Chen L, Li B, Li B. Allocating bandwidth in datacenter networks: A survey. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 29(5): 910–917 Sept. 2014. DOI 10.1007/s11390-014-1478-x

Allocating Bandwidth in Datacenter Networks: A Survey

Li Chen[1] (陈　丽), Baochun Li[1] (李葆春), *Senior Member, IEEE, Member, ACM*
and Bo Li[2] (李　波), *Fellow, IEEE*

[1] *Department of Electrical and Computer Engineering, University of Toronto, Toronto M5S 3G4, Canada*

[2] *Department of Computer Science and Engineering , The Hong Kong University of Science and Technology, Hong Kong China*

E-mail: {lchen, bli}@eecg.toronto.edu; bli@cse.ust.hk

Received February 13, 2014; revised July 21, 2014.

Abstract Datacenters have played an increasingly essential role as the underlying infrastructure in cloud computing. As implied by the essence of cloud computing, resources in these datacenters are shared by multiple competing entities, which can be either tenants that rent virtual machines (VMs) in a public cloud such as Amazon EC2, or applications that embrace data parallel frameworks like MapReduce in a private cloud maintained by Google. It has been generally observed that with traditional transport-layer protocols allocating link bandwidth in datacenters, network traffic from competing applications interferes with each other, resulting in a severe lack of predictability and fairness of application performance. Such a critical issue has drawn a substantial amount of recent research attention on bandwidth allocation in datacenter networks, with a number of new mechanisms proposed to efficiently and fairly share a datacenter network among competing entities. In this article, we present an extensive survey of existing bandwidth allocation mechanisms in the literature, covering the scenarios of both public and private clouds. We thoroughly investigate their underlying design principles, evaluate the trade-off involved in their design choices and summarize them in a unified design space, with the hope of conveying some meaningful insights for better designs in the future.

Keywords datacenter network, bandwidth allocation, fairness

1 Introduction

Large-scale datacenters have become the *de facto* standard computing platform for cloud applications, in the context of both public clouds maintained by Infrastructure as a Service (IaaS) providers — such as Amazon① — and private clouds owned by Web service providers such as Google. The essence of cloud computing is to allow applications of multiple entities to share resources in the underlying datacenters. It has been well understood that the performance of a cloud application heavily depends on its allocated share of resources, including computing resources, such as CPU and memory, to execute computation and network resources, i.e., link bandwidth in datacenter networks, to support communication between pairs of virtual machines (VMs) or tasks.

As allocations of computing resources have been extensively studied and effective mechanisms have been widely adopted, the computing behavior of an application in cloud computing is quite predictable. However, with respect to network performance, interference and variability are frequently observed, due to the lack of appropriate allocation schemes in the presence of bandwidth competition among flows of multiple entities. With Amazon EC2 as an example, while a tenant could enjoy predictable computing performance by renting a number of VMs with specific types of CPU and memory as desired, its network performance is highly unpredictable, as the communicating flows between its VMs interfere with those from other tenants, as shown in Fig.1. Such uncertainty of network performance results in risks of cost increases and revenue losses for the tenant. As such, link bandwidth allocation in datacenter networks becomes arguably the most critical issue, and it has justifiably attracted a substantial amount of research attention from both academia and industry in the recent five years.

As a starting point, a branch of research efforts[1-5] attempts to provide bandwidth guarantees through sta-

Survey

The research was support in part by the Research Grants Council (RGC) of Hong Kong under Grant No. 615613, the National Natural Science Foundation of China (NSFC)/RGC of Hong Kong under Grant No. N_HKUST610/11, the NSFC under Grant No. U1301253, and the ChinaCache Int. Corp. under Contract No. CCNT12EG01.

①Amazon Elastic Compute Cloud. http://aws.amazon.com, July 2014.

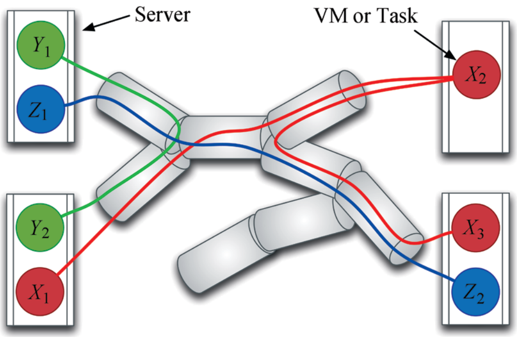

Fig.1. Illustration of bandwidth sharing among multiple tenants (X, Y and Z) in datacenter networks.

tic bandwidth reservations for each competing entity according to its demand. These solutions ensure strong protection, in that bandwidth available to a tenant is independent of the network traffic from other tenants. However, when the demand from a tenant decreases, the extra bandwidth reserved for the tenant cannot be further utilized by other tenants, resulting in a waste of bandwidth and low resource utilization. Therefore, despite perfect performance isolation and predictability provided by static reservations, its inherent drawback — low utilization — has largely restricted its applicability to datacenter networks.

Another attempt[6-12], in contrast, is to share bandwidth in a weighted sharing fashion that can fully utilize available bandwidth and improve resource utilization. Since bandwidth guarantees are no longer ensured with best-effort sharing, it becomes most critical to provide relative performance isolation among competing entities through bandwidth allocation that achieves weighted proportional fairness. In particular, in the public cloud, if we assign weights to competing VM-pairs of tenants according to their payments, each VM-pair can obtain a specific portion of bandwidth, regardless of the number of flows each VM instantiates. In this way, no VM-pair is able to obtain more bandwidth than it deserves, thus achieving weighted fairness at the VM-pair level. In a similar vein, by assigning weights to source VMs, weighted fairness at the source level is achieved, irrespective of the misbehaving source VM that attempts to grab more bandwidth by connecting to more VMs.

The aforementioned wisdom on fair bandwidth sharing, however, is not applicable to the context of private cloud, where a datacenter typically hosts a wide variety of computationally intensive applications that embrace data parallel frameworks such as MapReduce[13] and Dryad[14]. Rather than being achieved across competing VM-pairs or tenants according to their payments, in a private cloud, it is pointed out that fairness should be maintained with respect to performance across multiple data parallel applications, defined as performance-centric fairness[15]. The guiding principle is that the reciprocal of the time required to complete the data transfers in their communication stages should be proportional to their weights across competing applications.

In this article, we attempt to offer an in-depth investigation of the design principles, objectives and characteristics of existing mechanisms in the literature, in the context of bandwidth allocation in datacenter networks. By discussing their advantages and weaknesses, and by presenting their similarities and differences across-the-board, we bring forward an insightful overview of bandwidth allocation in datacenter networks, with the hope of inspiring better designs in the future.

2 Why Do We Care — Importance of Bandwidth Allocation

In cloud computing, resource allocation is critical to both IaaS providers, who are in charge of datacenters, and tenants, who rent resources from these datacenters to deploy their applications. For an IaaS provider, making efficient use of datacenter resources by multiplexing is significant to its revenue, as a higher resource utilization indicates a larger number of tenant requests being handled, and thus more rental income being generated. Meanwhile, for coexisting tenants in the multiplexed datacenter, the amounts of resources they are allocated largely determine their application performance, which further impact their costs.

Such a critical research issue in cloud datacenters has been extensively studied, with practical mechanisms proposed and widely adopted in the industry to allocate computing resources, achieving computing performance isolation among tenant VMs or tasks on each server. However, with respect to the allocation of network resources, it is still an open and active area of research, which has drawn significant attention since 2010.

Unlike CPU and memory that are locally shared at each server, network resource is globally shared among an arbitrary set of tenants that share any link along their network paths. In other words, the allocated share of bandwidth for each VM (or task) is dependent upon the VMs (or tasks) that are co-located with it on the same server, the VMs (or tasks) that are communicating with it, and the VMs (or tasks) that pass through the same bottleneck link with it. This significantly differentiates the allocation of bandwidth from that of computing resources in datacenter networks, incurring more difficulties and bringing new challenges.

As a typical example of the public cloud, Amazon EC2 provides a simple interface for each tenant to specify the number and the type of VMs. This enables tenants to request computing resources on demand, and to enjoy predictable computing performance accordingly. Yet, the network performance is beyond the predictability of a tenant, as EC2 does not make any promise about the amount of bandwidth allocated to each tenant. Interference among tenants' traffic frequently occurs, leading to variations in network performance, as exemplified by the fact that throughput experienced by medium instances in EC2 varies by 66%[16-17].

To encourage enterprises to deploy their applications in the cloud, performance variations should be avoided and performance isolation should be ensured. First, with the existence of performance variations, a range of applications that rely on predictable performance cannot be satisfied, such as high performance computing (HPC) and scientific computing applications in the cloud[18]. Second, for each tenant, the total charge incurred depends on the occupation time of its VMs. Variations in the network performance make it difficult for the tenant to predict and bound its maximal cost. Third, without performance isolation, a selfish and malicious tenant may increase its network share to the detriment of others. The root of this problem lies in the best-effort sharing at the flow level by the Transmission Control Protocol (TCP), the traditional and default standard that is prevalently used in datacenter networks.

To better illustrate the problem inherent with TCP, let us investigate the example shown in Fig.2. In the current public cloud, a tenant is offered much freedom to set the number of flows between its VM-pairs, the responsive behavior to network congestion and so on. In a regular case, each of tenants X and Y has a TCP flow between their respective VM-pair, sharing a bottleneck link. Both flows can obtain an equal share of the link bandwidth, as congestion control in TCP ensures flow-level fairness. Now, suppose the selfish tenant Y establishes three flows between its VM-pair. Still, TCP achieves fairness across the four competing flows at the bottleneck link. In this way, Y grabs a larger share (3/4) by cheating. Such allocation of bandwidth achieved by TCP is unfair to the well-behaved X. Therefore, better schemes of bandwidth allocation are required to ensure performance isolation among tenants, preventing ill-behaved ones from grabbing more bandwidth disproportionately.

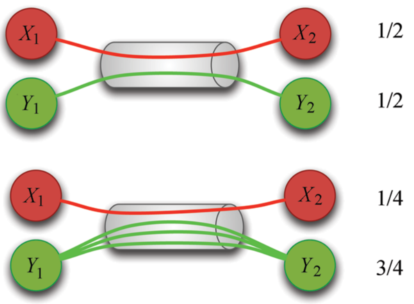

Fig.2. Illustration of unfairness among tenants caused by TCP.

3 What Do We Care — Requirements of Bandwidth Allocation

The significance of bandwidth allocation to tenant costs and the problems arising from the existing poor default allocation call for better designs of bandwidth allocations in datacenter networks, with the following desirable properties.

3.1 Guarantees

Deterministic bandwidth guarantees can ensure predictable performance for each tenant, irrespective of any behavior of competing tenants. Such a property ensures perfect performance isolation across all the tenants, effectively protecting well-behaved tenants while restraining ill-behaved ones. A relaxed version is the *minimum bandwidth guarantees*, which allow tenants to obtain more bandwidth beyond their guarantees when there is available bandwidth. This enables tenants to achieve bounded worst-case performance and thus bounded costs.

3.2 Fairness

Fairness, as a complement to or a substitute for guarantees in guiding bandwidth allocations, is able to provide relative protection to the well-behaved tenants with the presence of ill behavior.

A typical type of fairness is defined according to payment proportionality, which indicates that bandwidth should be divided among tenants in proportion to their payments. To put it simply, tenants with equal numbers of identical VMs, and thus with equal payments, should obtain the same amount of bandwidth. This type of fairness is perfectly suitable for public clouds.

When it comes to the context of private cloud, where communicating tasks of different data parallel applications share the datacenter networks, the aforementioned fairness is no longer applicable. Instead, a better definition of fairness customized for a private cloud should be based upon the final performance achieved by each application, rather than the sheer amounts of bandwidth obtained by individual tenant VMs as in a public cloud.

3.3 Utilization

Intuitively, high utilization indicates that bandwidth should not be left idle as long as there are unsatisfied demands, usually referred to as the *work-conservation* property. With this property, an application with heavy network demands is allowed to use the entire available bandwidth when other applications are inactive. This helps to improve the performance of tenants whose applications have bursty traffic. An IaaS provider also benefits from a high utilization of bandwidth, with the ability to accommodate more tenant requests and thus generate more income.

An implicit indication of high utilization in the context of public cloud is *utilization incentives*. It has been observed that some tenants may be contrived with selfish schemes to improve their performance to the detriment of others, which also lowers the overall utilization of bandwidth[8]. Hence, providers should employ mechanisms with utilization incentives, to discourage misbehavior and encourage tenants to utilize the available bandwidth better.

3.4 Practicality

Practicality is always an important concern for system design and implementation. In particular, with respect to bandwidth allocations in cloud datacenters, practicality has two implications: *simplicity* and *scalability*. First, an intuitive and simple interface is desired for the convenience of tenants to specify their bandwidth demands. This relies on a proper abstraction of bandwidth allocation, which hides complexities to tenants and allows feasible implementations for providers as well. Second, scalability is a necessity for a qualified design of bandwidth allocation in a large-scale cloud datacenter with thousands of servers and switches, where thousands of tenants are hosted, each with tens to thousands of VMs or tasks. Moreover, rapid changes of traffic demands and frequent arrivals of new flows result in over ten million flows per second[19], which makes the requirement of scalability even more important.

4 What Has Been Achieved — Walk-Through and Comparison of Existing Solutions

Bandwidth allocations among multiple tenants in datacenter networks have become a hot topic since 2010, with quite a few solutions proposed in the literature. In this section, we will walk through these solutions, highlight important issues and make comprehensive comparisons, to present a systematic overview that is helpful for readers to gain a better understanding of this topic.

4.1 Deterministic Guarantees – With Static Reservations

4.1.1 Merits and Disadvantages of Static Reservations

The most significant merit of static reservations is the assurance of deterministic bandwidth and predictable performance for each tenant, irrespective of other tenants' behavior. Mechanisms of this type[1-5] provide tenants with an interface to explicitly specify their bandwidth demands. In accordance with its demand, each tenant is supposed to have a virtual network with guaranteed link bandwidth.

Such guarantees are achieved by means of proper placement of tenant VMs and static enforcement of rate limits. In particular, admission control is implemented upon each tenant request, ensuring sufficient bandwidth reservations for all the existing tenants. Meanwhile, VM placement is executed for each admitted tenant, mapping its virtual network to the real datacenter network topology. For the profit maximization of the provider, the optimization objective of VM placement is to allow a maximum number of concurrent tenants in the system. In an online fashion, this is equivalent to minimizing the aggregated bandwidth reservations at the core network, so that future tenant requests can be satisfied. Finally, to enforce bandwidth reservations, rate limiters in hypervisors are applied to constrain the bandwidth of VMs, thus isolating network performance among tenants.

Despite the advantages of deterministic bandwidth guarantees and strong performance isolation, static reservations are not amenable to the high utilization of network bandwidth in datacenters. When a tenant does not fully utilize its bandwidth reservations, the idle bandwidth is unable to be utilized by other tenants in need, thus resulting in a waste of bandwidth and inefficient utilization.

4.1.2 Models of Virtual Network Abstraction

Among representative studies of static reservations, three models are generally used for the virtual network abstraction: the Pipe model[1], the Hose model[2-4] and the Tenant Application Graph (TAG)[5].

As the first attempt of providing bandwidth guarantees, SecondNet[1] provides an interface for tenants to specify end-to-end bandwidth demand for each pair of their VMs. Such a model depicting pairwise bandwidth guarantees is referred to as the Pipe model, as shown in Fig.3. The drawback of the Pipe model is the significant increase of complexity, along with an increasing number of tenant VMs, both to tenants for specifying pairwise bandwidth demand matrices and to the provider for placing tenant VMs with constraints of bandwidth guarantees.

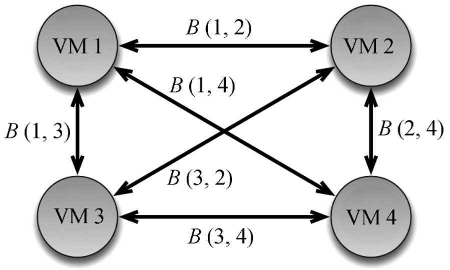

Fig.3. Bandwidth guarantees based on the Pipe model.

In contrast, the basis of a Hose model is that all VMs of a tenant are connected to a central virtual switch by dedicated links (as hoses) with bandwidth guarantees. Based on the Hose model, Oktopus[2] proposes two abstractions — virtual clusters (VC) and virtual oversubscribed clusters (OVC). In correspondence with the Basic Hose model shown in Fig.4, the abstraction of VC requires each tenant to specify its request with a 2-tuple (N, B), where N is the number of VMs and B is the bandwidth guaranteed to each VM. With respect to the OVC abstraction which relies on the hierarchical Hose model shown in Fig.5, a tenant specifies a 4-tuple (N, B, S, O), where N is the number of clusters. At the lower level, each cluster consists of S VMs interconnected by per-VM hoses, with bandwidth guaranteed as B. At the upper level, the clusters are connected through per-cluster hoses each with a bandwidth capacity of $B \times S/O$.

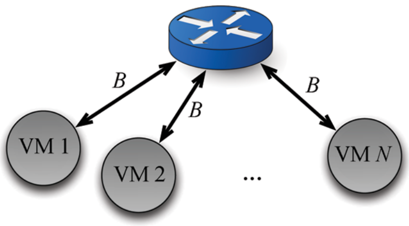

Fig.4. Bandwidth guarantees based on the Basic Hose model.

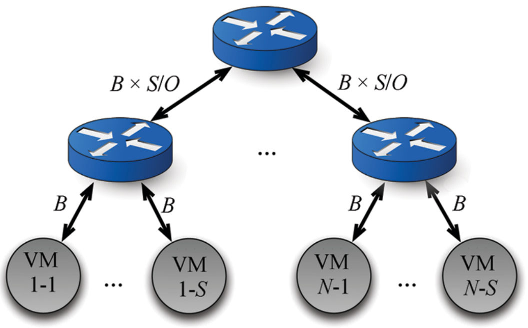

Fig.5. Bandwidth guarantees based on the hierarchical Hose model.

As a follow-up to Oktopus, Zhu *et al.*[4] extended the basic Hose model by considering heterogeneous bandwidth demands of each tenant. They focused on designing an effective VM placement algorithm to address the challenges and complexities resulted from demand heterogeneity. With the observation that data-intensive applications typically exhibit predictable time-varying bandwidth demands, Xie *et al.*[3] pointed out the inefficient nature of previous reservation models that provide constant bandwidth guarantees for a tenant throughout the entire execution of its job. To tackle this, they proposed the abstraction of time-interleaved virtual cluster (TIVC), which extends the basic Hose model to incorporate the time-varying nature of bandwidth demands. Employed in their system — Proteus, the TIVC abstraction allows tenants to multiplex reserved links in the manner of time-sharing, thus improving bandwidth utilization.

The third type of the virtual network abstraction is the Tenant Application Graph (TAG) proposed in CloudMirror[5]. With the TAG, bandwidth demands of a tenant are specified by a graph, which characterizes the structure and the communication pattern of the tenant's application. Specifically, each vertex of the TAG represents an application tier, consisting of a set of VMs with the same functionality, while each directed edge specifies the per-VM bandwidth guarantees for the traffic between a pair of tiers. As shown in Fig.6, the directed edge between C_1 and C_2 is labeled with (B_1, B_2), representing the amount of bandwidth guaranteed for each VM of C_1 to send traffic to C_2, and that for each VM of C_2 to receive traffic from C_1, respectively.

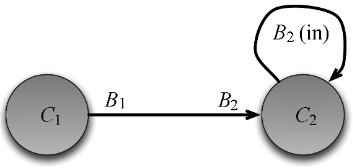

Fig.6. Bandwidth guarantees based on the Tenant Application Graph (TAG).

An equivalent representation of the TAG is shown in Fig.7 for a better understanding. The directed edge between C_1 and C_2 in the original TAG is extended as the directed hoses, connecting all VMs of C_1 and C_2 to a virtual trunk, with the capacities of B_1 and B_2 respectively. In this way, the inter-tier bandwidth guarantees are equivalently represented by a directional Hose model. In a similar vein, the self-edge in the original TAG, which represents the intra-tier bandwidth guarantees, can be transformed to a Hose model for all the VMs of the tier. Due to the awareness of application semantics, the TAG is able to accommodate more tenants and improve the utilization of bandwidth.

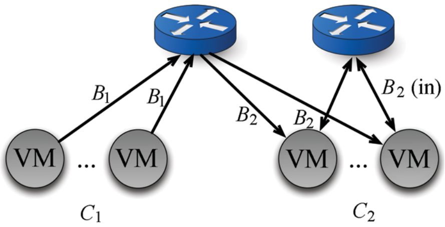

Fig.7. Equivalent representation of the TAG.

4.2 Minimum Guarantees

Minimum guarantees allow tenants to utilize the spare bandwidth beyond their guarantees, which are more efficient compared with deterministic guarantees.

Sharing the Edge with Congestion Control. A branch of solutions achieves minimum guarantees based on the assumption of a congestion-free network core, which is fairly reasonable given the following facts: first, admission control and VM placement can ensure sufficient network capacities for all the tenants; second, the bisection bandwidth in datacenter networks has been significantly improved by multi-path routing (i.e., [20]) and multi-tree topologies (i.e., [21]). With congestion removed from the core, the contention for bandwidth is pushed to the edge. Hence, bandwidth allocations are completely managed at the edge, with the simple abstraction of a giant switch connecting all the competing VMs, each associated with the bandwidth guaranteed into and out of the switch.

As representative solutions, Gatekeeper[9] and EyeQ[10] employ similar mechanisms based on end-to-end congestion control, to guarantee minimum bandwidth while achieving work-conservation in a simple and scalable manner at end servers. Guo *et al.*[22-23] designed a work-conserving allocation strategy based on game theory, with the main focus on balancing the trade-off between minimum bandwidth guarantees and network proportionality.

Sharing the Overall Network with Congestion Control. ElasticSwitch[11] and Hadrian[12] both leverage a combination of admission control, VM placement and congestion control, to achieve minimum bandwidth guarantees and work-conservation. Without the assumption of a congestion-free core, their congestion control mechanisms are able to make full utilization of any bottleneck link in the network.

The highlight of ElasticSwitch is a two-layer scheme of distributed bandwidth allocation, which is merely implemented in each hypervisor, without any dependence on network topologies or switch features. The higher layer divides guarantees for VMs into guarantees for VM-pairs, and the lower layer employs a TCP-like mechanism, which utilizes the spare bandwidth by dynamically increasing bandwidth of VM-pairs beyond their guarantees.

Hadrian[12] studies cloud network sharing among both the intra-tenant and the inter-tenant traffic. The hierarchical Hose model is used to characterize the intra-tenant and inter-tenant bandwidth guarantees for each VM, as well as the communication dependencies among tenants. Moreover, the Hose-Compliant Allocation is proposed to ensure that the maximum bandwidth allocated to a tenant is proportional to its payment. To enforce bandwidth allocations, a weighted congestion control mechanism is implemented over hypervisor-to-hypervisor tunnels, which requires support from switches, unable to scale efficiently.

Sharing the Overall Network with WFQ. The PS-P allocation scheme proposed in FairCloud[8] is based on Weighted Fair Queuing (WFQ), with a properly designed weight assignment. This scheme provides minimum guarantees only when the network topology is tree-based. Moreover, it is not practical, as it requires switches to maintain per-VM queues.

4.3 No Guarantees — With Best-Effort Sharing

Without explicit demands specified by tenants, bandwidth can be shared in a best-effort fashion across flows, VMs, VM-pairs, or tenants, which achieves work-conservation. However, the network performance of competing tenants is no longer strongly isolated or deterministically guaranteed. Instead, fairness plays a significant role in providing relative isolation among competing tenants.

The default way to share bandwidth in current datacenter networks is the end host mechanisms such as congestion control in TCP. Despite its merit of scalability, TCP does not function effectively in cloud network sharing, because its inherent nature of per-flow fairness would enable misbehaving tenants to increase their bandwidth allocations to the detriment of others, by initiating more TCP connections.

To accommodate the requirements of fair sharing and performance isolation among tenants, flow-level fairness is extended to the level of source VMs, VM-pairs, tenants, etc. with existing solutions[6-8], which assign weights to the corresponding competing entities according to their payments.

In particular, NetShare[6] assigns weights to tenants and allocates bandwidth among them based on weighted proportionality, which ensures per-tenant fairness at congested links. In Seawall[7], with each source VM assigned a network weight, a hypervisor-based VM-level congestion control is applied to achieve per-source fairness, i.e., the bandwidth achieved by a source VM at any link, which is the aggregated bandwidth of its outgoing flows passing through the link, is proportional to its weight.

The PS-L proposed in FairCloud[8] allocates bandwidth among tenants at each link using per-tenant WFQ. This ensures fairness at the tenant level and the link level, in that the total bandwidth achieved by a tenant at each link, which is the aggregated bandwidth of all its flows passing through this link, is proportional to its weight. FairCloud[8] also proposes the PS-N, which applies WFQ among source-destination VM-pairs, with their weights properly assigned to achieve fairness at the VM-pair level and the network level.

All the aforementioned efforts center around payment proportional fairness, which is suitable for a public cloud but not applicable to a private cloud, where the datacenter is shared among multiple data parallel applications. To fill this gap, Chen *et al.*[15] proposed a definition of performance-centric fairness, with the intuition that the performance achieved by each application should be proportional to its weight, which is specified according to its importance. Having investigated the trade-off between performance-centric fairness and bandwidth utilization, they designed a distributed bandwidth allocation algorithm that can manipulate and improve the bandwidth utilization with a tunable degree of relaxation on fairness.

To summarize, the weight-based best effort sharing schemes are efficient in utilizing bandwidth. However, none of them provides bandwidth guarantees, as the share of an entity can be arbitrarily reduced with the number of competing entities increasing. NetShare[6] and FairCloud[8] are not practical to be deployed in a large scale, since they rely on per-VM queues in switches for rate control.

5 Conclusions

The rise of research interests in cloud bandwidth allocations is largely fueled by the performance requirements of cloud applications, as well as the characteristics of the underlying datacenter networks. In this article, we have walked through existing efforts in sharing datacenter networks, with an in-depth investigation on the important issues arisen, including guarantees, fairness, utilization and practicality. We categorized and compared existing mechanisms, which help to present a general overview of this active research area.

With the development of software defined networking (SDN), datacenters of large-scale Internet service providers such as Google are interconnected by the software defined Wide Area Network (WAN). In such a context, the scope of bandwidth allocations is broadened to cover the scenario where applications compete for the bandwidth of inter-datacenter links. In 2013, Google and Microsoft have taken the initiative to propose their schemes of inter-datacenter bandwidth allocations in their respective software defined WANs[24-25], which may inspire a variety of more effective solutions in the near future.

References

[1] Guo C, Lu G, Wang H J, Yang S, Kong C, Sun P, Wu W, Zhang Y. SecondNet: A data center network virtualization architecture with bandwidth guarantees. In *Proc. ACM International Conference on Emerging Networking Experiments and Technologies* (*CoNEXT*), Nov. 2010, p.15.

[2] Ballani H, Costa P, Karagiannis T, Rowstron A. Towards predictable datacenter networks. *ACM SIGCOMM Computer Communication Review*, 2011, 41(4): 242-253.

[3] Xie D, Ding N, Hu Y C, Kompella R. The only constant is change: Incorporating time-varying network reservations in datacenters. *ACM SIGCOMM Computer Communication Review*, 2012, 42(4): 199-210.

[4] Zhu J, Li D, Wu J, Liu H, Zhang Y, Zhang J. Towards bandwidth guarantee in multi-tenancy cloud computing networks. In *Proc. the 20th IEEE International Conference on Network Protocols* (*ICNP*), Oct. 30-Nov. 2, 2012.

[5] Lee J, Lee M, Popa L, Turner Y, Banerjee S, Sharma P, Stephenson B. CloudMirror: Application-aware bandwidth reservations in the cloud. In *Proc. USENIX Workshop on Hot Topics in Cloud Computing*, Jun. 2013, pp.225-238.

[6] Lam T, Radhakrishnan S, Vahdat A, Varghese G. NetShare: Virtualizing data center networks across services. Technical Report, CS2010-0957, Department of Computer Science and Engineering, University of California at San Diego, 2010.

[7] Shieh A, Kandula S, Greenberg A, Kim C, Saha B. Sharing the data center network. In *Proc. USENIX NSDI*, Mar. 2011.

[8] Popa L, Kumar G, Chowdhury M, Krishnamurthy A, Ratnasamy S, Stoica I. FairCloud: Sharing the network in cloud computing. In *Proc. ACM SIGCOMM*, Aug. 2012, pp.187-198.

[9] Rodrigues H, Santos J R, Turner Y, Soares P, Guedes D. Gatekeeper: Supporting bandwidth guarantees for multi-tenant datacenter networks. In *Proc. the 3rd Conference on I/O Virtualization* (*WIOV*), Jun. 2011.

[10] Jeyakumar V, Alizadeh M, Mazieres D, Prabhakar B, Kim C, Greenberg A. EyeQ: Practical network performance isolation at the edge. In *Proc. USENIX NSDI*, Apr. 2013, pp.297-312.

[11] Popa L, Yalagandula P, Banerjee S, Mogul J. ElasticSwitch: Practical work-conserving bandwidth guarantees for cloud computing. In *Proc. ACM SIGCOMM*, Oct. 2013, pp.351-362.

[12] Ballani H, Jang K, Karagiannis T, Kim C, Gunawardena D, O'Shea G. Chatty Tenants and the cloud network sharing problem. In *Proc. USENIX NSDI*, Apr. 2013, pp.171-184.

[13] Dean J, Ghemawat S. MapReduce: Simplified data processing on large clusters. *Communications of the ACM*, 2008, 51(1): 107-113.

[14] Isard M, Budiu M, Yu Y, Birrell A, Fetterly D. Dryad: Distributed data-parallel programs from sequential building blocks. *ACM SIGOPS Operating Systems Review*, 2007, 41(3): 59-72.

[15] Chen L, Li B, Li B. Towards performance-centric fairness in datacenter networks. In *Proc. IEEE INFOCOM*, Apr. 2014, pp.2373-2381.

[16] Li A, Yang X, Kandula S, Zhang M. CloudCmp: Comparing public cloud providers. In *Proc. ACM SIGCOMM Conference on Internet Measurement* (*IMC*), Nov. 2010.

[17] Wang G, Ng T S E. The impact of virtualization on network performance of Amazon EC2 data center. In *Proc. IEEE INFOCOM*, Mar. 2010, pp.1163-1171.

[18] He Q, Zhou S, Kobler B, Duffy D, McGlynn T. Case study for running HPC applications in public clouds. In *Proc. ACM International Symposium on High Performance Distributed Computing*, Jun. 2010, pp.295-401.

[19] Benson T, Akella A, Maltz D A. Network traffic characteristics of data centers in the wild. In *Proc. ACM SIGCOMM Conference on Internet Measurement* (*IMC*), Nov. 2010, pp.267-280.

[20] Raiciu C, Barre S, Pluntke C, Greenhalgh A, Wischik D, Handley M. Improving datacenter performance and robustness with multipath TCP. In *Proc. ACM SIGCOMM*, Aug. 2011, pp.266-277.

[21] Greenberg A, Hamilton J, Jain N *et al.* VL2: A scalable and flexible data center network. In *Proc. ACM SIGCOMM*, Oct. 2009, pp.51-62.

[22] Guo J, Liu F, Zeng D, Lui J CS, Jin H. A cooperative game based allocation for sharing data center networks. In *Proc. IEEE INFOCOM*, Apr. 2013, pp.2139-2147.

[23] Guo J, Liu F, Tang H, Lian Y, Jin H, Lui J CS. Falloc: Fair network bandwidth allocation in IaaS datacenters via a bargaining game approach. In *Proc. IEEE International Conference on Network Protocols* (*ICNP*), Oct. 2013.

[24] Jain S, Kumar A. B4: Experience with a globally-deployed software defined WAN. In *Proc. ACM SIGCOMM*, Oct. 2013, pp.3-14.

[25] Hong C, Kandula S. Achieving high utilization with software-driven WAN. In *Proc. ACM SIGCOMM*, Oct. 2013, pp.15-26.

Li Chen is currently pursuing her Ph.D. degree at the Department of Electrical and Computer Engineering, University of Toronto. She received her B.E. degree from the Department of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, in 2012. Her research interests include cloud computing and datacenter networking.

Baochun Li received his Ph.D. degree from the Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, in 2000. Since then, he has been with the Department of Electrical and Computer Engineering at the University of Toronto, where he is currently a professor. He holds the Bell Canada Endowed Chair in computer engineering since August 2005. His research interests include large-scale distributed systems, cloud computing, peer-to-peer networks, applications of network coding, and wireless networks. He is a member of ACM and a senior member of IEEE.

Bo Li received his B.E. degree in computer science from Tsinghua University, Beijing, and Ph.D. degree in electrical and computer engineering from University of Massachusetts at Amherst. He is a professor in the Department of Computer Science and Engineering, the Hong Kong University of Science and Technology. He holds the Cheung Kong Chair Professor in Shanghai Jiao Tong University. Prior to that, he was with IBM Networking System Division, Research Triangle Park, North Carolina. He was an adjunct researcher at Microsoft Research Asia (MSRA) and was a visiting scientist at Microsoft Advanced Technology Center (ATC). He has been a technical advisor for ChinaCache Corp. (NASDAQ CCIH) since 2007. He is an adjunct professor in Huazhong University of Science and Technology, Wuhan. His recent research interests include: large-scale content distribution in the Internet, peer-to-peer media streaming, the Internet topology, cloud computing, green computing and communications. He is a fellow of IEEE for "contribution to content distributions via the Internet". He received the Joint Research Fund for Overseas Chinese Scholars and Scholars in Hong Kong and Macao from the National Natural Science Foundation of China (NSFC) in 2004. He served as a distinguished lecturer for IEEE Communications Society (2006~2007). He was a co-recipient for three Best Paper Awards from IEEE and the Best System Track Paper in ACM Multimedia (2009).