CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE
Concurrency Computat.: Pract. Exper. 2012
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/cpe.2963

SPECIAL ISSUE PAPER

Quantitative comparisons of the state-of-the-art data center architectures

Kashif Bilal[1], Samee U. Khan[1,*], Limin Zhang[1], Hongxiang Li[2], Khizar Hayat[3], Sajjad A. Madani[3], Nasro Min-Allah[3], Lizhe Wang[4], Dan Chen[5], Majid Iqbal[3], Cheng-Zhong Xu[6] and Albert Y. Zomaya[7]

[1]*North Dakota State University, Fargo, ND 58108, USA*
[2]*University of Louisville, Louisville, KY 40292, USA*
[3]*COMSATS Institute of Information Technology, Pakistan*
[4]*Chinese Academy of Sciences, Beijing, China*
[5]*China University of Geosciences, Wuhan, China*
[6]*Wayne State University, Detroit, USA*
[7]*University of Sydney, Australia*

SUMMARY

Data centers are experiencing a remarkable growth in the number of interconnected servers. Being one of the foremost data center design concerns, network infrastructure plays a pivotal role in the initial capital investment and ascertaining the performance parameters for the data center. Legacy data center network (DCN) infrastructure lacks the inherent capability to meet the data centers growth trend and aggregate bandwidth demands. Deployment of even the highest-end enterprise network equipment only delivers around 50% of the aggregate bandwidth at the edge of network. The vital challenges faced by the legacy DCN architecture trigger the need for new DCN architectures, to accommodate the growing demands of the 'cloud computing' paradigm. We have implemented and simulated the state of the art DCN models in this paper, namely: (a) legacy DCN architecture, (b) switch-based, and (c) hybrid models, and compared their effectiveness by monitoring the network: (a) throughput and (b) average packet delay. The presented analysis may be perceived as a background benchmarking study for the further research on the simulation and implementation of the DCN-customized topologies and customized addressing protocols in the large-scale data centers. We have performed extensive simulations under various network traffic patterns to ascertain the strengths and inadequacies of the different DCN architectures. Moreover, we provide a firm foundation for further research and enhancement in DCN architectures. Copyright © 2012 John Wiley & Sons, Ltd.

Received 17 June 2012; Revised 30 August 2012; Accepted 31 October 2012

KEY WORDS: data center networks; network architecture; data center architectures

1. INTRODUCTION

A data center is a pool of computing resources clustered together using communication networks to host applications and store data. Major information and communication technology components of the data center are the following: (a) servers and (b) network infrastructure. Conventional data centers are modeled as a multilayer hierarchical network with thousands of low-cost commodity servers as the network nodes. Data centers are experiencing exponential growth in number of hosted servers. Google, Yahoo, and Microsoft already host hundreds of thousands of servers in their respective data centers [1, 2]. Google was having more than 450,000 servers in 2006 [3, 4], and the servers are doubling in number every 14 months at the Microsoft data centers [5].

*Correspondence to: Samee U. Khan, North Dakota State University, Fargo, ND 58108, USA.
†E-mail: samee.khan@ndsu.edu

Increased number of servers demands fault tolerant, cost-effective, and scalable network architecture with maximum inter-node communication bandwidth. Another important design aspect of the data center is the use of low-cost commodity equipment. The server portion of data centers has experienced enormous commoditization, and low-cost commodity servers are used in data centers instead of high-end enterprise servers. However, the network portion of the data center has not seen much commoditization and still uses enterprise-class networking equipment [6]. Increased number of servers demands high end-to-end bisection bandwidth. The enterprise-class network equipment is expensive, power hungry, and is not designed to accommodate Internet-scale services in data centers. Therefore, the use of enterprise-class equipment experiences limited end-to-end network capacity, non-agility, and creation of fragmented server pools [6].

Data center network (DCN) is typically based on the three-tier architecture [7]. Three-tier data center architecture is a hierarchical tree-based structure comprised of three layers of switching and routing elements having enterprise-class high-end equipment in higher layers of the hierarchy. Example of the three-tier DCN architecture is shown in Figure 1 [7, 8]. Unfortunately, deployment of even the highest-end enterprise-class equipment may provide only 50% of end-to-end aggregate bandwidth [9]. To accommodate the growing demands of data center communication, new DCN architectures are required to be designed.

Most of the Internet communication in the future is expected to take place within the data centers [10]. Many applications hosted by data centers are communication intensive, such as more than 1,000 servers may be touched by a simple web search request. Communication pattern in a data center may be one-to-one, all-to-all, or one-to-all [11]. The major challenges in the DCN design include the following: (a) scalability, (b) agility, (c) fault tolerance, (d) end-to-end bisection bandwidth, (e) robustness against single point of failure, (f) automated naming and address allocation, and (g) backward compatibility.

Data center network architecture is a major part of data center design acting as a communication backbone and requires extreme consideration. Numerous DCN architectures have been proposed in the recent years [9, 10, 12–18]. This paper provides a comparative study and analysis of major DCN architectures that are proposed in the recent years by implementing: (a) proposed network architectures, (b) customized addressing scheme, (c) customized routing schemes, and (d) different network traffic patterns.

We have implemented the fat-tree based architecture [9], recursively defined architecture [12, 13], and legacy three-tier DCN architecture to compare the performance under six different network traffic patterns. For the fat-tree DCN architecture, we implemented the n-pod based network interconnection design, customized network addressing scheme for servers and switches at different levels, and customized two-level routing algorithm. For the recursive-based DCell DCN architecture, we applied customizable n-level network architecture (up to four levels scalable for more than 3.6 million servers), a generic network addressing scheme, and the DCell routing algorithm. DCell routing algorithm [12] returns a series of nodes (e.g. [001] [010]) as intermediate hops between source and destination. We formulated an algorithm to find the network address-based end-to-end

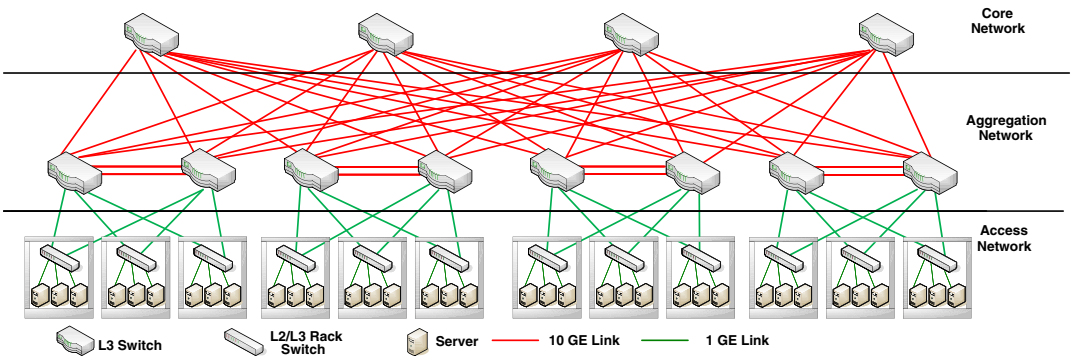

Figure 1. Three-tier data center architecture.

DOI: 10.1002/cpe

path and implemented source-based routing in the ns-3 simulator. Moreover, the DCell routing algorithm pseudocode had some missing information for implementation and working. We formulated the missing information to complete the algorithm. For the legacy three-tier DCN architecture, we implemented customizable network architecture as reported in [7, 8]. We used the Equal Cost Multi-Path (ECMP) [19] routing to obtain realistic results for the three-tier DCN architecture. Presumably, it is the very first comparative study of DCN architectures employing implementation and simulation techniques.

A simple simulation analysis introduced in this paper allows us to compare the behavior and performance of the considered DCN architectures under different workload and network conditions. The DCN architectures used in the analysis [9,12] have been implemented on a very small-scale system, with 20 servers in the case of DCell model [12] and 10 machines in the fat-tree model [9]. The simulation analysis may be considered as a general test bed for realistic networks with large number of hosts and various communication and traffic patterns. The analysis may also be used for the 'green data centers' for designing energy-efficient communication protocols in DCN architectures [20–26].

2. STATE-OF-THE-ART

Data center network architecture is one of the most significant components of large-scale data centers, which wields a great impact on the general data center performance and throughput. Numerous empirical and simulation analysis show that almost 70% of network communication takes place within a data center [27]. The cost of the implementation of the conventional two-tier and three-tier-like DCN architectures is usually too high and makes the models virtually ineffective in the large-scale dynamic environments [7]. Over the last few years, the fat-tree based and the recursively defined architectures are presented as the most promising core structures of the modern scalable data centers. On the basis of the different types of the traffic routing models, the DCN architectures can be classified into the following three basic categories: (a) switch-centric models [9,14], (b) hybrid models (using server and switch for packet forwarding [12, 13]), and (c) server-centric models [18].

The switch-centric DCN architectures rely on the network switches to perform routing and communication in the network, such as three-tier architecture and the fat-tree based architecture [9]. Hybrid architectures use a combination of switches and servers that usually are configured as routers within the network to accomplish routing and communication, such as DCell [12]. The server-centric architectures do not use switches or routers. The basic components of such models are servers that are configured as computational devices and data and message processing devices.

The legacy three-tier architecture is by far the most extensively used DCN architecture [8]. In the three-tier architecture, the switches are primarily arranged in three layers: (a) access, (b) aggregate, and (c) core (Figure 1). The pool of servers is thereby connected to access layer switches. The core layer makes the foundation of the network tree, and each core layer switch is connected successively to all of the aggregate layer switches. High-end enterprise switches are usually used at aggregation and core layers, rendering three-tier DCN an excessively expensive and power hungry architecture [6,9]. Different layers of three-tier architecture are oversubscribed at different threshold values. Variation in the oversubscription ratio at the various network layers is based on the physical infrastructure. The oversubscription is defined for optimizing the cost of the system design. Oversubscription can be calculated as a ratio of worst-case aggregated bandwidth available to end hosts and the overall bisection bandwidth of the network topology [9]. For instance, the oversubscription 4:1 means that the communication pattern may use only 25% of the available bandwidth. The typical oversubscription values are between 2.5:1 and 8:1, and 1:80 to 1:240 for the paths near the root at highest level of system hierarchy [9,14].

The basic model of the fat-tree DCN architecture has been proposed by Al-Fares *et al.* [9]. The fat-tree model is promoted by the authors as an effective DCN architecture by using a set of commodity switches to provide more end-to-end bandwidth at a considerably lower monetary cost and energy consumption as compared with the high-end network switches. The proposed solution is backward

DOI: 10.1002/cpe

compatible, and only requires modification in the switch forwarding functions. The fat-tree based DCN architecture aims to provide 1:1 oversubscription ratio.

Al-Fares *et al*. [9] adopted a special topology called fat-tree topology [28]. The network structure is composed of n pods. Each pod contains n servers and n switches organized in two successive layers of $n/2$ switches. Every lower layer switch is connected to $n/2$ hosts in the pod and $n/2$ upper layer switches (making the aggregation layer) of the pod. There are $(n/2)^2$ core level switches, each connecting to one aggregation layer switch in each of n pods. The exemplary interconnection of servers and switches for $n = 4$ pods is presented in Figure 2.

The fat-tree based DCN architecture [9] uses a customized routing protocol, which is based on primary prefix and secondary suffix lookup for next hop. Routing table is divided into two levels. For each incoming packet, destination address prefix entries are matched in primary table. If longest prefix match is found, then the packet is destined to the specified port. If there is no match, then the secondary level table is used, and the port entry with longest suffix match is used to forward the packet.

A recursively defined DCN architecture, referred to as the *DCell* model was reported in [12]. In this model, the whole system is organized in the cells or pods with n servers and a commodity switch. A 0 level cell $DCell_0$ serves as the building block of the whole system. A $DCell_0$ comprises of n commodity servers and a mini network switch. Higher levels of cells are built by connecting multiple lower level ($level_{l\text{-}1}$) *DCells*. Each $DCell_{l\text{-}1}$ is connected to all of the other $DCell_{l\text{-}1}$ within the same $DCell_l$. The DCell provides an extremely scalable architecture. A 4 level DCell, having six servers in $DCell_0$ can accommodate around 3.26 million servers. Figure 3 shows a *level 2* DCell having two servers within each $DCell_0$. The figure shows the connection of only $DCell_{1[0]}$ to all other $DCell_1$.

Unlike the conventional switch-based routing used in the hierarchical and fat-tree based DCN architectures, the DCell uses a hybrid routing and data processing protocol. Switches are used to communicate among the servers within the same $DCell_0$. The communication with servers in other *DCells* is performed by servers acting as routers. In fact, computational servers are also considered as the routers in the system. The DCell routing scheme is used in the DCell architecture to compute the path from the source to destination node exploiting divide and conquer approach. Source node (s) computes the path from s to destination (d). The link that interconnects the DCells that contain the s and d at the same level is calculated first, and then the sub-paths from s to link and from link to d are calculated. Combination of both of the sub-paths results in the complete routing path between s and d. The DCell routing is not a minimum hop routing scheme. Therefore, the calculated route possesses more hops than the shortest path routing.

Zhang *et al*. [29] compared the performance of two DCN architectures for the three-tier transection system in a virtualized environment at a low scale. The authors mainly focused on the following:

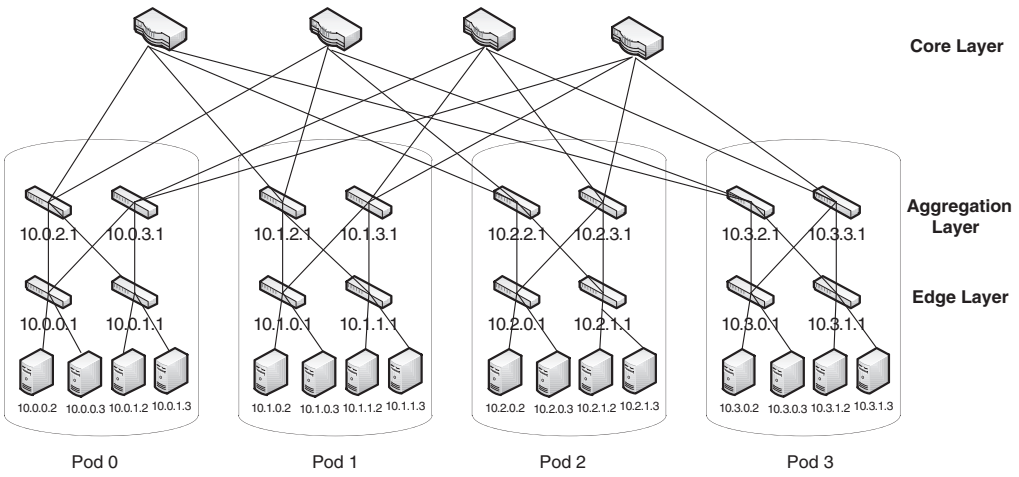

Figure 2. Fat-tree based architecture.

DOI: 10.1002/cpe

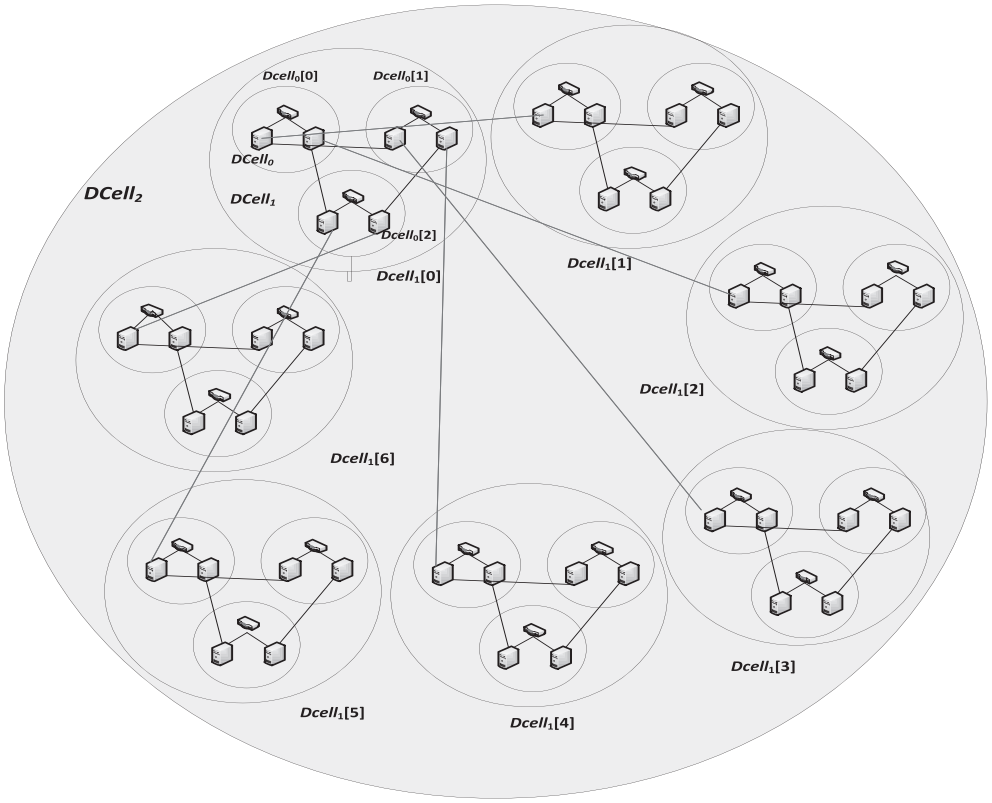

Figure 3. A level 2 DCell (DCell2).

(a) service fragmentation and (b) failure resilience. The authors implemented the fat-tree and FiConn [15] DCN architectures employing 12 nodes setup. The results illustrated that fat-tree outperforms FiConn architecture in terms of failure resilience and service fragmentation. Popa *et al*. [30] presented a methodology of the theoretical approximation of cost of different DCN architectures by using the system performance metrics, namely, network latency and capacity. The authors also presented, in an overwhelmingly sound manner, a cost comparison of different DCN architectures by using current market price of energy and equipment. Gyarmati *et al*. [31] compared the energy consumption in different DCN architectures. The authors have derived the results from mathematical analysis by considering the number of servers, total number of ports, and switches. They considered the static predefined measurement of energy consumption for devices. Chen *et al*. [11] have surveyed the routing protocols used in the major DCN architecture models and have addressed some open questions and security issues in DCN routing. Implementation of DCN architectures would be discussed in the next section.

3. SIMULATIONS AND COMPARATIVE STUDY

3.1. Environment

The main goal of the empirical simulation analysis presented in this section is to provide a comprehensive insight of different DCN architectures in a realistic manner. Three DCN core architectural models, namely: (a) the legacy three-tier architecture [8], (b) fat-tree based architecture [9], and (c) recursively build DCell architecture [12], have been used for the simulation of the multilevel DCN performance. We used ns-3 discrete-event network simulator for implementing the considered DCN architectures [32]. The ns-3 simulator allows modeling of various realistic scenarios. The most important salient

DOI: 10.1002/cpe

features of the ns-3 simulator are the following: (a) implementation of real Internet Protocol (IP) addresses, (b) Berkeley Software Distribution socket interface, (c) multiple installations of interfaces on a single node, (d) real network bytes contained in simulated packets, and (e) packet traces can be captured and analyzed using tools such as Wireshark. In this work, the DCN architectures uses the following: (a) the customized addressing scheme and (b) the customized routing protocols that strongly depend on the applied addressing scheme (e.g. [9]). Therefore, ns-3 deemed as the most appropriate network simulator for our work. One of the major drawbacks of using the ns-3 simulator is a lack of the network switch module in the ns-3 library. Moreover, the conventional Ethernet protocol cannot be implemented in ns-3. Therefore, we configured point-to-point links for the connection of switches and nodes. Moreover, we also implemented customized routing protocols for the DCN architectures in ns-3. All of our implementation will be made publically available for researchers and students.

3.2. Implementation details

The considered DCN architectures have been implemented by using the multiple network interfaces at each node as required. We implemented the three-tier architecture with an oversubscription ratio of 4:1 at the access layer and 1.5:1 at the aggregate layer. We used the interconnection architecture for three-tier architecture as reported in [7, 8] and used ECMP routing for enhanced performance, as available in the high-end switches. In the case of fat-tree based topology, the primary and secondary routing tables are generated dynamically and are based on the number of pods. The realistic IP addresses have been assigned to all of the nodes within the system and linked to appropriate lower layer switches. Three layers of switches have been created, interconnected, and populated with primary and secondary routing tables. We have tailored the general simulator model by extending it with an additional routing module for processing two-layered-based primary and secondary routing tables in ns-3. A simulation representation of 8-pod fat-tree is shown in Figure 4.

In the DCell architecture, the DCell routing protocol is implemented to generate the end-to-end path at the source node. We have specified a scalable addressing protocol for this model. The DCell routing lacks the generic protocol description, and a specific routing scenario is discussed by authors. Moreover, the DCell routing does not take the IP addressing scheme into consideration. We generalized and

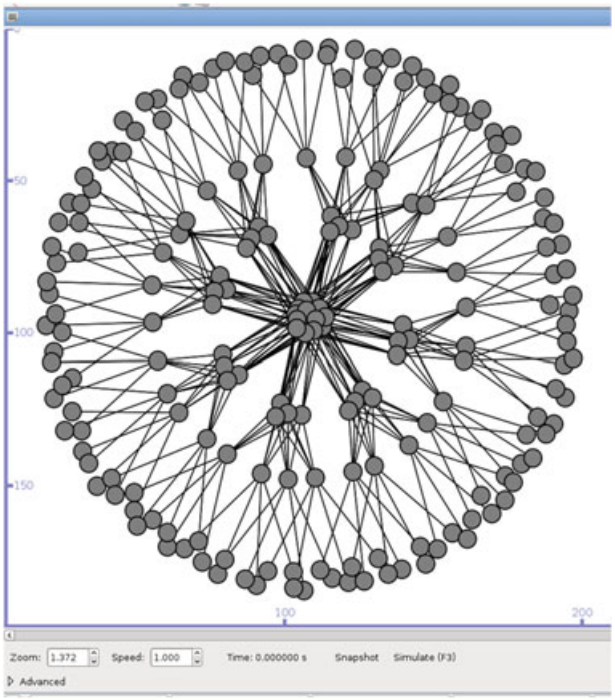

Figure 4. 8-pod fat-tree in ns-3 simulation.

DOI: 10.1002/cpe

implemented the routing protocol, which is now fully compatible with the IP. We implemented the source-based routing procedure to route the packets from the source to destination using the IP.

We found some important details missing in the DCell routing protocol presented in Section 4.1 of [12]. In the function *GetLink*, the authors state that if ($s_{k\text{-}m} < d_{k\text{-}m}$), then the link that interconnects both of the sub-DCells can be found as '($[s_{k\text{-}m}, d_{k\text{-}m} - 1]$, $[d_{k\text{-}m}, s_{k\text{-}m}]$)'. The 'else clause' for the aforementioned 'if statement' is missing, which makes the routing algorithm incomplete and erroneous. We formulated the missing 'else clause' to complete the algorithm. That is to say that if ($s_{k\text{-}m} \geq d_{k\text{-}m}$), then the interconnection link can be found as '($[s_{k\text{-}m}, d_{k\text{-}m}]$, $[d_{k\text{-}m}, s_{k\text{-}m} - 1]$)'. Moreover, the intermediate path between nodes 021 and 121, presented in Section 4.1 (Theorem 4) of [12] has a typographical error that may mislead and confuse readers. The underlined node within the path ([0,2,1], [0,2,0], [1,0,0], [0,0,0], [1,0,0], [1,0,1], [1,2,0],[1,2,1]) should be [0,0,1] instead of [1,0,0]. For reference, a simulation representation of three *level$_3$ DCells* is shown in Figure 5.

3.3. *Traffic patterns*

Benson *et al*. observed an on–off network traffic behavior within data centers. The network traffic logs collected at 19 various data centers provided evidence of the on–off network traffic and short-lived traffic bursts [33]. To generalize our simulation results, we used six different network traffic patterns to evaluate the DCN architectures for one-to-one, one-to-many, and all-to-all communications, namely: (a) uniform random, (b) exponential random, (c) one-to-one for 1 s (*1-1-1*), (d) one-to-one for random time interval (*1-1-R*), (e) one-to-many for 1 s (*1-M-1*), and (f) one-to-many for random time interval (*1-M-R*).

In the uniform random and exponential random traffic generation scenarios, every node within the data center communicates with some other arbitrarily chosen node. Inter-node communication occurs at random time intervals following the uniform random distribution and exponential random distribution, respectively. In the *1-1-1* traffic generation pattern, every node within the network communicates with some other randomly chosen node for an on period of 1 s. That is to say that the sender nodes send the data at a constant bit rate (CBR) for flow duration of 1 s. For the *1-1-R* traffic, the sender nodes send the CBR data in an on period for a randomly chosen time interval between 0.1 and 5.0 s, followed by an off period of random time interval.

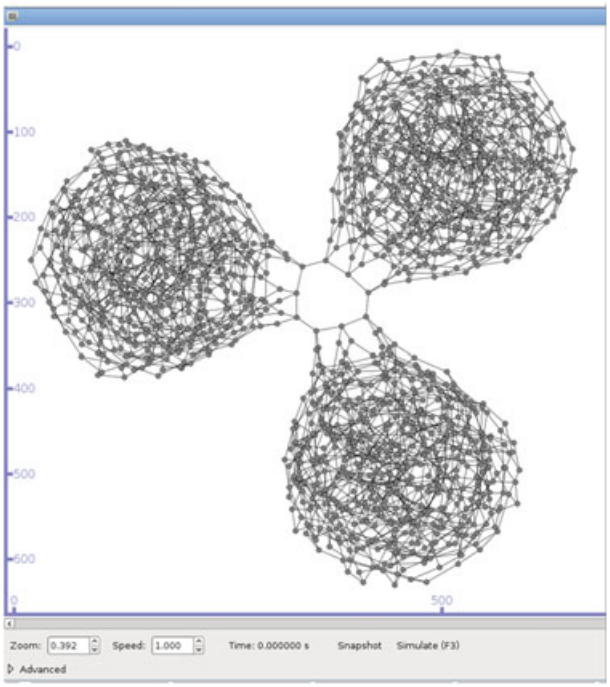

Figure 5. Three *DCell$_3$* ns-3 simulation.

DOI: 10.1002/cpe

In the *1-M-1* traffic generation scenario, a single node communicates with n other arbitrarily chosen nodes for an on period of 1-s duration. The value for n is also chosen at random from a range of [1–10]. In the *1-M-R* scenario, a single node communicates with n other nodes for an on period of randomly chosen duration. In one-to-many network scenarios, the number of sender nodes is around 1/8 of the network size.

We simulated the aforementioned four traffic generation scenarios with two different data rates for the CBR communication, namely: (a) 1 Mbps and (b) 10 Mbps. In the 10-Mbps data rate, each sender sends 10-Mb data to the receiver within a 1-s time slot. Similar analogy will hold for the 1-Mbps data rate.

3.4. Comparative analysis

We have simulated all of the DCN architectures under the six scenarios discussed in Section 3.3. The performances of the considered architectural models have been verified by using the following criteria:

a. *Average packet delay*: Average packet delay in the network is calculated using Eq. (2).

$$D_{\text{agg}} = \sum_{j=1}^{n} d^j, \tag{1}$$

$$D_{\text{avg}} = \frac{D_{\text{agg}}}{n}, \tag{2}$$

where D_{agg} calculated in Eq. (1) is the aggregate delay of all of the received packets, d_j is the delay of packet j, n is the total number of the packets received in the network, whereas D_{avg} is the average packet delay.

b. *Average network throughput*: Average network throughput is calculated using Eq. (3).

$$\frac{\tau = \left(\sum_{i=1}^{n} (P_i) \times \delta\right)}{D_{\text{agg}}}, \tag{3}$$

where τ is the throughput, P_i is the ith received packet, δ is the size of the packet (in bits), and D_{agg} is the aggregate packets delay.

The parameters used in the simulation of the fat-tree, DCell, and three-tier DCN are documented in Tables I–III, respectively. Simulations were performed by varying the aforementioned parameters under six different traffic scenarios to achieve results in respective topologies. Network topologies with different number of nodes ranging from 16 to 4096 nodes were created for the respective DCN architectures for every traffic pattern. Around 74 different simulation scenarios were created for each of the DCN architecture, resulting in 222 different configurations. The simulation results for the network throughput and packet delay are shown in Figures 6–14. The FAT, DCell, and 3T in the chart legend represent fat-tree, DCell, and three-tier DCN architectures, respectively.

The simulation results depict a steady behavior for DCN architectures under various traffic patterns and data rates. Because the network throughput is inversely proportional to average packet delay,

Table I. Simulation parameters for the fat-tree.

Number of pods	4–72
Number of nodes	16–93,312
Simulation running time	10–1,000 s
Packet size	1,024 bytes
Routing algorithm	Two-level routing protocol

DOI: 10.1002/cpe

Table II. Simulation parameters for the DCell.

Number of levels	0–3
Number of nodes in $DCell_0$	2–8
Total nodes in the DCell	16–100,000
Simulation running time	10–1,000 s
Packet size	1,024 bytes
Routing algorithm	DCellRouting

Table III. Simulation parameters for the three-tier data center network architecture.

Number of modules	4–170
Nodes connected with each access layer switch	8
Oversubscription ratio at access layer	4:1
Oversubscription ratio at aggregate layer	1.5:1
Simulation running time	10–1,000 s
Packet size	1,024 bytes
Routing algorithm	ECMP global routing

ECMP, Equal Cost Multi-Path.

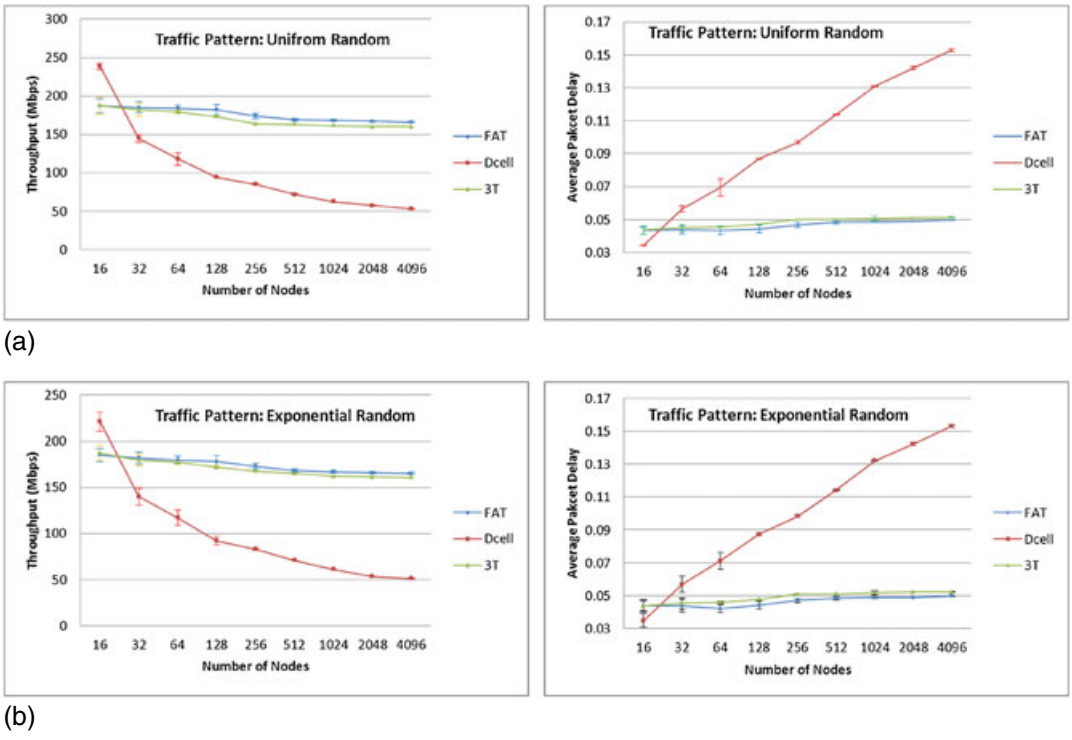

Figure 6. (a) Throughput (left) and average packet delay (right) using uniform random traffic distribution; and (b) throughput (left) and average packet delay (right) using exponential random traffic distribution.

large packet delays result in small throughput. The throughput and average packet delay for uniform random and exponential random traffic distribution is shown in Figure 6. Figures 7–10 report the results for *1-1-1*, *1-1-R*, *1-M-1*, and *1-M-R* traffic patterns for a data rate of 1 Mbps, respectively. Figures 11–14 show the results for 10 Mbps.

It can be observed in Figures 6–14 that the fat-tree DCN architecture outperforms the DCell and three-tier architecture in terms of throughput and packet delay. The three-tier architecture performance is almost equivalent to that of the fat-tree architecture with a very little difference in

DOI: 10.1002/cpe

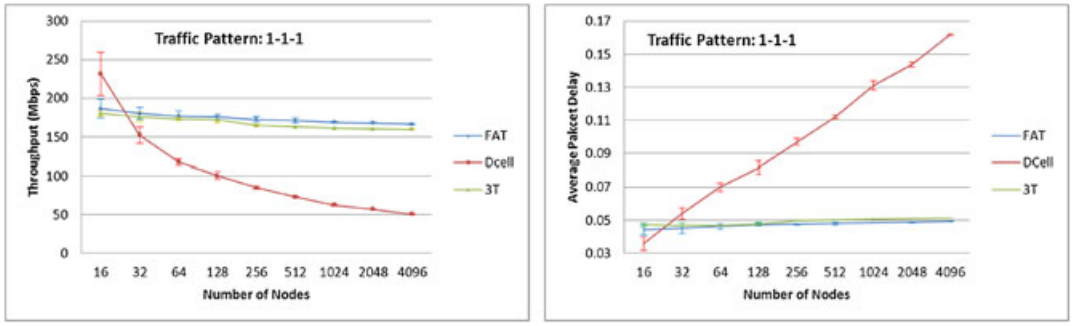

Figure 7. Throughput (left) and average packet delay (right) for 1-1-1 traffic pattern with 1-Mbps data rate.

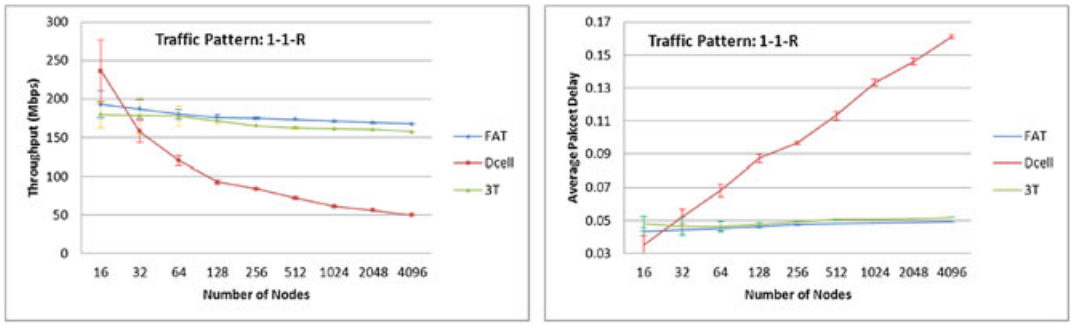

Figure 8. Throughput (left) and average packet delay (right) for 1-1-R traffic pattern with 1-Mbps data rate.

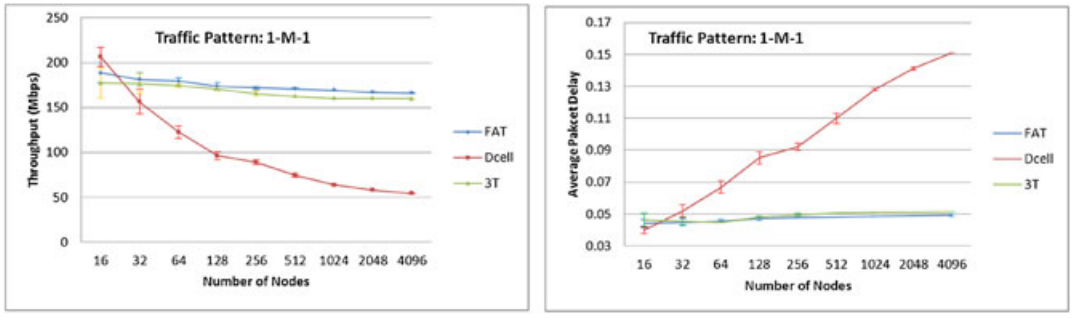

Figure 9. Throughput (left) and average packet delay (right) for 1-M-1 traffic pattern with 1-Mbps data rate.

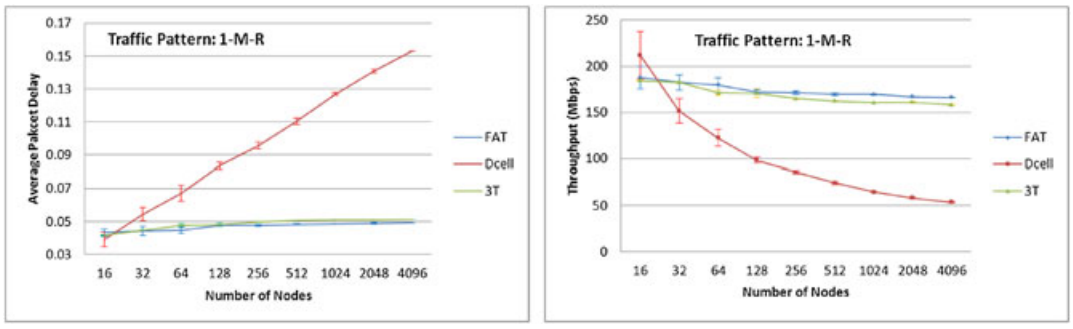

Figure 10. Throughput (left) and average packet delay (right) for 1-M-R with 1-Mbps data rate.

DOI: 10.1002/cpe

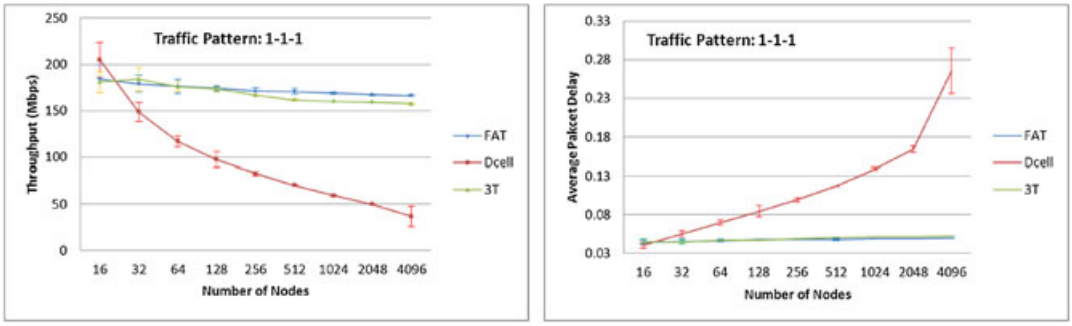

Figure 11. Throughput and average packet delay for 1-1-1 traffic pattern with 10-Mbps data rate.

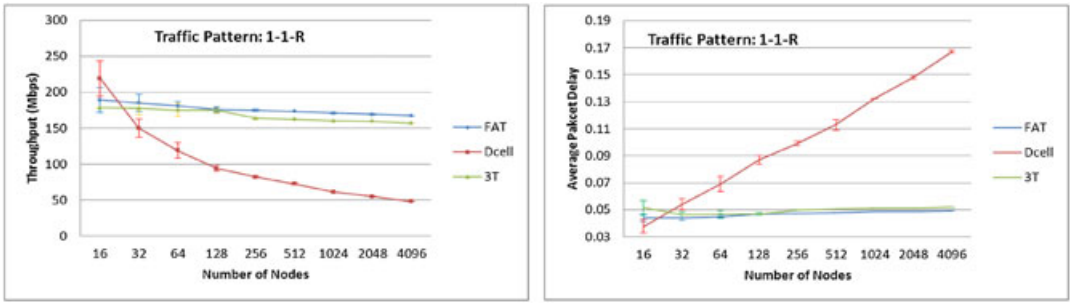

Figure 12. Throughput (left) and average packet delay (right) for 1-1-R traffic pattern with 10-Mbps data rate.

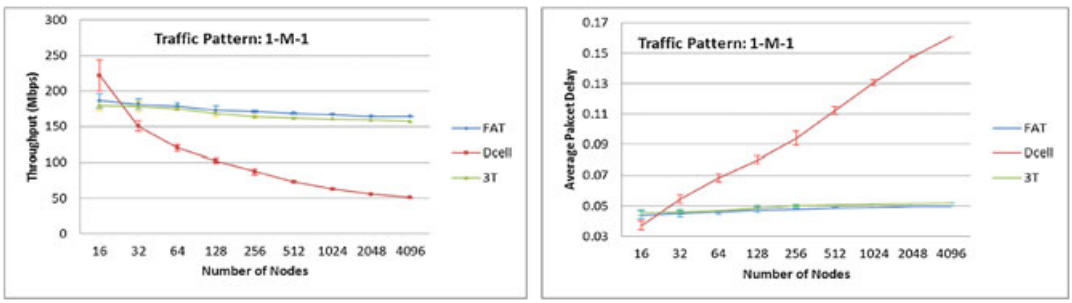

Figure 13. Throughput (left) and average packet delay (right) for 1-M-1 traffic pattern with 10-Mbps data rate.

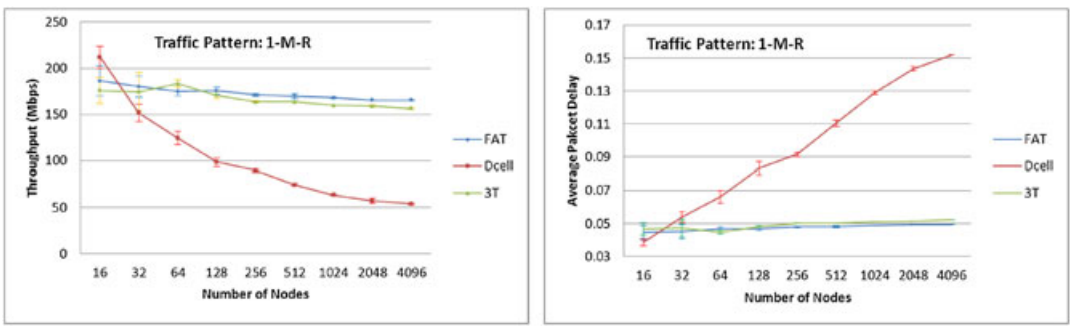

Figure 14. Throughput (left) and average packet delay (right) for 1-M-R traffic pattern with 10-Mbps data rate.

DOI: 10.1002/cpe

the average throughput. The DCell architecture outperforms the fat-tree and three-tier architecture for small network topologies but as the number of nodes within the network is increased, the DCell architecture experiences degradation in the network throughput and exhibits increased average packet delay.

The reason for the steady performance of the fat-tree architecture is the inherent network topology. A large number of network switches are structured in such a way so as to provide more end-to-end bandwidth for better and steady-state performance.

Although the performance of three-tier architecture seems similar to that of the fat-tree architecture, the performance is achieved at a much higher cost. Some important aspects for the better performance of three-tier architecture are the following: (a) The three-tier architecture uses costly high-end network equipment at the higher layer; (b) the ECMP routing also contributes to the better performance; and (c) the oversubscription ratio of 4:1 and 1.5:1 at the access layer and aggregation layer, respectively. The actual oversubscription ratio may be much higher and may vary from a data center to a data center at the access and aggregation layers. The data provided in [33] depicts a great variety in oversubscription ratios at the different layers of the three-tier data centers.

The performance of the DCell architecture depicts a strong dependency on the network size. We illustrate this phenomenon through Figure 4. All of the *inter-DCell* network traffic must pass through the network link connecting the *DCells* at same level leading to increased network congestion, packet delay, and packet loss. Smaller network topologies experience larger throughput because the network traffic load on the *inter-DCell* link is low and the links serve lesser number of nodes. For larger topologies, such as in our case of the network with 4096 nodes, each link connecting two $DCell_3$ experience an oversubscription ratio of 256:1 that obviously decreases the throughput for larger networks. Another reason for the throughput degradation in the DCell is the number of intermediate hops between the sender and receiver. DCell routing is not a shortest path routing algorithm, and for large network topologies, the number of intermediate hops may be as large as $2^{k+1}-1$, without considering the switching in the $DCell_0$ as a hop [12]. Intermediate hops including the network switches in the $DCell_0$ as a hop may result in more than 20 intermediate hops.

The simulation results reveal that the performance of fat-tree DCN architecture is independent of the network size. Alternatively, performance of the DCell architecture is heavily dependent on the network size. The performance of the three-tier architecture is dependent on physical topology and oversubscription ratio at different network layers. We hope that our thorough investigation of the most commonly used data center architectures will spark further investigation in developing scalable data center architectures.

4. CONCLUSIONS

We presented a comparison of the major DCN architectures that addressed the issues of network scalability and oversubscription. We simulated the performance of the major DCN architectures in various realistic scenarios under different network configurations. The simulation results showed that the fat-tree based DCN architecture outperformed the DCell and three-tier DCN architectures in terms of average network throughput and packet delay. In the future, we plan to compare the DCell-customized routing scheme with the shortest path routing based procedures. Moreover, we are also interested in enhancing the routing schemes to make the DCN architectures 'green'. Furthermore, we are also interested in introducing the workload consolidation features within the routing schemes to make use of the idle and under-utilized links to save energy by utilizing the dynamic power management based techniques.

ACKNOWLEDGEMENTS

Samee U. Khan's work was partly supported by the Young International Scientist Fellowship of the Chinese Academy of Sciences, (Grant No. 2011Y2GA01). A shorter version [21] of this paper appeared in ECMS 2012.

REFERENCES

1. Carter A. 2007. Do it green: media interview with Michael Manos. http://edge.technet.com/Media/Doing-IT-Green/, accessed, Feb. 20, 2012.

DOI: 10.1002/cpe

2. Rabbe L. 2006. Powering the Yahoo! network. http://yodel.yahoo.com/2006/11/27/powering-the-yahoo-network/, accessed February 20, 2012.
3. Arnold S. 2007. Google Version 2.0: the Calculating Predator. Infonortics Ltd.
4. Ho T. 2007. Google architecture. http://highscalability.com/google-architecture, accessed February 20, 2012.
5. Snyder J. 2007. Microsoft: datacenter growth defies Moore's law. http://www.pcworld.com/article/id,130921/article.html, accessed February 20, 2012
6. Sengupta S. Cloud data center networks: technologies, trends, and challenges. *ACM SIGMETRICS Performance Evaluation Review* 2011; **39**(1):355–356.
7. Kliazovich D, Bouvry P, Khan SU. GreenCloud: a packet-level simulator of energy-aware cloud computing data centers. *The Journal of Supercomputing* Forthcoming.
8. Cisco Data Center Infrastructure 2.5 Design Guide. Cisco press, March 2010.
9. Al-Fares M, Loukissas A, Vahdat A. A scalable, commodity data center network architecture. In Proceedings of the ACM SIGCOMM 2008 conference on Data communication (Seattle, WA). 2008; 63–74.
10. Mysore RN, 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. In Proceedings of the ACM SIGCOMM 2009 conference (Barcelona, Spain). 2009; 39–50.
11. Chen K, Hu CC, Zhang X, Zheng K, Chen Y, Vasilakos AV. Survey on routing in data centers: insights and future directions. *IEEE Network* 2011; **25**(4):6–10.
12. Guo, C, Wu H, Tan K, Shi L, Zhang Y, Lu S. Dcell: a scalable and fault-tolerant network structure for data centers. *ACM SIGCOMM Computer Communication Review* 2008; **38**(4):75–86.
13. 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 Proceedings of the ACM SIGCOMM 2009 conference (Barcelona, Spain). 2009; 63–74.
14. Greenberg A, Hamilton JR, Jain N, Kandula S, Kim C, Lahiri P, Maltz D, Patel P, Sengupta S. VL2: a scalable and flexible data center network. In Proceedings of the ACM SIGCOMM 2009 conference (Barcelona, Spain). 2009; 51–62.
15. Li D, Guo C, Wu H, Tan K, Zhang Y, Lu S. FiConn: using backup port for server interconnection in data centers. In Proceedings of the IEEE INFOCOM. 2009; 2276–2285.
16. Wang G, David G, Kaminsky M, Papagiannaki K, Eugene T, Kozuch M, Ryan M. c-Through: part-time optics in data centers. In Proceedings of the ACM SIGCOMM 2010 conference (New Delhi, India). 2010; 327–338.
17. Farrington N, George P, Sivasankar R, Hajabdolali B, Vikram S, Yeshaiahu F, George P, Vahdat A. Helios: A hybrid electrical/optical switch architecture for modular data centers. In Proceedings of the ACM SIGCOMM 2010 conference (New Delhi, India). 2010; 339–350.
18. Abu-Libdeh H, Costa P, Rowstron A, O'Shea G, Donnelly A. Symbiotic routing in future data centers. In Proceedings of the ACM SIGCOMM 2010 conference (New Delhi, India). 2010; 51–62.
19. Hopps C. 2000. Analysis of an equal-cost multi-path algorithm. RFC 2992, Internet Engineering Task Force.
20. Bilal K, Khan SU, Kolodziej J, Zhang L, Hayat K, Madani SA, Min-Allah N, Wang L, Chen D. (Forthcoming). A survey on Green communications using Adaptive Link Rate. *Cluster Computing* 2012a. DOI: 10.1007/s10586-012-0225-8
21. Bilal K, Khan SU, Kolodziej J, Zhang L, Hayat K, Madani SA, Min-Allah N, Wang L, Chen D. A comparative study of data center network architectures. In *Proceedings of 26th European Conference on Modelling and Simulation*. Koblenz: Germany; 2012b.
22. Bianzino P, Chaudet C, Rossi D, Rougier J. A survey of green networking research. *Communications Surveys and Tutorials, IEEE* 2012; **14**(1):3–20.
23. Zeadally S, Khan SU, Chilamkurti N. Energy-efficient networking: past, present, and future. *The Journal of Supercomputing* 2012; **62**(3):1093–1118.
24. Khan SU, Zeadally S, Bouvry P, Chilamkurti N. Green networks. *The Journal of Supercomputing* Forthcoming-a.
25. Khan SU, Wang L, Yang L, Xia F. Green computing and communications. *The Journal of Supercomputing* Forthcoming-b.
26. Wang L, Khan SU. Review of performance metrics for green data centers: a taxonomy study. *Journal of Supercomputing* Forthcoming. DOI: 10.1007/s11227-011-0704-3
27. Mahadevan P, Sharma P, Banerjee S, Ranganathan P. Energy aware network operations. INFOCOM Workshops 2009, IEEE. 2009; 1–6.
28. Leiserson, CE. Fat-trees: universal networks for hardware-efficient supercomputing. *IEEE Transactions on Computers* 1985; **34**(10):892–901.
29. Zhang Y, Su A, Jiang G. Understanding data center network architectures in virtualized environments: a view from multi-tier applications. *Computer Networks* 2011; **55**(9):2196–2208.
30. Popa L, Ratnasamy S, Iannaccone G, Krishnamurthy A, Stoica I. A cost comparison of datacenter network architectures. In Proceedings of the 6th International Conference (Philadelphia, Pennsylvania). 2010; 1–16.
31. Gyarmati L, Trinh T. How can architecture help to reduce energy consumption in data center networking? In Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking (Passau, Germany). 2010; 183–186.
32. ns-3. 2012. http://www.nsnam.org/, accessed February 21, 2012.
33. Benson T, Anand A, Akella A, Zhang M. Understanding data center traffic characteristics. *SIGCOMM Computer Communication Review* 2010; **40**1:92–99.

DOI: 10.1002/cpe