Power Consumption Analysis of Data Center Architectures

Rastin Pries, Michael Jarschel, Daniel Schlosser, Michael Klopf, and Phuoc Tran-Gia

University of Würzburg, Institute of Computer Science,
Chair of Communication Networks, Würzburg, Germany
{pries,michael.jarschel,
schlosser,trangia}@informatik.uni-wuerzburg.de

Abstract. The high power consumption of data centers confronts the providers with major challenges. However, not only the servers and the cooling consume a huge amount of energy, but also the data center network architecture makes an important contribution. In this paper, we introduce different data center architectures and compare them regarding their power consumption. The results show that there are some differences which should not be neglected and that with only minor modifications of the architecture, it is possible to save a huge amount of energy.

Keywords: data center, energy efficiency, networking.

1 Introduction

Data centers are attracting more and more interest, offering a large variety of services such as online gaming, data storage, data processing, and online office products. However, there are still a lot of challenges to be solved, e.g., overall performance, energy efficiency, resilience, scalability, and how to transport the data to the consumer. Most data center providers currently focus on building their data centers only with commercial off-the-shelf (COTS) hardware to reduce the cost and to be easily maintainable. In addition, the data center should be easily extensible and should scale up to 100,000 servers. Therefore, the new data centers are compromised of containers, each carrying up to 2,500 servers.

Besides this information, most cloud providers keep their data center architectures as a secret. Only facebook lately set up the Open Compute Project [1] releasing their open hardware especially designed for data centers. However, the data center network architecture is not yet released and it is stated that they work within the also newly created Open Networking Foundation [2] to create a new, energy efficient data center network architecture.

There are several ways of how to reduce the power consumption in a data center, ranging from energy efficient server hardware as proposed by facebook over coordinated cooling and load management to full virtualization. Generally, the energy efficiency of a data center is measured using the Power Usage Effectiveness (PUE) which was developed by the green grid consortium. The PUE is computed as follows:

$$PUE = \frac{\text{total facility power}}{\text{IT equipment power}}. \quad (1)$$

Joel J.P.C. Rodrigues et al.: (Eds.): GreeNets 2011, LNICST 51, pp. 114–124, 2012.

An ideal PUE is 1.0, whereas the state-of-the-art industry average is 1.5. The new facebook data center has a PUE of 1.07 calculated at full load over an 8 hour period in December 2010.

In this paper, we instead focus on the power consumption of data center network architectures and evaluate the currently deployed architectures and some proposed architectures according to their power consumption. We evaluate the following six architectures, two-tier, three-tier, DCell, BCube, fat-tree, and elastic-tree. So far, these architectures have only been compared by Wu et al. [3] regarding the number of necessary switches, cables, etc. and Chen et al. [4] provided an overview of routing in data centers, also considering energy-efficiency on the routing layer. Another paper looking at the power consumption of today's data centers is proposed by Poess and Nambiar [5]. In the paper, a power consumption estimation model for TPC-C benchmarks is proposed. The model is applied to published TPC-C benchmarks and the performance and energy performance trends are shown. The only paper looking at a similar direction as in this paper is published by Gyarmati and Trinh [6]. Unfortunately, their power consumption figures only show some isolated results and thus, the architectures are difficult to compare. In contrast to their publication, we evaluate more data center architectures and show the power consumptions for architectures from a few server to up to 70,000 servers.

The remainder of the paper is structured as follows. In Section 2, we describe the evaluated data center architectures. Section 3 shows the used parameters for the evaluation of the data center architectures. The results from the performance evaluation are described in Section 4. We conclude the paper by summarizing our main contributions in Section 5.

2 Data Center Architectures

Several different network architectures have been proposed for data centers ranging from switch-centric approaches such as butterfly, Clos network, and VL2 to server-centric approaches such as mesh, torus, star, ring, hypercube, DCell, and BCube. In this paper, we only focus on the most promising and well-known approaches and evaluate their impact on the total power consumption. All six considered architectures are introduced in the following.

2.1 Two-Tier Architecture

A two-tier data center architecture is shown in Figure 1. The servers are arrange into racks and form together with the Top of Rack (ToR) switch the tier one. A number of racks together form a Performance Optimized Data center (POD) which are nowadays 20 or 40 ft. containers. The servers are usually connected via a 1 Gbps Ethernet cable to the ToR switch who are also connected with the same bandwidth to the second tier. The second tier is formed by layer-3 switches which on the one hand connect the racks within the containers and on the other hand interconnect the containers using currently 10 GE links. According to Kliazovich et al. [7], Equal Cost Multi-Path (ECMP) routing is used for load balancing. Typically, a two-tiered design can support between 5,000 to 8,000 hosts [8]. To reduce the number of links and thus the costs of the equipment for the two-tier architecture, the branches of the trees are usually oversubscribed by a factor of 1:2.5 to 1:8 [8].

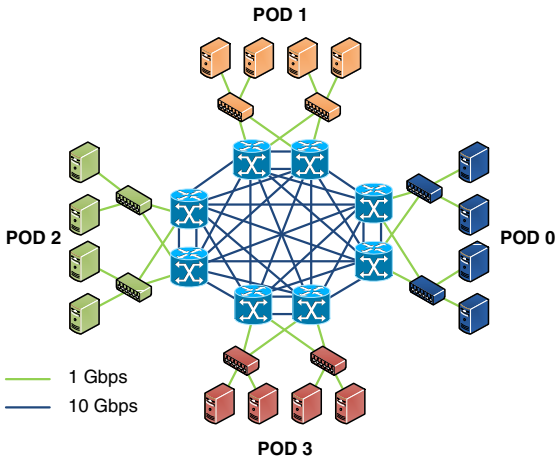

Fig. 1. Two-tier data center architecture

2.2 Three-Tier Architecture

The three-tier data center architecture is currently the most common architecture. It consists of three different layers, the access layer, the aggregation layer, and the core layer as shown in Figure 2. The aggregation layer facilitates the increase in the number of server nodes (more than 10,000 servers) while keeping inexpensive layer-2 switches in the access network for providing a loop-free topology. Similar to the two-tier architectures, the branches of the tree are oversubscribed and the highest levels of the tree can be oversubscribed by a factor of 1:80 to 1:240 [9]. The reason is that the three-tier architecture is often used for data processing such as the MapReduce algorithm. For this, the exchange of data is mostly kept within one rack and only one-tenth of the traffic is sent outside a rack. The three-tier architecture also normally uses ECMP for load balancing and as the maximum number of allowed ECMP paths is eight, a typical three-tier architecture consists of eight core switches. Figure 2 only shows two core switches. The current connection between the layers is similar to the two-tier architecture. However, it is intended to increase the link speed between the aggregation layer and the core layer to 40 GE or even 100 GE links [7].

2.3 DCell Architecture

The DCell data center architecture was developed to provide a scalable infrastructure and to be robust against server failures, link outages, or server-rack failures [10]. A DCell physical structure is a recursively defined architecture whose servers have to be equipped with multiple network ports. Each server is connected to other servers and to a mini switch, cf. Figure 3. In the example, $n = 4$ servers are connected to a switch, forming a level-0 DCell. According to Guo et al. [10], n should be chosen ≤ 8 to be able to use commodity 8-port switches with 1 Gbps or 10 Gbps per port. A level-1 DCell is constructed using $n + 1$ level-0 DCells, in our example 5 level-0 DCell form the level-1

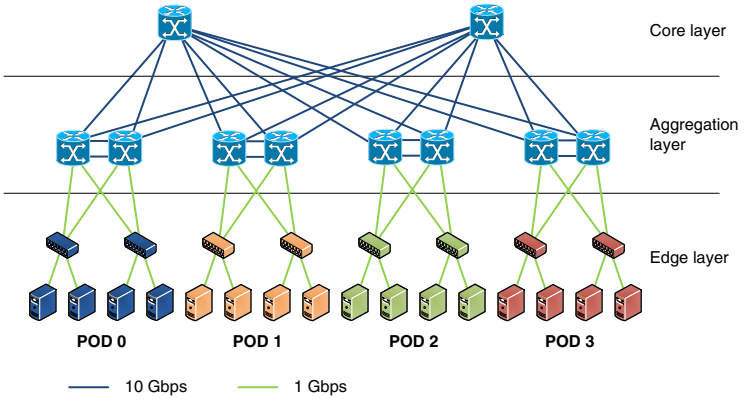

Fig. 2. Three-tier data center architecture

DCell. In order to connect the level-0 DCells, each DCell is connected to all other DCells with one link. A level-2 DCell and the level-k DCell are constructed the same way. Thus, the DCell architecture is a server-centric structure which uses commercial switches and the fewest number of switches of all presented data center architectures. However, the cabling complexity might prevent large deployments.

The goal of the DCell scheme is according to Guo et al. [10] to interconnect up to millions of servers. Thus, a global link-state routing scheme cannot be applied. Therefore, a new routing protocol is proposed, called DCell Fault-tolerant Routing (DFR) which is a decentralized touring solution. More information about the routing protocol can be found in Guo et al. [10].

2.4 BCube Architecture

BCube is similar to the DCell structure, just that the server-to-server connections are replaced by server-to-switch connections for faster processing [11]. Figure 4 shows a BCube_k ($k = 1$) architecture with $n = 4$ servers per switch. From the figure we can see that the total number of servers is $N = n^{k+1}$ and each server has to be equipped with $k + 1$ ports. Each level has n^k switches and the total number of levels is $k + 1$. Similar to DCell and in contrast to the following fat-tree architecture, BCube is server-oriented and can use existing commercial Ethernet switches. To be able to fully utilize the multi-path structure of the BCube and to automatically load-balance the traffic, a BCube Source Routing (BSR) protocol is proposed by Guo et al. [11]. In the paper it is also shown that the BCube architecture is more robust against server and switch failures compared to the DCell architecture and the following fat-tree architecture. However, in contrast to the DCell architecture, the BCube architecture should mainly be used for server interconnection within a container. To create larger data center architectures with more than 2,500 server, another architecture is proposed which is called Modularized Data center Cube (MDCube) [3]. With MDCube, multiple BCubes are interconnected by using 10 Gbps interfaces of switches in BCube. The routing between the different containers is realized using single-path routing.

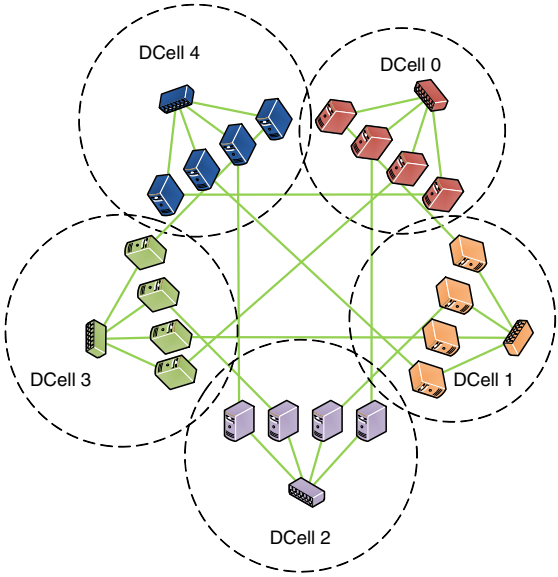

Fig. 3. DCell data center architecture

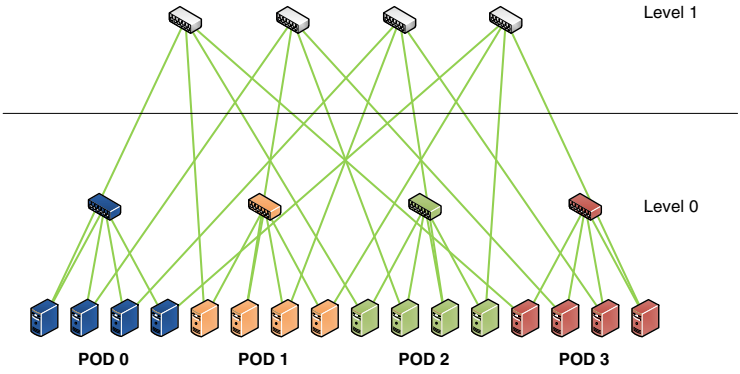

Fig. 4. BCube data center architecture

2.5 Fat-Tree Architecture

In contrast to the general three-tier topology and similar to the DCell and BCube architecture, a fat-tree topology uses commercial Ethernet switches [8, 12]. The fat-tree architecture was developed to reduce the oversubsciption ratio and to remove the single point of failures of the hierarchical architecture. As similar switches are used on all layers of the architecture, the costs for setting up a fat-tree data center can be kept low. The architecture is not achieving complete 1:1 oversubscription in reality, but offers rearrangeably non-blocking paths with full bandwidth. An example of a fat-tree data center architecture is shown in Figure 5. The figure shows a 4-ary fat-tree which is build up of $k = 4$ PODs, each containing two layers of $k/2$ switches.

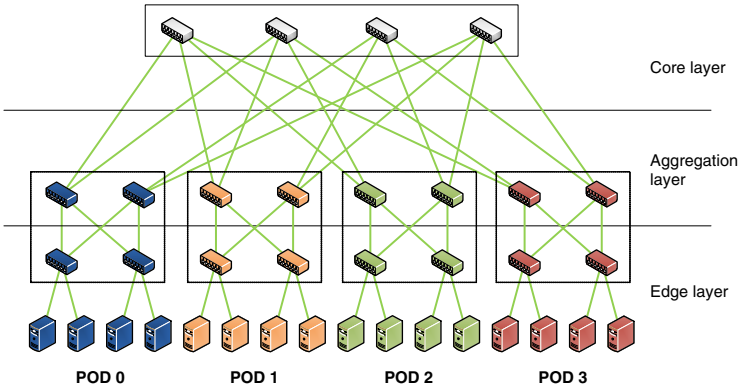

Fig. 5. Fat-tree data center architecture

The switches in the edge layer are connected to $k/2$ servers and the remaining ports of the edge switches are connected to the aggregation layer, cf. Figure 5. The core layer consists of $(k/2)^2$ k-port core switches where each of them is connected to each of the k PODs [8]. A fat-tree data center architecture built with k-port switches support $k^3/4$ servers. Thus, when using 48-port switches, up to 27,648 server can be supported. The example in Figure 5 shows that fat-tree is a switch-centric structure where the switches are concatenated. The VL2 architecture proposed by Greenberg et al. [9] is quite similar to fat-tree except that fewer cabling is needed. They claim that switch-to-switch links are faster than server-to-switch links and therefore use 1 Gbps links between server and switch and 10 Gbps links between the switches. By this, they reduce the number of cables required to implement the Clos. However, high-end intermediate switches are needed and thus, the trade-off made is the cost of those high-end switches.

2.6 Elastic-Tree Architecture

All the above mentioned mesh-like approaches help to be robust against failures by using more components and more paths which of course also increases the power consumption. However, although the number of traffic fluctuates during the day, the power consumption is fixed, see e.g. Google production data center [13]. Thus, Heller et al. [13] propose to reduce the power consumption by dynamically turning off switches and links that are not needed. The approach is called elastic-tree whose underlying topology is a fat-tree. Figure 6 shows an example of the elastic tree, where 7 switches are turned off compared to the normal fat-tree topology.

Using such energy-efficient data center architecture, it has to be ensured that the performance does not degrade, meaning that in case of high load, the switches should be able to start up almost immediately to enable multi-path transmissions. In addition, also in case of switch failure, the elastic-tree architecture has to immediately react to it. Taking these challenges into account, we will later see the effect on the overall power consumption.

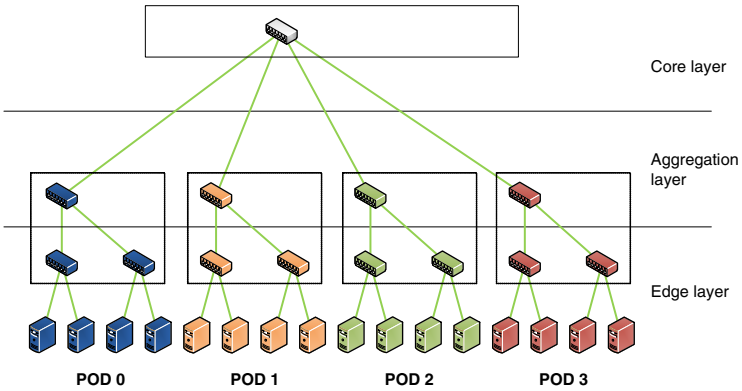

Fig. 6. Elastic-tree data center architecture

3 Evaluation Setup

To evaluate the power consumption of the six introduced data center architectures, we use the parameters shown in Table 1. The parameters were either measured ourselves, taken from published papers, or taken out of the handbook of the switches and routers. For the evaluation in the next section, we use these parameters and choose the required switch depending on the data center architecture as well as on the size of the data center. We scale the number of servers from one or a few hundred, depending on the architecture, to up to 70,000 servers. The evaluation of the power consumptions and the shares of the different parts responsible for the energy consumption is done using Matlab. In the next section, we show the results of our study.

Table 1. Parameters used for evaluation

	Consumption	**Reference**
server	145 Watt	HP ProLiant 2.13 GHz 10 GB RAM
cabling	0.4 Watt (1 Gbps) 6 Watt (10 Gbps)	[7]
linecard	5 Watt	[11]
COTS switch	145 Watt (48 port) 100 Watt (24 port) 13.4 Watt (16 port) 6 Watt (8 port)	NEC IP8800 NEC IP8800 D-Link DGS-1016D D-Link DGS-1008D
Core switch/ router	198 Watt (48 port) 3,500 Watt (128 port) 10,700 Watt (512 port)	 HP A9508-V HP A12500

4 Performance Evaluation

Using the parameters described in the previous section, we first compare all data center power consumption values for a varying number of servers. The results are shown in Figure 7. The results show that the overall power consumption is quite similar, with only minor differences. The two-tier and three-tier architectures together with the BCube architecture have the lowest power consumption while the DCell architecture shows the worst performance. However, all architectures have a power consumption between 10 and 12 MWatt for 70,000 servers.

The similarity of the results rises to the suspicion that the servers are the main contributors of the overall power consumption. To underline this, we now take a look at the shares of the power consumers for the architectures. This is shown in Figure 8. The figure illustrates that more than 88 percent of the total power is consumed by the servers for all data center architectures.

The second largest consumer when using the DCell or the BCube architecture are the linecards. The reason for this huge amount of power consumption is that the servers are included in the switching process and that for each hierarchy level an additional linecard is needed within the server. For all other architectures, the switches are the second largest consumer. Surprising is that the two-tier and three-tier switches have a lower power consumption compared to the fat-tree switches although layer-3 core switches with a lot more power consumption are used. The reason is that the fat-tree architectures uses a lot more switches compared to the other two architectures to be resilient against network failures. Now that we know that the main power consumers are the servers, we can focus on the network equipment to see the differences of the architectures. Figure 9 shows these differences again for an increasing number of servers. For less than 18,000

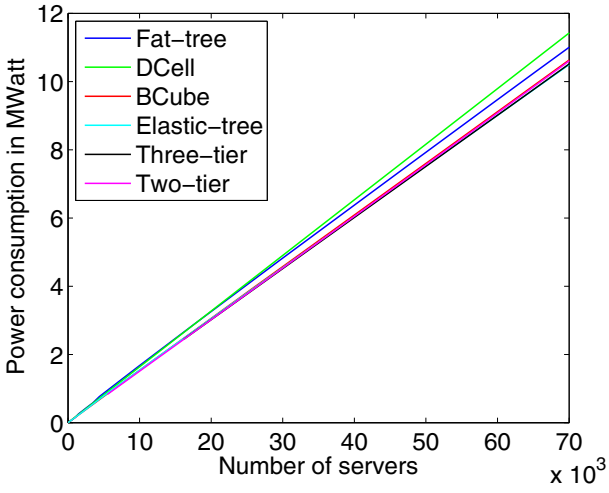

Fig. 7. Overall power consumption for different data center architectures

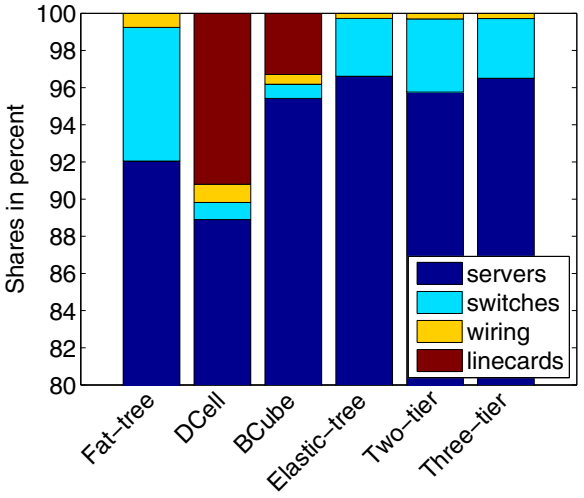

Fig. 8. Relative total power consumption

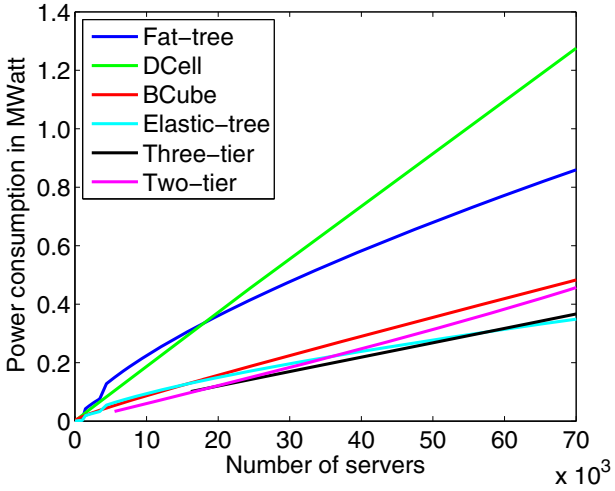

Fig. 9. Network power consumption

servers, the fat-tree architecture shows the worst performance but when increasing the number of servers, the power consumption of the DCell architecture overtakes the fat-tree power consumption. The reason is the increasing number of linecards within the servers. The best performance is shown by the three-tier and the elastic-tree architecture. Both power consumptions are less than one-third of the DCell power consumption for 70,000 servers. However, we have to keep in mind that the elastic-tree architecture uses COTS hardware while the three-tier architectures requires costly layer-3 switches.

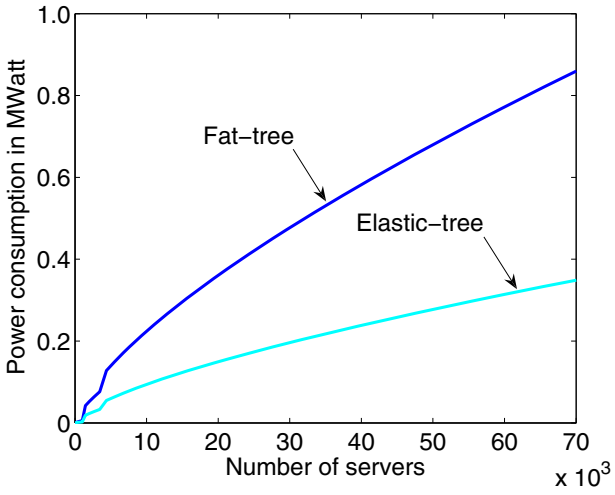

Fig. 10. Power savings of elastic-tree approach

Finally, we want to directly compare the fat-tree architecture with the elastic-tree approach as both use the same architecture, with the only difference that the elastic-tree approach switches off unused components to save energy. The direct comparison is shown in Figure 10.

It can easily be seen that the network equipment of the elastic-tree architecture consumes about half of the power compared to the fat-tree architecture. Thus, the potential for energy saving is tremendous just by turing off unused network equipment. However, in case of a network failure, the unused equipment has to be switched on as fast as possible to avoid data loss.

In addition to the network equipment, also the servers can be switched off when the load in the data center is low. In such a case, the jobs can be migrated to as few servers as possible, while the other are switched off. However, also here the startup time of the servers have to be taken into account and thus, there is always the trade-off between energy-efficiency and Quality of Service.

5 Conclusion

Although the servers in a data center consume most of the power, we showed in this paper that the power consumption of the network equipment should not be neglected. About 4% to 12% of the overall power consumption can be attributed to the networking hardware. Here, the three-tier architecture shows the best performance but uses the most costly hardware. However, the results in this paper illustrate that the total power consumption depends not only on the used data center architecture but also on the implemented energy saving mechanisms. For example, the fat-tree architecture - when used as proposed - consumes a lot of power due to the resilient paths to the servers.

When not used networking components are switched off, the power consumption can be reduced by about 60% as shown with the elastic-tree architecture.

In future work, we will implement the elastic-tree approach in real hardware and we want to consider also a possible server switch off. Therefore, we will have to consider the time needed for virtual machine migration as well as the time needed to switch a server on.

Acknowledgments. The authors would gratefully thank Michael Düser and Fritz-Joachim Westphal from Deutsche Telekom Laboratories for the fruitful discussions and support on this paper.

References

1. Facebook: Open Compute Project (2011), http://opencompute.org/
2. ONF: Open Networking Foundation (2011), http://www.opennetworkingfoundation.org/
3. Wu, H., Lu, G., Li, D., Guo, C., Zhang, Y.: MDCube: A high performance network structure for modular data center interconnection. In: Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies (CoNEXT), Rome, Italy, pp. 25–36 (2009)
4. Chen, K., Hu, C., Zhang, X., Zheng, K., Chen, Y., Vasilakos, A.V.: Survey on routing in data centers: Insights and future directions. IEEE Network 25(4), 6–10 (2011)
5. Poess, M., Nambiar, R.O.: Energy cost, the key challenge of today's data centers: A power conumption analysis of TPC-C results. VLDB Endowment 1(2), 1229–1240 (2008)
6. Gyarmati, L., Trinh, T.A.: How can architecture help to reduce energy consumption in data center networking. In: e-Energy 2010: Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking, Passau, Germany, pp. 183–186 (2010)
7. Kliazovich, D., Bounvry, P., Audzevich, Y., Khan, S.U.: Greencloud: A packet-level simulator of energy-aware cloud computing data centers. In: IEEE Globecom, Miami, FL, USA (2010)
8. Al-Fares, M., Loukissas, A., Vahdat, A.: A scalable, commodity data center network architecture. In: SIGCOMM 2008: Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication, Seattle, WA, USA, pp. 63–74 (2008)
9. Greenberg, A., Hamilton, J.R., Jain, N., Kandula, S., Kim, C., Lahiri, P., Maltz, D.A., Patel, P., Sengupta, S.: VL2: A scalable and flexible data center network. SIGCOMM Comput. Commun. Rev. 39(4), 51–62 (2009)
10. Guo, C., Wu, H., Tan, K., Shi, L., Zhang, Y., Lu, S.: DCell: A scalable and fault-tolerant network structure for data centers. SIGCOMM Comput. Commun. Rev. 38(4), 75–86 (2008)
11. Guo, C., Lu, G., Li, D., Wu, H., Zhang, X., Shi, Y., Tian, C., Zhang, Y., Lu, S.: BCube: a high performance, server-centric network architecture for modular data centers. In: SIGCOMM 2009: Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication, Barcelona, Spain, pp. 63–74 (2009)
12. Mysore, R.N., Pamboris, A., Farrington, N., Huang, N., Miri, P., Radhakrishnan, S., Subramanya, V., Vahdat, A.: Portland: a scalable fault-tolerant layer 2 data center network fabric. SIGCOMM Comput. Commun. Rev. 39(4), 39–50 (2009)
13. Heller, B., Seetharaman, S., Mahadevan, P., Yiakoumis, Y., Sharma, P., Banerjee, S., McKeown, N.: Elastic tree: Saving energy in data center networks. In: 7th USENIX Symposium on Networked System Design and Implementation (NSDI), San Jose, CA, USA, pp. 249–264 (2010)