On Capacity Provisioning in Datacenter Networks for Full Bandwidth Communication

Wenda Ni*, Changcheng Huang*, and Jing Wu‡
*Dept. of Systems and Computer Engineering, Carleton University, Ottawa, ON, K1S 5B6, Canada
‡Communications Research Centre Canada, Ottawa, ON, K2H 8S2, Canada
E-mail: {wendani, huang}@sce.carleton.ca, jingwu@ieee.org

***Abstract*—Recent advances in datacenter network design have enabled full bandwidth communication based on the notion of Valiant load balancing. In this paper, we study from the link capacity viewpoint how such communication paradigm can be supported in the context of link failures, the study of which is currently absent. In particular, we target full bandwidth communication among all the servers, for all valid traffic patterns, and under k arbitrary link failures. We derive the minimum link capacity required on two typical datacenter topologies—VL2 and fat-tree. Our main finding is that given the same server scale, fat-tree requires less total link capacity than VL2 for $1 \leq k \leq \frac{n}{4}$, where n denotes the port count of homogeneous switches used in fat-tree. For $k > \frac{n}{4}$, there exists a turning point from which VL2 becomes more capacity-efficient.**

I. Introduction

Emerging cloud services are driving the creation of datacenters, which consist of tens to hundreds of thousands of servers. Communication among servers is supported by a datacenter network, which typically consists of multiple tiers of switches. Conventional datacenter networks have a tree-like topology designed using the *scale-up* method [1], [2]. Higher-end switches with higher port speed are required at higher tiers to accommodate higher amount of aggregate traffic. Ideally, port speed moving up the hierarchy should be scaled up accordingly so that any server can communicate with any other servers at the maximum rate of its network interface card (NIC). This is generally referred to as full bandwidth communication [2]. Unfortunately, the cost of such a communication network is prohibitively high due to the deployment of high-price non-commodity switches at higher tiers. Consequently, conventional datacenter networks are constructed with significant oversubscription ratio, meaning that under certain traffic patterns a server can only reach part of its access limit due to the existence of network congestion at higher levels of the switching hierarchy.

To address the oversubscription problem, novel datacenter network infrastructures have been proposed. Typical designs include VL2 [1], fat-tree [2], and BCube [3]. All these new designs take the *scale-out* approach, which leverages a large number of inexpensive commodity switches. Link capacity, or equivalently port speed, is dimensioned to enable full bandwidth communication among all the servers (when there are no failures). The rich connectivity inherent with the scale-out method provides multiple paths between any server pair, allowing resiliency against network failures [1], [2] and Valiant load balancing (VLB) to handle highly variable traffic without "hot-spot" links [1], [4]. However, despite the availability of redundant paths, capacity provisioning in current practice [1], [2] does not support full bandwidth communication in the presence of network failures, which are common and frequent within a datacenter [5]. In other words, any failure can cause network congestion, which is further manifested as increased service latency.

In this paper, we deal with the capacity allocation problem in datacenter networks subject to link failures. We focus on two typical topologies—VL2 and fat-tree. To cope with the highly dynamic traffic [1], [6], we employ Valiant load balancing, a two-phase routing scheme capable of handling traffic variations in a congestion-free manner. This paper answers two important questions: for each topology, how much link capacity is needed at minimum to support full bandwidth communication for arbitrary valid traffic patterns among all the servers under k arbitrary link failures? Given datacenters with the same server scale and failure tolerance level, which of the two topologies is better in terms of the total link capacity required?

The remainder of the paper is organized as follows. In Section II, we present background on network topologies, traffic model, routing structure, and link capacity requirement with no failures. In Sections III and IV, we derive the required capacity to tolerate k link failures for VL2 and fat-tree, respectively. Capacity comparison between the two topologies is presented in Section V. We conclude the paper in Section VI. Due to the page limit, we will omit some of the proofs for brevity.

II. Network Models

Both VL2 and fat-tree consist of three layers of switches, namely, edge layer, aggregation layer, and core layer. Switches at the corresponding layers are referred to as edge switches, aggregation switches, and core switches, respectively. Let $\mathcal{N}_E$ and $\mathcal{N}_A$ denote the sets of edge switches and aggregation switches, respectively. We define links between the edge and the aggregation layers as edge links, and links between the aggregation and the core layers as core links. Let $\mathcal{L}_E$ and $\mathcal{L}_C$ denote the sets of edge links and core links, respectively.

All servers are connected to the network via edge switches, each to one and only one edge switch. Thus, all traffic enters or leaves the network at edge switches. We represent network

978-1-4673-4620-7/13/$31.00 ©2013 IEEE

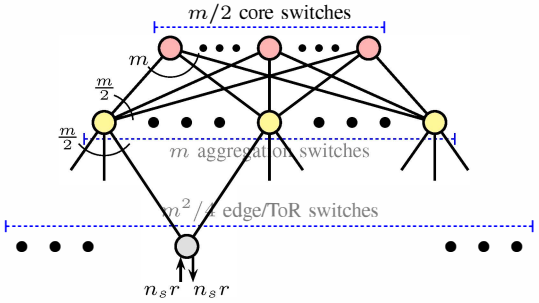

Fig. 1. VL2 topology.

traffic by matrix $\mathbf{\Lambda} = \{\lambda_{ii'}\}_{|\mathcal{N}_E|\times|\mathcal{N}_E|}$, where element $\lambda_{ii'}$ $(i \neq i')$ denotes the traffic demand from edge switch i to edge switch i'. $\lambda_{ii} \equiv 0$. Due to the traffic variations, traffic matrix is constantly changing at both large and small timescales.

A. VL2

1) Topology: Fig. 1 shows the VL2 topology. Interconnection between the aggregation and the core layers forms a complete bipartite graph. If we construct both layers using m-port switches, m switches are deployed at the aggregation layer. The aggregation switches use half its ports to connect to the core layer. This leads to $m/2$ switches at the core layer. The other $m/2$ ports of the aggregation switches are connected to the edge layer, which uses top of rack (ToR) switches as edge switches. Each ToR switch has two links connected to the aggregation layer. In particular, two links are connected to two different aggregation switches for redundancy. VL2 topology has $m^2/2$ edge links and $m^2/2$ core links.

2) Traffic Model: Let r denote the maximum sending/receiving rate of each server NIC. Let n_s denote the number of servers supported by one ToR switch. The ingress/egress capacity of each edge switch is thus bounded by $n_s r$. By "ingress/egress", we mean traffic that indeed goes through the network, and thus excludes local traffic, which bounces off edge switches. Note that "local traffic" refers to traffic among servers that are hosted by the same edge switch. Any valid traffic matrix $\mathbf{\Lambda}$ satisfies the following constraints:

$$\sum_{i'\in\mathcal{N}_E, i'\neq i} \lambda_{ii'} \leq n_s r, \qquad i \in \mathcal{N}_E, \tag{1}$$

$$\sum_{i\in\mathcal{N}_E, i\neq i'} \lambda_{ii'} \leq n_s r, \qquad i' \in \mathcal{N}_E. \tag{2}$$

3) VLB: VL2 topology provides two two-hop paths between an edge switch and a core switch. Ingress traffic from edge switch i to edge switch i' is first sent to a randomly chosen core switch over a path chosen at random [1]. The core switch then forwards the traffic to destination i' over a randomly chosen path [1]. All traffic is forwarded on a per-packet basis.

The above process can be equivalently viewed as follows: In the first phase, traffic from i to i' is evenly split over m two-hop paths that go to $m/2$ core switches. In the second phase, traffic is forwarded from all core switches to destination i' over m two-hop paths with equal split. It is easy to find that routing in two phases is symmetric.

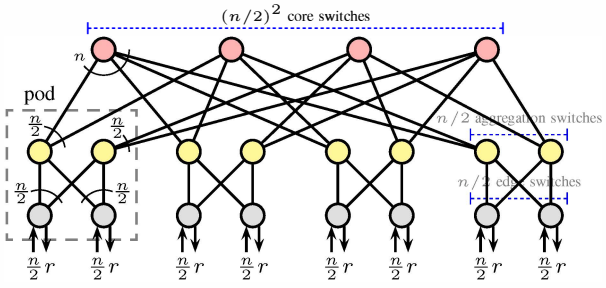

Fig. 2. Fat-tree topology with $n = 4$.

4) Link Capacity with no Failures:

Proposition 1: To guarantee full bandwidth communication among all the servers under the no-failure scenario (i.e., $k = 0$), the minimum capacity required on each link is

$$c_l(0) = \frac{n_s r}{2}, \qquad l \in \mathcal{L}_E \cup \mathcal{L}_C. \tag{3}$$

The proof is omitted for brevity. The minimum capacity required for $k = 0$ is consistent with that given in [1] .

B. Fat-Tree

1) Topology: Fat-tree is a multi-rooted tree topology. Edge and aggregation switches are arranged in the form of switching modules called pods, which are interconnected by core switches representing the multiple roots of a fat-tree.

If n-port switches are used to construct a fat-tree, each pod consists of $n/2$ edge switches and $n/2$ aggregation switches. Within a pod, each edge switch is connected to each aggregation switch by one port, forming a complete bipartite graph. Externally, each pod is connected to each core switch by one of the other half $(n/2)^2$ ports of its aggregation switches. Consequently, $(n/2)^2$ switches are required at the core layer. On the other hand, as each core switch has n ports, n pods are supported. To make the topology regular, interconnection between pods and core switches should satisfy the following condition: there exists a partition of core switches into $n/2$ equal-sized mutually-disjoint sets such that each aggregation switch of each pod is connected to core switches belonging to the same set. An illustrative topology with $n = 4$ is given in Fig. 2.

Each pod has $(n/2)^2$ edge links and $(n/2)^2$ core links.

2) Traffic Model: Each edge switch has $n/2$ ports connected to $n/2$ servers. Thus, the ingress/egress capacity limit of each edge switch is $\frac{n}{2}r$. Recall that ingress/egress traffic refers to traffic that originates and terminates at different edge switches. Any valid traffic matrix $\mathbf{\Lambda}$ satisfies the following constraints:

$$\sum_{i'\in\mathcal{N}_E, i'\neq i} \lambda_{ii'} \leq \frac{n}{2} r, \qquad i \in \mathcal{N}_E, \tag{4}$$

$$\sum_{i\in\mathcal{N}_E, i\neq i'} \lambda_{ii'} \leq \frac{n}{2} r, \qquad i' \in \mathcal{N}_E. \tag{5}$$

3) VLB: Each edge switch can communicate with another edge switch in the same pod or any edge switch in a remote pod. We refer to traffic among edge switches of the same pod

as intra-pod traffic, and traffic destined for an edge switch in a remote pod as inter-pod traffic.

As the core layer connects a pod to a remote pod, intra-pod traffic does not go outside of a pod. Specifically, packet from an edge switch is first sent to a randomly chosen aggregation switch inside a pod. The aggregation switch then forwards the packet to the destination edge switch resided in the same pod. In both phases, packets are delivered through direct links. From the end-to-end viewpoint, intra-pod traffic is evenly distributed over $n/2$ two-hop paths between any two edge switches.

On the other hand, inter-pod traffic goes through the core layer outside of an individual pod. Core switches take the role of intermediate nodes in the two-phase routing. It is easy to find that routing in two phases is symmetric. As each core switch has only one two-hop path towards each edge switch, we can virtually concatenate the two paths in two phases for each core switch. Then, from the end-to-end viewpoint, inter-pod traffic is evenly split over $(n/2)^2$ four-hop paths between any two edge switches.

4) Link Capacity with no Failures:

Proposition 2: To guarantee full bandwidth communication among all the servers under the no-failure scenario (i.e., $k = 0$), the minimum capacity required on each link is

$$c_l(0) = r, \qquad l \in \mathcal{L}_E \cup \mathcal{L}_C. \tag{6}$$

We omit the proof for brevity. The minimum capacity required on each link coincides with the original design in [2], where VLB is not employed.

III. Capacity Allocation for VL2

In this section, we derive link capacity requirement on VL2 topology that experiences k arbitrary link failures. The goal is to guarantee full bandwidth communication among all the servers. Given k, the basis to such end is that the topology remains connected in any k link failures. For VL2 topology above the aggregation layer, it requires at least $\frac{m}{2}$ link failures to disconnect an aggregation switch from any other aggregation switches. This allows us to consider k up to $\frac{m}{2} - 1$. On the other hand, however, each edge switch has only two links connecting to the aggregation layer. Thus, any multiple link failures can disconnect any edge switch. This severely limits the fault tolerance capability of VL2 topology. To facilitate comparison with fat-tree topology, which supports k up to $\frac{n}{2} - 1$, we consider k up to $\frac{m}{2} - 1$. For $k \geq 2$, we focus on failure scenarios where all edge switches remain connected.

A. Edge Links

Due to the limited connectivity of each edge switch, capacity requirement on edge links is derived straightforward. We establish the following theorem.

Theorem 1: Let k be a given integer with value $1 \leq k \leq \frac{m}{2} - 1$. To guarantee full bandwidth communication among all the servers under k arbitrary link failures which do not partition the topology, the minimum capacity required on each edge link is

$$c_l(k) = n_s r, \qquad l \in \mathcal{L}_E. \tag{7}$$

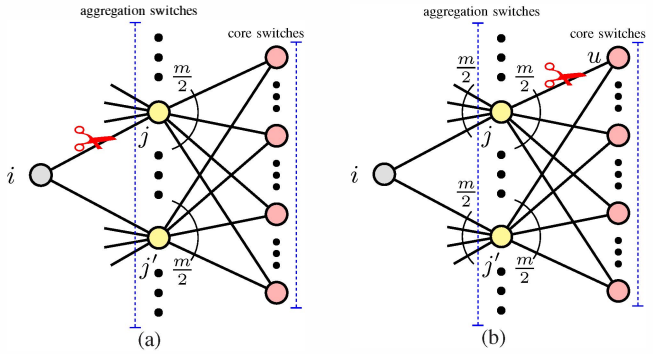

Fig. 3. Single link failure on VL2. (a) A failed edge link. (b) A failed core link.

Proof: Assume that edge switch i is connected to the aggregation layer through aggregation switches j and j'. In the case that edge link (i, j) fails, link (i, j') carries all traffic originating from i. This leads to a maximum load of $n_s r$ on link (i, j'). The reverse direction of (i, j') carries all traffic that terminates at i. From the routing symmetry, we know immediately that the maximum load on link (j', i) is $n_s r$. Considering all link failure scenarios, capacity allocation on all edge links is uniform, with a minimum value of $n_s r$. ■

B. Core Links

In this subsection, we derive link capacity requirement for each core link. We begin with single link failures, and then deal with k arbitrary link failures.

Theorem 2: To guarantee full bandwidth communication among all the servers under arbitrary single link failures, the minimum capacity required on each core link is

$$c_l(1) = \frac{n_s r}{2} + \frac{n_s r}{m}, \qquad l \in \mathcal{L}_C. \tag{8}$$

Proof: As shown in Fig. 3, a single link failure can occur on an edge link or a core link. For both cases, we discuss load increase on a core link to carry the disrupted traffic. We first consider the first routing phase, where traffic goes from the edge layer to the core layer.

In the case that an edge link fails, let (i, j) denote the failed edge link, which is incident to edge switch i and aggregation switch j. Let j' denote the other aggregation switch i is connected to. The failed link disrupts half the paths from i to the core layer, and thus half the traffic from i to any other edge switch i'. The disrupted traffic is evenly distributed to the remaining $m/2$ paths that go to the core layer from i. Each core link (j', u) is on one of the remaining paths. Thus, load increase on link (j', u) is given by $\sum_{i' \in \mathcal{N}_E, i' \neq i} \frac{\lambda_{ii'}/2}{m/2}$, with a maximum value of $\frac{n_s r}{m}$ following from (1).

Let $\delta(j)$ denote the set of edge switches connected to aggregation switch j. We have $|\delta(j)| = m/2$. In the second case that a core link fails, let (j, u) denote the failed core link incident to aggregation switch j and core switch u. The failed link disrupts one of the m paths for each edge switch in set $\delta(j)$, and thus $\frac{1}{m}$ the amount of traffic from each i in $\delta(j)$. The disrupted traffic is evenly assigned to the remaining $m-1$ paths from i to the core layer. For aggregation switch j, each operating core link incident to j, i.e., (j, u'), $\forall\, u' \neq u$,

is on one of the remaining paths for all $i \in \delta(j)$. Thus, load increase on core link $(j,u'), \forall u' \neq u$ is given by $\sum_{i\in\delta(j)} \frac{\frac{1}{m}\sum_{i'\in\mathcal{N}_E, i'\neq i}\lambda_{ii'}}{m-1}$, with a maximum value of $\frac{n_s r}{2(m-1)}$ following from (1) and $|\delta(j)| = m/2$. For any aggregation switch j' other than j, each core link incident to j' is on one of the $m-1$ operating paths for edge switches that are connected to both j and j', i.e., $i \in \delta(j) \cap \delta(j')$. Thus, the maximum load increase on core link $(j',u'), \forall j' \neq j, \forall u'$ is $|\delta(j) \cap \delta(j')| \cdot \frac{n_s r}{m(m-1)}$, which is upper bounded by $\frac{n_s r}{2(m-1)}$ due to the fact that $|\delta(j) \cap \delta(j')| \leq \frac{m}{2}$.

Considering both cases with all failure scenarios, the maximum load increase experienced on any core link $(j,u), \forall j \in \mathcal{N}_A$ is $\frac{n_s r}{m}$. The reverse direction of (j,u) carries traffic from the core layer to their destination edge switches. From the routing symmetry in two phases, we know immediately that the maximum load increase on link (u,j) is $\frac{n_s r}{m}$. Therefore, the minimum extra capacity required on any core link l is $\frac{n_s r}{m}$, and the theorem follows immediately. ■

Now we move to k arbitrary link failures with the following theorem.

Theorem 3: Let k be a given integer with value $1 \leq k \leq \frac{m}{2} - 1$. To guarantee full bandwidth communication among all the servers under k arbitrary link failures which do not partition the topology, the minimum capacity required on each core link is

$$c_l(k) = \begin{cases} \frac{n_s r}{2} + \frac{k n_s r}{m} & \text{if } 1 \leq k \leq \frac{m}{6}, \\ n_s r \cdot \max\limits_{k_c \in \left\{\lfloor \hat{k}_c^1 \rfloor, \lceil \hat{k}_c^1 \rceil\right\}} f(k_c, k) & \text{if } \frac{m}{6} < k \leq \frac{m}{2} - 1, \end{cases} \quad l \in \mathcal{L}_C, \quad (9)$$

where

$$\hat{k}_c^1 = \frac{k}{2} + \frac{m}{4} - \frac{1}{4}\left[(3m-2k)(m-2k)\right]^{\frac{1}{2}}, \quad (10)$$

and function $f(k_c, k)$ is defined as

$$f(k_c, k) \triangleq \frac{k - k_c}{\frac{m}{2} - k_c} + \frac{\frac{m}{2} - k + k_c}{m - k_c}. \quad (11)$$

Proof: Omitted for brevity. ■

IV. Capacity Allocation for Fat-Tree

In this section, we consider fat-tree subject to link failures. Fat-tree requires at least $\frac{n}{2}$ link failures to disconnect the topology. Thus, with sufficient link capacity, we can guarantee full bandwidth communication among all the servers up to $\frac{n}{2} - 1$ arbitrary link failures. In the following, we investigate the minimum link capacity requirement to such end. Let $\mathcal{P}$ denote the set of pods. Let $\mathcal{Q}_p \in \mathcal{N}_E$ denote the set of edge switches in pod p. We have $|\mathcal{Q}_p| = \frac{n}{2}, \forall p$.

A. Edge Links

Each edge link only carries traffic that originates and terminates at the incident edge switch. This allows us to derive capacity requirement on edge links for the general case straightforward. We establish the following theorem.

Theorem 4: Let k be an integer with value $1 \leq k \leq \frac{n}{2} - 1$. To guarantee full bandwidth communication among all the servers under k arbitrary link failures, the minimum capacity required on each edge link is

$$c_l(k) = r + \frac{kr}{\frac{n}{2} - k}, \qquad l \in \mathcal{L}_A. \quad (12)$$

Proof: We first consider load increase on edge links caused by originating traffic, which all goes in the direction from the edge layer to the aggregation layer. Consider i to be a general edge switch. Let i be resided in pod p. From the background discussions in Section II-B, we know that when there are no failures, all intra- and inter-pod traffic originating from i is evenly distributed among the $n/2$ edge links incident to i with a maximum load of r per link. Each edge link is on one and only one of the $n/2$ paths carrying intra-pod traffic from i to any other edge switch $i' \in \mathcal{Q}_p$, and is on $n/2$ of the $(n/2)^2$ paths carrying inter-pod traffic from i to any remote edge switch $i' \in \mathcal{N}_E \setminus \mathcal{Q}_p$.

When k failures are on k of the $n/2$ edge links incident to i, the failed links disrupt the maximum number of paths from i to any other edge switch i' ($\forall i' \in \mathcal{N}_E \setminus \{i\}$). The disrupted traffic is then evenly distributed to the remaining operating paths that traverse the residual $\frac{n}{2} - k$ operating links incident to i. As all paths traversing these operating links are not affected under the failure scenario, the amount of disrupted traffic assigned to each of the links is thus maximized and with equal split. As the total amount of disrupted traffic from i is kr at maximum under the considered failure scenario (and under any k arbitrary link failures), the maximum load increase on each of the operating edge links incident to i is thus $\frac{kr}{\frac{n}{2}-k}$.

The reverse direction of the operating edge links carries traffic that terminates at i. Due to the symmetry of the two routing phases, it immediately follows that the reverse direction of these links experiences the same amount of maximum load increase. Considering all failure scenarios, the maximum load increase on each edge link is uniform. Consequently, the extra capacity required on each edge link l is $\frac{kr}{\frac{n}{2}-k}$, and the theorem follows immediately. ■

B. Core Links

Each core link in pod p carries inter-pod traffic that originates and terminates at each edge switch in p. We start with single link failures to gain insights, and then generalize the case to k arbitrary link failures.

Theorem 5: To guarantee full bandwidth communication among all the servers under all single link failures, the minimum capacity required on each core link is

$$c_l(1) = r + \frac{r}{\left(\frac{n}{2} - 1\right)\frac{n}{2}}, \qquad l \in \mathcal{L}_C. \quad (13)$$

Proof: We first consider load or load increase on core links caused by traffic in the first routing phase; that is, traffic that goes to the core layer. Consider p to be a general pod. The failed link can be in p or in a remote pod, and can be an edge link or a core link. This leads to four different cases as

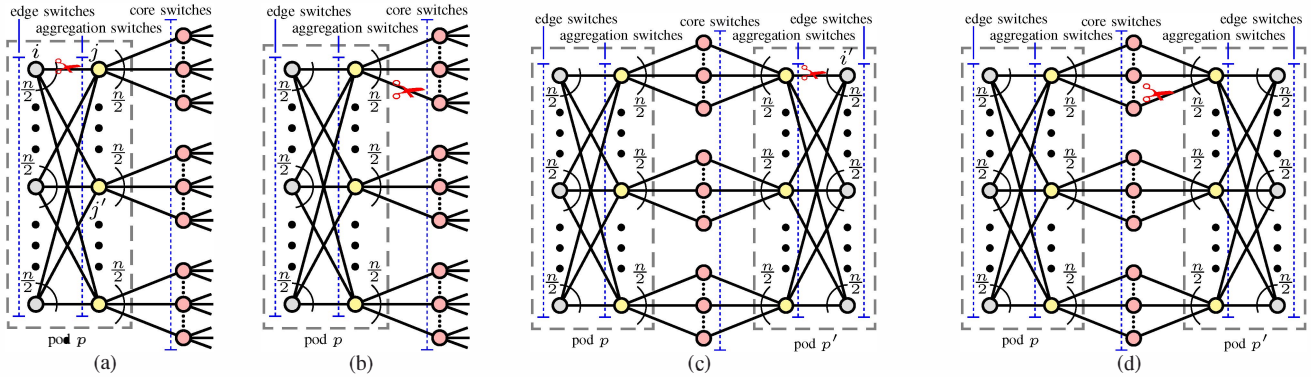

Fig. 4. Single link failure on fat-tree. (a) A failed edge link in local pod p. (b) A failed core link in local pod p. (c) A failed edge link in remote pod p'. (b) A failed core link in remote pod p'.

shown in Fig. 4. In the following, we discuss the maximum load or load increase on a core link in p for all four cases.

In the case that an edge link in p fails, let (i, j) denote the failed link incident to edge switch i and aggregation switch j. The failed link disrupts $n/2$ of the $(n/2)^2$ paths from i to any remote edge switch i' ($\forall\, i' \in \mathcal{N}_E \setminus \mathcal{Q}_p$), and thus $\frac{1}{n/2}$ the amount of traffic from i to any remote i'. The disrupted traffic is evenly assigned to the remaining $\left(\frac{n}{2}-1\right)\frac{n}{2}$ operating paths from i to i'. Let j' denote an aggregation switch other than j in pod p. Each core link incident to aggregation switch j' is on one of the remaining operating paths for each remote edge switch. Thus, load increase on each core link incident to j' is expressed as $\sum_{i' \in \mathcal{N}_E \setminus \mathcal{Q}_p} \frac{\frac{1}{n/2}\lambda_{ii'}}{\left(\frac{n}{2}-1\right)\frac{n}{2}}$, which takes the maximum value of $\frac{r}{\left(\frac{n}{2}-1\right)\frac{n}{2}}$ when inter-pod traffic from i reaches the ingress capacity limit given by (4). Note that due to the absence of traffic from i, the maximum load on each core link incident to j decreases.

The case that a core link in p fails can be developed in the similar fashion. The failed core link disrupts one of the $(n/2)^2$ paths that go from any edge switch in p to any remote edge switch, and thus disrupt $\frac{1}{(n/2)^2}$ the amount of traffic from any edge switch in p to any remote edge switch. The disrupted traffic is evenly distributed to the remaining $(n/2)^2 - 1$ operating paths. As each of the operating core links in p is on one of these paths from any edge switch to any remote edge switch, load increase on each of the operating core links is thus computed as $\sum_{i \in \mathcal{Q}_p} \sum_{i' \in \mathcal{N}_E \setminus \mathcal{Q}_p} \frac{\frac{1}{(n/2)^2}\lambda_{ii'}}{(n/2)^2-1}$. The maximum value is reached at $\frac{r}{(n/2)^2-1}$ when all $n/2$ edge switches in p send inter-pod traffic at the maximum rate $\frac{n}{2}r$.

In the case that an edge link fails in remote pod p', let the failed link be incident to edge switch i' in p'. Unlike the previous case of edge link failure in local pod p, the failed link only affects inter-pod traffic destined for i'. Specifically, it disrupts $n/2$ paths from each edge switch in p to i'. A core link in p is either on one operating path from each edge switch in p to i' or on one failed path from each edge switch in p to i'. From the discussions in Section II-B, we know that when there are no failures, each core link carries the maximum load r when inter-pod traffic from p is $(\frac{n}{2})^2 r$. That is, each edge switch in p sends inter-pod traffic at the maximum rate $\frac{n}{2}r$. Among the $(\frac{n}{2})^2 r$ amount of traffic, traffic destined for i' is $\frac{n}{2}r$ at maximum, which is limited by the egress capacity at i' given in (5). Under such traffic patterns, the total traffic carried by the paths to fail is at its maximum amount r. This amount of traffic is evenly assigned to the remaining $\left(\frac{n}{2}-1\right)\frac{n}{2}$ operating paths from the corresponding edge switches to i' under the considered failure scenario. In this setting, core links on these operating paths experience the maximum load. As each core link on one of the $\left(\frac{n}{2}-1\right)\frac{n}{2}$ operating paths from one edge switch in p to i' is also on one of the $\left(\frac{n}{2}-1\right)\frac{n}{2}$ operating paths from any edge switch in p to i', the disrupted traffic is thus evenly split over $\left(\frac{n}{2}-1\right)\frac{n}{2}$ core links, regardless of the source edge switches in p. It immediately follows that the maximum load on each of these links is equal to $r + \frac{r}{\left(\frac{n}{2}-1\right)\frac{n}{2}}$. Note that the maximum load on the counterpart core links in p, i.e., $n/2$ core links on the failed paths to i', is still r considering the variety of traffic patterns.

In the last case of a failed core link in a remote pod p', the failed core link disrupts one of the $(n/2)^2$ paths from each edge switch in p to each edge switch in p'. A core link in p is either on one of the operating paths from each edge switch in p to each edge switch in p' or on the one failed path from each edge switch in p to each edge switch in p'. When there are no link failures, each core link carries the maximum load r when all traffic from p is inter-pod traffic. Among this traffic, traffic towards pod p' can be of amount $(n/2)^2 r$ at maximum when all traffic is destined for p'. Under such traffic patterns, the total traffic carried by the paths to fail is maximized, with the maximum value being r. This amount of traffic is evenly distributed to the corresponding $(n/2)^2 - 1$ operating paths. Accordingly, core links on the operating paths experience the maximum load in the event of the considered failure scenario. Note that the operating paths can be diverse in terms of source and destination in p and p', respectively. However, as each core link on one of the $(n/2)^2 - 1$ operating paths from one edge switch in p to one edge switch in p' is also on one of the $(n/2)^2 - 1$ operating paths from any edge switch in p to any edge switch in p', the disrupted traffic is thus evenly split over $(n/2)^2 - 1$ core links, regardless of the source and destination edge switches in p and p', respectively. Consequently, the maximum load on each of these links is $r + \frac{r}{(n/2)^2-1}$. Note that the maximum load of the one core link on the failed paths is still r considering all possible traffic

patterns.

Considering all failure scenarios of all four cases as well as the symmetry in two routing phases, we know that the maximum load in both directions of each core link is $r + \frac{r}{\left(\frac{n}{2}-1\right)\frac{n}{2}}$, and the theorem follows immediately. ■

With the knowledge on single link failures, we now generalize the capacity requirement to k arbitrary link failures with the following theorem.

Theorem 6: Let k be an integer with value $1 \leq k \leq \frac{n}{2} - 1$. To guarantee full bandwidth communication among all the servers under k arbitrary link failures, the minimum capacity required on each core link is

$$c_l(k) = r + \frac{kr}{\left(\frac{n}{2} - k\right)\frac{n}{2}}, \qquad l \in \mathcal{L}_C. \tag{14}$$

Proof: Omitted for brevity. ■

V. Capacity Comparison between VL2 and Fat-Tree

In this section, we compare the total link capacity required on VL2 and fat-tree. As the minimum capacity required in both directions of each link is the same, we model both topologies as undirected graphs. The numbers of edge and core links are given in Section II. Based on the minimum link capacity derived in Sections III and IV, we establish the following theorem.

Theorem 7: Given the same total number of supported servers, the total link capacity required on VL2 for $k = 1$ is approximately the same as that on fat-tree for $k = \frac{n}{4}$.

Proof: Omitted for brevity. ■

Corollary 1: For $1 \leq k \leq \frac{n}{4}$, fat-tree outperforms VL2 in terms of total link capacity.

Corollary 1 follows immediately from Theorem 7.

Fig. 5 compares the total link capacity against the number of link failures. We assume that the maximum rate of each server NIC is 1 Gb/s, i.e., $r = 1$ Gb/s. We set $n_s = m = n$ so that both networks support the same number of servers, and are with the same range of k.

We observe that given the number of servers, the capacity gap between $k = 0$ and $k = 1$ is large on VL2. This is because capacity on all edge links is doubled when we move from the case of no failures to the case of single link failures considering (3) and (7). This part of capacity remains constant among all $k \geq 1$. For $k \geq 1$, capacity increase comes from core links. In particular, the total capacity increases linearly with k when k is within $1 \leq k \leq \frac{m}{6}$, and increases super-linearly with k when k is in the range $\frac{m}{6} < k \leq \frac{m}{2} - 1$. Such increase trend is dictated by the minimum capacity requirement on core links given in (9).

In sharp contrast, all curves are flat for small values of k on fat-tree. However, the total link capacity increases at a more rapid pace as k gets larger, showing an "exponential-like" increase trend.

The graceful capacity growth of VL2 with $k \geq 1$ and the "exponential-like" capacity increase on fat-tree lead to a cross-point between the two curves, as seen in Fig. 5. Clearly, the cross-point is no smaller than $\frac{n}{4}$ according to Corollary 1. The exact point can be found in Fig. 5. We see that for $n_s = m = n = 20, 40, 60$, and 80, the total link capacity of fat-tree is smaller than that of VL2 when k is no greater than 6, 12, 18, and 25, respectively. In all cases, the value of the cross-point is not significantly larger than $\frac{n}{4}$.

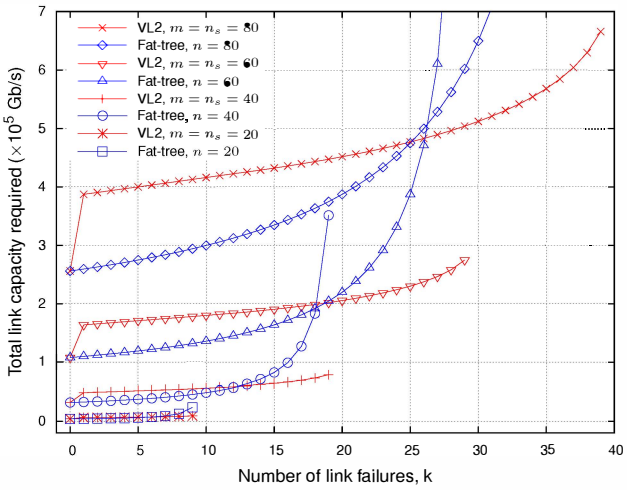

Fig. 5. Total link capacity comparison between VL2 and fat-tree.

VI. Conclusion

We studied capacity allocation problem in datacenter networks that employ VLB to handle highly variable traffic. Our design goal is to guarantee full bandwidth communication among all the servers, for all valid traffic matrices, and under k arbitrary link failures. From the connectivity viewpoint, k is supported up to $\frac{n}{2} - 1$ on fat-tree whereas k is limited to 1 on VL2 in the strict sense. Thus, for multiple link failures on VL2, we consider failure scenarios that do not disconnect the topology. In such context, we derived the minimum capacity required on links. We showed that given the same total number of supported servers, fat-tree requires less total capacity than VL2 for $1 \leq k \leq \frac{n}{4}$. For $k > \frac{n}{4}$, there exists a turning point beyond which VL2 is better due to the sharp capacity increase on fat-tree in this regime.

Acknowledgment

Dr. Jing Wu acknowledges the research support from the State Key Laboratory of Advanced Optical Communication Systems and Networks, Shanghai Jiao Tong University, China.

References

[1] A. Greenberg, *et al.*, "VL2: a scalable and flexible data center network," In *Proc. ACM SIGCOMM*, pp. 51–62, 2009.

[2] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," in *Proc. ACM SIGCOMM*, pp. 63–74, 2008.

[3] C. Guo, *et al.*, "BCube: A high performance, server-centric network architecture for modular data centers," in *Proc. ACM SIGCOMM*, pp. 63–74, 2009.

[4] W. J. Dally, and B. Towles, *Principles and Practices of Interconnection Networks*. San Francisco, CA: Morgan Kaufmann, 2004.

[5] R. N. Mysore, *et al.*, "PortLand: a scalable fault-tolerant layer 2 data center network fabric," in *Proc. ACM SIGCOMM*, pp. 39–50, 2009.

[6] A. R. Curtis, T. Carpenter, M. Elsheikh, A. López-Ortiz, and S. Keshav, "REWIRE: an optimization-based framework for unstructured data center network design", in *Proc. IEEE INFOCOM*, pp. 1116–1124, 2012.