to request non-existent contents under *example.me* for 160 rounds (~20 days during 2017/10). In total, Fig. 1 shows the scanned 899062 proxies, distributed at 127578 prefixes, 16620 ASes, 226 countries and 38028 cities. Averagely, each source AS has 147 measurements and covers 45.7 IPs, 27 /24 prefixes, 5.74 cities and 1.04 countries. As described in Section 2, the overall client mapping is very stable.

3.3 Traceroute measurement for routing path and latency

Using Atlas to perform extensive traceroutes faces challenges of strict capacity limitation (both concurrent and total quota), probe churns and expensive credit costs. We thus need a careful and economical design of traceroute measurement.

Sources. To combine client-mapping relations obtained from HTTP measurement, we select Atlas probes intersecting with HTTP proxy vantage points. Prior works [10, 18] align clients of different platforms at AS level while we align at finer prefix level. Because a few huge ASes, *e.g.,* transit providers, have widely distributed PoPs and region-specific routing policies, which routes different parts to different replicas. Accordingly, we select 1632 probes initially and at last 1140 produce complete results due to probe change or exit. Similar as prior studies using Atlas [9, 18], probe distribution in this paper are also Europe-centric. Nonetheless, other continents still have more vantage points (at least 40) than other platforms and it does not impact later analysis since we differ between continents.

Targets. To evaluate **anycast efficiency** ratio as described in Section 2, each probe's targets include anycast addresses and potential replicas indexed by unicast addresses. Basically, each probe measures all 149 anycast targets, which ideally should be identical but practically a small portion (7%) of probes observe less 10% inconsistent routes due to routing engineering, such as splitting address space into more specific prefixes [24].

Additionally, to obtain all potential paths, measuring latency to all replicas (n) from all probes (m) for efficiency estimation is unbearably expensive ($O(n * m)$) and invasive in this case, both numbers of source and target are an order of magnitude more than most early studies on DNS [6, 15]. Moreover, all prior studies only use *ping*, which costs an order of magnitude less than *traceroute*. A recent measurement on an anycast CDN [10] with similar scale (~60 replicas) shows that the potential minimal latency to the nearest N replicas shows significantly diminishing return as N increases. Therefore, each probe in our measurement also similarly measures the three geo-closest replicas (each with ~2.4 unicast targets) as potential candidates to reduce cost.

To simultaneously measure anycast targets and potential candidates at each probe while satisfying Atlas strict limits of concurrent and total quota of measurement jobs, we use 4 accounts and carefully compute job partitions on probe-target matrix of O(1632*(149+227)). To suppress outliers, we repeat all traceroute 10 times periodically. By amortizing cost caused by Atlas limits, the measurement takes about 50 days during 2017/10-2017/11.

At last, we map all IPs in the dataset to their geographic locations (city, country, continent, timezone, longitude and latitude) using Maxmind database. As many studies [9, 10] show, although no geolocation database has perfect accuracy, *e.g.,* IPs belong large organizations are often located at their headquarters, the overall accuracy is still satisfactory. Note that we *do not* use it for anycast IP geolocation. Additionally, if a Atlas probe is registered with its geographic location, we prefer it than Maxmind.

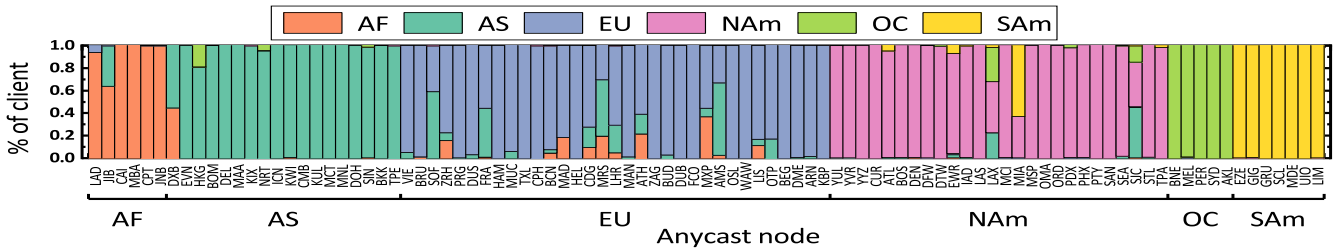


Figure 2: Client distribution by continent of 96 anycast nodes.

(a) LHR, London (b) LAX, Los Angeles (c) MIA, Miami (d) BOM, Bombay

Figure 3: Representative examples of client distribution (catchment) of 4 anycast nodes.

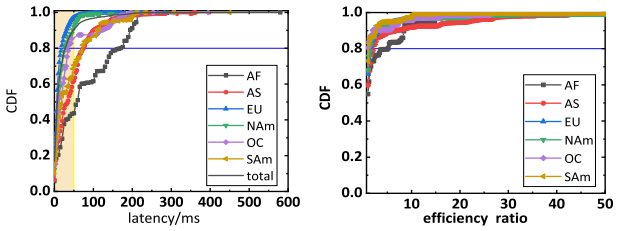
4 PROXIMITY AND ANYCAST EFFICIENCY

In this section, we first gain intuitive visibility into global proximity represented by catchments, then explore whether high latency is contributed by inefficient anycast routing. We follow up in the next section with a systematic analysis on anycast routing inefficiency.

4.1 Client mapping proximity

To characterize the comprehensive proximity lucidly, we visualize each anycast node's catchment similarly as CAIDA [8] based on widely and densely distributed HTTP proxies. Fig. 2 presents the distribution of clients for each anycast node by continents, indexed by Cloudflare's identifier code on the X-axis and arranged by continent. Note that geographic distance is indeed a good approximation of expected latency because Cloudflare [25] investigates various transit provider networks, such as their PoPs, to choose proper direct ISPs. They correspondingly engineer the routing so that clients could be mapped to geo-close replicas. The result shows good overall proximity, as can be obviously observed by consistent color clusters. In particular, about 81% of anycast replicas serve at least 85% of corresponding clients from the same continent the replicas are located in; the metric for country level is similar.

Meanwhile, there exist a few exceptions that the catchments are very scattered and suboptimal. We directly show 4 primary ones on the map in Fig. 3. Most of the scattered cases in Europe are like LHR's catchment, widely distributed across Europe, Asia and Africa. Especially, a mass of clients in India are routed to LHR while there exist local nodes, *e.g.,* DEL and BOM. As we will show later this is due to client network mistakenly chooses large transit provider AS10310 over local AS9498. Among replicas in North America (NAm), LAX and SJC exhibit very similar abnormal pattern, namely, their clients concentrate around west coast in NAm, east coast in Asia and Oceania. This is due to persistent switch, which is also observed in [16]. Node MIA, though seeming more reasonable, serves large quantity of clients in South America while local replicas such as LIM (Lima) and EZE (Buenos Aires) should serve those clients. We also give an example (Fig. 3(d)) with good proximity like most replicas, showing clear constrained geographic boundaries.



(a) (b)

Figure 4: Cumulative distribution of latency (a) and anycast efficiency ratio (b).

Table 1: Latency summary and inflation threshold T

	80pct/ms	90pct/ms	95pct/ms	% < 50ms	T/ms
Overall	27.4	50	72.5	90%	
AF	170	195	210	43.1%	75
AS	75	100	142.6	61.9%	75
EU	20	32.5	42.5	97.2%	50
NAm	27.2	47.4	57.5	92.1%	50
OC	35	120	150	85.4%	50
SAm	72.5	110	149.7	70.5%	75

4.2 Latency and anycast efficiency

After demonstrating the unsatisfactory proximity cases, we now explore how much high latency is contributed by anycast inefficiency, *i.e.,* clients are directed to a distant, high-latency replica over a close, low-latency one. Latency distribution alone cannot reflect how well anycast routing performs. For example, latencies in Africa are generally high, but this may be due to lack of infrastructure [26] rather than inefficient anycast choice, *i.e.,* anycast routing already selects the lowest-latency replicas from all potential ones. Hence in this section, by properly evaluating the distribution of latency and efficiency ratio, we choose thresholds between normal (0) and inflation (1, abnormally high) for all anycast paths. They are

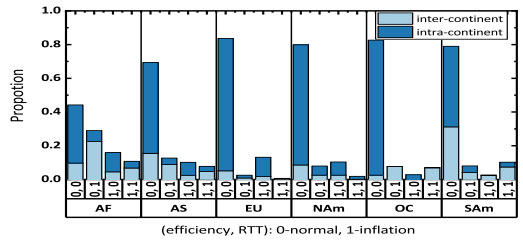


Figure 5: Fraction of paths classified by (ratio, latency).

hereafter classified into 4 categories at the dimensions **(efficiency, latency)** and we further analyze the two features and their relationship. Later analyses are across individual paths, finer-grained than probes, but metric distributions across them are nearly equal since paths from a given probe to all anycast prefixes are almost the same as described before.

Fig. 4 shows the cumulative distribution of anycast latency and anycast efficiency ratio. The overall latency appears very good with 80-percentile (pct) as 27.4ms, but also shows long tail with 97-pct as 110ms. Similarly as geographic proximity, different continents exhibit significantly different as summarized in Table 1. Specifically, Europe and North America have best performance, respectively 97.2% and 92.1% latencies are less than 50ms. Africa shows terrible performance with 80-pct as high as 170ms. Oceania has an obvious step that 80-pct is 35ms while 90-pct jumps to 120ms, probably due to geographic isolation. To diagnose abnormal latencies, we conservatively set latency threshold **T** for each continents as in Table 1 due to their significant difference.

For each path to anycast addresses, we refer to the ratio of its latency and lowest latency among potential replicas from the same probe as **anycast efficiency ratio** and their difference as **additional latency**. If anycast selects the best path, then the ratio should be equal to 1 (or almost due to jitter); otherwise, ratio> 1. The vast majority of efficiency ratios are optimal, 85% of which are nearly equal to 1. Theoretically, efficiency ratio greater than 1 means inflation to diagnose. But latencies are inevitably subject to jitter, thus need sensitivity analysis by observing normal latency distributions as the threshold increases. If latencies show an obvious up-rush at certain efficiency point, then it would be a good threshold; otherwise, we should set the threshold low to avoid false negative. We plot latencies for each continents setting efficiency threshold respectively at 1, 80-pct, 90-pct and none, the latency lines stay very close and we thus set the threshold as 80-pct. We leave out the figures due to limited space. The reasonability of ratio threshold is demonstrated in Table 2 (pink cells), *i.e.,* if the efficiency ratio is normal, the additional latency is always very low no matter the total latency is normal or inflated.

We now analyze the impact of anycast on latency performance. Fig. 5 intuitively plots the fraction of the four categories with a further geographic hint of whether paths stretch across continents. Clearly, the majority paths are performing well with normal latency and efficiency. These paths are almost constrained in one continent except that a small fraction in north SAm are mapped to MIA, whose latencies are still good.

Importantly, the terrible latencies in AF are not caused by inefficient anycast routing since the fraction of (normal ratio, high latency) is 3 times more than that of (high ratio, high latency). In

Table 2: Fraction and statistics of paths at (ratio, latency)

	(ratio, latency)	ratio med.	addi. med.	latency med.	addi./ latency	path fraction
AF	0, 0	1.06	0.9	18.6	0.05	44.20%
	0, 1	1.04	4.3	146.1	0.03	29%
	1, 0	8.53	59.4	68.7	**0.87**	16.03%
	1, 1	3.7	158.9	195.8	**0.81**	10.77%
AS	0, 0	1.1	0.5	20.8	0.02	69.37%
	0, 1	1.3	19.6	95.8	0.2	12.69%
	1, 0	4.6	29	40.5	**0.72**	10.18%
	1, 1	17.3	109.5	124.2	**0.88**	7.76%
EU	0, 0	1.05	0.3	7	0.04	83.63%
	0, 1	1.11	7.4	65.6	0.11	2.53%
	1, 0	4.63	22.9	34.3	**0.67**	13.19%
	1, 1	4.38	45.9	58.2	**0.79**	0.65%
NAm	0, 0	1.06	0.42	11.9	0.04	79.85%
	0, 1	1.05	2.65	61.3	0.04	7.96%
	1, 0	6.5	10.37	13.6	**0.76**	10.35%
	1, 1	7.37	49.69	66.8	**0.74**	1.85%
OC	0, 0	1.02	0.2	11	0.02	82.62%
	0, 1	1.04	4.5	116	0.04	7.66%
	1, 0	6.06	28.5	34.2	**0.83**	2.77%
	1, 1	10.79	147.2	169.9	**0.87**	6.95%
SAm	0, 0	1.06	0.6	13.9	0.04	79.02%
	0, 1	1.01	1.3	112	0.01	8.06%
	1, 0	2.77	30.8	48.1	**0.64**	2.69%
	1, 1	3.72	128.3	152.4	**0.84**	10.22%

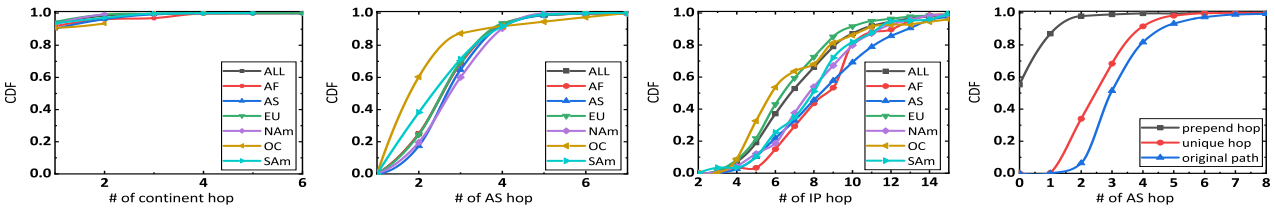
med.: median; addi.: additional latency

fact, among high latencies across all continents, the fractions of normal ratio are always greater than that of inflated ratio, except for a little difference in SAm. However, when anycast ratio is inflated, no matter the total latency is high or normal, the additional latencies account for about 80% of total latencies as shown in Table 2 (yellow cells). In summary, although anycast inefficiency only accounts for a small fraction of inflated paths, it makes latencies exhibit a 5-fold increase. Therefore, it is necessary to diagnose inefficient anycast routing for performance improvement.

5 INEFFICIENT ANYCAST ROUTING DIAGNOSIS

This section presents a systematic diagnosis on the behavior and causes of inefficient routing for the anycast CDN. Generally, the first step of analyzing inefficient routing is to roughly classify and formulate routing pathologies based on observed characteristics of measured routes. Early seminal work [27] comprehensively analyzes routes from the view of **topology** (lack of good available paths) and **policy** (choosing poor paths).

- On topology, this kind of routing detour is usually caused by lack of proper infrastructure, which corresponds to unicast inflation (relative to straight-line distance traveled by light) . Some recent works [24, 28] study routes based on locations of ingress points and peering points in various scenarios, *e.g.,* cellular network.
- On policy, wide-area network routing policy is notoriously challenging to analyze since it is nearly artificial and proprietary for most ISPs. Routing behavior analysis is often based on classical routing model theories, such as "Prefer Shortest AS Path" and "Valley-Free" (prefer customers/peers



(a) continent (b) AS (c) IP (d) control-plane AS path

Figure 6: Length distribution of anycast paths aggregated at different levels.

over providers) models, which usually lacks awareness of practical engineering.

In our specific scenario, we focus on anycast problems where the anycast-chosen path is worse than the **benchmark path** to the best potential replica (ratio=1). Therefore, unicast inflations are not of our concern. We first explore unique characteristics of anycast paths in our extensive measurements. Based on the observations, we propose an inefficient anycast routing pathology of two complementary routing patterns. We then describe our methodology of identification and verification. Diagnosed results are given at last.

We begin by characterizing anycast paths in data plane. Fig. 6 (a-c) examines the lengths of anycast paths aggregated at levels of continent, AS and IP. We map each IP hops in all traceroutes to their belonging ASes and geographic locations, and then collapse consecutively same hops. Note that although anycast IPs in the last hops are geolocated to null, it does not affect the geographic path since the penultimate hops are often very close the destinations [7], thus completing geographic information. Besides, we leave out 7% raw traceroutes whose penultimate hops do not respond.

Apparently, nearly 93% of anycast paths are constrained within one continent. An extremely few portion (2%) of paths are across more than 3 continents, highly likely due to imperfect geolocation database. Importantly, almost 90% of AS paths are very short for nowadays tangled Internet, with only 2-4 AS hops, which is consistent with a study in 2015 [29] measuring the length of paths to popular Web services. This feature indicates that Cloudflare, as one of the largest global CDNs, invest hugely in expanding its infrastructure and widely connecting with eyeball networks, where end-users reside. Apart from source ASes and the destination AS, Cloudflare's AS13335, the intermediate ASes in anycast paths only account for 1-2 hops or even 0 if end-ASes are connected directly. This highlights the importance of *direct ISPs*. Cloudflare states that they prefer two kinds of direct ISPs [25]: transit network providers and domestic/local network providers, both likely owning multiple PoPs, especially for the former.

5.1 Pathologies of inefficient anycast routing

Based on the above characteristics that ~90% paths traverse as few as 2-4 ASes and the direct ISPs may have several PoPs, we orthogonally categorize inefficient anycast paths as below:

- If an AS path is different with the benchmark AS path, it has **Inter-domain inefficiency**.
- If an inefficient anycast AS path is the same with the benchmark path, direct ISPs with multiple PoPs, especially large ones like Tier-1 providers, could still select the distant "wrong" PoP with higher latency. We note this as **Intra-direct ISP inefficiency**.

Fig. 8 exemplifies them, derived from real cases in our dataset. It is worth noting that the second pattern is subtle since traceroutes can only reflect AS path while the inflation may happen at PoPs level inside direct ISPs. Ideally, direct ISPs should obey "early-exit" policy within its own network, but may practically misbehave due to routing misconfiguration.

Technically, there exist hybrid cases where both inefficiencies happen, *i.e.,* selecting an inefficient AS path as well as a distant PoP. We consider this as *Inter-domain inefficiency* because if the direct ISP does not see the "right" ingress point, it should not be blamed for exiting to distant PoPs.

5.2 Identification of inefficient anycast routing patterns

Theoretically, for an inflated path, we could determine its pattern by directly comparing it with the benchmark path. But we still need to rigorously verify if **a direct ISP** route **a given client** to **a distant PoP** among all its **potential ones** and analyze characteristics of inflated paths. To this end, for each inflated path we have the following process:

(1) Identify PoPs of the direct ISP, *i.e.,* the potential nodes list (PTList) that the direct ISP could choose from, detailed identification method is in the below paragraph.
(2) Identify the actual replica that anycast chose for the client. We mainly use the client mappings obtained in Sec. 3.2.
(3) Rank all nodes in PTList for the clients based on geographic distance. As explained below, PoPs are represented as the closest replicas, we therefore use replicas' rank from 0 to 95 for normalization.
(4) Compare the rank of actually selected node with PTList to determine whether clients are routed to distant PoPs.

Identifying PoPs of direct ISPs We first identify direct ISPs of Cloudflare, then further find and verify the PoPs of direct ISPs. Specifically, our extensive data-plane traceroutes detect 462 ASNs of direct ISPs. To verify if enough direct ISPs are identified, we cross-validate with control-plane monitoring system Routeveiws/RIPE NCC, which only has vantage points in the core Internet. Consequently, it observes 302 neighbor ASes of AS13335 and we thus detect 17% more, which covers enough direct ISPs.

After that, we identify PoPs of direct ISPs based on that Cloudflare normally connects its replicas to the most close PoPs of direct

Table 3: Example details of the first 5 direct ISPs out of 462 (Cum.: cumulative)

Tier-1 provider name	ASN	CF Rank	CAIDA Rank	Coverd continents	Coverd nodes	# conti-nents	# count-ries	# nodes	Cum.# conti-nents	Cum.# count-ries	Cum.# nodes
Telia	1299	1	3	NAm;EU	VIE;BRU;DTW;LAX;SOF;DEN;DUB;BOS;DFW;IAD;PRG;MIA;TPA;BUD;AMS;CPH; MCI;OMA;TXL;CDG;ZRH;BCN;BNA;MAD;HAM;OSL;DUS;PDX;YUL;KBP;DME; HEL;MXP;MRS;ARN;ATL;FRA;MSP;STL;LHR;WAW	2	22	41	2	22	41
Cogent	174	2	2	EU;NAm; AF	VIE;YYZ;LIS;SJC;IAD;DEN;DFW;BOS;PRG;TPA;CAI;BUD;AMS;MCI;TXL;CDG; ZRH;BCN;BNA;MAD;OSL;DUS;YUL;KBP;FCO;MUC;HEL;LAS;ARN;ATL; EWR;PHX;LHR;YVR;BEG;PHL;ATH;MAN	3	21	38	3	27	55
GTT	3257	3	5	NAm;EU	OTP;VIE;BRU;YYZ;DTW;LAX;SOF;DEN;DUB;BOS;IAD;PRG;ORD;AMS;CPH;TXL; ZRH;BCN;OSL;DUS;YUL;FCO;MUC;MXP;MRS;EWR;MSP;SAN;LHR;YVR;MAN;WAW	2	18	32	3	28	58
Yahoo	10310	4	626	AS;NAm; SAm;EU	GRU;YYZ;LAX;SJC;IAD;DEN;DUB;BOS;DFW;PRG;MIA;ORD;HKG;SEA; AMS;CDG;NRT;PDX;ARN;KUL;ATL; FRA;TPE;EWR;MSP;PHX;SIN;LHR;YVR	4	15	29	5	34	65
Tata	6453	5	7	AS;NAm; EU	VIE;BRU;KIX;YYZ;LIS;LAX;IAD; DFW;MIA;ORD;HKG;BUD;CDG;ZRH; BCN;MAD;OSL;NRT;YUL;MXP;MRS;KUL;ATL;FRA;TPE;EWR;LHR;WAW	3	18	28	5	34	66

ISPs [25]. Therefore, *if there exists a path where a direct ISP X connects to a replica Y, then X covers a PoP/replica in the location (city) of Y, the PoP is denoted as Y.*

Validation We cross-validate with CAIDA IXP (Internet eXchange Point) dataset, which includes two kinds of mapping relations:

- *IX-ASN* contains many-to-many mapping between IXPs and ASes. An IXP provides a convenient rendezvous for many ASes to exchange routes; an AS can learn routes at many IXPs.
- *IX-facility* provides the facilities, *i.e.,* PoPs, of IXPs. To provide service conveniently, IXPs usually present themselves at many facilities, which are publicly listed by database like PeeringDB.

Together, the dataset has 857 IXPs and 1270 facilities with geographic details, *e.g.,* city. Although subject to incomplete statistics of Internet, it is by far the most reliable dataset for validation.

We query the IXPs that direct ISPs choose to join via *IX-ASN* and further query the facilities (city) of the obtained IXPs, which gives the potential PoPs of direct ISPs. We therefore cross-validate traceroute-inferred PoPs with the obtained facilities at city level. As a result, 25 direct ISPs do not show up in CAIDA dataset and 403 ones are consistent, which validates our method. Table 3 give an example of the first 5 large ISPs.

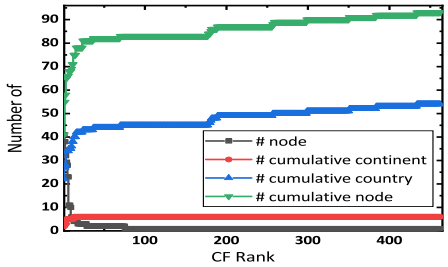
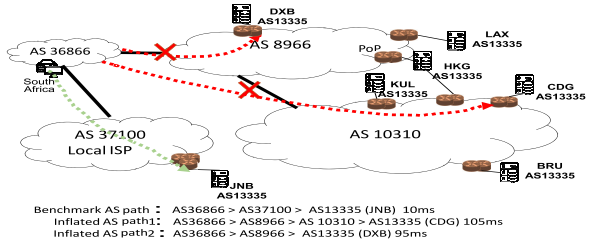


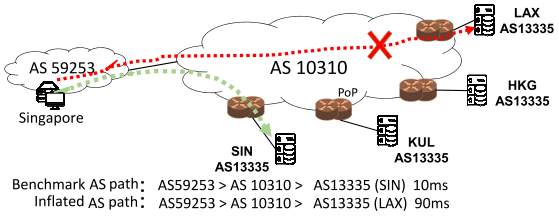
Figure 7: Coverage scale of direct ISPs represented as nodes, cumulative unique nodes, countries and continents against their CF Rank.

5.3 Diagnosis result

Based on the proposed pathologies and measured dataset, we now quantify the characteristics of inefficient anycast routing. We first demonstrate why classical routing models are not appropriate for



(a) Inter-domain inefficiency



(b) Intra-direct ISP inefficiency

Figure 8: Inefficient routing patterns for anycast CDN

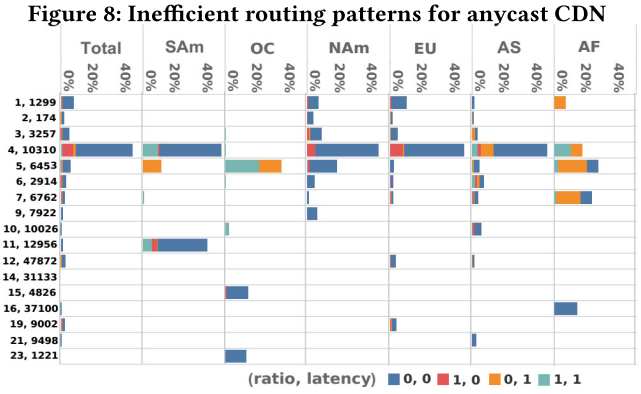


Figure 9: Fraction of paths carried by each direct ISP, Y-axis: (CF Rank, ASN) of direct ISPs

this scenarios. We then propose to use quantitative metrics of direct ISPs to analyze inefficient anycast routing.

First, it is hard to tell whether anycast falsely chooses inflated paths due to follow "prefer shortest AS path", since AS paths obtained by traceroute are data-plane representation while routing

decision is based on control-plane status, *i.e.,* AS paths exchanged in BGP sessions. The latter often has routing engineering such as AS prepending, especially for CDNs. This leads to that data-plane status cannot fully restore control-plane status, further causing false inference. For instance, suppose vantage point VP observes two paths by traceroute: Path1 is (A,B,C) and Path2 is (A,D,E,C), and VP chooses inflated Path1. If we make inference based on the traceroute information, we may owe the fault to "prefer shortest AS path" due to that Path1 is short than Path2. However, the corresponding control-plane in fact may be: (A,B,B,B,C) for Path1 and (A,D,E,C) for Path2, the routing decision is not based on "prefer shortest AS path", *i.e.,* the inference is wrong.

We collect control-plane AS paths of AS13335 using all available monitors from Routeveiws/RIPE NCC and find there does exist large amounts of AS prependings. As shown in Fig. 6(d), the length of unique hops are nearly consistent with traceroute measurements, mainly between 2-4 hops. However, about half of control-plane paths have at least 1-hop AS prepending, which makes traceroute inappropriate to analyze "prefer shorter path". Additionally, there are only about 300 control-plane vantage points, an order of magnitude less than current data-plane probes, and they are also hard to be aligned to leverage the control-plane status.

Second, "prefer customers/peers" is also not the main reason for anycast false choose. Due to the short lengths, we can directly compare the inflated anycast paths with benchmark paths and analyze the diverging points. For instance, Path1 is (A,B,C) and Path2 is (A,B,E,C), and VP chooses inflated Path1. Then B is denoted as the diverging point, C and E are the downstream AS of B. "prefer customers/peers" in this case means C is B's customer/peer while E is B's provider. However, we find that for all inefficient anycast paths in the measured data, downstream AS C and E are both B's provider, *i.e.,* their customers-cone ranks [30] are higher.

Therefore, we group inefficient paths by direct ISPs, analyze their characteristics and identify primary ones that contribute more than 50% in the given group (noted as culprit AS). We consider the direct ISPs are very important. First, for intra-direct ISP inefficiency, the culprit AS is the direct ISP itself; for inter-domain inefficiency, we observe the raw data appears to show some common behaviors: although direct ISPs may not directly cause the fault, *e.g.,* the diverging ASes choose local direct ISPs over large transit direct ISPs, the overall path length are around 2-4 ASes, resulting in that the downstream ASes of the diverging ASes usually are the direct ISPs. We thus still extract direct ISPs as relevant culprits. Besides, it is convenient for CDN administrators to troubleshoot with an unified metric. Since routing policy is artificial and proprietary for most ISPs, it does not make sense and is intractable to exhaust all encountered problems case by case.

Quantitative metrics of direct ISPs. By quantitatively representing direct ISPs at a finer granularity than AS, *i.e.,* PoP, we find that the scale of direct ISPs are extremely imbalanced. We use **CF RANK** to represent direct ISPs, *i.e.,* sorts direct ISPs descendingly by the number of nodes they cover and break the tie using the ranks of customer-cone size [30]. Fig. 7 shows the number of covered nodes and cumulative nodes of direct ISP indexed by CF Ranks. Obviously, the first 15 (3%) direct ISPs, mostly huge Tier-1 providers,

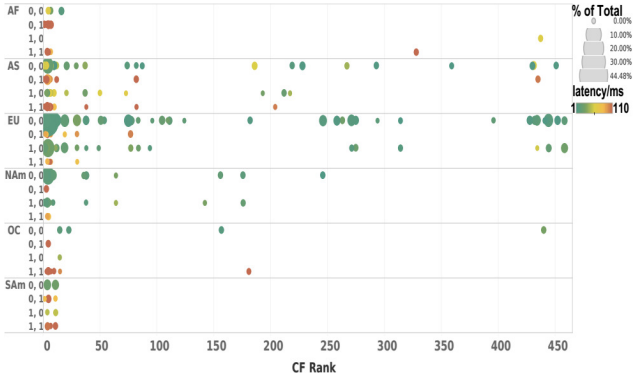


Figure 10: Under each (efficiency, latency) category, fraction of paths traversing each direct ISP. Circle size: fration; Color: median latency; X-axis: CF Rank.

cumulatively covers 75 nodes out of 96 and 387 only covers one node. Table 3 gives detailed information.

Further, Fig. 9 shows the fraction of paths each direct ISP carries and X-axis is (CF Rank, ASN) of direct ISPs. It quantitatively reveals Cloudflare's policy [25] in choosing direct ISPs. The first 3% large direct ISPs, *e.g.,* transit providers, are used to delivery the majority of contents and local providers assist to expand their reach.

Characteristics of inflated anycast paths. Table 4 summaries the fraction of each pattern against the continent for (inefficient ratio, high latency) paths and intuitively present corresponding cases. Inter-domain inflations generally account more than purely intra-direct ISP inflations for high latencies, as represented by their corresponding percentages. Moreover, as indicated by the typical cases, inter-domain inflations often manifest as upstream small ASes preferring large transit ISPs with high CF Ranks over local direct ISPs with low CF Ranks. Besides, 88.7% inefficient paths traverse 1-2 more AS hops than benchmark ones. This can also be statistically demonstrated in Fig. 10, which plots the fraction of paths traversing each direct ISPs under each category. Obviously, inefficient paths with high latencies mainly concentrate around high-ranked providers, especially the first 3% ones; while low-ranked local providers only show up in *normal-latency* paths, especially for AS and EU. Further, the primary culprits for inter-domain inflations are mainly a few Tier-1 providers, *i.e.,* AS6453 (CF Rank:5) for NAm and OC, and AS10310 (CF Rank:4) for others. We consider it is due to traffic engineering as pointed out by many studies [24], which are related to less/more specific prefix route leakages.

We briefly discuss typical cases ignored by Table 4 due to space limitation. We find that the intra-direct ISP inflations usually cause very low additional latencies. Although the percentages of (high inflations, normal latency) for OC (2.77%) and NAm (2.69%) are very low, they are both primarily caused by local direct ISPs, *i.e.,* AS4826 (CF Rank:15, Australia company) for OC and AS 12956 (CF Rank:11, Telefonica) for SAm. These primary culprits have potentially outsized impact and deserve priority attention.

6 CONCLUSION

Combining two large-scale active measurements, we take a first step into evaluating the performance and diagnosing causes and characteristics of inefficient anycast deployment for a large-scale

Table 4: Fraction of each inefficiency, primary culprits and corresponding cases

		Intra-direct ISP	Inter-domain
AF	%	2.78%	6.58% (AS10310, 3.26%; AS8966, 2.38%)
	case		prb22215, Nairobi, KE 36866>37100>13335 11.2ms MBA;JNB;CPT;LHR 0;3;9;41 MBA-0 36866>8966>13335 59.3ms DXB DXB-6 36866>8966>10310>13335 200.7ms PRG;FRA;CDG... 28;31;35;39... SIN-49 KE>AE>SG
AS	%	1.90% (AS10310, 0.75%)	4.41% (AS10310, 2.55%)
	case	prb13821, Tokyo, JP 2518>10310>13335 7.3ms NRT;TPE;HKG... 0;3;4... NRT-0 2518>10310>13335 97.3ms NRT;TPE;HKG... 0;3;4... LAX-33	prb34024, Delhi, IN 17747>9498>13335 16.6ms DEL;BOM;MAA 0;1;2 DEL-0 17747>9498>10310>13335 83.4ms HKG;KUL;SIN... 10;11;13... SIN-13
EU	%	0.18%	0.30% (AS10310, 0.08%)
	case		prb4352, Stupino, RU 23242>198297>28917>13335 4.5ms DME 0 DME-0 23242>200130>10310>13335 52.1ms ARN;PRG;FRA... 4;10;17... ARN-4
NA	%	0.48%	0.97% (AS6453, 0.52%)
	case		prb21031, Billings, US 33588>10310>13335 30ms DEN;SEA;PDX... 0;1;2;4... SEA-1 33588 >174>6453>13335 77.3ms LAX;ORD;DFW... 10;13;14... DFW-14
OC	%	1.17%	4.04% (AS6453, 3.89%)
	case		prb20825, Prahran, OC 1221>13335 7.4ms MEL;SYD;BNE 0;1;2 MEL-0 1221>4637>6453>13335 193.1ms KUL;TPE;HKG... 7;9;10... LAX-29 AU>HK>US
SA	%	6.02% (AS10310, 5.9%)	1.14%
	case	prb30101, Caxisa do Sul, BR 2716>1916>10310>13335 21.4ms GRU;MIA;ATL... 1;9;14... GRU-1 2716>1916>10310>13335 177.1ms GRU;MIA;ATL... 1;9;14... MIA-9	

Format: 1. %: % to the corresponding continents, culprit direct ISP in brackets. 2. Case: probe ID, city, country (the 1st path is benchmark and others are inflated) AS path; latency/ms; direct ISP's PoPs sorted by distance for the probe; corresponding PoP ranks among 96 nodes for the probe; actually selected node-rank

CDN. Our results reveals some critical features. Specifically, a substantial number of access paths are very short, traversing only 2-4 ASes. Moreover, among the important intermediate ASes, Cloudflare unevenly uses large transit providers to delivery the majority of contents. Based on these observations and raw traceroute data, we classify inefficient anycast routing into two patterns, identify possible causes and quantify their impact. We find that Tier-1 ISPs have outsized impact in that they could not only have path inflation inside their own networks, but also are related to majority of inter-domain inflation. Hence, we believe it is important to optimize routing and peering arrangement with large Tier-1 ISPs.

REFERENCES

[1] Li Jin H Cheng, Angela. Measuring and evaluating large-scale CDNs. In *Proceedings of the IMC 2008, Vouliagmeni, Greece, October*, 2008.

[2] Ao-Jan Su, David Choffnes, Aleksandar Kuzmanovic, and Bustamante. Drafting behind akamai. In *ACM CCR*, volume 36, pages 435–446, 2006.

[3] Zhuoqing Morley Mao, Charles Cranor, Fred Douglis, and Michael Rabinovich. A precise and efficient evaluation of the proximity between web clients and their local dns servers. In *Proc. of USENIX ATC*. USENIX, 2002.

[4] RFC1546. tools.ietf.org/html/rfc1546.

[5] Giovane C. M. Moura, Ricardo de Oliveira Schmidt, and John S. Heidemann. Anycast vs. DDoS: Evaluating the november 2015 root DNS event. In *Proceedings of the IMC 2016, Santa Monica, CA, USA, Nov.*, 2016.

[6] Lorenzo Colitti. Evaluating the effect of anycast on dns root. In *RIPE-393*, 2006.

[7] Xun Fan, John S. Heidemann, and Ramesh Govindan. Evaluating anycast in the domain name system. In *Proceedings of the IEEE INFOCOM 2013, Turin, Italy, April 14-19, 2013*, pages 1681–1689. IEEE, 2013.

[8] Ziqian Liu, Bradley Huffaker, Marina Fomenkov, Nevil Brownlee, and Kimberly C. Claffy. Two days in the life of the DNS anycast root servers. In *Proceedings of the PAM 2007, Louvain-la-neuve, Belgium, April*.

[9] Ricardo de Oliveira Schmidt, John S. Heidemann, and Jan Harm Kuipers. Anycast latency: How many sites are enough? In *Proceedings of the PAM 2017, Sydney, NSW, Australia, March*.

[10] Matt Calder, Ashley Flavel, Ethan Katz-Bassett, Ratul Mahajan, and Jitendra Padhye. Analyzing the performance of an anycast CDN. In *Proceedings of the IMC 2015, Tokyo, Japan, October*, 2015.

[11] LinkedIn anycast. engineering.linkedin.com/network-performance/tcp-over-ip-anycast-pipe-dream-or-reality.

[12] Doug Madory and Chris Cook. Who are the anycasters? In *NANOG59*, 2014.

[13] Danilo Cicalese Drossi. where are anycasters. In *RIPE71 report*, 2014.

[14] Danilo Cicalese, Jordan Augé, Diana Joumblatt, Timur Friedman, and Dario Rossi. Characterizing ipv4 anycast adoption and deployment. In *Proceedings of the 11th ACM CoNEXT, Heidelberg, Germany, December 1-4*, pages 16:1–16:13. ACM, 2015.

[15] Hitesh Ballani, Paul Francis, and Sylvia Ratnasamy. A measurement-based deployment proposal for IP anycast. In *Proceedings of the IMC 2006, Rio de Janeriro, Brazil, October 25-27, 2006*, pages 231–244. ACM, 2006.

[16] Lan Wei and John S. Heidemann. Does anycast hang up on you? In *Network Traffic Measurement and Analysis Conference, TMA 2017, Dublin, Ireland, June 21-23, 2017*, pages 1–9. IEEE, 2017.

[17] Sandeep Sarat, Vasileios Pappas, and Andreas Terzis. On the use of anycast in DNS. In *Proceedings of the IEEE ICCCN 2006, October, Virginia, USA*, 2006.

[18] Wouter B. de Vries, Ricardo de Oliveira Schmidt, Wes Hardaker, John S. Heidemann, and Pieter-Tjerk de Boer. Broad and load-aware anycast mapping with verfploeter. In *Proceedings of the IMC 2017, London, UK, Nov.*

[19] M. Levine et al. Operation experience with TCP anycast. In *NANOG 37*, 2006.

[20] Cloudflare IP range. www.cloudflare.com/ips/.

[21] Cloudflare anycast. support.cloudflare.com/hc/en-us/articles/203491930.

[22] Cloudflare HTTP headers. support.cloudflare.com/hc/en-us/articles/200170986-How-does-Cloudflare-handle-HTTP-Request-headers-.

[23] CAIDA, AS to organization mapping. www.caida.org/research/topology/as2org.

[24] Rupa Krishnan, Harsha V Madhyastha, Sridhar Srinivasan, Sushant Jain, and Arvind Krishnamurthy. Moving beyond end-to-end path information to optimize cdn performance. In *Proceedings of the IMC 2009, Chicago, Illinois, Nov.*

[25] Cloudflare. Routing for an anycast cdn. www.menog.org/presentations/menog-14/286-MENOG14-Routing-for-an-Anycast-CDN-Martin-Levy-CloudFlare.pdf.

[26] Yasir Zaki, Jay Chen, Thomas Pötsch, Talal Ahmad, and Lakshminarayanan Subramanian. Dissecting web latency in Ghana. In *Proc. of IMC*, 2014.

[27] Neil T. Spring, Ratul Mahajan, and Thomas E. Anderson. The causes of path inflation. In *Proceedings of SIGCOMM 2003, Aug. 25-29, Karlsruhe, Germany*.

[28] Kyriakos Zarifis, Tobias Flach, Srikanth Nori, David Choffnes, Ramesh Govindan, and Ethan Katz-Bassett. Diagnosing path inflation of mobile client traffic. In *Proceedings of the PAM 2014*, pages 23–33.

[29] Yi-Ching Chiu, Brandon Schlinker, Abhishek Balaji Radhakrishnan, Ethan Katz-Bassett, and Ramesh Govindan. Are we one hop away from a better internet? In *Proceedings of the IMC 2015, Tokyo, Japan, Oct.*

[30] Matthew J. Luckie, Bradley Huffaker, Amogh Dhamdhere, and Vasileios Giotsas. AS relationships, customer cones, and validation. In *Proceedings of the IMC 2013, Barcelona, Spain, October.*