International Journal of Engineering & Technology, 7 (2.6) (2018) 77-82

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Review Paper

Saving energy and improving performance in SDN using rate adaptation technique

Suchismita Rout [1] *, Sudhansu Sekhar Patra [2], Bibhudatta Sahoo [3]

[1] *School of Computer Engineering, KIIT University, Bhubaneswar, India*
[2] *School of Computer Application, KIIT University, Bhubaneswar, India*
[3] *Department of Computer Science and Engineering, NIT Rourkela, India*
**Corresponding author E-mail: suchismita.rout28@gmail.com*

Abstract

Energy efficiency in wired networks has received considerable attention over the past few years for its heavy economic impact. Energy consumption in such network is becoming a challenging issue. Among all these wired networks Software Defined Networking (SDN) is considered as an intelligent network in which use of programmability provides the best solution for network operation. The logical separation of data plane and control plane enhances its network functionality. Despite of all these mechanisms, energy consumption is also an alarming issue in SDN. Various researchers propose many approaches in order to utilize the energy in its best way. Herein we consider the normal scenario of a data center network, and propose a new energy efficient routing algorithms to improve the performance in SDN, called SERAT. In this work, we focus on discrete of data rate of the link in order to minimize energy consumption in the network. In wired network the underutilized links are even operated in high data rate, increasing energy consumption. In order to reduce energy consumption of the network the links must be operated as per their utilization factor. We have used an output buffer queue with two levels of threshold values in order to monitor the performance of the network. Simulation result shows that the proposed technique leads to major amount of energy savings without mush affecting the throughput.

Keywords: *Wired Network; SDN; Data Rate; Energy Efficiency; Flow Table; Incoming Traffic; Queue Buffering.*

1. Introduction

Software defined networking (SDN) has attracted the maximum attention in the field of computing and networking. It enhances the communication facility, which has a positive impact on many telecommunication and computing field [1]. This tremendous growth on the other hand has negative impact on high energy consumption in these sectors. The report, which records the fastest growth of telecom sector, is the first eyewitness of the demand and love of the inhabitants towards internet access. This report accounts increase in mobile subscriptions by 7 fold growth, from 6.5% to 43% in between 2000 and 2015. The proportion of households with internet access growth rate is 46% in 2015 whereas in 2005 it was 18% [2]. These growing numbers of users have led to tremendous growth in energy consumption globally. Studies show that more than 4.7% of worldwide energy is consumed by ICT [3]. Major organizations like Microsoft, eBay, Google, etc. spend million dollars on power consumption in ICT [4]. Simultaneously, they are also accountable for approximately 2% CO2 emissions worldwide. With a rise in the demand for telecommunication services, energy consumption becomes a serious issue from the angle of economical and environmental perspective. This draws attention for more energy efficient and eco-friendly networks.

The energy consumption issue is a serious problem in conventional network as compare to SDN. It is very challenging to use any energy efficiency policies in conventional network due to tight coupling of data and control plane in each network device [5]. Any change in data forwarding decision cause to alter in the low-level controlling command in each device, which is a very complex job. For more simplification and to bring innovation in networking industry software defined networking was developed. In SDN the data plane is logically separated from the control plane. The data plane is doing the same job of packet forwarding, but the control plane of all devices, combined logically to form a central controller. This controller is taking the responsibility for optimization of the network operation and configuration of the network remotely. With the help of this controller, SDN can take network wide decision globally. Information collected from the entire network is measured as traffic consolidation. With the help of central control logic SDN is able to take the decision regarding the energy consumption and other important metrics related to the traffic. It is possible only because of the central controller which controls the distributed network. This feature is not possible in traditional network as it has very limited information restricted to the local control decision. In summary, SDN provides more opportunities of providing energy saving features of the network [6].

The paper is organized as follows. Section 2 represents related work. Problem formulation and solution for the same is discussed in section 3 and 4 respectively. Section 5 represents performance evaluation results. Finally, the paper concludes in section 6.

2. Related work

With the growing demand of SDN, energy savings are also becoming an important issue to gain attention. Various parameters are responsible for rising energy related issues in SDN. Routing is one of the major integral parts of network devices. Most of the energy related issues are associated with this. As the name implies, SDN speaks about programmability in the network, so here special care must be taken while routing packets in the link. This unique character makes SDN to be superior to that of the conventional network. With the help of central controller these routes can be programmed to be more energy efficient. In order to solve this problem several researchers focus in this area of the energy field in SDN. Among all these approaches, two most common approaches are link rate adaptation and switching off idle network components without introducing any delay in packet forwarding [7].

Link rate adaptation or speed scaling approach is based on scaling power of network based on the amount of traffic it carries. Here the devices are getting power, based on the amount of traffic they handle. In switching off approach it conserves energy by switching off unused network elements [8]. In the second approach the elements either operates at full power or zero power. Both these approaches can save significant energy savings with negligible packet loss and delay in packet forwarding. Each of this approach has its own pros and cons.

Andrew et al. in [9] reported that route optimization is a function of its energy consumption. In order to optimize routing operation in network focus should be given in energy savings also. The authors also studied network wide routing problem to satisfy a set of demand rate requirements without the cost of the energy consumed. They used power down approach as their next level of extension in their paper. They proved ON-OFF oscillation can be condensed in routing and studied the behavior of energy consumption and end-to-end delay.

Nedevschi et al.,in [10] confirmed that in SDN by using both the two approaches powering down and speed scaling, a significant amount of energy savings can be done without hampering much on packet loss and end-to-end delay. This principle is not possible to be followed in conventional network as it is very difficult to get network wide status, link utilization factor, network topology information and complete network status. In SDN with the help of central controller it is quite possible. The central controller collects all the network wide information and takes the decision to switch off number of links based on the utilization and vary the link rate as per link utilization.

Some authors [11, 12] prefer to maximize the number of OFF links in order to save a good amount of energy in SDN. New algorithms are proposed in order to reroute the traffic using a limited number of active links. The ON links are operating at their maximum and almost run in the highest data rate. They ignore the amount of energy savings that can be achieved by adjusting the linkage rates as per their utilization factor. The variation for the same is seen in other papers.

Wang et al., in [13] used rate adaptation technique to achieve a good amount of energy savings in network. New routing solutions are proposed that transform the discrete link rates to continue one. It provides a good solution for uniform demand. However, it is not possible to put these strict requirements in order to satisfy current networks.

In this paper, we propose an energy aware routing in SDN which has the ability to manage at a discrete link rate. All the network related information is with the central controller, which is responsible for taking any decision globally. Depending on traffic conditions the controller helps the link to switch from high to low data rate. This solution helps the link to operate in low power active state in the low transmission period instead of using sleep mode to diminish the power. Switching off unutilized network element no doubt gives a comparatively good amount of power saving, but at the same time challenges sleep mode recovery issues. In a wired network links operated on idle or fully utilized mode consume the same amount of power. In this case power consumption is independent of link utilization. In order to conserve the required amount of energy link can be operated in low data rate in low utilization period. So by varying the link rate to match with the utilization a considerable amount of energy can be saved. In these situations energy usage in wired links is proportional to link utilization.

3. Problem formulation

SDN architecture constitutes of three layers, infrastructure layer, control layer and application layer. The lowest layer is the infrastructure layer which consists of various forwarding elements (FEs) in order to forward the packets coming in the form of a request to the network. The path is determined based on the forwarding rules that are set by the control layer [13]. Along with that, the forwarding elements share the network related information to the central controller. The controller serves as the manager of the whole network. It works as a bridge between the two layers infrastructure layer and application layer through north bound interface and south bound interface respectively. The application layer role is to optimize the network through various schemes such as load balancing, resource allocation, mobility management. This is possible due to the control layer which is taking the responsibility of injecting the flow rules in FEs.

The software defined network is modeled as a single central controller with its subordinates as FEs. Each FE follows the instruction of the controller in order to forward the packet as per the flow rules. For each traffic demand the corresponding flow rules are incorporated. In order to reduce the energy consumption in the network, we focus on the number of active switches and links. The energy consumption can be optimized by carefully monitoring the number of active switches in a particular time. Following with the same rule, we try to decrease the number of active links. The load of these links is bypassed through other active links in such a way that their maximum utilization is under the link capacity. The energy consumption of the link is directly proportional to the number of data flows through it. The moment any link is switched off care must be taken that its adjacent switches should not depend on these links for packet flow. If a switch adjacent to any switched off link sits idle, then it can be put to off state till any further demand of this switch. At the same time in the process of switching on and off, that the transition period does not give any guarantee on the potential power savings. The system also needs to be adapted with varying link rate. Our proposed system is based on all these assumptions.

For this the network is modeled as undirected graph G (V, È), where V is the set of switches or FEs and È is the set of bidirectional links connecting the FEs. The set È does not contain the link connecting the FEs to the central controller. The link originated from FE is denoted as $E^{\rightarrow}$ and that terminated at FE is denoted by $E^{\leftarrow}$.

Such that $E^{\rightarrow} \cup E^{\leftarrow} = \grave{E}$ (1)

There are set of flows F= {1, …, f…F} that travel through these links. Each flow is originated from origin denoted as org(f) and reached at the destination denoted as dest(f), where f stands for the f^{th} flow. For routing each flow f has to follow a path p^f. This path is the interconnection of various links serving as the route of a flow. The data rate of each flow is denoted as r_e^f. This is a decision variable. If $r_e^f \geq 0$, it denotes that data rate r^f of a flow f through link e, where e Є È. If $r_e^f=0$ denotes that no flow is traversing through link f. So the flow conservation rule is applicable to the origin and destination of each flow. For each flow f, f Є F, and FE Є V the flow conservation rule says

$$\sum_{e \in E^{\rightarrow}} r_e^f - \sum_{e \in E^{\leftarrow}} r_e^f = \begin{cases} 0 & v \neq org(f) \, and \, v \neq dest(f) \\ r^f & v = org(f) \\ -r^f & v = dest(f) \end{cases} \tag{2}$$

The sum of the data flows through a given link e, e Є È tells the data rate of the link. Depending on the incoming traffic demand to network the central controller sets of one out of the several data rates. Here, all the available data rates are sorted in ascending order. The data link rates for any link e, e Є È is given by

$$\sum_{e \in E} \Delta_e r_e \geq \sum_{f \in F} r^f s^f(\Delta) \qquad \forall \Delta \geq 0.$$

$$\bar{r}_e = \begin{cases} R_0 & 0 < r_e \le R_0, \\ R_1 & R_0 < r_e \le R_1, \\ . & . \\ . & . \\ . & . \\ R_{\max} & R_{\max-1} < r_e \le R_{\max} \end{cases} \quad (3)$$

Energy consumption value associated with each link rate is different and given by,

$$\varpi_e(\bar{r}_e) = \begin{cases} \omega_0 & if\ \bar{r}_e = R_0, \\ \omega_1 & if\ \bar{r}_e = R_1, \\ . & . \\ . & . \\ . & . \\ \omega_{\max} & if\ \bar{r}_e = R_{\max}. \end{cases} \quad (4)$$

So, based on the above description the flow rules for each link e, e Є È consists of finding the flow rate. As this flow rate will effectively give an idea about the amount of energy consumption in a network. So, the problem of energy consumption in the network is reduced to the following mathematical problem

$$\min \sum_{e=1}^{\mathrm{E}} \varpi_e(\bar{r}_e) \quad (5)$$

$$subject\ to\ r_e = \sum_{f\in F} r_e^f \quad e \in \text{È} \quad (6)$$

$$\sum_{e\in E^{\rightarrow}} r_e^f - \sum_{e\in E^{\leftarrow}} r_e^f = \begin{cases} 0 & f \in F, v \ne org(f), v \ne dest(f), \\ r^f & f \in F, v = org(f), \\ -r^f & f \in F, v = dest(f). \end{cases} \quad \forall v \in V \quad (7)$$

$$\bar{r}_e = \begin{cases} R_0 & 0 < r \le R_o, \\ R_1 & R_0 < r \le R_1, \\ . & . \\ . & . \\ . & . \\ R_{\max} & R_{\max-1} < r \le R_{\max}. \end{cases} \quad e \in \text{È} \quad (8)$$

The objective function to minimize energy consumption is given in equation 5. The constraint in equation 6 states that all the data rates through this link should support for this. Equation 7 guarantees about flow conservation constraint. The link rate $\bar{r}_e$ is the summation of all the flows through the link e is given in equation 8. The mathematical formulation given in equation (5-8) is equivalent integer-linear programming. This problem is NP hard based on integral routing constraint and the discreteness of the objective function.

4. Solution based on queue length and link rate

Energy consumption of the wired network is minimized by carefully monitoring each link cost. Let Δ is the link cost associated with each flow f, f Є F from org(f) to dest(f). So for any flow f, f Є F, the optimal route cost is given by $cost^f(\Delta)$. The value of Δ is considered as the lowest routing cost in order to flow the packet in their respective shortest paths.
It is given by equation 9.

$$\sum_{e\in E} \Delta_e \bar{r}_e \ge \sum_{f\in F} r^f s^f(\Delta) \quad \forall\Delta \ge 0. \quad (9)$$

The lower bound of $cost^f$ is given by equation 10.

$$\sum_{e\in E} \Delta_e \bar{r}_e = \sum_{f=1}^{F} cost^f(\Delta) \times r^f \quad \forall\Delta \ge 0 \quad (10)$$

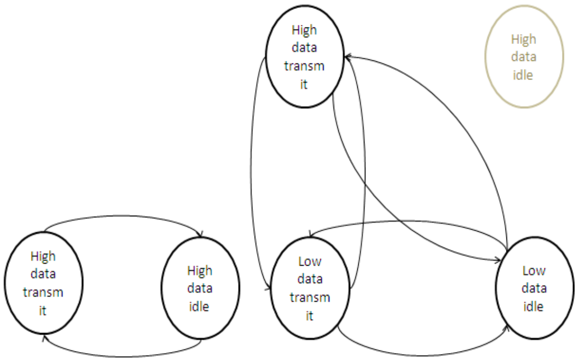
Equation 10 states that the lowest routing cost of a network is the sum of all the shortest path cost multiplied by the required data rate of that path. Energy consumption in a link varies with the data rate. The higher is the data rate in order to entertain a number of traffic, cost of communication is more.
So the problem formulation given in equation 5 can be reformulated as follows:

$$\min \sum_{e=1}^{\mathrm{E}} \varpi_e(\bar{r}_e) \quad (11)$$

$$subject\ to\ \sum_{e\in E} \Delta_e \bar{r}_e = \sum_{f=1}^{F} cost^f(\Delta) \times r^f \quad \forall\Delta \ge 0 \quad (12)$$

$$\bar{r}_e \in \bar{R} \quad e \in \text{È} \quad (13)$$

Equation 12 clearly indicates that to minimize link cost, data rate of the link should be minimized. As different cost value is associated with different link rates. So with higher data rate, the cost of a link is increased as given Figure 1 (a). So to minimize the cost of communication special care must be taken in order to check the condition for higher data rate in the network.



(a) Transmission state in conventional wired network

(b) Transmission state in Proposed one

Fig. 1: Comparison of Transmission State in Conventional Ethernet and Proposed One.

Minimizing energy consumption in wired network is a major issue. Different techniques have been proposed in order to reduce the overall energy consumption IEEE 802.11az facilitates network components to sleep and wake-on-arrival when a packet arrives. Care must be taken so that no transmission is lost during wake up phase and no delay to be incurred in the transition period from sleep to active mode. Some network interfaces also facilitate variation in data rate depending upon the traffic pattern. The sleep mode approaches have several drawbacks for reducing the energy consumption [9].

- Sleep modes may be too harsh: Sleep mode approach is very general approach of energy conservation. It may have an impact on the state of a link and also software support issues in operating systems. Protocol extensions are required to recover from sleep mode. The sleep mode approach may provide the highest amount of power saving, but it does not guarantee to provide the best solutions in terms of its transition into sleep mode with all system types and traffic patterns.
- Slow rate adaptation: In a network the change in traffic pattern occurs so quickly during the operation of a network. For the smooth conduction of network processes, faster rate adaptation is always preferable. It provides a good amount of energy savings and also provides better network performance. With the slow rate of adaptation the network is not suitable to all the kinds of network.

In order to overcome these drawbacks, we propose to use an Ethernet's ability to operate at a different link rate and dynamically varying the data rate with the traffic conditions. With this solution the link never goes for sleep mode mechanisms to save power. Rather, it can operate in low data rate conditions to save more power. In order to explain this we can refer to the state diagram given in Figure 1(b).

In a conventional network mostly the wired link operates with the highest common data rate, so that it is preferred for all the incoming packets. This will continue until the link is no more active. So in this section we use the different data rates in a dynamic fashion instead of considering a single rate of connectivity for the entire network duration. The transmitter can switch between the high and low data rate. The benefit of the above process is that the transmitter is either in low power mode or high power mode, but not in idle state. The scenario is different when links operate in high data rate for the entire connectivity, as more chances of getting idle state, and to save power, switch off state is preferable over idle state. So in our proposed approach we try to avoid the high data rate idle state. Depending on the arrival traffic pattern data rate varies from low to high but never goes idle.

Power savings of the link are directly linked to data rate at which the link has operated on. In physical layer the clock speed is different for different data rate. For example, for 10 Mbps, the clock is 2.5 MHz, for 100 Mbps, clock is 25 MHz, etc. The switching activities are increased rapidly with the increase of clock speed causes use of more power. By limiting the link to operate in low data rate, part of the circuit working for high data rate can turn off, resulting in saving more power.

4.1. Working principle

For the proper functioning of the network, the controller is maintaining the route for the data packets in the flow table of the router. When the incoming request comes at the source router, the router checks its routing table. It finds out whether the root to destination router is present or not. If it presents, the router forwards the packets in that route. Otherwise the controller finds the route for that incoming packet.

To decide the power for incoming traffic special care must be taken to determine the valid data rate. Since the traffic to the network is not constant at all time. So there must be a mechanism to take a decision on the amount of incoming traffic for transmission. For this we are taking the help of a queue data structure.

The data rate of any link should be within the threshold value. Because in case of wired network higher data rate leads to more amount of energy consumption. And also frequently making any link to be switched off and on is having a direct impact on its energy consumption value.

For this consider a scenario of the network. Here the packet needs to be sent from source S to destination D. On the arrival of the packet-in messages at the source node, that the first packet of any flow, the switch S checks about the path in its routing table. If the path exists, it uses that as the route. Otherwise, it sends a control signal to the controller to find the path. After the controller finds the route, the same is updated in the routing table of the router. To determine the speed it depends on the queue length of the source node. One queue is maintained at the source node. The queue length determines the data transmission rate. To monitor this we are maintaining three status values: status 1, status 2 and status 3. These values are maintained in the controller for every path. Two threshold values are maintained for each queue. The queue length gives information about the status values.

Status 1- The packet length is less than the threshold 1 value.

Status 2- The packet length is in between threshold 1 and threshold 2.

Status 3- The packet length is greater than the threshold 2 value.

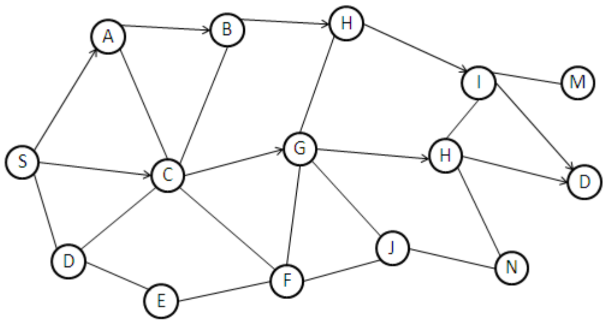


Fig. 2: An Exemplar Network.

Each status value leads to a different data rate. Status 1 to 3 gives information about low to high data rate for transmission of packets. So to determine the data rate for transmission the queue length is used as a metric. Data rate switching can only occur at the frame boundaries based upon the queue length in order to avoid any type of disruption of active packet transmission. In figure 2 let the source start using the path S-C-G-H-D to send the packets. Initially the number of packets in source queue is less than threshold q_{th1}. In this condition the rate of data transmission is low set by the controller. With the increase in the number of packets in the queue the threshold value is greater than 1. The status value changed to 2. The rate of data transmission is set higher by the controller. The advantage of increasing the data transmission value has a positive effect of lowering the delay in packet reception in the desired path. The moment when queue length touches q_{th2}, status value changes to 3. At this condition, the controller decides for the alternate path S-A-B-H-I-D for new transmission. The data transmission in both the path is within the limit. With the change in transmission rate energy consumption parameter varies. The low data rate leads less energy consumption.

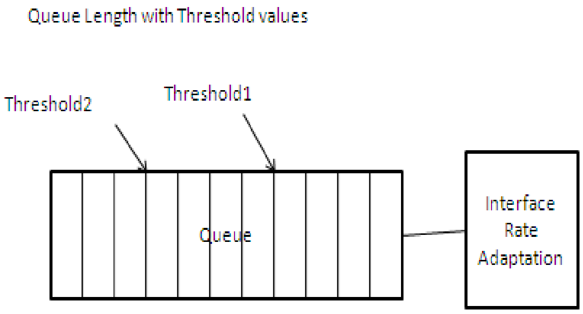


Fig. 3: Queue Length Based System Architecture.

Now, the main objective is to find out the queue size as given in Figure 3. Its impact is on the data transmission rate of the packets. Since the amount of buffer space gives the space for allowing the incoming frames to be present in a queue.

$$q_{\max} = q_{th1} + q_{th2} + \frac{rate_{high}}{rate_{low}} \times frame_{\max} \tag{10}$$

Here the queue size is given by $q_{\max}$. The incoming rate of the frame is restricted by frame size. $\text{Frame}_{\max}$ gives the information about the maximum incoming frame size.

Proposed Algorithm:

Algorithm 1: SERAT

Input: Incoming Packets

Output: Packet_out

```
for all the incoming Packets coming to the network do
    if no designated route in the network then
        Send the incoming request to the controller to find the path
        Update the flow table of the router status and value in the controller
    if status==3 then
        Ignore that path and find another path for packet forwarding
```

```
7)  end //End If
8)  else
9)  The packet is forwarded in the desired path and check its status values
10) end // End If
11) end //End For
```

SERAT algorithm is summarized in algorithm 1. For all incoming packets in the network the first job is to find the route. If the routing table of the source node is updated for the same, then the packet is forwarded in the desired path. Otherwise, the packet is sent to the controller for finding the path. At the time of estimating the new routes, the controller job is to take care about the status information of the path. As this status value is directly proportional to the data rate of the path. In a wired network higher data rate cause the more amount of energy consumption than the path with a lower data rate. At any point of time, if any path is having the status value is 3, and then the controller will find an alternative path to lessen the burden of that path. In this way the overall energy consumption of the network can be minimized. In the next section we study the performance of the proposed system and comparing its behavior with EXR algorithm.

5. Performance evaluation

In this section, we present the simulation results to evaluate the behavior of SERAT. We simulate the behavior of SERAT and measure its performance with respect to the baseline algorithm EXR algorithm under the typical traffic of SDN using Mininet [14]. Mininet is very popular among the researchers for running the larger network at the cost of a limited number of resources. It is very helpful in the research direction of SDN and OpenFlow. Though several other simulators are available, but in comparison with Mininet, they are very expensive. By comparing all the parameters; Mininet is the best suitable simulator for running a network simulation on a single computer without much modifying the code.
In the existing platform, we calculate the total energy consumption of the network by considering the total power consumption of servers, switches and links involved in data transmission. Here, we are considering TCP application with Northbound interface. The network flows are selected for a random host which is set to be received at a designated time interval. The performance parameter is studied with network speed 10 Mbps and 50 Mbps.

We measure the performance of the EXR algorithm as the baseline methods and compare its behavior with SERAT. In EXR flows are routed in available path and then it will set as per the priority. We use different flow metrics to measure the behavior of the routing algorithm. First is total energy consumption and second is throughput.

Performance measure at 30 Mbps.

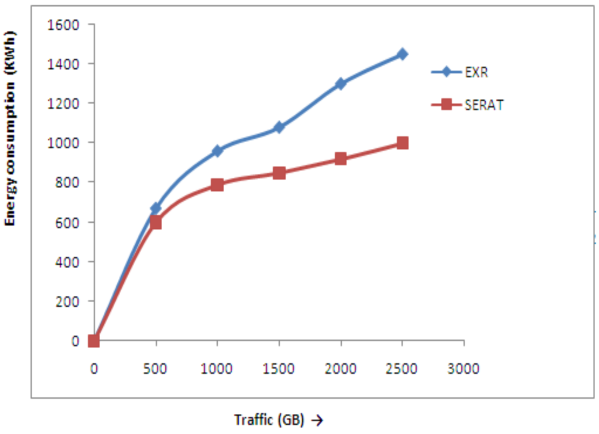


Fig. 4: Energy Consumption vs. Traffic.

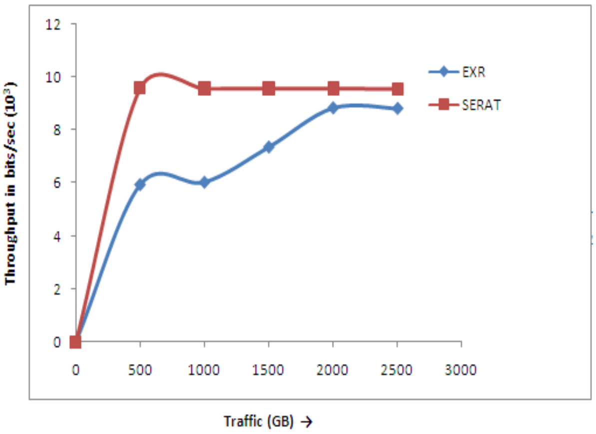


Fig. 5: Throughput vs. Traffic.

Performance measure at 50 Mbps.

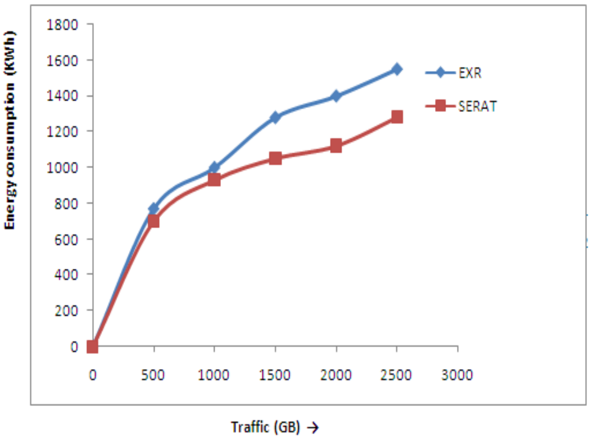


Fig. 6: Energy Consumption Vs. Traffic.

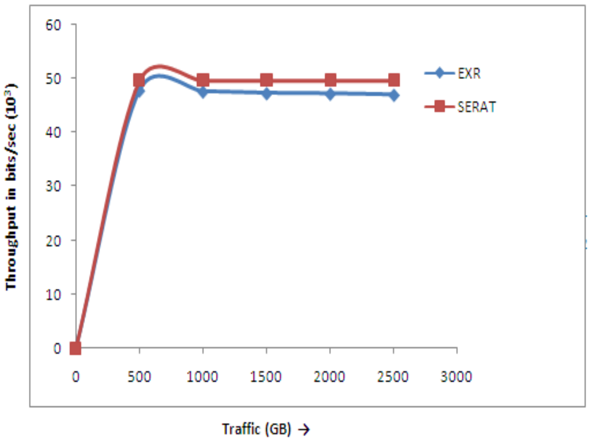


Fig. 7: Throughput vs. Traffic.

The plot for the energy consumption vs. traffic is shown in Figure 4 and Figure 6 in the speed of 10 Mbps and 50 Mbps respectively. It has been observed from the figure that the energy consumption of network in SERAT is lower than EXR. It is because we are avoiding high data rate traffic in SERAT.

Next, we plot the graph for throughput vs. traffic in a speed of 10 Mbps and 50 Mbps as shown in Figure 5 and Figure 7 respectively. It is observed from the figure that in SERAT the throughput is almost maintained same or greater than that of EXR.

6. Conclusions

With the rising demand for networking energy consumption is increasing rapidly. In order to satisfy the growing demand for network usage, traditional network causes a more amount of energy consumption. To control the network globally and adding innovation in networking, software-defined networking come into existence. The programmability behavior of SDN makes it more superior than traditional network. In this paper, we investigated some of the existing approaches for energy aware routing in network application. We considered rate adaptation techniques in order to minimize energy consumption. We proposed a new technique using buffer queue length to control the data rate in the link. The variation in link rate is considered as a function of link utilization. In the future we want to study the network behavior with variation in data rate, whether violating Service level Agreement (SLA) or not.

References

[1] W. Xia, Y. Wen, C. Foh, D. Niyato, H. Xie, "A survey on software-defined networking", IEEE Commun. Surveys Tuts., vol. 17, no. 1, pp. 27-51, 2015. https://doi.org/10.1109/COMST.2014.2330903.

[2] M. K. Awad, Y. Rafique, R. A. M'Hallah, "Energy-aware routing for software-defined networks with discrete link rates: A benders decomposition-based heuristic approach", Sustainable Computing: Informatics and Systems, pp. 31-41, 2016.

[3] E. Gelenbe, Y. Caseau, "The impact of information technology on energy consumption and carbon emissions", Ubiquity, vol. 2015, pp. 1:1-1:15, Jun. 2015.

[4] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, B. Maggs, "Cutting the Electric Bill for Internet-Scale Systems", Proc. ACM Conference on Applications Technologies Architectures and Protocols for Computer Communications (SIGCOMM 2009), Aug. 2009.

[5] D. Li, Y. Shang, C. Chen, "Software defined green data center network with exclusive routing", Proc. IEEE INFOCOM, pp. 1743-1751, Apr. 2014. https://doi.org/10.1109/INFOCOM.2014.6848112.

[6] D. B. Rawat, S. R. Reddy, "Software defined networking architecture security and energy efficiency: A survey", IEEE Commun. Surveys Tuts., vol. 19, no. 1, pp. 325-346, 1st Quart. 2017.

[7] N. Huin et al., “Bringing Energy Aware Routing closer to Reality with SDN Hybrid Networks”, Proc. IEEE Global Communications Conference (GLOBECOM 2017), 2017. https://doi.org/10.1109/GLOCOM.2017.8254456.

[8] Y. Hanay et al., "Saving energy and improving TCP throughput with rate adaptation in ethernet", Proc. IEEE Int. Conf. Commun. (ICC), pp. 1249-1254, Jun. 2012. https://doi.org/10.1109/ICC.2012.6364655.

[9] M. Andrews, A. F. Anta, L. Zhang, and W. Zhao, "Routing and scheduling for energy and delay minimization in the powerdown model," Networks, vol. 61, no. 3, pp. 226-237, 2013. https://doi.org/10.1002/net.21474.

[10] S. Nedevschi, L. Popa, G. Iannaccone, D. Wetherall, S. Ratnasamy, "Reducing network energy consumption via sleeping and rate-adaptation", Proc. 5th USENIX Symp. on Networked Systems Design and Implementation (NSDI '08), pp. 323-336, 2008.

[11] A. Markiewicz, P. N. Tran, A. Timm-Giel, "Energy consumption optimization for software defined networks considering dynamic traffic", Proc. IEEE 3rd Int. Conf. Cloud Netw. (CloudNet), pp. 155-160, 2014.

[12] R. Wang et al., "Energy-aware routing algorithms in software-defined networks", Proc. IEEE 15th Int. Symp. WoWMoM Netw., pp. 1-6, Jun. 2014.

[13] L. Wang, A. F. Anta, F. Zhang, C. Hou, Z. Liu, "Energy-efficient network routing with discrete cost functions", TAMC, pp. 307-318, 2012. https://doi.org/10.1007/978-3-642-29952-0_32.

[14] K. Kaur, J. Singh, N. Ghumman, "Mininet as Software Defined Networking Testing Platform", International Conference on Communication Computing & Systems (ICCCS-2014), 2014.