Scaling of Capacity and Reliability in Data Center Networks

Milan Bradonjić  
Bell Labs, Alcatel-Lucent  
Murray Hill, NJ 07974  
milan@research.bell-labs.com

Iraj Saniee  
Bell Labs, Alcatel-Lucent  
Murray Hill, NJ 07974  
iis@research.bell-labs.com

Indra Widjaja  
Bell Labs, Alcatel-Lucent  
Murray Hill, NJ 07974  
i_widjaja@yahoo.com

ABSTRACT

The traditional connectivity model within the data center is that of a hierarchical tree with redundant connections (“fat tree”) and with a top node consisting of one or more routers that bring in (and send out completed) requests for processing. In this paper we examine alternative connectivity models for large-scale data centers. In the first model, we examine hypergrids as the structure connecting switches and routers to the edge server racks. In the second model, we examine random graphs as the interconnecting network. We compare and contrast the capacity, congestion and reliability requirements for these relative to fat-trees. We show that, as the system size increases and for uniform switch-end-to-switch-end demand, the fat-tree configuration emerges as an expensive option demanding higher port density switches but has low congestion and high reliability. In contrast, the random graph model shows the same low level of congestion, lower cost due to fewer ports and reasonable reliability, whereas the hypergrid model does not require scaling of switch ports, provides high reliability but exhibits higher congestion.

Categories and Subject Descriptors
C.0 [General]: System architectures; C.4 [Performance of Systems]; H.3.4 [Systems and Software]

Keywords
Data center networks, capacity, reliability, fat-trees, hypergrids, random graphs

1. INTRODUCTION

Large scale data centers in which hundreds of thousands of commodity servers, switches and routers are used in place of specialized equipment are changing the profile of communication systems. The network that interconnects servers forms an important component of a data center. Such a network has to provide: (1) high-performance connectivity among the servers, such as low server-to-server latency and low bottleneck at switches or links and (2) be low cost. These two divergent objectives present interesting technical challenges to the design of data center networks. The traditional fat-tree model is reaching its limits due to port limitation of the top-of-the-rack (TOR) switches. Recently, researchers have identified a variety of shortcomings of current data center architectures, e.g., [1]. Other studies have considered alternative data center configurations with focus on cost [2, 3, 5]. Amongst these [2, 5] consider 1) hypergrid model and 2) random graph model of data center racks for analysis of routing. As the size of a data center increases to accommodate more servers in the future, one may expect the load of the most congested switch to also increase according to some function \( f(S) = O(S^\alpha) \) and \( \alpha > 0 \), where \( S \) is the number of servers (proportional to the number of racks), as a result of one unit of flow between all pairs of servers. Such a measure has been investigated in the context of Internet or IP-based networks [4] but not in the context of data center architecture.

In this work, we investigate the scaling behavior of the congestion function as well as reliability, for different data center configurations. Specifically, we focus on three data center network topologies: fat-tree, hypergrid, and random graph. We quantify the pros and cons of each topology, via \( f(S) \), based on congestion, reliability, and cost.

2. BASIC PROPERTIES OF DATECENTERS

We study three types of data center configurations: fat-tree, hypergrids and random graphs. We define \( R=\) no. of racks, \( N=\) no. of switches, \( S=\) no. of servers, \( M=\) no. of links, \( K=\) no. of parallel links, and \( P=\) no. of ports and \( f=\) E-E flow, the end-to-end or TOR-switch-to-TOR-switch flow. The nominal link and port capacity used in this work is assumed to be \( C = 10\text{Gb/s} \) and we assume 40 servers per rack, thus \( R = S/40 \).

2.1 Fat-trees: \( \text{FT}(k,l) \)

Toplogy. A fat-tree network is typically built by replicating a number of constant-degree switches at each level of a tree. Figure 1(a) shows an example of a 3-level fat-tree supporting 8 racks of servers, each TOR switch has 4 ports. Typical data center fat-tree networks adopt two or three levels [1].

Number of switches and racks. A fat-tree network using \( k\)-port switches with \( l\) levels, denoted by \( \text{FT}(k,l) \), requires a total of \( (2^l - 1)(k/2)^{l-1} \) switches. The number of switches at all levels \( l \geq 2 \), equals \( 2(k/2)^{l-1} \), and half as many at level 1. The total number of racks supported by \( \text{FT}(k,l) \) is \( R = (k/2)^l \), without redundancy and \( R = 2(k/2)^l \) with 2-redundancy.

Links and Paths. The total number of “internal” links is \( 2(l - 1)(k/2)^l \), plus \( 2(k/2)^l \) links connected to the racks. For the network in Figure 1(a), \( k = 4 \) and \( l = 3 \), there are 32 internal links, 16 rack links, and \( (4l - 2)(k/2)^l = 80 \)
ports. It is not hard to see that the network diameter is $2l$ and the average shortest path length between two racks is $\sum_{i=0}^{l-1} 2(l-i)(1/2)^{i+1} = (2^{l+1} - l - 2(2^l - 1) + l)/2$.

**Congestion.** Under uniform traffic, where each layer-1 switch sends one unit of flow to other level-1 switches, the maximum load is $l - 1$, where $l$ is the number of level-1 switches. Therefore, the maximum switch-to-switch flow size at level-$l$ is $(k/2)C/(l-1)$, where $C$ is the link capacity. Since each level-1 switch is connected to $k/2$ racks, the rack-to-rack flow size is approximately $(2/k)C/(l-1)$. Table 1(a) summarizes key metrics for a fat-tree.

### 2.2 Hypergrids: $HG(p,q)$

**Topology.** A hypergrid $HG(p,q)$ is a planar graph where each internal node has a fixed degree $q$ and each face is a $p$-gon, where $1/p + 1/q < 1/2$, see Figure 1(b) when $p = 3$ and $q = 7$. Hypergrids are good candidates for connectivity in data centers because (1) their degree (number of ports) could be fixed ahead of time and this is key limit in expanding current fat-tree topologies, see [1], (2) as in trees, they have diameters that grow proportionally to the logarithm of the total number of racks. By increasing $q$ to the maximum port number allowed and selecting $p \geq 3$, one creates a large parameter space to trade off many engineering constraints.

**Number of switches and racks.** In general, a hypergrid may be built using an arbitrary number of levels $l$, with $\lambda^l$ switches at each level $l$, for some $\lambda = \lambda(p,q) > 1$, depending on the desired size of racks to inter-connect. The total number of switches $N = 1 + \lambda + \cdots + \lambda^{l-1} = (\lambda^{l+1} - 1)/(\lambda - 1)$ is proportional to the number of racks on the outer ring $\lambda^l$. Hence the hypergrid achieves very large scales with small diameter as well as massive redundancy without increasing switch or router port size.

**Links and Paths.** The number of disjoint paths between any pair of nodes is $q$, and the number of paths increases exponentially. The shortest path length between two racks is easily seen to be $O(\log N)$ and by exponential growth also equal to $O(\log R)$, as $R \approx N$.

**Congestion.** Given $R$ racks in the outer periphery of a hypergrid, it is shown that for uniform traffic of one unit between all pairs of racks, there is at least one switch which experiences load in the scale of $R^2$, e.g., roughly $R^2/6$ for $HG(3,7)$, see [4]. Table 1(b) summarizes the key metrics of a hypergrid. We see the decrease in port size in the hypergrid compared to the fat-tree. The upper half of Table 1(b) uses only 7-port switches, while the lower half uses 14 and 35 port switches in the inner 1-2 layers, respectively. The quadratic scaling of the load in the hypergrid can be alleviated (shown in the bottom half of Table 1(b)), by marginal addition of links and ports in the inner two layers.

### 2.3 Random Graphs: $G(N,p)$

**Topology.** Here we consider the Erdős–Rényi model $G(N,p)$ that consists of $N$ switches. The degree of a switch follows the binomial $\text{Bin}(N-1,p)$ distribution and the expected number of edges contained in the graph is $p(N-1)$. We assume racks are connected to each switch, so there are no inner-level switches, as shown in Figure 1(c).

**Number of switches and racks.** It is known that a graph $G(N,p)$ becomes connected with high probability when $p \geq \log N/N$. We consider a connected $G(N,p)$, where $p = 2\log N/N$ and Figure 1(c) shows one realization of $G(20,0.3)$, with $N = 20$ switches and the edge probability $p = 2\log N/N \approx 0.3$. Each switch is connected to 4 racks of servers.

**Links and Paths.** The length of shortest paths in random graph with $N$ nodes ranges from 1 to $O(\log N)$ with
high probability. The average shortest path length is of the same order as diameter.

**Congestion.** The maximum nodal load for uniform flows between all node pairs of a random graph is not known analytically. We evaluate maximum switch load in the Erdős-Rényi graph architecture for different numbers of switches from $N = 10$ to $N = 1000$ via simulations. For each value of $N$ the experiment is repeated 100 times and the results are plotted in Figure 2. Using linear regression on the log-log plot: Maximal Load vs Number of Switches, Bars represent 25%, 75% around the median value of the maximal loads.

3. RELIABILITY

Reliability is measured by the number of randomly deleted edges between switches which would result in disconnecting a pair of racks. We assume server racks are singly or doubly connected to the edge switches. The experimental results after 100 repetitions for fat-tree, random graph and hypergrid, are presented in Table 2. For the random graph model we additionally provide explicit formulas. Under uniform random edge deletion the expected number of links $\Delta M$ to be deleted in $G(N, p = c \log N / N)$ in order to disconnect the graph is asymptotically given by $\mathbb{E}\Delta M = \frac{(c - 1)n}{2} \log n$, and $\Delta M$ is asymptotically concentrated around its mean $\mathbb{E}\Delta M$. So deleting $\frac{1}{2}(c - 1)n \log n$ ensures that $G_{N, p}$ will disconnect asymptotically almost surely.

4. COMPARISON

The cost and congestion characteristics of each of the three designs we have considered are summarized in Table 2. Fat-trees have the least congestion as measured by average port number per rack is roughly 5 for fat-tree, 3 for hypergrid and 2.5 for random graph. For emerging very large data centers with over 1/2 million servers, the high port density requirement of fat-trees creates a serious impediment to scaling. Similarly, random graphs have low congestion, have lower port requirements than fat-trees but have lower reliability than fat-trees. Finally, hypergrids have fixed port size for any data center size, have lower port density per rack than fat-trees but suffer from lower rack-to-rack flow size than fat-tree and random graph models, Table 2.

2.5 for random graph. For emerging very large data centers with over 1/2 million servers, the high port density requirement of fat-trees creates a serious impediment to scaling. Similarly, random graphs have low congestion, have lower port requirements than fat-trees but have lower reliability than fat-trees. Finally, hypergrids have fixed port size for any data center size, have lower port density per rack than fat-trees but suffer from lower rack-to-rack flow size than fat-tree and random graph models, Table 2.

5. REFERENCES


