Examination of WAN Traffic Characteristics in a Large-scale Data Center Network

Zhaohua Wang[†] [‡], Zhenyu Li[†] [‡] [*], Guangming Liu[§], Yunfei Chen[§], Qinghua Wu[†] [‡] [*], Gang Cheng[§]
[†]ICT, Chinese Academy of Sciences [‡]University of Chinese Academy of Sciences [§]Baidu [*]Purple Mountain Laboratories
{wangzhaohua,zyli,wuqinghua}@ict.ac.cn,{liuguangming,chenyunfei,chenggang06}@baidu.com

ABSTRACT

Large cloud service providers have built an increasing number of geo-distributed data centers (DCs) connected by WAN to host their diverse services. While we have seen a large body of work on traffic engineering of WAN, the WAN traffic characteristics of production DC networks remain not well understood. In this paper, we report on the network traffic observed in Baidu's DC network (DCN) that consists of tens of geo-distributed DCs. Baidu hosts both traditional services like Web and Computing, as well as emerging services, such as Analytics, AI, and Map. We analyze WAN traffic characteristics in Baidu's DCN from the perspectives of traffic demands, traffic communication among DCs, and traffic characteristics of diverse services. Specifically, we focus on the disparity that might exist among different types of services. We also discuss the implications of our findings for WAN traffic engineering, fabric design, and service deployment.

CCS CONCEPTS

• **Networks** → **Network performance evaluation**; **Network measurement**;

KEYWORDS

DC-WAN, traffic pattern, traffic locality, traffic stability

ACM Reference Format:
Zhaohua Wang[†] [‡], Zhenyu Li[†] [‡] [*], Guangming Liu[§], Yunfei Chen[§], Qinghua Wu[†] [‡] [*], Gang Cheng[§]. 2021. Examination of WAN Traffic Characteristics in a Large-scale Data Center Network. In *ACM Internet Measurement Conference (IMC '21), November 2–4, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3487552.3487860

1 INTRODUCTION

Large cloud service providers use tens of geographically distributed data centers (DCs) that are interconnected by wide-area network (WAN) to host their diverse services. Services are replicated across these DCs to process users' requests locally for better QoE (Quality-of-Experience). As such, traffic of bulk transfers flows among DCs for data synchronization or backup. For example, search engines synchronize the indexes between DCs; cloud storage services backup user data in multiple DCs for reliability. These bulk transfers may have deadlines but are not delay sensitive [17, 19, 23]. WAN also carries delay sensitive traffic [14]. For instance, the front-end web search servers in a DC may need to communicate with the ads service that is located in other DCs to consolidate the response; distributed machine learning may need to aggregate the gradients from many DCs when the raw data are not allowed to move to other regions due to local regulations [15]. Delay sensitive traffic is marked as *high-priority* traffic demand, while the bulk transfers for data synchronization are *low-priority* traffic demands.

DC-WAN is an expensive resource, but providers have to balance between utilization and availability [7]. In the past few years, we have seen several solutions aiming at achieving a better balance by using software defined networking constructs [14, 16, 17] or fine-grained policy enforcement [22]. As any traffic engineering solution, the effectiveness of these approach depends on the traffic patterns. For instance, both SWAN [14] and BwE [22] assume the good predictability of the high-priority interactive traffic. Although real inter-DC traffic traces were used for evaluation in some of these studies, the WAN traffic patterns of large-scale production DCs remain not well understood.

Besides, apart from traditional web service and Hadoop service, modern DCs carry many new services. AI and big data analytic services, which are very common and important for large service providers, now often leverage servers in geo-distributed DCs for location-based services and distributed deep learning [15, 18, 26]. For instance, map and the relevant location-based recommendation rely on geo-distributed front-end servers for user-facing requests, which may interact with servers in other DCs for real-time road traffic information and target ads. These emerging services may exhibit new characteristics and change the overall WAN traffic patterns. An examination of these services from the perspective of traffic characteristics will benefit the design and improvement of traffic engineering in DCN.

Indeed, many DCN designs are motivated by measurement results. The design of VL2 [11] is driven by the tremendous volatility in the traffic among servers in a DC; the traffic mix pattern and the switch buffer behavior motivate the design of DCTCP [2]; the predictability of traffic at short time-scales motivates the design of MicroTE [6]. There are also some measurement studies that report the nature of traffic in DCN. Kandula *et al.* described the characteristics of traffic in an operational distributed query processing cluster using socket-level logs [20]. Benson *et al.* examined the flow characteristics, traffic locality and link utilization of several DCs [4], and observed the high locality of traffic within individual racks in cloud DCs. Roy *et al.* extended the previous studies that are mostly about a single DC provider and examined the traffic patterns in the Facebook's data center [27]; they found different characteristics of traffic than previous measurements. The distinctions in traffic

IMC '21, November 2–4, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-9129-0/21/11…$15.00
https://doi.org/10.1145/3487552.3487860

patterns of different DC providers indeed call for more reports of other DCs. It is also worth noting that all these studies focus on *intra* DC traffic. Chen *et al.* [8] studied the inter-DC traffic in Yahoo!, but the scale is very small (only 5 DCs) and the application mix is much simpler than cloud WAN application mixes.

This paper measures Baidu's DCN, with a special focus on the traffic that flows across DCs and clusters. Baidu uses tens of geo-distributed DCs serving millions of users each day, where each DC contains multiple clusters to organize servers through a set of racks. It is one of the largest web service providers, and in recent years, it has launched AI and auto-driving services [3]. Large-scale geo-distributed DCs with a complex service mix make Baidu's DCN one of the good examples for the examination of traffic patterns of modern DCs.

Specifically, motivated by challenges encountered by network designers and operators, we study traffic characteristics along three dimensions:
1) Examining traffic demands. The effective design of the DC-WAN resource allocation depends heavily on the traffic demands. We thus first examine the DC-level traffic locality and link utilization.
2) Characterising traffic communication. Higher utilization of links carrying WAN traffic motivates a further analysis of traffic communication between DCs, with a special focus on the stability.
3) Analyzing traffic of services. Service migration and service-level WAN bandwidth allocation require a deep understanding of the traffic characteristics at service level. To this end, we analyze service interaction and stability from the traffic perspective.

To this end, we develop and deploy a Netflow collector that runs in both core switches and data center switches in all Baidu's DCs. We also utilize SNMP statistics from these data center switches. We make the following key observations from our study:

- Despite that services are highly replicated in many DCs, about 20% of high-priority traffic that leaves clusters still flows across DCs over WAN. This percentage, however, varies across service categories and over time of a day; the emerging services (AI, Analytics and Map) deviate significantly from the traditional Web and Computing services.
- The links that carry WAN traffic experience higher utilization than those in DCs, and their loads are well balanced thanks to ECMP. Besides, WAN traffic and DC traffic leaving from clusters in individual DCs show a high temporal correlation in terms of their incremental value, suggesting a separation of two types of traffic (WAN traffic and DC traffic) on two kinds of switches (as opposed to using one type of data center switches in [28]) to avoid interference.
- Although communications are prevalent among DCs, a small portion (8.5%) of DC pairs contribute 80% of high-priority traffic; these heavy hitters are also persistent over time. The traffic communication within DCs is also imbalanced: 17% of rack pairs generate 80% of the traffic.
- The aggregated high-priority traffic over WAN and the high-priority traffic exchanges among DCs remain stable over time, leading to good predictability of overall traffic demands. Intra-DC traffic exchanged among clusters, however, is variable; the design of fabrics need to adapt to this volatility in traffic scheduling [11].
- Our analysis reveals different interaction patterns among services (from the perspectives of WAN traffic): traditional Web and Computing services heavily interact with each other, implying a close bind between them; Analytics, AI, Map and Security services, on the other hand, distribute their traffic to others more evenly, implying their fundamental contributions for other services.
- We see a great disparity of the stability of high-priority WAN traffic among services. The stability is partially impacted by the service interaction pattern. Existing traffic estimation methods when applied for service-level WAN traffic prediction may perform poorly, especially for those services whose traffic stability does not persist for a long time. These observations call for more accurate estimation methods for WAN traffic engineering at the service level.

We further discuss the implications of the above findings for WAN traffic engineering, service migration and placement, network fabric design for DCs, and switch configuration. We are also aware that as with any large-scale empirical study, our results are subject to the limitation of considering only one DC provider. While these observations indeed need to be reexamined further to confirm their generalisation, they do provide us a deeper understanding of the traffic and service characteristics in modern DCs (especially for the DC-WAN traffic). We hope that our findings can shed light on DC interconnect design, traffic engineering in DC-WAN and service placement in DCs.

The rest of this paper is organized as follows: Section 2 provides a brief description of Baidu's DCN and introduces the measurement methodology. We then analyze the traffic patterns across DCs and clusters with respect to traffic demands (Section 3) and traffic communication (Section 4). At last, we give insight into the WAN traffic characteristics in view of services (Section 5). Finally, we introduce related works in Section 6 and conclude this paper in Section 7.

2 BACKGROUND & DATA

This section begins with a brief description of Baidu's data center (DC) network, followed by details of the measurement methodology and the data we used in this paper.

2.1 Baidu's Data Center Network

Baidu's DC network (DCN) hosts various large-scale services; it is built on an infrastructure of DCs connected through high bandwidth (Tbps) wide area networks (WANs). As shown in Figure 1, the network consists of multiple DCs connected to the WAN via core switches, which form a full-meshed core network at the overlay layer. Inside a DC, tens of clusters are connected by links of Tbps through DC switches that are responsible for traffic inside the DC. The traffic that goes out of a DC flows through xDC (cross-DC) switches to the core switches. Each cluster either employs a typical 4-post structure or a Spine-Leaf Clos design. Server machines are organized into racks and connected to a top-of-rack switch (ToR switch). In the 4-post structure, racks are connected through cluster switches, which in turn communicate with each other via

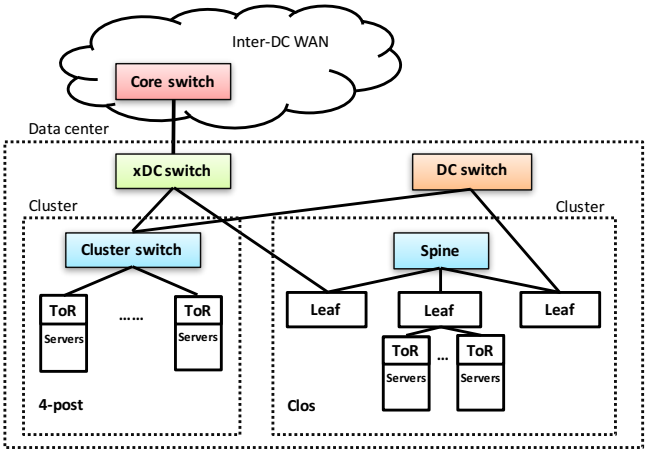


Figure 1: Datacenter topology.

DC switches. In the Spine-Leaf Clos structure, racks are connected through leaf switches; racks in the same pod are served by the same set of leaf switches. Leaf switches are then full-meshed connected with spine switches for inter-pod traffic. A particular set of leaf switches are dedicated to intra-DC traffic, as such they connect to DC switches in the DC; another set of leaf switches connect to xDC switches for inter-DC traffic. Overall, Baidu's DCN is similar to others (*e.g.* Facebook, Microsoft) [11, 27] that were previously reported in literature from the topology perspective. Note that WAN that connects DCs is an expensive resource.

From the service hosting perspective, however, some differences are notable. First, Baidu's DCN hosts many services that were not reported in other DC networks, despite the dominance of web service. These services include the emerging distributed AI and location based services (*e.g.* Baidu Map). We will detail major services later in this section. Second, Baidu's DCN allows any service to be run on any server. This flexibility leads to the fact that, while a physical server in Baidu's DCs only hosts one specific service, a rack may host many types of services; this is different from Facebook's DCN [27] where a rack deploys the same service.

2.2 Data Collection Methodology

This paper focuses on the traffic that crosses DCs and clusters, with an emphasis on the impact of new types of services. That said, we do not capture the micro view of fine-grained flow characteristics as in [4, 5, 20, 27] or traffic burst behavior as in [21, 34]. To this end, we utilize the sampled Netflow data and the SNMP data collected across Baidu's DCN. [1] Note that we focus on the traffic related to Baidu specific in-house services in this paper, *i.e.*, our dataset does not include cloud customer traffic. [2]

[1]The DCs we examined in this paper are distributed across multiple regions inside China.

[2]Indeed, the cloud customer services may show different behavior than Baidu's in-house services in terms of traffic volume, replications in DCs, interaction with other services and etc. Besides, we do not have service category information for individual flow records of cloud customer traffic. This prevented us to include cloud customer traffic as we focus on traffic characteristics at service-level.

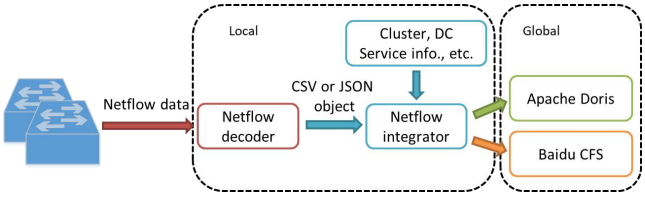


Figure 2: Netflow data collection architecture.

2.2.1 ***Netflow data.*** Cisco's Netflow service provides network administrators with access to summarized IP flow records within their networks [9]. Figure 2 shows the process of Netflow data collection, where Netflow data were from switches in Figure 1 across Baidu's entire DCN. Specially, we collected Netflow data from core switches for inter-DC traffic analysis, and Netflow data from DC switches for inter-cluster (but intra-DC) traffic analysis.

The active timeout for NetFlow on all switches is set to 1 minute, *i.e.* a Netflow record is exported every 1 minute for long-lived flows. Each flow records the aggregated flow information obtained from the sampled packet headers with 1:1024 sampling rate; a log contains the source and destination IP addresses, transport-layer port numbers and IP protocol. These collected flow data along with other metadata such as collection machines' IP addresses and capture time are first processed by the *Netflow decoders*, which convert each log into a CSV or JSON object[3]. These parsed data are then streamed to *Netflow integrators* using a distributed subscribing and streaming system. *Netflow integrators* aggregate the traffic flow data at one minute interval and further annotate it with additional attribution information such as the cluster, DC, service identifications and QoS information (indicating the priority of the flow) corresponding to each flow log by querying other data sources. The service information is identified via querying a directory that keeps the mapping between IP addresses and port numbers to services. *Netflow integrators* then feed data into *Apache Doris*, a fast MPP database for big data analytics [10] and *Baidu CFS*, a cloud file system built in Baidu for data storage and offline analysis. *Netflow decoders* and *Netflow integrators* are deployed locally in DCs for processing data collected from individual DCs, while the data analytics and storage systems are centralized deployed for processing globally collected flow data.

In aggregation, over 10 TB raw Netflow data from core switches and 10 TB from DC switches are generated every day. Note that, during the collection of the data used in this paper, we did not notice any abnormality of our Netflow data collection system.

2.2.2 ***SNMP data.*** In order to investigate the link utilization of cluster-to-DC, cluster-to-xDC, and xDC-to-core links, we also collected SNMP data from the interfaces of DC switches and xDC switches in multiple DCs that host considerably traffic volume. Every 30 seconds, the SNMP manager requests traffic statistics from DC switches and xDC switches. The collected SNMP data is streamed into time series tables and Apache Doris in Baidu for analysis and storage. We note the possible measurement inaccuracy caused by SNMP data collection, *e.g.* SNMP packet loss or delay. As

[3]Those records that fail to be parsed due to format issues are discarded. The percentage of failed records is around 0.00001%.

such, instead of directly using collected statistics, we aggregated them into 10-minute intervals and used the aggregated statistics for analysis. Note that SNMP data were used only for link utilization analysis.

2.3 Major Services

Overall, there are over 1,000 services hosted in Baidu's DCN. Nevertheless, the aggregated traffic volumes over services follow a skewed distribution: less than 20% of services account for over 99% of traffic volume. These services generally can be divided into 10 categories based on their functionalities as shown in Table 1, where the categories are sorted in a descending order of traffic volumes. We also present the number of services and the percentage of high-priority traffic of each category. *High-priority* traffic are delay sensitive traffic driven by Internet-facing requests (*e.g.* web search queries); *Low-priority* traffic, on the other hand, are usually from batch computing services (*e.g.* Hadoop, Spark, *etc.*), which can tolerate a certain delay with pre-assigned deadlines on the completion.

To put the categories into context, we briefly describe each category of services. *Web* services are for search engine, which account for the largest share of traffic. This is expected as Baidu is the largest search engine provider in China. As expected, web services are dominated by high-priority traffic. On the contrary, majority of *Computing* traffic is of low priorities as the services (*e.g.* Hadoop and Spark) are of batch nature. *Analytics* services are used for news feeds, ads and user behavior analysis. They are also more sensitive to delay because they are called by other services like web searches. *DB* consists of database services such as SQL, NoSQL and Redis; *Cloud* provides cloud storage and cloud computing services. These two types of services are mostly non-interactive. *AI* services are for distributed machine learning and deep learning, which emerge recently given the huge data and models [25]. They are fundamental for Baidu to support vehicle auto-driving applications, search and recommendations. These services may sync data among clusters/DCs, and generate 65% of low-priority traffic. They may also perform distributed training, which corresponds to high-priority traffic (35%). *FileSystem* represents the distributed file systems. *Map* provides location-based services, including navigation and location based recommendation. Map services are often triggered by Internet users' requests, leading to the dominance of high-priority traffic. Finally, *Security* provides security management for DCN. While the traffic patterns of DCN hosting conventional services like Web and Computing have been reported before [4, 8, 27], the impact of the emerging services, like AI and Map, has not been examined before. Indeed, with such a variety of commercial application services, Baidu's DCN provides research community a unique opportunity to understand the traffic patterns of modern DCNs. The priority of a flow's traffic is labeled by end servers in each packet using the DSCP field. Bandwidth allocation and traffic engineering at WAN level depend on this field, in order to provide low-latency transmission for high-priority traffic. Hence, we focus on high-priority traffic when examining inter-DC WAN traffic patterns, while we do not distinguish these different types of traffic for inter-cluster traffic pattern examination as in other measurement studies [14, 19].

Table 1: Major service categories, presented with the number of top services and the percentage of high-priority traffic for each category.

Category	Service #	Highpri %	Description
Web	15	78.1	Searching engine
Computing	25	17.8	Stream and Batch computing
Analytics	23	67.3	Feeds, Ads and user Analysis
DB	10	31.2	Databases
Cloud	15	30.0	Cloud storage and computing
AI	17	35.4	AI techniques
FileSystem	3	50.2	Distributed file systems
Map	2	76.7	Geo-location and navigation
Security	3	0.8	Security management
Others	16	43.2	Network operation
Total	129	49.3	

3 TRAFFIC DEMANDS

The effective design of WAN resource allocation depends heavily on the traffic demands of services. In this section, we first investigate how much traffic that leaves clusters flows out of DCs to WAN (*i.e.* the traffic locality) with an emphasis on variations among services. We observed this percentage is as high as 20% for high-priority traffic. Given this observation, we further analyze the utilization of the links that carry WAN traffic for better network design and configuration.

3.1 Traffic Locality

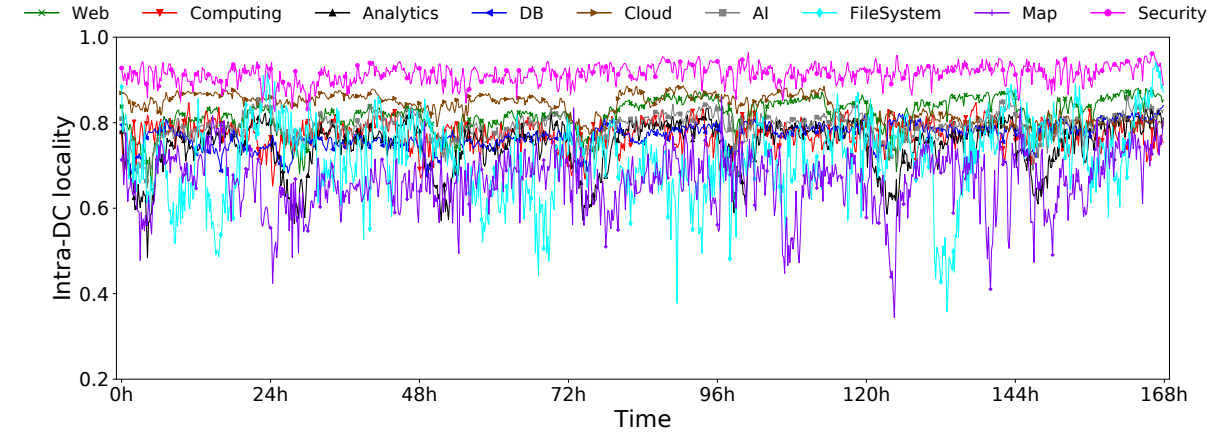
We first examine whether the traffic leaving clusters flows to clusters inside DCs (intra-DC traffic) or to other DCs in Table 2, where we break down traffic based on the priority. We calculate the intra-DC traffic percentages over one week of data and make the following findings: 1) Most of the aggregated traffic (78.3%) leaving clusters resides within DCs, indicating a high intra-DC traffic locality. This percentage is much higher than the Facebook's DCN, which is around 40% [27]. 2) We also see higher traffic locality for high-priority traffic. Low-priority traffic, on the other hand, is about 2× likely to flow out DCs than high-priority traffic (32.9% vs. 15.7%); this is due to the data sync of individual services among DCs.

Traffic locality across service. We break down traffic into service categories to examine the discrepancy among services. The first relevant question is whether the constituents of individual services in terms of intra-DC traffic and inter-DC traffic are similar. To this end, we compute the rank correlation between two service lists, where services in one list are ranked by their intra-DC traffic volumes and services in the other one are ranked by their inter-DC traffic volumes. The Spearman coefficient is above 0.85 and the Kendall's tau coefficient is 0.7, both implying a large overlap of two lists and a high similarity of the constituents of services.

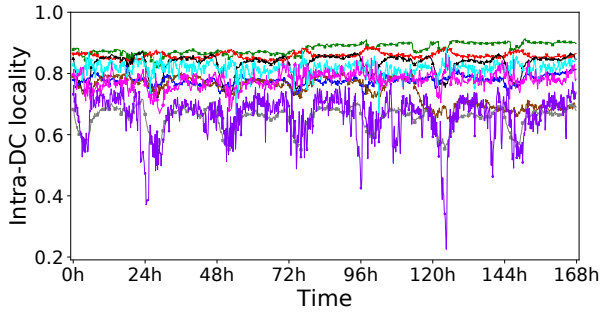
Table 2 compares the traffic locality of different types of services. The locality varies across services greatly. For instance, Map services show the least DC locality for both the aggregated traffic and high-priority traffic. A close investigation reveals that users may request real-time road traffic information of other geo-distant regions that is computed and stored locally. Besides, while we see a higher locality of high-priority traffic from the perspective of aggregated traffic, some service categories show the opposite. For instance, only 66.4% of the high-priority traffic generated by the

Table 2: Traffic locality for different categories of services.

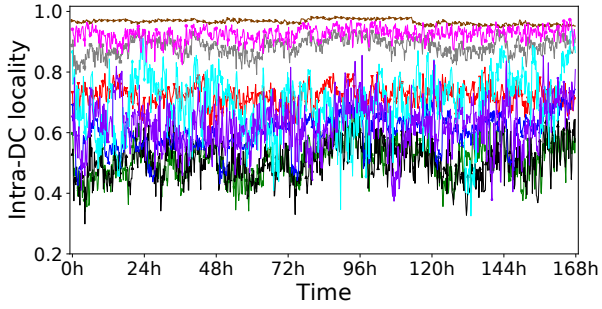
Intra-DC locality %	Total	Web	Comput.	Analytics	DB	Cloud	AI	FileSys.	Map	Security
All traffic	78.3	82.4	77.2	75.7	76.9	84.2	79.5	71.1	66.0	91.5
High-priority	84.3	88.2	85.6	83.9	77.9	75.3	66.4	81.7	66.0	78.1
Low-priority	67.1	50.5	72.0	50.3	59.7	96.7	88.7	69.3	63.5	92.8



(a) All traffic



(b) High-priority



(c) Low-priority

Figure 3: Dynamics of traffic locality for different types of services during a week; the labels on x-axis mark the hours of the week; each data point corresponds to a 10-minute interval.

AI services remains within DCs, which is much lower than that from the aggregated perspective (79.5%) and from the low-priority perspective (88.7%). Similar results can also be observed in Cloud and Security services, which probably stems from the deployment of geo-distributed jobs. For example, the geo-distributed machine learning system spans multiple data centers to reduce large data transfers and meet the constraints of privacy and data sovereignty laws [15]. These disparities in traffic locality among services urge the need to examine traffic patterns of different services.

Traffic locality dynamics. Next, we examine the dynamics of traffic locality over a course of one week in Figure 3, where every 10 minutes we compute the fraction of intra-DC traffic for each category of services. The locality of total traffic keeps relatively stable for most of the services, except for those services that have a higher ratio of high-priority traffic (see Table 1). Specifically, for Web, Map, Analytics and FileSystem services, the coefficient of variation of their traffic locality ranges from 0.05 to 0.13, while this value of other services is less than 0.04. The locality dynamics of high-priority traffic (Figure 3(b)) indeed show a clearly diurnal pattern for most of the services as the traffic is driven by Inter-facing user requests. The lowest intra-DC locality for the high-priority traffic happens between 2 to 6 a.m., indicating that the high-priority traffic during this time period is more likely to flow out of DCs. Special care should be taken here as periodical jobs for data sync and backup (thus generating low-priority traffic) are often scheduled during this period too. The low-priority traffic does not show a clearly diurnal pattern in locality dynamics, but its variation can be

very large (Figure 3(c)). This is because low-priority traffic is more driven by planned jobs that may be scheduled periodically.

3.2 Link Utilization

We next examine the link utilization using the SNMP data collected from DC switches and xDC (cross-DC) switches. We find, in general, the utilization of xDC-core links is higher than that of cluster-DC/xDC links, *i.e.*, the link utilization increases with higher levels of aggregation. This observation complies with previous measurements of other DCNs employing similar structure [4, 5, 27].

Given the high utilization of xDC-core links, we are curious about whether the load among these links are balanced or not. Note that ECMP is applied in Baidu's DCN for load balancing. Despite of known shortcomings of ECMP in load balancing, *e.g.* hash collision may lead to significant imbalance if there are a few large flows [1], we observe it can achieve a good balance for traffic going through xDC-core links to WAN. To show this, we calculate the coefficient of variation of the utilization among links [4] between each xDC-core switch pair at 10-minute intervals. Figure 4 depicts the median over 10-minute intervals in a week for individual xDC-core switch pairs. The coefficient of variation is as low as 0.04 for over 80% xDC-core switch pairs, indicating a good load balance.

Link utilization dynamics. While some studies [28] assume that a single switch carries both WAN traffic and DC traffic, Baidu's DCN uses two types of switches, namely xDC switches and DC switches, to separate the WAN traffic from the DC traffic. There are several reasons for this. First, using consolidated switches to host both WAN traffic and DC traffic creates significant challenges for WAN flows that are bottlenecked at data center switches due to the shallow buffer and bursty DC traffic [28]. Second, the long-term traffic of the two types compete with each other, as we find in Figure 5, where we examine the variation of average link utilization for cluster-DC links and cluster-xDC links in a typical DC over a week. The utilization of these two types of links exhibit strong daily and weekly patterns with lower utilization on weekends. The cross correlation between the increments of these two time series is as high as over 0.65, indicating the high correlation between two types of traffic. Separating the intra-DC and inter-DC traffic into different switches thus help avoid the possible competition of switch resources. The third reason is to enable two types of switches to upgrade separately, which in turn saves cost. As we have seen in Table 2, DC traffic overall is far more than WAN traffic, so it requires increasing number of switches to hold the growing DC traffic. DC switches are mostly commodity low-cost switches, while the xDC switches require higher aggregated bandwidth to core switches and are much more costly. Fortunately, xDC switches upgrade much less frequently due to the relative lower and stable traffic volume.

3.3 Summary and Implications

In summary, despite that services are highly replicated in many DCs, 20% of high-priority traffic that leaves clusters still flows across DCs over WAN. This percentage, however, varies across service categories and over time of a week. Specially, the emerging services (AI, Analytics and Map) exhibit much disparities in comparison with the traditional Web and Computing services. For example, Map services show the least intra-DC locality for both the aggregated traffic and high-priority traffic; the locality of the high-priority traffic for AI services is much lower than that in terms of the aggregated and low-priority traffic. These observations indicate the challenges of WAN traffic engineering, which needs to take the disparity among services into consideration for service-level bandwidth allocation. The observations also imply the urgent need of a detailed analysis on traffic characteristics for different services, which is the object of this work.

[4]These links are with the same capacity.

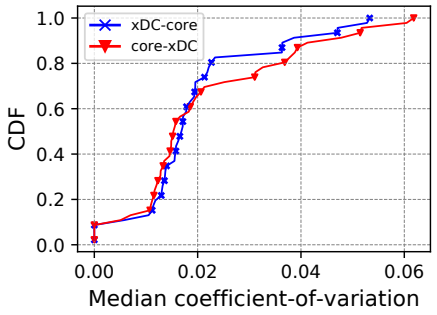


Figure 4: The coefficient of variation of the utilization among links between xDC and core switches

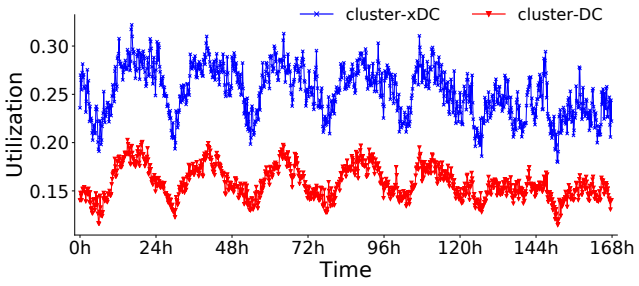


Figure 5: Utilization of cluster-DC and cluster-xDC links is temporally correlated over a week; the labels on x-axis mark the hours of the week; each data point corresponds to a 10-minute interval.

The link utilization analysis reveals the links that carry WAN traffic experience higher utilization than those in DCs, and we observe a balanced load among the links carrying WAN traffic. We also find a high correlation between WAN traffic and DC traffic time series. These findings show the viability of ECMP in our DCN in practice considering its simplicity, and suggest a separation of inter-DC (WAN) and intra-DC traffic into two types of switches (as opposed to [28]). In doing so, we can avoid interference of the two types of traffic, simplify the congestion control design, and achieve low cost and high scalability.

4 TRAFFIC COMMUNICATION

Understanding traffic communication patterns across DCs and clusters (within DCs) is critical for WAN traffic engineering and fabric design. In this section, we examine both inter-DC (high-priority)

traffic matrix and inter-cluster (all) traffic matrix. Our major findings include: (1) biased WAN traffic distribution over DC pairs towards a small set of heavy hitters; (2) the traffic exchange pattern of these heavy hitters remains stable, leading to a good predictability of the aggregated high-priority traffic; (3) the total traffic across clusters inside a DC is imbalanced and variable.

4.1 Inter-DC Traffic Matrix

Communication patterns among DCs provide a view of load in interconnect WAN, and guide the design of WAN traffic engineering and bandwidth allocation. We specially focus on high-priority traffic, because priority queuing at switches will ensure enough capacity for the high-priority traffic at first if resource contention does occur [14]. We find a skewed traffic distribution on the inter-DC traffic matrix for high-priority traffic, where 8.5% of DC pairs contribute 80% of high-priority traffic. Moreover, the set of heavy hitters that contribute 80% of traffic remains the same over time.

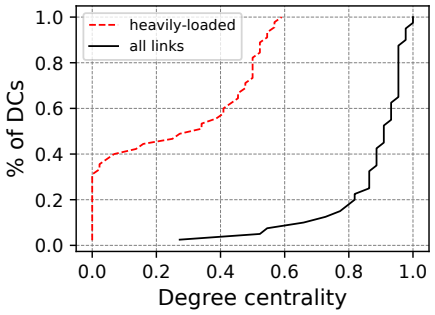


Figure 6: Degree centrality of each data center.

To further inspect the interaction patterns among different DCs, we count the number of DCs that each DC communicates with (*i.e.*, degree centrality) and plot the distribution in Figure 6. We observe an extensive communication pattern: out of all DCs, 85% communicate with more than 75% of the other DCs. We further look into the heavily-loaded connections, where a connection is labeled as heavily loaded one if the traffic volume exceeds 1Gbps. We find that still over 50% of DCs communicate with 40%-60% of other DCs. These results indicate that although communications are prevalent among DCs, most of traffic is concentrated on a few DC connections.

Traffic matrix variation. We next examine the variation of the high-priority traffic matrix over time, which reflects the flux in the traffic exchange pattern among DC pairs. For a time point t, the change rate r_{TM} of a traffic matrix TM is computed as [20]:

$$r_{TM}(t) = \frac{|TM(t+\tau) - TM(t)|}{|TM(t)|} \quad (1)$$

where the numerator is the absolute sum of the entry wide differences of the two matrices at adjacent time intervals, and the denominator is the absolute sum of entries in $TM(t)$, which equals to the aggregated traffic $T(t)$. For comparison, we also compute the change rate of the aggregated traffic as:

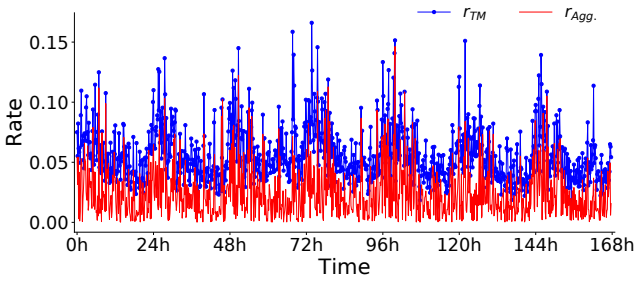


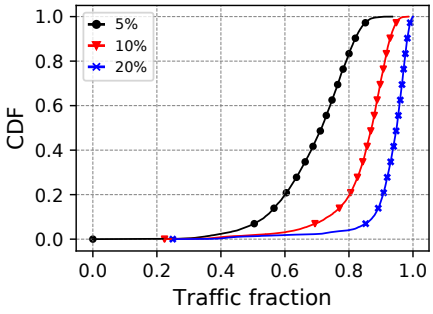
Figure 7: Change rates of the aggregated high-priority traffic and the traffic matrix of the heavy DC pairs in a week; the labels on the x-axis mark the hours of the week; each data point corresponds to a 10-minute interval.

$$r_{Agg.}(t) = \frac{|T(t+\tau) - T(t)|}{T(t)} \quad (2)$$

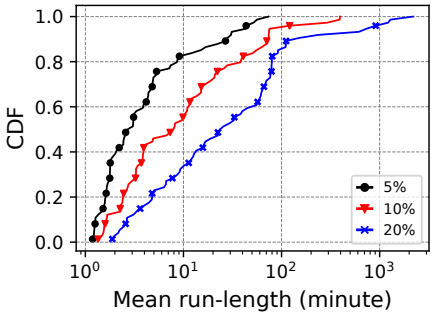
Note that even the aggregated traffic remains unchanged (*i.e.* $r_{Agg.} = 0$), the exchange traffic pattern among DCs (measured by r_{TM}) may change greatly. For instance, let us suppose TM contain two elements, and $T(t) = 4$, $TM(t) = [2, 2]$; at time point $t + \tau$, if $TM(t+\tau) = [1, 3]$, $r_{Agg.}(t) = 0$, but $r_{TM} = 2/4 = 0.5$.

Figure 7 plots the $r_{Agg.}$ and r_{TM} on 10-minute intervals during a week (*i.e.*, $\tau = 10min$). Here we only consider the heavy hitters (8.5%) that contribute 80% of the traffic. Overall, both the aggregated traffic and the traffic exchange across DCs remain stable with the change rate below 10% for most of the time intervals. Nevertheless, the traffic exchange pattern among DC pairs may change in some time intervals when the aggregated traffic remain almost unchanged ($r_{Agg.}$ close to 0). This observation shows that even when the aggregate inter-DC traffic remains stable, the traffic exchanged among DCs may change. Besides, the change rate of the inter-DC high-priority traffic follows a typical daily pattern, which is possibly driven by the variation of load.

Predictability analysis. The above analysis indicates that the overall inter-DC traffic is relatively stable over time. Next, we give insight into how the traffic exchanged between each pair of heavy DCs changes and whether it is predictable on a 1-minute time scale, which is crucial for bandwidth allocation and traffic engineering in the WAN [14, 16, 22]. To this end, we compute the fraction of total traffic contributed by those DC pairs that have no significant change in traffic like in [6] (*i.e.*, change rate is below the stability threshold thr on a 1-minute time scale), and depict the distribution in Figure 8(a), where thr is set to 5%, 10%, 20%. Even with a stringent stability threshold of 5%, for 80% of 1-minute intervals, over 60% of the total traffic is contributed by the DC pairs that remain stable in terms of traffic; this traffic share goes beyond 90%, if we allow 20% of change (*i.e.*, $thr = 20\%$). In addition, we find that the coefficient of variation of the high-priority traffic volume for each DC pair ranges from 0.05 to 0.82 (with 0.32 on median), which is consistent with the observations in [19]. Taken together, these results imply the possibility of estimating the aggregated high-priority WAN traffic based on the historical traffic data. Given the trend of incorporating multiple service priorities into the WAN traffic engineering [7, 14],



(a) Distribution of the fraction of total inter-DC traffic under 3 different change rate upper bounds



(b) Distribution of run-length of insignificant change

Figure 8: High-priority WAN traffic predictability.

we return to studying the service-level traffic predictability and traffic estimation in Section 5.

DC-WAN is an expensive resource; in order to make full use of the network bandwidth, providers usually conduct traffic scheduling during the time periods when traffic remains stable. We thus further analyze the persistence of the stability. Specially, we examine the run-length of the sequence of minutes where change in traffic for each DC pair remains insignificant, *i.e.*, less than *thr* compared to the demand at the beginning of the sequence. Figure 8(b) shows the results. We observe that 40% of the DC pairs remain predictable for over 5 minutes when $thr = 5\%$; this percentage goes up to 80% if we can tolerate 20% of change (*i.e.* $thr = 20\%$).

4.2 Inter-Cluster Traffic Matrix

Using the same analysis methodologies, we analyze the traffic communication among clusters in a typical DC, in view of the significant amount of traffic among clusters (see Table 2). A good understanding of inter-cluster traffic characteristics is also useful for the DC fabric design [27].

We first focus on the inter-Cluster traffic matrix in a typical DC over one week. Note that as in [27], we consider the aggregated traffic here without distinguishing high-priority traffic and low-priority traffic. This matrix shows the similar distribution to that in Facebook's DCN [27] that traffic is densely distributed among a few heavy clusters. About 80% of traffic interactions are owed to the

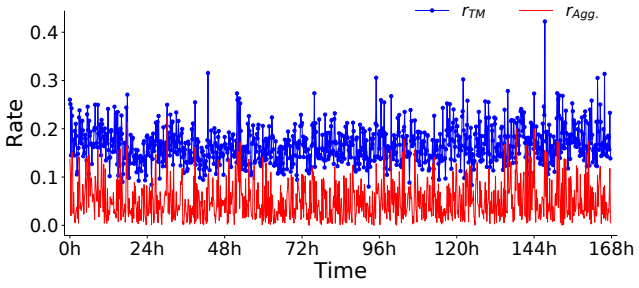
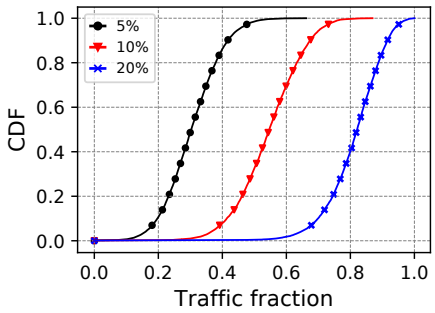
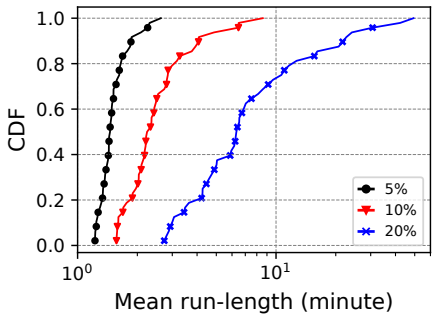


Figure 9: Change rates of the aggregated traffic and the traffic matrix of the heavy cluster pairs in a week; the labels on the x-axis mark the hours of the week; each data point corresponds to a 10-minute interval.



(a) Distribution of the fraction of total inter-Cluster traffic with 3 different change rate upper bounds



(b) Distribution of run-length of insignificant change

Figure 10: Inter-cluster traffic predictability.

top 50% of cluster pairs, and the set of heavy cluster pairs remains not changed over time. A further look at the racks reveals that 80% of inter-Cluster traffic is from communications among less than 17% of rack pairs; this implies the viable of heterogeneous fabrics in DCs.

We further plot the $r_{Agg.}$ and the r_{TM} of inter-Cluster traffic in Figure 9. It shows that the aggregated traffic remains relatively stable with the median change rate of 4.2%. However, the fluctuation of traffic exchanges is much greater: the median change rate is about

16.3%, implying dynamic traffic exchanges among different cluster pairs. A possible reason is that the interconnect within a DC is often abundant, and traffic within a DC is not well scheduled (*i.e.*, more dynamic).

Predictability analysis. Finally, we examine the predictability of inter-cluster (but intra-DC) traffic. Figure 10(a) plots the distribution of the fraction of total traffic contributed by those cluster pairs that have no significant change in traffic on a 1-minute time scale. Again, we use three different change rate thresholds (5%, 10% and 20%) to identify those pairs experience no significant traffic change. While for 80% of 1-minute intervals, about 45% of the total traffic is contributed by the cluster pairs with change rate less than 10% ($thr = 10\%$), less than 10% of the cluster pairs remain predictable for over 5 minutes with this moderate change rate threshold (see Figure 10(b)). This observation implies relatively low stability of inter-cluster traffic exchanges.

4.3 Summary and Implications

Our analysis reveals the very biased WAN (high-priority) traffic distribution towards a small portion (8.5%) of DC pairs. Overall, the *aggregated* high-priority inter-DC (WAN) traffic and the traffic exchange patterns among DCs remain stable over time, resulting in a good predictability of overall traffic demands. The inter-cluster (all) traffic also shows an imbalanced distribution, especially at the rack-level communication. We also observe a relatively low stability of inter-cluster traffic.

The biased distribution of WAN traffic suggests a unbalanced resource allocation, with more resource to the stable heavy hitters. The good stability of aggregated WAN traffic between DC pairs implies to allocate bandwidth based on the historical traffic volume as proposed in [14, 16]. Nevertheless, providers like Baidu that host hundreds of services, prefer to allocate WAN bandwidth allocation at service level [22]; we will return to study the service-level traffic prediction in next section. The observation on the imbalanced inter-cluster traffic suggests a heterogeneous fabric design. Besides, network fabric needs to take special consideration to cope with the inter-cluster traffic dynamics. For instance, as in [11], to cope with the dynamics of traffic load, network fabrics need to incorporate randomness in forwarding path selection for individual flows.

5 WAN TRAFFIC CHARACTERISTICS OF SERVICES

This section examines traffic at the service level, motivated by the disparity of traffic patterns that we have observed in the previous analysis. In particular, we study the interaction patterns and temporal correlations among services from the perspective of traffic volume in WAN. We observe different interactions patterns and high temporal correlations among services. We next investigate the predictability of high-priority inter-DC traffic for different services. Finally, given the differences in terms of traffic predictability across services, we examine the WAN traffic prediction accuracy at service level using the widely used methods in SD-WAN solutions.

5.1 Service Interaction

Interaction pattern. We first examine the service interaction pattern across DCs from the perspective of traffic volume. We found that while many services interact with each other, in aggregation, traffic distribution over services in WAN is highly skewed: 16% of services generate 99% of WAN traffic. The interaction matrix of individual services is also very sparse: as few as 0.2% of service pairs account for over 80% of traffic, and 20% of traffic comes from the interaction of services with themselves.

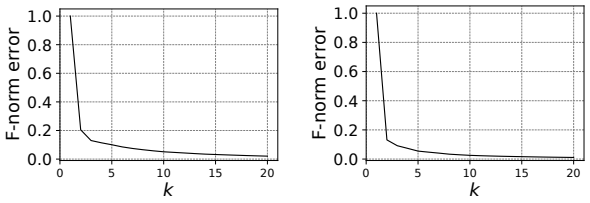
For each category of services, we further inspect the traffic shares among its interacted services and present the results in Table 3. For the traffic generated from each type of source services, we compute its distribution over all types of destination services (*i.e.* each row in Table 3). For instance, 28% of the traffic that is generated by Web services is destined to Computing services. Three observations are notable. First, Web, DB and Cloud services are the most extensive self-interactions; they either sync data (*e.g.* web search indexes) among DCs [35] (thus generating low-priority traffic), or use geo-distributed replicas to collaboratively serve Internet users (thus generating high-priority traffic). Other types of services are less likely to be called by the services of their own types, which is particular true for FileSystem services. Second, Web and Computing services have considerable interactions with other services; this observation is evidenced by the considerable fractions of traffic from other types of services towards them. This is possibly because that Web is the dominant application in this DCN and Computing services are the foundation for other services. Third, the two newly types of services, Analytics and AI, also interact with all other types of services fairly frequently—they are indeed becoming new foundations for other services.

We also studied the service interaction pattern for high-priority traffic across DCs as shown in Table 4. Recall high-priority traffic serves Internet-facing requests, and thus is delay sensitive. Compared with the total traffic shares, we observe much more extensive self-interactions for the high-priority traffic of Web, DB and Cloud services. Besides, because Web services are less likely to use Computing services to fulfill requests from Internet users, the traffic share of Computing services to Web services drop greatly compared to that for aggregated traffic (the ratio declines from 40.3% to 16.6%). Analytics services, on the other hand, are more likely to communicate with Computing services when serving Internet-facing requests (the ratio increases from 15.5% for all traffic to 33.9% for high-priority traffic). At last, overall, the Web, Computing, Analytics and AI services still have considerable interactions with other types of services to respond to Internet-facing requests.

Low rank of service traffic matrix. Extensive traffic interactions of diverse services among DCs may lead to distinct traffic characteristics of services in WAN networks. Although the overall inter-DC traffic has diurnal patterns, specific services may hold different characteristics. Understanding the temporal correlations between different services is thus important for developing better strategies for service deployment and traffic scheduling. To this end, we form the temporal traffic matrix for individual services $M = [\boldsymbol{m}_1, \boldsymbol{m}_2, \ldots, \boldsymbol{m}_n]$, where $\boldsymbol{m}_i \in \mathbb{R}^l$ is the traffic volume of the i-th service in 10-minute intervals in a day ($l = 144$), *i.e.*, the j-th

Table 3: Service interaction among DCs (over WAN) from the perspective of *aggregated* traffic (including both high-priority and low-priority traffic): traffic volume interacted among different types of services; normalized by the total traffic from source services.

% Dst service / Src service	Web	Computing	Analytics	DB	Cloud	AI	FileSystem	Map	Security	Total
Web	51.7	28	9.3	2.5	1.3	4.1	2.3	0.5	0.4	100
Computing	40.3	32.9	15.5	2.6	1	5	1.1	1	0.7	100
Analytics	15.5	44.4	24	1.8	2.3	8.9	1.3	1	0.8	100
DB	18.7	12.7	5.3	47.6	7	4.5	0.5	3.3	0.4	100
Cloud	16.7	9.6	7.8	1.9	59.9	2.8	0.7	0.5	0.2	100
AI	16.1	23.6	29.8	4.7	2	18.6	2.1	2.8	0.2	100
FileSystem	43.4	29.9	11.2	0.9	1.7	9.3	1.6	1.6	0.5	100
Map	6.2	34.3	13.5	4.6	1.5	12	3.3	24.1	0.4	100
Security	18.5	26.9	12.1	5.4	2.4	5.6	1.7	1.1	26.4	100



(a) All (b) High-priority

Figure 11: Low rank for the temporal traffic matrix among services with the relative F-norm error of rank-k approximation. (a) All traffic; (b) High-priority traffic.

value of $\boldsymbol{m}_i$ is the traffic volume of the i-th service in the j-th 10-minute interval. Given the skewed WAN traffic distribution among services, we consider only the top 20% of services (*i.e.*, the top 144 services, $n = 144$). That said, M is a 144×144 matrix.

We analyze the temporal correlation among services by applying the Singular Value Decomposition (SVD) on the matrix. The goal of SVD is to find the effective rank-k approximation to M. If $k \ll min(n, l)$, then we say M has low rank. Low rank is important because it means that elements of M are related; only a small amount of information is needed to construct M, so some elements of M can be computed as linear combination of other elements. Likewise, if a matrix has low rank, then some elements can be approximated as linear functions of other elements [12]. To find k, we need to use the error distance of reconstructing the columns of M using these k vectors. The error distance is defined as the Frobenius norm of $M - M^{(k)}$, denoted as $\|M - M^{(k)}\|_F = \sqrt{\sum_{i=k+1}^{r} \sigma_i^2}$, where σ_i is the diagonal value in the Singular value matrix $\Sigma_{r \times r}$.

Figure 11 shows the variation of the relative F-norm error when varying k. We can see that for both the total and the high-priority traffic, using the top 6 strongest features to describe the whole temporal traffic patterns can reach less than 5% relative error. In other words, the matrix has a low rank of 6. With such a low rank, we can measure a few elements in M to infer other elements. The low rank also implies a limited number of WAN traffic variation patterns across services.

5.2 Service-level Traffic Predictability

Traffic stability analysis. In Section 4, we study the predictability of overall traffic exchanged between heavy DCs. When running various commercial services, providers (like Baidu) may apply service-level traffic engineering to offer different levels of service-level objectives, as suggested in [7]. To this end, we need to examine the predictability of high-priority inter-DC traffic for different services. Figure 12(a) plots the distribution of the fraction of total traffic contributed by the DC-pairs experiencing less than 10% of traffic change on a 1-minute time scale. The stability indeed varies greatly across services. First, the Web, Cloud and DB services exhibit a very good stability for most fraction of inter-DC traffic: for over 80% of 1-minute intervals, 90% of traffic remains stable. Due to the large fraction of self-interactive traffic of these three services (see Table 3), the traffic stability is mainly determined by their inherent traffic characteristics, leading to a higher stability. In contrast, the Computing service exhibits less stable with under 60% of traffic remaining stable for over 80% of 1-minute intervals. As Computing has wide range of interactions with many other services (see Table 3), its stability is affected by the diverse usage of other services. Second, we see AI and Analytics, which constitute the new foundations for other services, exhibit less predictability than those of traditional Web services, urging the need of fine-grained WAN traffic engineering. At last, Map and Security services are least stable, possibly because of their unpredictable usage patterns and the relatively low traffic volume.

Figure 12(b) further shows the run-length of stability. The Web services has the longest run-length: as many as 70% of the DC pairs remain predictable for over 5 minutes. Given that the web search services are still the major services of Baidu and contribute the most of traffic, the good stability of their high-priority traffic means a high accuracy of traffic prediction and better performance guarantee. The run-length for the FileSystem and Map services is much shorter: only about 20% of the DC pairs remain predictable for over 5 minutes. Given that the high-priority traffic of Map services is more likely to cross DC boundary (see Table 2), special cares should be taken when traffic engineering for this type of services. We also observe that although the high-priority traffic of Cloud services is quite stable for the most fraction of inter-DC traffic (see Figure 12(a)), its stability cannot persist for a long time.

Table 4: Service interaction among DCs (over WAN) from the perspective of *high-priority* traffic: high-priority traffic volume interacted among different types of services; normalized by the total traffic from source services.

% Dst service / Src service	Web	Computing	Analytics	DB	Cloud	AI	FileSystem	Map	Security	Total
Web	71.3	9.5	8.4	3.9	1.4	2.9	2.5	0.2	0.1	100
Computing	16.6	33.8	33.9	3.6	3.2	6.4	0.4	2	0.1	100
Analytics	18.3	29.1	32.6	2.8	4.2	10.5	1.3	1.2	0.1	100
DB	13.8	5.3	4.8	60.8	6.5	4.5	0.2	3.7	0.4	100
Cloud	6.9	7.7	11.6	2.3	67.9	2.4	0.4	0.6	0.1	100
AI	13	16.8	35.4	5.8	2.5	22	1.7	2.8	0.1	100
FileSystem	63	8.3	12.3	0.8	1.7	12	0.4	1.4	0.1	100
Map	3.7	36	13.2	5.5	1.9	10.9	1.9	26.6	0.4	100
Security	12.2	8.2	5.9	19.8	7.3	5	1.1	3.8	36.7	100

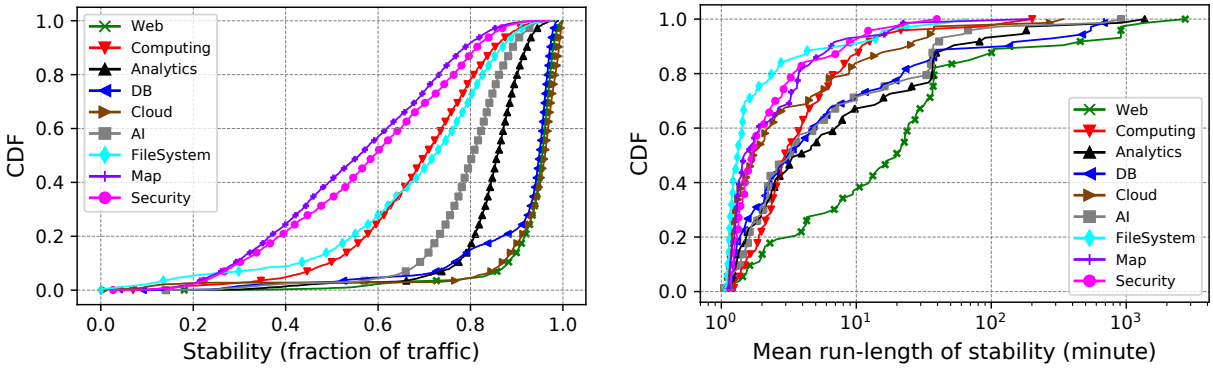


Figure 12: High-priority traffic predictability across services. (a) Distribution of the fraction of total high-priority inter-DC traffic; (b) Distribution of run-length of insignificant change.

we find distinct service interaction patterns and predictability for individual services in terms of high-priority WAN traffic. We next examine how well the traffic of individual services can be estimated using existing methodologies [14, 19].

Service-level traffic prediction. We first depict the high-priority inter-DC (WAN) traffic of each type of services by utilizing the traffic data collected during one week period. Figure 13 shows the normalized traffic volume of each type of service in the first four days of the week on a 1-minute time scale. We can see different diurnal patterns among services. The coefficient of variation varies of these time series from 0.13 (DB) to 0.62 (Cloud), confirming wide diversity of variations for different services.

The distinct temporal variation of different types of services challenges the existing methods used by SD-WAN in accurately estimating the high-priority WAN traffic, where they often rely on the average or median traffic volume in the last few minutes to estimate the interactive service's demand [14, 19]. A popular solution to tolerate prediction error is setting aside different headrooms [22, 32]. A larger prediction error will require larger headrooms, which however will lower the utilization of WAN links and degrade the performance for bulk transfers over WAN.

To answer this question, we select three simple yet widely-used time series models to examine the traffic demand prediction accuracy, namely Historical Average, Historical Median, and SES (Simple Exponential Smoothing). Historical Average/Median calculates the average/median of the historical data as a prediction; SES (Simple Exponential Smoothing) calculates the weighted average of the historical data, where the weights decrease exponentially as observations become older. Specially, with SES, the traffic demand at time $t+1$ is estimated as $\widehat{y}_{t+1|t} = \alpha \sum_{i=0}^{t-1}(1-\alpha)^i * y_{t-i}$, where α ($0 \le \alpha \le 1$) controls the rate at which the weights decrease and is set as 0.2 and 0.8 in our experiments.

Recall that we aim at service-level traffic engineering for high-priority traffic. To that end, we evaluate the above estimation methods using the high-priority WAN traffic data spanning over one week on a 1-minute time scale. That said, we perform a 1-minute-ahead prediction using the historical traffic data within a 5-minute window. For each type of services, the prediction is applied on the inter-DC WAN links that carry large amounts of traffic of that type of services. We compute the median prediction error on each link, which is $|\widehat{y}_{t+1} - y_{t+1}|/y_{t+1}$. We finally report the average error along with the standard deviation for each type of services over all links in Figure 14.

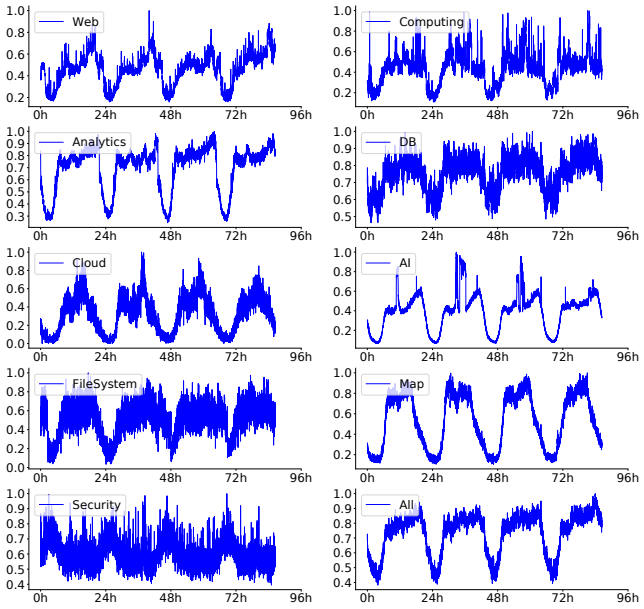


Figure 13: The high-priority WAN traffic of different types of services on a 1-minute time scale; the x-axis shows the hours of the week and the y-axis shows the traffic volume which is normalized by the peak volume.

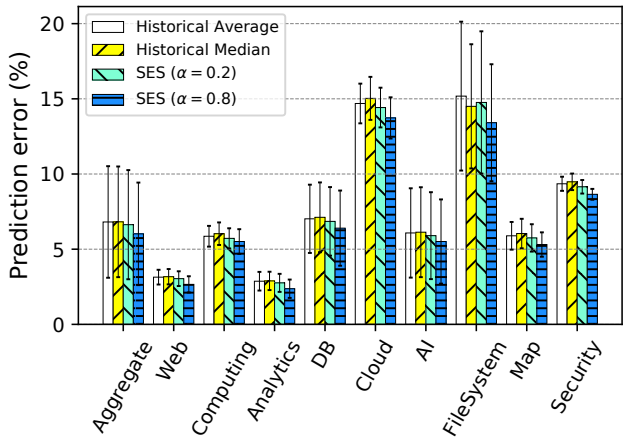


Figure 14: The WAN traffic prediction errors using the methods based on historical statistical information.

We can see from the figure that the prediction accuracy varies greatly across different services. While the models perform well for Web and Analytics services with the prediction error less than 5%, the error of other services such as Cloud and FileSystem reaches nearly 15% as their short run-length of stability (see Figure 12(a)). Besides, for each type of service, the historical average/median model predicts slightly less accurately than the SES models with α close to 1, indicating that the more distant observations have less effect on the prediction due to the non-stationary pattern. These observations, on one hand, reaffirm our analysis on traffic stability. On the other hand, they imply that for some services, a large headroom is required if the above methods are applied, leading to less efficient use of the experience WAN bandwidth. A possible way to improve prediction accuracy is to leverage neural network-based prediction models (*e.g.* LSTM), which can capture more features of time series. Such models may take a longer time to get the estimation, but we think they are viable given that the traffic engineering is often performed on time scales over 1 minutes.

5.3 Summary and Implications

Our analysis reveals different interaction patterns among services. Web and Computing services heavily interact with each other showing that they are closely bound to each other. Analytics, AI, Map, and Security services send their traffic to others more evenly, implying their prevalence. These observations shed light on service migration and deployment, *e.g.* co-locating Web and Computing services in few DCs, replicating Analytics, AI, Map and Security services into each DCs. We also observe a high correlation between traffic time series on a 10-minute time scale across individual services. This observation indicates a limited number of traffic variation patterns, implying one may estimate the traffic trend of a service using the trends of other correlated services.

We further give insight into the service-level traffic predictability, finding the stability varies greatly across services. Indeed, the traffic predictability of services is affected by their interaction patterns. The test of existing methods for the estimation of high-priority traffic demand reveals that while they provide good estimation for the services showing good stability (*e.g.* Web, Analytics services), they perform poor for the services whose stability cannot persist for a long time (*e.g.* Cloud, Filesystem, Security services). Using these estimation methods will either lower the link utilization (using large headroom) or hurt the performance of interactive applications (using small headroom). Leveraging neural network for more accurate estimation indeed needs further investigation.

6 RELATED WORK

Traffic characterization of DC networks. A large body of measurement studies have reported the nature of traffic within DCs, including Microsoft (Web search) datacenter [4, 5, 11, 20], and Facebook (social network) datacenter [27]. The authors in [11, 20] analyzed the traffic characteristics from the perspectives of traffic exchange within DCs and flow characteristics [11, 20]. Benson *et al.* examined the packet arrival patterns at switches and link utilization in different layers [5], and also investigated flow characteristics and rack-level locality in [4]. Roy *et al.* extended the previous studies that are primarily performed in Microsoft DCs [27], and found different traffic characteristics of traffic locality and flow characteristics. These works provide implications for designing novel connection fabrics [11], traffic engineering protocols [6, 13] and advanced switches [29] in DCs. Some other studies [21, 34] inspected the packet burst behavior in DCs, which is useful for device buffer configuration [1, 33] and efficient congestion control [2, 24].

These measurements focus mostly on traffic characteristics inside DCs. In contrast, our work focuses more on WAN traffic; with

this in mind, our findings shed useful light in WAN traffic engineering, service migration and deployment, WAN switch configuration as well. In this perspective, our work is closely related to [8] that studied the inter-DC traffic in Yahoo!, but their scale is much smaller (5 DCs) and the application mix is much simpler.

Traffic engineering for Interconnect WAN. Given that DC-WAN is an expensive resource, more flexible and responsible traffic engineering techniques are increasingly needed for refining bandwidth allocation and congestion control for cross-DC networks [14, 19, 22, 31], and accelerating bulk transferring across DCs [17, 23, 30]. The effectiveness of these traffic engineering approaches depends heavily on the WAN traffic patterns that we examined in this paper. For example, both SWAN [14] and Tempus [19] estimate the aggregated traffic demand with the average traffic demand in the last few minutes. While our analysis reveals the stability for aggregated traffic in large time scale (*e.g.* 10 minutes), we find applying their estimations directly to different service groups may lead to low accuracy for service-level traffic allocation [22].

7 CONCLUSION

This paper has examined the WAN traffic characteristics in Baidu's DCN. We find a considerable fraction of traffic demands over DC-WAN, which leads to higher utilization for links carrying WAN traffic and great challenges for WAN traffic engineering. We further observe that traffic communications among DCs are extensive but highly skewed toward a small number of persistently heavy DC pairs. Besides, the aggregated traffic and traffic exchanges among these heavy DCs remain stable over time; the stability provides the possibility for predicting overall traffic demands in the WAN. However, the stability and traffic characteristics vary across different services on a 1-minute time scale, leading to different prediction accuracy of high-priority traffic for different types of services. The large prediction error for some types of services using existing methods motivates our further investigation of better prediction methods for fine-grained traffic engineering at the service level.

Our focus in this paper is on the traffic pattern of Baidu's in-house services. That said, the traffic generated by the cloud customer services that run on Baidu's public cloud is not included in this paper. The cloud customer services may show different behavior than Baidu's in-house services. For instance, most cloud customer services are not as popular as the in-house services (*e.g.*, Web search), as such they may use fewer DCs, leading to lower DC-WAN traffic than the in-house services; they may also much less rely on other services than the in-house services, leading to less correlation between services in terms of traffic dynamics. We leave the further investigation of cloud customer traffic as our future work.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments and helpful suggestions. This work was partially supported by Beijing Natural Science Foundation (JQ20024), Natural Science Foundation of China (U20A20180, 62072437), and CAS-Austria Joint Project (171111KYSB20200001). Corresponding Author: Zhenyu Li.

REFERENCES

[1] Mohammad Alizadeh, Tom Edsall, Sarang Dharmapurikar, Ramanan Vaidyanathan, Kevin Chu, Andy Fingerhut, Vinh The Lam, Francis Matus, Rong Pan, Navindra Yadav, et al. 2014. CONGA: Distributed congestion-aware load balancing for datacenters. In *Proceedings of the 2014 ACM conference on SIGCOMM*. 503–514.

[2] Mohammad Alizadeh, Albert Greenberg, David A Maltz, Jitendra Padhye, Parveen Patel, Balaji Prabhakar, Sudipta Sengupta, and Murari Sridharan. 2010. Data center tcp (dctcp). In *Proceedings of the ACM SIGCOMM 2010 conference*. 63–74.

[3] Baidu Apollo. 2020. Smart Transportation Solution; Autonomous Driving Solution; Intelligent Vehicle Solution. https://apollo.auto/index.html

[4] Theophilus Benson, Aditya Akella, and David A Maltz. 2010. Network traffic characteristics of data centers in the wild. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*. 267–280.

[5] Theophilus Benson, Ashok Anand, Aditya Akella, and Ming Zhang. 2010. Understanding data center traffic characteristics. *ACM SIGCOMM Computer Communication Review* 40, 1 (2010), 92–99.

[6] Theophilus Benson, Ashok Anand, Aditya Akella, and Ming Zhang. 2011. MicroTE: Fine grained traffic engineering for data centers. In *Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies*. 1–12.

[7] Jeremy Bogle, Nikhil Bhatia, Manya Ghobadi, Ishai Menache, Nikolaj Bjørner, Asaf Valadarsky, and Michael Schapira. 2019. TEAVAR: striking the right utilization-availability balance in WAN traffic engineering. In *Proceedings of the ACM Special Interest Group on Data Communication*. 29–43.

[8] Yingying Chen, Sourabh Jain, Vijay Kumar Adhikari, Zhi-Li Zhang, and Kuai Xu. 2011. A first look at inter-data center traffic characteristics via yahoo! datasets. In *2011 Proceedings IEEE INFOCOM*. IEEE, 1620–1628.

[9] Benoit Claise, Ganesh Sadasivan, Vamsi Valluri, and Martin Djernaes. 2004. Cisco systems netflow services export version 9. (2004).

[10] Apache Doris. 2020. A fast MPP database for all modern analytics on big data. http://doris.apache.org/master/en/

[11] Albert Greenberg, James R Hamilton, Navendu Jain, Srikanth Kandula, Changhoon Kim, Parantap Lahiri, David A Maltz, Parveen Patel, and Sudipta Sengupta. 2009. VL2: a scalable and flexible data center network. In *Proceedings of the ACM SIGCOMM 2009 conference on Data communication*. 51–62.

[12] Gonca Gürsun and Mark Crovella. 2012. On Traffic Matrix Completion in the Internet. In *Proceedings of the 2012 Internet Measurement Conference (IMC '12)*.

[13] Keqiang He, Eric Rozner, Kanak Agarwal, Wes Felter, John Carter, and Aditya Akella. 2015. Presto: Edge-based load balancing for fast datacenter networks. *ACM SIGCOMM Computer Communication Review* 45, 4 (2015), 465–478.

[14] Chi-Yao Hong, Srikanth Kandula, Ratul Mahajan, Ming Zhang, Vijay Gill, Mohan Nanduri, and Roger Wattenhofer. 2013. Achieving high utilization with software-driven WAN. In *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*. 15–26.

[15] Kevin Hsieh, Aaron Harlap, Nandita Vijaykumar, Dimitris Konomis, Gregory R Ganger, Phillip B Gibbons, and Onur Mutlu. 2017. Gaia: Geo-distributed machine learning approaching {LAN} speeds. In *14th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 17)*. 629–647.

[16] Sushant Jain, Alok Kumar, Subhasree Mandal, Joon Ong, Leon Poutievski, Arjun Singh, Subbaiah Venkata, Jim Wanderer, Junlan Zhou, Min Zhu, et al. 2013. B4: Experience with a globally-deployed software defined WAN. *ACM SIGCOMM Computer Communication Review* 43, 4 (2013), 3–14.

[17] Xin Jin, Yiran Li, Da Wei, Siming Li, Jie Gao, Lei Xu, Guangzhi Li, Wei Xu, and Jennifer Rexford. 2016. Optimizing bulk transfers with software-defined optical WAN. In *Proceedings of the 2016 ACM SIGCOMM Conference*. 87–100.

[18] Srikanth Kandula, Ishai Menache, Joseph Seffi Naor, and Erez Timnat. 2019. An Algorithmic Framework for Geo-Distributed Analytics. In *Network Games, Control, and Optimization*. Springer, 89–105.

[19] Srikanth Kandula, Ishai Menache, Roy Schwartz, and Spandana Raj Babbula. 2014. Calendaring for wide area networks. In *Proceedings of the 2014 ACM conference on SIGCOMM*. 515–526.

[20] Srikanth Kandula, Sudipta Sengupta, Albert Greenberg, Parveen Patel, and Ronnie Chaiken. 2009. The nature of data center traffic: measurements & analysis. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement*. 202–208.

[21] Rishi Kapoor, Alex C Snoeren, Geoffrey M Voelker, and George Porter. 2013. Bullet trains: a study of NIC burst behavior at microsecond timescales. In *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*. 133–138.

[22] Alok Kumar, Sushant Jain, Uday Naik, Anand Raghuraman, Nikhil Kasinadhuni, Enrique Cauich Zermeno, C Stephen Gunn, Jing Ai, Björn Carlin, Mihai Amarandei-Stavila, et al. 2015. BwE: Flexible, hierarchical bandwidth allocation for WAN distributed computing. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*. 1–14.

[23] Nikolaos Laoutaris, Michael Sirivianos, Xiaoyuan Yang, and Pablo Rodriguez. 2011. Inter-datacenter bulk transfers with netstitcher. In *Proceedings of the ACM SIGCOMM 2011 conference*. 74–85.

[24] Radhika Mittal, Vinh The Lam, Nandita Dukkipati, Emily Blem, Hassan Wassel, Monia Ghobadi, Amin Vahdat, Yaogong Wang, David Wetherall, and David Zats. 2015. TIMELY: RTT-based Congestion Control for the Datacenter. *ACM SIGCOMM Computer Communication Review* 45, 4 (2015), 537–550.

[25] Heng Pan, Zhenyu Li, JianBo Dong, Zheng Cao, Tao Lan, Di Zhang, Gareth Tyson, and Gaogang Xie. 2020. Dissecting the Communication Latency in Distributed Deep Sparse Learning. In *Proceedings of the ACM Internet Measurement Conference*. 528–534.

[26] Qifan Pu, Ganesh Ananthanarayanan, Peter Bodik, Srikanth Kandula, Aditya Akella, Paramvir Bahl, and Ion Stoica. 2015. Low latency geo-distributed data analytics. *ACM SIGCOMM Computer Communication Review* 45, 4 (2015), 421–434.

[27] Arjun Roy, Hongyi Zeng, Jasmeet Bagga, George Porter, and Alex C Snoeren. 2015. Inside the social network's (datacenter) network. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*. 123–137.

[28] Ahmed Saeed, Varun Gupta, Prateesh Goyal, Milad Sharif, Rong Pan, Mostafa Ammar, Ellen Zegura, Keon Jang, Mohammad Alizadeh, Abdul Kabbani, et al. 2020. Annulus: A Dual Congestion Control Loop for Datacenter and WAN Traffic Aggregates. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*. 735–749.

[29] Arjun Singh, Joon Ong, Amit Agarwal, Glen Anderson, Ashby Armistead, Roy Bannon, Seb Boving, Gaurav Desai, Bob Felderman, Paulie Germano, et al. 2015. Jupiter rising: A decade of clos topologies and centralized control in google's datacenter network. *ACM SIGCOMM computer communication review* 45, 4 (2015), 183–197.

[30] Zhenjie Yang, Yong Cui, Xin Wang, Yadong Liu, Minming Li, Shihan Xiao, and Chuming Li. 2019. Cost-efficient scheduling of bulk transfers in inter-datacenter WANs. *IEEE/ACM Transactions on Networking* 27, 5 (2019), 1973–1986.

[31] Gaoxiong Zeng, Wei Bai, Ge Chen, Kai Chen, Dongsu Han, Yibo Zhu, and Lei Cui. 2019. Congestion Control for Cross-Datacenter Networks. In *2019 IEEE 27th International Conference on Network Protocols (ICNP)*. IEEE, 1–12.

[32] Hong Zhang, Kai Chen, Wei Bai, Dongsu Han, Chen Tian, Hao Wang, Haibing Guan, and Ming Zhang. 2017. Guaranteeing Deadlines for Inter-Data Center Transfers. *IEEE/ACM Transactions on Networking* 25, 1 (2017), 579–595. https://doi.org/10.1109/TNET.2016.2594235

[33] Hong Zhang, Junxue Zhang, Wei Bai, Kai Chen, and Mosharaf Chowdhury. 2017. Resilient datacenter load balancing in the wild. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*. 253–266.

[34] Qiao Zhang, Vincent Liu, Hongyi Zeng, and Arvind Krishnamurthy. 2017. High-resolution measurement of data center microbursts. In *Proceedings of the 2017 Internet Measurement Conference*. 78–85.

[35] Yuchao Zhang, Junchen Jiang, Ke Xu, Xiaohui Nie, Martin J. Reed, Haiyang Wang, Guang Yao, Miao Zhang, and Kai Chen. 2018. BDS: A Centralized near-Optimal Overlay Network for Inter-Datacenter Data Replication *(EuroSys '18)*. Article 10, 14 pages.