On Traffic Fairness in Data Center Fabrics

Dallal Belabed, Stefano Secci, Guy Pujolle, Deep Medhi*
Sorbonne Universities, UPMC Univ Paris 06, UMR 7606, LIP6, F-75005, Paris, France. Email: firstname.lastname@upmc.fr
* Univ. of Missouri–Kansas City, 5100 Rockhill Road, Kansas City, MO 64110 USA. Email: dmedhi@umkc.edu

***Abstract*—A present challenge in data center networks (DCN) is to better understand the impact of novel flattened and modular DCN architectures on congestion control protocols, and vice-versa. One of the major concerns in congestion control being the fairness in the offered throughput, the impact of the additional path diversity and forwarding features, brought by the novel DCN architectures and protocols, on the throughput of individual endpoints (servers) and aggregation points (edge switches) is unclear. This paper attempts to answer these open questions. Specifically, how best is the allocation of the competing elastic demand flows and how is this allocation impacted with the increase in capacity? We provide an optimization formulation of the problem based on the proportional fairness principle of TCP. We conducted a series of test scenarios on the fat-tree data center topology by considering load balancing and link capacities for different network cases in order to present our analysis on the results. We observed that the traffic allocation fairness is primarily impacted by the weights associated with the TCP implementation in use.**

***Index Terms*—Data Center Networks, Resource fairness, TCP throughput, Fat-tree topology**

I. Introduction

The emergence of network virtualization solutions, such as Infrastructure as a Service (IaaS), offers several advantages to organizations in terms of both operational and capital expenditures [19]. The transition from physical independent networks to virtual de-localized networks operated in the Cloud can be facilitated if, besides security concerns, connectivity performance is at an acceptable level and shows desirable fairness properties.

With the growth in customer volumes, service differentiation and elastic demands, avoiding bottlenecks is a critical point in Data Center Network (DCN) architectures. With the de-facto dominating trend of deploying services using virtualization servers, a non negligible ratio of the traffic is horizontal traffic between virtualization servers, in support of virtual machine migration and storage synchronisation. The amount of intra-DC horizontal traffic can overcome the access vertical traffic volume [2]. This has eventually favored the emergence of novel DCN architectures that expose additional horizontal capacity between server racks and clusters of racks such as fat-tree [1], and BCube [5].

An open question is: how best is the traffic allocation of the competing elastic demand flows for horizontal traffic between edge servers in data center fabrics, and how is this allocation impacted with the increase in capacity? To address this question, we assume that all traffic uses TCP allowing multipath forwarding. More specifically, we are interested in understanding this impact in equilibrium. It has been shown that several variants of TCP are proportionally fair in the equilibrium, which have been verified through simulation [18], [9]. We, therefore, use a proportional fairness model to understand the allocation for competing demands in data center fabrics. Our study is focused on a fat-tree data center topology, one of the popular data center network architectures, as this allows us to understand the impact between intra-pod and inter-pod traffic among all the horizontal traffic.

The rest of the paper is organized as follows. Section II presents the background of our work. Our optimization model is formulated in Section III and the study results are presented in Section IV. Section V concludes the paper.

II. Background

In this section, we present an overview of the TCP proportional fairness model and of existing multipath forwarding protocols.

A. TCP Proportional Fairness Model

In a network with multiple competing TCP sessions sharing links, several studies [14], [8], [18], [9] have shown that TCP implicitly solves a utility problem in equilibrium. This utility problem is formally described as a maximization of an aggregate utility subject to capacity constraints:

$$\max_{X \geq 0} \sum_{j \in \mathcal{J}} U(X_j) \tag{1}$$

subject to

$$\sum_{j \in \mathcal{J}} \delta_{je} X_j \leq c_e, \quad e = 1, 2, ..., E \tag{2}$$

The above model maximizes the utility function $U(X_j)$ of each session $j \in \mathcal{J}$ where X_j denotes the rate of session j while δ_{je} is the indicator that takes the value 1 if session j uses link e, 0 otherwise.

For the *Proportional Fairness* (PF) [10], [6] allocation that is applicable to TCP, utility $U(x_j)$ is set to $\omega_j \, log \, x_j$, where ω_j is the weight of the session j. Hence, the resource allocation corresponding to this utility function is commonly referred to as *weighted proportionally fair*, or, if all ω_d are equal to one, as *proportionally fair*. Thus, (1) for PF becomes:

$$\max_{X \geq 0} \sum_{j \in J} \omega_j \log X_j \tag{3}$$

B. Multipath forwarding protocols

There are many recent protocols designed in the last decade that include forms of multipath forwarding, also referred to as packet load-balancing techniques. They can act either at the data-link, network, or transport levels.

At the data-link layer, a protocol has been designed specifically for data-center networks, the Transparent Interconnection of a Lot of Links Protocol (TRILL) [17]. It allows a switch and even a virtualization server, acting as virtual bridge, to balance the load over multiple destination TRILL bridges for the same pair of nodes. However, no forms of congestion control are implemented here as the evolution of IP networks is such that this has been left to the transport layer.

At the network layer, Equal Cost MultiPath (ECMP) [16] is adopted in the Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (ISIS) protocols [12]: it allows balancing the load over multiple next hops. ECMP can also be implemented in TRILL. However, this is typically performed in such a way that for a specific TCP flow, only one path is used in order to avoid packet disordering and buffer explosion at TCP endpoints.

At the transport layer, there have been two major propositions. Stream Control Transmission Protocol (SCTP) [15] allows end hosts to use several paths concurrently, when devices are multihomed. The way it has been designed, however, makes SCTP weak against the de-facto pervasive presence of middleboxes in the Internet such as firewalls, performance optimizers, load balancers at lower layers and interfaces. In many cases, SCTP connections cannot be opened or maintained. More recently, the Multipath TCP (MPTCP) [4] has been designed with retrocompatibility and incremental deployability as the first design requirements so that using multiple paths simultaneously is made possible, falling back to standard TCP in case of middlebox blocking. Major attention has also be given to congestion control and fairness. An important requirement is that an MPTCP connection over a given link should not take more resources than legacy TCP connections running on the same link. However, as shown in [7], [11], the congestion control algorithm is a key choice when fairness with respect to other connections needs to be guaranteed as it is a major concern of network operators.

Our study is agnostic about the specific multipath forwarding protocol that could be adopted in the DCN fabric, and the related analysis is conceptually applicable to any configuration including multipath forwarding and congestion control in whatever layer.

III. Problem Formulation

Following [13], we now generalize the basic proportional fairness model for DCN allowing multipath forwarding for elastic demands that use TCP. First, while the actual TCP sessions are between edge servers in a DCN, we can consider the model in terms of elastic demands between a pair of edge switches since all such sessions must enter and exit through edge switches (see Figure 1). Thus, moving away from sessions (identified by j earlier), we identify a demand between a pair of edge switches by d with the elastic demand denoted by X_d. Secondly, due to multipath forwarding, we identify traffic flow along each path p associated with demand d by using x_{dp} (notations are summarized in Table I). Therefore, for a given demand, the sum of traffic amount allocated to the paths is equal to the total elastic demand X_d given by:

TABLE I
Mathematical notations

Indices	
$d = 1, 2, \dots D$	demands associated with pairs of edge switchs
$p = 1, 2, \dots P_d$	candidates paths for demand d
$e = 1, 2, \dots E$	links
Variables	
x_{dp}	amount allocated to path p of demand d
X_d	amount allocated to d
Parameters	
δ_{edp} = 1	if link e belongs to path p of demand d; 0, otherwise
α	a minimum sub-flow ratio allocated to each path p available to a demand d
c_e	capacity of link e
ω_d	weight of demand d (constant)

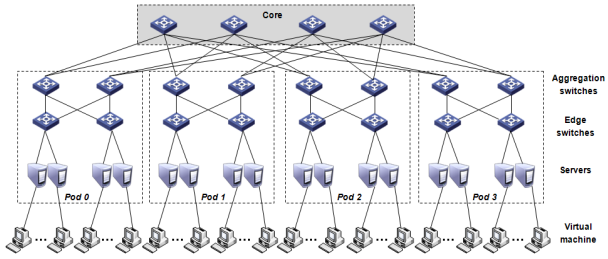


Fig. 1. Fat-tree topology with four pods.

$$\sum_p x_{dp} = X_d \quad d = 1, 2, \dots, D \tag{4}$$

Next, the sum of all the flows using a particular link e must satisfy the link capacity constraint:

$$\sum_d \sum_p \delta_{edp} x_{dp} \leq c_e \quad e = 1, 2, \dots, E \tag{5}$$

The goal is to maximize the utility objective:

$$\max_{X, x \geq 0} \quad F(X) = \sum_d \omega_d \log X_d \tag{6}$$

where ω_d is weight for demand d, which is discussed further in Section III-B.

To summarize, our model is to address the goal given by (6) subject to constraints (4) and (5). It should be noted that while elastic demand X_d is non-negative, the logarithm function ensures that no elastic demand takes the value zero, i.e., every demand must get its share according to proportional fairness subject to capacity constraints and any influence due to ω_d.

In addition to the above model, we are also interested in understanding the impact when we enforce at least some traffic to be carried on each path of a demand d, which can be imposed using the following additional constraint (7):

$$x_{dp} \geq \alpha X_d \quad d = 1, 2, \dots, D \quad p = 1, 2, \dots, P \tag{7}$$

TABLE II
LINEAR APPROXIMATION

Indices	
$k = 1, 2, \ldots, K$	Consecutive pieces of the approximation of $\log x$.
Variables	
f_d	approximation of $\log X_d$.
Parameters	
a_k, b_k	coefficients of the linear pieces of the linear approximation of $\log x$.

Here, each candidate path has to carry at least αX_d, i.e., the minimum of rate allocated to demand d on each candidate path.

A. Linear approximation of the objective

We note that in the previous formulation, the objective function is non-linear due to the logarithm function. We use a linearization approximation [13] of the logarithm function as follows:

$$\log X_d = \min_{k=1,2,\ldots,K} \{a_k X_d + b_k\}. \quad (8)$$

Then, the optimization problem becomes

$$\max_{X,x,f \geq 0} \quad F = \sum_d \omega_d f_d \quad (9)$$

subject to:

$$\sum_p x_{dp} = X_d \quad d = 1, 2, \ldots, D \quad (10)$$

$$\sum_d \sum_p \delta_{edp} x_{dp} \leq c_e \quad e = 1, 2, \ldots, E \quad (11)$$

$$f_d \leq a_k X_d + b_k \quad d = 1, 2, \ldots, D \quad k = 1, 2, \ldots, K \quad (12)$$

The advantage of this approximation is that it is a linear programming problem that can be solved using a well-known software package such as CPLEX.

B. On weights w_d

We now elaborate on w_d taking into consideration two valid TCP implementations [9]. This was a result of different interpretations of TCP Vegas [3]: the one based on bytes per *round trip time* and the other based on bytes per *second* leading to utility functions

$$U(X_d) = \log X_d \quad (13)$$

$$U(X_d) = \bar{\omega}_d \log X_d, \quad (14)$$

respectively. Here $\bar{\omega}_d$ corresponds to the propagation delay of session d. The first situation (13) does not give any weight to the session, and we name it the fixed-delay case. The second situation (14) gives weight to the propagation delay through $\bar{\omega}_d$ for session d and we name it the weighted-delay case. Besides the two valid implementations of TCP Vegas, FAST TCP follows the weighted-delay case [18]. For comparison purposes, we use a simplification for the weighted-delay case by setting $\bar{\omega}_d$ to be based on the number of hops between the source and the destination to serve as a rough approximation of the delay being the number of hops [11].

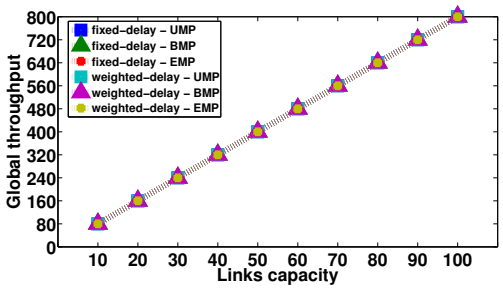


Fig. 2. Uniform capacity case: global throughput (All-to-All)

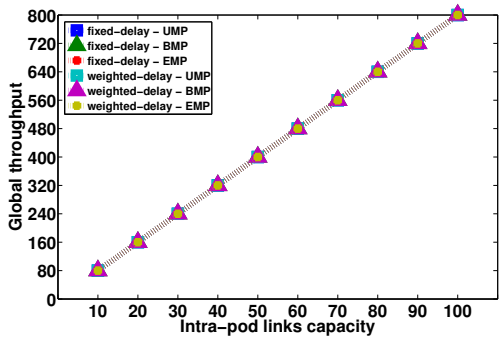


Fig. 3. Asymmetric capacity case: global throughput (All-to-All)

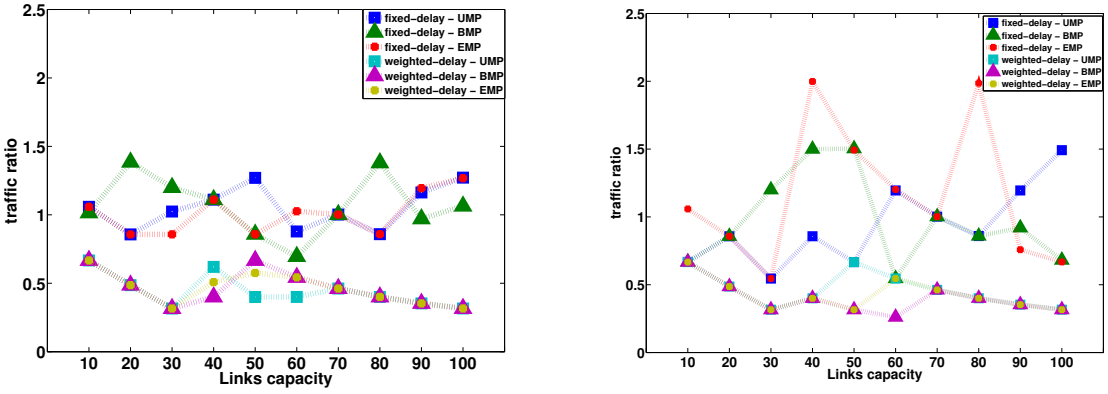
IV. PERFORMANCE EVALUATION

We evaluate our proportional fairness model to understand the fair allocation for competing demands, focusing on a specific data center topology and studying cases to understand the impact of the DCN capacity on traffic fairness. We implemented our study set up using C++ and CPLEX as the solver for the linear programming formulation given by ((9)) - ((12)). In the following, we present the study framework and discuss the simulation results.

A. DCN topology

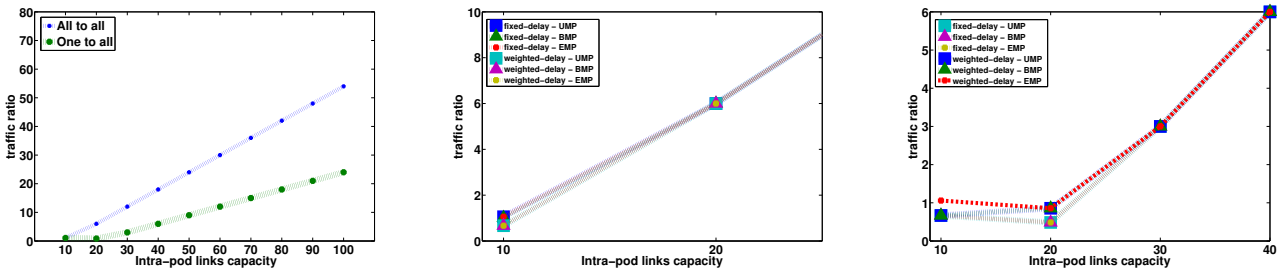
We run our study cases on the fat-tree topology [1], a popular novel DCN architecture, depicted in Figure 1. It interconnects commodity Ethernet switches as a k-ary network where all switches are identical and organized in two layers: core layer and pod layer. At the pod layer, there is an aggregation stage and an edge stage. There are k pods, each one containing two layers of $\frac{k}{2}$ switches. Each k-port switch in the lower layer is directly connected to $\frac{k}{2}$ hosts. Each of the remaining $\frac{k}{2}$ ports is connected to $\frac{k}{2}$ of the k ports in the aggregation stage. There are $(\frac{k}{2})^2$ k-port core switches. Each core switch has one port connected to each of the k pods. The i^{th} port of any core switch is connected to pod i. Figure 1 shows a fat-tree example for $k = 4$ that was used in our study.

The advantage of considering this topology is that it has intra-pod traffic and inter-pod traffic. Secondly, the capacity may be set different for links with pods compared to the links that connect aggregation switches to core switches.



(a) All-to-All (b) One-to-All

Fig. 4. Uniform capacity case: intra-to-inter pod traffic ratio



(a) One-to-All and All-to-All (b) All-to-All (c) One-to-All

Fig. 5. Asymmetric capacity case: intra-to-inter pod traffic ratio

B. Study cases

In order to assess traffic fairness with different levels of the horizontal DCN capacities, we consider the two following DCN dimensioning cases:

- Uniform capacity: all link capacities are set equally. In this case, we consider different capacity configurations, increasing the capacity on all the links from 10 to 100 units in increments of 10 units.
- Asymmetric capacity: the starting configuration has an equal capacity of 10 units per link. We then increase the capacity only on the intra-pod links (link between edge and aggregation switches) from 10 to 100 units by increments of 10 units. The capacity of links between the aggregation and cored switches ("extra-pod links") remains fixed at 10 units.

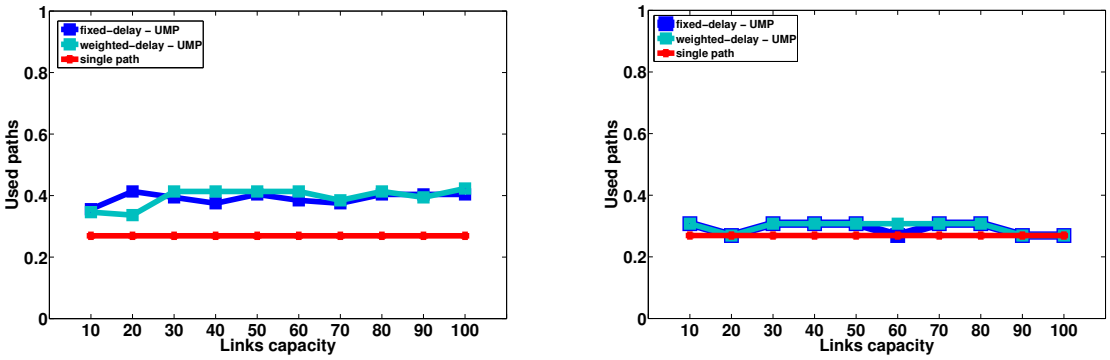
We run our cases for different values of α, i.e., the minimum sub-flow traffic ratio allocated to each path available to a demand d. We consider the following cases:

- Unbounded MultiPath (UMP) case, with $\alpha = 0$, so that multipath forwarding is not forced for any demand, but can be used;
- Bounded MultiPath (BMP) case, with $\alpha = 0.125$, so that multipath forwarding is lightly forced on all available paths for all demands, and can be freely used;
- Equi-distribution MultiPath (EMP) case, with α being replaced by $\alpha_d = 1/N_d$ in (7), where N_d is the number of paths available to demand d, so that traffic distribution is forced to be even over the paths available to each demand.

It is worth noting that for the fat-tree topology, intra-pod traffic can have two paths, while inter-pod traffic can use up to four paths.

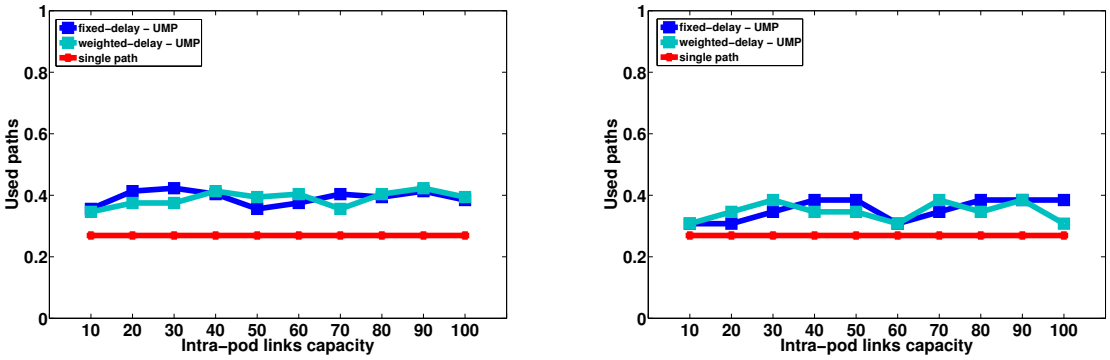
Moreover, we evaluate the results for both the utility functions presented in Section III: the fixed-delay situation given by (13) and the weighted-delay situation ($\bar{\omega}_d$ = hop count) given by (14). These two options allow us to see how fairness is guaranteed for intra-pod and inter-pod traffic. More importantly, a data center provider can decide to deploy its preferred TCP implementation (as they own the servers) by taking advantage of the lessons learned from this study, and accordingly allocate jobs to servers to target traffic fairness. In other words, this study also helps the Cloud provider to decide on fine-grained scheduling of jobs that meets traffic fairness requirements.

In order to show the impact on throughput allocation between intra-pod and inter-pod edge switches, we measure the intra-to-inter-pod traffic allocation ratio. Finally, we measure the path diversity of the solution for the UMP case for all the edge-to-edge demands ("All-to-All") and from the point of view of a single edge switch ("One-to-All").



(a) All-to-All (b) One-to-All

Fig. 6. Uniform capacity case: used paths ratio



(a) All-to-All (b) One-to-All

Fig. 7. Asymmetric capacity case: used paths ratio

C. Results

This subsection illustrates the results of the proportional fairness model concentrating the analysis around three key aspects: the throughput allocation, the traffic distribution within and across pods, and the achieved path diversity.

1) Throughput allocation: First, consider the global throughput when traffic between all pair of edge switches are allowed ("All-to-All"). Next, assume that all extra-pod links are dropped, i.e., there are only intra-pod links with a capacity of 10 each. It is easy to see that each pod is isolated in this case (and there is no inter-pod traffic). Thus, the traffic throughput between the two edge switches in a pod is limited by the capacity of the intra-pod links. Since two links form a path, we can see that the throughput within a pod between its two edge switches is 20. Thus, for the 4-pod fat-tree topology, the total throughput is 80. In this case, also the traffic allocation between intra-pod and inter-pod is most skewed. It is interesting to note that when extra-pod links have a positive capacity, the total throughput still remains at 80 as long as the capacity of the intra-pod links are at 10 units each.

In Figures 2 and 3, the global throughput is plotted as the capacity is increased. We find that it grows linearly as dictated by the capacity of intra-pod links, irrespective of the capacity of the links connecting aggregation and core switches. More importantly, it is not affected by the multipath case (UMP, BMP, EMP) nor the type of the utility function (fixed-delay vs. weighted delay). It is not so trivial to observe that the global throughput appears as being directly proportional to the fixed link capacity at eight times the link capacity for the All-to-All case.

2) The traffic distribution: Next we investigate how the traffic distribution is affected between intra-pod and inter-pod edge switches, for which we use the metric intra-to-inter pod traffic ratio. We characterize the traffic distribution sensibility with respect to the various cases focusing on the intra-to-inter pod traffic ratio and on the traffic ratio between neighboring pods and between non-neighboring pods.

For the uniform capacity case with regards to the intra-to-inter pod traffic ratio for all-to-all demands (Figure 4), we find that on average the allocation between intra-pod and inter-pod traffic is similar with the fixed-delay situation. On the other hand, with the weighted-delay situation, the inter-pod traffic has a traffic proportion that is almost twice that of the intra-pod traffic. This can also be explained since the path hop count of an inter-pod demand (4 hops) is twice that of an intra-pod demand (2 hops)—this is reflected in the weights for the weighted-delay situation.

For the asymmetric capacity case, the observation is strik-

ingly different than the uniform case. The intra-pod demands have 60 times more throughput than inter-pod demands (Figure 5-a) for the all-to-all traffic case when the capacity of the intra-pod links reaches 100, while the extra-pod links capacity was kept fixed at 10. This gain is in alignment with the special case we discussed earlier when there is no capacity on extra-pod links, the most skewed case.

From Figures 5b and 5c, we note that the fixed-delay situation also favors intra-pod traffic and becomes steady when the extra-pod links capacity is three times higher than the intra-pod link capacity. When we have only one source, it becomes steady when the extra-pods link capacity is twice the capacity of the intra-pod links (for the two cases the curves converge when the ratio is equal to 6).

3) Path Diversity: Figures 6 and 7 illustrate path diversity for the UMP case, measured as the ratio of the overall used paths to the number of overall available paths. We also plot the line corresponding to the single-path situation. It is worth remembering that for the BMP and EMP cases, all the paths are used (so it would be a top line at a ratio equal to 1).

Any path diversity of the solution does not seem to be affected by the specific utility function (TCP behavior). We can see that although path diversity was allowed, the unconstrained multipath case did not take full advantage of it. This seems to imply that path diversity is not necessary to maintain the highest throughput.

V. Conclusion

Data center networking is a challenging field of applications of old and new technologies and concepts. In this paper, we investigated DCN capacity sharing among competing greedy demands from a proportional fairness perspective provided by TCP utility functions in the equilibrium.

We presented a generalized formulation of the basic proportional fairness model for DCN allowing multipath forwarding for elastic demands. We also described and evaluated our model under two different TCP utility functions: fixed-delay and weighted-delay.

Through a series of scenarios studied on the 4-pod fat-tree topology, we discovered a number of interesting results. In particular, we found out that the weighted-delay utility function gives twice as much importance to the inter-pod traffic, which may be exploited by the data center provider for high-level scheduling of traffic intensive applications. Another implication is that for a very large IaaS composed of numerous virtual machines needing to span more than a pod, this bias towards intra-pod traffic may be an undesirable behavior, while for a small IaaS this could be a desirable behavior. In our opinion, this should influence the cloud orchestration logic and IaaS management algorithms in VM placements, to be properly designed.

We also measured the path diversity of the solutions in the case in which a systematic multipath mode over all demands was not forced, and multiple path selection was left to the congestion control. We found that only a fraction of the paths was eventually chosen for demands.

Our study, to the best of our knowledge, was the first one to address the impact of DCN topology design, capacity planning and multipath forwarding in traffic fairness in DCN fabrics. We believe the results are interesting and deserve further study, especially grounded on real data as soon as this becomes publicly accessible to researchers.

Acknowledgement

This work was partially supported by the Systematic FUI 15 project RAVIR (http://www.ravir.io), by the EU FP7 IRSES MobileCloud Project (Grant No. 612212), and by National Science Foundation grant CNS-0916505.

References

[1] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," in *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 4. ACM, 2008, pp. 63–74.

[2] T. Benson, A. Akella, and D. Maltz, "Network traffic characteristics of data centers in the wild," in *Proceedings of the 10th annual conference on Internet measurement*. ACM, 2010, pp. 267–280.

[3] L. S. Brakmo and L. L. Peterson, "TCP vegas: End to end congestion avoidance on a global internet," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 8, pp. 1465–1480, 1995.

[4] A. Ford, C. Raiciu, M. Handley, S. Barre, and J. Iyengar, "Architectural guidelines for multipath TCP development," *RFC6182 (March 2011), www. ietf. ort/rfc/6182*, 2011.

[5] C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, Y. Shi, C. Tian, Y. Zhang, and S. Lu, "BCube: a high performance, server-centric network architecture for modular data centers," in *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 4. ACM, 2009, pp. 63–74.

[6] F. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research society*, vol. 49, no. 3, pp. 237–252, 1998.

[7] R. Khalili, N. Gast, M. Popovic, U. Upadhyay, and J.-Y. Le Boudec, "MPTCP is not pareto-optimal: performance issues and a possible solution," in *Proceedings of the 8th international conference on Emerging networking experiments and technologies*. ACM, 2012, pp. 1–12.

[8] S. Kunniyur and R. Srikant, "End-to-end congestion control schemes: Utility functions, random losses and ecn marks," *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 689–702, 2003.

[9] S. H. Low, L. L. Peterson, and L. Wang, "Understanding TCP Vegas: a duality model," *J. ACM*, vol. 49, no. 2, pp. 207–235, Mar. 2002.

[10] R. Mazumdar, L. Mason, and C. Douligeris, "Fairness in network optimal flow control: optimality of product forms," *IEEE Transactions on Communications*, vol. 39, no. 5, pp. 775–782, 1991.

[11] D. Medhi, "Applications with multiple parallel flows: Assessing their unfair advantage with proportional fair sharing TCP," in *Proc. of IEEE International Conference on Communications (ICC'2014)*, Sydney, Australia, June 2014.

[12] D. Oran, "OSI IS-IS intra-domain routing protocol," 1990.

[13] M. Pióro and D. Medhi, *Routing, flow, and capacity design in communication and computer networks*. Elsevier/Morgan Kaufmann, 2004.

[14] R. Srikant, *The mathematics of Internet congestion control*. Birkhauser, 2004.

[15] R. Stewart, "Stream control transmission protocol," 2007.

[16] D. Thaler and C. Hopps, "Multipath issues in unicast and multicast next-hop selection," RFC 2991, November, Tech. Rep., 2000.

[17] J. Touch and R. Perlman, "Transparent interconnection of lots of links (TRILL): Problem and applicability statement," 2009.

[18] D. Wei, C. Jin, S. Low, and S. Hegde, "FAST TCP: Motivation, architecture, algorithms, performance," *IEEE/ACM Transactions on Networking*, vol. 14, no. 6, pp. 1246–1259, 2006.

[19] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: state-of-the-art and research challenges," *Journal of Internet Services and Applications*, vol. 1, no. 1, pp. 7–18, 2010.