A Novel Approach to Adaptive Flow Scheduling for Energy Efficient Data Center Network

Ranjana R [1], Radha S [2], Raja J [3]

[1] Department of Information Technology, Sri Sai Ram Engineering College, Chennai
[2] Department of Electronics and Communication Engineering, SSN College of Engineering, Chennai
[3] Department of Electronics and Communication Engineering, Sri Sai Ram Engineering College, Chennai
ranjana.g@gmail.com, radhas@ssn.edu.in, jraja.ece@sairam.edu.in

Abstract

Virtualization and cloud computing have changed the computing paradigm and offers everything as service. The backbone of cloud computing is the massive data center servers. The major concern with the cloud data center is the massive amount of energy they consume. The data center network contributes to major share in energy consumption. So approaches to design energy efficient data center networking techniques are gaining importance. This work aims to conserve energy in a Spine leaf data center network. The idea of the proposed Adaptive flow scheduling technique (AFS) is to constantly monitor the traffic characteristics of the data center and accordingly activate or deactivate the spine switches. An adaptive threshold is used to identify the minimum utilized switches and a deadline based threshold is used to identify the switches that have flows with minimum completion time. The identified switches are put to sleep mode. The proposed work is evaluated using the Cloudsim toolkit with traffic patterns that model the real data center traffic. The simulation results effectively demonstrate that the proposed system consumes 15% less energy as against the baseline data centers with static set of active switches and state of art technique.

Keywords: Energy efficient data center, Flow scheduling, Adaptive scheduling

1 Introduction

Data center is a repository of massive computing and storage nodes interconnected through a network to provide wide range of services. Cloud computing is a computing model that provides these services on a pay per use basis using the datacenter facility. The successful adoption of cloud computing model fosters the rapid growth of data centers which are self sufficient with their own power supply equipments, networks and cooling equipments, thereby resulting in massive energy consumption. It is observed that data centers consume energy that is equivalent to nearly 25000 households [1].

The data center resources like the servers and network devices, account for about 40% of the total energy consumption of the data center [2-3]. Energy efficient methods for resource management can help to reduce the energy consumption by using minimum number of resources efficiently. The major areas where in energy efficiency can be achieved in a datacenter is the management of physical servers and network resources.

Several methods such as Dynamic Voltage Frequency Scaling (DVFS), server consolidation and energy efficient virtual machine placement have been explored for data center server energy reduction. However it is also studied that the network devices too consumes 10-20% of total power [4] making it a prospective area for energy conservation. Practically data centers are provisioned to meet their peak load demands in order to ensure high availability. So there is always a huge difference between the available resources and the utilised resources. Major energy savings could be gained if this difference is managed efficiently in a data center network.

Data center switches is one of the major component of a data center network. As the number of users and the services provided by the cloud increases, the traffic volume inside the data center network also rapidly increases. Figure 1, gives an estimate of the growth of data center traffic over the years. It has been shown in several studies that [5], the energy savings could be improved if it is possible to use network components according to the traffic flow volume of the data center.

Another study [6] finds that network elements contribute to nearly 33% of the power consumption of the data center equipments. Therefore, techniques focusing on data center network energy reduction must think of minimizing the active number of network equipments. Traditional data center architecture normally has switches [7] at three different layers namely core, aggregate and access. Another study by [8], confirms that the energy consumption in the

*Corresponding Author: Ranjana R; E-mail: ranjana.g@gmail.com
DOI: 10.3966/160792642019072004008

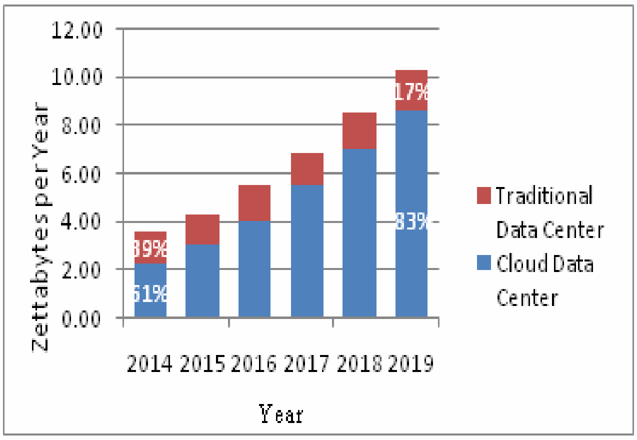

Figure 1. Worldwide data center traffic growth (data source: Cisco) [3]

aggregate layer alone is 57% of the energy consumption of the data center.

Therefore it is obvious that energy conservation techniques need also to concentrate on reduction of power by network equipments specifically the switches at the aggregate layer. Given the importance of data center network energy conservation and that aggregate switches are the major consumers of energy in DCN, the proposed work aims at reducing the available number of active spine switches in a spine leaf topology. The major contribution of this paper are as follows:

1. An adaptive flow management technique that uses flow scheduling technique that considers flow migration.
2. Adaptive switch management technique that reduces the number of spine switches that are active.
3. Adaptive threshold for selecting the list of spine switches that can be deactivated based on switch utilization.
4. Adaptive threshold for selecting the spine switches to be deactivated based on the flow completion
5. Performance evaluation of the proposed technique through simulation for different types of traffic flows.

The remaining part of the paper is organized as follows. Section 2 discusses the related work. In section 3 the system model and problem definition is presented, section 4 elaborates on the design of the proposed system, section 5 explains the simulation setup, results are discussed in section 6 and the paper is concluded in section 7.

2 Related Work

The major areas of research interest in the energy conservation of data center is minimizing the energy consumed by the network components. Few works like [9] have considered the network latency among communicating virtual machines in cloud data centers.

In [10] an optimal power conservation strategy namely PowerNets is proposed. This work considers the workload correlation in a data center and optimizes the energy consumption of servers and the network in the data center. The authors claim that more energy could be saved if workload correlations are accounted, because the server load and the network load do not peak at the same time. [11] studies a time aware VM flow placement that calculates the utilization of servers and switches and turns off those with low utilization ratios to conserve energy. It consolidates the network links so that few switches are active in the data center network. CARPO [12] proposes a power optimization algorithm for data center network, it takes into consideration the correlation of traffic flows. Claiming that traffic flows seldom peak at the same time, it tries to consolidate flows onto fewer set of links and switches. The other devices are shut down. However, the method used to identify the correlation becomes important because of the highly dynamic nature of the data center traffic.

In [13] the energy incurred during the VM migration is also considered for energy optimization. Application requests are modeled using a request graph. The paper proposes to consolidate both VM and switches and models as a joint optimization problem rather than individual problems. It also proposes to denote communications between VMs as a general graph. However, the significance of such a representation and its impact on the performance is not justified.

[14] Proposes a novel switch management technique that targets the aggregate layer and reduces the active Spine switches. This tends to conserve energy in Data center with Spine-leaf topology. The authors claim that the work is one of its first kind proposed for the spine leaf topology. The authors introduce a backup spine switch, to take care of the network traffic when the existing active switches are unable to handle peak demand. The additional energy consumption of this switch is not justified. Moreover the authors have proposed a static threshold for switch utilization, which may not be appropriate given the dynamic comportment of the data center. In [15], a distributed flow scheduling technique for data centers with Fat tree topology, is proposed. The major idea of this work is to present a decentralized control that allows switches to cooperate for link consolidations.

It provides global thresholds to balance the flows across data center networks. It requires the support of a modified switch logic which may not be feasible for all data centers. [16] proposes a mechanism called Adaptive trim tree, which aims at minimizing DCN energy consumption. It adopts a heuristic technique that partitions the DCN into subsets namely active and passive subsets. The active subsets are chosen in such a way so as to satisfy the traffic conditions and switch off the unused network components. The components in passive subset are subsequently put to sleep mode. The energy conservation is a tradeoff for latency and packet loss, which is claimed to be negligible.

[17] proposes FLOWP, a flow scheduling algorithm that saves energy by path and network equipment consolidation. However consolidation is based on static threshold, which is not suitable for the dynamic data center network. It can be observed that most of the data center network power management techniques depends on device consolidation based on static threshold. Few works depend on passive network device subset or a backup switch to handle additional traffic peaks during consolidation. The proposed AFS differs from the above mentioned works in a way that it uses adaptive threshold for switch consolidation and it allows flow migration based on the remaining duration of the traffic flow.

3 System Model and Problem Formulation

3.1 Spine Leaf Topology

There are basically several topologies in which the data center network can be interconnected. The study [18] elaborately surveys the different data center network architecture with their respective advantages and limitations. Among all these architectures, we take into consideration the Spine –Leaf architecture as per the specification by Cisco [19], which is shown in Figure 2. This topology uses the highly scalable data centers and it is made up of Spine Switches in the aggregate layer and the leaf switches at the access layer. The physical servers are connected to the leaf switch. Every spine switch connects to every leaf switch. That is there is a mesh between the Leaf layer and spine layer. In order to connect to the public network or outside the datacenter, the traffic is routed through the border switch which is similar to the core switch in a 3 tier DCN.

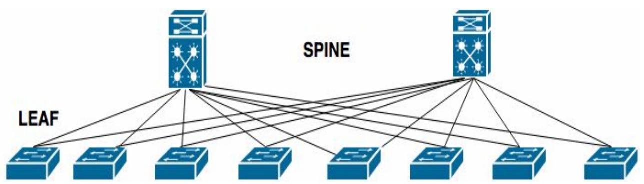

Figure 2. Spine Leaf topology

The advantages of Spine-leaf topology [14] can be summarized as under:

1. In this topology one Leaf connects to multiple Spine, so the traffic flow is not limited by the single link that connects the leaf layer and the spine layer.
2. As each spine is connected to every leaf, the delay inside the data center is predictable [14].
3. Because of the well connectedness, link failures are handled gracefully. If one spine switch failed, traffic can be easily distributed via other spine switches [19].
4. The topology is highly scalable, new servers and leaf switches can be added without any change to the existing network structure.

3.2 Data Center Network Model

A Spine leaf data center network, which is the interconnection of compute resources of a data center can be defined as DCN (PMList, VMList, SSList), as illustrated in table 1. User applications are deployed in virtual machines that are hosted in the physical machines within the data center. Based on the number of user requests or applications deployed, the required number of virtual machine also varies. At any instant, the number of active VMs in a data center is represented as VMList. The total number of physical machines available within the data center is represented as PMList. The notations representing the various resources used in data centers is listed in Table 1.

Table 1. Parameter Notations

Notations	Meaning
VMi	i^{th} Virtual machine
VMList	List of Virtual Machines
PM	i^{th} Physical Machine
PMList	List of Physical Machines
SSi	i^{th} spine switch in list
SSi, capacity	Capacity of the i^{th} Spine Switch
SSi, util	Utilisation of the i^{th} Spine Switch
SSi, count	Count of active flows allocated to the i^{th} Spine Switch
SSi, status	Status of the i^{th} Spine Switch, Active or sleep
SSList	List of Spine Switches
ASSlist <*SS*>	List of Active Spine Switch
DSSlist <*SS*>	List of Deactivated Spine Switch
Thi	Initial threshold for switch utilization
Tha	Current Adaptive Threshold for Spine Switch
FThi	Initial flow threshold
FTha	Adaptive threshold for flow threshold
Flow (*id*, *t*, *VMi*, *VMj*, *bw*, *d*)	Data transfer between VMi, VMj with bandwidth bw, initiated at time t and duration d
Flowi,status	Status of the flow at time t, could be allocated or deleted
A	Increase factor for adaptive threshold for switch consolidation
F	Increase factor for adaptive threshold for flow consolidation

3.3 Modeling the Characteristics of Data Center Traffic

The network traffic in a data center is elaborately studied in [20] and the traffic modeling to match the real time data traffic is given. The same modeling is adopted in this work so that the results obtained are similar to the evaluation using traffic that match the real time traffic.

3.4 Modeling Data Flow in Data Center Network

Data flow can be modeled as a data movement from a source VM to the destination VM. The data movement is created using a random distribution following On/ Off traffic pattern. The traffic patterns are adapted from [2]. Four types of data flow are taken for performance analysis of the algorithm based on the distance between the source node and destination node and the volume of data. For those data flows for which the source and destination connect to the same leaf switch, the pattern is termed as Near traffic flow. If the traffic flows between nodes that are connected to different leaf switches then it is called as Far Traffic. The flow volume is generated using a Pareto distribution with base volume as small or large. Small values are in the range from 3000Kbps to 15000 Kbps and large values range are in the range from 30Mbps to 180Mbps. So we have four types of flows.

1. Near traffic / Small
2. Near Traffic / Large
3. Far Traffic / Small
4. Far Traffic / Large

3.5 Modeling the Flow Duration

Real time analysis of data center network traffic [20] shows that 80% of the flows have duration that is not more than 11 seconds. The remaining flows have duration more than 200 seconds. This can be modeled as a Pareto distribution with shape parameter ap and scale parameter Mp with values 1.504 and 1.0001 respectively as given in [20].

3.6 Modeling Inter Arrival Time and the Flow Size

The Poisson process is used to model the inter arrival time between the flows in a data center network. As observed in [21], the data center traffic flows are not dependant on each other. The flow arrival rate is constant and the inter arrival time is exponentially distributed. The flow size is modeled using the Pareto distribution. The base bandwidth size is varied to simulate the dynamic traffic of the data center network. The base bandwidth ranges from 3000Kbps to 15000Kbps for small traffic and from 30Mbps to 180Mbps for large traffic.

4 Design of the Proposed System

The proposed AFS attempts to reduce the total active spine switches, thereby conserving the DCN energy consumption at the aggregate layer. The diagram of the proposed adaptive flow scheduling technique is illustrated in Figure 3. The functions of each component and the associated algorithms is explained in sections 4.1 to 4. 3

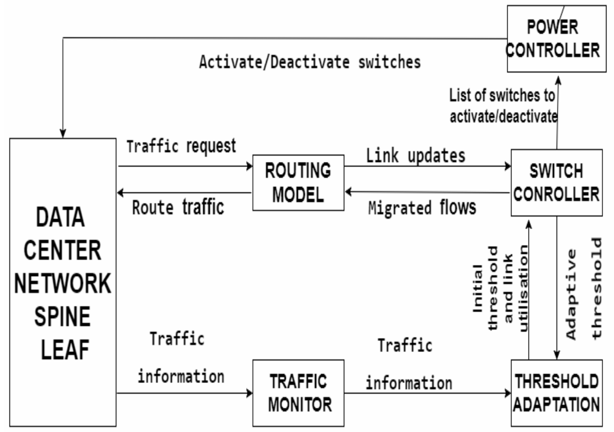

Figure 3. System design of the adaptive flow scheduling technique

4.1 Routing Module

The routing module accepts the traffic flow requests from the data center. The flow request can be categorized as new request or migrated flow request. The routing module takes care of routing the near and far traffic request across the Data center network based on Algorithm 1. It uses the Best Fit heuristic to select the switch with the required capacity to allocate the traffic request.

Algorithm 1. Flow Routing

```
Input: Flow (id, t, VMi, VMj, bw, d)
Output: Flow route
If(current time-Flow(t) > d)
{
Sort SSi in increasing order of ( SSi, cap – Ssi, util)
Iterate each SSi in ASSList do
        {
            If(SSi, util < Tha) && ((SSi, cap –
            SSi, util) > bw)
            {
            Allocate flow to SSi
            Allocated: = True
            Update SSi, util = SSi, util + bw
            }
Else
            Continue ;
}
If(Allocated:= False)
{
Activate SSi from DSSlist
Add SSi to ASSList
Allocate Flow_i to SS_i
Flowi, status = allocated
Update link utilization of SSi
}
Else
{
Flowi, status = deleted
Update link utilization of SSi
}
```

4.2 Switch Controller

The switch controller manages the spine switches. It maintains two lists namely, the spine switches that are active, and the list of deactivated spine switches. The switches are selected as per the algorithm 2 and will be activated or deactivated based on two threshold values. The adaptive threshold value for the switch is defined as Tha and that of the flow is defined as FTha. The initial threshold is decided by the switch controller and the threshold is adaptively changed based on the switch utilization. The adaptive threshold values are generated by the threshold adaptation module.

Algorithm 2. Switch controller

```
Input: ASSList, Tha, FTha
Output: updated ASSList, DSSList
Iterate each switch in ASSList
{
        If (SSi, util < Tha) && (SSi, Count < FTha)
        {
        migrate flows from SSi to switch in ASSList
        add SSi to DSSList
        set SSi, status = sleep
        }
}
```

4.3 Threshold Adaptation Module

The major function of this module is to provide the adaptive threshold for switch utilization Th and flow FTh based on the remaining duration. Let Thi be the initial threshold for any active switch SSi and FThi be the initial flow threshold. The input status represents whether a spine switch had been activated or deactivated during the previous observation period. During the previous observation period, the switch would have been in any one of the two states namely active, sleep or idle. The adaptive threshold Tha is generated by modifying the previous threshold with a factor A based on the cumulative switch utilization. FTha is the flow threshold generated based on the remaining time to complete for the active flows in any switch in ASSList. FTha is increased or decreased by a factor F. The increase and decrease factors are defined as in equations 1 to 2. The thresholds are dynamically generated based on the algorithm 3. Let it be assumed that T be total number of spine switches in the data center. At any point of time t, let us assume that A denotes the total number of active spine switches in the data center. The initial threshold for switch consolidation is Tha and is equal to 75 percent of the current utilization of the switches in the first monitoring period. Let C_i denote the cumulative current utilization in the i^{th} monitoring period. During the i+1 monitoring period, if utilization C_{i+1} of the switches is greater than the previous utilization C_i, then, Tha = Tha + A where A is defined in equation 1 Else Tha = Tha – A. If at time t = i, Tha = 80% and a new switch was activated in that period, then at time t = i+1, Tha = 80 – A, if new switch was not activated then Tha = Tha + A.

$$A = \frac{\Sigma SSi,\ util}{size\ (ASSList)} * \alpha \quad 0 \le \alpha \ge 1 \tag{1}$$

$$F = \frac{\Sigma SSi,\ count}{size\ (ASSList)} * \beta \quad 0 \le \beta \ge 1 \tag{2}$$

Algorithm 3. Adaptive Threshold

```
Input: initial threshold Thi, FThi, status
Output: Adaptive threshold Tha, FTha, SSi
If (status = =increase)
{
Tha = Thi – (Tha *A)
FTha = FThi – F
Update Thi to previous Tha
Update FThi to previous FTha
}
Else
{
Tha=Tha+A
FTha = FTha+F
Update Thi to previous Tha
Update FThi to previous FTha
}
```

5 Simulation Setup

We use the Cloudsim toolkit [22] for simulating the datacenter network with spine leaf topology, because it is difficult to experiment with test beds given the size and volume of the data center traffic [23]. Most of the existing studies [23-25] depend on customized simulators for performance evaluation of the data center networks. The experimental parameters were chosen as in [14]. Accordingly, the network topology consists of 16 leaf switches, 8 aggregate switches and 15 servers per leaf switch.

The applications that are submitted as VM requests are generated randomly starting from 240 VMs. Each VM is a traffic source. The traffic flow is generated with appropriate traffic arrival rate, size, inter arrival time and duration as explained in Section 2. The traffic is also categorized as low rate traffic or small traffic (3000, 5000, 10000 and 15000 kbps) and high rate traffic or large traffic (30, 80, 130, 180 Mbps) as in [11]. The categorization of traffic allows for observing the performance of the proposed AFS with a heterogeneous traffic. The total number of flows taken for simulation is 4099. If the source that generates the flow, and the destination that receives the flow are connected to the same leaf, then the flow is called a near flow. For far flow the source and destination are connected to different leaf switch. Meeting this constraint, the source and destination are chosen at random.

The results obtained are compared with a data center network that is not power aware. The parameters that are studied for analyzing the performance of the proposed algorithm are the total number of switches used and the total power consumed for a given set of flow requests.

6 Results and Discussions

The proposed system is evaluated based on three performance metrics adapted from [14], and are calculated according to the equations (3), (4) and (5). The three metrics are the average number of aggregate spine switch used (AASS), the total percentage of power consumed by spine switches (POAS) and the blocking probability (BP). The experimental results are compared with data center with static set of switches namely 6 and 8 denoted as NSS:6 and NSS:8. It is also compared with state of art technique GSSMS (Green Spine Switch Management). The difference between AFS and the above mentioned techniques are: Switch consolidation is not done in NSS:6 and NSS:8. However, switch consolidation with static threshold is studied in GSSMS but flow migration is not accounted. As reported in [14], it is assumed that 84% of power consumption is reduced if a switch is moved to its hibernation mode.

$$\text{AASS} = \frac{\Sigma \text{number of active spine switches x duration}}{\text{total duration}} \quad \textbf{(3)}$$

$$\text{BP} = \frac{\text{number of flows}}{\text{total number of flows}} \quad \textbf{(4)}$$

$$\text{POAS} = \frac{\text{AASS} + (\text{N} - \text{AASS}) * 0.16}{\text{N}} \times 100 \quad \textbf{(5)}$$

The average number of active switches when the AFS is evaluated with large flow is shown in Figure 4. The X axis represents the bandwidth of the flow in Mbps and the average number of switches that are active during the simulation run is given in the Y-axis. It is compared against the number of switches active in datacenter with 6 and 8 switches represented as NSS:6 and NSS:8 respectively. From the graph it is inferred that the proposed AFS technique results in usage of less number of switches than the fixed NSS:6 and NSS:8. However when compared to the state of art technique namely GSSMS(Green Spine Switch Management System) [14], the AASS linearly increases with that of the increase in bandwidth. However, for the maximum bandwidth of 180 Mbps, the difference in AASS for AFS and GSSMS is negligible. This is because GSSMS always keeps an active backup switch irrespective of the flow volume. So the minimum average of active spines is always 2. Since AFS migrates the traffic flows, the additional overhead of a backup switch is eliminated.

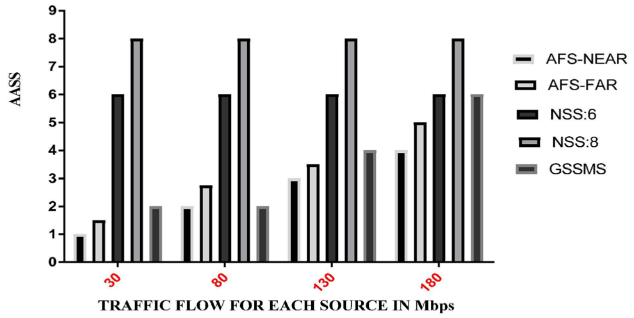

Figure 4. AASS vs. large flow bandwidth

Figure 5 represents the AASS values obtained when the bandwidth of the flow is small. The X-axis denotes the bandwidth in Kbps while the Y axis denotes the average number of active spine switches used. It is seen from the graph that the maximum spine switches used is only 1, proving the effectiveness of the proposed AFS. For the same scenario, GSSMS uses two switches, including the backup switch. This implies that during the time when the DCN is having low traffic, the proposed AFS manages with the minimum spine switches as against a DCN with fixed number of active switches (NSS:6, NSS:8) and GSSMS.

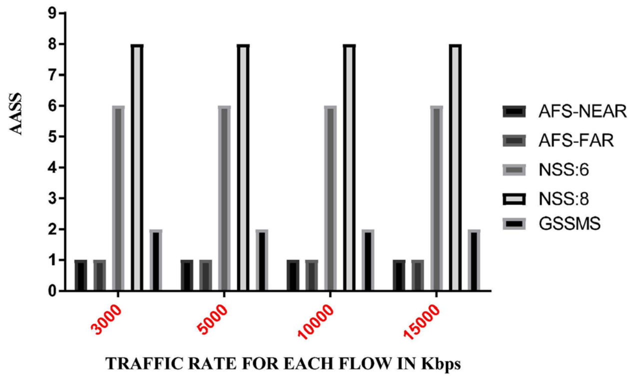

Figure 5. AASS vs. small flow bandwidth

Figure 6 and Figure 7 represent the power consumed when the proposed AFS is evaluated using large flow and small flow respectively. It is observed that the difference in POAS for AFS-Near, AFS-Far, GSSMS, is minimum when the flow bandwidth is 30Mbps. Otherwise it can be noted that there is a linear increase in the power consumption for both AFS-Near, AFS-FAR and GSSMS as the bandwidth tends to increase. This graph is not plotted against NSS:6 and NSS:8 because it is obvious that fixed number of switches without energy management will consume 100% of their rated power.

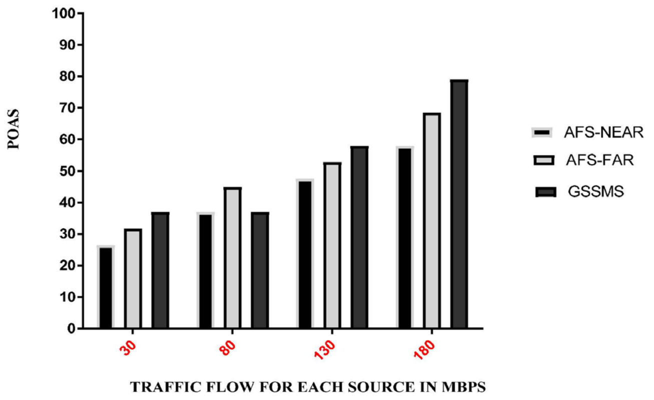

Figure 6. POAS vs. large flow bandwidth

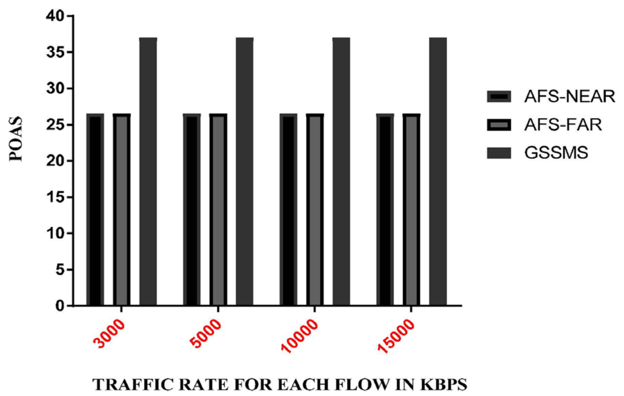

Figure 7. POAS vs. small flow bandwidth

Figure 8 represents the blocking probability for large flow. The flow bandwidth is denoted in the X-axis and the blocking probability is plotted in the Y axis. The blocking probability is studied only for large traffic with traffic volume ranging from 40000 Kbps to 90000 Kbps with a step size of 10000 Kbps. We have seen that for small traffic the number of active switches required is very minimum. The maximum flow for the small traffic at any instant could be accommodated in the available switches, so blocking is not present. So the blocking probability is studied for large traffic using the AFS-Near and AFS-Far techniques. From the Figure 8, it can be observed that the number of failed flows or the blocking probability increases with the increase in the flow size. It is also inferred from the graph that as the traffic volume increases, the difference in the blocking probability between the far and near traffic increases. This can be attributed to the fact that as the flow volume increases, the aggregate switch will reach its maximum capacity and hence cannot accommodate more near flows.

7 Conclusion

A novel adaptive flow scheduling technique (AFS) to conserve energy in data center network with spine leaf topology is presented. This technique uses two

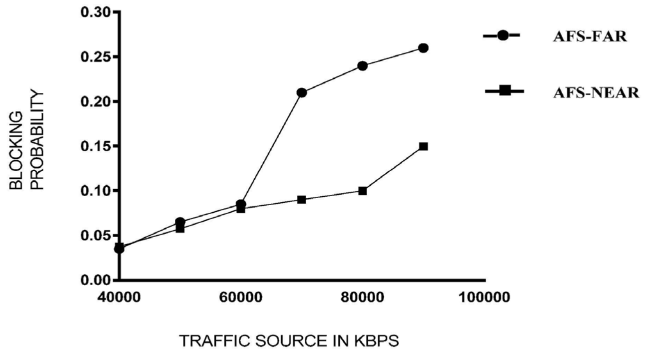

Figure 8. Blocking probability vs. large flow bandwidth

different adaptive thresholds to select the number of spine switches to be active in a data center network for a given traffic demand. The remaining switches are deactivated and put into sleep mode to be activated as required. The AFS is evaluated using traffic modeling based on real data center traffic characteristics based on three metrics namely active number of spine switches, energy consumed and blocking probability. The innovation factor is usage of adaptive threshold in flow scheduling and flow preemption based on adaptive threshold. The obtained results shows that 15% of energy is conserved when AFS is used as against the baseline non power efficient DCN and the state of art method GSSMS.

References

[1] J. Nie, J. Luo, L. Yin, Energy-aware Multidimensional Resource Allocation Algorithm in Cloud Data Center, *KSII Transactions On Internet and Information Systems*, Vol. 11, No. 9, pp. 4320-4333, September, 2017.

[2] A. Greenberg, J. Hamilton, D. A. Maltz, P. Patel, The Cost of a Cloud: Research Problems in Data Center Networks, *ACM SIGCOMM Computer Review*, New York, NY, 2009, pp. 68-73.

[3] M. H. Ferdaus, M. Murshed, R. N. Calheiros, R Buyya, An Algorithm for Network and Data-aware Placement of Multi-Tier Applications in Cloud Data Centers, *Journal of Networks and Computer Applications*, Vol. 98, pp. 65-83, June, 2017.

[4] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, N. McKeown, Elastictree: Saving energy in Data Center Networks, *Proceedings of the 7th USENIX conference on Networked systems design and implementation*, Berkeley, CA, 2010, pp. 17-31.

[5] P. Mahadevan, P. Sharma, S. Banerjee, P. Ranganathan, A Power Benchmarking Framework for Network Devices, in: L. Fratta, H. Schulzrinne, Y. Takahashi, O. Spaniol (Eds.), *NETWORKING 2009*, Springer, 2009, pp. 795-808.

[6] D. Kliazovich, P. Bouvry, Y. Audzevich, S. Khan, Greencloud: A Packet-level Simulator of Energy-aware Cloud Computing Data Centers, 2010 *IEEE Global Telecommunications Conference*, Trento, Italy, 2010, pp. 1-5.

[7] M. Al-Fares, A. Loukissas, A. Vahdat, A Scalable, Commodity Data Center Network Architecture, *Proceedings of ACM SIGCOMM Conference on Data Communications*, Seattle, WA, August 17-22, 2008, pp. 63-74.

[8] O. Popoola, B. Pranggono, On Energy Consumption of Switch-centric Data center Networks, *Journal of Super Computing*, Vol. No. 74, pp. 334-369, September, 2017.

[9] J. Wei, S. Wang, L. Zhang, A. Zhou, F. Yang, Virtual Machine Placement to Minimize Data Transmission Latency in MapReduce, *Journal of Internet Technology*, Vol. 18, No. 6, pp. 1379-1391, November, 2017.

[10] K. Zheng, X. Wang, L. Li, X. Wang, Joint Power Optimization of Data Center Network and Servers with Correlation Analysis, *Proceedings of IEEE INFOCOM*, Toronto, Canada, May, 2014, pp. 2598-2606.

[11] A. Dalvandi, M. Gurusamy, K. C. Chua, Time-aware Vmflow Placement, Routing, and Migration for Power Efficiency in Data Centers, *IEEE Transactions on Network and Service Management*, Vol. 12, No. 3, pp. 349-362, September, 2015.

[12] X. Wang, X. Wang, K. Zheng, Y. Yao, Q. Cao, Correlation-aware Traffic Consolidation for Power Optimization of Data Center Networks, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 27, No. 4, pp. 992-1006, April, 2015.

[13] L. Ma, H. Liu, Y. W. Leung, X. Chu, Joint VM-Switch Consolidation for Energy Efficiency in Data Centers, *2016 IEEE Global Communications Conference (GLOBECOM)*, Washington, DC, 2016, pp. 1-7.

[14] X. Li, C. Lung, S. Majumdar, Green Spine Switch Management for Datacenter Networks, *Journal of Cloud Computing: Advances*, Systems and Applications, Vol. 5, No. 9, December, 2016.

[15] W. Cui, Y. Yu, C. Qian, DiFS: Distributed Flow Scheduling for Adaptive Switching in FatTree Data Center Networks, *Journal of Computer Networks*, Vol. 105, pp. 166-179, August, 2016.

[16] S. Zafar, S. A. Chaudhry, S. Kiran, Adaptive TrimTree: Green Data Center Networks through Resource Consolidation, *Selective Connectedness and Energy Proportional Computing, Energies 2016*, Vol. 9, No. 10. p. 797, October, 2016.

[17] J. Luo, S. Zhang, L. Yin, Y. Guo, Dynamic Flow Scheduling for Power Optimization of Data Center Networks, *2017 Fifth International Conference on Advanced Cloud and Big Data (CBD)*, Shanghai, China, 2017, pp. 57-62.

[18] Y. Liu, J. Muppla, M. Veeraraghavan, D. Lin, M. Hamdi, *Data Center Networks*, Springer Briefs in Computer Science, 2013.

[19] P. Beck, P. Clemens, S. Freitas, J. Gatz, M. Girola, J. Gmitter, H. Mueller, R. O'Hanlon, V. Para, J. Robinson, A. Sholomon, J. Walker, J. Tate, IBM and Cisco: Together for a World Class Data Center, *IBM Redbooks*, 2013.

[20] S. Chandrasekaran, *Understanding Traffic Characteristics in a Server to Server Data Center Network*, MS Thesis, Rochester University of Technology, New York, 2017.

[21] Y. Han, J. Yoo, J. W. Hong, Poisson shot-noise Process Based Flow-Level Traffic Matrix Generation for Data Center Networks, *IEEE International Symposium on Integrated Network Management (IM)*, Ottawa, Canada, 2015, pp. 450-457.

[22] R. N. Calheiros, R. Ranjan, A Beloglazov, CAF De Rose and R Buyya, CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms, *Software: Practice and Experience*, Vol. 41, No. 1, pp. 23-50, 2011.

[23] M. Al-Fares, S. Radhakrishnan, B. Raghavan, N. Huang, A. Vahdat, Hedera: Dynamic Flow Scheduling for Data Center Networks, *Proceedings of USENIX NSDI*, San Jose, CA, 2010, pp. 19-19.

[24] S. Sen, D. Shue, S. Ihm, M. J. Freedman, Scalable, Optimal flow Routing in Datacenters via Local Link Balancing, *Proceedings of ACM CoNEXT*, 2013, Santa Barbara, CA, 2013, pp. 151-162.

[25] A. Singla, C.-Y. Hong, L. Popa, P. B. Godfrey, Jellyfish: Networking Data Centers Randomly, *Proceedings of USENIX NSDI*, San Jose, CA, 2012, pp. 225-238.

Biographies

Ranjana R., is working as Associate professor in Sri Sai Ram Engineering College and pursing research in Anna University, Chennai. She has 16 years of teaching experience. She has around 10 publications in conference and journals to her credit. Her area of interests include cloud Computing and computer networks.

Radha S., Professor & Head, Department of ECE, has 24 years of teaching and 15 years of research experience in the area of Wireless Networks. She has 62 publications in International and National Journals and 75 papers in International and National conferences to her credit. 13 scholars are graduated their Ph.D under her guidance and presently 8 research scholars, 1 Post-Doctoral Research Fellow, and 1 Senior Research Fellow are doing research.

Raja J., works as a Professor in the department of Electronics and communication Engineering, Sri Sai Ram Engineering College, Chennai. His has 3 decades of teaching experience. He has 36 journal publications and 50 conference papers to his credit. His areas of expertise includes Computer networks and digital communication.