2019

Rethinking Routing and Peering in the era of Vertical Integration of Network Functions

Prasun Kanti Dey
University of Central Florida

RETHINKING ROUTING AND PEERING IN THE ERA OF VERTICAL INTEGRATION OF NETWORK FUNCTIONS

by

PRASUN KANTI DEY
M.S. University of Nevada Reno, 2016

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical and Computer Engineering
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Fall Term
2019

Major Professor: Murat Yuksel

ABSTRACT

Content providers typically control the digital content consumption services and are getting the most revenue by implementing an "all-you-can-eat" model via subscription or hyper-targeted advertisements. Revamping the existing Internet architecture and design, a vertical integration where a content provider and access ISP will act as unibody in a sugarcane form seems to be the recent trend. As this vertical integration trend is emerging in the ISP market, it is questionable if existing routing architecture will suffice in terms of sustainable economics, peering, and scalability. It is expected that the current routing will need careful modifications and smart innovations to ensure effective and reliable end-to-end packet delivery. This involves new feature developments for handling traffic with reduced latency to tackle routing scalability issues in a more secure way and to offer new services at cheaper costs. Considering the fact that prices of DRAM or TCAM in legacy routers are not necessarily decreasing at the desired pace, cloud computing can be a great solution to manage the increasing computation and memory complexity of routing functions in a centralized manner with optimized expenses. Focusing on the attributes associated with existing routing cost models and by exploring a hybrid approach to SDN, we also compare recent trends in cloud pricing (for both storage and service) to evaluate whether it would be economically beneficial to integrate cloud services with legacy routing for improved cost-efficiency. In terms of peering, using the US as a case study, we show the overlaps between access ISPs and content providers to explore the viability of a future in terms of peering between the new emerging content-dominated sugarcane ISPs and the healthiness of Internet economics. To this end, we introduce *meta-peering*, a term that encompasses automation efforts related to peering – from identifying a list of ISPs likely to peer, to injecting control-plane rules, to continuous monitoring and notifying any violation – one of the many outcroppings of vertical integration procedure which could be offered to the ISPs as a standalone service.

It is often said that

the best way to express your gratitude to your parents is to become the reason of their smiling.

Trust me, it is true. I have seen it, and it is wonderful.

To my parents.

ACKNOWLEDGMENTS

It is my pleasure to acknowledge the people who bestowed their blessings on me as a form of continuous support, kept faith in my capability to pursue this ambitious endeavor, and had direct or indirect contributions in reshaping me till today.

First and foremost, I am so much indebted to my supervisor and mentor, Dr. Murat Yuksel, for advising such an adolescent like me. Way back in a day, more than five years ago, when I had the opportunity to first met with Dr. Yuksel, I found him extremely sincere and passionate about research. I can say without any reservation that he had the most profound influence on my research work. He not only taught me how to tackle a problem and investigate for a more long-lasting solution over short-term gains but also instilled in me the confidence to explore new areas for identifying appropriate research problems that have real-world impact. Without his consistent guidance and positive attitudes during the period of setbacks, I would not have much success in grad school.

I would like to thank my dissertation committee: Dr. Rickard Ewetz, Dr. Samiul Hasan, Dr. Jun Wang, and Dr. Wei Zhang for their constructive comments. All these feedback helped me to improve the final version of this dissertation. Specially, Dr. Wei Zhang's course on Network Optimization is one of the best classes that I had taken during my graduate years.

Regarding course works, I consider myself extremely lucky to bag the opportunities of attending some amazing professors' classes both at the University of Central Florida (UCF) and the University of Nevada, Reno (UNR), where I completed my Master's degree. In particular, the courses offered by Dr. Shamik Sengupta and Dr. Mehmet Gunes were instrumental during my initial years to help me understand the details of networking, and its associated security issues.

I would also like to express gratefulness to my friends and lab mates, from UCF as well as from UNR. You were there, forever, by my side, and from time to time, you were generous enough to take the time from your busy schedule to provide me constructive criticism. Often, your suggested solutions might appear to be the weirdest one, but surprisingly they worked! You guys are awesome. I consider myself fortunate that your paths have crossed with mine, at least for a very brief period.

Pursuing a Ph.D. is a long journey and requires significant commitment. I am sorry, my loving family members, that I had to choose either you or my research during the last five years. But, you were always beside me even though I could not spare the time that you deserve. My parents, brother, sister-in-law, thank you for being so supportive. Your care, warmth, and kindness kept me going. You have always been the pillar of encouragement and a source of inspiration. Little cutie, Riddhi, I miss you so much. I wish I had more time to play with you.

Last but not least, I would like to thank my wife, Akhi, to whom I owe an enormous debt of gratitude for tolerating my extremely unsocial and impatient behavior. Sometimes I wonder, how can she keep herself so calm and be so enthusiastic at the same time! I know, no words are adequate to thank you enough. I could not have made it so far without you. You deserve all the credits, and this achievement belongs to you as much as it pertains to me.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

The Internet continues to witness a dramatic growth (4.53 Billion end users in June 2019 [4]) in traffic as people are more interested in diverse applications varying from watching high-quality videos, streaming music, and playing online games to transferring bulk-data or making financial transactions over the Internet. With data being generated continuously, complexity grows, and to keep the "net" working, layered architecture and standard protocols were introduced. Key stakeholders of the current design are Internet eXchange Points (IXPs) and Internet Service Providers (ISPs), which can be either content, transit, or access ISPs [123]. While content providers/ISPs generate various contents to be consumed by the end customers, transit and access ISPs are responsible for delivering the data to end-users smoothly by setting up new fibers and maintaining the existing infrastructure to ensure the connectivity.

The current Internet topology is roughly hierarchical. In Figure 1.1, we see the logical architecture of the ISP types in the Internet. This tier-model divides ISPs horizontally, based on their nature.

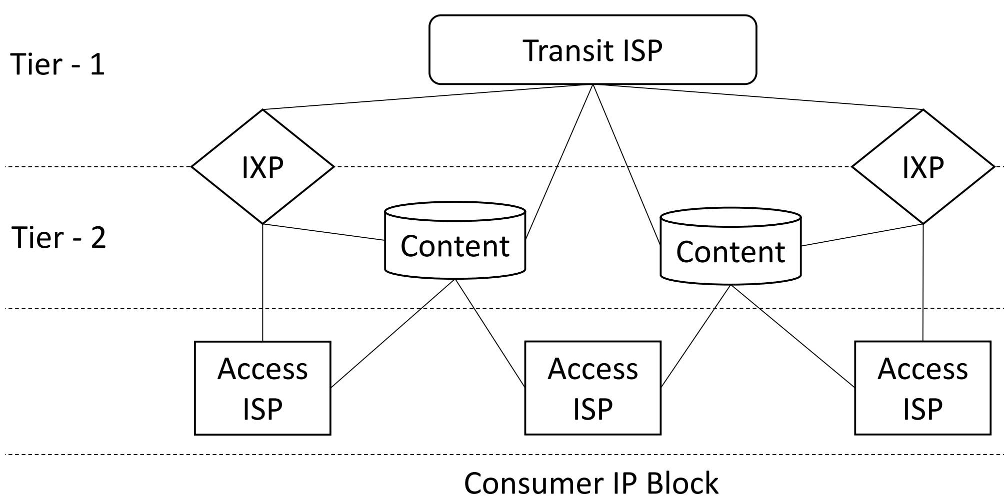

Figure 1.1: Internet ISP Architecture

New tiers or service providers can be added to this model as they emerge, like attaching a new piece of Lego in an existing Lego construction [122]. Note that real ISPs may perform some of these logical services at the same time. An ISP can own one or more Autonomous Systems (ASes) to separate its business and to maintain better operational segregation. Any privately-owned and/or government-regulated organization that implements its internal network, clearly states its external routing policy, and is willing to connect with other organizations via upstream transit or exchanging traffic directly with similar-sized entities can be treated as an AS [105]. Starting from content providers (CPs) or content-delivery networks (CDNs) to transit or access (eyeball) providers, any enterprise or educational institution (profit or non-profit) who have aforementioned qualifications can emerge as an AS by purchasing an Autonomous System Number (ASN) from one of the five Regional Internet Registries (RIRs)[1]. Access ISPs who sell Internet connectivity to the end-users are also ASes, and they have their ASNs as well.

With almost 89K Autonomous Systems (ASes) [136, 4] composing the Internet, it is nearly impossible for an individual AS to connect with all the rest of them by establishing direct physical fiber cable and obtain its global reachability. Though some organizations are capable of maintaining such an enormous network backbone, they are of a scarce breed [145] and restrictive in nature. If an AS wants to connect with one of these giant ASes, it needs to be capable of ticking all the points in a stringent criteria checklist, set up by those ASes [161].

Interconnection between ISPs depends mainly on ASes' sizes, their geo-coverage, existing user counts, and the cost of traffic-offloading at a specific place. If an ISP is smaller in size, it will generally purchase *transit* services from a bigger provider. Size can be reflected in various ways; ISPs can calculate the customer cone (detailed in Section 3.1) and/or market capital in this regard.

[1]Five RIRs are: 1. African Network Information Center (AFRINIC), 2. American Registry for Internet Numbers (ARIN), 3. Asia-Pacific Network Information Centre (APNIC), 4. Latin America and Caribbean Network Information Centre (LACNIC), and 5. Réseaux IP Européens Network Coordination Centre (RIPE NCC).

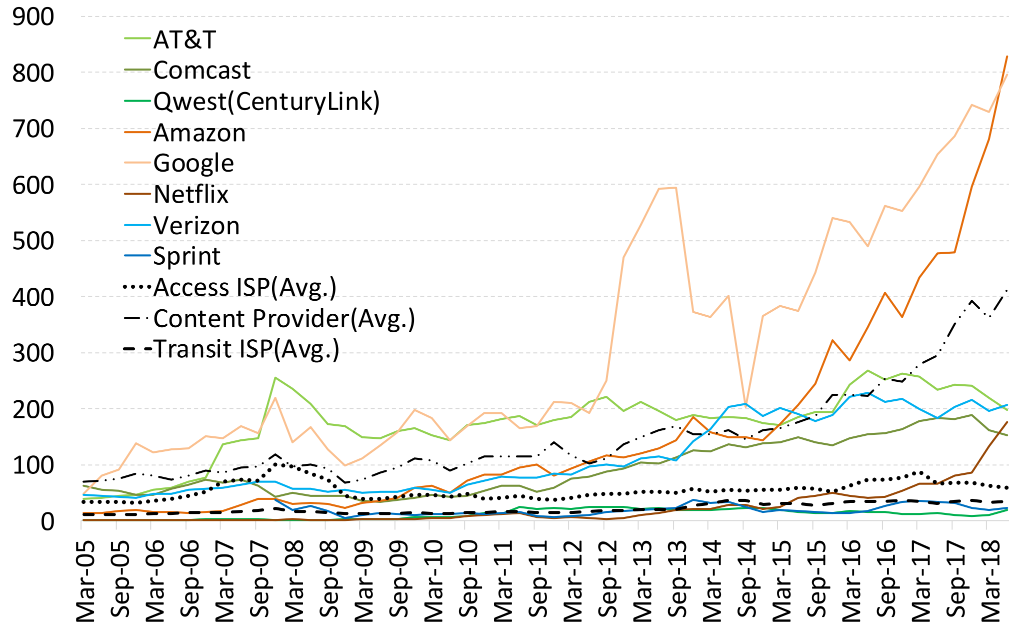

Figure 1.2: Market Capital (in Billion $) of different ISP types. (Averages of 9 Access, 9 Content and 10 Transit ISPs)

An ISP can also choose to exchange its traffic directly with another ISP without paying anything in a settlement-free manner. The second type of arrangement is known as *peering*. IXPs play a crucial role here as they house multiple carrier ISPs[2] (covering different geographic areas) in the same place so that ISPs can exchange their traffic in strategically located facilities. However, the scale and the characteristics of these ISPs are significantly heterogeneous from each other. *AS* and *ISP* have their distinguished meaning in network studies, and as previously mentioned and according to RFC 1930 [106], multiple ASes (with individual AS number) can act together as a single ISP. But, to highlight more on business strategy and to keep aligned with what we observe from our data, from here onwards, unless explicitly mentioned, *AS* and *ISP* infers the same entity.

In the traditional horizontally organized system, carrier ISPs charge end-user a fee for connectivity, and CPs for ensuring high-speed data delivery (Apple to Comcast [166]) or improved stream qual-

[2] A carrier ISP can be either an access/eyeball and/or transit ISP.

ity (Netflix to AT&T [201] or Comcast). Still, charging more to CPs do not give much breathing space to carrier ISPs. Most of the money is in the content business (see Figure 1.2), so the trend is to invest more in creating own content and serve it, or acquire an existing CP to gain its control in an "if you can't beat them, buy them" manner.

In order to dimension and scale the Internet along with the demand growth, it is prudent to understand its structure as shown in Figure 1.1, the critical patterns in its underlying infrastructure, and the dynamics that govern the Internet usage as well as financial and political factors. With CPs growing more prominent and gaining the market share, whether there will be any eminent reshaping of the Internet architecture and, if so, identifying the driving force behind that is crucial. Another essential area to explore is how these ISPs will interact with each other in a synchronized manner to ensure the Internet connectivity and to deliver online content, efficiently.

Moreover, from a technical point of view, the paradigm is shifting towards a complete separation of the router's control and data plane for better management that also offers higher scalability. The number of new devices connected to the Internet is increasing consistently, and ISPs feel a dire need of optimizing these devices' connectivity information. The Border Gateway Protocol (BGP) table size has already exceeded the 800K entries (as we see from Figure 1.3) and, the introduction of longer prefixes (usage of IPv6) due to shrinking IPv4 space worsens the overall scenario. Since the routing complexity is growing and the need for software platform becomes inevitable, a methodical analysis of recent trends and their coherent interpretations become more vital than ever.

1.1 ISP Business Dynamics

The ever-increasing large demand for the Internet usage and ISP industry growth [59] triggered great interest to understand the nature of its economics and future techno-economical sustainability.

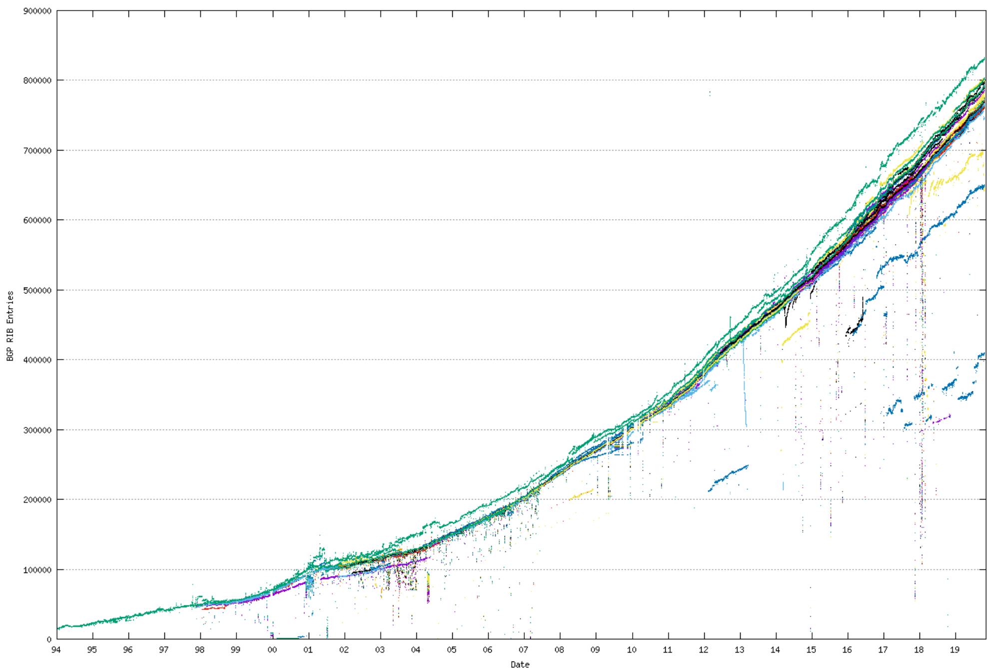

Figure 1.3: Growth of the BGP Table - 1994 to Present [38]

Due to the complexity of negotiations and dynamics among many entities involved, germinating an efficient pricing scheme for ISP services have attracted a lot of researchers [162, 178], too.

Understanding the dynamics of the ISP industry is also of great importance to develop models for the long-term health of the ISP market. Prior research looked at the techno-economic reasons why ISP businesses fail or succeed [179] and promoted enhanced revenue sharing [134] and game-theoretic settlement models [41]. Early studies indicated that the fraction of connections (either peering or customer-provider relationship) derived from AS-Path information of BGP announcements reflect only 60% of the link types. This means any information obtained from BGP lacks around 40% of the edges [75, 84, 108]. Dhamdhere and Dovrolis [84] showed that available Route-

Views [12] and RIPE [9] monitors cannot detect or infer all peering links while CAIDA[3] provides a more detailed classification of the inter-AS relationships [16, 101]. In this dissertation, we utilize the CAIDA classification for studying inter-AS relationships.

Analyzing the interaction between network and business properties of ISPs have received attention as well. One fundamental point of the connectivity on the Internet is that every AS establishes and optimizes its connectivity, focusing on its requirements and business goals [88]. The role of topological properties and policies have been investigated to maximize revenue while meeting performance requirements. Agreements between ISPs are not new [58], but based on their business model and analysis, these linkings are consistently evolving. Having said that, neither the relationship between the Internet's topological characteristics and stock market nor dependency and inference of inter-ISP economic ties have been investigated deeply from the routing and forwarding measurement data.

Mainard et al. [137] studied the relationship between the physical network of ASes and the business performance of the ISPs of the same type. They showed that ISPs providing the same service types are positively correlated in terms of stock market performance, and, further that, geographical proximity increases the correlation – verifying the intuition that the ISPs compete for the same or similar customer pool. Rather than cross-correlating ISPs in the stock market, here, in this dissertation, we correlate the network characteristics of ISPs to their stock market performance and attempt to find out the dependency of ISPs' businesses to their network characteristics, particularly to their relative position in the inter-AS topology.

[3]Center for Applied Internet Data Analysis (http://www.caida.org/)

1.2 Vertical Peering

The recent ruling on Net Neutrality favors carrier ISPs and allows them to distinguish traffic. ISPs can now *legally* [37] prioritize data before delivering to end-users, and a user may experience a delay while streaming content from a provider or accessing a certain website. Breaking the status quo means carrier ISPs now have an unfair advantage over CPs if there raises any conflict of interest between individual CP's traffic vs. another provider affiliated to a particular ISP. The merger of AT&T and Time Warner ignites the following question, *what if AT&T starts favoring its own content over its competitors [163]?* On the opposite side, some CPs have also started provisioning access or making "paid peering" deals [148] to make their content reach end-users faster and offer better Quality of Service (QoS).

It appears that the existing Internet structure, which separates the providers *horizontally*, will no longer be applicable, rather a *vertical* integration [165] of multiple players from different layers seems to emerge. This nascent architecture will eventually eliminate the typical access and transit ISPs. In this new Internet design, CPs will likely dominate, sitting on the top, and the means of content delivery (i.e., transit and access) will be vertically integrated to the CPs all through the carrier ISPs to the end-users. We name such *vertically* integrated ISPs as *sugarcane* ISPs, which will (or can) be a conglomerate of ISPs. This event can take place in two ways: A carrier ISP can acquire a CP (Verizon acquiring Yahoo!) or vice versa. Analyzing ISP business dynamics will set us on solid ground to identify the potential acquisition or facilitate ISPs to ascertain possible merging opportunities and form a *sugarcane* ISP.

Akin to how existing ISPs benefit from peering with other ISPs, the new *sugarcane* ISPs also need to peer or come to a business consensus between themselves to attain low latency and reduced end-to-end delivery cost. We call such agreement as "*sugarcane* peering". In general, refraining from peering calls upon selfish routing [174] and introduces unnecessary traffic re-routing, additional

end-to-end delay, and instead of bringing the world closer, it could steer the Internet into isolated parts. Without peering, end users could be secluded and forced to see only specific contents from a selected group of providers, which is unacceptable and violates the ground rule of the *Open Internet*. At this point, we detail the benefits of peering and how ISPs can identify potential peers.

1.3 Meta Peering

Typically, in settlement-free peering, it is "sender keeps it all" [111], i.e., sender ISP keeps all the money it charges from its end-customers and hands the traffic over to peer. It is upon the peer, who has to carry the traffic towards the destination without charging the originating ISP. Since peering is not transitive, neither entities can route traffic from other peers through the direct connectivity between them.

Peering is often preferred over transit to achieve better control on routing, ensuring low latency for the end customers by reducing the path stretch and, most importantly, to slash the cost. Traffic can be directly delivered to the destination AS, which, in return, will make the propagation delays for peering paths smaller compared to paths via transits. Peering provides increased redundancy and more control on routing for a substantial amount of traffic and improves performance since it redirects traffic via friendly peers to avoid bottlenecks. For instance, by leveraging peering paths, 68% of ASes connected to 920 access ISPs experienced 10 ms improvements in latency, and for 91% of those ASes, peering paths outperform transits [42]. Lack of peering causes extraneous traffic detours and increases the path stretch [151], like what African or Latin American ISPs are now dealing with [104, 96]. Such fragmented routing causes local traffic (i.e., both the origin and destination located in the same or neighboring country) to traverse unnecessarily other continents (in the above cases, Europe and the US, respectively) and ultimately degrades the end-to-end performance. This circumstance is likewise for the US providers, as well. A study [210] found that

mobile client traffic from AT&T Seattle had to enter Google's network in the Bay Area due to a lack of peering point nearby. At this point, it is not to be assumed that settlement-free means traffic delegation without any cost at all. But, depending on the transit fee and the total cost involved in peering, narrowed down to per unit, 1 Gbps peer connection can be cheaper when exchanging at least 100 Mbps or 250 Mbps traffic [194, 44].

If ISPs agree on peering, based on the amount of total exchanged traffic, they can peer either in private facilities or public IXPs. In private peering, a higher volume of traffic is exchanged, and both ISPs establish direct physical connectivity [100]. If an ISP decides to peer with another ISP in the same location, it has to establish separate direct connectivity. On the other hand, in an IXP, an ISP can peer with multiple ISPs in a shared environment by only paying the port fee for each new peer and/or membership fees incurred from renting space at the IXP. Such peering type can still be beneficial even if the amount of traffic is relatively small. Either way, ISP has to carry its traffic to a common Point of Presence (PoP), which can be in the same building of the city or a completely different common location [100, 194]. In public peering, an ISP can form either bi-lateral peering with only one ISP or has the flexibility to negotiate a multi-lateral peering agreement with many ISPs by joining in a centralized route-server (RS). In this dissertation, we are specifically interested in bi-lateral public peering.

Whether it is peering or transit, an ISP has to establish new BGP sessions for each of its neighboring ISPs and repeat this process for every single one of them. ISP has to continuously monitor each separate session to check if those are up and running. Blending BGP with Interior Gateway Protocol (IGP) without conflicting the external and internal policies is quite complicated. Errors due to human involvement increase the chance of network outages, and thus maintenance of these sessions become time-consuming and cumbersome. To ease the process by reducing human coordination delay, some ISPs have already implemented tools for automating the BGP peering process [115, 60]. Existing tools (e.g., NetFlow, sFlow, IPFIX) are also capable of network

monitoring [127]. They can identify the routes causing an outage so that stratagems like manual reconfiguration or temporary shut-down can be initiated.

Even though the BGP peer establishment process has been notably automated, the peering process is a more challenging one to automate. Choosing the right peer and the PoP(s) is difficult due to the following facts:

- 99% of peering is a handshake [204]. There are no fixed written rules. ISPs have a list of prerequisites, but those may not be strictly maintained.
- Peering with multiple ISPs is a hassle, although modern switches (e.g., BIG-IP 2000S) offer better management and improved application performance by selecting the best route for both in- and outbound traffic [89, 164].

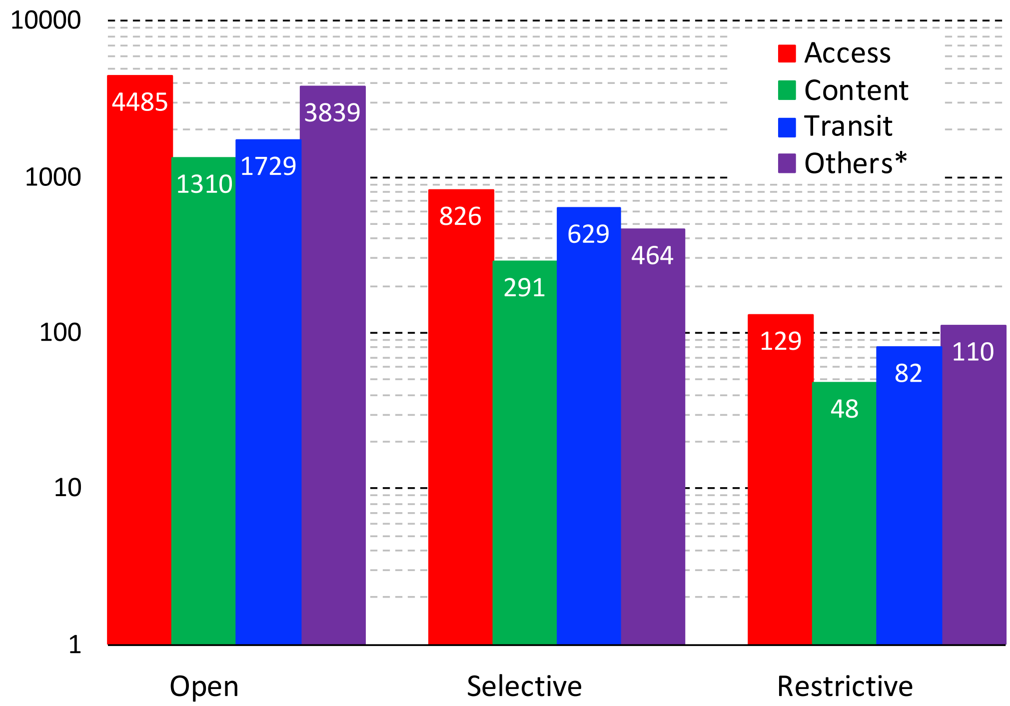

Figure 1.4: ISPs' peering policies in PeeringDB (* = Educational, Non-profit, Enterprise, etc.)

- Before connecting to an RS, an ISP needs to evaluate its capacity to carry all the peers' combined traffic.
- Peers try to balance "bit miles" between them [186]. So, identifying a PoP close to only one ISP may not always be helpful.
- ASes implement different peering policies according to their own network infrastructure. We use PeeringDB [7] data of 15,078 ASes to perceive their preferred policy and classify ISP types (access, content, transit, and others). Regardless of the ISP type, most of the ISPs (in orders of magnitude) are *open* to peer, while very few are *restrictive* about peering (see Figure 1.4). 1,136 ISPs either have no preferred policy, or they do not want to publish it in PeeringDB. Though it may be easy to find an ISP willing to peer, it is challenging to motivate ISPs from the Selective and Restrictive groups without providing enough evidence for peering incentives.

Considering these issues and the current automation efforts, we believe that it is needed for the ISPs to identify the potential peer ISPs based on the estimated traffic, customer cone size, peering policy, and overall cost of peering in a PoP. Since the main motive behind peering is to reduce the cost, there has been extensive research on game-theoretic modeling of peering [49], and understanding the economics behind Internet pricing where multiple ISPs are involved [47, 178]. However, automating the identification of ISPs to peer has lacked enough attention.

This dissertation focuses on answering a key question in the Internet peering: *"How far peering relationships can be automated?"* We envision the entire peering process as automatic as possible, where the system will suggest a list of peers, the feasible PoP locations with the least costs, and, if the other peer accepts the peering proposal, the system will automatically generate the BGP configuration to establish the session. Once everything is set up, the monitoring phase will initiate and take control of the system. Similar to how *metadata* refers to data about data or *meta-economics*

is the discussion beyond economics, we consider the entirety of every tool, algorithm, and other necessary components needed for an automated peering process as *Meta-peering*.

1.4 Hybrid Routing

Another critical challenge ISPs need to deal with is to efficiently fit the ever-growing number of prefixes in the existing routers. As the cost of a routing unit is not keeping up with Moores Law and has been decreasing slowly in comparison to the escalation of data traffic surge rate [173], this is quite concerning. The apparent measurement taken by ISPs, to cope up with the need for expanded Forwarding Information Base (FIB) table, is installing additional Ternary Content-Addressable Memories (TCAMs) or Dynamic Random Access Memories (DRAMs) in routers. But still, it is questionable how sustainable this approach will be in the future. It is difficult to point out the notable cost reduction in basic routing at the Internet core, because of the high expectation of packet processing capacities from a BGP router and the increasing line rates to accommodate the data traffic growth which is virtually doubling per year. It is necessary to add more switches and routers in the network backbone to attain parallelism, which will not only improve the load balancing but also will act as a backup route. Software-Defined Networking (SDN) is poised to be an effective solution for overcoming this situation, which breaks the limitation of current infrastructure by separating the control plane and data plane. With the development of open interfaces like OpenFlow [140], the trend to offload control plane complexity from core network components like routers to software platforms for centralized optimizations [92] or outsource the control plane tasks to remote platforms [128], such as cloud, are gaining their popularity.

In a service-oriented and software-driven market, ISPs are more interested in offering cheaper but better services using programmable network infrastructure that ensures QoS too. This inference is valid for all types of ISPs. SDN's complete separation of the data and control planes model

offers great flexibility in managing the overall network. Its vendor-agnostic behavior and programmability make individual development of these planes natural and encourages innovations to patch a fix on the fly. As an example, OpenFlow v1.3 gives network administrators the luxury to configure QoS for the entire network from a single controller and delegate the responsibility to update all the remaining routers to the controller instead of updating each of them manually by themselves, where OpenFlow v1.0 lacked this support [112]. However, credible performance and scalability concerns have already been raised as this may not be sufficient enough for supporting data-extensive applications such as multimedia streaming.

Nevertheless, most of the SDN research encompasses use cases specific to data-centers. ElastiCon [85] deals with the static mapping of a switch to the controller, Avalanche [113] was developed to enable multicasting in switches, and a Network Virtualization Platform [121] was proposed for enterprise-level multi-tenant data-centers. The success of SDN in data-centers motivates researchers to extend the software-defined approach to find solutions in other scenarios like wide-area networks (WANs) that require high-end routers, firewalls, optimizers, and complex configurations for consistent performance. Centralized control, following the SDN technique, in Software-Defined WAN (SD-WAN) can exploit the holistic view of the network for dynamic load-balancing, handle various types of connectivity, and reduce complexity in management [146]. Yet, economics and scalability of wider area SDN deployments as in SD-WAN are not explored well.

On the other hand, since its emergence, cloud computing has been exhibiting a continuous decline in its pricing with increased competition. Another popular trend in cloud computing is its pay-as-you-go pricing model. As people are using more cloud services and storage (in general), cloud providers are willing to provide more precise and detailed billing. These trends can make cloud platforms an ideal place to mix and match various computation and storage components to come up with a cost-effective hybrid design involving cloud resources and network (hardware) components – potentially low-level network functions like routing.

Considering the fact that complete separation between planes may not offer the best solution when it comes to scalability and flexibility [198], new architectures involving hybrid separation of control and data plane are already attracting the attention [177]. In this dissertation, we explore the economics of wider area SDN designs and characterize how sustainable SDN solutions can be at longer distances than inside of a data-center. In particular, we study how beneficial it may be to utilize cloud services for solving the increasing memory complexity of routers. We formulate the overall concept as "Cloud-Assisted Routing" (CAR), a potential solution to the scalability concerns of wider area SDN, and compare it with legacy routing. The key question we aim to answer is that *"can the partial placement of control and data plane routing functions to a remote cloud, reachable only via public Internet transit, be economically viable?"*

1.5 Major Contributions

This dissertation makes the following contributions:

- We portray the evolution of peering in Section 2.1.
- Based on stock-market and AS-level data, we quantify ISPs dependency on their networks in Section 3.1.
- We analyze the possibility of vertical-integration and future of *sugarcane* peering using inter-ISP geographic overlap and economic distance in Section 4.2.
- We identify the most frequently asked peering requirements, ISPs PoP frequencies and peering points in Section 5.1
- We develop a supporting framework for ISPs to identify the potential peers in Section 5.2 and formulate an optimization problem to select the best peering deal in Section 5.2.2.

- Presenting the architectural overview of CAR in Section 6.1, we detect the possible loop scenarios and suggest how to tackle them in Section 6.3.
- We present the prototype of CAR in Section 6.4.
- We formulate cost models for legacy routing and CAR in Section 6.5.
- We present more than 30 years of DRAM pricing data showing the trend in router memory price in Section 6.6.1.
- We perform a detailed empirical analysis and modeling of cloud storage and service prices for the last nine years in Sections 6.6.2 and 6.6.3.
- With respect to the legacy routing, we characterize the FIB size needed at a local router to attain a target cost reduction in the CAR framework in Section 6.7.
- We describe the peering influence on the sustainability of wider area SDN concepts like CAR and future routing scalability under two extreme scenarios: no peering at all vs. complete peering in Section 6.8.

1.6 Dissertation Outline

Chapter 2 discusses the related works, our motivation behind vertical peering, and presents a detailed timeline of peering evolution. In chapter 3, we examine the correlation between the stock market values as well as the revenue of each major ISP, operating in the United States, to their degree and customer cone size. In Chapter 4, we show the overlaps between access ISPs and content providers to explore the viability of a future in terms of peering between the new emerging content-dominated *sugarcane* ISPs and the healthiness of Internet economics. We then focus on how to

instrument the automation of the peer selection process and describe how *meta-peering* can be implemented by integrating some of the existing tools that are already being used for automating the BGP session establishment and monitoring purposes in Chapter 5. In Chapter 6, we propose the hybrid SDN routing architecture where the data plane partially resides in the remote cloud while discussing the necessary and sufficient conditions to avoid possible loops. We also analyze the economics of such architecture in this chapter. Finally, Chapter 7 summarizes our contributions and lists the possible future extensions.

CHAPTER 2: LITERATURE REVIEW

Cable TV is losing the battle against the over-the-top (OTT) services (i.e., Netflix, Hulu, and Youtube) as it lost 1.7 million subscribers during 2017 [147]. With no cords attached and the flexibility to stream on multiple screens using the mobile device or laptop, OTT services are gaining immense popularity and motivate more viewers to migrate to OTT services. Major TV content companies like Disney [53] and NBCUniversal are introducing their *direct-to-consumer* service as early as 2019, while DirectTV has already merged with AT&T to offer its services as an overarching bundle.

Research discussions have focused mainly on how watching videos has evolved with the emergence of OTT services [55] and how they win over traditional cable TV [48]. But, how content providers who are controlling these OTT services is going to disrupt the access ISPs, remains yet to be explored. Market capital (as of June 2018) of some content providers like Disney ($159B), Netflix ($170B) are close or even bigger than some access ISPs like Verizon ($195B), Comcast ($151B) or Charter ($70B) [149]. There are some giant CPs like Google, Facebook, who are enormous in market capital size with a huge number of end-users. These CPs want to ensure that the consumers– without being worried about exceeding data caps– get the best experience while streaming their videos, and thus sometimes intentionally downgrade the video quality to reduce the end-user data consumption. For example, AT&T and Verizon charge customers if they exceed their data caps, and to prevent this from happening, Netflix checks the user device and throttle-down the streaming quality [152].

Throttling traffic quality rejuvenates the net neutrality issue, which the networking community, along with the government regulatory body, has been consistently debating for a long time. Following the general idea of the net neutrality principle [99], Federal Communications Commis-

sion (FCC) imposed *non-discriminating treatment towards data* on carrier ISPs with the hope that it would ensure the creation and unrestricted distribution of content or services for end consumers [76]. However, FCC has no such control on CPs about how they distinguish between different carrier users and raises the main controversy against net neutrality, whether this controlling hinder the innovation and restrict the opportunity of further investment without enough incentives [195].

Though apprehensions regarding net neutrality do not impede the possibility of *vertical* integration between CPs and carrier ISPs [206], not many works have been done about detailing the architecture, economic perspective, or peering settlement among those vertical ISPs. There are a few studies [90, 196], that explored the *vertical* integration and try to argue how it may motivate access providers not to block or slow down other competitors' content from reaching to the end-users, and how this trend can strengthen CPs economically as they do not have to pay the access providers while enjoying more control on end-to-end delivery network.

In general, carrier ISPs' routing business involves myriad items in the cost model that impact the price of network services [107]– geo-location, traffic amount, intellectual property or software licensing, and infrastructure or operational expenses required in the process to name a few. But, the primary contributing factor remains the unit cost of router memory. Access time in Static Random-Access Memory (SRAM) is minimal ($\sim$ 4 ns) compared to DRAM ($\sim$ 40 ns), which makes SRAM a perfect choice for router memory. But, due to SRAMs power-hungriness and overheated nature, its usage is stringent (few Megabytes), and additional DRAMs (CISCO 4400 Series has 2-8 Gigabytes) [19] are being introduced to store routing tables. Again, on-chip memory (CPU cache or FPGA block RAM) usage has not increased as well because of being very expensive with respect to a conventional off-chip memory (DRAM) [208]. As a result, research in this area mainly focuses on developing memory-management algorithms (e.g., SMALTA [193], FIFA [132]) to optimize IP lookup time by aggregating FIB [212].

Multibit-trie architectures such as Tree Bitmap [87] have attained much popularity in high-end routers (e.g., Cisco CRS-1 Multishelf System [197]) because of its faster updates and searching capability. But, this approach requires more memory, and thus other tree-based architectures (e.g., FlashTrie [52], PopTrie [46]) have been explored to overcome the shortcomings. Another effort was taken by Rtvri et al. [171] that demonstrated whether it was possible to guarantee IP lookup performance by squeezing the existing router hardware memory to facilitate the ever-expanding FIB table.

Network economics aim to explore strategies to minimize the Capital Expenditures (CapEx) and Operational Expenditures (OpEx) to maximize ISPs' profit. Ma et al. have investigated a game-theoretic approach [134] to achieve an efficient, fair, and optimal routing among a group of profit-sharing ISPs. A Cost-Aware (CoA) caching [43] scheme has also been proposed to see the feasibility of cost minimization. This approach contradicts the popular cache algorithms whose fundamental intentions are attaining maximum hit-ratio, and instead, emphasizes specifically to reduce cost and offers economic incentives.

Keeping technical complexity in FIB memory management aside, from a pure business perspective alone, ISPs tend to form bilateral peering settlements or customer-provider relationships without considering the global view when it comes to maximizing their profit. These service level agreements (SLAs) sometimes result in inefficient routing, and thus, have an adverse impact on the overall Internet ecosystem and ultimately less aggregated market profit. We explore the literature to see how this peering business has evolved and became more complex, later on, in Section 2.1.

Although *peering is a handshake*, finding the right person is difficult. To ease the process, network admins typically meet other ISPs' representatives personally in informal events hosted by North American Network Operators' Group (NANOG), MENOG (Middle East), RIPE, Global Peering Forum, or in CEE Peering Days [188]. After a discussion on the traffic volume to see if peering

would generate enough savings for both the entities, nondisclosure agreements (NDA), peering policies are negotiated. However, sometimes conflicts happen between the ISPs either due to the lack of prior knowledge about other ISPs traffic amount before peering or as a part of a strategic tussle between the participating entities, which may lead to disputes or incidents like de-peering. For instance, Cogent de-peered Level3 (in the year 2003), AOL (2002), and Telia (2008) due to imbalanced traffic ratio, and Sprint (2008) for not respecting the exchange criteria [50].

Deficiency in the number of peer selection algorithm case-studies implies less preparation for ISP admins about the opponent ISP's traffic behavior. Each ISP has complete knowledge of its traffic matrices and router-level network topology, but it does not have any detailed information about its counterpart. It can have only a rough estimation about which ISP is sending the maximum amount of traffic and attracts most of its customers. But, the fundamental question here is whether they can estimate each-others traffic? Without proper estimation, it is difficult for ISPs to commit a long-term peering relationship. To avoid any future disputes, ISPs undergo a "trial peering", which is temporary, for a period of week or month to determine the exchanged traffic amount before provisioning the peering session [190].

Optimal peer selection is a hard problem. Game-theoretic approaches mostly focus on economic analysis by considering both routing and congestion costs [176] to study the capacity and pricing decisions made by the service providers [181]. Earlier works [116, 135] mainly focused on either formulating an optimal peering problem to determine the maximum peering points and their strategic placement on the globe, or on presenting a negotiation-based platform where adjacent ISPs could jointly determine the routing path of their exchanged traffic while disclosing a limited amount of preference information to other competitors. The goal of this researches was the same, minimizing the interconnection cost without compromising the service quality. Still, those were not automation efforts rather mathematical interpretations and steps towards understanding Internet-wide negotiation mechanisms.

Despite the emphasis of most network economics analysis are limited to either peering business relationship between multiple ISPs or how to shape the traffic to conciliate individual ISP budget for reducing the cost, Motiwala et al. [150] presented a cost model and considered the total volume of traffic flow including the cost of carrying them through the network. It offered the operators an opportunity for traffic engineering in path selection by identifying the most expensive flows and route them through a less-utilized and more economical alternative transit. This model is the closest to what we are proposing. Indeed, they classified the main cost contributors into two categories, namely *Interconnect costs* that comprise transit fees, port costs, or some fixed costs and *Backhaul costs*, which represented circuit, capital, and operational costs altogether. We are interested in one of the components of *Backhaul costs*, i.e., router cost, to be precise.

In this regard, recent observations [28] on the cloud being cheaper, closer, and higher quality (cloud challenges are reducing) attract the networking community for a longer-term as they continue to explore more novel mechanisms to solve router-level problems. Cloud service providers are not only reducing the price but also are investing more to offer newer feature sets and innovative services by developing their high-computation infrastructure that is capable of supporting a wide range of new applications. Features like load-balancing and auto-scaling have already become a common practice by major providers like Amazon, Microsoft, or VMWare. Vendors are shifting towards *per-second* from *per-hour billing* schemes to provide more accurate and detailed billing, which benefits enterprises with much flexibility [32]. Offerings of additional discounts (up to 75%), albeit the requirement of a committed usage over one to three years, make the cloud a lucrative choice to include while designing a cost-effective architecture.

Since the emergence of cloud computing, Amazon leads the industry with Amazon Web Services (AWS), while other big companies like Google and Microsoft have branded their services as Google Cloud Platform and Microsoft Azure. All three provide file *storage capability* (Simple Storage Service (S3) by Amazon, Google Drive by Google, OneDrive by Microsoft) as well as

facilitate services like *Software as a Service (SaaS)*, *Platform as a Service (PaaS)* or *Infrastructure as a Service (IaaS)* and charge users accordingly. Being the dominant player in the cloud computing market, Amazon sets the tune by continuously slashing cloud service prices by 16% to 28% (varies by region and services) [169, 54], which, in turn, compels other competitors to follow the trend.

Customer like *Netflix* has already migrated to AWS to handle its 1000x growth in monthly streaming hours [23]. All of Netflix's video contents, business logic, data analysis, and service availability are now hosted on Amazon. *Airbnb*, *Adobe*, is also housed in AWS and is using Elastic Compute Cloud (EC2) with other services for load balancing, simplified auto-scaling, or efficient supervising purposes [35, 34]. Similarly, Google Cloud customers like *Snapchat* uses storage service for storing images, compute engine for image processing, and BigQuery for their data analysis [200]. *Spotify* also uses BigQuery to analyze users' listening patterns to provide a better experience while *Best Buy* is using Google App Engine to reduce the maintenance overhead for agile development and scalability [36].

Armbrust et al. [45] discuss the elasticity of the cloud system by showing an example of a pay-per-use pricing scheme for the cloud. In business, it is beneficial to have the ability to add or remove resources at any time instantaneously. Methods of cloud pricing are always an issue between the user and the cloud system [199]. But, earlier works primarily accentuate the computational expenses caused by the distributed systems on the cloud instead of evaluating the storage price, and thus, effectively overlooked the affordability discussion.

In addition to analyzing the router hardware cost in our model, we need to understand the future cost-effectiveness of this trend of *delegation to the cloud*. Our study investigates this issue and, specifically, we look at the consequences of delegating routing (network layer) functionality, partially, to cloud with a focus on FIB table caching and its causation of data packet delegation to

cloud. In this regard, we also consider the transit cost incurred due to the packets delegated towards the cloud using physical infrastructure. To augment our cost model, we use the data transmission cost model based on the quality of service, proposed by Fishburn [93].

Our work differs from earlier ones as we:

- show the service coverage area of different ISPs and assess whether these ISPs will be benefited from a possible vertical merger. Highlighting on the nature of peering, we aim to provide guidelines for the newly formed *sugarcane* ISPs.
- focus primarily on automating the peer selection process and suggest the possible PoP locations in a sorted manner based on ISP's own pre-defined sorting criteria. ISPs can decide whether it wants to offload as much traffic as it can without even considering the outbound vs. inbound traffic ratio, or it can prioritize ISP with lower traffic difference. In our opinion, this will help ISPs to collaborate with each-other to provision seamless connectivity to the end-users.

2.1 Evolution of Peering

Figure 2.1 demonstrates the evolution of peering from a top-level view, and Table 2.1 presents an *approximate* timeline of each stage as they took place. During the mid-'90s [202], the legacy model of ISP peering prevailed, where CPs and access ISPs were horizontally separated and had to buy transit service from transit ISPs. CDNs gained momentum in around 2002 [213], and companies started relying on CDN services to deliver their content faster to the end-users after observing the significant market exposure received by *Akamai*. Moving forward to 2009 [17], with large enough fiber networks of their own, CPs started to by-pass both transit and CDN providers. They took the initiative of establishing direct peering with access ISPs following the "Donut Peering" model.

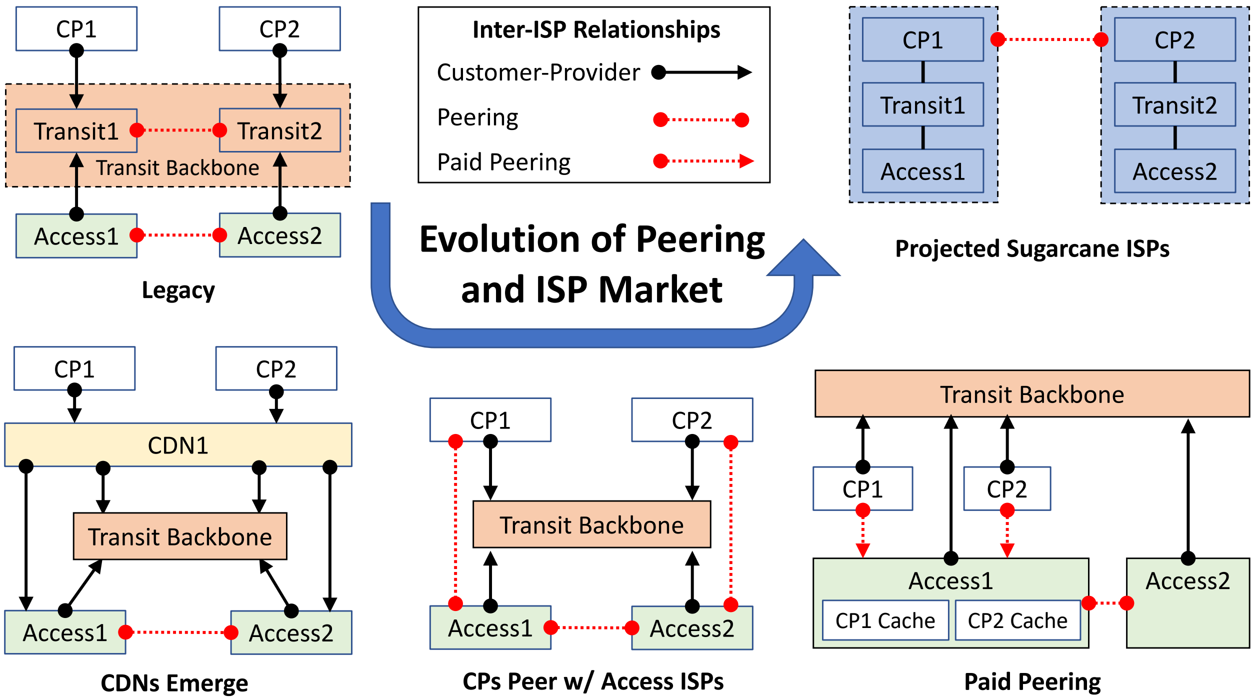

Figure 2.1: Architectural overview of peering evolution in the Internet

Table 2.1: Peering evolution timeline

Year	Peering type	Description
1996	Legacy	Horizontal (separate) ISPs; Peering among transit and access ISPs.
2002	CDN Emerge	CDNs act as intermediary for CPs; do finer granularity optimization of content and money flow.
2009	Donut Peering	CPs have large enough fiber network to peer with access ISPs and circumvent CDNs.
2010	Paid Peering	CPs put caches inside access ISPs and engaging paid peering with them; Transit ISPs are becoming less relevant and critical to the money flow.
20??	Sugarcane Peering	CPs do not have to pay transit ISPs as they are directly engaging with access ISPs; CPs will likely to grow their own backbone and even access.

Some times later [202], it was noticeable that the traffic ratio between CPs and access ISPs are not even. For CPs, it was more beneficial to put caches directly at the access ISPs' end [70] and

compensate access ISPs for their asymmetric traffic. To complement this, access ISPs introduced the term "Paid Peering" to charge CPs as the generic peering policy was not enough.

As CPs continue earning a larger share of the revenue, they will expand their fiber footprints, and will gradually minimize their dependency on transit ISPs. Finally, it is likely that CPs will vertically integrate with access ISPs forming a *sugarcane* structure to improve end-to-end optimizations of their content delivery performance. Once transit ISPs effectively disappear, either access ISPs survive or not, peering among access ISPs *only*, will not be sufficient to attain low end-to-end latencies. Similar to how transit ISPs have been traditionally peering among each other to reduce the data transmission costs, the new *sugarcane* ISPs, dominated by content, will have to peer with each other to attain strong end-to-end performance in "*sugarcane* peering". This means one *sugarcane* ISP could deliver its content using another *sugarcane* ISP, effectively, without paying the other ISPs' carrier networks.

CHAPTER 3: ON CORRELATING ISP TOPOLOGIES TO THEIR BUSINESSES

To quantify the ISPs' dependency on their networks, we introduce a new metric, *Network Dependency Index (NDI)*. Utilizing about a decade of data, we quantify *NDI* of various types of ISPs by comparing their stock market value to their AS-level degree and customer cone size. We also examine the AS relationships against the stock market performance of major ISPs. This chapter scratches the answers to the key questions of: *a)* how much dependent the ISP market will be on the network structure, *b)* what will be the network characteristics of a successful ISP, and *c)* will there be enough incentives for ISPs to invest in their network infrastructure as content is getting increasingly more weight in value.

The rest of the chapter is organized as follows: Section 3.1 discusses ISPs' topological data, their stock market performances, as well as our data collection methodology, and the definition of the metrics we introduced. Section 3.2 explains the correlations, NDIs of ISPs, and extracts insights from the results.

3.1 Methodology

The heterogeneity of AS relationships and the types of ISP businesses bring up different perspectives of understanding the ISP market. We, therefore, consider three prominent AS relationships (i.e., *Peering*, *Customer*, *Provider*) and six types of ISPs (i.e., *Content*, *Access*, *Transit*, *Transit & Access*, *CDN*, and *IXP*) here. As mentioned in the earlier chapter, we have considered each ISP as a single node consisting of multiple ASes and used CAIDA's classification to classify them.

Most of the content of this chapter previously appeared in proceedings [79].

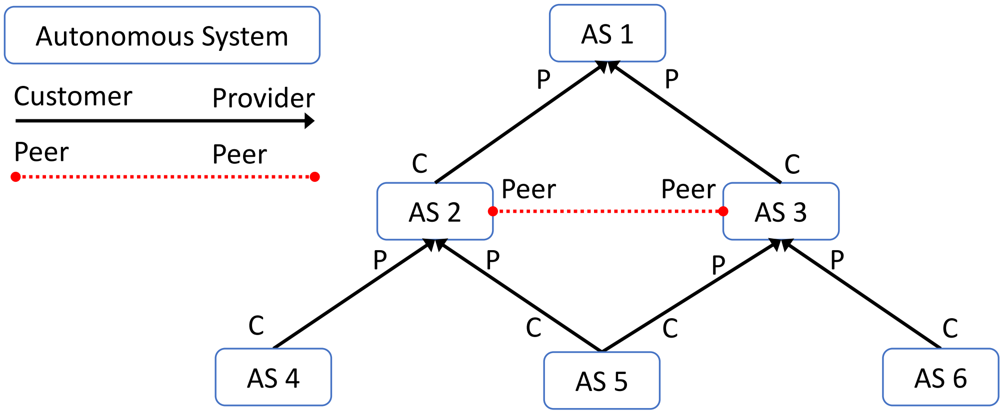

Figure 3.1: AS relationships [16]

Using ISPs' primary business areas, we identified which type an AS belongs to and clustered them accordingly. We chose six ISP types based on the terminology used in discussions on the NANOG mailing list and Norton's white papers [158].

Based on the definitions of inter-AS business relationships we previously discussed, Figure 3.1 shows the different types of AS relationships: *peering* (between AS 2 and AS 3), *customer* (AS 4 is a customer of AS 2), and *provider* (AS 1 is the provider of both AS 2 and AS 3). AS relationships determine the importance of the ISPs owning them. In particular, we consider two AS-level network properties of an ISP: *degree* and *customer cone*. First, the *degree* of an ISP is defined as the number of direct relationships it has with other ASes. Second, the *customer cone* [1] of an ISP A is the sum of A's direct subscribers' count and customer cones of all the downstream customer ASes of A. As shown in Figure 3.2, customer cone does not include a peer ISP's customers. We calculate the number of users in a particular AS by counting the number of IP addresses in the subnets provided by CAIDA [15].

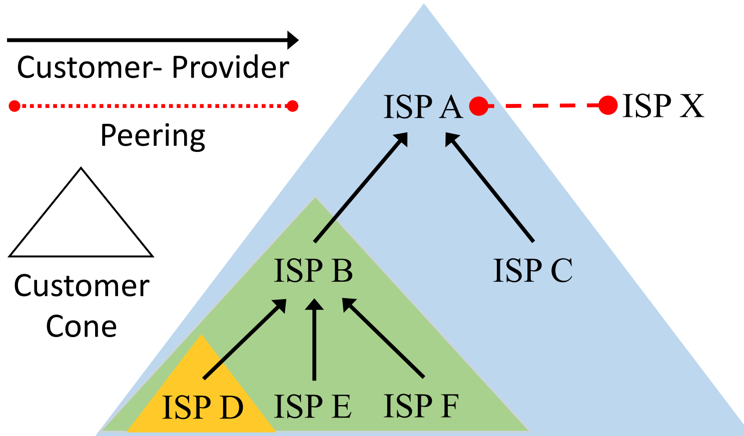

Figure 3.2: Customer Cone of an AS [1]

3.1.1 Dataset

We used three datasets to collect the topological and economic metrics of ISPs. The AS list for each ISP is obtained from CAIDA's *AS Rank* [15]. For this purpose, in total, we investigated 32 major ISPs operating in the United States and providing six different types of services, as listed in Table 3.1. ISPs offer a variety of services such as wired access, wireless access, transit and wireless access together; but, we base our selection on their salient service. For instance, Alphabet Inc. owns both Google and Youtube. Equinix has dedicated businesses in the US, Europe, Asia Pacific, and so on. Each of these franchises maintains its separate set of ASes. For our study, we consider only one to resolve these cases and select the top-ranked AS from that service provider as a sample for showing its information in the table.

We also collected a second dataset from CAIDA on *AS relationships* [16], that are visible in BGP announcements and derived via methodologies described in [101]. This particular dataset contains inter-AS relationships of ISPs and their types. For our purpose, we fetched the data from January 2007 to May 2017. We extracted the number of peers and providers an ISP has, from the dataset.

Table 3.1: ISPs grouped by service types

(sorted by Market Capital (from NASDAQ, * from NYSE) on June. 2017,
AS dataset from CAIDA 06-01-2016 [15])

Type	Name	Stock Name	AS Count	AS Degree	Top AS Number	AS Rank	Provider-Customer-Peer-Sibling Count
Transit & Access	AT&T	T*	12	255,273,728	7018	21	0-2,348-45-3
	Verizon Business Inc.	VZ	30	416,381,184	701	12	0-1,321-26-12
	Time Warner Cable	TWC*	24	45,897,216	7843	47	4-22-30-10
	Sprint	S*	32	330,382,848	1239	19	0-521-47-3
	Internet Initiative Japan Inc.	IIJI	2	36,473,856	2497	86	2-158-138-0
CDN	Akamai Tech. Inc.	AKAM	17	13,843,200	32787	141	4-210-89-0
	Rackspace Hosting Inc.	RAX*	7	1,853,952	12200	2781	4-4-63-3
	Limelight Networks Inc.	LLNW	7	17,302,272	22822	302	12-16-107-0
	Internap Net. Serv. Corp.	INAP	31	5,117,696	13789	181	9-131-0-2
Content	Google Inc.(Alphabet Inc.)	GOOGL	5	1,529,088	15169	1704	3-9-192-1
	Microsoft Corp	MSFT	28	20,050,432	8075	1491	17-10-188-7
	Amazon.com Inc.	AMZN	2	16,422,144	16509	3196	22-3-170-1
	Facebook Inc.	FB	2	63,488	32934	3919	16-2-172-0
	IBM Corp	IBM*	1	17,408	2140	5878	2-1-1-0
	Yahoo! Inc.	YHOO	5	593,664	10310	626	5-38-134-2
	Yelp! Inc.	YELP*	2	3,072	33445	5903	4-1-21-1
Access	Comcast Cable Com.	CMCSA	38	124,974,848	7922	28	2-111-51-31
	Charter Communications	CHTR	9	10,285,568	20115	94	5-307-29-4
	Liberty Global Plc	LBTYA	2	31,805,184	6830	30	0-524-247-1
	Cablevision Systems	CVC*	1	6,864,128	6128	79	4-431-33-0
	Frontier Communications	FTR	6	12,677,888	5650	204	4-106-69-4
	General Communications	GNCMA	1	437,504	8047	1303	3-13-13-0
	Qwest Comm. (CenturyLink)	CTW*	13	281,121,280	209	15	0-1,686-62-6
	WideOpenWest Inc	WOW*	9	1,407,232	12083	568	7-36-30-3
Transit	Level 3 Communications	LVLT*	32	793,745,920	3356	1	0-4,361-75-8
	Zayo Group Holdings, Inc.	ZAYO*	2	120,272,640	6461	13	0-1,416-150-1
	Cogent Communications	CCOI	16	616,423,936	174	2	0-4,826-71-2
	Tinet (GTT Comm.)	GTT*	3	565,089,024	3257	5	0-1,313-63-0
IXP	Equinix	EQIX	27	11,520	14609	7673	3-0-25-0
	Telx (Digital Realty)	DLR*	9	263,424	36086	451	7-53-25-2
	CyrusOne Inc	CONE	6	1,683,456	62	649	8-29-2-2
	Any2 (CoreSite Realty Corp.)	COR*	5	104,704	2734	1740	9-9-20-0

In doing so, we accumulated the number of every customer-provider and peer-to-peer links for each ISP. Therefore, in Table 3.1, continuing with Google's example, we mark AS-15169 as the top AS for Google and put its global ranking 1704, considering this particular AS's position as the highest-ranked among all five ASes that Google owns. The numbers, 3-9-192-1, in the last

column, consecutively represents the count of providers, customers, peers, and sibling ISPs Google is connected with.

Stock market performance tends to be volatile over time, and a snapshot in time t can be different from another time t+n. In our analysis, stock market data is harvested from End of Day US Stock Prices from Quandl [2] for the dates of the AS relationship data recorded. CAIDA dataset includes data from 1/1/2007 to 5/1/2017 (mostly once per month). We queried Quandl to get the financial data for every available date during that period. If data was provided on the weekend, we took the stock price for the Friday before that weekend. There were some dates, in the AS relationships data provided by CAIDA, that we could not obtain the stock price for. In total, we queried about 120 timestamps for each ISP.

3.1.2 Network Dependency Index (NDI)

We introduce *Network Dependency Index* (NDI) as the *correlation* of the inter-AS network structure, size and/or position of an ISP to its economic status. In particular, NDI is computed as the correlation coefficient of the underlying network to the economic aspects of an ISP business by comparing its stock market price and value to its AS-level degree or customer cone size. In other words, NDI of an ISP quantifies the dependency of that ISP's economic status to its underlying network properties. We use the ISP's stock market value to express its financial condition and its AS-level degree or customer cone size to express its network's status.

We calculate NDI of an ISP as the Pearson correlation coefficient [56] between two separate time series: *i)* the stock market value of the ISP over time, and *ii)* its AS-level degree (producing NDI-d) or the customer cone size (to generate NDI-c). This value expresses ISP's network and economic fluctuations. We choose to employ the AS-level network properties of the ISP for simplicity and initial observations. It is worthy to note that NDI could also be calculated in other ways with more

temporal and spatial granularity. For instance, router-level topologies of the ISPs could be used to attain a more accurate NDI with better spatial granularity, but obtaining such realistic information is difficult, as ISPs keep them confidential.

A negative NDI for an ISP implies that ISP's small network does not play much of an impactful role in its profitability, while a positive NDI indicates that a more extensive network appears to benefit from its business. Zero NDI means that ISP's business is ambivalent to its network size and associated properties. Note that NDI does not express the causation of why an ISPs business is (un)successful, but merely suggests one of the potentially several arguments for why.

3.2 Results

Using both Quandl and Yahoo! Finance[4] dataset along with the market capitals (i.e., the total market value of the shares outstanding of a publicly-traded company and adjusted closing stock values) are used to compare the companies and their prices. We also used the market capacity as a sorting criterion for the ISPs.

Figure 3.3 shows the market capitals of the ISPs in June 2017 (using Yahoo! Finance) except for Rackspace, which was acquired by Apollo Global Management in Nov 2016 [25], Time Warner Cable, that merged with the parent in May 2016 [26], and Cablevision, due to its acquisition in June 2016 [21]. The list of ISPs are clustered according to their types and sorted within each group by their market capitals. This figure elaborates previously discussed Figure 1.2 and exposes the apparent dominance of CPs as they are strikingly ahead in market capital. We shall discuss this ascendancy, in detail, in the following chapters.

[4] https://finance.yahoo.com

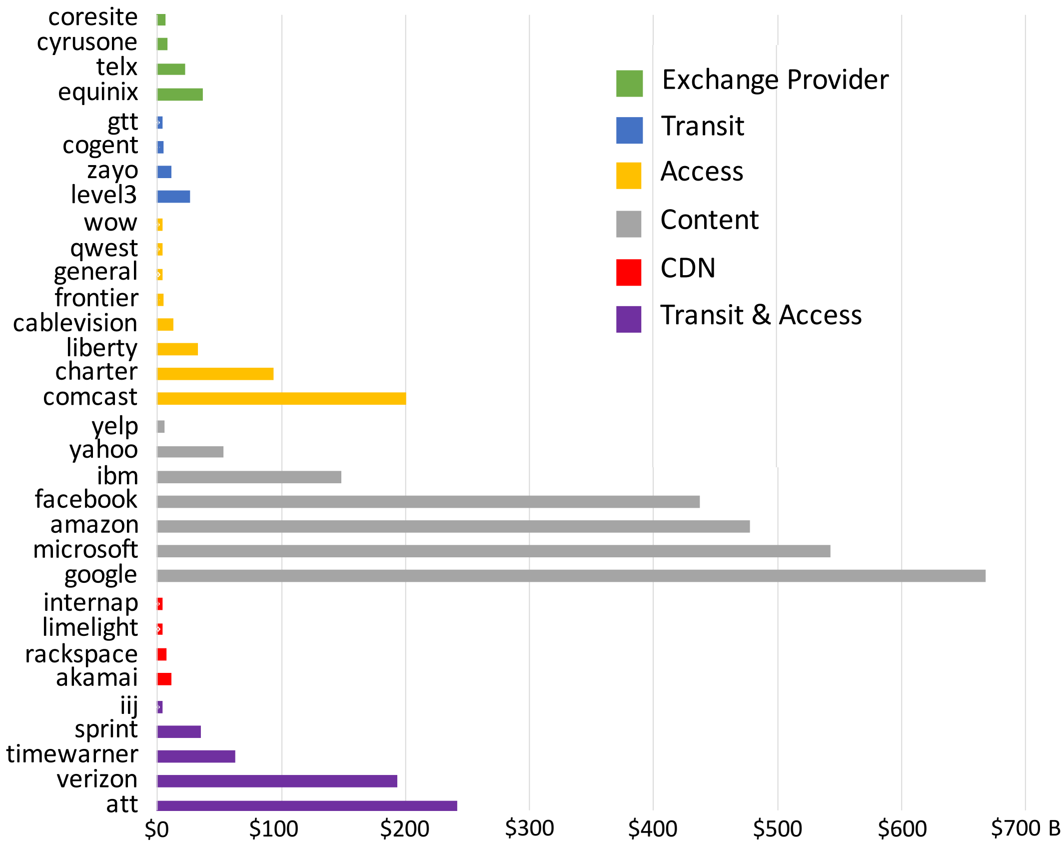

Figure 3.3: Market capital (in Billions) comparison of 32 ISPs

3.2.1 NDI Based on Degree (NDI-d)

Figure 3.4 shows *NDI-d*, the correlation between the degree and the stock market price of each ISP sorted by market capital of the ISP along with circle points representing the average *NDI-d* for that specific group of ISPs. As a first observation, ISPs that provide similar services mostly show close *NDI-d*s with strong confidence except for the transit & access ISP group. Three of the transit & access ISPs have negative *NDI-d* while others have positive *NDI-d*. ISPs showing negative *NDI-d* are *AT&T*, *Verizon*, and *Sprint*, which are among the largest cellular operators in the US. Their negative *NDI-d* could mean, *(i)* they perform more transit themselves rather than paying other transit ISPs, and/or *(ii)* they have a more diversified business beyond the ISP market.

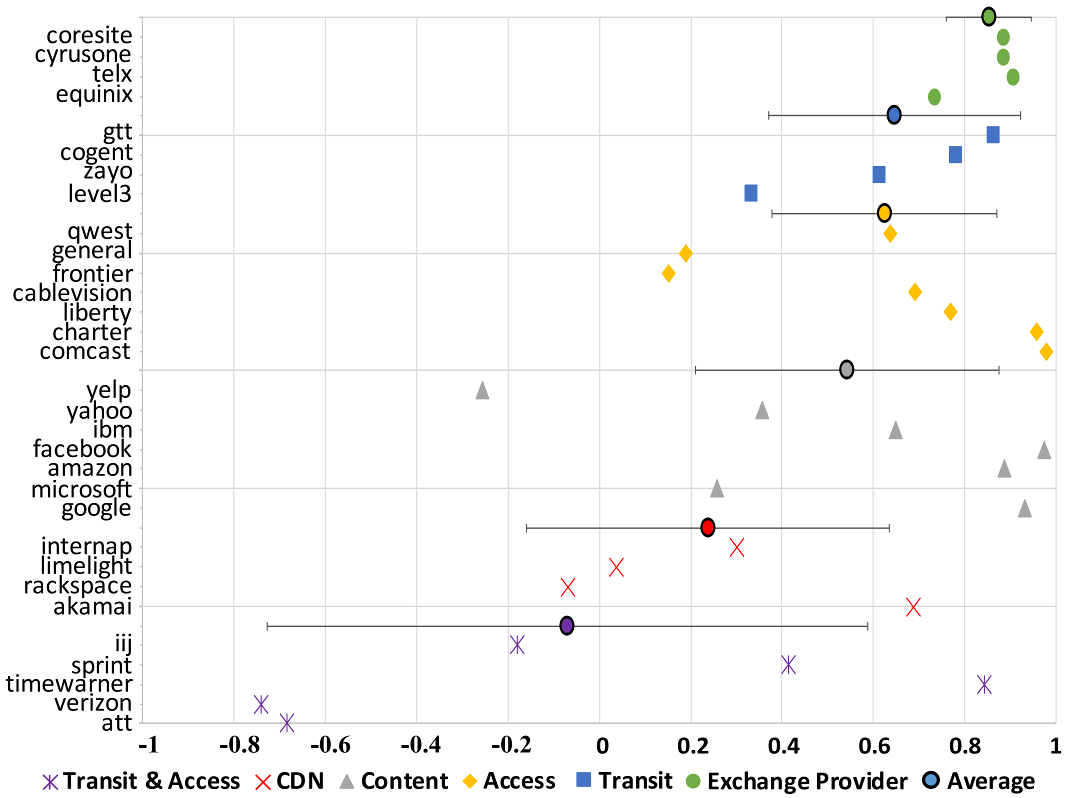

Figure 3.4: Correlation to ISP stock market price (90% CI) for number of *connections* of an ISP

These ISPs do not seem to get directly affected by their AS degree since, maybe, they mainly focus on providing more cellular coverage and are involved in maintaining their existing network for continued support.

As mentioned earlier, with substantial revenues, the CPs are growing as the leading players [84, 158], and the way these ISPs peer, or the peering policies they implement could explain why we are observing a higher positive NDI-d phenomenon for CPs. We shall see the proportion of ISPs peering policies in the next chapter, but, for now, CPs adhere to more open peering policies compared to the access and transit providers. Open interactions with other players allow more exposure and easier access to their content – a business advantage for the CPs, which we can notice from our data. Google, Facebook, and Amazon have high positive NDI-d while other CPs

are also on the positive side. As an example of having a sizable AS degree (due to more peering) benefiting ISP business, Facebook's peer count has increased from 79 to 180 between the year 2012 to 2017 (the number of providers also risen to 19 from 11) while operating only three ASes.

CDNs show smaller *NDI-d*, and *Rackspace* is on the negative side. CDNs are well-connected multi-homed ASes and rely on transit providers to carry the content to end-users. Although it is not significant, the average *NDI-d* of CDNs is lower than CP's average *NDI-d*, indicating that CPs take extra steps to increase their degrees to reduce the delays to end-users.

Tccess & access providers with smaller revenues may naturally act more biased on peering due to their interest in the mainline business of provisioning a higher quality network rather than mere exposure and trade of content. *Time Warner* shows positive *NDI-d*; this is not surprising because of its earlier nature of being a CP, now entered into the transit & access business.

IXPs showing higher *NDI-d* is self-explanatory due to their nature of acting as the common point for exchanging traffic among other ISPs. For instance, Equinix has increased its outreach to more than 30 global markets, supporting more than 1,000 networks globally [40]. This is a rise compared to their distribution of 19 facilities in 17 metro areas back in 2013 [74]. The higher the number of associated ISPs an IXP has, the higher the chance of attracting a new ISP to its facilities.

Comparing the average *NDI-d*s of transit and access providers exposes the fact that ISPs involved in more transit business yield a higher *NDI-d*. It verifies the intuition that transit ISP's stock market price is highly dependent on the number of connections they have with other ISPs. Higher degree ASes are crucial for a successful transit as this enables shorter paths to end-users and a more natural reach to other potential customer ISPs needing transit services. Even though we can make a similar analogy for the access providers, but, the existence of some access ISPs with lower *NDI-d*s suggests otherwise. That, the performance of the access network has more weight than transit network performance, e.g., transit ASes need to be well connected to the rest of the ASes.

For instance, *AT&T*, *Sprint*, and *Verizon* are known to have strong wireless access networks while owning large transit networks as well. Yet, we observe their *NDI-d*s to be negative, suggesting that a transit network, alone, may not make a substantial impact in determining business success. This perception aligns with the recent trends of weakening transit services as a profitable business. Another reason could be that access ISPs serve a specific region and do not establish a lot of AS neighboring relationships, which explains *Frontier* and *General*.

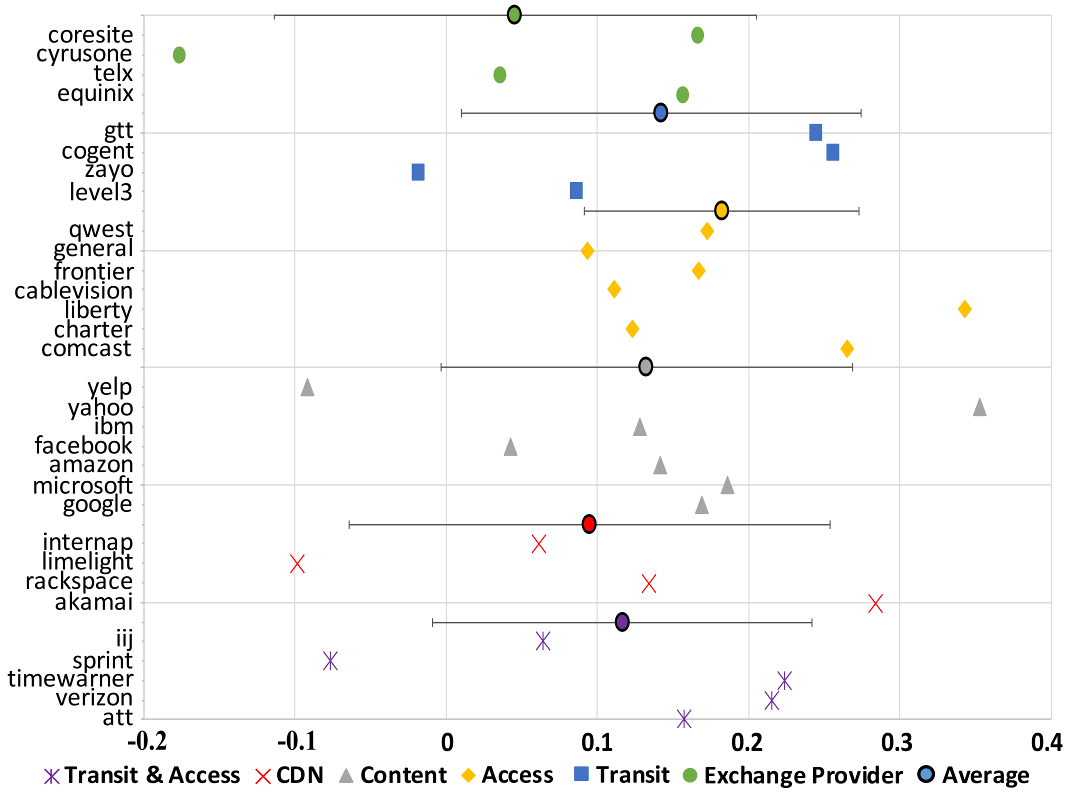

Figure 3.5: Correlation to ISP stock market price (90% CI) for number of *customers* of an ISP

3.2.2 NDI Based on Customer Cone (NDI-c)

Figure 3.5 displays *NDI-c*, the correlation of the customer cone and the stock market price of the ISPs along with the circle points that represent the average *NDI-c* for that particular ISP group. Most of the *NDI-c* values are on the positive side since the performance of an ISP business positively correlates with the total number of end-users it serves.

Low positive *NDI-c* between the customer cone and the stock market prices verifies the intuition that when the number of customers increases, the demand for ISP's stock increases. Even so, the *NDI-c*s of CPs and CDNs, here, are on the low positive side. This can be because of CPs' typical behavior of not willing to manage the delivery infrastructure by themselves, instead of depending on other carrier ISPs for ensuring the best service quality of their contents. As a result, the customer cone of a CP can be small even though they have a big list of subscribers. However, this trend is gradually changing, as we discussed previously. *Yelp!* shows negative values for both *NDI-d* and *NDI-c*. According to our dataset, they seem to have very small connectivity compared to others. They only have 20 peers and seven providers. But, they recently increased the count of ASes they own, from two to three, which means, with an increasing number of customers, they are expanding their networks too. Furthermore, *Limelight* has a noticeably lower *NDI-c* among the CDNs, which is likely caused by its focus on data center management [110].

Examining the average *NDI-c* for all ISP categories, we observe, only access ISPs demonstrate all positive *NDI-c* values, while for IXP's case, the numbers are approximately zero. Since IXPs do not engage with end-users, rather deal with other ISPs directly, customer cone size impacts them less compared to access ISPs who are serving the last-mile customers.

ISPs offering a similar type of service remain closer to each other, as seen in the *NDI-d* case, but the distribution of the ISPs across *NDI-c* is less skewed. Moreover, the values are less scattered

in *NDI-c* as opposed to *NDI-d*. Overlapping confidence intervals prevent us from judging whether these ISPs' operations are statistically different or not. Nonetheless, based on *NDI-d* values, it is evident that the IXP's business model (in network connectivity) significantly differs from that of CDN or transit & access providers.

3.2.3 *Stock Value Analysis Based on Providers Count*

Another dimension to explore is the effect of ISP's providers' count on its stock value. We should mention here that the results for peer-to-peer links we show in this study should be considered as lower bounds on the actual number of peer-to-peer connections since the CAIDA dataset we use consists of only a portion of the actual links [84].

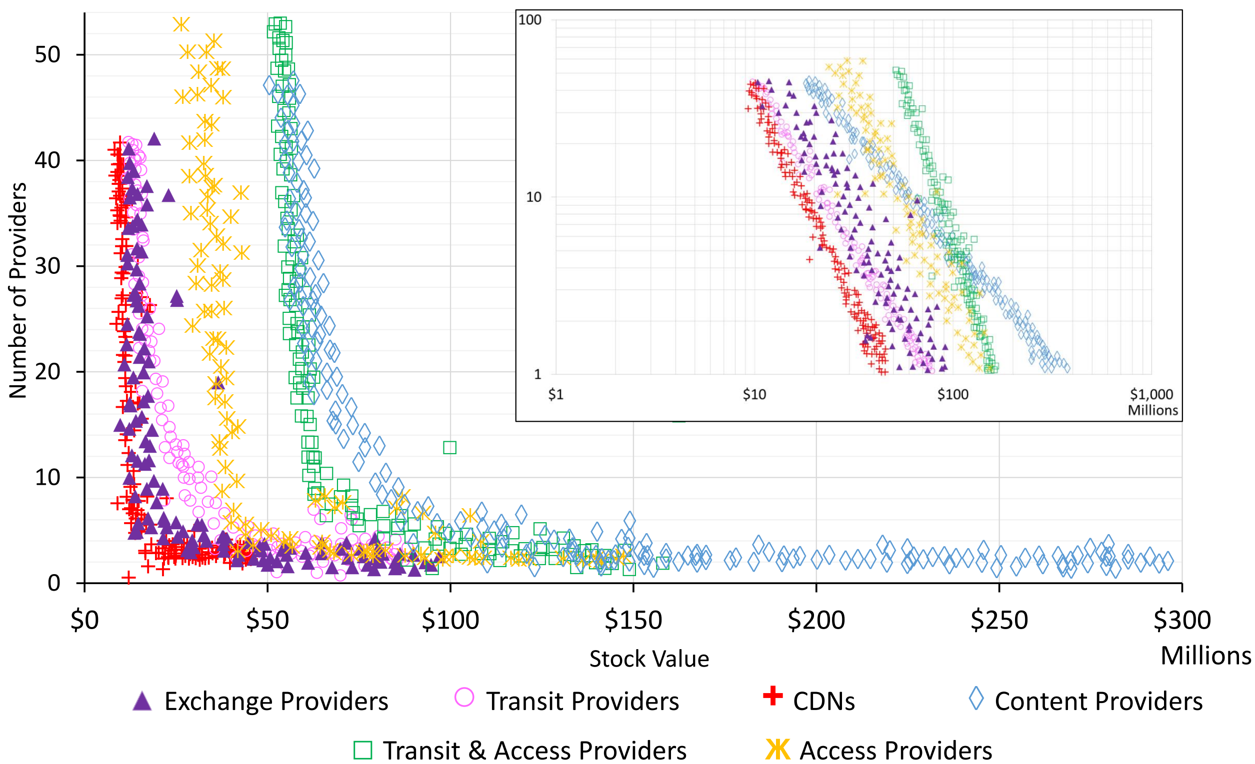

Figure 3.6: The number of providers an ISP has versus its stock value (with log-log plot)

The NDI we calculated based on ISPs provider was somewhat unclear and challenging to interpret. We assume that the reason for producing such intricate values for ISPs is because we have fewer providers' information in our data than these ISPs usually have. In further analysis of the data, we have faced a striking representation of the provider-stock value relationship for the ISP groups. When we plot the number of providers versus stock values for each ISP group, the plots appear as power-law-like heavy-tailed distributions. In Figure 3.6, different from Figures 3.4 and Figure 3.5, we plot the data points of providers count and stock value pairs for each date we have from the dataset.

Figure 3.6 illustrates the relationship between the number of providers and the stock prices of ISPs. The data points for higher stock values are on the right side of the x-axis of the graph. The inside figure presents the same data on a log-log scale. It shows that there is an exponential decrease in the number of providers for all ISP types as the stock value is increasing. When the number of providers is high, the market value of that ISP is small; but when the number of providers decreases, the market value increases non-linearly.

Since these data points are sorted from the past to a more recent date, we see that all ISPs are trying to reduce the number of providers from which they receive services. As a result, they end up displaying a non-linear increase in their stock values. Interestingly, the number of providers for CPs does not diminish at the same speed compared to other ISP groups. Earlier, the number of providers for CPs is similar to the transit and access ISPs; however, CPs reduce their providers count at a slower pace. This episode of events might be due to the distinctive arrangements of IPSs in the business spectrum. While the speed of infrastructural growth is a must for transit or access ISP's existence, CPs can be more reluctant in this regard as their fundamental quest is not laying the fiber cable. But, unlike access ISPs, CPs target much broader areas (i.e., the distance around the globe) for their infrastructure expansion, and to serve their unique purposes, they often establish new data centers.

These results show that the stock value of an ISP is highly dependent on the number of providers the ISP has. The trend of ISPs shows that they are trying to get rid of customer-provider contracts over time and prefer establishing a peering relationship. The fewer providers an ISP has, the better the stock market performance it exhibits. This follows the intuition that an ISP with more providers is a smaller ISP in terms of market performance and needs purchasing services from bigger ISPs. As an ISP gets stronger in business value, it can persuade other ISPs to form peering relationships and thus reduces the number of its providers.

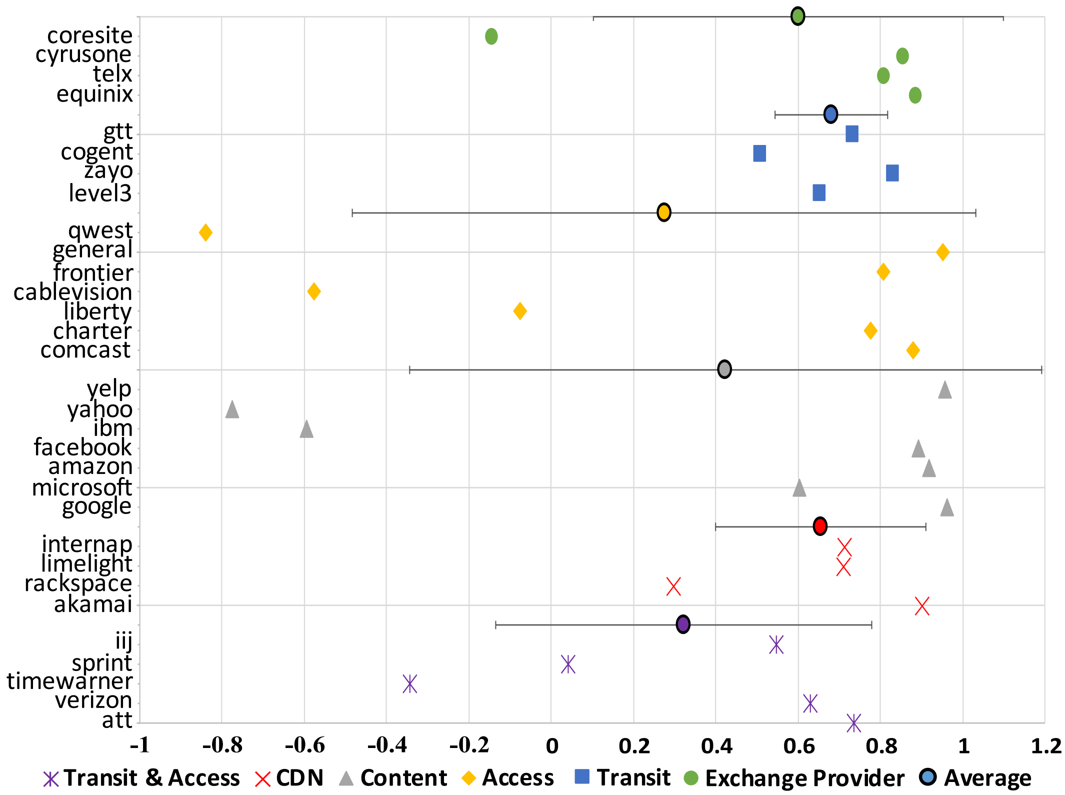

Figure 3.7: Correlation to ISP revenue (with 90% Confidence Interval) for the number of *connections* an ISP has

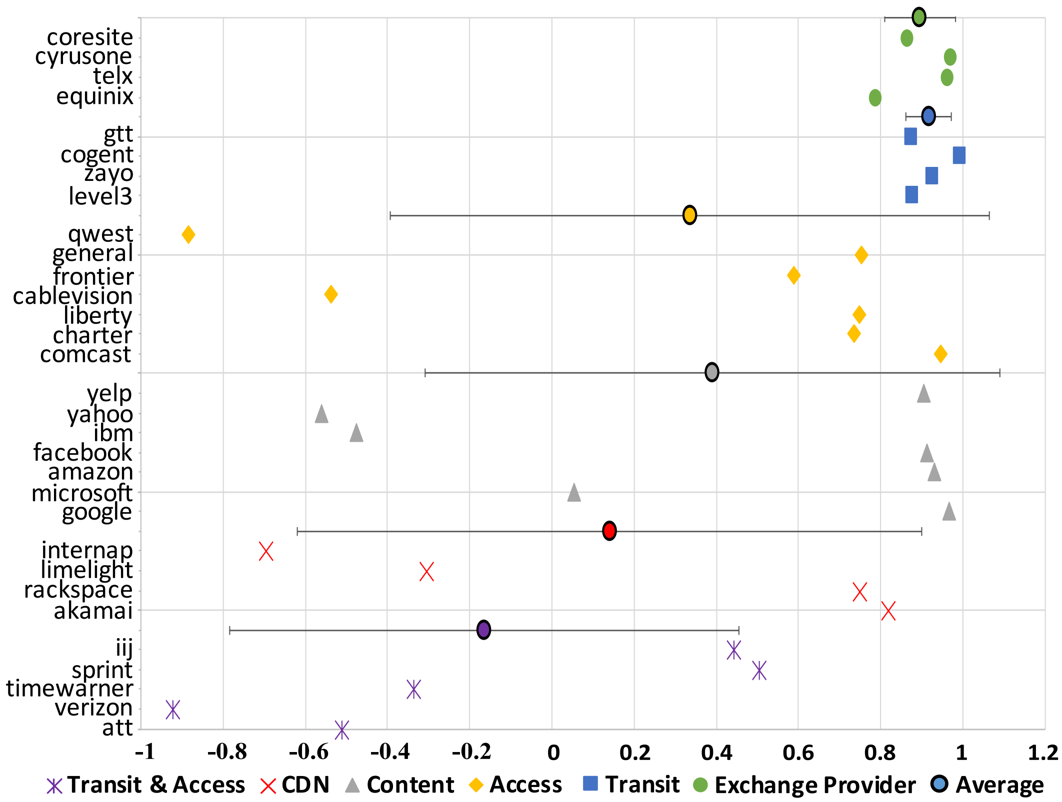

Figure 3.8: Correlation to ISP revenue (with 90% Confidence Interval) for the number of *customers* of an ISP

3.2.4 NDI Using ISP Revenue

We extend our work by calculating the NDI using ISP's revenue as an alternative economic status metric for the stock market price. Revenue numbers generally reflect the financial status of an organization more accurately, while stock prices may fluctuate depending on numerous incidents. Investors' sentiments and expectations may sometimes get biased by various events, e.g., the announcement of launching new products, rumors of merging with other companies, employee layoffs, or even company scandals can contribute to the inconstancies of the stock market price. [191].

We primarily gathered revenue data for our concerned ISPs from Ycharts [14] from 2007 to 2017

June. Similar to the stock market data, we matched quarterly revenue dates with the closest date for network information and calculated *NDI-d* and *NDI-c* values.

Figure 3.7 and Figure 3.8 show the revenue-based *NDI-d* and *NDI-c* values, consecutively. A couple of interesting observations can be made from these analyses. For example, most of the ISPs show a strong correlation compared to the stock price-based *NDI-c* in Figure 3.5. This is because, with more customers being served, ISP's revenue is increased, which may not reflect in the stock price always. Some ISPs have switched their place from being on the negative side in earlier discussion to positive *NDI* side in the revenue-based analysis and vice-versa (e.g., *Qwest* and *Cablevision*).

As expected, both transit and exchange ISPs show higher positive *NDI-d* and *NDI-c* values. Due to their business model of carrying traffic on behalf of other ISPs, or acting as a common point of interest, the more these ISPs remain connected to others, the more they can charge, and higher their profit margins are. This narrative is more accurate for the North American region, where ISPs are least interested in peering than charging others for transit [168]. Since revenue is a better representation of the profitability of a company than its stock market prices, this outcome is validating our intuitions.

Most of the access ISPs are grouped together and exhibit higher positive values except for *Qwest* and *Cablevision*, which is opposite to what we have seen from the stock-based *NDI* values. Similar can be said for CPs, with *Yahoo!* and *IBM* being on the negative side. This is not entirely surprising if we consider the fact that *Yahoo!* was going through an economic hardship as it was on the verge of getting out of business and acquired by *Verizon*. While this merger had a positive effect on *Yahoo!*'s stock market value to give it positive *NDI* in Figure 3.5, but it suffered from fewer revenue earnings and thus show negative values for revenue-based *NDI*. For *IBM*, more than one-third of its revenue comes from technology service and cloud platform [24]. Strategic outsourcing, integrated

technology services, or even selling software adds up the income for them, and thus explain why they have negative *NDI* value as its business has become less dependent on its network. However, during 2015, *Cablevision* lost about 6000 subscribers but still managed to increase its profit [86]. This hints, losing a good chunk of customers sometimes have less impact on revenue for some of the ISPs.

CDNs show an almost identical sparse behavior in both revenue-based and stock-based analysis with *Rackspace* having negative *NDI* values. As *CDN*s sit in the middle to deliver content from CPs to end-users using both transit and access ISPs network infrastructure, they are less interested in managing their network delivery system. They may have already come up with a business agreement with their carriers. This explains why sometimes CDN's revenue growth may not directly correlate with the number of connections or end customers they have.

AT&T, *Verizon*, and *TimeWarner* from transit & access ISP group show negative revenue-based *NDI-c*, which is similar to what we observed from their stock-based values. This justifies our assumption about them having diversified businesses beyond just acting as an ISP. All these ISPs also have their content subsidiaries and advertise a consolidated revenue. More research is needed to comprehend these ISPs' rationale, operating under the same management.

CHAPTER 4: PEERING AMONG CONTENT-DOMINATED VERTICAL ISPS

This chapter explores the plausibility of *vertical* integration of multiple ISPs from different layers by showing the geographic coverage of access ISPs vs. CPs within the US, and tries to understand the content-dominated vertical ISP market. Such *vertical* integration will certainly improve the end-user performance in terms of both delay and throughput. As the CP already possesses the detailed information about the traffic volume and thus, can also prepare its downstream carrier network to handle any sudden burst of traffic at any time. "Transaction cost"[203], a term frequently used by the economists, will also be reduced if providers adopt this new architecture. Since vertically-integrated ISPs will be governed under the single umbrella, it will neither charge its subsidiaries or sister concerns nor discriminate based on their business market, intended customers or some other attributes. With all of these possible benefits, *vertical* integration also brings new challenges like:

a) How will CPs establish the end-to-end network?

b) How big their footprints (geographically) will be? How will they inter-operate?

c) Once vertical merging is complete, how these new providers will peer with each other since they will grow bigger in size and their business strategy may shift?

One can explain challenge-*a* by arguing that the new management will most likely rely on the access ISP's already established infrastructure and will be more keen to fine-tune their specific requirements to offer more curated services as a bundle for the end-users. Challenge-*b* can be

Most of the content of this chapter previously appeared in proceedings [83].

visualized as the *union set* of previously separated access and/or transit ISPs and CPs existing coverage footprint. This is because these operators have been supporting either end-users regionally or are big enough to have global coverage. Primary concern will be whether there will be a healthy peering policy among these ISPs when *vertical* integration becomes the new norm.

The rest of this chapter is organized as follows: Section 4.1 details the procedure and mentions the data source. Section 4.2 explains the coverage area overlap and discusses the peering opportunity between the access ISPs and the CPs.

4.1 To Peer or Not to Peer

Generally, peering is beneficial and ISPs tend to peer if they can. In peering, ISPs carry their own traffic to another ISP's PoPs and agree to exchange traffic without paying any fees [130] to gain the reciprocal access to each other's customers. Alternatively, ISPs have to purchase transit service from transit providers for global reachability. Two ISPs will likely to peer if the following conditions occur:

a) both of them are similar in (customer cone) size and market value;

b) each of them cover multiple locations and their coverage areas are mostly non-overlapping; and

c) they generate similar traffic volumes.

In this dissertation, we consider conditions *a* and *b* only, as these pieces of information are publicly available, and condition *c* is mostly private. We show how much overlapping of coverage area exists between ISPs of different types and their market values. If ISPs are located in the same region, and their business interest is identical, it is expected that they are competing for the same

customer base and have a smaller chance of peering. ISPs, with their already set up infrastructure, running the business smoothly, would have less incentive to peer with a legitimate contender for the same customer pool in the same location. But, if two ISPs are operating in different areas, they will be motivated to peer with a specific set of settlement parameters (e.g., up to a particular traffic exchange ratio of 2:1 or 3:1 for traffic out to traffic in). It will expand both the networks' reachability without compromising their customer base. It is also expected that if the ISPs' market values are close to each other, there is a higher possibility of peering. It is because if an ISP is more prominent in size and market value, it will immediately start charging the smaller ISPs for carrying their traffic towards destinations.

4.1.1 ISP Peering Locations and Coverage Area

The internal topology of an ISP is proprietary information, and ISPs are restrictive when it comes to disclosing it. As a result, identifying the exact coverage of an ISP is difficult, and we have to rely on publicly available information only. ISPs share their PoPs so that other interested ISPs may consider them for potential peering depending on the policy they want to adhere. The peering policy type can be, from less to more conservative, either open, selective, or restrictive. ISPs advertise peering policy based on geographic scope, their backbone capacity, traffic volume and exchange ratio, and customer cone size. The geographical scope is a crucial parameter as overlapping coverage among two ISPs will reduce the likelihood of them peering as mentioned above.

We consider, in total, 37 (listed in Appendix A.1) major US-based ISPs of three types: 15 access ISPs, 15 transit ISPs, and 10 CPs.

As briefly discussed previously, ISPs use dedicated ASes for separating their multiple businesses operating under the same hood. For instance, Comcast is acting primarily as an access ISP even though it owns NBC and also is one of the owners of Hulu, a content provider. Similarly, Time

Warner is marked as an access provider, even though it creates content and recently merged with AT&T. We relied mainly on PeeringDB [7] for ISP categorization and used specific ASes associated with the categories. For example, *WOW*, and *GTT* mark themselves in PeeringDB as both access and transit, while *Verizon* identifies itself as content (AS15133) and transit (AS701). To make our analysis more comprehensive and inclusive, we tagged these ISPs in both categories. PeeringDB does not distinguish between CP and CDN. Both are treated the same, as, *content*. So, from now on, unless we compare CP with CDN, *content ISP* will refer them both. PeeringDB also provides peering locations of ISPs, which we utilize in our analysis.

We relied heavily on PeeringDB data, as all of the ISPs participate in updating their information there. Since the data is self-reported voluntarily, there is a chance of not getting 100% accurate and up-to-date data. However, a collaboration between CAIDA and Georgia Tech [133] cross-checked PeeringDB data with CAIDA's AS-rank, actual BGP announcements, and individual ISP's peering information published on their websites, and found that 99% of the ISPs reported correctly and updated regularly about their presence at different IXPs. Another recent work [62] has highlighted the fact that PeeringDB is very popular among the network operators, for whom reliable and curated peering information is a prerequisite. Further, some of the biggest ISPs (like Google, Cloudflare) are consistently using PeeringDB for either referring to their capabilities or use information directly collected from there to provision their configurations.

We have found that only *AT&T* does not provide any peering location information in PeeringDB. As such, we collected its peering location information from "AT&T Global IP Network Peering Policy" [20] instead. According to AT&T, only AS7018 is dedicated to US peering, and they require new peers to select at least six US locations from the list of cities which is available on their website. The cities are *New York City/Newark NJ; Washington DC/Ashburn VA; Atlanta; Miami; Chicago; Dallas; Seattle; San Francisco/San Jose; and Los Angeles.* So, we assume AT&T peers at least at these PoPs.

We used the PoP locations of an ISP to identify its coverage area, keeping in mind that an ISP can also have multiple PoP facilities in certain cities, depending on its business strategy. We listed all the peering locations (latitude, longitude) and PoP facilities count in a city for each ISP. Then, we calculated the *geometric median* [3] of those PoP locations. We consider this metric to represent the focal point of an ISP's coverage area because the sum of (geographic) distances from this point to all other PoPs is minimum and indicates the interest region of an ISP. The fact that we can generalize the *geometric median* to include weighted distances and convert it to a 'Weber problem' [73] also motivated us in selecting the metric. We shall call it 'centroid' onward.

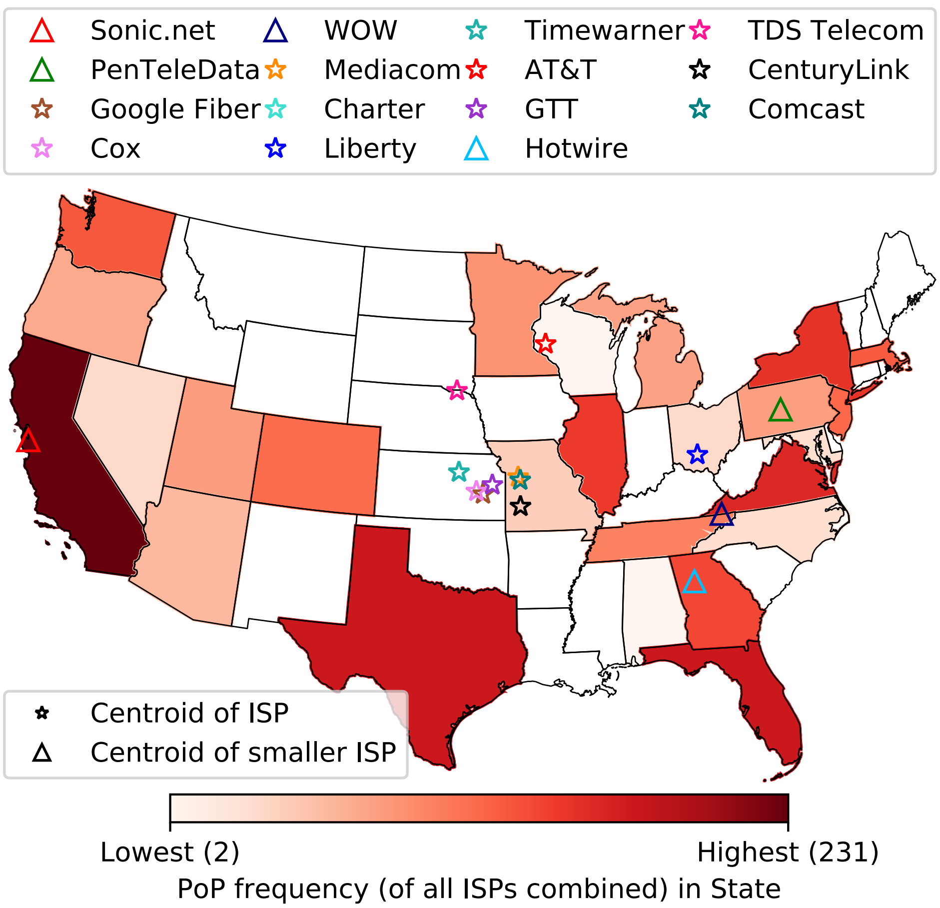

Figure 4.1: Access ISP coverage areas as centroids

An alternative methodology could be calculating the *geographic center* to represent the centroid. But, side-effects, like, two providers with PoPs in entirely different locations may end up having their *geographic center* nearby, prevented us from selecting this metric. As an example, let us consider a big ISP who covers coast-to-coast; its centroid would be approximately in Kansas or Missouri, while another small ISP operating mainly in Midwest may have the centroid located in Missouri as well. In contrast, the *geometric median* always tries to be closer to where most of the PoPs are present. It perfectly aligns with what we want to express; thus, we prefer this metric over the *geographic center*. Based on our data, access ISPs are mostly East Coast oriented (*Sonic* has been a dedicated operator in West Coast), as shown in Figure 4.1. We categorized access ISPs (especially) into small and big groups according to Wikipedia [5] since it marked an ISP with more than 1 million residential customers as big ISP and ISPs with fewer numbers were marked as small. Higher population attracts ISPs to expand, and this reflects in more access PoPs on East Coast.

4.1.2 Market Capital

Similar to what we did for our *NDI* analysis, here, we again collected market capital of 26 ISPs from either the NYSE or NASDAQ using Intrinio [13] and Microtrends [11]. It appears there is no unanimous preference from ISPs' side when it comes to choosing a trading platform, e.g., AT&T, IBM, GTT Communications are trading in NYSE while Verizon, Google, and Cogent are using NASDAQ. We collected data for the period of March 31, 2005, to June 30, 2018, with an interval of three months. To avoid holidays and weekends, we used the last business day of each month. Not all these ISPs were actively trading in the stock market during our data collection period. For instance, Level3 (stock ticker: LVLT) merged with Centurylink (ticker: CTL) in 2017; as a result, we have market capital for Level3 until 2017. On the other hand, Facebook (ticker: FB) announced its initial public offering (IPO) in February 2012, and so we have its data since then. On average, we have approximately 80% of the data points for each ISP over the time duration we investigated.

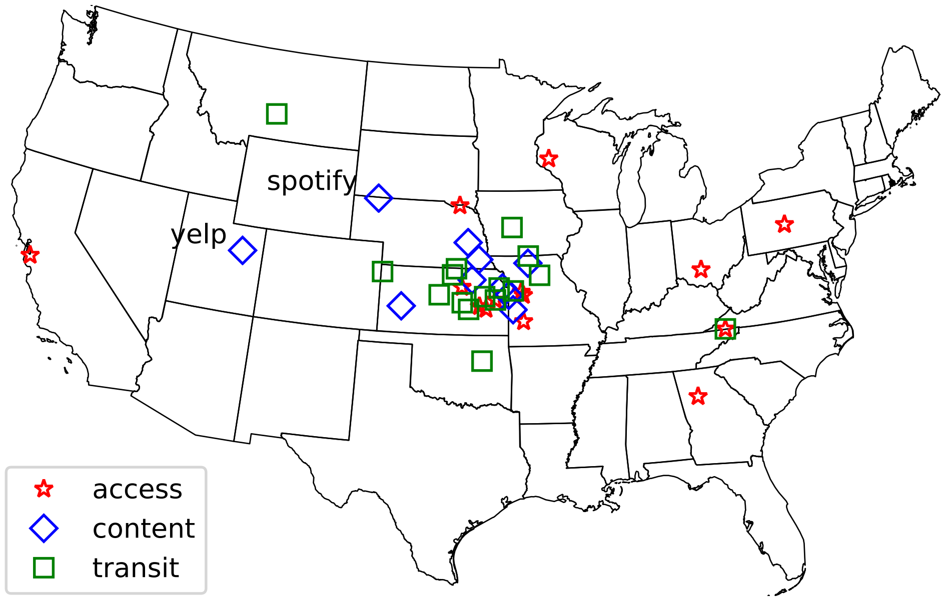

Figure 4.2: ISP coverage areas as centroids

4.2 Results

We plot the centroids of all ISPs in Figure 4.2. As expected, most of the CPs' centroids are condensed in the center of the US, which means they are operating coast-to-coast and have coverage areas spanning the whole country. Only two CPs are slightly on the left side of the central region: *Yelp!* and *Spotify*. They are smaller in comparison to the other CPs and prefer to peer mostly on the west coast as they have fewer peering points in the eastern region. The only transit ISP we see on the north of the country is *General Communications*, and it is mostly operating in the northwest. Therefore, it peers at two locations in Illinois (one private facility and one public exchange point), two in Washington, and one in Oregon. This behavior is similar to what we have observed from regional access ISPs, which are smaller in size and operating in either east or west coast only. *WOW* is a compelling case. We have found that it uses different PoPs for its access and transit services, even though it is operating in the southeast region.

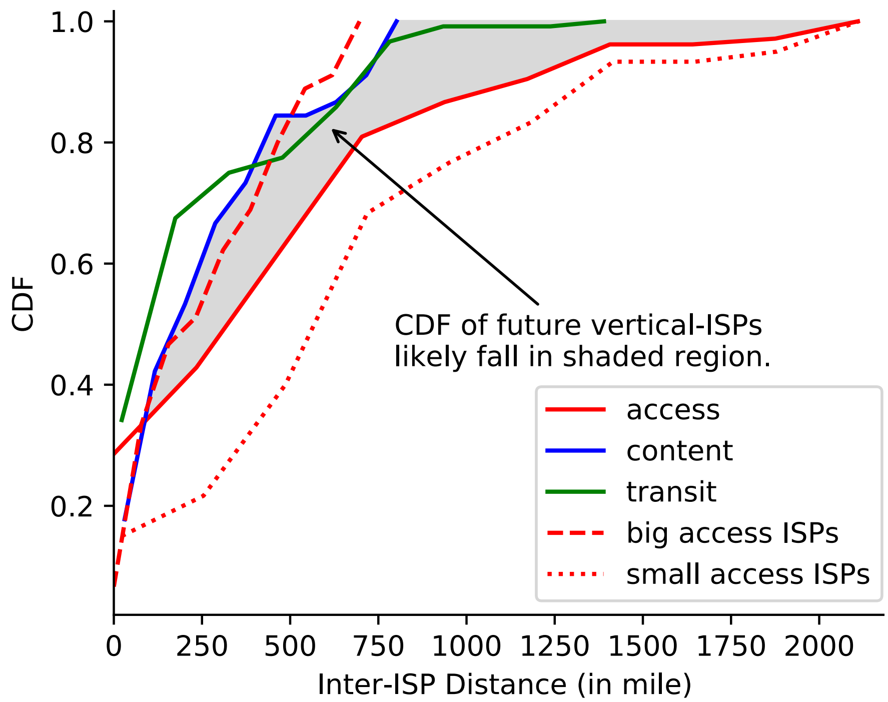

Figure 4.3: CDF of inter-ISP distance for all ISP types

4.2.1 Inter-ISP Overlap

To estimate how much an ISP overlaps with another one, we measure the distance between their centroids. We calculate 'inter-ISP distance' between pairs of same type ISPs and plot the CDFs in Figure 4.3. The outcome is pretty revealing. CPs are located very close to each other with the least inter-ISP distance among them, and access ISPs are the farthest from each other, while transit ISPs are sitting in the middle.

It is relatively easy to expand the business coverage area for CPs. Once the content is created, it is there forever; the only requirement is to peer or contract with a carrier provider who agrees to deliver the content to end-users. It gives CPs an upper hand as they continue to penetrate different locations without, effectively, laying any physical fiber cable, or establishing new data centers. As their coverage area expands, their centroids concentrate at the geographical center of the country.

Although, this classical behavior of CPs is changing as they are thriving for ensuring the best QoS for their subscribers and try to regulate the end-to-end delivery path from the very beginning. We observe that transit ISPs have centroids gathering at the center of the country, though they are a bit more dispersed than CPs. Transit providers have a strong backbone, and they usually lay their network in major cities where access or CPs (including CDNs) purchase transit support from them. They do not want to spread their coverage as far-reaching as CPs, but their footprints are *complete* and "well connected enough" [184] to cover the whole country.

Contrary to content and transit ISPs, access ISPs have significantly more dispersed centroids. It requires a notable amount of investments to provide Internet access service to a location. Unless an access provider has significant enough capital, it is bound to serve only regional consumers. To get further insight, we plot CDFs for small and big access ISPs separately. Bigger ones are more sparse, while the smaller ones are more regional and oriented towards a specific area. This presents a possible merger or peering scenario for small access ISPs with CPs since big access ISPs will not be interested in peering with its competitor. This causation may also explain the recent trend of paid peering agreements among content and regional access ISPs. All these events pave the way for the emergence of *sugarcane* ISPs.

Although it is hard to predict, content-dominated *sugarcane* ISP in the future will likely have centroids dispersed more than the current CPs but less than the current access ISPs. This prediction assumes that existing CPs and access ISPs will merge, which translates into an inter-ISP CDF in between the existing content and access ISPs' CDF plots, as illustrated in Figure 4.3. Further studies are needed to understand the dynamics involved in these mergers and their effect on the trajectory of the ISP market. Historical analysis of some of the content-access mergers could be made to gain more specific insights on how *vertical* integration will evolve.

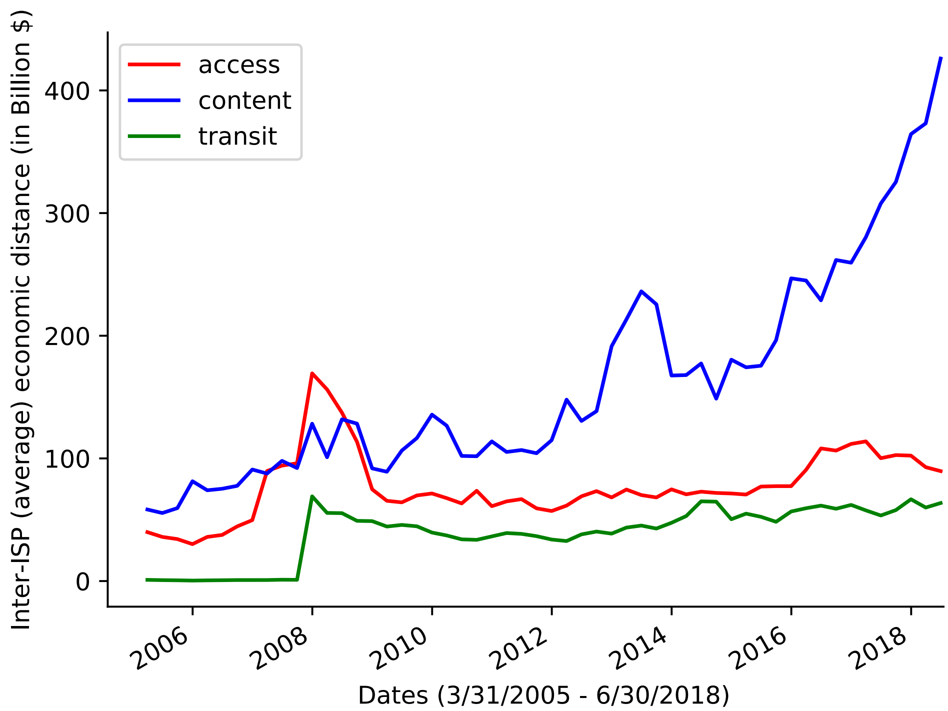

Figure 4.4: Inter-ISP *economic* distance for all ISP types over the years

4.2.2 Inter-ISP Economic Distance

A key measure to determine the peering likelihood between two ISPs is the similarity of their market value. To quantify how similar two ISPs are to each other in terms of value, we look at the *inter-ISP economic distance*, which is the absolute difference in market capitals of ISP pairs.

We present the average inter-ISP economic distance for each type of ISPs from March 2005 to June 2018 in Figure 4.4. Our observations from this measurement are multi-fold:

First, it is uncanny that the economic growth pattern for access and transit ISPs remains almost identical for the entire period except that the inter-ISP economic distance is higher among access than transit ISPs. This perception means there is a chance for small ISPs to survive in access business even if they serve only to a small number of customers. But, it will be exceptionally

challenging to run a transit business with little market capital because the maintenance of such an extensive network infrastructure requires more significant financial stability than access. Perhaps, this statement can be supported further by recognizing the fact that transit ISPs are losing its customers (ISPs) and becoming less profitable, while CPs and access ISPs continue to by-pass them and engage in “donut peering”, or even “paid peering” relationships.

Second, inter-ISP economic distance is consistently increasing for CPs. This episode of events can be either good or bad from a financial point of view. For instance, it represents a flourishing market with new contestants coming forward with innovations and value creation, which is good. Or, it may indicate an unhealthy competition where only a few players are dominating. However, if we refer to Figure 1.2, we can see that multiple players are contributing to the betterment of the overall content market. So, the latter is unlikely to be true.

To emphasize more on the content’s market dominance over the other two, we present CDFs of inter-ISP economic distance in Figure 4.5. Unlike Figure 4.4, which discussed the average of inter-ISP economic distance, this graph precisely compares the CDFs of market capital differences at two different timestamps, 8 years apart from each other. During this time, neither access nor transit market has noticed any clear dominance, while in the content market, some ISPs have taken a profound lead over others. This observation supports our claim for the future of a *vertically* integrated ISPs, where a CP can acquire an access and/or transit ISP and will decide whom to peer with.

Even though all three ISP markets have shown the tendency of becoming more skewed (see Figure 4.5) in terms of ISPs’ market value, of them, the content market has grown to be a more skewed one. A potential drawback is that peering may not be extensive in a highly skewed market since the market values of ISPs will not be similar. It is of particular concern for a content-dominated ISP market where highly skewed CPs will most likely propel the contracts and peering agreements.

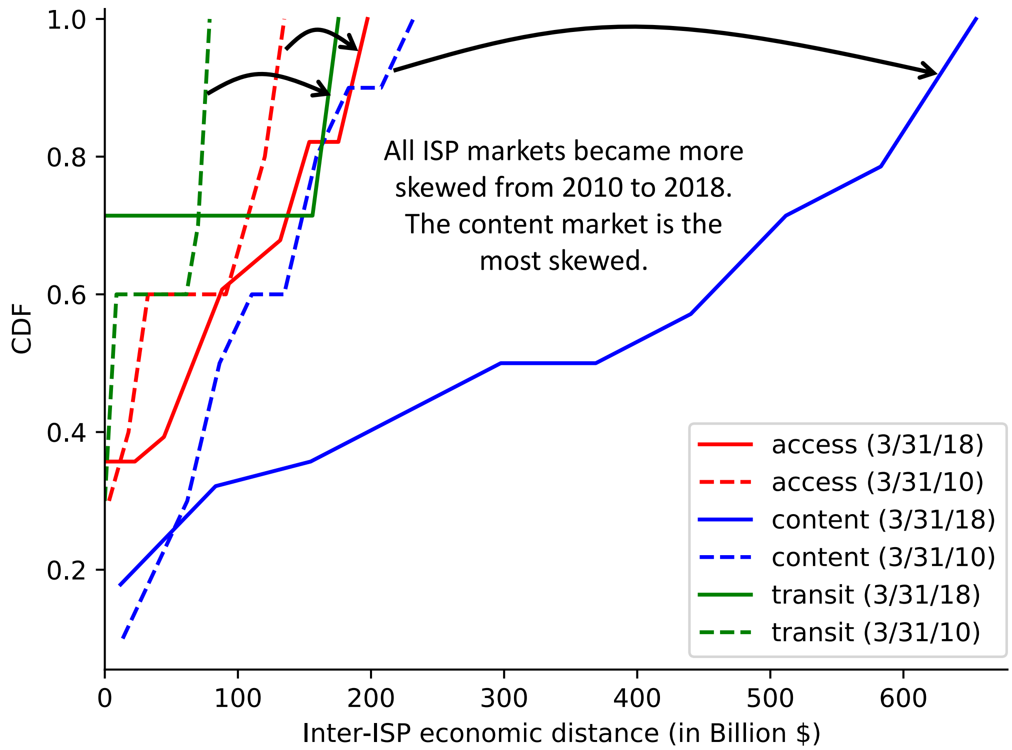

Figure 4.5: CDF of Inter-ISP *economic* distance for all ISP types from March 2005 to June 2018 (two different time snapshot)

Further, the skew will also increase the incentive of CPs to acquire others, which may drive towards an unhealthy oligopoly market.

In a content-dominated *sugarcane* ISP architecture, access and transit infrastructures will be used to carry traffic according to CPs' peering policies. As such, a *sugarcane* ISP will have to peer with other *sugarcane* ISPs for delivering its content to the end consumers in a different location where it does not own the network. Since access providers are the downstream retailer, peering will be mandatory; unless all individual ISPs want to build the underlying physical infrastructure for their own. Annual spending on broadband infrastructure in the US, an ambivalent indicator of whether ISPs coverage areas are expanding, has just recovered from its consecutive two years of downfall and hit $76.3B mark in 2017, still less than 2014's expenditure of $78B [65]. Net neutrality repeal,

in this regard, can be treated as an initiative to persuade ISPs to raise their network expansion investments and to attract more market competition. Yet, it is questionable if the *sugarcane* ISPs will be incentivized to peer since the existing content ISP market is highly skewed. If the skewness stays even after vertical integration, it is going to be less likely that the *sugarcane* ISPs will be eager to peer, which may degrade the overall end-to-end user-perceived experience. Detailed research is required to understand how the market capital distribution among the *vertically* integrated ISPs will shape up and how they will identify a potential peer without revealing confidential information about internal topology.

CHAPTER 5: META-PEERING: AUTOMATING THE ISP PEER SELECTION PROCEDURE

The term “Meta-peering” refers to automating the entire peering process. Starting with populating the list of ISPs that are likely to peer, it will automatically write the BGP rules to help establish a session between the two interested ISPs and extend the service to monitor all running BGP sessions for notifying any major outage or peering agreement violations. Here, we are not proposing any new tool; instead, we mainly focus on how to instrument the automation of the peer selection process and describe how *meta-peering* can be implemented by integrating some of the tools that the network community is already using for automating the BGP session establishment and monitoring purposes.

In this chapter, Section 5.1 details *meta-peering* and explains the peering phases. Section 5.2 expounds on the methodology of automating the peer selection process and outlines the data source. Section 5.3 and Section 5.4 describe the heuristics we have used throughout the peering deals selection process, and to identify the best peers. Finally, the results with sanity checks and validations are presented in Section 5.5.

5.1 Meta-Peering

Every ISP is informed about its own traffic matrices, transport costs for carrying traffic via its physical network topology, and current routing topology information so that it can select the least expensive (egress) PoP for each flow to minimize the total cost. This information also guides the ISP to determine the suitable PoP locations for exchanging traffic with other networks. If an ISP is purchasing transit service, it does not care anything but its interest because it is paying the fees.

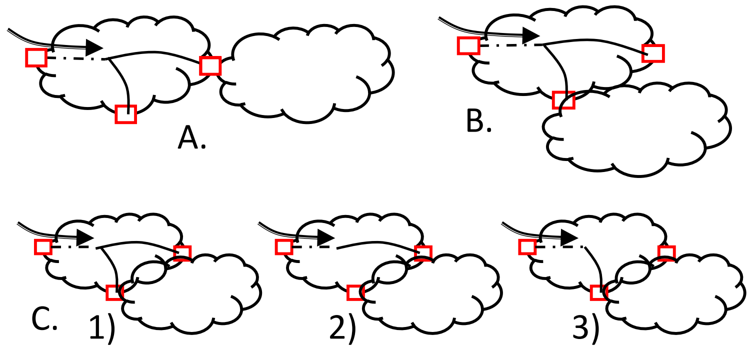

Figure 5.1: Possible PoP locations

While, in the case of peering, associated ISPs need to be careful, and consider only those PoP locations that will benefit both ISPs.

Figure 5.1 shows three possible combinations of PoP locations for two ISPs, where they can peer. *A* and *B* are the cases when ISPs are not located in the same PoP but willing to peer and essentially agree on a place that is closest from both. There is only one way of peering in such cases. For *C*, ISPs overlap, and there are at least two common PoPs between them, (there may be three or more), so they can either exchange traffic at all of them (*case 1*) or at only one location (*case 2* and *3*). ISPs can also peer without being physically present in an IXP and/or connect through a third-party reseller in a *Remote Peering* (RP) [157] manner. Despite being an option in practice, we do not consider RP in our model as they are opaque and controversial in terms of their performance benefits.

As we mentioned in earlier chapters, an IXP hosts ISPs inside a physical location (PoP) equipped with multiple Ethernet switches (to perform layer 2 functionality) so that they can route their traffic among themselves by shortening the path to reduce latency and improve round-trip time (RTT) and to avoid higher transit cost. All an ISP needs to do is to bring its traffic to the PoP, purchase port capacity from IXP to connect its router to the interfaces of 1/10/100 GbE Ethernet switch fabric.

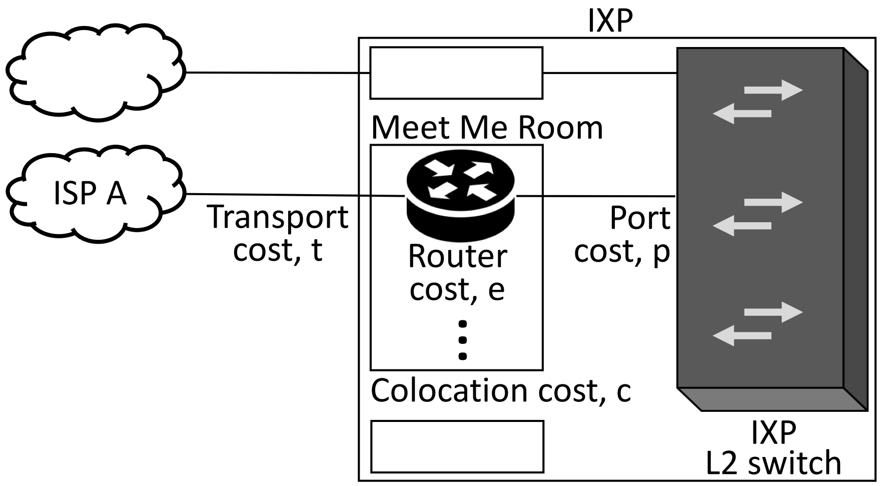

Figure 5.2: Peering at IXP and associated cost

IXP also charge ISPs for the colocation cost (combined electricity bill, cooling fee, security and others) [44]. The IXP allocates a Meet-Me Room (MMR) for each participating ISP where it can land its fiber and house all of its routers to interconnect with other ISPs. Figure 5.2 portraits a high-level representative architectural overview of an IXP and the associated costs that an ISP has to pay. An IXP also hosts *direct peering* where two ISPs interconnect each-other without any involvement of the switch. IXPs also operate RS to simplify the cumbersome process of setting multiple bilateral peering sessions and allow ISP to establish only one BGP session to connect with other networks.

ISPs have to be extremely cautious before selecting the IXP PoP because more (geographic) distance translates into more transport costs. Another deciding factor for choosing an IXP is the number of member ISPs that IXP has – having more ISPs co-located in the same location makes it more convenient for an ISP who wants to peer with other ISPs without spending extra.

Inspired by Norton's *Peering Playbook* [159], we break-down the entire peering process into four phases (see in Figure 5.3) and restrict our focus on the automation effort undertaken in each stage.

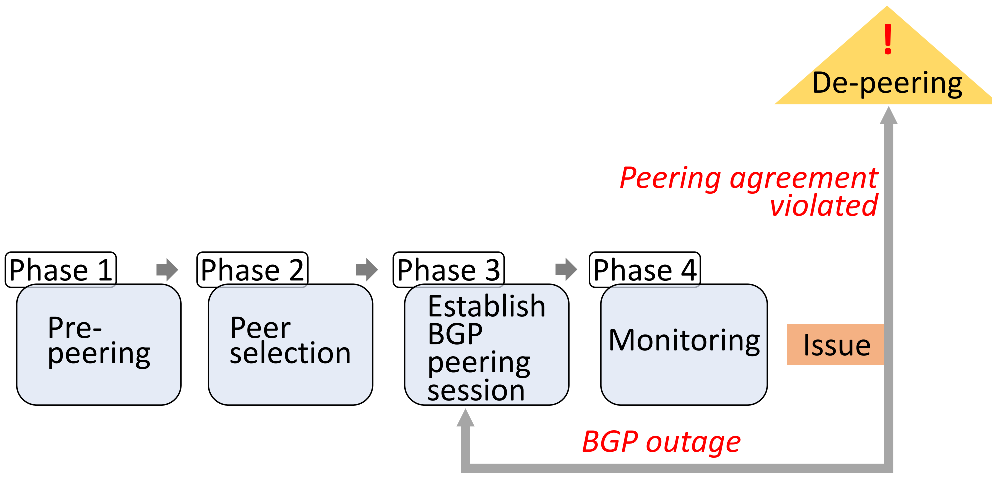

Figure 5.3: Peering phases

As mentioned in Section 1.3, *meta-peering* encompasses everything, including the tools used by ISPs, algorithms developed by researchers or academia, and associated companies who are contributing to this area and tackling the issues related to automation.

5.1.1 Pre-peering Phase: Key Peering Metrics

Choosing the peering partner depends mainly on the policy of the *requester* ISP, however, deciding whether peering is beneficial depends on the ISP itself. In the beginning, *requester* ISP sorts its priority, and accordingly, lists the potential *candidate* ISPs. From a purely economic perspective, if the peering cost is less than the transit cost, and the amount of transit traffic is larger but can be easily rerouted through the peering channel, requester ISP will be interested in forming peering relationship with a candidate ISP.

More Control: Regardless of economic benefits, an ISP may be interested in peering with more ISPs to gain control over its traffic and influence route path selection rather than letting someone else (upper transit provider) treat it as hot-potato. Thus, a requester ISP always looks for such

candidate ISPs to peer who has a significant presence in some other areas and is willing to deliver requester ISP's traffic there while keeping their individual traffic local. It is practically true for the CPs and explains why they are looking for peering with access ISPs, even if some of them might be very small in size or charge "paid peering" fees. Also, to avoid the "tromboning" effect [175] and to reduce the latency, requester ISP prefers peering over transit. Each time an ISP peer with someone new, the congestion reduces, reliability increases, and therefore, the end-to-end service quality for the users get improved [189].

As previously mentioned in Figure 1.4 from Section 1.3, most of the ISPs are usually open to peer, while there is a fair number of ISPs who implement either selective or restrictive peering policy. It may be beneficial for a requester ISP to follow the *80-20 rule* here and find out the top candidate ISPs where most of its traffics are destined towards, and then, try to peer with each of them for reducing the transit dependency.

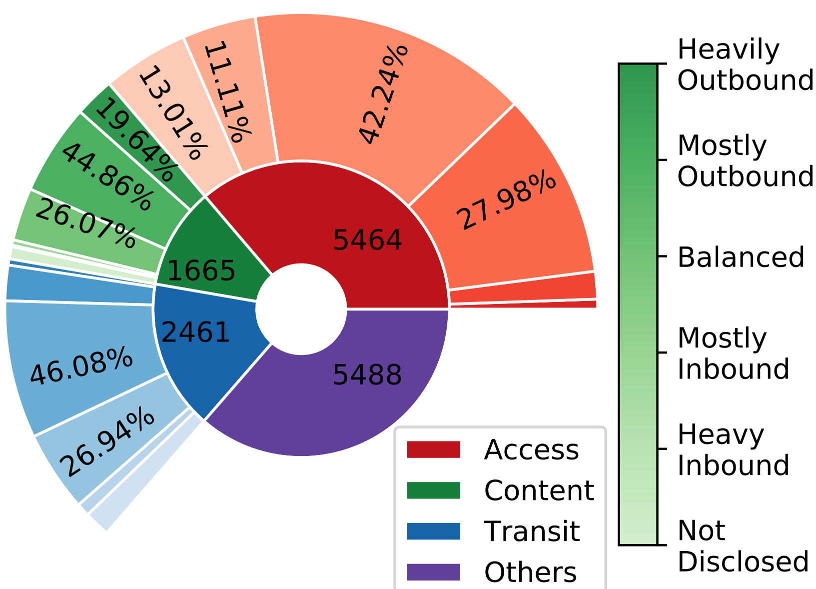

Figure 5.4: ISP traffic ratio for (only) US based ISPs

Traffic Ratio: A useful metric to consider for identifying potential peers is the balance of inbound vs. outbound traffic for an ISP. Figure 5.4 summarizes an interesting overview of the traffic ratios of different ISP types using all the ISPs currently present in PeeringDB. We have already stated, CPs are mainly interested in producing and disseminating the content; as a result, 90% of them are either outbound or balanced in nature; while access ISPs care about the end-user connectivity, thus, are mostly (82% of them) balanced to heavily inbound. On the other hand, transit ISPs serve in the middle to connect content and access ISPs. So, it is not surprising that almost half of the transit ISPs' traffic ratio is balanced. Using this information, a requester ISP, depending on its business strategies and traffic requirements, may identify a candidate ISP for peering more easily.

PoP frequency: Having more PoPs makes an ISP attractive to a requester ISP, who is expanding its global reach, for considering it as a potential candidate. So, the PoP frequency of an ISP is also a key metric to identify the potential peers. Figure 5.5 shows the CDF of the ISPs' PoP frequencies.

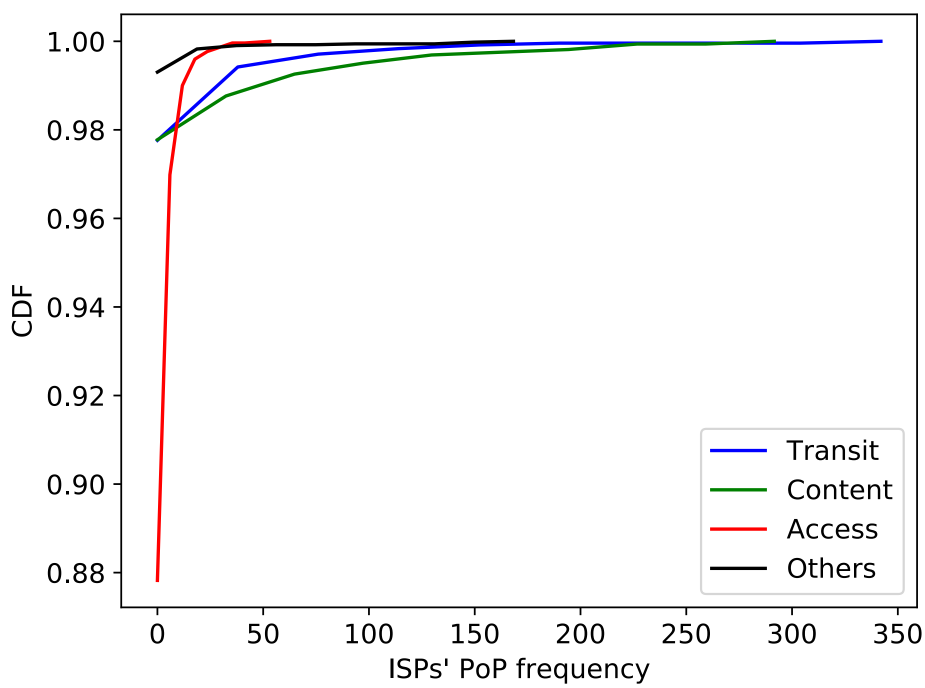

Figure 5.5: CDF of ISPs' PoP frequency

Access ISPs' objective is serving the end customers, and they mostly operate in specific regions. This is why they have fewer PoPs compared to other types of ISPs. Transit ISPs form the backbone of the Internet by laying actual fiber across the country and establishing PoPs in different locations so that other ISPs can purchase transit services from them. As such, they usually have more PoPs than any other ISPs. For instance, the maximum number of PoPs for a particular ISP from access, content, and transit categories are 59, 324, and 380, respectively. One notable observation from our analysis is, by putting caches directly inside of an access ISP or purchasing racks from datacenters, content ISPs continue to spread their footprint and increase the number of PoPs. In PeeringDB, there are a few institutions that mark themselves as educational, government, or similar non-profit organizations. We label them as *others*. Some of them are research-oriented and have PoPs all across the globe for probing, but do not have as many PoPs as either content or transit; and due to the fact that they are not involved in any business, we exclude them from our study.

Customer Cone Size: Traffic volume and advertised IP address space play vital roles in deciding the peering partners as well. A requester ISP, owning large customer cone by itself, tries to peer with such candidate ISP who has higher traffic volume and advertise higher number of address space. This is because a requester ISP needs to find a candidate ISP whom it can offer similar incentives from directly connecting with its network. Earlier work [133] shows a strong correlation between the advertised prefix count and traffic volume for both the access and transit ISPs except for the content ISPs; since they do not serve the end-users directly, not much address space is needed for them. This validates our assumption of peering between two ISPs with similar traffic volumes; because publicly available BGP-advertised address space can provide an estimation of other networks' traffic volume. We use the same definition of customer cone here, as we have already seen in Section 3.1

Table 5.1: Frequently asked requirements by ISPs

Requirements	% of all ISPs
24*7*365 Support	83.33
Minimum traffic volume (inbound or outbound)	69.44
No static route/ default route	66.67
Do not announce third party routes. (Only self customer cone)	66.67
Consistent route announcements in all inter-connecting locations.	63.89
Minimum geographic presence (peering at least in PoP count)	58.33
Interconnection speed at each point	52.78
Provide security; handle DDoS and abuse	44.44
Traffic ratio (in-bound: out-bound)	44.44
Routes registered in IRR, RIPE, ARIN	33.33
Existing transit customer can not peer	33.33

5.1.2 Peer Selection

ISPs typically outline their criteria for peering and describe the general guidelines so that potential requester can initiate the discussion. To sort out the most common prerequisites, we considered the top 50 ASes according to CAIDA's AS-Rank [15] and utilized PeeringDB categorization for their ISP types. We gathered the peering policies from each ISP's website and classified them as *i)* interconnection requirements, *ii)* routing requirements, and *iii)* general conditions and requirements. Some ISPs do not publish their policies, so we did not consider them. Some ISPs have multiple entries (different ASes) in the CAIDA's top 50 AS list, but we consider them as one single entity. Furthermore, no CPs were ranked in the top 50, so we considered some prominent CPs varying from the gaming industry (Blizzard), social media (Facebook) to CDNs like Limelight. Eventually, we accumulated a list for peering criteria using 36 ISPs (see Appendix A.2) from all over the globe.

Table 5.1 consolidates the most popular requirements and shows how many ISPs ask for them. Most of the ISPs want to ensure that the requester operates a 24*7*365 Network Operations Center (NOC) and provide an escalation path for resolving networking issues. Apart from the listed items in the table, other trendy desiderata are an omnipresent geographic footprint, adequate backbone capacity, essential peering port size in PoPs, maintaining financial stability, and redundancy of the requester network. Candidate ISP also does not peer with its direct customer or a customer of any of its peers.

For backbone capacity, candidate ISP intends to peer with the similar-sized requester. As a result, most of the ISPs look for an ISP who runs backbone with at least 10 Gbps (Gigabits per second) capacity or has at least 50% of the candidate ISP [187]. Big ISPs like Telia, Orange, or Liberty ask for 100 Gbps backbone, while ISPs like Frontier, Blizzard do not set the amount, but want to make sure that requester has enough backbone capacity to handle the projected load without any congestion. Candidate ISPs also decide the interconnection port capacity as well by mentioning the requirement of 10G LR or 100G LR4 Ethernet interface (e.g., Netflix [154]). According to Figure 5.6, which shows the peering port capacity of ISPs in various PoPs, most ISPs exchange at a speed of 1Gbps to 10Gbps while peering.

Not only the backbone capacity, but candidate ISP also specifies the minimum traffic volume that needs to be exchanged between them in the prominent direction (i.e., inbound/outbound, either way) with the requester. While most of our concerned ISPs ask for minimum traffic amount close to 10 Gbps, Verizon varies the traffic amount from place to place, and asks for 1500 Mbps traffic for peering in the US, 150 Mbps for peering in Europe, and only 30 Mpbs of exchanged traffic in Asia-Pacific. For measuring the traffic volume, the 95th percentile is usually considered by the ISPs (e.g., Cox, Charter), but some ISPs (e.g., Comcast) also use traffic average as the deciding factor.

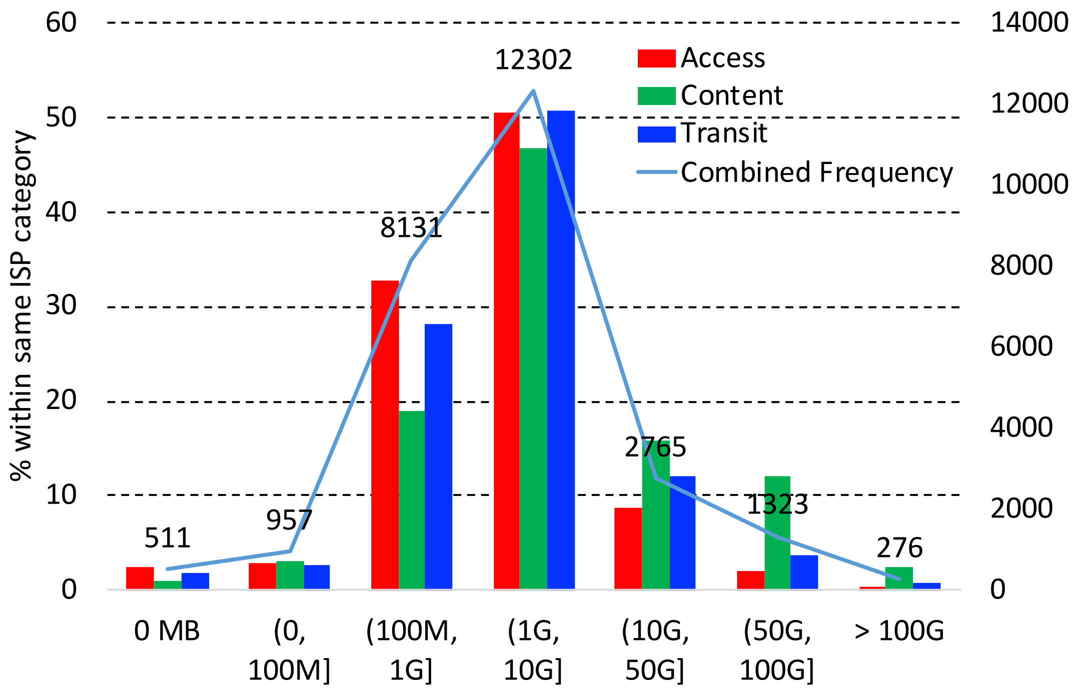

Figure 5.6: Port capacity of ISPs

ISPs also define the BGP routing policy before establishing the peering process. Some ISPs expect requester to announce an aggregated subnet size of /24 or higher (e.g., Time Warner and Zayo), some are more specific, requiring /30 or above (e.g., Google), while some ISPs (e.g., Swisscom) are more reluctant and ask for address range of at least /11. It is recommended that the requester ISP will alwasy respect the peering agreement, will not alter the *next-hop*, and will not sell transit service using only candidates' AS. To ensure these, candidate ISP looks for such a requester ISP who already serves at least 500 ASes or has 1,500 BGP customers globally. Although it may vary, having a wide geographic presence is desirable from a requester ISP. Depending on the nature of the candidate, it may require at least two (Cox) to eight (Charter, GTT) locations for peering in the US (Telstra requires presence in eight Australian cities), two (AT&T) to five (Orange) countries in Europe or have 50% of the candidate's (e.g. Verizon) geographic coverage. Peering with Telia is more rigorous as the requester needs to be capable of interconnecting in at least

40 unique metropolitan areas in Europe, 35 cities in North America, and three cities in the Asia Pacific/Oceania region.

To identify the potential candidate, ISPs can implement Netflow [78] inside their own network to measure the traffic and see which ISP exchanges the maximum amount of traffic flow in either direction. After that, PeeringDB can come handy for comparing the PoP locations and select the suitable one for peering. But, the problem is, all these peering prerequisite information have to be collected manually by either visiting individual ISPs' websites, or browsing PeeringDB consistently, and implementing the network measurement tools inside their own data center. Unfortunately, there is no single source that can provide a detailed guideline on the specific requirements of different ISPs. As mentioned earlier, discrete efforts have been made for storing ISPs' peering locations, basic peering policy, point-of-contact (PeeringDB), collecting geographical and topologically diverse data (CAIDA [22], Packet Clearing House [6], RIPE [9]), but, compiling the peering requirements of ISPs with comprehensive information about traffic and exchange ports, to automate the peering process, lacks till today.

5.1.3 Establishing BGP Session Phase

Even though ISPs may have agreed for settlement-free peering, they need to consider multiple factors before establishing the BGP sessions. A business relationship, whether the neighbor is a direct customer/provider/peer, as well as the intention to limit the routing table size for scalability, and to gain control over in vs. outbound traffic (by implementing MED [141] or LocalPref features) play vital roles in setting up a BGP session [67]. Therefore, how traffic will be routed through an ISP depends on how well the ISP blends all the policies. Erroneous manual configurations often lead to instability and generate excessive misconfigured route announcements, as such, causing unintentional blackhole routes where victim destinations become unreachable.

Recent incidents like inadvertent prefix leak causing service outage for Google and Cloudflare customers in 2018 [102], or sending entire Japan's Internet into the dark for more than an hour in 2017 [77] amplifies the importance of meticulous BGP configuration. To prevent such occurrences from happening, and to reduce message exchanges among ISP admins each time when either ISP expands its geographic presence by joining in a new IXP collocation center, *Coloclue*, the first-ever "peering-over Github" [115] approach has been introduced. Since most ISPs keep their information updated in PeeringDB [133], *Coloclue* can leverage this information to find out the common IXPs, calculate the max-prefix, and establishes BGP sessions. This is, by far, the only automation effort towards setting new or updating the existing BGP sessions between two networks.

5.1.4 Post-peering (Monitoring) Phase

Once the BGP session is established among the neighbors, an ISP has two salient activities to do. First, it keeps monitoring all the remote BGP services with its neighbors for ensuring the least amount of BGP outage or black hole, and second, it compares both ways aggregated traffic, so that, the measured ratio does not violate the peering agreement.

To automate the process, ISPs can either set up their internal monitoring systems or can purchase such services from several third-party providers like BGPmon [207]. In general, with real-time analysis using giga-bytes of BGP log data, assessment of the networks' routing health, and accurate detection of the unforeseen events like fiber cut or network anomalies due to prefix hijacking can be quickly done. RING is another effort towards enhancing the BGP sessions monitoring process, with the specific focus on network debugging. RING has been developed with the motive to provide friendly access for a participating entity to all the other participants' networks so that it can view its network from outside [182]. However, this requires proper knowledge of debugging and manual integration with the network components to troubleshoot the problem.

Among other BGP manipulators, the Noction Intelligent Routing Platform (Noction IRP) [156] is one of the well-known services which integrates intelligence to the routing decisions. By actively probing remote prefixes for packet loss, better latency, improved throughput, and long-term reliability, Noction IRP optimizes the performance of the routes. It is also capable of bypassing congestion and outages. With constant monitoring, it can automatically alert the network admin for various types of errors instantaneously and help him/her to troubleshoot the issues as well.

5.2 Automating Peer Selection

Intrigued by these case studies, we have developed a framework that considers the PoP locations, traffic matrices, port capacities in IXPs, or private facilities where both the requester and the candidate have their presence; and suggests the network administrator whether the candidate may agree with a particular peering offer or not. Utilizing this tool, the network administrator can come up with a list of potential ISPs; select the appropriate ones, and identify possible peering contract offers to discuss with these potential peers.

5.2.1 Methodology and Terminologies

In general, ISP admins possess traffic statistics and detailed intelligence about their internal network, but, have limited access about their competitor ISP's data while making the peering decision. Our algorithm leverages the publicly available dataset and, based on the internal policy of the requester ISP, produces a guideline of peering contracts. The algorithm contains a heuristic function that runs for both ISPs independently and enumerates the possible options separately. After that, comparing the lists, it generates an ultimate settlement points list. The algorithm always runs from one ISP's perspective and simulates the counterpart from publicly available information, depend-

ing on who is running the algorithm; hence, the heuristic function requires limited shared data and can run independently without breaching any security or privacy issue. Before getting into the details of the algorithm, we would like to categorize the data we shall use in the algorithm, introduce the terminology, and develop an optimization problem that our algorithm aims to solve.

Known data: Population at PoP locations, requester's Traffic Matrix (TM), port capacity at PoPs.

Estimated data: Candidate's TM, we call it Estimated Traffic Matrix (ETM).

Generated data: Possible Peering Points (PPPs), Possible Peering Contracts (PPCs), Acceptable Peering Contracts (APCs).

5.2.1.1 Possible Peering Points (PPPs)

ISPs publish the list of PoPs publicly (e.g., PeeringDB) and prefer to peer in one of those locations unless peering at a completely new place offers them enough benefits. In our study, we compile the PoP list from PeeringDB and identify the common PoPs for both the requester and the candidate ISPs. Traffic can be exchanged at any combination of these common PoPs, except for the empty set. We call the list of all PoP combinations as Possible Peering Points (PPPs).

5.2.1.2 Possible Peering Contracts (PPCs)

PPPs give us only the combinations of locations for possible peerings. After that, we feed all the TM and ETM information to the algorithm for generating the list of PPCs. We use TM to compute the traffic flow at particular PoPs. A PPC contains the offloaded traffic amount for both ISPs separately, and the total exchanged traffic between them, the difference of traffic from each individual ISP's point of view, and the ratio of exchanged traffic at every PPP.

For example, if there are three common PoPs between two ISPs, there would be seven different PPPs. Assuming the common cities are A, B, and C for both ISPs, PPPs, in this case, is: peering at all three cities (A-B-C), at different combinations of two locations (one of A-B, A-C, or B-C), or peering at a single location (either A, B, or C). Since *no peering* is not a valid option here, we discard that. ISPs having more common locations will have more PPPs, such as PPPs count rises to 15 for four common locations, to 31 if they have five common PoPs. r being the common PoP count between the two ISPs and, with the TMs being calculated, the PPC count would, then, be:

$$|PPC| = 2^r - 1. \tag{5.1}$$

5.2.1.3 Acceptable Peering Contracts (APCs)

Once all the PPCs have been populated, the algorithm sorts them according to ISP's sorting strategy and starts eliminating impermissible options from the list if they do not qualify. We call these selected contracts as APCs. The sorting criteria can be any of the following three options:

own: maximize requester ISP's own traffic amount that it can offload, regardless of how much traffic it receives from the candidate,

diff: minimize the absolute difference between in vs. outbound traffic for requester, or

ratio: choose peers with lower in vs. outbound traffic ratio.

After that, we rank all the APCs and normalize their rank values from zero to one. We call these scores as APC *willingness* score. The most preferred deal for an ISP is positioned first in its APC list ($\mathcal{APC}_R$ or $\mathcal{APC}_C$ for the requester and the candidate ISP, respectively), and gets the highest willingness score of one. Using $\mathcal{APC}_R$ and $\mathcal{APC}_C$, we populate the combined APC list, $\mathcal{APC}$, for these two ISPs. We shall detail this process in the next sections.

5.2.2 Optimization Problem Formulation

Suppose $\bar{\omega}_{Ri}$ and $\bar{\omega}_{Ci}$ be the individual willingness scores (described in next section) of the requester and the candidate ISPs for the same i-th contract, APC_i, in $\mathcal{APC}$, and $\bar{\omega}_{RCi}$ be the combined willingness of these two ISPs, for $i = 1, \ldots, |\mathcal{APC}|$. From now on, APC_i means an APC from $\mathcal{APC}$, unless otherwise explicitly mentioned. We need to find the best ordering of the APCs and generate the optimal set

$$\mathcal{APC}^{opt} = \{APC_1^{opt}, APC_2^{opt}, \ldots, APC_z^{opt}\};$$
$$\text{for which } \bar{\omega}_{RC1} \geq \bar{\omega}_{RC2} \geq \cdots \geq \bar{\omega}_{RCz}. \tag{5.2}$$

Previous work [116] assigned linear cost for every flow traveling through each link and formulated an optimization problem to minimize the total cost. Without specifying the nature of the cost, it suggested that ISPs could treat the timeliness of traffic delivery or the increment of the network capacity as cost. In contrast, we want to maximize the peering willingness between the ISP pairs in such a way that there is a better chance of peering between these two. An APC should not only benefit the requester but also seem reasonable for the candidate.

Let j be the rank of APC_i, the order of APC_i in $\mathcal{APC}$, and is expressed as $j = \Re(APC_i)$. $j = 1$ means the best choice, the next preference gets $j = 2$, and the rest follow accordingly. We

formulate the APC list generation as an optimization problem for the requester ISP as follows:

$$\mathcal{APC}^{opt} = \arg\min_{j \in \mathcal{J}} \sum_{i=1}^{z} j_i * \bar{\omega}_{RCi} * \bar{\omega}_{Ri} \tag{5.3}$$

$$\text{s.t. } \Re(APC_1^{opt}) < \Re(APC_2^{opt}) < \cdots < \Re(APC_z^{opt}) \tag{5.4}$$

$$\mathcal{J} = \{j_x, \ldots j_y\} \text{ and } j_x \neq j_y \forall x, y \text{ and } |\mathcal{J}| = z \tag{5.5}$$

$$z \leq 2^r - 1 \tag{5.6}$$

$$\text{policies are satisfied.} \tag{5.7}$$

where $\mathcal{PPC}$ is the set of all possible PPCs over r common PoPs, for which $z = |\mathcal{APC}| \leq |\mathcal{PPC}|$.

The optimal solution to the problem above happens when the ordering of APCs in the solution set is monotonically decreasing in terms of the combined willingness of both the requester and the candidate ISPs. Higher combined willingness score ($\omega_{\bar{R}Ci}$) may not always reflect the best APC choice for the requester, so Eq. 5.3 also considers the willingness score of the requester ($\bar{\omega}_{Ri}$) for that specific APC_i. The constraint in Eq. 5.4 also explicitly assures that the optimal ordering requirement is satisfied. Regardless of the policy constraint in Eq. 5.7, an earlier study [116] founded that "peering point placement problems" under traffic cost constraints were NP-complete. Our optimal APC generation problem also solves the same peering placement issue as part of each APC being generated. As such, we made certain assumptions to simplify the $\mathcal{APC}$ generation process and describe these heuristics next.

5.3 Heuristics for Peer Selection

5.3.1 ISP Network Model

Analyzing a candidate network to come up with the traffic volume, routing topology, link capacities are the most critical when deciding the PPCs. We have used TMs for estimating the traffic amount between every possible origin-destination (OD) node pairs, in this case, ingress and egress points, of the network to model the candidate's traffic volume. Existing TM calculation techniques rely either on statistical estimation based on routing matrix inference from SNMP link counts [142, 211] or measuring OD flows for a certain period [192]. Although how to achieve a proper balance between these methodologies of measurement, inference, and TM modeling has been heavily investigated [183], we have taken a simplistic but most common way to model TM of an ISP network.

We have collected the population data for each PoP of candidate ISP to estimate its TM using the *gravity model*. Since calculating the TM is not our primary focus, we borrow the idea of gravity model from earlier work on teletraffic modeling [124] and instrument the concept of *tomography* [72] in our algorithm. The *gravity model* contains mass and distance in the equation and brings the metaphor of physical gravity. Supported by the previous study [209] to observe similar behavior on the Internet, we assume the traffic amount between two PoPs should be proportional to the population of these PoPs. So, the traffic flow should be higher between the more crowded cities. With this rationale, we use the following derivation where F, m, a, G represent the

gravitational force, mass of the items, acceleration due to attraction, and gravitational constant:

$$F = G * \frac{m_1 * m_2}{d^2} \tag{5.8a}$$

$$F = m_1 * a_1 \qquad\qquad F = m_2 * a_2 \tag{5.8b}$$

$$\Rightarrow a_1 = G * \frac{m_2}{d^2} \qquad\qquad \Rightarrow a_2 = G * \frac{m_1}{d^2} \tag{5.8c}$$

$$\Rightarrow T_{1,2} = G * \frac{m_2}{d^2} \qquad\qquad \Rightarrow T_{2,1} = G * \frac{m_1}{d^2} \tag{5.8d}$$

where a_1 represents the acceleration of m_1 towards m_2. From ISP's business context, let us consider m_1, m_2 be the total population of two cities where the ISP has its presence (PoP1 and PoP2), and d be the distance between these cities. $T_{1,2}$ represents the amount of traffic flow from PoP1 to PoP2 and vice-versa for $T_{2,1}$. As everyone in a certain location (city or state) may not connect with the Internet, we define G as the usage factor by combining the Internet penetration percentage in a specific state [30] and per-person Internet usage (calculated globally [39]). We used the state-wise Internet usage information for the US only; G can be changed to represent more precise information for other countries to serve additional requirements as well. Let, s_o and s_d be the Internet penetration percentage in source and destination states (for US regional peering) or countries (global peering), respectively, and u be the per-person usage percentage. We express G as:

$$G = \frac{s_d}{s_o} * u. \tag{5.9}$$

As of now, we define G as the ratio of Internet penetration percentage only, but this variable can be extended to consider the country-wise per-person network-connected device count, max peak-hour behavior, and other factors to achieve further fine-tuning. Assuming s_1 and s_2 be the Internet penetration percentage of the two locations where PoP1 and PoP2 are located (PoPs can be in the

same locations as well, then $s_1 = s_2$), we can rewrite Eq. 5.8 as:

$$\Rightarrow T_{1,2} = \frac{s_2}{s_1} * u * \frac{m_2}{d^2} \quad \Rightarrow T_{2,1} = \frac{s_2}{s_1} * u * \frac{m_1}{d^2} \tag{5.10}$$

In a real-world scenario, the requester ISP should possess its own TM and only estimate the candidate's network. However, in this work, we relied on the above model for estimating both the requester and the candidate's TM as we did not have available TM information for any ISP. However, in the following, we still call requester traffic as TM and candidate's traffic as ETM.

5.3.2 APC Willingness Score

Figure 5.7 presents an overview of our proposed APC-generator framework. APC generation algorithm run by an ISP uses its own TM information, PoPs of both ISPs, population data for the ingress and egress cities, and tolerance information as parameters, and generates the list, $\mathcal{APC}$.

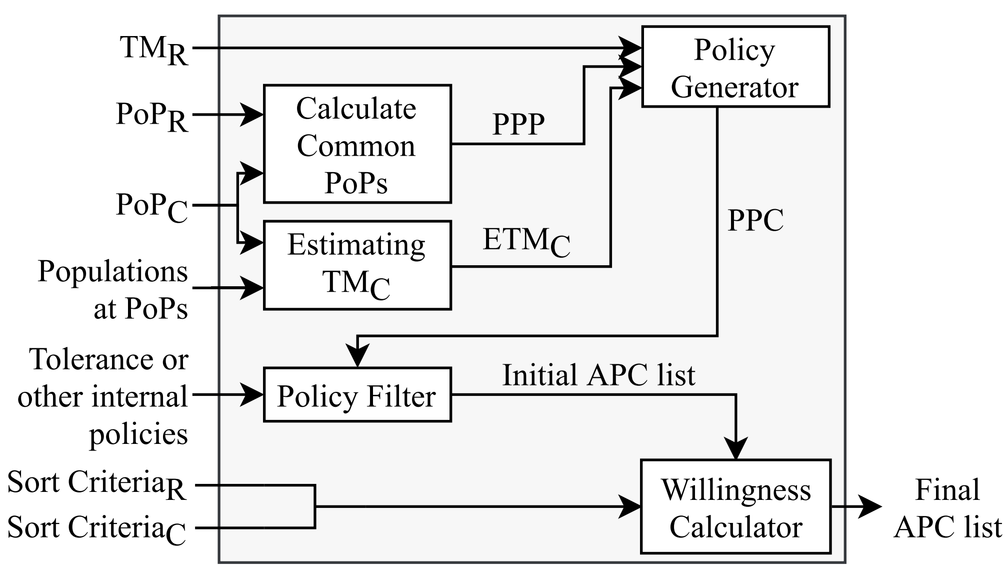

Figure 5.7: APC generator framework

The framework automatically crawls PoP information from PeeringDB and consults the population database every time before executing the algorithm.

From the point of the requester ISP, the algorithm produces two preliminary ranked lists $\mathcal{APC}_R$ and $\mathcal{APC}_C$ that show the APCs sorted according to their sorting criteria. Each item contains the list of suggested PoPs where ISPs should engage in traffic exchange, and associated in vs. outbound traffic amount. There may not be any settlement at all, which is not a desired output. To quantify the 'robustness' of an APC, we take these reports, and, for each APC, APC_i, we calculate the distance of ISPs' preferences, the rank of APC_i in $\mathcal{APC}_R$ and $\mathcal{APC}_C$, and take the square of it for a positive value. We then normalize this value by the squared of the maximum distance regardless of any specific APC to include the worst-case scenario when an APC is most preferred by the requester ISP but least preferred by the candidate. Let $\Re_{Ri}$ and $\Re_{Ci}$ be the rank of a particular APC_i in $\mathcal{APC}_R$ and $\mathcal{APC}_C$. We calculate the individual ISP's willingness scores $\bar{\omega}_{Ri}$, and $\bar{\omega}_{Ci}$ as following:

$$\bar{\omega}_{Ri} = 1 - \frac{\Re_{Ri} - 1}{|\mathcal{APC}|}, \qquad \bar{\omega}_{Ci} = 1 - \frac{\Re_{Ci} - 1}{|\mathcal{APC}|} \tag{5.11}$$

We want both of these values to be comparable to each other; and, thus, use the *geometric mean* to calculate the combined willingness score. We can express the combined willingness score, $\bar{\omega}_{RCi}$ using the following formula:

$$\bar{\omega}_{RCi} = \sqrt{\bar{\omega}_{Ri} * \bar{\omega}_{Ci}} \tag{5.12}$$

5.3.3 Offloaded Traffic Estimation

During our study, we found only *Microsoft* to explicitly mention that they would try to carry the data through their network to a PoP nearest to the user location. They also expect the requester ISPs to announce their entire and consistent set of prefixes in all the PoPs so that Microsoft can

take advantage of choosing the exit point. In contrast, many ISPs do not follow this practice and implement the *hot potato* technique. While calculating ETM of the candidate, we assumed that the traffic would enter or exit with proper discretion, and, the entire volume would be proportionately distributed to the port capacities between all the common PoPs. As such, we did not consider the 'closer to geo-location' phenomenon here. We used unit values while calculating data flow for both the ISPs. To make the model (in Eq. 5.10) more realistic, rather than only averaging the total traffic and distributing them, we utilized the port capacity of each PoP as the weight factor to calculate the weighted average. This gave us a better approximation of ETM for ISPs.

Consider Figure 5.8, the requester (R), and the candidate (C) both have 4 PoPs individually and P_{RC1}, P_{RC2} are their two common exchange points. Traffic from R can reach to any of R's PoPs.

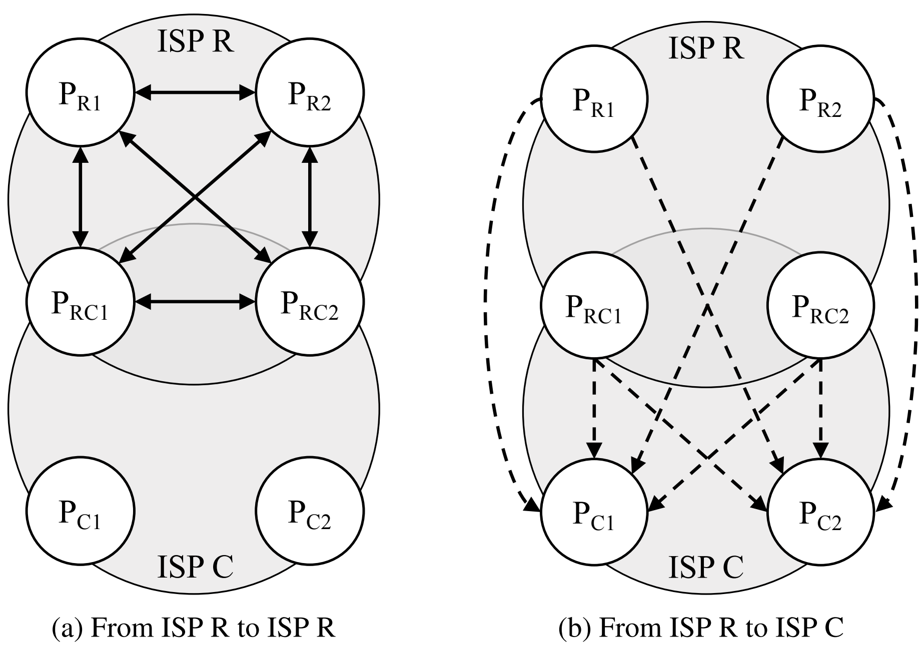

Figure 5.8: Traffic among Requester & Candidate ISPs

Traffic from R can also go P_{C1}, P_{C2} via P_{RC1} or P_{RC2} if they agree to peer. From R's view point,

$$\text{Outgoing traffic from } P_{R1} = \sum_{i=1}^{m} T_{R1,Ri} + \sum_{j=1}^{n-r} T_{R1,Cj} \tag{5.13}$$

where, m, n are the number of PoPs that R and C have, while r is the number of common PoPs. Each ISP will carry its own customers' traffic destined to its peer via these common points. Since an ISP will not be interested in their own traffic for this case, we set

$$\sum T_{R1,Ri} = 0 \tag{5.14}$$

Renameing the rest of the traffic as *delegated* traffic, we get:

$$\text{Delegated traffic from } P_{R1} = \sum T_{R1,Cj} \tag{5.15}$$

This traffic has to go via either P_{RC1} or P_{RC2}, and based on their port capacity, we distribute the traffic among these exit points. Considering v_{RCk} be the port capacity of k-th common PoP, P_{RCk}, we formulate the traffic that R will offload to C via P_{RCk} as:

$$T_{RCk} = \frac{v_{RCk} * \sum_{i=1}^{m} \sum_{j=1}^{n-r} T_{Ri,Cj}}{\sum_{k=1}^{r} v_{RCk}} \tag{5.16}$$

In general, ISPs do not exhaust the total capacity of a single port; rather, they limit the utilization to a certain threshold. Should the need arise, an additional port is either purchased from the IXP or a new port is activated (if ISP owns the switch) to support the extra load. Previous studies found that 95th percentile average utilization varies from 36% port usage to maximum utilization of 50% at peak hours [63, 91]. There were a few incidents when the utilization reached up to 90%, but those accounted for less than 10% of the cases [29]. For our analysis, we assumed that ISP would

take care of the bandwidth requirement, and as such, we did not enforce any kind of port capacity constraints.

5.3.4 *Common PoPs and Traffic Behavior*

Figure 5.9 shows the CDF of peering PoP frequency for ISP pairs. We use CAIDA AS Relationship – with geographic annotations dataset [68], which provides the list of all the PoPs where two ISPs are peering, as seen by the BGP communities. According to the figure, more than 95% of the ISPs, peering globally, use 15 or less number of PoPs inside the US for exchanging data with their peers. To reduce the complexity of our analysis and search space, we set the maximum common PoP count to 15 for any ISP pair.

Figure 5.4 in earlier discussion states that most ISPs announce their traffic ratio policy beforehand.

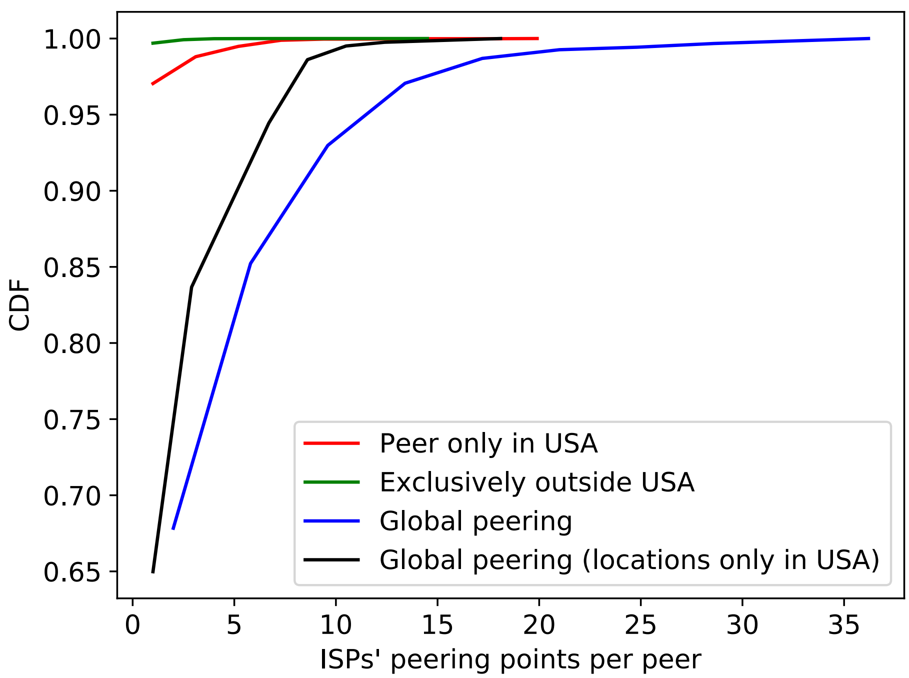

Figure 5.9: CDF of PoP frequency ISP use per peer

Table 5.2: Peering pairs traffic ratio type percentage according to CAIDA

	HI	MI	B	MO	HO
Heavy Inbound	0.8	2.3	2.6	1.1	0.4
Mostly Inbound	2.9	8.7	11.4	4.6	1.5
Balanced	3.4	14.1	19.2	7.9	2.4
Mostly Outbound	1.0	3.7	5.3	2.2	0.6
Heavy Outbound	0.4	1.2	1.6	0.6	0.1

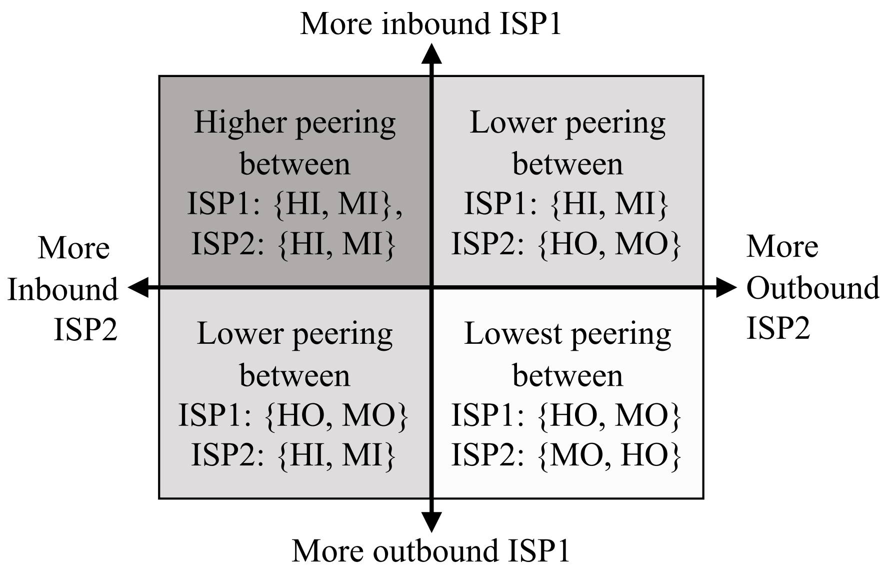

Figure 5.10: Peering pairs traffic ratio (based on Table 5.2)

This practice is followed by most of the ISPs for minimizing the number of requests they receive and also to assist the requester ISP in finding its appropriate candidate. CAIDA published the inter-ISP relationship information of a total of 339,357 ISP pairs; of them, 209,533 pairs were peering. Ignoring 57,711 pairs, as involved ISPs did not publish their traffic ratio type in PeeringDB, we show the rest of them as the percentage of peering ISP pairs according to their traffic ratio type in the Table 5.2. Balanced ISPs are the most popular for peering, and the maximum number of peering happens when both of them are balanced (19.149%). But if either of them is balanced, the chance of peering settlement is still higher compared to the other combinations. We populate a

peering possibility quadrant in Figure 5.10 based on ISPs traffic nature and employing the insight from Table 5.2. Here, ISP1 can be the requester and ISP2 can be the candidate, or vice-versa. It does not impact the conjecture.

A recent discussion among NANOG email-list members refutes the general idea of traffic ratio playing an essential role in peering candidate selection procedure. According to this conversation, most of the ISPs do not calculate their out vs. inbound traffic ratio when labeling their traffic ratio type. They merely monitor their average traffic, and based on the summary graphics, they introduce themselves to others.

Nevertheless, our study on ISPs' peering policy suggests that candidate ISPs frequently ask requester ISPs not to exceed a certain traffic ratio. In practice, for outbound to inbound traffic amount, the ratio varies from 1:1.5 (*Telstra, CenturyLink*) to 1:3 (*Liberty Global*). The most popular one is 1:2, maintained by *Zayo, AT&T, GTT Communication* and others. These ratios are essentially peering policy decisions ISPs make, and we include them as a measure of tolerance while eliminating PPCs to prepare the $\mathcal{APC}_R$ or $\mathcal{APC}_C$.

While choosing from multiple peering deals, ISPs tend to accept the one favoring them the most. Often, the best deal would be the one which allows them to transmit more traffic to their peer than the amount they receive from their counterparts. In consequence, it introduces one of the trenchant criticisms involved in peering, as, in many cases, ISPs try to abuse the friendly relationship between themselves by offloading more traffic than they are supposed to (see Chapter 2).

5.3.5 Negotiation Settlement

Despite the in-depth analysis of traffic volume, cost benefits, performance improvement, and others, negotiation plays a crucial role in settling a peering relationship. For instance, there are cases

where locations are not that important for some ISPs who have global coverage; the only thing that matters is the traffic volume. This is particularly true for CPs [133]. Policies that ask the requester to notify the candidate before any update in its side, or consider requester's financial stability, or expect the requester to provide equal quality services, and mandate the requester to maintain the aggregated capacity same as the candidate ISP offers, may not always be interpreted as numerical values. It is a well-known fact that BGP suffers extensively from implementing many policy-related issues, and often, traffic takes a longer route ignoring the shortest path [97] causes unnecessary path stretch. However, we do not consider such policies and only focus on a simplistic negotiation scenario, in which both ISPs want to settle at the best contract option based on their APC list.

We speculate a reasonable co-relation between the negotiation process and the size of the ISPs. Assuming C_A, C_B be the customer cones of two corresponding ISPs A and B, while T represents the transferred traffic (T_{AB} refers to the traffic from A to B, and vice-versa), we get the following possible scenarios:

$$C_A > C_B \Rightarrow T_{BA} > T_{AB} \qquad \rightarrow B\ wins \tag{5.17a}$$

$$C_A < C_B \Rightarrow T_{BA} < T_{AB} \qquad \rightarrow A\ wins \tag{5.17b}$$

$$C_A \approx C_B \qquad \rightarrow both\ win \tag{5.17c}$$

5.4 Selecting the Best Peer

From the previous section, we only gather the willingness scores for each of the APCs. But these do not necessarily answer the question of whether or not the candidate ISP will accept the requester ISP as its peer. Utilizing both ISPs' coverage areas and a weighted willingness score based on all the APCs from $\mathcal{APC}$ could give us a procured picture of the two ISPs' peering possibility. We

shall call this as *felicity* score. The higher the score is, the happier these two ISPs are with each other, and the higher the chance of peering.

5.4.1 Coverage Area Overlap

The internal topology of an ISP is proprietary information, and ISPs are restrictive when it comes to publishing it. As a result, identifying the exact coverage of an ISP is difficult, and we have to rely on publicly available information about their PoP locations. Large interconnection hubs (i.e., IXPs) are usually located in bigger business cities or near the trans-oceanic cable landing points [185]. Thus PoPs may reflect the business interest of an ISP. If an ISP has more PoPs in a certain area, that means its customer cone is concentrated around that region [61], and, to support higher volume of customers, it increases its presence in nearest IXPs or private facilities, ensures higher port capacity [71] for traffic exchange, and improves infrastructure [185]. But, ISPs, thriving for country-wise reachability, mostly have a presence in coast-to-coast (in US case), or those who want to expand beyond a specific country and ensure global connectivity, they have to have a presence in different countries. To gain the benefits of broader coverage, ISPs peer. In this regard, geographical footprint coverage acts as one of the deciding factors to consider, and hence, overlapping among two ISPs will reduce the likelihood of them peering. We have already discussed about this issue, in detail, in earlier chapters.

To represent the coverage area of an ISP, we use a different technique here. Instead of using a single point (*geometric median*) to represent the ISP's area of business interest as we did previously in *sugarcane* ISP analysis, this time, we calculate the *Convex-hull* using all of its PoP locations. Owning multiple PoPs in the same city does not necessarily change the overall coverage area of an ISP, as by nature, convex-hull takes only the out-most point of a network and ignores the rest of the points falling inside. Figure 5.11 presents the coverage area of two ISPs as convex hulls.

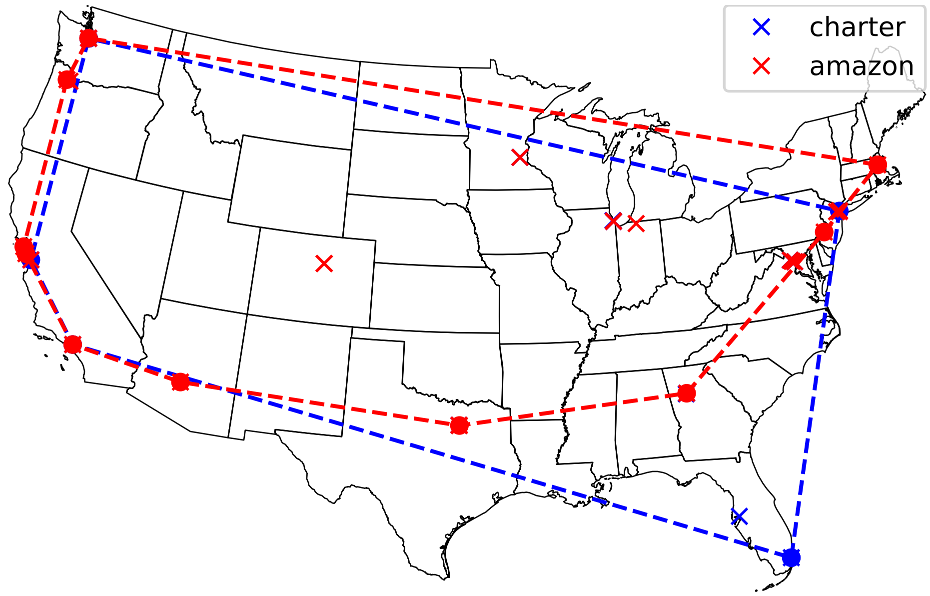

Figure 5.11: Inter ISP overlap between two ISPs

An ISP will be interested in peering with another ISP if the relationship would expand its coverage area. Otherwise, there may not be enough incentive to peer with someone who is covering the same locations or has a smaller coverage area. We call this interest as an *affinity* score of an ISP to peer with another ISP. We use *set* theory to calculate the affinity scores α_R and α_C, respectively, for requester R and candidate C. Let A_R and A_C be the coverage area of R and C, respectively, and A_o be the overlapped area. We express the affinity scores based on the overlap as:

$$\alpha_R = \frac{A_C - A_o}{A_R \cup A_C} = \frac{A_C - A_o}{(A_R - A_o) + (A_C - A_o) + A_o} \tag{5.18}$$

$$\alpha_C = \frac{A_R - A_o}{(A_R - A_o) + (A_C - A_o) + A_o} \tag{5.19}$$

Just like calculating combined willingness scores, we use a similar method of geometric mean to calculate the combined affinity score between the requester and candidate:

$$\alpha_{RC} = \sqrt{\alpha_R * \alpha_C} \tag{5.20}$$

5.4.2 *Felicity in Peering*

Using individual APCs' willingness score for each item in $\mathcal{APC}$ produced from Section 5.3.2, we formulate the willingness score based on APCs as:

$$\omega_{RC} = \frac{\sum_{i \in \mathcal{APC}} \bar{\omega}_{RCi}}{|\mathcal{APC}|} \tag{5.21}$$

At this point, we have an affinity score based on coverage area overlap (from Eq. 5.20) and willingness score based on APCs (from Eq. 5.21). We take the geometric mean of these two values to generate the ultimate felicity score for the requester-candidate ISP pair as:

$$f_{RC} = \sqrt{\omega_{RC} * \alpha_{RC}} \tag{5.22}$$

This felicity scoring emphasizes that the ISPs would want to peer more if they have both low coverage overlap and high willingness for possible peering deals.

5.5 Results

Although the number of IXPs is higher in Europe compared to North America [119], Table 5.3 suggests that except for ISPs which are operateing *globally*, most of the ISPs are serving regionally and have their presence in a single continent. Based on this, we argue that peering is more relevant for regionally operated ISPs compared to what it means for the inter-continental ISPs. With global reachability, these inter-continental giant ISPs will charge from the smaller ISPs for transit service anyway. As such, we restrict our analysis among 22 US-operated ISPs (see Appendix A.3) and assess only those PoPs which are located in the US. A similar anatomy of the peering relationship of European ISPs can be made. We used the US Census Bureau [8] for the population database.

Table 5.3: ISPs' geographic scopes vs. PoPs (from PeeringDB)

ISPs' geographic scope	Total ISPs	PoPs located only in 1 continent
Africa	216	92.13%
Asia Pacific	1157	92.74%
Australia	192	94.27%
Europe	2127	95.77%
Middle East	57	91.23%
North America	818	96.58%
South America	885	96.27%
Regional	4092	96.92%
Global	1426	62.69%
Not Disclosed	3993	98.87%
No Info	115	93.91%

Because of the observed diversity in traffic ratio policy, IXP presence, port capacity, and aggregated traffic volume among ISPs, we performed a careful sanity check on the data to validate our algorithm and to ensure the results align with expected behavior.

Most of the ISP pairs did not have many common PoPs, and as such, both of their $|\mathcal{APC}|$ and $|\mathcal{PPC}|$ were smaller. We have found that 79% of ISP pairs that we were considering had less than or equal to 15 common PoPs while, for 11%, the count was between 16 to 21. It is no surprise that *Google* and *HE* pair was leading the sequence with 51 common PoPs. Currently, the algorithm does not impose any stringent filtering except for *traffic ratio*, as discussed in Section 5.3.4 on the $\mathcal{PPC}$. With relaxed screening, ISP pairs are allowed to generate maximum $\mathcal{APC}$ and can achieve $\frac{|\mathcal{APC}|}{|\mathcal{PPC}|}$ ratio up to 1.0. Another tenable filtration criteria can be *bit mileage* which we did not use. We should also mention that $max(|\mathcal{PPC}|)$ is 32,767 with 15 maximum common PoPs (as per Section 5.3.4). No APC was generated, specifically, when one of the peers was *Cogent*. In PeeringDB, *Cogent* did not disclose its traffic ratio, and, at first, we designed our algorithm to ignore a pair for possible peering if either of them did not disclose their traffic ratio. Considering

Cogent being a tier-1 transit provider, we assumed Cogent's traffic to be balanced, and accorded full consideration for peering, as we shall see in the following discussion. Out of 392 possible pairs, our algorithm identified 243 as potential peering pairs, and 149 pairs were not recommended.

5.5.1 Sanity Check

CAIDA published the customer cone of 73,148 organizations from 76 countries (as of April 2019). Using CAIDA AS-Rank API [15], We accumulated all of the announced prefixes and covered address spaces to get the total numbers. We used this dataset to identify the similar-sized ISPs based on their customer-cone size and their coverage areas.

Figure 5.12 compares ISPs' felicity scores with their similarity scores. ISPs pairs are least intrigued for the APCs, which are sorted according to *diff* strategy. This observation is valid for all three similarity scoring criteria. We separate the ISP pairs where either of them is a content ISP on the top and the rest pairs on the bottom. We show the trend-line of felicity scores for each graph. While they are astonishingly close to each other for all three categories, similarity based on ISPs' pop count (i.e., the ratio of the both ISPs' pop count, with smaller one being at the numerator) performs the best. The left (two) and the middle (two) figures are synced with each other as more prefix count means more IP addresses and more customers. ISP pairs are concentrating on the left, and very few pairs are scattered as we move towards a higher similarity score for these four figures. This is because there are not many ISPs with the same number of prefixes and thus exhibiting lower similarity. Higher felicity scores with low *prefix count* based similarity scores can be explained by the right figures, which compare two ISPs' similarities based on their PoP counts. ISPs, with a varying number of users, or coming from a different section of the business spectrum, may share the same underlying attitude of expanding their presence as much as possible, and this made the pairing ISPs similar to each other.

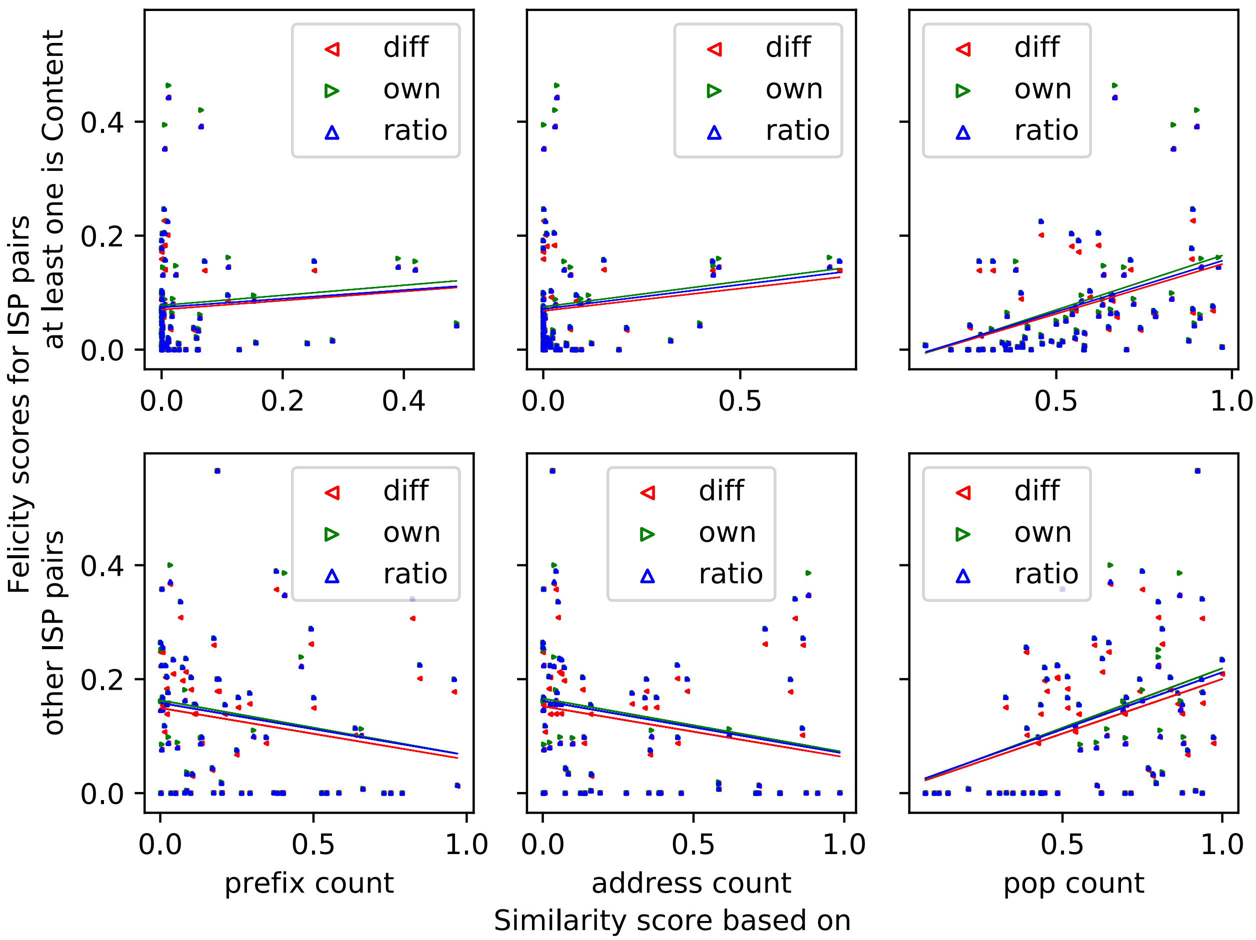

Figure 5.12: Felicity scores of ISP pairs peer

A compelling case is the non-content ISP pairs as they show lower felicity scores despite their prefix/IP address-based similarity scores are increasing. One possible explanation can be as following: since they are either transit or access ISPs, they try to avoid peering with someone who is competing for the same end-users. For example, *Verizon* covers 13.52% of total IP prefixes (according to CAIDA), and *PCCW Global* has 25.98% coverage. Still, their affinity score is quite low, meaning their geographic footprint overlaps too much, making it less lucrative for peering. Our analysis shows that out of 55 possible access-transit pairs, eight pairs produce felicity score of zero due to complete overlapping between them (for which their affinity scores are zero) while their

similarity scores are higher than 0.4. It is also true for transit ISP pairs as almost 15% of possible transit pairs (four among 27) produce a felicity score of zero. This measurement indicates that, as a transit or access ISP grows more prominent, it starts building its infrastructure, and increase its footprint. Which, in turn, makes it less interested in peering with a similarly sized competitor unless the counterpart can help in stashing its transit cost, or reduce latency. This interpretation aligns with our analysis from earlier chapters.

eBay and *Cable One* pair attain the highest felicity score of at least 0.4425 for all three sorting criteria. Similarity scores of this pair justify our discussion above as the numbers are 0.01114 (based on prefix), 0.03449 (IP address), and 0.6667 (pop count). Being a CP, *eBay* neither provides Internet connectivity to end-customers nor has a big list of IP prefixes, while *Cable One* provides the last-mile services for many end-users as an access ISP. Generally, CPs treat carrier ISPs as vendors, and it is beneficial for a CP to peer with as many access ISPs as possible. For *Cable One*, peering with a CP will save its transit service cost, allow upgrading the capacity more efficiently, and, at any time, if congestion happens, it can manage the traffic more efficiently.

5.5.2 Validation

To validate our algorithm's performance, we compared the results with the CAIDA data, and show the performance in Table 5.4. The second and third columns show the number of all possible ISP pair combinations and algorithm-suggested pairs that might be interested in peering as they had a felicity score greater than zero, respectively. The numbers in the last column of the table represent the number of pairs that are already peering in the Internet, according to CAIDA. The bracketed numbers right next to them indicate the ISP pairs that were peering and were successfully classified by our algorithm. Figure 5.13 plots the ISP pairs that our algorithm identifies as (*suggested*) potential pairs vs. (*already existing*) peering pairs as labeled by CAIDA with their felicity scores.

Table 5.4: ISPs possible peering combinations, suggested ISP pairs for peering, CAIDA verification based on empirical data, and in bracket, successful identification of pairs that are peering according to CAIDA

	Possible ISP pairs combination	Pairs we suggest for peering	Pairs found in CAIDA
Access (A)	21	17	4 (4)
Content (C)	21	9	0 (0)
Transit (T)	28	17	20 (15)
A-C	49	20	13 (6)
C-A	49	25	13 (9)
C-T	56	41	24 (20)
T-C	56	41	24 (20)
A-T	56	35	23 (15)
T-A	56	38	23 (18)

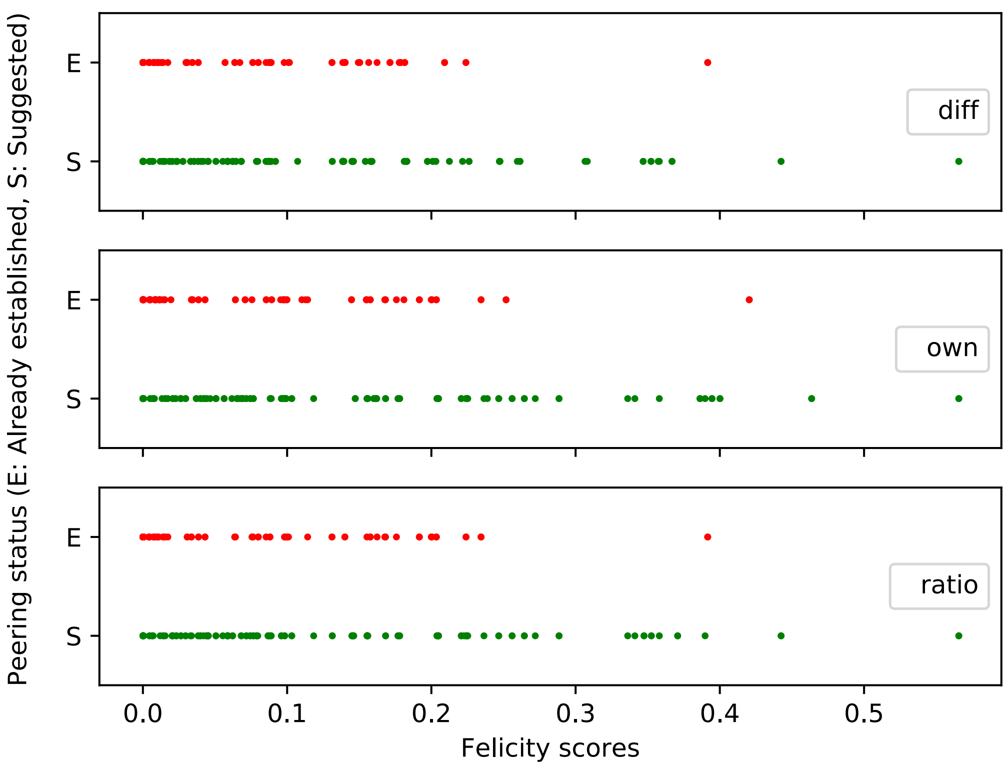

Figure 5.13: Comparison of ISP pairs peering status with CAIDA

One key finding from this figure is, ISP pairs that were actually peering in the real Internet market do not necessarily produce higher felicity scores. This behavior is counter-intuitive from what we were anticipating and assisted us to revise the threshold for potential peer suggestions as most of pairs that were peering exhibit felicity scores of less than 0.3.

Figure 5.14 summarizes the accuracy of our algorithm. We successfully identified 74.3% of the peering pairs from CAIDA with a false negative percentage of 25.7% (37 already peering pairs were not suggested). Figure 5.14 also compares the effect of not considering traffic ratio as a peering candidate discriminator. We ran the APC-generation algorithm without any traffic ratio restriction and got an almost identical outcome.

We found that our algorithm performed best to identify the peering possibility in which a transit ISP was involved. Though the algorithm suggested only 17 pairs for transit providers, 15 of them are actually peering. Similarly, out of 24 pairs of existing peering deals between content and transit ISPs, our algorithm successfully identified 20 cases. CPs do not peer among themselves as they treat each other as rivals and have no incentive in peering. We consider these as false-positives. Although the overall accuracy is 74.3%, our algorithm performed worst to identify peering between access and content ISPs. Almost 50% of the cases peering was not suggested even though they were actually peering. The explanation might be multi-folded. One, CPs are heavy outbound, meaning the incoming traffic ratio would be higher for access ISP, and as such, this deal was rejected, two, that content provider had smaller coverage area compared to itself, although overlap in this particular pairs' coverage area does not necessarily change the peering behavior, or three, they might have engaged in an RP or a paid-peering relationship. We need to explore these scenarios further.

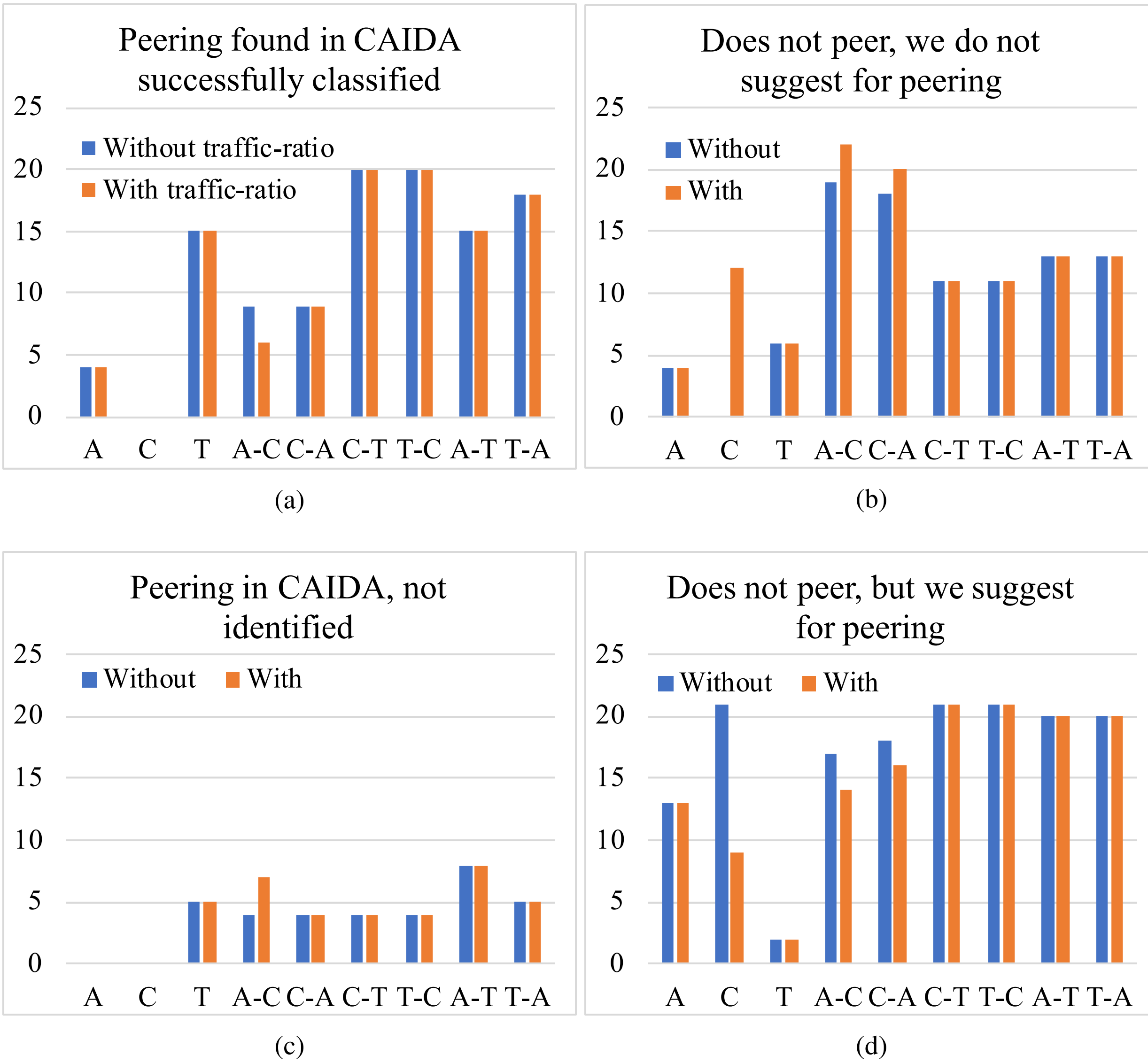

Figure 5.14: Impact of traffic-ratio in CAIDA validation for APC generation algorithm. (a) and (b) show correct classifications, (c) shows false negative, (d) shows false positive cases

5.5.3 Candidate Recommendation

From an ISP administrator's point of view, the algorithm generates a list of top three candidates, and for each candidate, it also recommends the best PoPs where peering can happen. The algo-

rithm uses combined willingness scores (calculated using Eq. 5.12) to suggest the relevant PoPs. Figure 5.15, Figure 5.16, and Figure 5.17 display the ISPs' willingness scores of APCs for different ISP pairs. In all three figures, the first ISP is the requester, and the second one is the candidate. The algorithm always emphasizes on requester's APC sorting criteria. In Figure 5.15, the algorithm sorts the APCs comparing *NTT*'s offloaded traffic amount and nothing else. As for this experiment, we set *NTT*'s sorting criterion to be maximizing its outbound traffic. Figure 5.16 employs a different sorting criterion than Figure 5.15 and prefers the higher out vs. inbound traffic ratio for *Cogent*. In Figure 5.17, we plot the APCs in the ascending order of out vs. incoming traffic difference for *eBay*. Among all possible ISP pairs, we found that most (237) of them preferred the *own* sorting criteria.

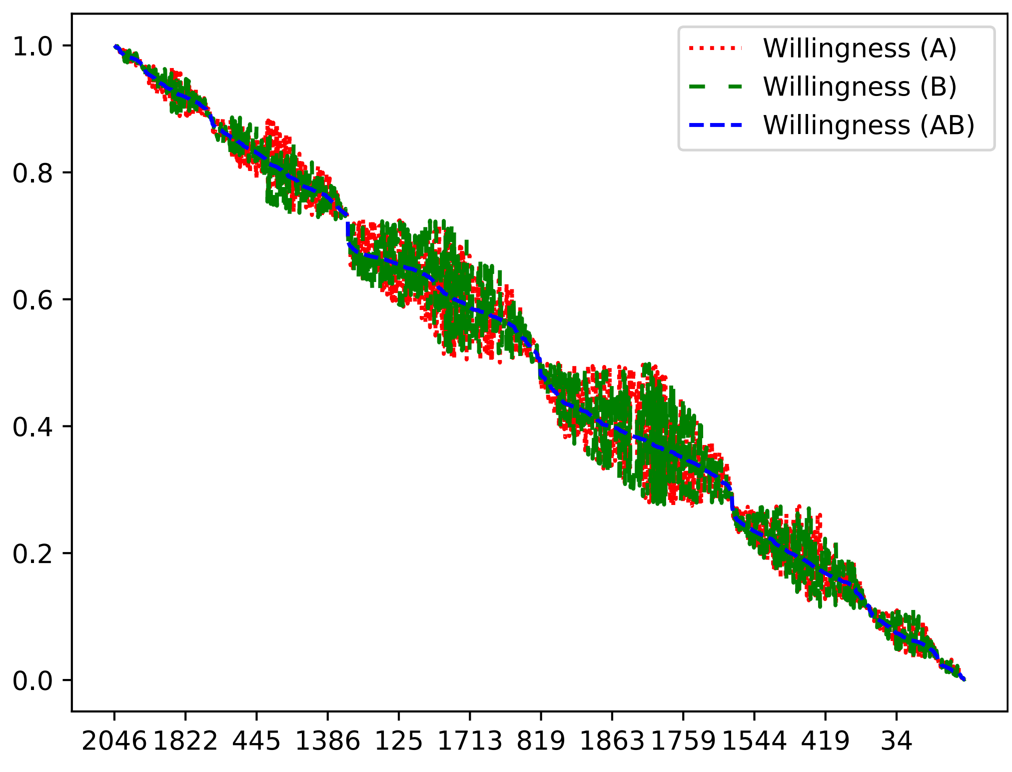

Figure 5.15: Maximize own outbound traffic for NTT (AS2914) and Comcast (AS7922)

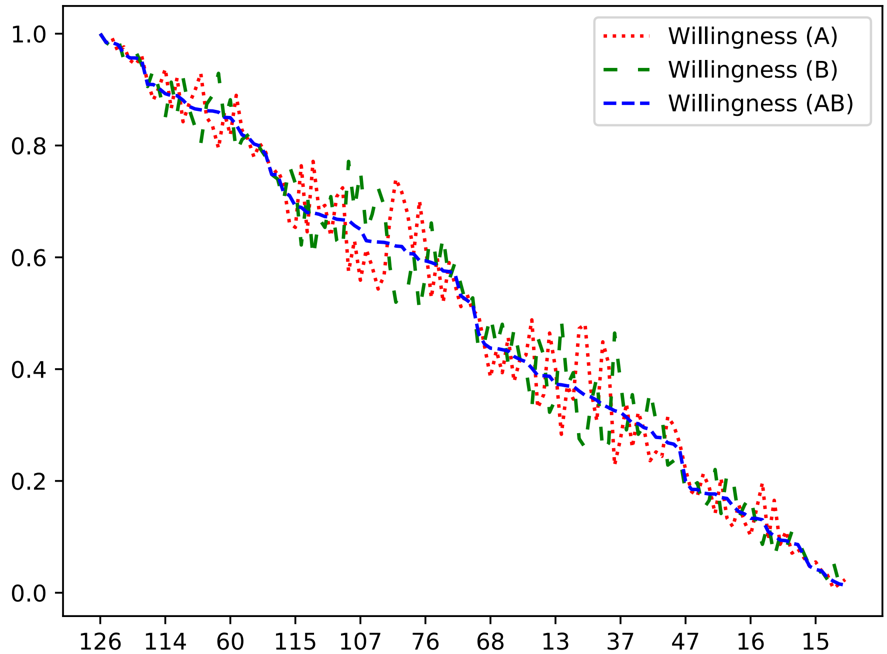

Figure 5.16: Prefer higher out vs. inbound traffic ratio for Cogent (AS174) and Netflix (AS2906)

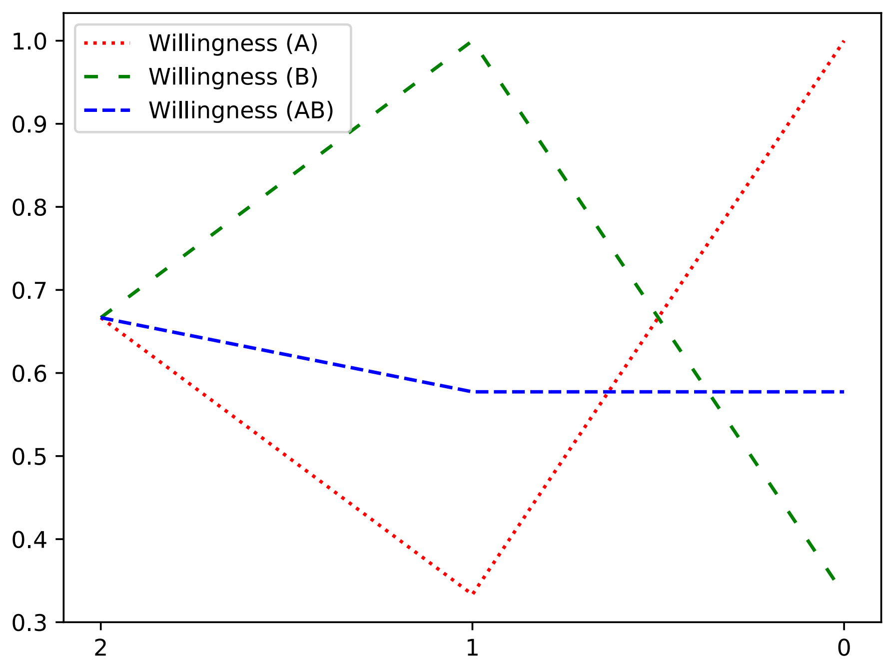

Figure 5.17: Minimizing out vs. inbound traffic difference for eBay(AS62955) and CableOne(AS11492)

In our experiments, we found *Columbus* receives the highest felicity score of 0.41 for candidate *TDS*. The top candidates for *Columbus* are *TDS, Windstream, and Cox*. For peering with candidate *TDS*, we suggest the best peering IXP locations with their PeeringDB URL. Note, "peeringdb.com/" should be added before all of them.

APC1 Equinix Chicago (ix/2), Equinix Ashburn Exchange (ix/1), Digital Realty NYC (fac/16)

APC2 Equinix Chicago (ix/2), Equinix Ashburn Exchange (ix/1)

APC3 Equinix Ashburn Exchange (ix/1), Digital Realty NYC (fac/16)

CHAPTER 6: HYBRID CLOUD INTEGRATION OF ROUTING CONTROL AND DATA PLANES

Because the complete separation between data and control planes in the router may not offer the best solution when it comes to scalability and flexibility [198], new architectures involving hybrid separation of these planes are already attracting the attention [177]. In this chapter, we first describe Cloud-Assisted Routing (CAR) architecture in Section 6.1 focusing on the placement of data plane functions to a remote cloud platform. Section 6.2 articulates the challenges and architectural opportunites of the hybrid SDN model supported by the cloud to gain more scalability and robustness. Section 6.3 covers the possible failure and loop scenarios for data plane delegation along with highlighting potential avoidance mechanisms, while Section 6.4 details our prototyping effort of CAR implementation on a real network. Section 6.5 presents the cost models for CAR and legacy routing, followed by a detailed price comparison analysis for all variables associated with the cost models in Section 6.6. We show the break-even points and compute the CAR savings in Section 6.7. Section 6.8 investigates the impact of peering decisions between two ISPs on CAR.

6.1 CAR Architecture Model

An architectural view of the hybrid "CAR router" is illustrated in Figure 6.1. It is to be noted that CAR [80] aims to find a middle ground where it can exploit both the local hardware to scale router performance and an utterly cloud-based approach for a highly flexible routing service. Following the basic SDN architecture presented in Figure 6.2, wider area SDN deployments will require southbound API protocols to be implemented over longer distances. CAR is looking at the case when the southbound API may need to be implemented over public transit.

Most of the content of this chapter previously appeared in proceedings [81] and journals [82].

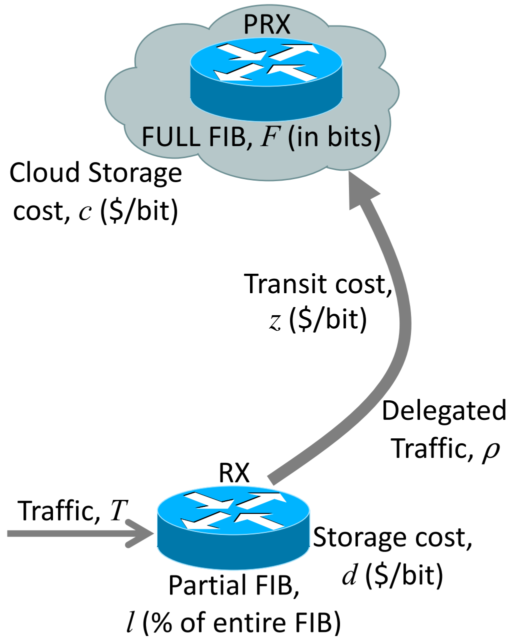

Figure 6.1: CAR architecture

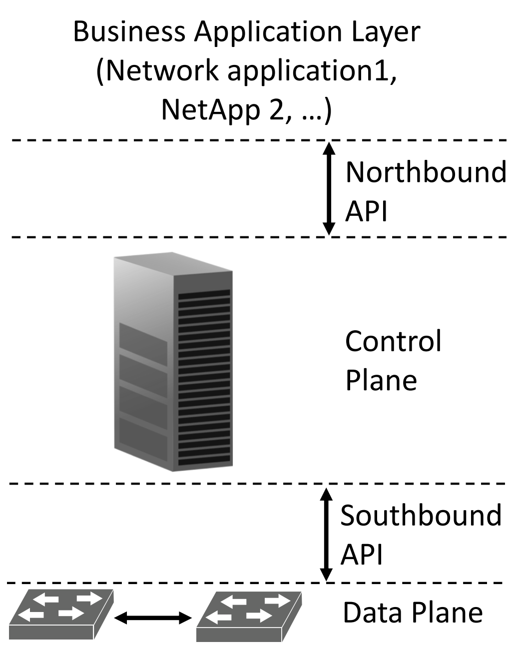

Figure 6.2: Basic SDN architecture

Akin to how virtual memory systems use secondary storage to maintain full content, CAR uses the cloud to implement the full functionality of *Router X (RX)* and keeps *RX* as 'active' while *Proxy Router X (PRX)* as 'passive'. CAR follows a similar approach of RouteFlow [153], previously known as QuagFlow. The differences are, in CAR, *a)* the controller sits on the cloud, and both *RX* and *PRX* can act as separate entities, and *b)* *RX* and *PRX* are capable of establishing BGP peering with others by themselves. CAR designers should follow these two principles [80]:

1. Computationally intensive but not-so-urgent tasks (e.g., BGP table exchange during peering establishment, shortest-path calculations, spanning all entries) should be offloaded to the cloud as much as possible. It is *PRX*'s responsibility to store the full FIB table, and in an unlikely event of *RX*'s failure to handle data and control plane functions, *PRX* should act as the default point of service. Checking the drop packet statistics in *RX*, periodically, or after a specified time interval, *PRX* should send updated FIB information to *RX*.

2. Keep data plane mostly at the *RX* while some of the control plane operations such as on-demand route computations due to failures, collection of flow-level simple statistics, or request for updated routing table will still be done at *RX*. However, CAR should orchestrate heavy routing optimizations at *PRX*.

Even though *PRX* can be designed to keep only the remaining not-so-popular prefixes that *RX* does not store, such implementation is not advisable since this will generate a higher volume of exchanged CAR messages between the proxy and the active routers. During every single *route update* phase, *RX* has to share its *partial* FIB with the cloud so that *PRX* can populate a new list of popular prefixes combining its own table and the table it received from *RX*. While in the "full list stored in *PRX*" case, *RX* can refrain from sending its FIB table to *PRX* and reduce the CAR message size.

With all the benefits that SDN offers, it is imperative to understand the scope of its flexibility and scalability. The fundamental premise of SDN technologies has been to offer high flexibility, i.e., programmability, and updating the overall functionality without modifying much of the underlying infrastructure. By delivering software-based abstractions, SDN has proven to be a successful approach to manage many network components with low labor costs. Yet, its slower packet lookup times and increased overhead of abstractions hinders its overall scalability and affects network's agility. Figure 6.3 illustrates the fundamental trade-off between routing scalability and flexibility.

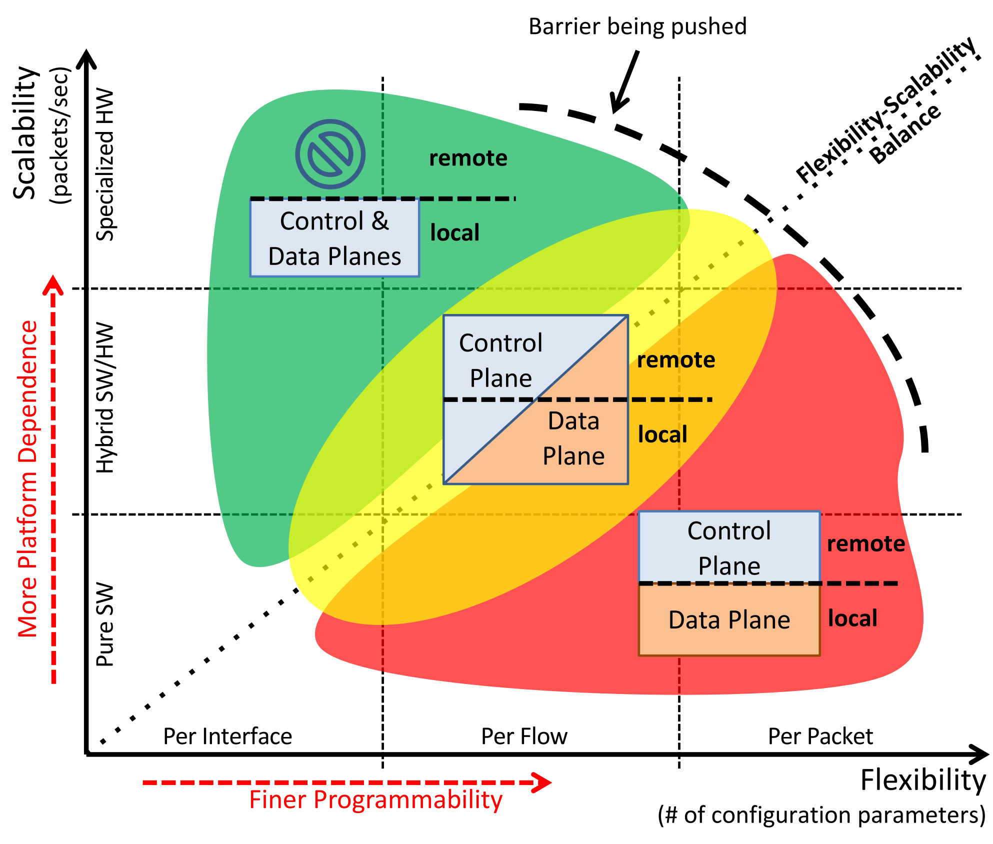

Figure 6.3: Routing scalability and flexibility trade-offs

The more we use specialized ASIC-based (e.g., in high-end Cisco/Juniper/HP switches) designs, the more scalable routing becomes. But, such designs are more platform-dependent and restrict the modification of inner functionality. On the horizontal dimension, the routing design's programmability increases. That is, the network administrator can define finer granularity rules, the coarsest being 'per interface', and the finest being 'per packet'. Yet, there is a fundamental trade-off between these two dimensions of flexibility and scalability.

Most of the current SDN solutions are per-flow approaches (e.g., OpenFlow [140] or Click [120]) with a clean separation between the control and the data planes. This approach falls below the flexibility-scalability balance (the diagonal line in Figure 6.3) in the design landscape. We argue for a hybrid architecture, where local router (*RX*) will have *most* of the data plane functionality while offloading *partially* to a remote software proxy controller/entity (*PRX*) and vice-versa for the control plane. Hybridization of the control plane, for example, means that calculation of most of the shortest-paths will be done at *PRX* while a small portion of them (with high importance) at the *RX*. Such trend of getting a small portion of control plane tasks at local router *RX* is already taking place in recent versions of OpenFlow. Simple counters and statistics collection on flow table entries is now being done at local router *RX* and reported to the controller [109], rather than relying on polls from the controller.

The hybridization in the data plane means that some of the packet forwarding tasks will potentially be delegated to *PRX* (discussed later, in Section 6.3). The frequency of such delegation will depend on the size of the partial Forwarding Information Base (FIB) at the local router *RX* as well as FIB caching techniques determining which entries from the full FIB table at *PRX* will be maintained at the partial FIB in *RX*. We argue that delegation of data plane tasks to the remote *PRX* will be a crucial step in advancing offloading of low-level networking functions to powerhouses like cloud computing platforms.

We think that such hybrid designs of the control and data plane tasks will be more attractive as cloud computing platforms are becoming more central to the network management. So, we consider a hybrid design where PRX is located in a cloud platform which, *may* be *i)* reachable over the public Internet, *ii)* operated by a separate business in return of CAR service fee, and *iii)* tens of milliseconds away from *RX*. In the following sub-sections, we shall discuss the key issues involved in such a hybrid integration of the cloud into routing functions.

6.2.1 Flexibility

Pushing the full control plane to separate third-party cloud providers offers new flexibility advantages in managing many routers. SDN provides better management of network middleboxes by eliminating the requirement of static policy setup. One prominent issue regarding middleboxes in SDN architecture is their proprietary behavior, which in return prevents the controller from setting proper forwarding rules due to the lack of overall topological information. However, as proposed [180], the cloud can act as a trusted third-party for providing a secured solution for such problems.

According to earlier work [177] and as portrayed in Figure 6.3, specialized ASIC based routers offer tremendous scalability as they are manufactured specifically to handle 10+ Gbps traffic. Due to the proprietary and black-box nature, they are not programmable, which is a notable drawback regarding their flexibility. Instead, typical SDN routers running on OpenFlow (or even Click routers) are very flexible and open for any sort of modification to support per interface, or even per flow based packets; albeit, digging into each packet is a highly computation-intensive job that may not be supported by any individual routers. With the power of parallelism in computation, and being backed up with a huge memory and processing power, cloud platforms can host this type of control plane task very efficiently.

Elasticity in on-demand resource allocation makes cloud a natural choice for supporting the seam-

less integration of multiple ASes with different programmable interfaces. Another optimization that can be achieved from the hybrid model if the same cloud is supporting multiple ASes is cloud provider itself can find inter-AS shortcuts and delegate the packets to the destination. Unfortunately, this kind of multi-hop optimization is restricted in legacy SDN due to its clean separation.

6.2.2 Performance and Scalability

Cloud computing, despite being mainly data center centric, intends to offer a distributed computing infrastructure that can be scaled globally. We follow the evident trend of cloud-based networking and explore the possibility of circumventing non-trivial complexity and the cost of legacy routers at the Internet core by using cloud services. Current SDN designs face two major scalability problems:

(1) *Fitting ever-growing BGP routing entries into FIB tables* remains an age-old challenge [38]. Because of the temporal and spatial data locality [117], it would be interesting to observe the performance of a hybrid approach where a local *RX* keeps a *partial* of the entire FIB table while storing the *full* FIB in a remote cloud. If engineered well, only a small portion of the data traffic will have to be delegated to *PRX*. If this delegated traffic is kept under control, the delay due to the delegation and its potential effects on fairness to data flows could be minimal.

(2) *Scaling controller for a larger number of switches and lower latency.* Tapering the overload on the controller and the communication latency between controller and switches are critical challenges. Till now, OpenFlow involves the controller to decide for every new flow that comes to the switches. As discussed earlier, existing OpenFlow controller (e.g., NOX [103]) can support up to 100K requests/sec [114]. Cloud can be a perfect fit to scale up the number of supported requests coming from a bigger network with more switches.

In terms of latency, our hybrid approach can potentially degrade the performance due to the delegated traffic flows having to travel extra path (from *RX* to *PRX*) to reach the cloud. As new players join the business of providing new cloud-assistance to routing, it is more likely to identify a cloud that is closer and has higher quality. For instance, one can use CloudCmp [126] to identify the nearest cloud provider with the lowest possible latency in response time for various computations and make sure the local router delegates the packets towards that cloud if necessary. On the positive side, such delegation of data traffic effectively makes routing a multi-path one, and the overall end-to-end delay may even improve, as have been observed in recent experimentations [80]. With careful engineering, we feel the drawbacks can be solved in return for more overall performance benefit.

6.2.3 Privacy, Security, and Fairness

Since *PRX* needs to communicate control information with *RX*, it is mandatory to maintain a secure tunnel between them. A separate TCP session or a GRE tunnel can be implemented between them to achieve the security. Further, originating from SDN's trend of "centralized brain", *PRX*s may be treated like a honey-pot by the attackers due to their locations (i.e., the cloud), and the single point-of-control-like functionality. Denial-of-service attacks will be more challenging to deter from these cloud points since they will likely not just serve routing assistance but other more generic services too. Dedicated sessions from *PRX*s to their corresponding *RX*s may be the only practical solution for these security challenges.

A different but relevant vulnerability may arise due to the basic SDN design, which involves supporting multiple ASes from the same cloud vendor. If the *PRX*, located at the cloud, treats all the *RX*s from different ASes just the way a controller would operate all of its switches in a single data center setup, then there is a chance that flow rules might get propagated to the wrong recipients.

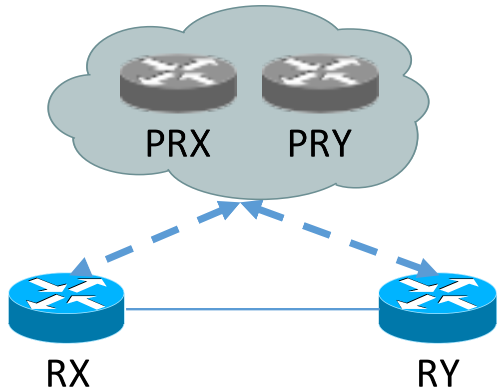

Figure 6.4: Single cloud vendor hosting multiple proxy

The provider needs to isolate the service layer agreements (SLA) of different ASes and has to come up with a solution similar to FortNox [167] for preventing possible inconsistency. For a scenario presented in Figure 6.4, the cloud provider can resolve inter-domain policy conflicts with the capability of monitoring and troubleshooting any issues more easily as multiple ASes share their policy with it.

Another issue is the fairness of traffic flows. Since some flows will be delegated to *PRX*, they will experience more delays. To comply with net neutrality expectations, operators configuring the hybrid cloud integration will need to ensure fair treatment of all flows irrespective of content, source of origination, and size. *RX* should follow non-discriminative *traffic shaping* to support the common practice of treating the flows equally. As previously found [117], there exists a high locality (10% popular prefixes are responsible for 97% of traffic), and the designer should try to support these flows locally. Yet, the fairness concerns require going beyond a straightforward selection of the heavy hitter flows but also consider caching small hitter flows belonging to unpopular prefixes. The overall routing experience of the flows (heavy or small) should be roughly similar. Techniques like punching holes [94] and supernetting [95] will have to be actively devised to attain such balance.

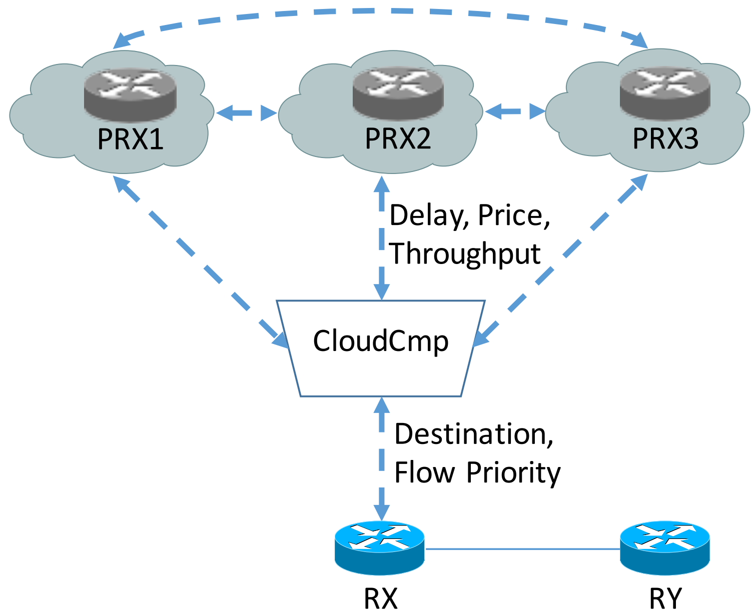

Figure 6.5: Hybrid design using multiple cloud vendors

6.3 Delegation of Data Plane

Legacy SDN designs did involve delegation of control plane tasks to remote platforms, but hybrid integration of routing functions with cloud support implies delegation of data plane tasks. We envision the delegation of data packets to *PRX*, which brings new challenges and opportunities.

6.3.1 Better Failure-Resiliency

Delegating data packets to cloud platforms and letting them further forward the packets open up exciting opportunities in terms of the robustness of routing. As shown in Figure 6.5, hybrid cloud integration can mirror *PRX* in multiple clouds. Then, *RX* (using an intermediary similar to CloudCmp [126]) can choose the best mirrored *PRX* based on the destination or priority of the flow, and service prices of the cloud providers hosting the mirrors.

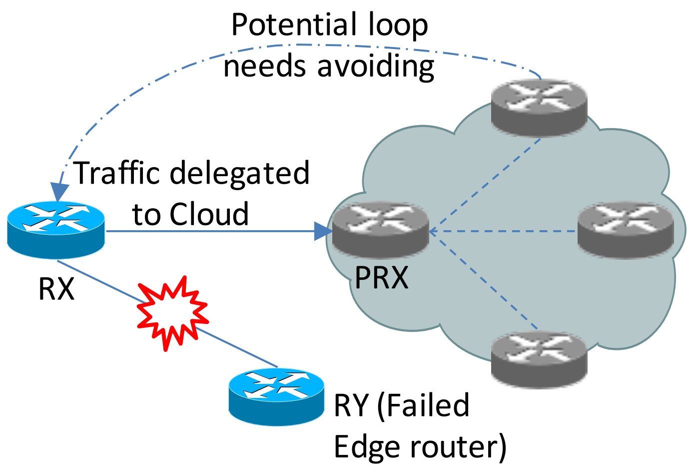

Figure 6.6: Traffic re-route due to failure

In conventional routing, if a router fails, a significant amount of traffic needs to be rerouted to an alternate path or through nearby devices. There are already lots of well-researched proposals to achieve better resilience by analyzing link failures [205, 143], designing over-provisioning models [144] for the legacy network. Figure 6.6 shows the packet delegation procedure in hybrid architecture where *RX* delegates packets towards its proxy when the neighboring link with router *RY* (this can be another hybrid SDN router or a legacy router) fails.

As in Figure 6.4, a single cloud provider can host proxy routers for multiple routers. The packets delegated to *PRX* can be handed to *PRY* if they are destined to *RY*. Cloud providers can improve the overall end-to-end forwarding delay of such delegated packets if they serve more such hybrid routers – perhaps to the extent that delegated packets may arrive faster. Such a centralized role of clouds in data plane tasks may significantly improve the overall robustness and efficiency.

6.3.2 Attaining Loop-Freeness

Due to the development of SDN designs with clean control plane separation, a prominent issue arises is whether there can be such a situation when a packet never reaches its final destination because of the loops [109]. It is always adverse to have a network inconsistency that has been generated due to the complex chaining of instruction. Similar concerns apply for the proposed hybrid SDN architecture. While designing such a scheme, network designers should be careful about possible inconstancies, particularly when the data traffic delegated to *PRX* is using public Internet routing.

Considering the existence of multiple hybrid SDN routers on a path, two possible loop scenarios may arise due to data traffic delegation:

6.3.2.1 Scenario A: While reaching PRX From RX

While delegating traffics towards *PRX*, a loop may emerge if the Internet routing chooses RX as the next-hop. This scenario is not possible unless there is an inconsistency in the Internet routing. Assuming that the delegated traffic will go through a GRE tunnel, the shortest path towards the destination of that tunnel (i.e., *PRX*) will not traverse *RX* in steady-state routing.

6.3.2.2 Scenario B: Between PRX and Destination

Once delegated to *PRX*, it is *PRX*'s responsibility to forward the data traffic towards its destination via the appropriate shortest-path Internet routing. As we can notice from Figure 6.7, such *delegated* traffic may get delegated again to another proxy router *PRY*. In general, such traffic may encounter a loop if the distance between the destination from *RX* (d_{RX}) is shorter than from *PRX* (d_{PRX}).

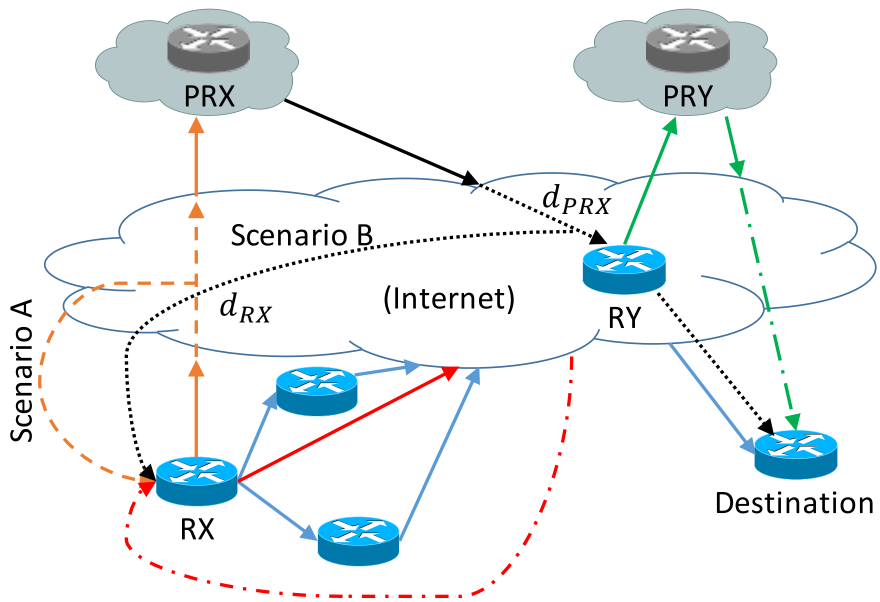

Figure 6.7: Possible loop scenarios

Further, a loop exists if the shortest path from *PRX* to the destination (SP_{PRX}) includes the shortest path from *RX* (SP_{RX}). So the following conditions should never be satisfied to attain a loop-free architecture:

1. *Necessary condition:* $d_{PRX} \geq d_{RX}$
2. *Sufficient condition:* SP_{PRX} includes SP_{RX}

One approach to prevent such loops is to *only* allow delegation of prefixes to *PRX* for which *PRX*'s forwarding table has strictly less cost towards the destination, i.e., $d_{PRX} < d_{RX}$. This inequality condition will ensure that the necessary condition for the loop is never met. Yet, such a preventive design may be too restrictive in terms of caching of prefixes in *RX* and limit traffic engineering possibilities.

In this regard, we shall keep in mind that, the above scenarios are for multi-hop design. We can

easily avoid loop from a single-hop architecture by keeping a separate lookup table in cloud where *PRX* resides (similar to BGP's poisoned reverse) and make sure the next-hop is not *RX*.

6.4 An (SDN-Backed) CAR Prototype

To show the proof-of-concept of CAR architecture, we stripped off some of the complexity from the original design and developed a simple prototype of CAR using SDN. The top-level architectural overview of our prototype is portrayed in Figure 6.8. In this figure, *Source, NH1, NH2*, and *PRX* all represent individual ASes and have their external BGP routers. *RX* implements SDN architecture internally, has an internal BGP speaker, and, from the BGP perspective, appears as a single AS. We used FRRouting (FRR) [129], an open-source routing protocol suite, on virtual machines, each running Ubuntu version 18.04.1 LTS, to act as BGP routers.

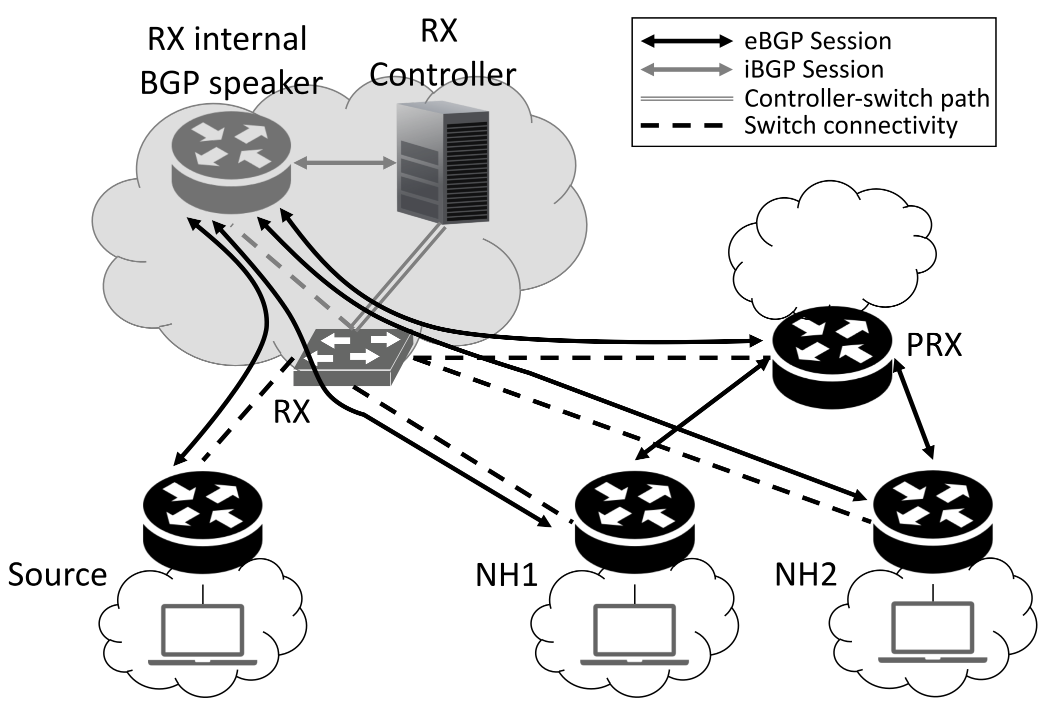

Figure 6.8: CAR prototype using RYU SDN controller

We used Ryu [10] controller to serve our objective as it has a BGP speaker library that we utilize. For the prototype purpose, we wrote our custom Ryu controller application that creates a BGP instance and maintains an iBGP session with the internal BGP speaker (a separate BGP router). We shall refer to this application as *CARbgp* application. The internal BGP speaker's only task is to maintain the eBGP sessions and receive the BGP announcements from peering ASes. We used this separate BGP router for managing the external communications to prevent any direct interaction between the controller and peer networks. This design choice also helps us keep the controller isolated and secured by not exposing it to the outside. Our custom application installs flow rules in the RX-switch, an Open vSwitch(OVS) so that only BGP control messages coming towards the internal BGP speaker are allowed through its data plane. Once the BGP sessions are established, external peers start announcing their prefixes, which are eventually received by the controller and sent to *CARbgp* application to handle.

Each time *CARbgp* receives a new prefix announcement from an external peer, it adds a new flow rule in the switch for that. *CARbgp* maintains a one-to-one mapping of RX-switch ports and external peer routers connected to it. Thus, it is easy to add new flow rules to forward associated packets through its announcing router's connected port. To follow the routing activity, we also decrement the packet TTL by 1 and update the destination MAC address with the next-hop router.

Our main argument for CAR is that RX does not need to keep all the entries in its FIB. In a real-world scenario, if RX can not serve a prefix, it will forward that packet to PRX, as PRX will have the entire FIB table. The actual network architecture between RX and PRX will be similar to Figure 6.9. Since in CAR, PRX will have the exact copy of RX's entire FIB, it is not mandatory that NH1 and NH2 have to be a direct BGP neighbor with both RX and PRX. However, in our stripped version of CAR, we made sure that all the BGP neighbors of RX were also establishing BGP peering relationships with PRX, so that PRX can at least have a direct path for the delegated packets to the destined AS.

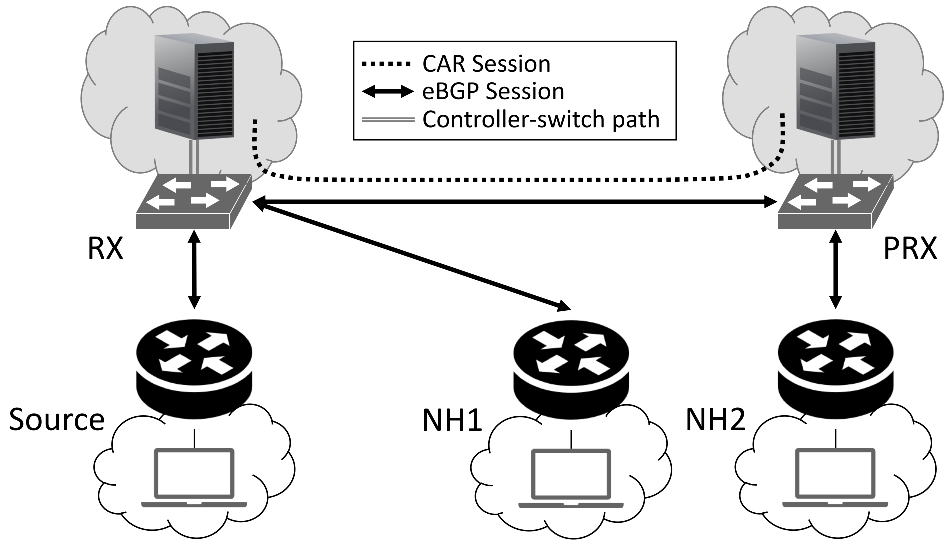

Figure 6.9: CAR prototype using CAR protocol

For the experiment, we employed a *crystal ball* behavior in RX. We assumed, the controller was aware of the most popular prefixes, and would keep those prefixes in its FIB to support the elephant flows to ensure smaller latency for the overall traffic. We varied FIB caching thresholds from one to 20 percent and compared the aggregate delays for them. *CARbgp* always keeps the statistics of flow entries in the flow table. During the experiment, in the beginning, as new prefixes were announced, *CARbgp* added a new flow rule in the RX's flow table for each new prefix until it maxed out the FIB cache size capacity, which, in this case, was the flow rule count.

6.4.1 Experimental Setup in GENI

We set up our experiment in GENI [57], a federated testbed that allows obtaining computational resources from different US locations, and installs various customized software on virtual machines if requested. This is an excellent opportunity for conducting large-scale networking experiments.

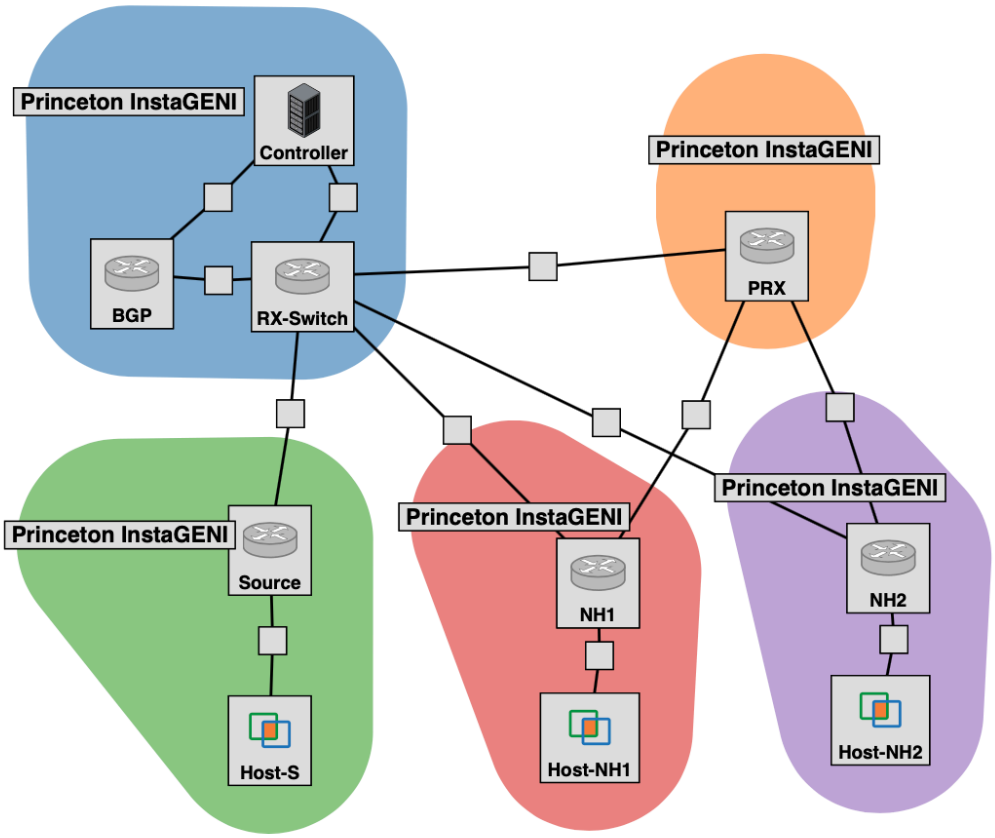

Figure 6.10: CAR setup in GENI

We reserved multiple *slices* [5] from the same *aggregate* [6], Princeton InstaGENI rack, and set up our topology (see Figure 6.10) there. Each slice represented one individual AS. It is also possible to use different slices from different InstaGENI racks (i.e., Cornell, Illinois, Stanford, etc.) and stitch together.

We assigned one single host to each of *Source*, *NH1*, and *NH2* ASes. Both *NH1* and *NH2* announced a list of 100 unique prefixes, separately, while exchanging the BGP announcements. To

[5] A *slice* is the context for a particular set of experiments, and contains reserved resources and the set of GENI users who are entitled to act on those resources [98].

[6] An *aggregate* is a software server that provides resources to clients based on the GENI aggregate manager API. The *aggregate* may provide virtual 'sliced' or non-virtual 'whole' resources to customers [98].

implement this, instead of adding 100 virtual machines, we assigned 100 virtual IP addresses from different subnets to both *Host-NH1* and *Host-NH2*. For example, *Host-NH1* was connected to the *NH1* BGP router and represented the IP block of 192.168.1.0/24 to 192.168.99.0/24, while *Host-NH2* was representing 192.168.100.0/24 to 192.168.199.0/24 networks.

To make our analysis more comprehensive and pragmatic, we used the *Anonymized Internet Traces* collected by CAIDA's *EQUINIX-NYC monitor* [69] during December 2018. We identified the unique IP addresses, and the packet counts destined towards each of them from these anonymized traffic traces. We sorted the IP addresses in descending order of their packet counts and clustered them into 199 groups. We then mapped each group one-to-one, sequentially, with one item from those as mentioned above, 199 distinct subnet IP addresses list. As a result, most popular IP addresses were grouped and are positioned at the beginning of our list of virtual IP addresses. For instance, there were 557,246 unique IP addresses in this particular traffic trace, and the most popular 2,801 IP addresses were grouped to be represented by 192.168.1.1. The rest were computed in a similar manner. We also accumulated the packet counts per group and treated that number as the incoming packet count for the associated virtual IP address. But, the numbers were too high for the first few IP addresses (we shall discuss this spatial locality in the later sections of this chapter), as such, we scaled down all the numbers by dividing them with the smallest number among them. Finally, for measuring the end-to-end delay, we send *ping messages* from *Host-S* to each virtual IP address using its corresponding incoming packet count.

Figure 6.11 plots the average ping response time for the IP addresses. We compare three design choices: *i)* using only BGP routers, *ii)* using CAR architecture, but keeping the full FIB in RX, and *iii)* using CAR architecture with 15% FIB in RX. Surprisingly, if we run the experiment using BGP routers only, the average delay for each IP address was higher than what a packet experienced in CAR. "CAR without delegation" performed the best because the full FIB was available, and related flow entries were already put in the switch when the BGP announcements were received.

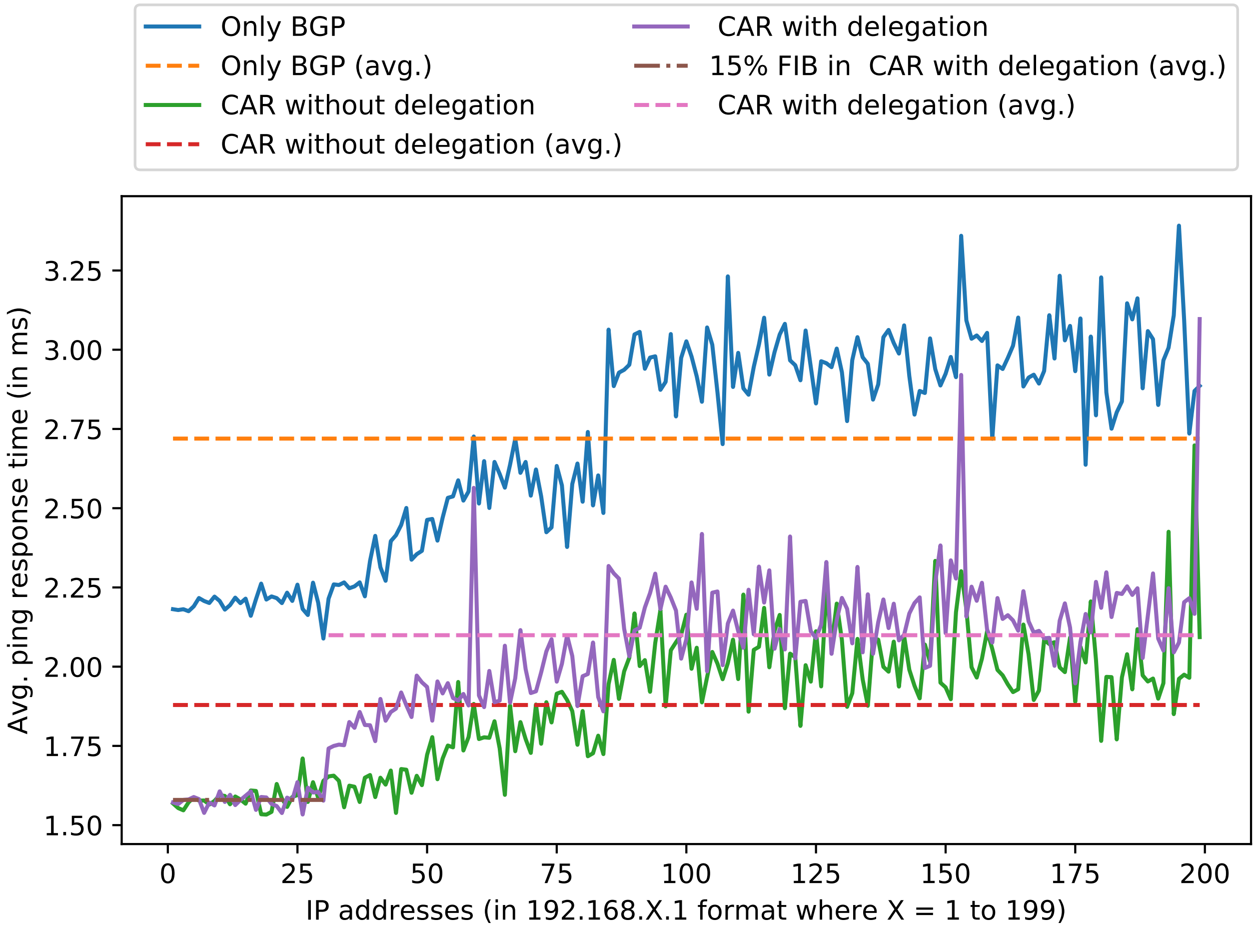

Figure 6.11: Average ping responses of IP addresses

When a packet arrived, the RX-switch could easily forward that to the appropriate outgoing port. "CAR with 15% delegation" showed an expected behavior. The cached prefixes (30, in this case) performed better than the delegated packets, which still outperformed the conventional BGP performance. We can see a rise in the average ping response time for the IP addresses on the right side of the figure. These IP addresses belonged to *Host-NH2* and were on a different virtual host. They always displayed a higher response time compared to *Host-NH1*. Since the physical resources are shared among the GENI users, we think, maybe, this specific machine was experiencing higher traffic volume from other researchers' experiments.

```
[redacted]:~$ traceroute 192.168.31.1
traceroute to 192.168.31.1 (192.168.31.1), 30 hops max, 60 byte packets
 1  Source-link-3 (192.168.200.254)  0.645 ms  0.573 ms  0.428 ms
 2  10.20.10.101 (10.20.10.101)  3.255 ms  3.141 ms  3.058 ms
 3  NH1-link-6 (10.20.20.1)  4.319 ms  4.064 ms  4.096 ms
 4  192.168.31.1 (192.168.31.1)  4.457 ms  4.340 ms  4.252 ms
```

(a) BGP Only

```
[redacted]:~$ traceroute 192.168.31.1
traceroute to 192.168.31.1 (192.168.31.1), 30 hops max, 60 byte packets
 1  Source-link-3 (192.168.200.254)  0.482 ms  0.422 ms  0.348 ms
 2  * * *
 3  NH1-link-6 (10.20.20.1)  2.282 ms  2.192 ms  2.091 ms
 4  192.168.31.1 (192.168.31.1)  2.975 ms  2.905 ms  2.840 ms
```

(b) CAR without delegation

```
[[redacted]:~$ traceroute 192.168.31.1
traceroute to 192.168.31.1 (192.168.31.1), 30 hops max, 60 byte packets
 1  Source-link-3 (192.168.200.254)  0.657 ms  0.485 ms  0.402 ms
 2  * * *
 3  PRX-link-7 (10.20.110.111)  2.053 ms  2.072 ms  1.999 ms
 4  NH1-link-6 (10.20.20.1)  2.239 ms  2.178 ms  2.102 ms
 5  192.168.31.1 (192.168.31.1)  2.811 ms  2.738 ms  2.663 ms
```

(c) CAR with 15% delegation

Figure 6.12: Sample traceroute from Host-S (192.168.200.1) to 192.168.31.1

We kept the experimental setup in GENI the same for all three scenarios. For the "BGP only" experiment, RX-switch acted as a simple layer-2 switch, and incoming packets from *Source* went to the internal BGP router residing inside *RX* and then traversed back via RX-switch to either *NH1* or *NH2*. This incident can be one possible explanation for BGP's higher delay as a packet had to travel extra in this specific scenario. But, in the case of CAR architecture, the packets do not go back-and-forth between RX-switch and the internal BGP router except for the BGP control plane messages. During the BGP establishment process, *CARbgp* becomes aware of the prefixes and put related flow rules in the flow table of the switch for them. So, the switch was able to make an immediate decision.

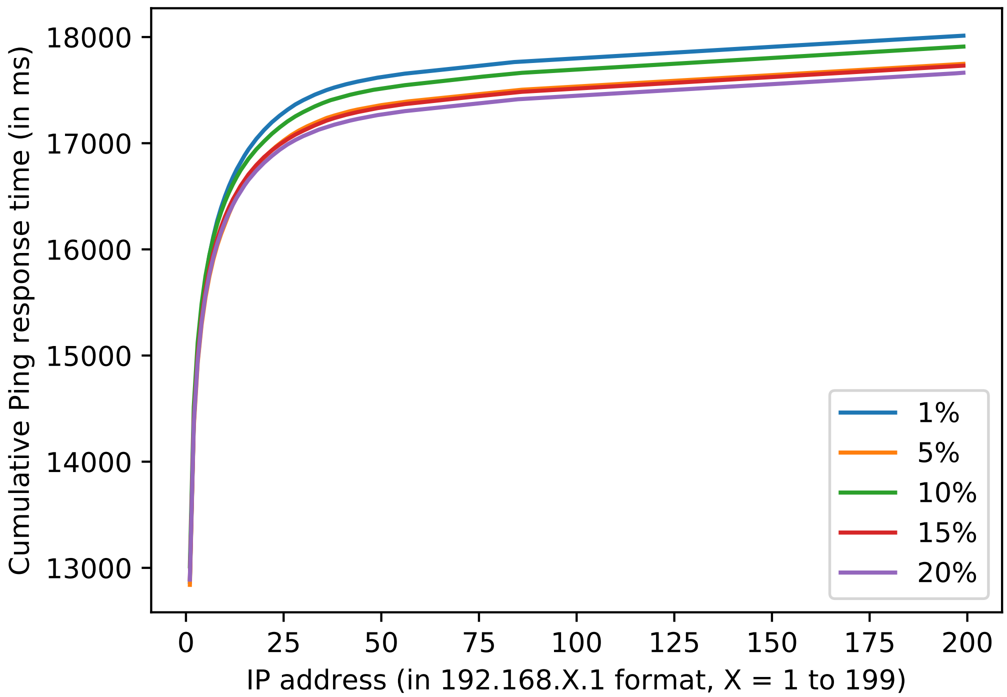

Figure 6.13: FIB caching performance comparison

Figure 6.12a, Figure 6.12b, and Figure 6.12c show the *traceroutes* for the same destination (IP 192.168.31.1), from *Host-S* (IP 192.168.200.1), for the three different experimental setup that we had. In Figure 6.12a ("BGP only" setup), RX-switch is not doing anything, and as such, from traceroute, there is no existence of it; rather, we can see the internal BGP router's IP address, which is residing inside RX's network. Contrary to it, in the CAR version, we hide the IP address of our BGP router, as we can see from Figure 6.12b, and Figure 6.12c. Both the time, the packet travels through the same path, except for "CAR with 15% delegation" delegates the packet to *PRX* instead of *NH1*. Since *NH1* and *PRX* are also BGP neighbors and are connected directly, *PRX* then takes care of the packet and forwards it to *NH1*.

Figure 6.13 justifies the FIB caching effectiveness by comparing the total delay experiences. We plot five different cache-sized "CAR with delegation" to show the overall performance in terms

of delay. We can see that, as we increase the FIB size in RX, the overall delay is reducing. This behavior is anticipated. If we support more prefixes from RX-switch, fewer packets will encounter an extra delay. But, the critical point we want to highlight from this figure is the overall gain. They are very close to each other. We think, because of the first few IP addresses which account for the majority of the traffic, as long as we can support these carefully, the overall performance will not degrade much.

We noticed something intriguing during this measurement process – the performance fluctuated from time to time, depending on what time of the day we are executing the experiment. Since GENI is using real computing devices, maybe, sometimes, congestion happens due to traffic overload and reduces the overall performances. For instance, Figure 6.13 shows the total time required for all ping responses to complete was approximately 18 seconds, but we have found them to complete within 15 seconds too. Nevertheless, the FIB cache performance plots were following the same trends, despite every one showed improved performance at that time.

6.5 CAR vs. No CAR Cost Models

To begin our analysis, first of all, we have constructed a very naive cost model for CAR and the traditional routing system based on DRAM price, transit cost, cloud storage, and service price. However, infrastructure costs like laying a sub-sea cable and purchasing equipment for running the business or overall administrative overhead of technical staffs are ignored in this study, As, we are focusing on long-term consideration, and these costs will eventually be distributed over time. For instance, we do not consider the fact that in 2016, US broadband providers invested approximately $76 Billion in network infrastructure, and the total expenditure from 1996 to 2016 was $1.6 Trillion [64].

Table 6.1: List of symbols used in this chapter

d	DRAM price
c	Cloud storage price
s	Cloud service price
z	Transit price
l	Percentage of FIB cache in *RX*
C	Traditional routing price
$\hat{C}$	CAR price (with cloud storage price only)
$\hat{C}_s$	CAR price (with cloud storage and cloud service price)
F	FIB size
T	Total incoming traffic to *RX*
ρ	Amount of traffic delegated to the cloud
ψ	First constant to represent temporal dynamics in router traffic
ω	Second constant to represent temporal dynamics in router traffic
γ	Scale factor
λ	Percentage of transit traffic

Let, d and c be the \$/bit cost of storage at the local DRAM and the cloud, respectively. Considering, F (in bits) as FIB size and l be the percentage of FIB that needs to be stored at the local router (basically, size of FIB cache) to sustain an acceptable average delay of forwarding time for packets towards the cloud. Here, "acceptable" means the new forwarding delay should be very close to the traditional router lookup delay. Assuming z (in \$/bit) being the delegated packets transmission cost to the cloud, models for the operational cost of traditional router C and CAR router $\hat{C}$ can be formulated as follows:

$$C = dF \tag{6.1}$$

$$\hat{C} = dFl + cF + z\rho(l) \tag{6.2}$$

where $\rho(l)$ is the amount of traffic delegated to the cloud and follows a Log-normal decay distribution function of l due to the significant locality in traffic.

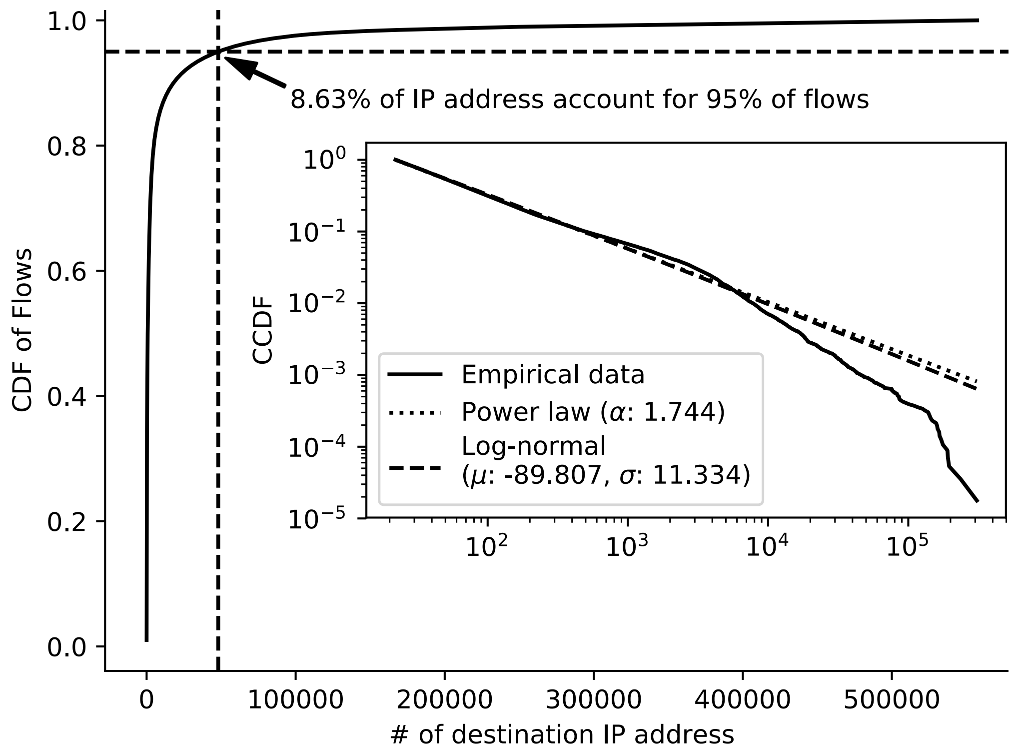

Figure 6.14: IP prefix count vs. flow count

We have analyzed the same *Anonymized Internet Traces* collected by CAIDA that we have used previously in Section 6.4.1, and observed the spatial (few popular prefixes) locality behaviors (see Figure 6.14) displayed by the traffic. The analysis shows that 8.63% of destination IP addresses account for 95% of traffic flows at this major traffic exchange. Previous studies also support this claim of very high locality [117]. Assuming ψ and ω be the two constants for such temporal dynamics of traffic destination in routers and being the total incoming traffic towards *RX*, we can express ρ as:

$$\rho(l) = [1 - \psi \ln(l) - \omega]T \tag{6.3}$$

The more FIB entries we store, as l increases, the more bits will be needed at the local DRAM, and less traffic, $\rho(l)$ will be delegated to the cloud. According to Figure 6.14, it is reasonable to expect

that $l < 10\%$ will be enough to support most of the traffic locally, and a minimal amount of traffic will be delegated. Since the transit cost depends on the amount of delegated traffic, it will hence stay low as long as l is small, e.g., for $l < 10\%$ less than 5% of the traffic will incur transit cost. Thus, the transmission cost (third term in $\hat{C}$ from Eq. 6.2) of the delegated traffic will be relatively low due to its smaller volume.

In the following sections, we use exponential decay to model the cloud storage, and transit costs as historical pricing data show that these services are becoming a commodity. This is in line with the overall trend of bulk storage and communication prices declining exponentially in terms of the per-unit price (e.g., $/bit). This does not mean that the providers will give them for free as the volume of the services customers needs is also increasing. Given this, our analysis aims to reveal which one of the three terms in Eq. 6.2 will be the dominant factor in regulating a CAR router cost. In that sense, since the last two terms have exponentially decaying per unit prices c and z, they will not be the key factors in the overall CAR router cost as long as their multipliers F and $\rho(l)$ are under control. Since F also exists in the first term, the second term will not be the determining factor in comparison to the first term. As for the third term, we observe in Figure 6.14 that $\rho(l)$ decays according to a Log-normal distribution. An exponential decay would mean a faster decline, but the delegated traffic can still be kept small by adequately exploiting the locality in the traffic, as seen in Eq. 6.3 and Figure 6.14. As such, the Log-normal distribution of the delegated traffic decays very fast, and once a small portion of the FIB is cached, $\rho(l)$ will be kept small.

Therefore, the driving factor will be the first term, dFl, which is less than C. Thus, as long as l is managed properly via good prefix replacement algorithms (i.e., FIB caching algorithms), CAR routers will always be more cost-effective. It is worth noting that C will likely have more terms in addition to dF due to the shifting of control plane tasks to remote platforms. Also, note that compressing FIB [131] does not really change the overall comparative analysis here since a similar study can be made involving SRAM costs.

6.6 Price Comparison

Forecasting memory price is uncertain due to its market dependency. Supply-demand mismatch, global politics [139], environmental hazards, or even company policies can impact the price variation. Regardless, this price is certainly not comparable with the current price offered by cloud providers. A back-end cloud service (including storage facility) and a transit service towards the cloud are two critical components of CAR. Historically, the prices for all of these services have been reducing, as we shall see in the following discussion.

6.6.1 DRAM Price, d

We use memory price data from McCallum [138] for our study. This dataset contains memory prices from 1957 (transistor Flip-Flops) to 2017 (DIMM DDR3-1600), but we restrict our starting date from 1984. Figure 6.15 shows the decaying trend in DRAM prices with ups and downs on multiple occasions like 1988-1990 or 1996-1998 periods. In 2013, the price was increased by 40% compared to its previous year, which is not visible in this figure as the recent price of DRAM is extremely low compared to its initial predecessors. To get a more refined view on the recent years, we plot average DRAM prices for the last 10 years in Figure 6.16 separately. It is noticeable that the price started falling after 2014 and reached its lowest ($3.55) in 2016. After that, the price is increasing again and is reported to climb at least 10% on average in 2018 [33].

Major DRAM makers like Samsung, Micron, SK Hynix are undergoing a transition, as they are competing to take the future lead and investing more to produce 18nm-class DRAM instead of 25nm or even 20nm wafers. Moreover, robust and continued demand from the mobile industry who are packing 4GB or 8GB of RAM in a smartphone contributes to the tight supply and thus leading to the most recent price hike.

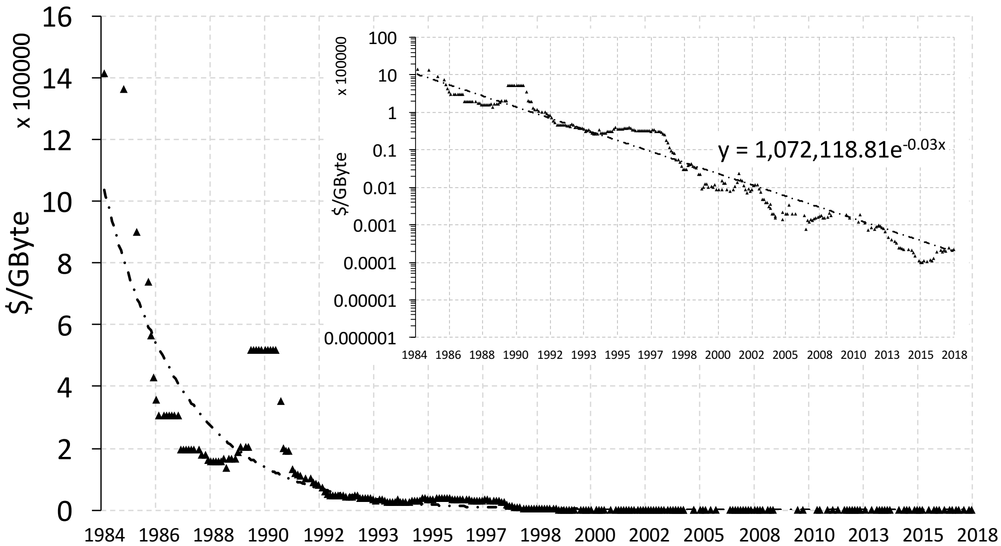

Figure 6.15: Historical DRAM price trend

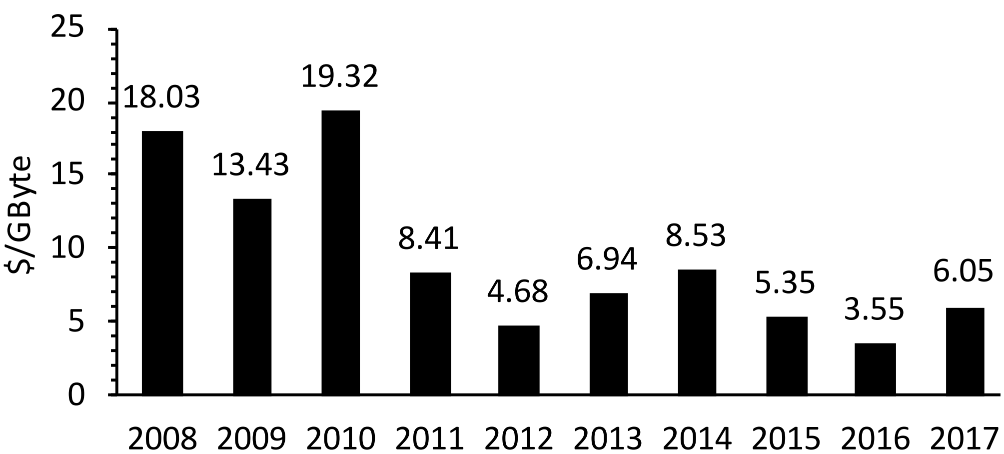

Figure 6.16: Average DRAM price (2008 to 2017)

Considering all these, it is unpredictable whether the (DRAM) memory will be cheaper in the future. According to the trend-line (in Figure 6.15), we expect the price to reduce exponentially with a small decaying exponent. Still, it will not necessarily be more economical in the near future. Based on the DRAM prices in 1984-2016 and favoring the traditional routers, we model

the DRAM price with an exponential decay with respect to time, t:

$$d(t) = 1,072,118.81e^{-0.03t} \tag{6.4}$$

6.6.2 Cloud Storage Price, c

Gartner introduces Magic Quadrant, a graphical representation of research providing a summarized insight about any given market, to position the competitors into four categories: leader, challenger, visionary, and niche. According to them, AWS and Azure are the market *leader*, and Google Cloud is marked as the top *visionary*, who is late to join the market (eight years later than Amazon) but can shift the momentum into its favor anytime [51]. We base on this criterion and will limit our discussion primarily on these three providers.

Cloud storage cost is advertised as per GB per month; even so, users have to pay some associated costs depending on the providers' business strategies. Costs like storage request or transaction count, HTTP operations (i.e., GET, PUT) counts are often referred to as "hidden costs", although service providers mention them in their 'terms & conditions' [18]. For instance, both Amazon S3 and Microsoft Azure charge for both of these operations while Google does not charge for PUT requests. For simplicity, we discard these variable costs and base our analysis only on the storage costs.

Figure 6.17 evaluates the declining price trend for cloud storage from January 2012 to January 2018. One interesting observation we have found, albeit of its absence in the figure, AWS initially launched S3 in 2006 setting the price at $0.15 per GB/month and continued to charge so ($0.14 in 2010) until 2013, which is the year, Google publicly introduced its cloud services with a cheaper price for the first time. To match Google cloud storage plan, AWS halved the price; and since then, these two providers and Microsoft are battling to offer the lowest price to attract new customers.

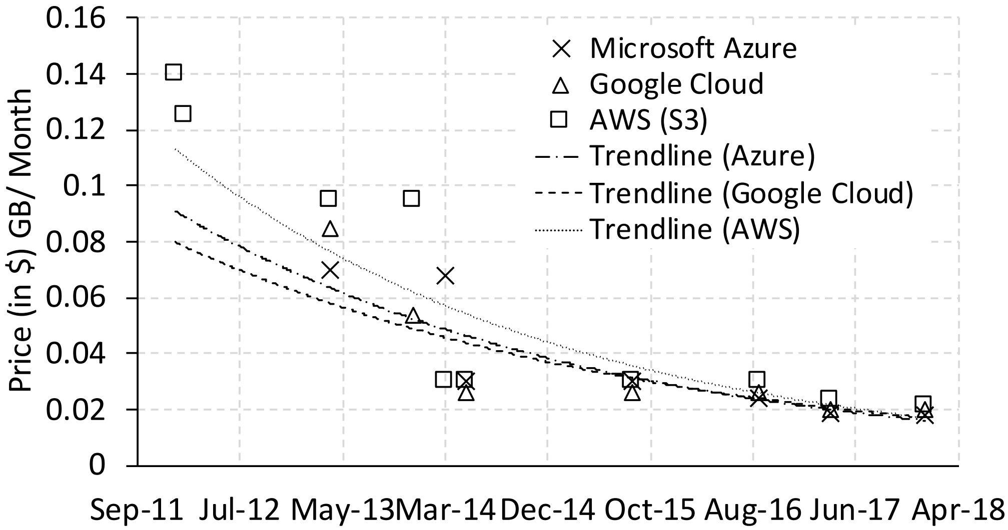

Figure 6.17: Cloud storage price-trend for three providers

Our proposed *CAR* architecture, for better efficiency, needs faster storage services; hence, we consider LRS-Hot (Local Redundant Storage) for Azure, S3 for AWS, and Google's Regional storage option. We discard other lower-priced options available from each of the vendors like LRS-Cold (Azure) with the price of \$0.0152, Glacier (Amazon) with \$0.004 and Coldline (Google) with \$0.007 as these are comparatively slower and will not grant frequent access that is needed for *PRX* in the CAR architecture. It may be instrumental for CAR to have a multi-regional redundancy to offer intra-ISP optimization. But, Amazon's lack of similar service till date and without enough concrete evidence of benefits achieved from such an approach, to maintain consistency, we have not included multi-regional storage price analysis in this dissertation.

If the current trend continues, each of the cloud providers will offer remarkably low-cost unlimited storage. As the price goes down, providers will be motivated to promote more value-added services with a very little tweaking in their infrastructure. It will encourage them to implement a routing feature on the cloud system, where the user will be able to rent a specific sized virtual router by paying a fixed (or on-demand) fee. It can be treated similarly to already existing cloud-based

virtual machines. From the pricing data in Figure 6.17, we have deduced the following decaying cost equation for cloud storage:

$$c(t) = 0.12e^{-0.03t} \tag{6.5}$$

It would be interesting to see how much these vendors are willing to lower the price to compete with the new providers like Rackspace or Backblaze, who are now offering similar services with almost one-fourth of the market price [66].

6.6.3 *Cloud Service Price, s*

Using memory and compute resources of a cloud provider also involves the "servicing" price for the various labor needed to set up the cloud service. A comparison between cloud service price and actual physical router performance cost is not straight-forward. Even so, Newnan [155] identified the fixed cost for setting up the entire routing infrastructure, the variable cost of power and employee salary, the marginal cost of each additional performance improvement as the priority. If we opt for cloud service instead of managing the router by ourselves, it becomes easier to calculate the cost. Cloud providers charge an hourly basis for on-demand service and offer discounts for a year-long commitment. For convenience, we have normalized all prices to hourly-basis to compare them.

Among the yearly committed discount options, AWS Reserved Instance (RI) requires one year or three years of commitment and saves around 24%-75% depending on the duration, Azure has a similar policy with savings of 15%-45%, and Google offers a flat 37%-55% discount per year on its sustained use policy. While IBM still does not have any yearly plan, it negotiates for a month-to-month agreement with about 10% reduced price. Discounts may vary based on the upfront payment method as well; for example, AWS allows no upfront, partial, or full upfront payment options.

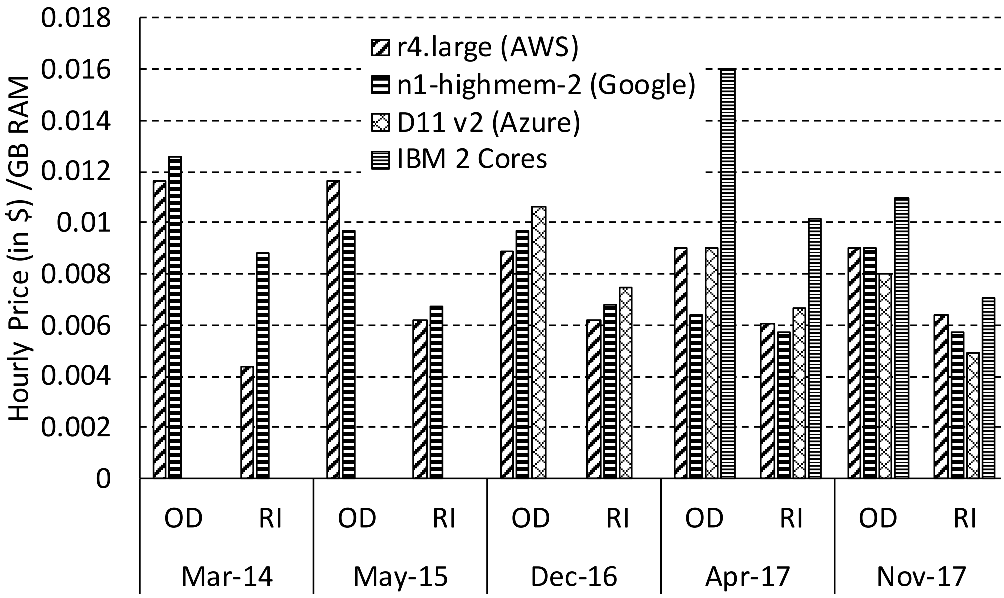

Figure 6.18: Hourly price comparison for On-Demand (OD) vs. Reserved Instance (RI) for one or three years (normalized to hours) cloud service

We plot the comparison between on-demand pricing and year-long commitment for four vendors in Figure 6.18. As AWS matures, we notice that it is not radically dropping the service price now, like it was doing before ("on 44 different occasions over the last six to seven years", stated in 2014 [118]), and rather introduces multiple new instance types with better performance for the same or even cheaper price, over the period we conducted our research. We always prefer the latest instance type, as suggested by AWS, with a similar capability and try to be as consistent as possible without remaining glued to a specific type. We choose *m2.xlarge* for the year 2014, *r3.large* for the year 2015, and continue with *r4.large* afterward as all of these have two vCPU with 15 GB of RAM. For compatibility, we select *n1-highmem-2* from Google Cloud, and *D11 v2* from Microsoft Azure as both of them have the same number of vCPU cores with RAM size of 13 GB and 14 GB, respectively.

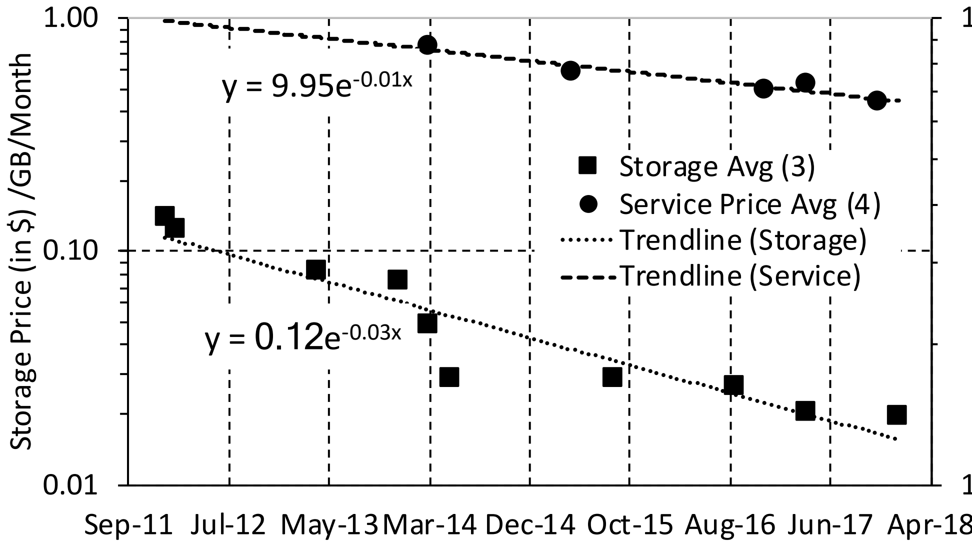

Figure 6.19: Cloud storage vs. service price trends (log-log scale)

Figure 6.19 combines all the data points from Figure 6.17 and Figure 6.18 to illustrate the trends of average cloud storage price and cloud service price simultaneously. It is clear from the graph that the service price trend-line is less steep than the storage one. This supports our observation of a market that is approaching a steady-state, where the cloud providers will not consider price reduction as their main selling point but will delve into developing new features to attract customers instead. Taking the average value to represent the entire business scenario, we get the following decay equation for cloud service price:

$$s(t) = 9.95e^{-0.01t} \tag{6.6}$$

6.6.4 *Transit Price, z*

Packets delegated from the router will be transmitted over the physical fiber link towards the cloud incurring transit cost. Depending on the link type (i.e., dedicated, or shared), the cost will vary.

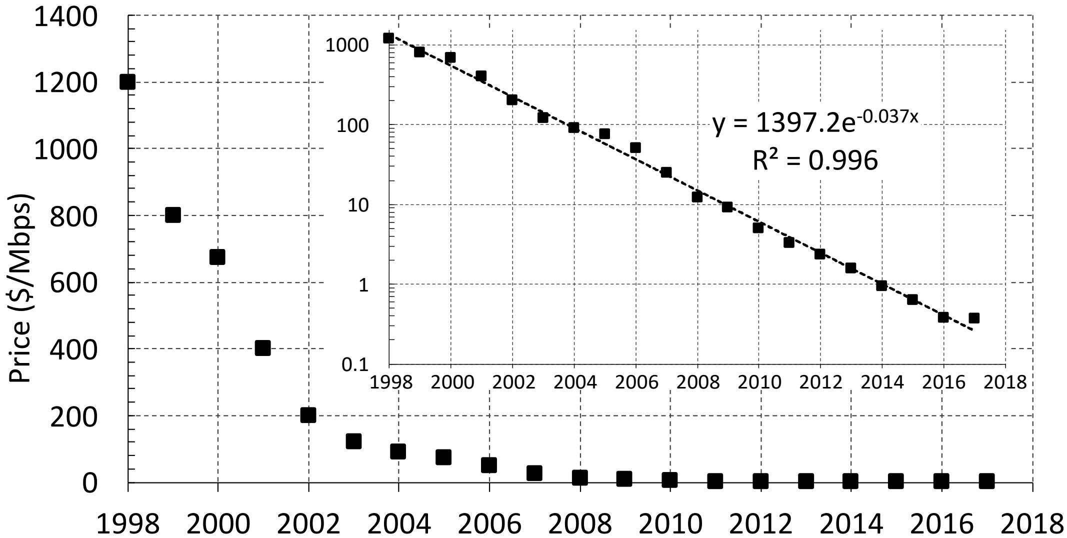

Figure 6.20: Transit price trend

ISPs treat data transmission cost as the epitome of monumental costs involved in Internet business, as it, alone, asks for almost half of the long-haul network expenses [93]. However, the improvement of WDM (Wavelength Division Multiplexing) empowers network operators to constantly reduce the price per unit bandwidth by facilitating the expansion of transmission capacity without any extra fiber line setup.

To explore the trend in transit business, Fishburn and Odlyzko [93] considered two types of data demands. First one is delay insensitive (A), and another one is sensitive to delay (B). They also proposed two different services for these data types: *a)* separate network for A and B with different pricing schemes; and *b)* a single network with a unified price for both.

Analyzing this work and based on our research with prices for minimum commitment [160, 31], we plot transit costs for the year 1998 to 2018 in Figure 6.20. We observe the decay trend in transit costs with larger R^2-ed value, supporting the well-fitting nature of our linear regression model. This decay trend will eventually promote more off-loading of packets to the cloud to do routing.

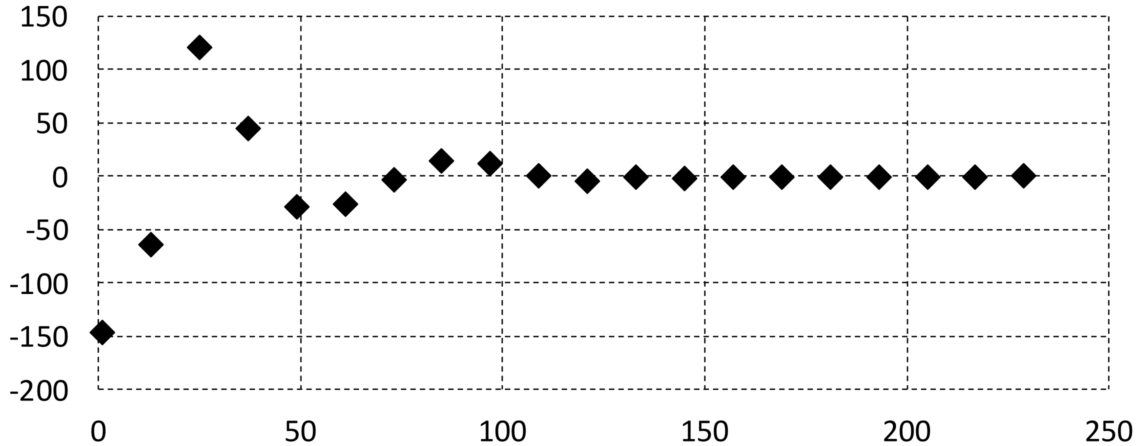

Figure 6.21: Residual plot

To ensure the validity of the model, we have also considered the existence of bias in the prediction by evaluating the residual plot (see Figure 6.21) and found that the values were scattered randomly around zero with the residual center at zero.

For CAR, utilizing a dedicated transit system to delegate the packets to the cloud would be the best. Having said that, such a dedicated transit service policy does not exist except leased fiber lines. But, since this would limit the deployment of CAR to only those *RX*s that can have leased fiber line connections, we do not consider such leased transit service in our model. The only available option for high-bandwidth transit is the ones offered for generic use, which we consider in our model. Fitting the data to an exponential decay gives us the following transit cost equation:

$$z(t) = 1,397.19e^{-0.037t} \tag{6.7}$$

6.7 Economic Viability

Our modeling effort thus far allows exploring the economic viability and scalability of cloud-assisted SDN architectures. We will now look at how legacy and cloud-assisted routing (CAR) will compare in terms of costs, find break-even points and explore regimes where one may be more beneficial in the emerging trends of the Internet routing ecosystem.

Figure 6.22 plots the cost comparison between cloud storage and DRAM. According to the graph, the cloud price is cheaper, and if the current trend continues, it will take at least 15 years for DRAM to catch up with cloud price. Referring to our discussion in Section 6.6.1, DRAM price does not exactly align with the decaying trend, which means, according to this graph, cloud storage will undoubtedly be more profitable compared to DRAM in future. Hence, usage of cloud storage for routing purposes will likely be economical and more common.

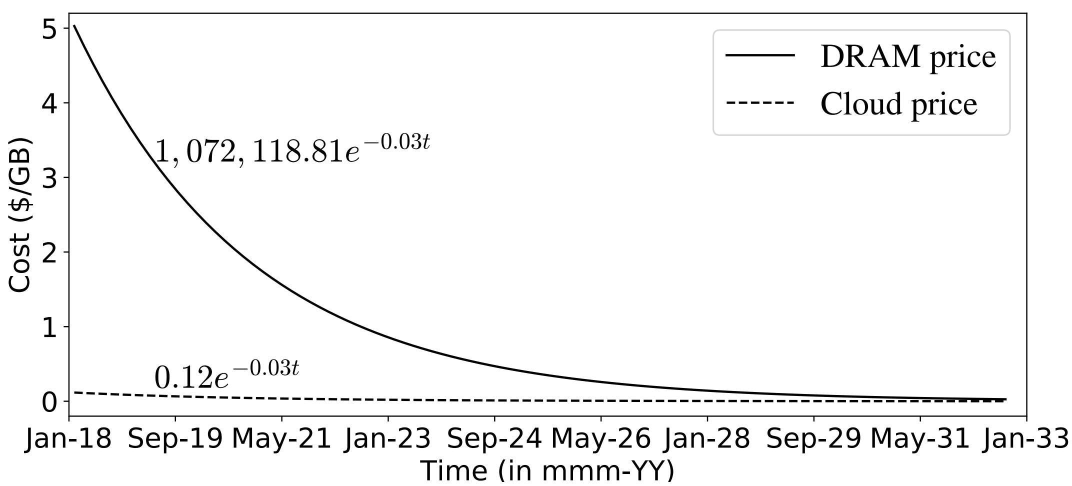

Figure 6.22: Projected Cloud storage price vs. DRAM price

6.7.1 Break-even Points

Based on the cost models (Eq. 6.1) and (Eq. 6.2), we find the break-even point between CAR ($\hat{C}$) and No CAR (C) cases in terms of the costs. Although Figure 6.20 plots the transit cost in $\$/Mbps$/month, for our calculation, we convert all the values to Gigabyte (GB) to match the units with cloud prices. As a parameter into this comparative analysis, we consider the percentage of FIB that needs to be stored in *RX*, i.e., the FIB cache size.

6.7.1.1 Considering no labor cost

Figure 6.23 reports the break-even between C and $\hat{C}$. To calculate ρ value using Eq. 6.3, we need to know the total traffic amount T, ψ, and ω. From CAIDA Anonymized Internet Traces and Chicago (dirA) trace statistics [27] we identified the values for ψ and ω as 0.064 and 0.78 consecutively. Furthermore, we have assumed the transit fee is charged for 20% of the total incoming traffic (2.31 Gb/s) towards *RX*. We shall discuss this assumption in detail and shall explore the feasibility of relaxing it later in Section 6.8.

For now, we see from the graph that $\hat{C}$ is lower than C when FIB cache size is zero, i.e., if we store the entire FIB table in the cloud and delegate all data traffic to *PRX* for processing, CAR is economically beneficial than No CAR. We observe that $\hat{C}$ continues to go down as FIB cache size increases. This is because $\hat{C}$ includes transit price for the delegated traffic to cloud, and as *RX* stores more entry in FIB, fewer packets need delegation to *PRX*, which, effectively, minimizes the total cost. After a certain point, $\hat{C}$ changes the direction and starts to climb up as FIB cache size increases. We name this turning point as *Plutus* after the Greek God of wealth as this marks the maximum profit for CAR. According to the graph, keeping around 10% of the entire FIB in *RX* will ensure the *Plutus* point.

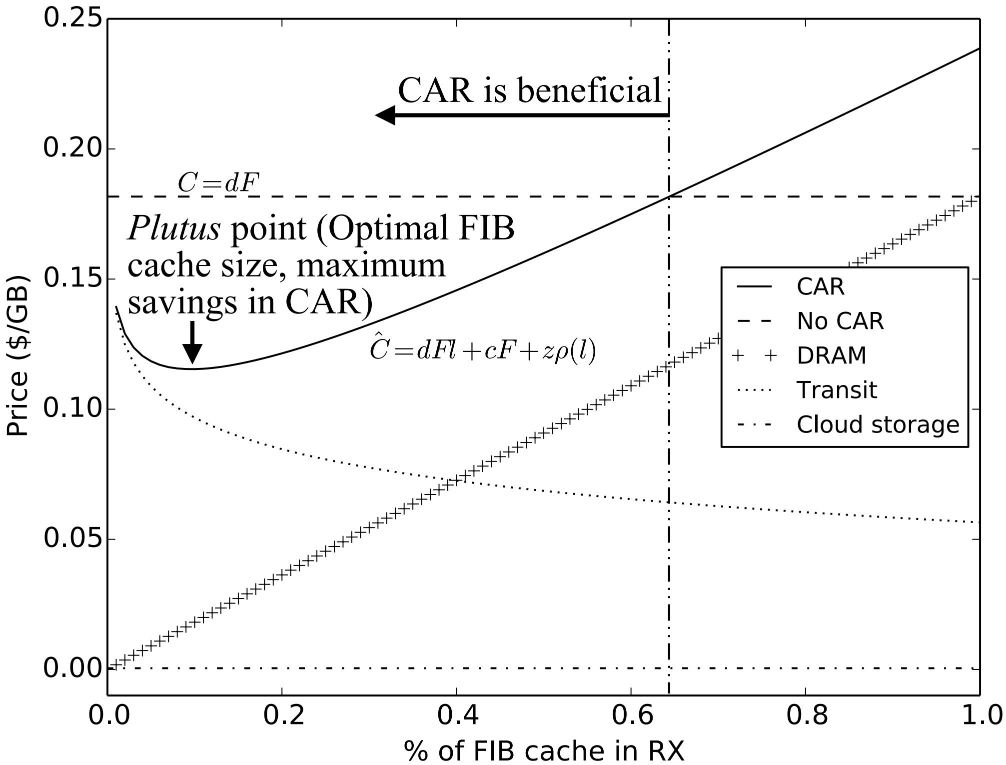

Figure 6.23: CAR vs. No CAR (projected at Jan 2033)

The fact that we need more DRAM to store more FIB entries explains the increase in $\hat{C}$ after the *Plutus* point. Even though *RX* will be able to handle most of the traffic by itself and reduce the transit cost, DRAM price trumps the other variables in $\hat{C}$ and cause the rise up. In the meantime, C maintains the constant value as it is independent of the FIB cache size. The cross-over between these two models happens when FIB cache size reached to 64%, which means $\hat{C}$ is more economical as long as we keep the FIB cache size less than or equal to around 64% of the total FIB size.

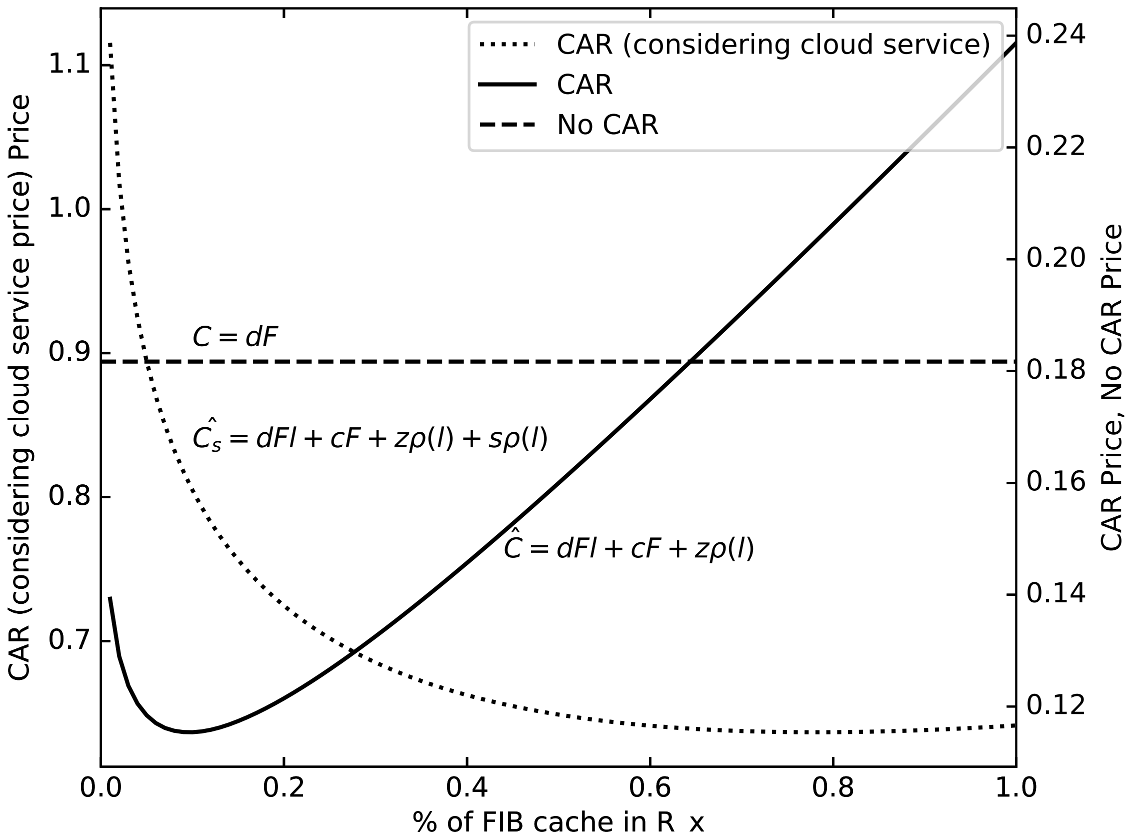

Figure 6.24: CAR with storage and service price vs. No CAR

6.7.1.2 Considering labor cost

According to Newnan's engineering economic cost discussion [155] and Gartner [125], for a five-year life-cycle, the maintenance/support cost may supersede the initial setup cost. So, the engineering/labor cost for physical router maintenance may not be negligible. Though most of these data are business proprietary information, we can still say the labor costs will likely have a sizable impact on the router service pricing. We compare our proposed system, CAR, with an unrealistic approach in Figure 6.24 to prove the cost-effectiveness. We assume zero maintenance cost for traditional routing, while CAR has both cloud service and storage cost. We think this comparison will help us to understand how beneficial the CAR model will be even if we consider it in an uneven

condition. In particular, Figure 6.24 considers three cases.

i) Traditional routing cost (DRAM cost only),

ii) CAR cost (only cloud storage expense), and

iii) CAR cost, considering both cloud storage and service cost. For expressing CAR cost with service cost, we revise Eq. 6.2 as follows:

$$\hat{C}_s = dFl + cF + z\rho(l) + s\rho(l) \tag{6.8}$$

According to the plots portrayed in Figure 6.24, increasing the FIB cache size will be more profitable for $\hat{C}_s$ (case-*iii*), while it is not, indeed, the case for $\hat{C}$ (case-*ii*). These cases may seem to contradict each other, but they do not. With FIB cache size reduction, we essentially delegate more packets to the clouds and thus injecting the additional cloud service cost.

Although this comparison does not show a break-even between $\hat{C}_s$ and C, we would like to emphasize that our cost model for C does not consider the recurring costs involved in traditional routing (e.g., human-labor charges, daily maintenance costs, utility bills, and security expenses), which cannot be amortized over time. A fair comparison would require consideration of the labor and management costs of traditional routing, and CAR's centralization benefits from possible management of multiple *RX*s being managed by the same cloud provider. Future work could explore these additional parameters involving inter-domain routing and business aspects spanning numerous autonomous systems.

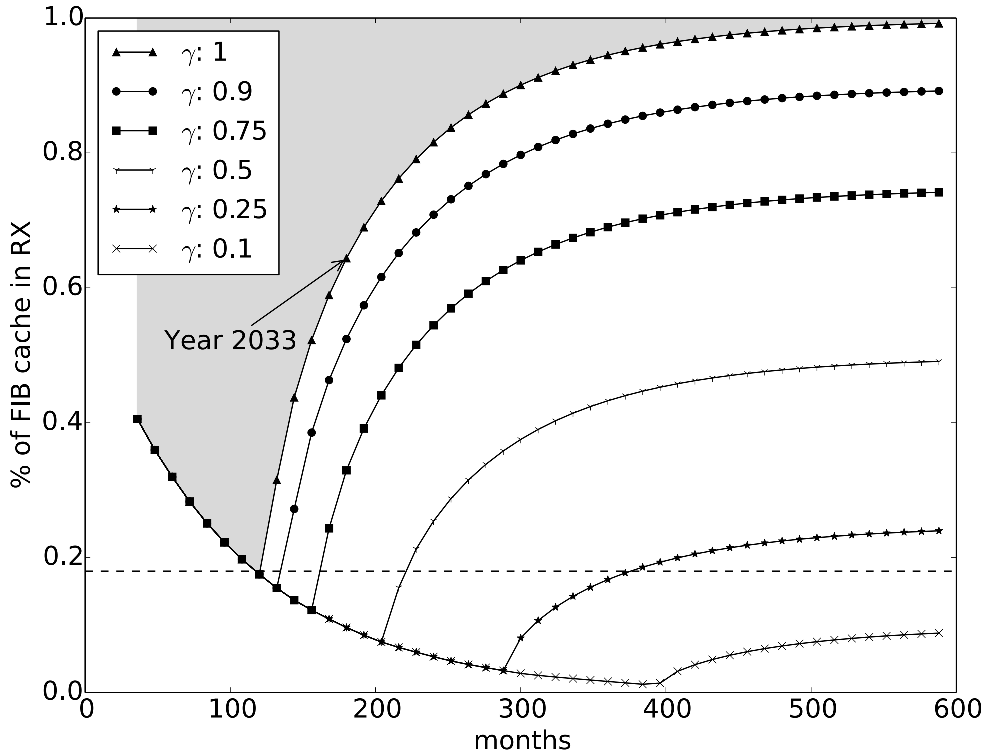

Figure 6.25: CAR savings compared to legacy routing with no CAR in future

6.7.2 CAR Savings

So far, we have explained how much savings CAR can offer at any specific time in the future. We have extended our work in Figure 6.25 to determine how much buffer with respect to in regards to FIB cache size reduction a CAR engineer can enjoy over a period of time. Here, the scale factor represents the percentage of CAR savings over traditional routing, i.e., scale factor value one means there is no extra saving, and both C and $\hat{C}$ are exactly the same (break-even), scale factor 75% means, $\hat{C}$ is 25% cheaper than C. Assuming γ be the scale factor, we state the relation as:

$$\hat{C} = \gamma C \tag{6.9}$$

Each point in individual curves represents the monthly break-even point (with a certain scale-factor) in the future. For example, without any extra savings ($\gamma = 1$), after 180 months, graph plots 0.64 as the break-even point for l (marked in Figure 6.25 as the *Year 2033*), which is exactly what we have seen in Figure 6.23. This means, referring to our earlier detailed discussion, storing less than 64% of FIB in *RX* will be profitable for $\hat{C}$ in the year 2033. The lower we set the γ at (to gain higher $\hat{C}$ savings), the more stringent FIB limitation is set for that specific month. In our analysis, we varied γ from 0.1 to 1 so that the impact of scale factor on maximum and minimum cost savings for CAR can be clearly demonstrated.

Another observation from here is that all the curves are concave, which implies that they converge to a certain maximum point. FIB cache size can not be reduced beyond that threshold for a specific scale factor to achieve further savings.

Finally, we want to emphasize on the shaded region of the figure (beyond $\gamma = 1$ curve), anything on this area is not profitable for CAR at all and engineers are advised to plan accordingly for finding a suitable FIB cache size to serve their own purpose. The dotted line represents 0.18 in the Y-axis of the graph, and this (18%) is the minimum FIB cache size at the break-even point for $\gamma = 1$. Anything below this FIB cache size in *RX* will make CAR cheaper than traditional routing and will be cost-effective.

Eq. 6.9 brings us to the following remarks:

Remark **1**: In order to achieve a long-term scaling factor of γ for fixed total traffic T arriving at RX, the FIB cache size l should be set to γ as long as FIB table size F monotonically increases:

$$\lim_{t \to \infty} l(t) = \gamma \tag{6.10}$$

Proof. Using C and $\hat{C}$ expressions from Eq. 6.1 and Eq. 6.2, we can re-write Eq. 6.9 as following:

$$d(t)Fl + c(t)F + z(t)(1 - \psi\ln(l) - \omega)T = \gamma d(t)F$$

Equating it for γ, we get:

$$\gamma = \frac{d(t)Fl + c(t)F + z(t)(1 - \psi\ln(l) - \omega)T}{d(t)F} \tag{6.11a}$$

Now substituting the values from Eq. 6.4, Eq. 6.5, Eq. 6.7,

$$= l + \frac{0.12e^{-0.03t}F + 1397.19e^{-0.04t}(1 - \psi\ln(l) - \omega)T}{1072118.81e^{-0.03t}F} \tag{6.11b}$$

$$= l + \frac{0.12}{1072118.81} + \frac{1397.19}{1072118.81}\frac{T}{F}(1 - \omega - \psi\ln(l))e^{-0.01t} \tag{6.11c}$$

Second term in the above equation will be very negligible, and ignoring this value we get:

$$= l + [-0.0013\frac{T}{F}\psi\ln(l) + 0.0013\frac{T}{F}(1 - \omega)]e^{-0.01t} \tag{6.11d}$$

Since, $\lim_{t\to\infty} e^{-0.01t} = 0$, we finally get:

$$\gamma = l$$

■

This means, in future, if total incoming traffic towards *RX* remains constant, FIB cache size in *RX* and the CAR savings will have a linear relationship between them. For instance, if we want CAR cost to be 50% cheaper than that of traditional routing, storing 50% of full FIB as cache would be enough to achieve the savings.

Table 6.2: Thresholds for $\frac{T}{F}$ ratio

t (in months)	1	12	24	36	48
$e^{0.01t}$	1.01	1.1275	1.2713	1.4333	1.616

Remark **2**: As long as the ratio of the total incoming traffic T and full FIB size F does not increase more than 12.75% per year, Eq. 6.10 will be always true.

Proof. We can re-write Eq. 6.11d as:

$$\gamma = l + [-0.0013\psi \ln(l) + 0.0013(1 - \omega)]e^{-0.01t}\frac{T}{F} \tag{6.12}$$

To maintain Eq. 6.10, the following must be true.

$$\frac{T}{F} < e^{0.01t}$$

■

Table 6.2 includes some threshold values that $\frac{T}{F}$ can obtain after a certain period of time in order to maximize CAR profit.

6.8 Peering Influence

Throughout our analysis, we assumed that 20% of the total traffic would incur transit cost, and the rest will be transmitted using either public or private peering. This hypothesis is based on the fact that Cloudflare, a prominent CDN that operates 122 data centers across 58 countries around the

globe, observed a significant shift towards peering from the year 2014 to 2016 and expected this trend to grow even more. According to them, 40% of their traffic goes through peered network relationships in North America, which is the lowest in peering as they observed, while Europe and Asia have 60%, and Africa has 90% peered traffic [170]. In the earlier chapters, we have also discussed how peering relationships have evolved, and to what extent, this incident shall reshape the overall Internet architecture. At this point, one question that arises is how CAR and No CAR costs will change as future trends of peering increases.

Figure 6.26 plots multiple $\hat{C}$ graphs by varying the amount of transit traffic in the year 2033.

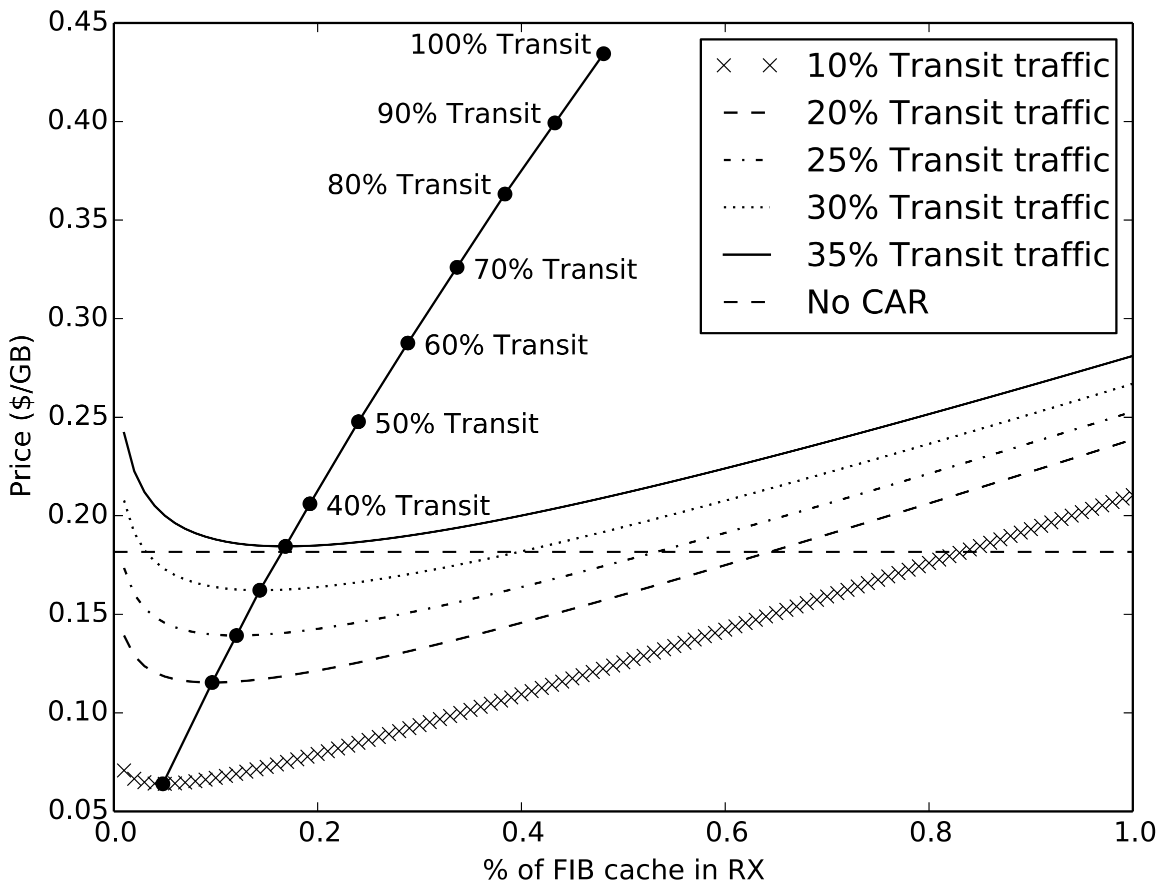

Figure 6.26: CAR pricing at varying FIB cache size

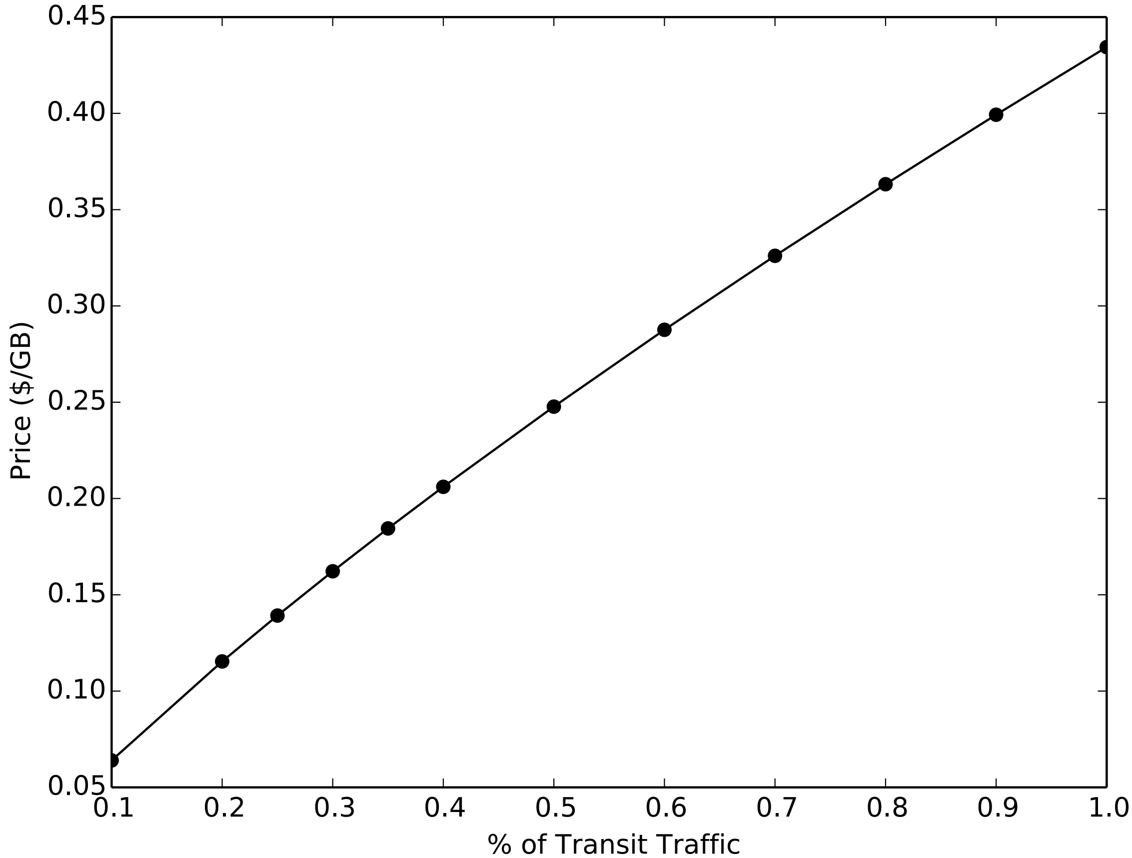

Figure 6.27: CAR pricing at Plutus point based on transit traffic amount

It identifies the *Plutus* points for individual graphs as well. As the transit percentage increases, *Plutus* point shifts towards the right, meaning less traffic delegation to *PRX* will be economically beneficial which is self-explanatory as more transit cost will be charged for more traffic to *PRX*.

To get a better understanding on the relationship between transit traffic amount and its associated $\hat{C}$ values at *Plutus* point, we vary the transit percentage from zero to 100 and charge that corresponding amount of traffic for transit to calculate $\hat{C}$. Depending on the observation from Figure 6.27, we can, conservatively, claim that if ISPs continue to peer more and carry traffic among themselves without charging extra, keeping a smaller FIB cache will be sufficient for CAR providers to offer cheaper routing services. As the amount of transit traffic rises, FIB cache size increases almost lin-

early. Still, CAR is capable of saving at least 50% of FIB at *Plutus* point when there is no peering, and the entire data traffic needs to pay the transit cost.

6.8.1 On Plutus Point

To evaluate the peering influence on *Plutus* point at any given time, we introduce a new variable, λ, that represents the percentage of the delegated traffic that will require transit cost.

Proposition **1**: $\forall \lambda \in \mathbb{R} \mid 0 < \lambda < 1$, the optimal FIB cache size l^* at the *Plutus* point is

$$l^* = \frac{\lambda \psi z T}{dF} \tag{6.13}$$

Proof. Taking the first order derivative of Eq. 6.2 with respect to l and equating it to zero, we get

$$\frac{d}{dl}[dFl + cF + z\rho(l)] = 0 \tag{6.14a}$$

Substituting $\rho(l)$ with $\lambda\rho(l)$ and using Eq. 6.3,

$$dF + \lambda z \frac{d}{dl}[T - \psi T \ln(l) - \omega T] = 0 \tag{6.14b}$$

Hence, to formulate the equation for l^*, we re-write Eq. 6.14b in the following closed-form expression:

$$l^* = \frac{\lambda \psi z T}{dF}$$

■

Our observations from Eq. 6.13 are multifold. First, the ratio of transit cost z and DRAM cost d converges to zero if transit cost keeps declining. If that is the case, then l^* will get closer to zero as

well. Second, based on how the peering relationship between the ISPs evolves in the future, it is possible that transit percentage λ may also increase or decrease. Third, as total traffic T continues to grow, with the same λ value, a larger l^* will be obtained, and force the architecture to ensure a modest FIB cache size to mitigate the traffic increase impact. However, as T keeps increasing, bigger full FIB will be required to store the entire BGP table, and the ratio of total traffic T and FIB size F will play an essential role in determining how CAR cost grows over time. Finally, with λ increasing, $\hat{C}$ at *Plutus* point is also increased. After a certain value of λ, $\hat{C}$ exceeds C (see in Figure 6.26 for *35% Transit* curve), and CAR becomes no longer economically beneficial beyond that point.

6.8.2 On Optimal CAR Design

Eq. 6.2 calculates the CAR cost for any FIB cache size. By using l^* instead of l, we can get the CAR cost when optimal FIB cache size is judicially picked by the CAR designer. This gives us a chance to observe the peering influence at optimal CAR design.

Proposition **2**: If ξ_1, ξ_2, ξ_3 are three positive constants and f is a logarithmic function of transit percentage λ, CAR cost at optimal FIB cache size $\hat{C}(l^*)$ will be:

$$\hat{C}(l^*) = \lambda\xi_1 - \lambda\xi_2 f(\lambda) + \xi_3 \tag{6.15}$$

Proof. To obtain the optimal CAR design case, we substitute l^* to Eq. 6.2:

$$\hat{C}(l^*) = dFl^* + cF + z\rho(l^*) \tag{6.16a}$$

Substituting $\rho(l)$ with $\lambda\rho(l)$ and using Eq. 6.3,

$$\hat{C}(l^*) = dFl^* + cF + \lambda z[1 - \psi \ln(l^*) - \omega]T \tag{6.16b}$$

Using l^* from Eq. 6.13, we get:

$$\hat{C}(l^*) = \lambda\psi zT + cF + \lambda z[1 - \psi \ln(\frac{\lambda\psi zT}{dF}) - \omega]T \tag{6.16c}$$

$$= \lambda zT[1 + \psi - \omega] - \lambda zT\psi \ln(\frac{\lambda\psi zT}{dF}) + cF \tag{6.16d}$$

$$= \lambda\xi_1 - \lambda\xi_2 f(\lambda) + \xi_3 \tag{6.16e}$$

Here, ξ_1, ξ_2, ξ_3 are constants while f is a function of λ and can be interpreted as following:

$$\xi_1 = zT[1 + \psi - \omega] \tag{6.16f}$$

$$\xi_2 = zT\psi \tag{6.16g}$$

$$\xi_3 = cF \tag{6.16h}$$

$$f(\lambda) = \ln(\frac{\lambda\psi zT}{dF}) \tag{6.16i}$$

■

Now, according to Eq. 6.15, CAR providers can consider $\hat{C}(l^*)$ as a function of λ (transit percentage) alone, and use this condition to maximize their profit. Next, by setting λ values to the extreme, we get the boundary conditions for $\hat{C}(l^*)$.

Proposition **3**: In a hypothetical environment in the future where ISPs will be peering heavily with each-other (i.e., $\lambda \to 0$), CAR cost at the optimal FIB cache size $\hat{C}(l^*)$ will be the direct product

of cloud storage cost c and the total FIB size F.

$$\hat{C}(l^*) = cF \tag{6.17}$$

Proof. Using $\lambda = 0$ in Eq. 6.16d,

$$\hat{C}(l^*) = \xi_3 \tag{6.18a}$$

$$= cF \quad \text{[From Eq. 6.16h]} \tag{6.18b}$$

■

This equation is independent of DRAM cost d and exactly resembles Eq. 6.1 except for the fact that d has been replaced with c (cloud-storage cost). This forecasts for a future where ISPs, in a completely peered environment among themselves, will be able to store the full FIB table in the cloud and delegate the entire traffic to cloud without any extra charge, since there will be no transit cost involved at all. If this actually happens, then cloud providers can extend their footprint into routing business aggressively by emerging themselves as new candidates for ISP market and eliminate the existing ones totally, or embrace CAR architecture and form partnerships to progress in a more conventional way.

Proposition **4**: For light peering (i.e., $\lambda \to 1$), if transit price z does not continue to drop, delegation to the cloud will cost additional charge, which in turn, will mandate CAR providers to deal with the incoming traffic locally instead of collaborating with the cloud. In such a case, the CAR cost at optimal FIB cache size will be:

$$\hat{C}(l^*) = zT[1 - \psi \ln(\frac{\psi zT}{dF}) - \omega] - zT\psi + cF \tag{6.19}$$

Proof. Using $\lambda = 1$ in Eq. 6.16d,

$$\begin{aligned}\hat{C}(l^*) &= zT - zT\psi - zT\omega - zT\psi \ln(\frac{\psi zT}{dF}) + cF \\ &= zT[1 - \psi \ln(\frac{\psi zT}{dF}) - \omega] - zT\psi + cF\end{aligned}$$

■

As T (total traffic through a router) increases, which is expected to be, "ln" function gives larger value and can even produce ∞, mathematically. However, based on processing speed, queue size, and consumption of power, we can safely ignore this possibility as every router will have its own threshold limit, and an electronic device can not perform indefinitely.

The multiplier of zT, in the first term of Eq. 6.19, is ρ (see Eq. 6.3), and the maximum value of it can be one, as a router can not delegate more traffic than its incoming traffic amount. To make this possible, ψ value can never be equal to zero for two reasons. First of all, potential heavy hitters (the popular IP prefixes) will always exist in the router to exhibit temporal dynamics and thus preventing ψ from being zero. And secondly, even for a capricious router with its arbitrary list of prefixes, if ψ becomes zero, "ln" value will be *undefined*, and the entire equation will become indeterminate.

For any $0 < \psi < 1$ value, $zT > zT\psi$ will be always true. However, zT and $zT\psi$ will be very close to each other unless ψ becomes exceptionally small, which will be a rare phenomenon for the Internet. Now, if z does not continue to drop as we have seen in Section 6.6.4, CAR providers have to face a harsh environment where they are bound to pay for every data traffic they delegate since there are zero peerings, a nightmare for them.

Finally, the minimum value of ρ can be zero if the router does not delegate any traffic at all

considering the transit situation. This is the worst-case scenario for CAR architecture, where *RX* avoids delegating towards *PRX*.

CHAPTER 7: CONCLUDING REMARKS

We sought to find the feasibility of a vertically-integrated ISP market where content providers will be dominating and inaugurated the idea of *sugarcane* ISP to explore the relationship between those new ISPs. To make the peering automated, we developed a tool, a stepping stone towards complete automation, that can suggest a list of similar-sized ISPs based on different ISP attributes. We also investigated the existence of a correlation between ISP's characteristics at the AS-level (e.g., the number of providers and the number of customers) and its stock market prices. We introduced the Network Dependency Index (NDI), a metric to quantify this correlation of the topological structure of an ISP to its economic value. Finally, we have introduced a new hybrid approach of SDN that leverages the computational power of the cloud while keeping the intelligence of router to some extent for reducing the FIB size and eventually offer monetary benefits to ISPs. Since reliability in communication links between the controller and the hardware switches in existing SDN remains unanswered, a trusted third party like the cloud can grab this opportunity to offer new kinds of services. If it can take over a significant portion of control plane functionalities and provide secure communication via public Internet service, concerns like switch-controller link failure, queuing delay in the controller can be easily mitigated.

7.1 Contribution Summary

We have introduced *inter-ISP economic distance* measure to quantify the thriving progress of content ISPs over the carrier ISPs. We, then, use the geographic and economic inter-ISP distance measures to make projections on how a market of content-dominated sugarcane ISPs may perform in terms of peering. Using three datasets (PeeringDB for peering locations; and Intrinio and Microtrends for market capital), we compare 37 ISPs from the US market. Based on our analysis,

content providers are dominating in market value, and their centroids are the closest to each other, meaning their coverage areas overlap the most. Access ISPs have the least overlap as some regional ISPs are serving in specific regions by purchasing the backbone service from transit ISPs. As content providers are vertically integrating with (or merely purchasing) access ISPs to form *sugarcane* architecture, it seems likely that the sugarcane ISPs of the future will have centroids closer to each other than the existing access ISPs. It may reduce the incentive for peering and hence, reduction in the overall end-to-end performance of the Internet services. A similar situation presented itself when we consider the inter-ISP economic distance. The content providers market is most skewed, implying less incentive for peering and collaboration among them.

Again, for understanding the relationship between the ISP networks and the stock market, we harvest data of 32 major United States ISPs. By utilizing four different datasets (i.e., CAIDA AS Relationships, AS Rank, Quandl, and Yahoo! Finance), we point out a high correlation between ISPs' stock values and their network characteristics such as customer cone size and AS degree. Even though there is a correlation between stock market values and ISP network characteristics, we have found that the correlation value strongly depends on the type of the ISP (i.e., content, transit, or access).

In this dissertation, we have introduced "meta-peering" as a combined effort towards automating the entire peering process among ISPs. As part of the complete automation process, we have focused on the peer selection technique. Describing the peering history, we segment the overall peering process into four phases and formulate the peer selection sub-process as an optimization problem. Using PeeringDB and CAIDA datasets, we estimate the traffic matrix of an ISP, identify its PoPs, and then describe an Acceptable Peering Contracts generator framework to suggest the best candidates for a requester ISP along with its best peering locations. We introduce the concept of felicity score to represent the interest of peering between an ISP pair.

We have found that ISPs mostly (more than half of them) prefer to offload as much traffic as they can, and we have successfully identified 107 ISP pairs that are already peering according to CAIDA. We could not identify 37 of the existing peering pairs, but that is mostly because some ISPs are big and cover the entire area of the other ISP, and so they do not have much motivation for peering.

To compare the economic aspect of our proposed hybrid routing, we have presented two cost models: one for traditional routing and the other for CAR. We have included the related variables that impact the cost of each service. We also show the trends for these associated variables separately and try to predict how they will behave in the future to equip the Internet providers with a better understanding of the nature of these variables. We have shown that cloud storage is cheaper than DRAM price, and transit cost is following an almost consistent decay. We then compare the economic viability of CAR with respect to traditional routing by finding a break-even between the two cost models. Although our analysis primarily focuses on the storage cost, we also consider the cloud service cost to replicate labor cost (as in traditional routing) and observe its effects on the break-even points. Later we demonstrate how much savings CAR can offer regarding FIB cache size, in future, by using scale factor against traditional routing, and we have found that it is not possible to achieve unlimited savings for any given scale factor. For example, we have shown for the Year 2033, keeping a FIB cache with less than 64% of full FIB will be profitable for CAR.

Further, as a proof-of-concept of CAR architecture, we develop a prototype on the GENI testbed and compared its performance with traditional routing. Using this testbed, we successfully analyze the impact of FIB cache size on CAR and discuss the overall latency.

We consider the Internet peering impact on our proposed architecture as it is specifically important for CAR providers to know how much luxury they can afford in delegating traffic to the cloud via paid transit. We show an example scenario for the year 2033, where at least 65% of the traffic needs

to be routed through a peered network so that adopting CAR will be economical in comparison to traditional routing.

7.2 Future Direction

Right now, we are on the verge of perplexity due to the fact that the DRAM price is not consistent enough to rely on, in the present condition, and FIB size is almost exploding. Inspired by the idea of virtualizing network functions for easier management and utilizing SDN architecture to attain significant performance improvement, our primary interest has been to manifest how much FIB reduction CAR can offer. In doing so, we explore the economics of a hybrid approach, rather than the complete separation of the control and data plane, supported by a trusted third party (like the cloud) that can provide high scalability. We believe there is considerable scope for further research in this area, and some of the key research questions include developing a failure-resilient architecture, and analyzing the intra- and inter-cloud optimization using multiple RXs and PRXs. Management of two different forwarding entities to act as a unified body, aggregating SLAs of these individual constituents can also be formulated as an exciting research topic.

Regarding the CAR prototype, significant efforts can be made to render a more realistic make-over. Right now, we only consider the *crystal ball* approach in terms of FIB caching. But, an improved version may require re-writing of the *CARbgp* controller application in a way that it will continuously monitor individual flows' statistics to identify the least used prefix. Once found, the controller will remove that from flow table of the switch, and add a new flow rule to make space for the new prefix. We can also consider BGP's *keepalive timer* to set the lifespan of flows from a particular neighbor. If a neighboring router is dead, packets should not be routed that way; but, the current prototype does not consider this yet.

We believe our novel approach will open new horizons on understanding the relationship between the ISP networks and the stock market, and provide valuable insights into the dynamics of operating an ISP. In the future, a similar study with more extensive datasets, including ISPs from other countries, can be conducted to get a better overall picture. Another possible expansion of our work may lead to analyzing the transit degrees of ISPs and confirm whether being highly connected (i.e., ingress points of the ASes which belong to these ISPs) is beneficial or not.

ISPs controlling huge swathes of the Internet market may yield higher prices for end customers and feel less compelled to be innovative. But, as long as the technology evolves and the reliance on the Internet continues to strengthen, ISPs will consistently keep investing in their infrastructure. More research is needed to understand the trend of vertical ISP integration, particularly in terms of peering quality and inter-ISP overlap quantity using distances among PoPs of potentially peering ISPs. Regulating such a vertical market, as new incentives and dynamics may emerge between new sugarcane ISPs, will also become quite challenging. Lack of any regulatory body in the world of sugarcane ISPs may lead to an unfortunate scenario of vertical foreclosure [172] where an ISP may deliberately degrade its competitors' contents to increase its own demand. Alternatively, CDNs may continue playing the interim role and act as an intermediary to prevent a purely vertical market. This approach could also maintain the existing carrier market highly motivated for peering. For those ISP markets owned/dominated by state or implementing regulations that impose horizontal competition, new studies are needed to understand the increasing dominance of content.

In our experiment, we have estimated TMs of both the requester and the candidate ISPs, which may cause some deviation from the accurate result. If we have at least either of the ISPs' genuine TM, our analysis and validation may have accomplished better. The validation relies on comparing with the current peering connections, but these peering may be done in an ad-hoc manner and perhaps not ideal. Developing a full-fledged prototype, implementing it within an ISP or an IXP, may provide better insight and will improve the prediction. Our procedure of felicity score calculations

needs further investigation and feedback from the ISP community to establish more precise and stable metric sets for peer selection, which we leave for the future.

As of now, the framework considers geographic overlapping and traffic exchange willingness between two ISPs. But, it is easy to add newer modules such as *cost benefits*, *security overheads* to extend the framework further. Besides, we have only considered US-based ISPs; as a result, (higher ranked in CAIDA AS-Rank) Telia Company AB, or GTT Communications has not been included. Expansion of the current ISP set and their operational regions in the study is another vein for future work, which we plan to embark on.

APPENDIX A: LIST OF ISPS

A.1 Sugarcane ISP analysis

Following is the list of ISPs that we have considered for analysing content-dominated vertical ISPs:

Access ISPs AT&T Corp., CenturyLink Communications, LLC, Charter Communications Inc, Comcast Cable Communications, LLC, Cox Communications Inc., Google Fiber, Hotwire Communications, Liberty Global B.V., Mediacom Communications Corporation, PenTele-Data, Sonic.net, TDS Telecom, Time Warner Cable, Wide Open West (WOW!).

Content Providers: Amazon, Facebook, Google, IBM, Microsoft Corporation, Netflix, Spotify, Verizon Communications, Inc., Yahoo!, Yelp!

Trasint ISPs: Cogent Communications, CoreSite LLC, Frontier Communications Solutions, General Communications, GTT Communications, Inc., Hurricane Electric LLC (HE), Internet Initiative Japan Inc. (IIJ), Level3, NTT America, PCCW Global, Qwest (CenturyLink), Sprint, Verizon Communications, Inc., Wide Open West (WOW!), Zayo.

A.2 For peering requirements table

Following is the list of ISPs that we have considered for tabulating peering requirements:

Access ISPs Charter Communications Inc, Comcast Cable Communications, LLC, Cox Communications Inc., Free SAS (France), Liberty Global B.V., Mediacom Communications Corporation, Suddenlink by Altice, Swisscom, TDS Telecom, Time Warner Cable.

Content Providers Blizzard Entertainment, Dropbox Inc., Facebook Inc, Google LLC, Limelight Networks, Microsoft Corporation, Netflix, OpenAccess Network Services (PogoZone), Visa Inc., Yahoo! (Oath)

Transit ISPs AT&T Corp., British Telecom, CenturyLink Communications, LLC, Frontier Communications Solutions, GTT Communications, Inc., Hurricane Electric LLC, KPN International, Orange Group, RETN Capital Ltd., Telefonica International, Telia Carrier, Telstra Corp., Verizon Communications, Inc., XO Communications, Zayo (Abovenet Communications Inc.)

A.3 For meta-peering analysis

Following is the list of ISPs that we have used for analysis, along with their specific AS numbers:

Access ISPs Cable ONE, Inc. (11492), CenturyLink Communications, LLC (209), Charter Communications Inc. (7843), Comcast Cable Communications, LLC (7922), Cox Communications Inc. (22773), TDS Telecom (4181), Windstream Communications LLC (7029)

Content Providers Akamai International B.V. (20940), Amazon (16509), Ebay (62955), Facebook (32934), Google (15169), Microsoft Corporation (8075), Netflix (2906)

Transit ISPs Columbus Networks USA, Inc. (23520), Cogent Communications (174), Hurricane Electric LLC (6939), NTT America, Inc. (2914), PCCW Global, Inc. (3491), Sprint (1239), Verizon Communications, Inc. (701), Zayo (Abovenet Communications Inc.) (6461)

A.4 ISPs' Traffic Ratios

Following is the list of ISPs' traffic ratios as observed from PeeringDB[7]:

Heavy Inbound (HI): Cable One Inc., Charter Communications Inc.

Mostly Inbound (MI): Columbus Networks (Liberty Global), Cox Communications, TDS Telecom, Windstream Communications

Balanced (B): Amazon, CenturyLink Communications, LLC, Comcast Cable Communications, LLC, Hurricane Electric LLC, NTT America, PCCW Global, Sprint, Verizon Communications, Inc., Zayo

Mostly Outbound (MO): Google, Microsoft Corporation

Heavy Outbound (HO): Akamai Technologies, Ebay, Facebook, Netflix

Not Disclosed (ND): Cogent Communications, Inc.

[7] As of November 19, 2019

LIST OF REFERENCES

[1] Customer Cone. CAIDA. Accessed: April, 2019. [Online] Available: https://asrank.caida.org/about#rank.

[2] End of Day US Stock Prices. Quandl. Accessed: June, 2017. [Online] Available: https://www.quandl.com.

[3] Geometric median. Wikipedia. Accessed: May, 2019. [Online] Available: https://en.wikipedia.org/wiki/Geometric_median.

[4] Internet World Stats. Accessed: October, 2019. [Online] Available: http://www.internetworldstats.com/stats.htm.

[5] List of broadband providers in the United States. Wikipedia. Accessed: November, 2018. [Online] Available: https://en.wikipedia.org/wiki/List_of_broadband_providers_in_the_United_States.

[6] Packet Clearing House. Accessed: June, 2019. [Online] Available: https://www.pch.net.

[7] PeeringDB facilitates the exchange of information related to Peering. PeeringDB. Accessed: November, 2019. [Online] Available: https://peeringdb.com.

[8] Population Estimates APIs. US Census Bureau. Accessed: June, 2019. [Online] Available: https://www.census.gov/data/developers/data-sets/popest-popproj/popest.html.

[9] RIPE NCC. Accessed: June, 2019. [Online] Available: https://www.ripe.net.

[10] RYU SDN Framework. Accessed: September, 2019. [Online] Available: https://osrg.github.io/ryu/.

[11] Stock charts. Macrotrends. Accessed: October, 2018. [Online] Available: https://www.macrotrends.net/.

[12] University of Oregon Route Views Project. Accessed: August, 2019. [Online] Available: http://www.routeviews.org/routeviews/.

[13] US Fundamentals & Stock Prices. Intrinio. Accessed: October, 2018. [Online] Available: https://intrinio.com.

[14] YCHARTS. Accessed: June, 2017. [Online] Available: https://ycharts.com/.

[15] AS Ranking. CAIDA, 2007. [Online] Available: http://as-rank.caida.org.

[16] The CAIDA AS Relationships Dataset. CAIDA, 2007. [Online] Available: http://www.caida.org/data/as-relationships/.

[17] The 'Donut Peering' Model: Optimizing IP Transit for Online Video. MZIMA, Sep 2009. Accessed: April, 2019. [Online] Available: http://tnarg.org/pdf/donut_peering.pdf.

[18] Top tips for doing a cloud storage cost analysis, Jan. 2013. Accessed: January, 2018. [Online] Available: http://searchstorage.techtarget.com/tip/Top-tips-for-doing-a-cloud-storage-cost-analysis.

[19] Memory Troubleshooting Guide for Cisco 4000 Series ISRs, June 2015. Accessed: January, 2018. [Online] Available: https://www.cisco.com/c/en/us/td/docs/routers/access/4400/troubleshooting/memorytroubleshooting/isr4000_mem.html#pgfId-1080950.

[20] AT&T Global IP Network Peering Policy. , Mar 2016. Accessed: December, 2018. [Online] Available: https://www.corp.att.com/peering/.

[21] CABLEVISION SYSTEMS CORP (CVC). TheStreet, June 2016. Accessed: June, 2017. [Online] Available: https://www.thestreet.com/quote/CVC.html.

[22] Center for Applied Internet Data Analysis (CAIDA). Technical report, 2016. Accessed: October, 2019. [Online] Available: https://www.caida.org/publications/posters/eps/caida-infosheet-2016.pdf.

[23] Completing the Netflix Cloud Migration, Feb. 2016. Accessed: January, 2018. [Online] Available: https://media.netflix.com/en/company-blog/completing-the-netflix-cloud-migration.

[24] IBM's global revenue from 2010 to 2016, by business segment (in billion U.S. dollars). Statista, 2016. Accessed: November, 2018. [Online] Available: https://www.statista.com/statistics/274823/ibms-global-revenue-by-segment/.

[25] RACKSPACE HOSTING INC (RAX). TheStreet, November 2016. Accessed: June, 2017. [Online] Available: https://www.thestreet.com/quote/RAX.html.

[26] TIME WARNER CABLE INC (TWC). TheStreet, 2016. Accessed: June, 2017. [Online] Available: https://www.thestreet.com/quote/TWC.html.

[27] Trace Statistics for CAIDA Passive OC48 and OC192 Traces, Apr. 2016. Accessed: March, 2018. [Online] Available: http://www.caida.org/data/passive/trace_stats/.

[28] Comparing Cloud Instance Pricing: AWS vs Azure vs Google vs IBM, Nov. 2017. Accessed: January, 2018. [Online] Available:

ttps://www.rightscale.com/blog/cloud-cost-analysis/comparing-cloud-instance-pricing-aws-vs-azure-vs-google-vs-ibm.

[29] Draft BEREC Report on IP-Interconnection practices in the Context of Net Neutrality. Technical report, 2017. Public Consultations. Accessed: November, 2019. [Online] Available: https://tinyurl.com/sb8mh8o.

[30] Internet usage penetration in the United States in 2017, by state. Statista, 2017. Accessed: October, 2019. [Online] Available: https://www.statista.com/statistics/184691/internet-usage-in-the-us-by-state/.

[31] IP Transit pricing summary (Flat-rate 1Gbps-on-GigE pricing: $380/mo (3-year term)), Aug. 2017. Accessed: January, 2018. [Online] Available: http://henet.p2knowledgebase.com/view/short_post/WnL8B.

[32] Is the cheapest cloud pricing flexible or granular?, Dec. 2017. Accessed: January, 2018. [Online] Available: https://www.cio.com/article/3243092/cloud-computing/is-the-cheapest-cloud-pricing-flexible-or-granular.html.

[33] [Market View] TrendForce Reports 3Q17 DRAM Revenue Hit a New High With a 16.2% Sequential Increase; DRAM Prices to Rise by 10% on Average in 4Q17, Nov. 2017. Accessed: January, 2018. [Online] Available: https://www.dramexchange.com/WeeklyResearch/Post/2/4828.html.

[34] Adobe Case Study, 2018. Accessed: January, 2018. [Online] Available: https://aws.amazon.com/solutions/case-studies/adobe/.

[35] Airbnb Case Study, 2018. Accessed: January, 2018. [Online] Available: https://aws.amazon.com/solutions/case-studies/airbnb/.

[36] Best Buy Slashes App Development Time and Resources with Google App Engine, 2018. Accessed: January, 2018. [Online] Available: https://cloud.google.com/customers/best-buy/.

[37] Restoring Internet Freedom. FCC, Jan 2018. Accessed: November, 2019. [Online] Available: https://www.fcc.gov/document/fcc-releases-restoring-internet-freedom-order.

[38] BGP Routing Table Analysis Reports, 2019. Accessed: October, 2019. [Online] Available: https://bgp.potaroo.net.

[39] Cisco Visual Networking Index: Forecast and Trends, 20172022 White Paper. CISCO, Feb. 2019. Accessed: October, 2019. [Online] Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html.

[40] Equinix Internet Exchange Data Sheet. Equinix, 2019. Accessed: November, 2019. [Online] Available: http://www.equinix.com/resources/data-sheets/equinix-internet-exchange/.

[41] G. Accongiagioco, E. Altman, E. Gregori, and L. Lenzini. Peering vs transit: a game theoretical model for autonomous systems connectivity. In *Networking Conference, 2014 IFIP*, pages 1–9. IEEE, 2014.

[42] A. Ahmed, Z. Shafiq, H. Bedi, and A. Khakpour. Peering vs. transit: Performance comparison of peering and transit interconnections. In *Network Protocols (ICNP), 2017 IEEE 25th International Conference on*, pages 1–10. IEEE, 2017.

[43] A. Araldo, D. Rossi, and F. Martignon. Cost-aware caching: Caching more (costly items) for less (ISPs operational expenditures). *IEEE Transactions on Parallel and Distributed Systems*, 27(5):1316–1330, 2016.

[44] D. Arena. WORKSHOP: STARTING AN IXP (Part I), 2011. Accessed: June, 2019. [Online] Available: `https://www.ripe.net/participate/meetings/regional-meetings/see/dubrovnik-2011/presentations/IXP%20Workshop%20Part%20I%20-%20Daniele%20Arena.pdf`.

[45] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, et al. A view of cloud computing. *Communications of the ACM*, 53(4):50–58, 2010.

[46] H. Asai and Y. Ohara. Poptrie: A compressed trie with population count for fast and scalable software IP routing table lookup. In *ACM SIGCOMM Computer Communication Review*, volume 45, pages 57–70. ACM, 2015.

[47] P. Baake and T. Wichmann. On the economics of Internet peering. *Netnomics*, 1(1):89–105, 1999.

[48] B. Baccarne, T. Evens, and D. Schuurman. The television struggle: An assessment of over-the-top television evolutions in a cable dominant market. 2013.

[49] N. Badasyan, S. Chakrabarti, et al. Private peering among Internet backbone providers. *Economics Working Paper Archive, Series on Industrial Organization, Washington University in St. Louis (WUSTL)*, 2003.

[50] S. Bafna, A. Pandey, and K. Verma. Anatomy of the internet peering disputes. *arXiv preprint arXiv:1409.6526*, 2014.

[51] R. Bala, A. Chandrasekaran, and J. McArthur. Magic Quadrant for Public Cloud Storage Services, Worldwide, July 2017. Accessed: January, 2018. [Online] Available: https://www.gartner.com/doc/reprints?id=1-47K8W1K&ct=170725&st=sb.

[52] M. Bando and H. J. Chao. Flashtrie: hash-based prefix-compressed trie for IP route lookup beyond 100Gbps. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE, 2010.

[53] B. Barnes. Disney's Streaming Service Starts to Come Into Focus. The New York Times, Aug 2018. Accessed: November, 2019. [Online] Available: https://nyti.ms/2MccswC.

[54] J. Barr. Amazon EC2 Price Reduction in the Asia Pacific (Mumbai) Region, Dec. 2017. Accessed: January, 2018. [Online] Available: https://aws.amazon.com/blogs/aws/category/price-reduction/.

[55] A. Begen, T. Akgul, and M. Baugher. Watching video over the web: Part 1: Streaming protocols. *IEEE Internet Computing*, (2):54–63, 2010.

[56] J. Benesty, J. Chen, Y. Huang, and I. Cohen. Pearson correlation coefficient. In *Noise reduction in speech processing*, pages 1–4. Springer, 2009.

[57] M. Berman, J. S. Chase, L. Landweber, A. Nakao, M. Ott, D. Raychaudhuri, R. Ricci, and I. Seskar. GENI: A federated testbed for innovative network experiments. *Computer Networks*, 61:5 – 23, 2014. Special issue on Future Internet Testbeds - Part I.

[58] S. M. Besen and M. A. Israel. The evolution of Internet interconnection from hierarchy to Mesh: Implications for government regulation. *Information Economics and Policy*, 25(4):235–245, 2013.

[59] M. E. Biery. The 10 Fastest-Growing Industries In The U.S. Forbes, April 2017. Accessed: November, 2019. [Online] Available: http://tinyurl.com/yb2jcg6t.

[60] N. Blog. Using PeeringDB to set up your internet exchange peering. Accessed: June, 2019. [Online] Available: https://www.noction.com/blog/using-peeringdb-to-set-up-your-internet-exchange-peering.

[61] T. Böttger, F. Cuadrado, G. Tyson, I. Castro, and S. Uhlig. Open connect everywhere: A glimpse at the internet ecosystem through the lens of the netflix cdn. *ACM SIGCOMM Computer Communication Review*, 48(1):28–34, 2018.

[62] T. Böttger, F. Cuadrado, and S. Uhlig. Looking for hypergiants in peeringDB. *ACM SIGCOMM CCR*, 48(3):13–19, 2018.

[63] J. Brodkin. Netflix packets being dropped every day because Verizon wants more money. Ars Technica, Feb. 2014. Accessed: October, 2019. [Online] Available: https://tinyurl.com/wkxafbu.

[64] P. Brogan. Broadband Investment Continues Trending Down in 2016. USTELECOM (The Broadband Association), Oct. 2017. Accessed: January, 2018. [Online] Available: https://www.ustelecom.org/sites/default/files/documents/Broadband%20Investment%20Trending%20Down%20in%202016.pdf.

[65] P. Brogan. U.S. BROADBAND INVESTMENT REBOUNDED IN 2017. USTelecom Research Brief, Oct 2018. Accessed: May, 2019. [Online] Available: url-https://tinyurl.com/y3krg8dz.

[66] G. Budman. Backblaze B2 Drops Download Price By 60%, Apr. 2017. Accessed: January, 2018. [Online] Available: https://www.backblaze.com/blog/b2-drops-download-pricing/.

[67] M. Caesar and J. Rexford. BGP routing policies in ISP networks. *IEEE network*, 19(6):5–11, 2005.

[68] CAIDA. AS Relationships – with geographic annotations. Accessed: April, 2019. [Online] Available: http://www.caida.org/data/as-relationships-geo/.

[69] CAIDA. The CAIDA UCSD Anonymized Internet Traces - 20181220, 2018. Accessed: February, 2019. [Online] Available: http://www.caida.org/data/passive/passive_dataset.xml.

[70] M. Calder, X. Fan, Z. Hu, E. Katz-Bassett, J. Heidemann, and R. Govindan. Mapping the expansion of Google's serving infrastructure. In *Proceedings of the 2013 IMC*, pages 313–326. ACM, 2013.

[71] J. C. Cardona Restrepo and R. Stanojevic. A history of an internet exchange point. *ACM SIGCOMM Computer Communication Review*, 42(2):58–64, 2012.

[72] R. Castro, M. Coates, G. Liang, R. Nowak, and B. Yu. Network tomography: Recent developments. *Statistical science*, pages 499–517, 2004.

[73] R. Chandrasekaran and A. Tamir. Algebraic optimization: the Fermat-Weber location problem. *Mathematical Prog.*, 46(1-3):219–224, 1990.

[74] N. Chatzis, G. Smaragdakis, A. Feldmann, and W. Willinger. There is more to IXPs than meets the eye. *ACM SIGCOMM Computer Communication Review*, 43(5):19–28, 2013.

[75] Q. Chen, H. Chang, R. Govindan, and S. Jamin. The origin of power laws in Internet topologies revisited. In *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 608–617. IEEE, 2002.

[76] A.-S. Cheng, K. R. Fleischmann, P. Wang, E. Ishita, and D. W. Oard. Values of stakeholders in the Net neutrality debate: Applying content analysis to telecommunications policy.

In *System Sciences (HICSS), 2010 43rd Hawaii International Conference on*, pages 1–10. IEEE, 2010.

[77] R. Chirgwin. Google routing blunder sent Japan's Internet dark on Friday, 2017.

[78] B. Claise. Cisco systems netflow services export version 9. Technical report, 2004.

[79] P. K. Dey, M. A. Canbaz, M. Yuksel, and M. H. Gunes. On correlating isp topologies to their businesses. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–7. IEEE, 2018.

[80] P. K. Dey and M. Yuksel. CAR: Cloud-assisted routing. In *Network Function Virtualization and Software Defined Networks (NFV-SDN), IEEE Conference on*, pages 100–106. IEEE, 2016.

[81] P. K. Dey and M. Yuksel. Hybrid cloud integration of routing control & data planes. In *Proceedings of the 2016 ACM Workshop on Cloud-Assisted Networking*, pages 25–30. ACM, 2016.

[82] P. K. Dey and M. Yuksel. An Economic Analysis of Cloud-Assisted Routing for Wider Area SDN. *IEEE Transactions on Network and Service Management*, 2019.

[83] P. K. Dey and M. Yuksel. Peering Among Content-Dominated Vertical ISPs. *IEEE Networking Letters*, 1(3):132–135, 2019.

[84] A. Dhamdhere and C. Dovrolis. Twelve years in the evolution of the internet ecosystem. *IEEE/ACM Transactions on Networking (ToN)*, 19(5):1420–1433, 2011.

[85] A. Dixit, F. Hao, S. Mukherjee, T. Lakshman, and R. Kompella. Towards an elastic distributed SDN controller. In *ACM SIGCOMM Computer Communication Review*, volume 43, pages 7–12. ACM, 2013.

[86] C. Dulaney. Cablevision Revenue Rises but Users Decline. The Wall Street Journal, 2015. Accessed: November, 2019. [Online] Available: `http://tinyurl.com/yc9brzv3`.

[87] W. Eatherton, G. Varghese, and Z. Dittia. Tree bitmap: hardware/software IP lookups with incremental updates. *ACM SIGCOMM Computer Communication Review*, 34(2):97–122, 2004.

[88] N. Economides. The economics of the Internet backbone. 2005.

[89] F5 Networks, Inc. BIG-IP Link Controller (Datasheet). Accessed: May, 2019. [Online] Available: `https://www.f5.com/pdf/products/big-ip-link-controller-ds.pdf`.

[90] J. Farrell and P. J. Weiser. Modularity, vertical integration, and open access policies: Towards a convergence of antitrust and regulation in the Internet age. *Harv. JL & Tech.*, 17:85, 2003.

[91] N. Feamster. Revealing utilization at internet interconnection points. TPRC, 2016.

[92] N. Feamster, J. Rexford, and E. Zegura. The road to SDN: an intellectual history of programmable networks. *ACM SIGCOMM Computer Communication Review*, 44(2):87–98, 2014.

[93] P. C. Fishburn and A. M. Odlyzko. Dynamic behavior of differential pricing and Quality of Service options for the Internet. *Decision Support Systems*, 28(1):123–136, 2000.

[94] B. Ford, P. Srisuresh, and D. Kegel. Peer-to-Peer Communication Across Network Address Translators. In *USENIX Annual Technical Conference, General Track*, pages 179–192, 2005.

[95] V. Fuller, T. Li, J. Yu, and K. Varadhan. Supernetting: An address assignment and aggregation strategy. Technical report, RFC-1338, June, 1992.

[96] H. Galperín. Localizing internet infrastructure: Cooperative peering in latin america. *Telematics and Informatics*, 33(2):631–640, 2016.

[97] L. Gao and F. Wang. The extent of AS path inflation by routing policies. In *Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE*, volume 3, pages 2180–2184. IEEE, 2002.

[98] GENI. GENI Glossary. Accessed: October, 2019. [Online] Available: `https://www.geni.net/documentation/glossary/`.

[99] A. A. Gilroy. Net Neutrality: Background and Issues. CRS Report for Congress, Sep 2008. Accessed: November, 2018. [Online] Available: `https://fas.org/sgp/crs/misc/RS22444.pdf`.

[100] V. Giotsas, G. Smaragdakis, B. Huffaker, M. Luckie, et al. Mapping peering interconnections to a facility. In *Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies*, page 37. ACM, 2015.

[101] V. Giotsas, S. Zhou, M. Luckie, et al. Inferring multilateral peering. In *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*, pages 247–258. ACM, 2013.

[102] D. Goodin. Google goes down after major BGP mishap routes traffic through China, 2018. Accessed: October, 2019. [Online] Available: `https://tinyurl.com/y7gcvasj`.

[103] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker. NOX: towards an operating system for networks. *ACM SIGCOMM CCR*, 38(3):105–110, 2008.

[104] A. Gupta, M. Calder, N. Feamster, M. Chetty, E. Calandro, and E. Katz-Bassett. Peering at the internets frontier: A first look at isp interconnectivity in africa. In *International Conference on Passive and Active Network Measurement*, pages 204–213. Springer, 2014.

[105] J. Hawkinson and T. Bates. Guidelines for creation, selection, and registration of an Autonomous System (AS). Technical report, 1996.

[106] J. Hawkinson and T. Bates. Guidelines for creation, selection, and registration of an Autonomous System (AS). 1996.

[107] H. He, K. Xu, and Y. Liu. Internet resource pricing models, mechanisms, and methods. *Networking Science*, 1(1):48–66, 2012.

[108] Y. He, G. Siganos, M. Faloutsos, and S. V. Krishnamurthy. A Systematic Framework for Unearthing the Missing Links: Measurements and Impact. In *NSDI*, 2007.

[109] F. Hu, Q. Hao, and K. Bao. A survey on software-defined network and openflow: from concept to implementation. *IEEE Comm. Surveys & Tutorials*, 16(4):2181–2206, 2014.

[110] C. Huang, A. Wang, J. Li, and K. W. Ross. Measuring and evaluating large-scale CDNs. In *ACM IMC*, volume 8, pages 15–29, 2008.

[111] G. Huston. Interconnection, peering, and settlements. In *proc. INET*, volume 9, 1999.

[112] A. Ishimori, F. Farias, E. Cerqueira, and A. Abelém. Control of multiple packet schedulers for improving QoS on OpenFlow/SDN networking. In *2013 Second European Workshop on Software Defined Networks*, pages 81–86. IEEE, 2013.

[113] A. Iyer, P. Kumar, and V. Mann. Avalanche: Data center multicast using software defined networking. In *Communication Systems and Networks (COMSNETS), 2014 Sixth International Conference on*, pages 1–8. IEEE, 2014.

[114] M. Jammal, T. Singh, A. Shami, R. Asal, and Y. Li. Software defined networking: State of the art and research challenges. *Computer Networks*, 72:74–98, 2014.

[115] Job Snijders. Automated Peering Operations. NANOG 70, 2017. Accessed: October, 2019. [Online] Available: https://archive.nanog.org/sites/default/files/2_Snijders_Interconnection_Track_Automation.pdf.

[116] R. Johari and J. N. Tsitsiklis. Routing and peering in a competitive Internet. In *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, volume 2, pages 1556–1561. IEEE, 2004.

[117] C. Kim, M. Caesar, A. Gerber, and J. Rexford. Revisiting Route Caching: The World Should Be Flat. In *PAM*, volume 9, pages 3–12. Springer, 2009.

[118] E. Kim. Amazon Web Services Had 44 Price Cuts And You Can Expect More To Come, July 2014. Accessed: January, 2018. [Online] Available: http://www.businessinsider.com/amazon-web-services-had-44-price-cuts-2014-7.

[119] R. Klöti, B. Ager, V. Kotronis, G. Nomikos, and X. Dimitropoulos. A comparative look into public IXP datasets. *ACM SIGCOMM Computer Communication Review*, 46(1):21–29, 2016.

[120] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek. The Click modular router. *ACM Transactions on Computer Systems (TOCS)*, 18(3):263–297, 2000.

[121] T. Koponen, K. Amidon, P. Balland, M. Casado, A. Chanda, B. Fulton, I. Ganichev, J. Gross, P. Ingram, E. J. Jackson, et al. Network Virtualization in Multi-tenant Datacenters. In *NSDI*, volume 14, pages 203–216, 2014.

[122] J. Kurose and K. Ross. Computer networks: A top down approach featuring the internet. *Peorsoim Addison Wesley*, 2010.

[123] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet inter-domain traffic. In *ACM SIGCOMM CCR*, volume 40, pages 75–86. ACM, 2010.

[124] D. Lam, D. C. Cox, and J. Widom. Teletraffic modeling for personal communications services. *IEEE communications magazine*, 35(2):79–87, 1997.

[125] A. Lerner. Save Money on Networking Support/Maintenance Costs, May 2014. Accessed: February, 2018. [Online] Available: `https://blogs.gartner.com/andrew-lerner/2014/05/15/sweetmoolahwithoutunclericoh/`.

[126] A. Li, X. Yang, S. Kandula, and M. Zhang. CloudCmp: comparing public cloud providers. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pages 1–14. ACM, 2010.

[127] B. Li, J. Springer, G. Bebis, and M. H. Gunes. A survey of network flow applications. *Journal of Network and Computer Applications*, 36(2):567–581, 2013.

[128] T. A. Limoncelli. Openflow: a radical new idea in networking. *Queue*, 10(6):40, 2012.

[129] Linux Foundation Collaborative Project. FRRouting (FRR). Accessed: October, 2019. [Online] Available: `https://frrouting.org`.

[130] S. Lippert and G. Spagnolo. Internet peering as a network of relations. *Telecommunications policy*, 32(1):33–49, 2008.

[131] Y. Liu, S. O. Amin, and L. Wang. Efficient FIB caching using minimal non-overlapping prefixes. *ACM SIGCOMM Computer Communication Review*, 43(1):14–21, 2013.

[132] Y. Liu, B. Zhang, and L. Wang. FIFA: Fast incremental FIB aggregation. In *INFOCOM, 2013 Proceedings IEEE*, pages 1–9. IEEE, 2013.

[133] A. Lodhi, N. Larson, A. Dhamdhere, C. Dovrolis, et al. Using peeringDB to understand the peering ecosystem. *ACM SIGCOMM CCR*, 44(2):20–27, 2014.

[134] R. T. Ma, D. M. Chiu, J. Lui, V. Misra, and D. Rubenstein. Internet Economics: The use of Shapley value for ISP settlement. *IEEE/ACM Transactions on Networking (TON)*, 18(3):775–787, 2010.

[135] R. Mahajan, D. Wetherall, and T. Anderson. Negotiation-based routing between neighboring ISPs. In *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2*, pages 29–42. USENIX Association, 2005.

[136] P. Maigron. World - autonomous system number statistics - sorted by number. Accessed: October, 2019. [Online] Available: `https://www-public.imtbs-tsp.eu/~maigron/RIR_Stats/RIR_Delegations/World/ASN-ByNb.html`.

[137] S. Mainardi, E. Gregori, and L. Lenzini. The twofold nature of autonomous systems: Evidence combining stock market data with topological properties. In *INFOCOM, 2013 Proceedings IEEE*, pages 3171–3176. IEEE, 2013.

[138] J. C. McCallum. Memory Prices (1957-2017), 2018. Accessed: January, 2018. [Online] Available: `http://jcmit.net/memoryprice.htm`.

[139] D. McGrath. China Tariffs to Hit the Chip Sector, 2018. Accessed: February, 2019. [Online] Available: `https://www.eetimes.com/document.asp?doc_id=1333566#`.

[140] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. OpenFlow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, 38(2):69–74, 2008.

[141] D. McPherson and V. Gill. BGP MULTI_EXIT_DISC (MED) Considerations. 2006.

[142] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot. Traffic matrix estimation: Existing techniques and new directions. In *ACM SIGCOMM Computer Communication Review*, volume 32, pages 161–174. ACM, 2002.

[143] M. Menth, M. Duelli, R. Martin, and J. Milbrandt. Resilience analysis of packet-witched communication networks. *IEEE/ACM transactions on Networking (ToN)*, 17(6):1950–1963, 2009.

[144] M. Menth, R. Martin, and J. Charzinski. Capacity overprovisioning for networks with resilience requirements. In *ACM SIGCOMM Computer Communication Review*, volume 36, pages 87–98. ACM, 2006.

[145] D. Meyer. Management of ISPs, Peering, China Firewall Cause Big Challenges for SD-WAN.

[146] O. Michel and E. Keller. SDN in wide-area networks: A survey. In *Software Defined Systems (SDS), 2017 Fourth International Conference on*, pages 37–42. IEEE, 2017.

[147] C. Mills. The future of cord-cutting looks grim right now. BGR, Mar 2018. Accessed: November, 2019. [Online] Available: `https://bgr.com/2018/03/19/best-streaming-service-vs-cable-2018/`.

[148] A. Mohammed. Verizon Executive Calls for End to Google's 'Free Lunch'. Washington Post, Feb 2007. Accessed: November, 2019. [Online] Available: `http://www.washingtonpost.com/wp-dyn/content/article/2006/02/06/AR2006020601624.html?noredirect=on`.

[149] R. Molla and P. Kafka. Here's who owns everything in Big Media today. Recode. Accessed: November, 2019. [Online] Available: `https://tinyurl.com/y8qmdu7t`.

[150] M. Motiwala, A. Dhamdhere, N. Feamster, and A. Lakhina. Towards a cost model for network traffic. *ACM SIGCOMM Computer Communication Review*, 42(1):54–60, 2012.

[151] W. Mühlbauer, S. Uhlig, A. Feldmann, O. Maennel, B. Quoitin, and B. Fu. Impact of routing parameters on route diversity and path inflation. *Computer Networks*, 54(14):2506–2518, 2010.

[152] S. Musil. Netflix: We're the ones throttling video speeds on AT&T and Verizon. GeoMidpoint, Mar 2016. Accessed: April, 2019. [Online] Available: `https://www.cnet.com/news/netflix-admits-throttling-video-speeds-on-at-t-verizon/`.

[153] M. R. Nascimento, C. E. Rothenberg, M. R. Salvador, C. N. Corrêa, S. C. de Lucena, and M. F. Magalhães. Virtual routers as a service: the routeflow approach leveraging software-defined networks. In *Proceedings of the 6th International Conference on Future Internet Technologies*, pages 34–37. ACM, 2011.

[154] Netflix. Peering Guidelines. Accessed: June, 2019. [Online] Available: `https://openconnect.netflix.com/en/guidelines/`.

[155] D. G. Newnan, T. Eschenbach, and J. P. Lavelle. *Engineering economic analysis*, volume 2. Oxford University Press, 2004.

[156] Noction. NOCTION-Network Intelligence. Accessed: June, 2019. [Online] Available: `https://www.noction.com/intelligent_routing_platform/automate-bgp-network-management`.

[157] G. Nomikos, V. Kotronis, P. Sermpezis, P. Gigis, L. Manassakis, C. Dietzel, S. Konstantaras, X. Dimitropoulos, and V. Giotsas. O Peer, Where Art Thou?: Uncovering Remote Peering

Interconnections at IXPs. In *Proceedings of the Internet Measurement Conference 2018*, pages 265–278. ACM, 2018.

[158] W. B. Norton. The evolution of the US Internet peering ecosystem. *Equinix white papers*, 2004.

[159] W. B. Norton. *The Internet peering playbook: connecting to the core of the Internet*. DrPeering Press, 2011.

[160] W. B. Norton. Internet transit prices-historical and projected. In *The 2014 Internet Peering Playbook: Connecting to the Core of the Internet*, chapter 2, page 13. DrPeering Press, 2014.

[161] W. B. Norton. The Tier 1 ISP. In *The 2014 Internet Peering Playbook: Connecting to the Core of the Internet*, chapter 9, page 112. DrPeering Press, 2014.

[162] A. M. Odlyzko. Pricing and architecture of the Internet: Historical perspectives from telecommunications and transportation. In *Proceedings of TPRC*, volume 4, 2004.

[163] P. Pachal. How the AT&T-Time Warner deal threatens net neutrality. Mashable, Oct 2016. Accessed: October, 2018. [Online] Available: `https://mashable.com/2016/10/23/att-time-warner-fcc/#94uxT7hwQEqb`.

[164] R. PERKIN. Peering with Route Servers, 2016. Accessed: November, 2019. [Online] Available: `https://www.rogerperkin.co.uk/routing-protocols/bgp/peering-with-route-servers/?doing_wp_cron=1546013473.1083629131317138671875`.

[165] J. Pil Choi and B.-C. Kim. Net neutrality and investment incentives. *The RAND Journal of Economics*, 41(3):446–471, 2010.

[166] B. Popper. Apple reportedly paying internet providers to ensure speedy delivery of its data. The Verge, Jul 2014. Accessed: June, 2019. [Online] Available: https://tinyurl.com/y9vcemgj.

[167] P. Porras, S. Shin, V. Yegneswaran, M. Fong, M. Tyson, and G. Gu. A security enforcement kernel for OpenFlow networks. In *Proceedings of the first workshop on Hot topics in SDN*, pages 121–126. ACM, 2012.

[168] M. Prince. The Relative Cost of Bandwidth Around the World. Cloudflare Blog, August 2014. Accessed: June, 2018. [Online] Available: https://blog.cloudflare.com/the-relative-cost-of-bandwidth-around-the-world/.

[169] S. Ragner. The cloud price war continues: Amazon cuts its cloud storage prices, again, Nov. 2016. Accessed: January, 2018. [Online] Available: https://tinyurl.com/gojphbb.

[170] N. Rao. Bandwidth Costs Around the World, Aug. 2016. Accessed: February, 2018. [Online] Available: https://blog.cloudflare.com/bandwidth-costs-around-the-world/.

[171] G. Rétvári, J. Tapolcai, A. Kőrösi, A. Majdán, and Z. Heszberger. Compressing IP forwarding tables: towards entropy bounds and beyond. In *ACM SIGCOMM Computer Communication Review*, volume 43, pages 111–122. ACM, 2013.

[172] P. Rey and J. Tirole. A primer on foreclosure. *Handbook of industrial organization*, 3:2145–2220, 2007.

[173] L. G. Roberts. A radical new router. *IEEE spectrum*, 46(7):34–39, 2009.

[174] T. Roughgarden and É. Tardos. How bad is selfish routing? *Journal of the ACM (JACM)*, 49(2):236–259, 2002.

[175] P. S. Ryan and J. Gerson. A primer on Internet exchange points for policymakers and non-engineers. *Available at SSRN 2128103*, 2012.

[176] S. Secci, J.-L. Rougier, A. Pattavina, F. Patrone, and G. Maier. Peering equilibrium multi-path routing: a game theory framework for internet peering settlements. *IEEE/ACM Transactions on Networking (TON)*, 19(2):419–432, 2011.

[177] S. Sezer, S. Scott-Hayward, P. K. Chouhan, B. Fraser, D. Lake, J. Finnegan, N. Viljoen, M. Miller, and N. Rao. Are we ready for SDN? Implementation challenges for software-defined networks. *IEEE Communications Magazine*, 51(7):36–43, 2013.

[178] S. Shakkottai and R. Srikant. Economics of network pricing with multiple ISPs. *IEEE/ACM Transactions On Networking*, 14(6):1233–1245, 2006.

[179] A. Sharma, N. Silberman, L. Subramanian, and N. Economides. On the rise and fall of ISPs. In *Proceedings of Netecon 09: Workshop on the Economics of Networks, Systems, and Computations*, 2009.

[180] J. Sherry, S. Hasan, C. Scott, A. Krishnamurthy, S. Ratnasamy, and V. Sekar. Making middleboxes someone else's problem: network processing as a cloud service. *ACM SIGCOMM Computer Communication Review*, 42(4):13–24, 2012.

[181] N. Shetty, G. Schwartz, and J. Walrand. Internet QoS and regulations. *IEEE/ACM Transactions on Networking (TON)*, 18(6):1725–1737, 2010.

[182] J. Snijders. NLNOG RING, 2010. Accessed: November, 2019. [Online] Available: `https://ring.nlnog.net`.

[183] A. Soule, A. Lakhina, N. Taft, K. Papagiannaki, K. Salamatian, A. Nucci, M. Crovella, and C. Diot. Traffic matrices: balancing measurements, inference and modeling. In *ACM SIGMETRICS Performance Evaluation Review*, volume 33, pages 362–373. ACM, 2005.

[184] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson. Measuring ISP topologies with Rocketfuel. *IEEE/ACM Transactions on Networking (ToN)*, 12(1):2–16, 2004.

[185] V. Stocker, G. Smaragdakis, W. Lehr, and S. Bauer. Content may be king, but (peering) location matters: A progress report on the evolution of content delivery in the internet. 2016.

[186] Y. Sverdlik. Google and Level 3 Interconnect Network Backbones, 2016. Accessed: November, 2019. [Online] Available: https://www.datacenterknowledge.com/archives/2016/02/08/google-and-level-3-interconnect-network-backbones.

[187] Swisscom. Swisscom Peering Policy IP-Plus (AS 3303).

[188] D. Temkin. The Real Cost of Public IXPs, 2016. Accessed: November, 2019. [Online] Available: https://www.nanog.org/sites/default/files/Temkin_The_Real_Cost.pdf.

[189] ThousandEyes. ISP Peering. Accessed: June, 2019. [Online] Available: https://www.thousandeyes.com/learning/techtorials/isp-peering.

[190] Time Warner Cable. IPV4 and IPV6 Settlement-Free Peering Policy.

[191] R. Tumarkin and R. F. Whitelaw. News or noise? Internet postings and stock prices. *Financial Analysts Journal*, 57(3):41–51, 2001.

[192] S. Uhlig, B. Quoitin, J. Lepropre, and S. Balon. Providing public intradomain traffic matrices to the research community. *ACM SIGCOMM Computer Communication Review*, 36(1):83–86, 2006.

[193] Z. A. Uzmi, M. Nebel, A. Tariq, S. Jawad, R. Chen, A. Shaikh, J. Wang, and P. Francis. SMALTA: practical and near-optimal FIB aggregation. In *Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies*, page 29. ACM, 2011.

[194] I. van Beijnum. Transit vs peering: what makes sense when?, 2015. Accessed: November, 2019. [Online] Available: `https://packetpushers.net/transit-vs-peering-makes-sense/`.

[195] B. Van Schewick. Towards an economic framework for network neutrality regulation. *J. on Telecomm. & High Tech. L.*, 5:329, 2006.

[196] B. Van Schewick. Network neutrality and quality of service: What a nondiscrimination rule should look like. *Stan. L. Rev.*, 67:1, 2015.

[197] G. Varghese. *Network Algorithmics: an interdisciplinary approach to designing fast networked devices*. Morgan Kaufmann, 2005.

[198] S. Vissicchio, L. Vanbever, and O. Bonaventure. Opportunities and research challenges of hybrid software defined networks. *ACM SIGCOMM Computer Communication Review*, 44(2):70–75, 2014.

[199] H. Wang, Q. Jing, R. Chen, B. He, Z. Qian, and L. Zhou. Distributed Systems Meet Economics: Pricing in the Cloud. In *Proceedings of the 2Nd USENIX Conference on Hot Topics in Cloud Computing*, HotCloud'10, pages 6–6. USENIX Association, 2010.

[200] M. Weinberger. Snap will pay Google at least $400 million a year for the next 5 years, 2017. Accessed: January, 2018. [Online] Available: `http://www.businessinsider.com/google-cloud-platform-wins-snapchat-2017-2`.

[201] C. Welch. Netflix is now also paying AT&T to improve streaming quality. The Verge, Jul 2014. Accessed: June, 2019. [Online]

Available: https://www.theverge.com/2014/7/29/5949615/netflix-now-paying-att-to-improve-streaming-quality.

[202] William B. Norton. The History of Internet Peering. The Internet Peering Playbook. Accessed: November, 2018. [Online] Available: http://drpeering.net/white-papers/History-Of-Internet-Peering/History-Of-Internet-Peering.html.

[203] O. E. Williamson. Transaction-cost economics: the governance of contractual relations. *The journal of Law and Economics*, 22(2):233–261, 1979.

[204] B. Woodcock and M. Frigino. 2016 survey of internet carrier interconnection agreements, 2016. Accessed: November, 2019. [Online] Available: https://www.pch.net/resources/Papers/peering-survey/PCH-Peering-Survey-2016/PCH-Peering-Survey-2016.pdf.

[205] J. Wu, Y. Zhang, Z. M. Mao, and K. G. Shin. Internet routing resilience to failures: analysis and implications. In *Proceedings of the 2007 ACM CoNEXT conference*, page 25. ACM, 2007.

[206] T. Wu. The broadband debate, a user's guide. *J. on Telecomm. & High Tech. L.*, 3:69, 2004.

[207] H. Yan, R. Oliveira, K. Burnett, D. Matthews, L. Zhang, and D. Massey. BGPmon: A real-time, scalable, extensible monitoring system. In *2009 Cybersecurity Applications & Technology Conference for Homeland Security*, pages 212–223. IEEE, 2009.

[208] T. Yang, G. Xie, Y. Li, Q. Fu, A. X. Liu, Q. Li, and L. Mathy. Guarantee IP lookup performance with FIB explosion. In *ACM SIGCOMM Computer Communication Review*, volume 44, pages 39–50. ACM, 2014.

[209] S.-H. Yook, H. Jeong, and A.-L. Barabási. Modeling the Internet's large-scale topology. *Proceedings of the National Academy of Sciences*, 99(21):13382–13386, 2002.

[210] K. Zarifis, T. Flach, S. Nori, D. Choffnes, R. Govindan, E. Katz-Bassett, Z. M. Mao, and M. Welsh. Diagnosing path inflation of mobile client traffic. In *International Conference on Passive and Active Network Measurement*, pages 23–33. Springer, 2014.

[211] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg. Fast accurate computation of large-scale IP traffic matrices from link loads. In *ACM SIGMETRICS Performance Evaluation Review*, volume 31, pages 206–217. ACM, 2003.

[212] X. Zhao, Y. Liu, L. Wang, and B. Zhang. On the aggregatability of router forwarding tables. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE, 2010.

[213] G. andrli. The History of Content Delivery Networks (CDN). GlobalDots, Dec 2012. Available online: `https://www.globaldots.com/the-history-of-content-delivery-networks-cdn/`.