DOI:10.1145/3604620

To view the accompanying Technical Perspective, visit doi.acm.org/10.1145/3604621

A Manifold View of Connectivity in the Private Backbone Networks of Hyperscalers

By Loqman Salamatian, Scott Anderson, Joshua Mathews, Paul Barford, Walter Willinger, and Mark Crovella

Abstract

As hyperscalers such as Google, Microsoft, and Amazon play an increasingly important role in today's Internet, they are also capable of manipulating probe packets that traverse their privately owned and operated backbones. As a result, standard traceroute-based measurement techniques are no longer a reliable means for assessing network connectivity in these global-scale cloud provider infrastructures. In response to these developments, we present a new empirical approach for elucidating connectivity in these private backbone networks. Our approach relies on using only "lightweight" (i.e., simple, easily interpretable, and readily available) measurements, but requires applying "heavyweight" mathematical techniques for analyzing these measurements. In particular, we describe a new method that uses network latency measurements and relies on concepts from Riemannian geometry (i.e., Ricci curvature) to assess the characteristics of the connectivity fabric of a given network infrastructure. We complement this method with a visualization tool that generates a novel manifold view of a network's delay space. We demonstrate our approach by utilizing latency measurements from available vantage points and virtual machines running in datacenters of three large cloud providers to study different aspects of connectivity in their private backbones and show how our generated manifold views enable us to expose and visualize critical aspects of this connectivity.

1. INTRODUCTION

A salient feature of today's Internet is that large cloud and content providers, called "hyperscalers," are building and operating their own private global-scale infrastructures (e.g., Google,[11] Amazon,[4] Microsoft,[9] Facebook,[12] Akamai,[13] and Alibaba[1]). These private network infrastructures serve to minimize the exposure of the traffic generated by their own suite of applications and services to the types of uncertainty and variability that their use of the "public" Internet (i.e., transit provider Internet) would entail. Accordingly, increasing portions of the overall Internet traffic utilize these large providers' private network infrastructures and thus bypass the public Internet.[24]

The growth and importance of these private infrastructures raise new questions, many of which call for empirical study. Historically, the principal tool used by researchers to understand Internet infrastructure has been `traceroute`. Developed in the late 1980s as a troubleshooting tool for network operators,[10] the Internet measurement community has enthusiastically adopted and improved it to study the properties of the routes taken by packets and to infer connectivity in the underlying physical infrastructure as described, for example, in Spring et al.[23]

While many of today's Internet stakeholders continue to support and use `traceroute` for purposes such as debugging, the emergence of hyperscalers has made the generic application of `traceroute` for network measurement and analysis increasingly problematic. As documented in online resources provided by, for example, Microsoft[16] and confirmed by our own measurements, some of today's hyperscalers can and do modify `traceroute` packets that traverse their private backbones or even disallow those packets altogether. When considered in conjunction with the increasing network-wide deployment of devices (i.e., middleboxes) that tamper with `traceroute`,[7] this development suggests that the utility of `traceroute` as a widely available, easy-to-use, and reliable general-purpose technique for (large-scale) Internet measurement experiments may be reaching a point of diminishing returns.

In this article, we posit that these trends are inevitable, and will continue. Accordingly, we ask the following motivating question: *How can we obtain useful insight about network structure when the basic mechanisms relied on by tools such as* `traceroute` *are either unavailable or can no longer be taken for granted?*

In considering how to answer this question, we argue that the increasing opacity of path internals in today's Internet calls for leveraging measurement techniques that are both more lightweight and more universally available than `traceroute`, dovetailed with more heavyweight mathematical analysis tools for extracting as much information as possible from these more limited information sources. To this end, we consider the *end-to-end round trip delay* (RTT) along an Internet path as an information source that is suitably

The original version of this paper was entitled "Curvature-based Analysis of Network Connectivity in Private Backbone Infrastructures," and was published in the *Proceedings of ACM Meas. Anal. Comput. Syst. 6*, 1 (Mar. 2022), Article 5.

lightweight and at the same time essentially universally available. Moreover, RTT can in most cases of interest also be readily augmented with metadata in the form of approximate geolocation of path endpoints. At the same time, our analysis of the resulting measurements is informed by mathematical concepts from Riemannian geometry (e.g., Ricci curvature and its extension to discrete graphs known as Ollivier-Ricci curvature[19]) and presented in the form of specialized visualizations; that is, continuous manifolds with embedded geo-information. This arguably heavyweight analysis tool only assumes that the "footprint" of a given network's physical infrastructure is specified in terms of a set of geo-located nodes that represent vantage points capable of performing the lightweight measurements of interest.

Thus, as an answer to our motivating question, we show that in a "post-`traceroute`" world, it is possible to rely solely on RTT measurements among a set of geo-located Internet nodes to elucidate important aspects of network structure. In the process, we present three key contributions:

(1) We describe a new methodology for assessing connectivity in private backbone infrastructures. Our approach offers third-party Internet researchers a promising alternative to inferring important aspects of network connectivity in these private infrastructures in a post-`traceroute` world.

(2) We assemble a set of newly-developed and existing techniques into a coherent and original methodology for illuminating important aspects of network structure—aspects that are either impossible or more difficult for `traceroute` to identify and discover as its ability to discern path internals in the Internet diminishes. This set of techniques includes a new specialized visualization tool that fuses Riemannian geometry with geographical maps to generate manifold views of a network's delay space.

(3) We demonstrate our methodology by inferring known and novel aspects of existing connectivity fabrics of the private backbones of three hyperscalers (Google, Microsoft, and Amazon). By comparing their inferred connectivity fabrics, we can highlight common features and important differences, identify "weak spots" in their existing connectivity, and hint at new capabilities for examining what future infrastructure changes would be most beneficial (or detrimental) for these large cloud providers.

In short, our proposed approach presents an exciting alternative to traditional methods for Internet connectivity research and enables third-party Internet measurement researchers to gain insight into the private backbone infrastructures of the large cloud/content providers. Having such insight is not only critical for independently verifying claims by these providers about the characteristics and performance of their network, but is also of practical importance for an industry that is expected to spend up to $150 billion on new fiber deployments in the U.S. alone.[21]

2. BACKGROUND

Since the early 1990s, the preferred method for studying Internet infrastructure has been to infer router-level topologies based on information gleaned from sufficiently many `traceroute` measurements. The key idea behind `traceroute` is to elicit self-identifying responses from *intermediate* routers along an end-to-end path. While many ingenious strategies have been developed for making this idea maximally informative[3] and efficient,[5] much can go wrong when performing this inference task in practice (e.g., see Motamedi *et al.*[17] and references therein).

2.1. The many problems with `traceroute`

Given that `traceroute` was originally designed as a debugging tool for network operators, assessing the quality of `traceroute`-based measurements when utilizing this data for rigorous scientific studies has been problematic for a number of reasons, including routers emitting their response from a randomly chosen interface or not responding at all to correctly-formatted probe packets, ambiguities in mapping interfaces to routers, and biases in the collected data due to the choice of vantage points for launching `traceroute` probes. Moreover, the increasing complexity of the Internet has made `traceroute` less effective. For example, the proliferation of underlying layer-2 technologies in today's Internet and `traceroute`'s inability to cope with them have reduced `traceroute`'s overall utility as robust Internet measurement tool. Also, the fact that the Internet has experienced a significant increase in the number of deployed middleboxes (e.g., firewalls, network address translation (NAT) boxes and proxies, and deep packet inspection (DPI) boxes) that possess the ability to drop packets carrying probe information renders `traceroute` as a means for exploring a network's infrastructure largely useless.

At the same time, the global Internet ecosystem itself has been changing in ways that question the use of `traceroute` as a reliable Internet measurement technique. In particular, the well-documented flattening of the Internet is the result of large multinational technology companies (i.e., "hyperscalers") building out private global-scale infrastructures that often carry traffic from where it is generated all the way to where it is consumed. However, these increasingly dominant infrastructures are opaque to `traceroute`. Global-scale cloud providers that serve as hosts for third parties are equally problematic. To be able to perform relevant measurements, third-party researchers have to become customers of such a provider, by purchasing resources in the form of virtual machines (VMs) and must adhere to the terms and conditions imposed by the provider that often prevent them from running tools such `traceroute` at will. And even if performing experiments such as running massive `traceroute` campaigns is technically permitted, it is completely up to each provider to determine how `traceroute` probes are handled within their own domain.[2]

As a result of these and similar challenges and trends, currently inferred connectivity infrastructures such as Internet router-level topologies are of largely unknown

quality (e.g., different degrees of completeness and/or accuracy[23]) and cannot be taken at face value. In fact, in many cases, inferred connectivity fabrics have nothing to do with router-level connectivity but instead provide instances of logical connectivity, or worse, present an entangled and largely meaningless mixture of layer-3 and layer-2 connectivity.[14, 25] Furthermore, current technology trends suggest that these problems will not go away, leading us to conclude that the practical utility of `traceroute` for studying and characterizing Internet infrastructure can no longer be presumed. Instead, we argue that these developments call for renewed efforts to design alternative Internet measurement methods and we require these methods to (i) be easily deployable, (ii) generate probe packets that are not susceptible to tampering by third parties, and (iii) yield high-quality end-to-end information that is comparable to what a properly executed `traceroute` probe generates.

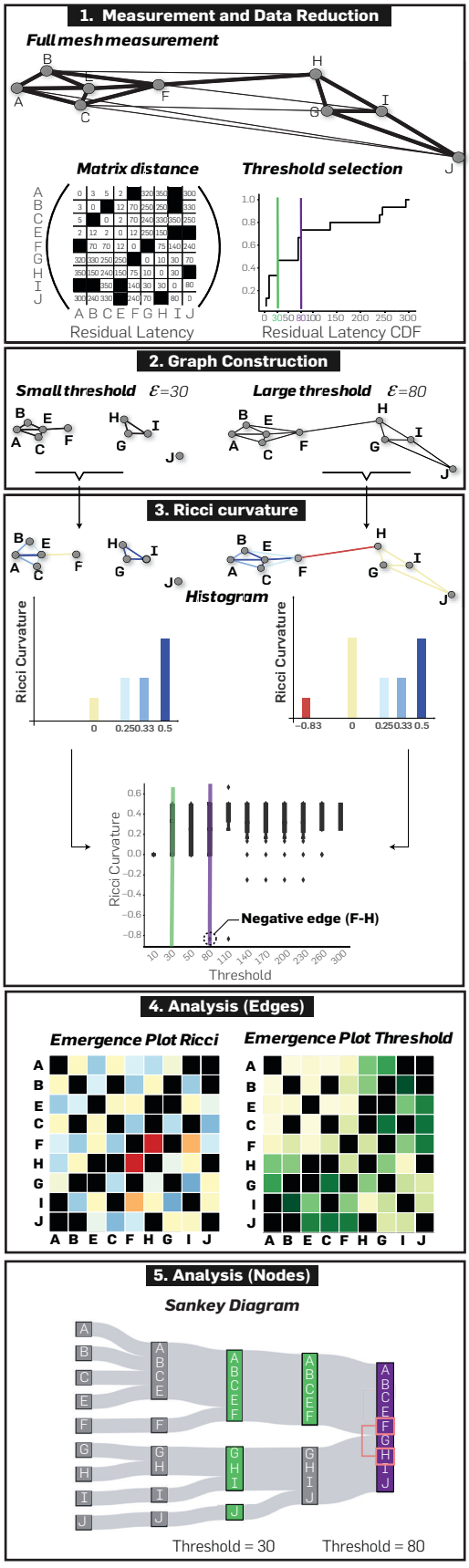
2.2. Getting back to basics

Based on the Internet's dominant service model, the type of measurements that necessarily must be available to third-party researchers are end-to-end measurements that reflect aspects of performance that packets experience as they traverse the Internet between end systems. Of these, the simplest is end-to-end round trip time (RTT) or latency. While `traceroute` probes such as simple ICMP ECHO packets can be blocked, RTT can be inferred from TCP connections, based on timing information provided by SYN/SYN-ACK packets, for any host that has an open port. Furthermore, it is generally possible to obtain at least approximate geo-location information about end systems either through delay-based geolocation or directly. As a result, we can assume that in the foreseeable future, neither RTT nor geo-location information will be fundamentally obscured or completely hidden by network operators from third-party researchers. Although both data sources can be manipulated, the capabilities of an adversary for doing so are limited.[8]

Starting from these two data sources, we envision a characterization of network structure in terms of the properties of paths between end systems. Our approach is to assume that the data (e.g., measured latencies) has been generated from an unknown complex topological space (e.g., continuous manifold), and our methodology provides a means for "learning" this manifold; that is, uncovering the geometry of this unknown delay space. In particular, the data points represent "geodesics" (i.e., shortest paths along the manifold's surface between measurement nodes) that encode a composite view of underlying physical network infrastructure and deployed routing configurations. Intuitively, the "curved" nature of the manifold sheds light on important local properties of the network, with positively curved regions representing more or less pronounced peaks (resulting in shortest paths that are indeed "short" or direct) and negatively curved areas forming saddle point-like shapes (producing shortest paths that are "long" or indirect).[18,20]

We expand on the connection between our approach and continuous manifolds in Section 3 where we describe

Figure 1. Our methodology in a nutshell.



how our methodology enables both local and global characterizations of a network's infrastructure. In particular, we show that our proposed method is ideally suited for providing insights into network connectivity in a post-`traceroute` world where `traceroute` probes have to be assumed to be either blocked or manipulated or result in measurements that are difficult to interpret.

3. METHODOLOGY

A schematic overview of our methodology is shown in Figure 1. In what follows, we describe the different steps that comprise our approach and are illustrated in Figure 1.

3.1. A step-by-step description

Step 1—Measurement and data reduction (Panel #1) The starting point of our methodology is a collection of all-pairs latency measurements among a set of nodes (shown as thin black lines in the graph at the top of the panel). We allow for a small portion of missing measurements, and the unknown physical connectivity fabric is highlighted with thick black lines. For each pair of nodes with latency measurements, we determine the *great circle latency,* which we define to be the great circle distance (GCD) between the nodes divided by $\frac{2}{3}c$, which is the speed of light through fiber.[6] Then, for each node pair, we compute the *residual latency*, which we define to be the measured latency (0.5 * *minRTT*) minus the great circle latency. Note that while residual latency can be influenced by a number of factors including physical paths, logical connectivity defined by routing, and congestion, our focus on private infrastructure and measured latency largely obviates the latter two issues. The resulting matrix X of residual latencies is shown on the left of the panel, with black squares indicating missing observations. We then use X to identify node pairs that are connected by *nearly straight* edges. By "straight" we mean following a great circle (geodesic), and by "nearly" we mean that the total deviation from the great circle is bounded. We formalize this notion by establishing a threshold ε on residual latencies; since it measures how close a measured latency is to its theoretical optimal, we call ε the *performance threshold*. This aspect of our methodology is captured on the right side of the panel, where we show the CDF of residual latencies along with two different performance threshold values (i.e., $\varepsilon = 30, 80$).

Step 2—Graph construction (Panel #2) For any given ε, the set of node pairs that are connected via "nearly straight" edges defines a graph. This graph is defined as the set of edges that have residual latency $\leq \varepsilon$, and the panel shows two examples of such graphs, one for each of the two different values of the performance threshold ε highlighted in the CDF plot shown in Panel#1. A lower threshold defines a graph with fewer edges, and the edges in the graph on the left are a subset of the edges in the graph on the right that results from applying the larger of the two thresholds. The low-threshold graph reveals structure locally that allows us to infer the underlying physical connectivity fabric of the various disconnected components. The more localized the target area of study, the more precise the latency measurements should be for our methodology to make accurate inferences. However, low-threshold graphs say nothing about how the resulting islands of connectivity are linked physically as part of the overall graph. That information can be inferred from what higher-threshold scenarios reveal in terms of important links that are on many shortest paths between nodes in the different local "regions" of the overall infrastructure.

Step 3—Ricci curvature (Panel #3) To assess the *importance* of each edge in a given graph, we use the notion of *Ollivier-Ricci curvature* of an edge[19] (Ricci curvature, or curvature, for short). Intuitively, the curvature of a graph edge can be thought of as a type of local "betweenness" measure, but rather than counting paths, it measures the "optimal transport" of mass from the neighbors of one edge node to the neighbors of the other edge node and can be formally defined using the earth mover's distance, a well-known distance measure between two probability distributions. In particular, negatively curved edges tend to be traversed by many of the local shortest paths; that is, paths that originate in the neighborhood of one end node of a negatively curved edge and connect to nodes in the neighborhood of the node at the other end of that edge. Conversely, positively curved edges tend to be traversed by only a few shortest paths in a local region. Building on this intuition, it is easy to show mathematically that while for graphs that are trees, all edges have negative curvature, the edges in grid-like graphs have approximately zero curvature and the edges in graphs that are cliques all have positive curvature. For each of the two threshold graphs depicted in Panel #2, we use different colors to annotate each edge with its computed curvature and show these annotated graphs at the top of Panel #3, followed by a plot of the histogram of computed edge curvatures for each graph in the middle of the panel. Finally, to obtain a macroscopic view of the system under study and develop an understanding of how the graphs derived from the available measurements evolve as we relax the constraint implied by using "nearly straight" edges, we repeat the graph construction and curvature computation steps for a wide range of performance thresholds and summarize the resulting distributions of computed edge curvature values in a series of boxplots (one per considered threshold) shown at the bottom of the panel.

Step 4—Characterization of edge behavior (Panel #4) To augment our picture of the system under study, we characterize how edges contribute to the connectivity of the system by examining the evolution of the threshold-dependent curvature-annotated graphs as the performance threshold increases. To this end, we study the properties of an edge at the threshold where the edge first emerges (i.e., the lowest $\varepsilon \geq$ the edge's residual latency). We define two metrics for each edge, namely *importance* and *performance*. We consider an edge's importance to be defined in terms of its impact on network robustness—how much disruption (i.e., affecting locally shortest paths) would occur were this edge not present—and use Ricci curvature as our metric, with lower (more negative) Ricci curvature indicative of greater edge importance. For edge performance, we use its residual latency—the value of ε at which the edge appears in the graph. Here, higher values indicate poorer-performing

edges, i.e., edges whose latencies are more inflated with respect to the optimal. We summarize the information provided by these two edge metrics by means of heatmaps as shown in Panel #4. For edge importance (Ricci curvature), key network edges appear as very negative (red) in the heatmap on the left; for edge performance, well-performing edges are the ones that have the lowest values (white or yellow) in the heatmap on the right. In each case, we use black boxes where no per-edge measurements are available.

Step 5—Characterization of node behavior (Panel #5) To further complete our understanding of the system under study, we use a *Sankey diagram* as shown in Panel #5 to illustrate how the nodes that comprise the threshold-dependent curvature-annotated graphs cluster as the performance threshold increases. This plot gives a macroscopic view of how these graphs evolve as the performance threshold increases from left to right. In particular, it shows which nodes form well-performing clusters at low-performance thresholds (left side of the Sankey diagram), which nodes form larger connected components at higher performance thresholds (right side of the Sankey diagram), and how the different clusters merge or split as we vary the performance threshold from low to high or vice versa. The plot also shows that at small thresholds, we typically deal with a disconnected graph—only node pairs with very "straight" edges are connected. As the threshold increases and additional edges are added, at some point the graph becomes connected. Importantly, negatively curved edges play the role of "bottlenecks" or "bridges"; that is, representing critical connectivity in the sense that their removal will either disconnect the graph or drastically lengthen the graph's local shortest paths.

3.2. The manifold view

The graphs that result from applying our methodology to a given system under study represent paths between physical vantage points situated in geographical space. Their analysis shows that deep insights into the underlying system can be obtained by providing a basic understanding of the relationship between the curvature of edges in our graphs and the properties of the underlying geographical space. However, while the notion of curvature is more intuitive for continuous (Riemannian) surfaces than it is for discrete graphs, the problem of representing combinatorial objects such as graphs using smooth topological spaces such as manifolds is notoriously difficult.

We deal with this problem by appealing to a well-known relationship between the Ollivier-Ricci curvature and the standard Ricci curvature.[19] In particular, we leverage this relationship to complement our methodology with a key visualization component in the form of a representation that we refer to as the *manifold view*. While a formal derivation of this manifold view is beyond the scope of this chapter and will be presented in future work, the basic idea behind realizing this novel visualization component in practice is simple. In particular, to generate this manifold view, we start from a "flat" (2D) geographical map and a geographically embedded graph, where the graph's edges are annotated with their edge curvature value. We then produce a new (3D) surface whose local Gaussian curvature in each region approximates the Ricci curvature of the edges that "pass near" that region. Technical difficulties that arise in this context and require effective and efficient algorithmic solutions include constructing suitable triangulations of the 3D surface we seek to generate, ensuring the smoothness of the triangulated surface for more refined triangulations, and guaranteeing convergence of the curvature of the triangulated surface to the desired graph curvature. In general, these and other requirements can be satisfied by formulating manifold view generation as an optimization problem with appropriately chosen objective functions.

Figure 2. A manifold view of the example network used in Figure 1.

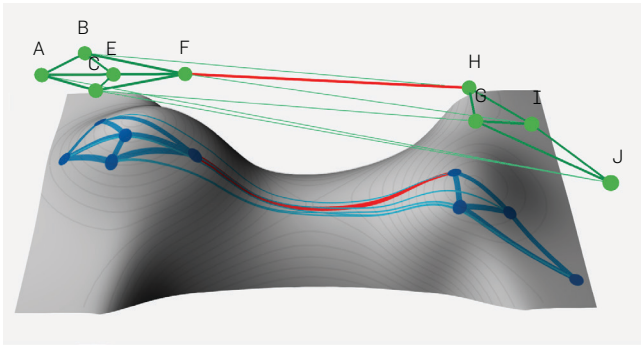


Figure 2 uses the example network from Figure 1 to illustrate the idea behind the proposed manifold view. It shows that while the positively curved edges in the graph (e.g., edges connecting nodes A–F, or nodes G–J) induce two positively curved peaks in the resulting manifold representation, the negatively curved edge (e.g., F–H) induces a saddle point-shaped contour. Recalling the informal description of edge curvature in Step 3 above, a negatively curved edge arises when local paths in the graph tend to make heavy use of that edge; that is, shortest paths are "attracted" to that edge and may be thus "lengthened" as a result of local connectivity properties. This intuition transfers directly to the manifold view (where geodesics define shortest paths) and gives the manifold view a useful interpretation. In regions of negative curvature on the manifold, paths also tend to curve inward toward each other, which in turn attracts the paths toward each other and lengthens them (compared, e.g., to shortest paths on a flat surface with zero curvature). Similar statements apply when viewing positively curved edges from the manifold representation perspective, except that in this case, geodesics tend to curve outward away from each other and thus "shorten" shortest paths on the manifold.

The net result is that a network's manifold view provides a concise, visually intelligible representation of how paths tend to be elongated or shortened by the underlying connectivity of the system under study. The strength of the manifold view is that it visualizes a large number of network measurements in a single representation (i.e., "delay space"), one that combines an easy-to-grasp, conventional geographical component with a more-complex set of all-pairs latency measurements.

4. A NEW LOOK AT HYPERSCALERS

Many of today's large cloud/content providers are busily expanding their private backbone networks, but at the same time, they often consider detailed information about their physical infrastructure to be proprietary and have become reticent to share the details publicly. These developments beg the question: as access to data about the physical infrastructures of the backbones of these hyperscalers is harder to come by while at the same time the importance of these Internet stakeholders increases, is it possible to study the pertinent characteristics of these private backbones without having detailed knowledge of their connectivity fabrics (i.e., existence and/or locations of fiber optic cable conduits)? We answer this question in the affirmative by applying the methodology described in Section 3 to examine the critical features of three hyperscalers: Microsoft (Azure), Amazon (AWS), and Google (GC).

To obtain the necessary datasets, we scheduled measurements in each of the datacenters that these providers operate in their respective availability zones across the globe. Availability zones can be associated with the physical nodes of these providers' global-scale private backbones whose connectivity fabrics we attempt to elucidate, and we approximated the nodes' physical location by using the geographic center of the city where the datacenter is situated. All measurements were collected in late 2019. We used RIPE Atlas,[22] an open, distributed Internet measurement platform, to collect the measurements for AWS. Since GC and Azure did not host RIPE anchors in their datacenters at the time when we performed our measurement campaigns (GC has since started hosting RIPE Atlas Anchors in most of its locations), we configured virtual machines on servers in their datacenters around the world. Separately for each provider, we used *ping* to perform all-pairs latency measurements among the respective set of nodes, annotated each (logical) edge between them with the corresponding measured minRTT value, and computed the physical distance between them based on GCD.

Relying on these datasets as main input, we obtain the analogs of the latency-annotated graph, the residual latency matrix, and residual latency CDF shown in Panel #1 in Figure 1 for each of the three hyperscaler. We then use these artifacts as a starting point for applying our overall methodology described in Section 3 to AWS, Azure, and GC, respectively. In particular, we consider the private backbone infrastructures of these hyperscalers to highlight the kind of empirical findings that we can derive from the proposed step-by-step curvature-based analysis as well as the type of features that become evident when examining the resulting manifold views. However, for brevity, we only discuss here the curvature-based analysis results for a single hyperscaler (i.e., Microsoft Azure) and limit a comparative study of the three hyperscalers to a description of the key features of their manifold views that result from selecting a particular threshold value (i.e., $\varepsilon = 90$).

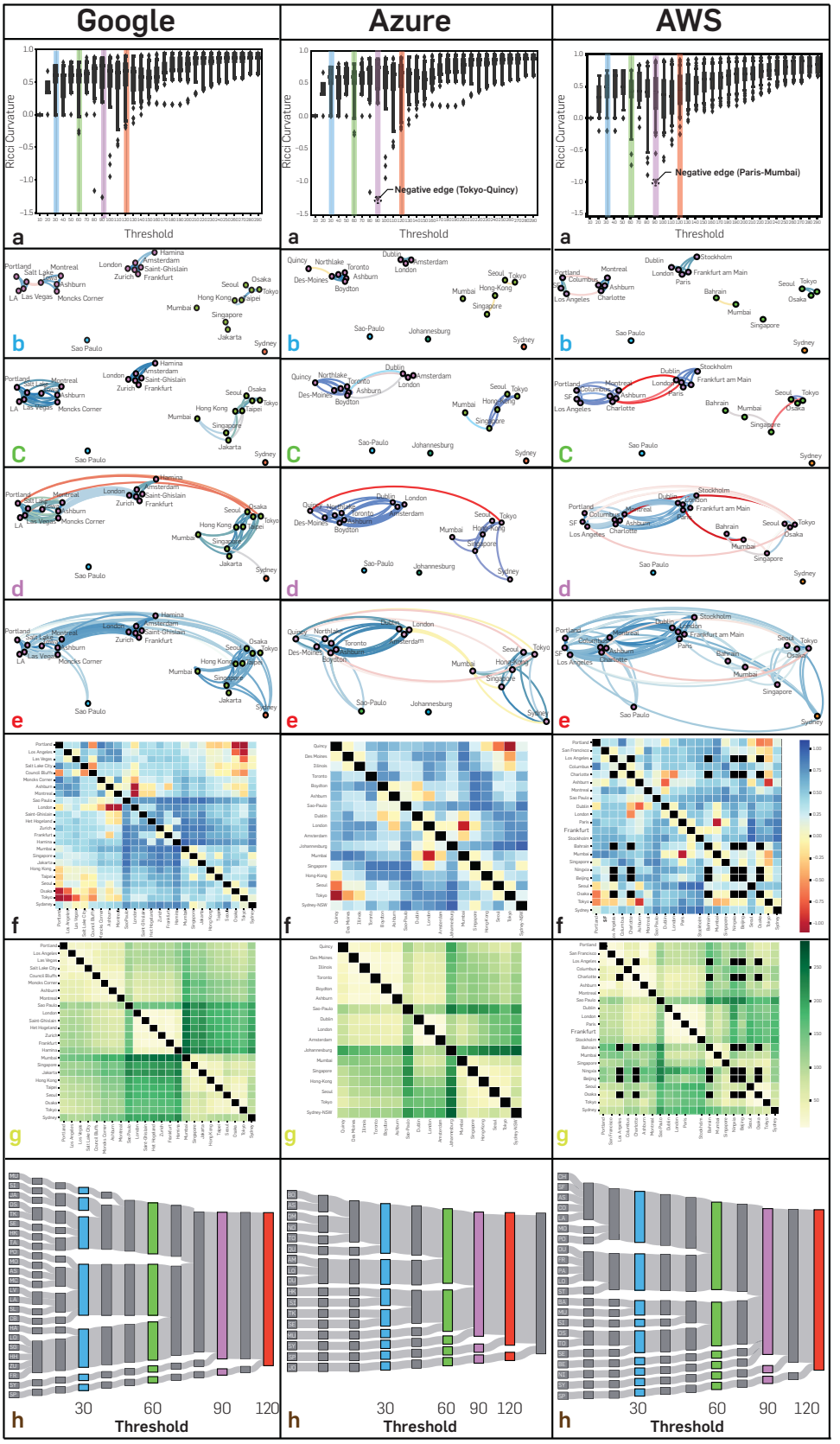
4.1. Through the lens of Ricci curvature

The main results of applying our curvature-based analysis to Azure's private backbone network are shown in the middle column in Figure 3. Panel (a) in the Azure column shows that there is a broad spectrum of performance thresholds, mainly due to the inclusion of submarine cables that result in residual latencies that tend to be significantly larger than those encountered in a strictly intra-continental setting. The threshold value of 60 results in the first edges with negative curvature and Azure's connectivity structure at that threshold is shown in panel (c). It clearly identifies the existence of two geographically meaningful connected components. One of these connected components appears as a pronounced North America-Europe cluster within which Azure operates a richly connected physical infrastructure—all intra-North America edges have positive curvature as do all intra-Europe edges. The formation of this cluster is due to three edges that connect Virginia, US, and the UK-Ireland. Consulting the Azure global network map[15] confirms the existence of direct under-sea cables between Ashburn, VA, and London, and between Boydton, VA, and Dublin. The other connected component is an easily discernible Asia-Pacific cluster with a more unbalanced internal connectivity compared to the North America-Europe cluster. In particular, while Mumbai is only connected to Singapore, there exists rich connectivity among the other four intra-cluster cities Singapore, Hong Kong, Seoul, and Tokyo. This intra-cluster structure suggests the existence of an undersea cable between Mumbai and Singapore. However, since this edge's curvature is not negative, we can conclude that its role is not critical for connectivity within the Asia-Pacific cluster.

Next, examining Azure's connectivity structure at threshold 90 in panel (d), we notice that the entire North America-Europe cluster now consists exclusively of positively curved edges, which implies the existence of rich fiber connectivity linking the two continents by means of transatlantic cables with a diverse set of landing points. As for the Asia-Pacific cluster at this threshold, we observe that it now also includes Sydney. Its intra-cluster connection fabric is also positively curved but is far more sparse than that of the North America-Europe cluster. Importantly, at this threshold, a single negatively curved edge emerges between Tokyo and Quincy, WA (shown in red) that connects the North America-Europe and Asia-Pacific clusters. However, it is only at threshold 120 (see panel (e)) that South America merges with the fused North America-Europe and Asia-Pacific cluster, and South Africa does not merge with the rest of the graph until a threshold beyond 120. Only at that point does the resulting graph represent the global-scale infrastructure of Azure as a single connected component.

These observations are further refined by examining the edge importance and edge performance heatmaps in panels (f) and (g). To illustrate, we focus on the dark red cell corresponding to the edge between Mumbai-London. As mentioned earlier, this edge first emerges between thresholds 90 and 120. While the appearance of light-red colored cells surrounding the Mumbai-London link indicates the emergence of more connectivity for transporting traffic between Europe and Asia, a look at the performance heatmap shows that these alternative routes all have similar suboptimal performance (for reasons that become clear

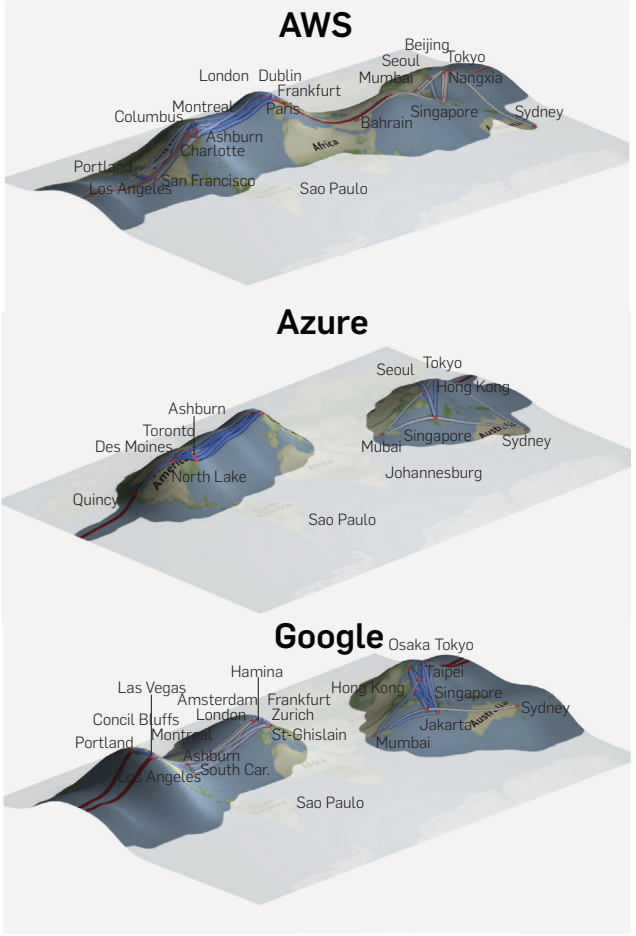
Figure 3. Curvature-based analysis results for AWS (left), Azure (middle), and GC (right): (a) Boxplots of curvature values for different thresholds; (b)–(e): threshold-based graphs projected onto a world map for thresholds $\varepsilon = 30, 60, 90$, and 120; (f) Edge importance (Ricci curvature) heatmaps; (g) Edge performance (residual latency) heatmaps; and (h) Sankey diagrams.



when discussing the manifold view in Section 4.2). More generally, the performance heatmap in panel (g) reveals four distinct aspects. First, we see the emergence of three separate well-performing regions: North America (upper left), Europe (center), and the Asia-Pacific region (lower right). Second, South America (Sao-Paulo) and Africa (Johannesburg) are only weakly connected to the rest of the network and experience therefore overall sub-par performance. Third, while the Europe and North America clusters are composed of uniformly light-colored cells, the cells that make up the Asia-Pacific cluster are colored in different shades of green, indicating that the different cities in this cluster connect via links that vary in performance. Fourth, the links between North America (or Europe) and the Asia-Pacific area emerge in a rather homogeneous manner, indicating overall consistent performance.

Finally, the Sankey diagram in panel (h) shows how the different regions that make up Azure's global-scale infrastructure merge to form a single connected component as the performance threshold increases. It summarizes in a single visual the above-mentioned key observations we derived from our curvature-based analysis of Azure's infrastructure, including (1) the Europe and North America clusters form early on at low thresholds, (2) the Asia-Pacific cluster forms at higher thresholds, (3) the integration of Johannesburg and Sao-Paulo into a connected graph representing the Azure global network takes place at yet higher thresholds, and (4) the emergence of the Tokyo-Quincy, London-Mumbai, Dublin-Boydton, and London-Ashburn links as critical connections in Azure's global-scale infrastructure.

Figure 4. Manifold view of AWS (top), Azure (middle), and GC (bottom) at threshold 90. The corresponding graph representations are shown in Figure 3, row (d).



4.2. With the help of manifold views

We complement our curvature-based analysis of the network infrastructure of the three hyperscalers with a depiction of their manifold views. Figure 4 shows their manifold views for threshold 90, with each sub-figure depicting both a 2D graph of network connectivity embedded in the surface for reference and a manifold projected onto a map (where edges that stop at one side of the boundary of the surface wrap around the earth and continue on the other side of the surface's boundary). We select this threshold because the resulting manifold views allow for an informative comparison of the hyperscalers' connectivity structures.

The positively curved peaks above the plane qualitatively confirm and precisely delineate three regions of rich connectivity at the global level for each of the hyperscalers: North America, Europe, and Asia/Australia. However, the boundaries of the richly connected regions and the nature of the connectivity between them vary considerably for the three networks. For example, GC's connectivity fabric in the Asia/Australia region is more efficient (i.e., has a higher degree of positive curvature) compared to that of Azure or AWS. Representing a comparatively compact region, Europe appears as the most positively curved region for each of the three hyperscalers. In contrast, North America shows two positively curved peaks, one for the West coast region and a more pronounced one for the Northeastern region of the US. Of the three networks, AWS shows the most consistent positive curvature overall for North America.

As far as inter-region connectivity is concerned, each of the positively curved regions is connected to two others by approximately east-west pathways generating negative curvature. However, while for all three hyperscalers, trans-Atlantic connectivity between Europe and North America is in general fairly efficient, with multiple high-performance edges, their Europe-Asia connectivity is overall poor, except for AWS whose manifold view also shows transpacific paths with neutral curvature (in contrast to GC and Azure's manifold views that show only negatively curved transpacific edges). This manifold view-derived observation further qualifies our curvature analysis-based findings about a general lack of connectivity between Europe and Asia/Australia. In particular, the manifold view can be used to identify areas of geopolitical tension that cause sub-optimal routes or examine the impact of particular topographical features (e.g., mountain ranges). For example, placing the earlier-mentioned Mumbai-London edge on a geographic map (as in the case of AWS) shows that any shortest path cable route would have to cross Iran and the Black Sea. However, existing geopolitical tensions afflicting those regions prevent most hyperscalers from targeting these areas for new cable

deployments, requiring them to instead use alternative suboptimal routes, at least in the foreseeable future.

5. CONCLUSION AND OUTLOOK

This article is motivated by recent developments that make it more onerous for third-party researchers to obtain data or study the connectivity of the increasingly important private infrastructures of today's hyperscalers. Our main contribution is a newly proposed methodology for elucidating connectivity in known or inaccessible network infrastructures by leveraging lightweight and readily available latency measurements and demonstrating the use of mathematical techniques from the field of Riemannian geometry for analyzing these measurements. Specifically, our approach is based on the application of Ricci curvature, which allows for encoding of a composite view of underlying physical infrastructure and deployed routing configurations into weighted graph representations of a set of measured network latencies. We illustrate our curvature-based analysis with examples of three hyperscalers and demonstrate its ability to expose critical aspects of their private connectivity fabrics.

The different threshold-based graph structures generated from a collection of simple pairwise latency measurements between nodes in a network allow for a geometric representation of the data as a complex topological space in the form of an idealized smooth manifold. Such manifolds offer unprecedented opportunities for characterizing a network's underlay that gave rise to the measured latencies in the first place. Unraveling the presumed metric space structure of this complex geometric object and leveraging it to identify pertinent and otherwise hard-to-detect features of a given network's connectivity fabric and exploiting them in practice looms as a promising avenue for future work.

Our generated manifold views are the first foray into this largely uncharted territory of using manifold representations of network measurements to visualize a network's performance or behavior in novel and informative ways. In particular, for the latency measurements considered in this work, the generated manifolds distill the main findings from our curvature-based analysis of each of the considered hyperscalers' connectivity fabric into a single view. This view provides a snapshot of that hyperscaler's delay space where the measured latency between any pair of embedded geolocated nodes is realized as geodesic distance. We hope that our work encourages further efforts on this topic, including the development of animations suitable for visualizing the results of a longitudinal study of a network's delay space or for using the network's delay space to understand the effects of real-world or artificially-induced infrastructure failure events (e.g., earthquakes, the addition of particular links); the use of metrics other than latency (e.g., throughput, packet loss); and applications to non-private network infrastructures such as the public Internet.

Acknowledgments

This work was supported in part by NSF grants CNS-1703592, CNS-2039146, and CNS-2106517, and a Google grant.

References

1. Reliance Communication plans undersea cable to meet data demands of Asia, Europe, 2018. https://tinyurl.com/ybjv7mdp.
2. Arnold, T., He, J., Jiang, W., Calder, M., Cunha, I., Giotsas V., et al. Cloud provider connectivity in the flat internet. In *Proc. ACM IMC'20* (2020), 230–246.
3. Augustin, B., Cuvellier, X., Orgogozo, B., Viger, F., Friedman, T., Latapy, M., et al. Avoiding traceroute anomalies with paris traceroute. In *Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement, IMC '06* (New York, NY, USA, 2006), ACM, NY, 153–158.
4. AWS. AWS Global Cloud Infrstructure, 2020. https://www.infrastructure.aws.
5. Beverly, R. Yarrp'ing the internet: Randomized high-speed active topology discovery. In *Proceedings of Internet Measurement Conference* (Santa Monica, USA, 2016), 413–420.
6. Bhattacherjee, D., Jyothi, S.A., Bozkurt, I.N., Tirmazi, M., Aqeel, W., Aguirre, A., et al. cISP: A speed-of-light internet service provider. *arXiv*, (1809.10897), 2018.
7. Detal, G., Hesmans, B., Bonaventure, O., Vanaubel, Y., Donnet, B. Revealing middlebox interference with tracebox. In *Proc. ACM IMC'13* (Barcelona, Spain, 2013), 1–8.
8. Gill, P., Ganjali, Y., Wong, B., Lie, D. Dude, where's that IP?: Circumventing measurement-based IP geolocation. In *Proceedings of the USENIX Security Symposium (USENIX Security 10)* (Washington, DC, USA, 2010).
9. Hong, C.-Y., Kandula, S., Mahajan, R., Zhang, M., Gill, V., Nanduri, M., et al. Achieving high utilization with software-driven WAN. In *Proceedings of the ACM SIGCOMM Conference on SIGCOMM* (Hong Kong, China, 2013), 15–26.
10. Jacobson, V. Traceroute, 1989. ftp://ftp.ee.lbl.gov/traceroute.tar.gz.
11. Jain, S., Kumar, A., Mandal, S., Ong, J., Poutievski, L., Singh, A., et al. B4: Experience with a globally-deployed software defined WAN. *ACM SIGCOMM Comput. Commun. Rev. 43*, 4 (2013), 3–14.
12. Jimenez, M., Kwok, H. Building express backbone: Facebook's new long-haul network, 2017. https://engineering.fb.com/data-center-engineering/building-express-backbone-facebook-s-new-long-haul-network/.
13. Kaufmann, C. ICN—Akamai's Backbone, 2018. https://www.linx.net/wp-content/uploads/LINX101-Akamai-ICN-ChristianKaufmann.pdf.
14. Li, L., Alderson, D., Willinger, W., Doyle, J. A first-principles approach to understanding the internet's router-level topology. *ACM SIGCOMM Comput. Commun. Rev. 34*, 4 (2004), 3–14.
15. Microsoft. Azure Microsoft Global Network Map, 2020. https://docs.microsoft.com/en-us/azure/networking/microsoft-global-network.
16. Microsoft. Azure Virtual Network frequently asked questions (FAQ), 2021. https://learn.microsoft.com/en-us/azure/virtual-network/virtual-networks-faq.
17. Motamedi, R., Rejaie, R., Willinger, W. A survey of techniques for internet topology discovery. *IEEE Commun. Surv. Tutorials 17*, 2 (2014), 1044–1065.
18. Ni, C., Lin, Y., Luo, F., Gao, J. Community detection on networks with ricci flow. *Sci. Rep. 9*, 1 (2019), 1–12.
19. Ollivier, Y. Ricci curvature of Markov chains on metric spaces. *J. Funct. Anal. 256*, 3 (2009), 810–864.
20. Ollivier, Y. A visual introduction to riemannian curvatures and some discrete generalizations. In *Analysis and Geometry of Metric Measure Spaces: Lecture Notes of the 50th Seminaire de Mathematiques Superieures (SMS)* (Montréal, 2011), 56.
21. Putzier, K. Property investors see fiber-optic cables as 'Railroads of the Future', 2020.
22. RIPE. RIPE Atlas, 2020. https://atlas.ripe.net.
23. Spring, N., Mahajan, R., Wetherall, D. Measuring ISP topologies with rocketfuel. *ACM SIGCOMM Comput. Commun. Rev. 32*, 4 (2002), 133–145.
24. Vahdat, A., Clark, D., Rexford, J. A purpose-built global network: Google's move to SDN: A discussion with Amin Vahdat, David Clark, and Jennifer Rexford. *ACM Queue 13*, 8 (2015), 100–125.
25. Willinger, W., Alderson, D., Doyle, J. C. Mathematics and the internet: A source of enormous confusion and great potential. *Not. Am. Math. Soc. 56*, 5 (2009), 586–599.

Loqman Salamatian, Columbia University, New York, NY, USA.

Scott Anderson, Joshua Mathews, and Paul Barford, University of Wisconsin-Madison, Madison, WI, USA.

Walter Willinger, NIKSUN, Inc., Princeton, NJ, USA.

Mark Crovella, Boston University, Boston, MA, USA.