##### A HEURISTIC APPROACH TOWARDS ENERGY EFFICIENT SOFTWARE DEFINED DATA CENTERS

##### INTERNSHIP PROJECT REPORT



###### ***Submitted by***

##### KUMARI RENUKA

IN

COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF SCIENCE AND TECHNOLOGY

PALUR HILLS, BERHAMPUR-761008

JUNE 2017

MONTH & YEAR

**BONAFIDE CERTIFICATE**

This is to certify that this project report entitled **“A Heuristic Approach Towards Energy- Efficient Software- Defined Data Centers”** submitted to **National Institute of Science and Technology, Berhampur**, in connection with the NIST- Summer Research Internships & Fellowships Program is a bonafide record of work done by **“Kumari Renuka”** under my supervision at the **“Prof R K Shial**” from **“17 June 2017**” to “**30 June 2017”.**

Rabindra K Shial

Assistant Professor

National Institute of Science and Technology

Palur Hills, Berhampur-76100

Place: Berhampur

Date: 27 June 2017

**ACKNOWLEDGMENT**

**…………………………………………………**

I would like to express my heart felt gratitude to Prof. SangramMudali, Director of NIST and Prof. GeetikaMudali, Placement Director of NIST  for the planning of summer internship program and providing the funding of the fellowship and other all facilities. **……………………………………………………………**

Date: 27 June 2017

Kumari Renuka

2nd year Btech

Computer Science And Engineering

NIIT University

NEEMRANA, RAJASTHAN-301705

**EXECUTIVE SUMMARY (ABSTRACT)**

There has been rapid development of cloud computing. Due to this rapid development of cloud computing the demands on cloud services have been increased steadily, due to these huge demands of cloud services many Data Centers have been deployed to meet the traffic demand. Due to deployment of large Data Centers and in order to meet the traffic demand a large amount of energy is consumed. So, the explosive demands on cloud services imposes a heavy burden on modern data centers, this have raised the big concerns over the high energy consumption. In order to satisfy the ever-increasing data traffic needs, the energy consumption of data center network(DCN) also take a significant proportion. The main and important issue which is to be considered is the energy consumption by this data centers, solution to this problem need to be considered in order to optimize the energy consumption as well as to satisfy the huge traffic demand.

Software- defined Networking is the newly emerging technology which allows flexible control of the network devices (switches) and brings a new opportunity towards the DCN energy optimization. Software-defined networking allows decoupling of control plane and data plane which leads to more flexible maintenance and controlling to DCN.

In this paper, an energy-efficient network management strategy is designed which guarantees the satisfaction of network traffic demands in Software Defined Data Center Networks(SD-DCNs).

Basically, three main issues are tackled

1) the subset of switch which is to be activated (Switch Activation).

2) multi-path routing scheduling for all flows and

3) forwarding rule placement in SDN switches.

All these issues are jointly considered and formulated as integer linear programming (ILP) problem. A heuristic algorithm is proposed to solve this problem and to deal with its high computational complexity. Many simulation-based results are considered to investigate the efficiency of the proposed algorithm.

**TABLE OF CONTENT**

|  |  |  |
| --- | --- | --- |
| **SL. NO** | **CONTENTS** | **PAGE NO** |
| 1 | Introduction | 6 |
| 2 | Review of Literature/ Theoretical Background | 10 |
| 3 | Objectives | 14 |
| 4 | Research Methodology | 15 |
| 5 | Data Analysis, Results and Interpretation | 21 |
| 6 | Conclusions | 31 |
| 7 | Limitations of the project | 32 |
| 8 | References | 33 |

1. **INTRODUCTION**

The explosive demands on cloud services impose a heavy burden on modern data centers, especially on the energy consumption. Due to the huge energy consumption, a recent study [1] shows that electricity cost has become the dominant operational expenditure (OPEX), even surpassing the capital expenditure (CAPEX), e.g., the hardware cost, to data center providers. So, the energy consumption by the Data center energy consumption optimization has become an emergent issue for researchers from both academia and industry. It has become important to find a solution in order to optimize the energy consumed by this Data Centers. In a typical Data Center, the energy is consumed mainly by two parts- servers and network devices e.g. switches. Many efforts have been devoted and many strategies have been proposed for server energy optimization. On the other hand, since the network energy consumption accounts for the 20% of the total energy hence it cannot be ignored.

The modern DCN is usually deployed as multi-rooted network topology such as Fat- Tree, DCell, Port-Land and Bcube.

**FIG1: FAT TREE TOPOLOGY WITH 4 PODS**

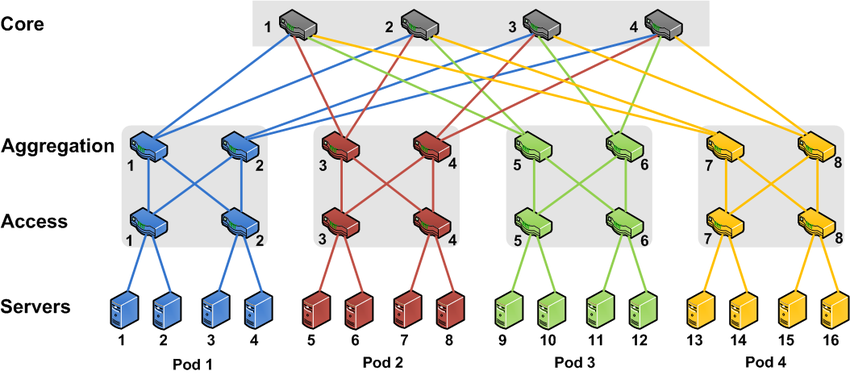


Figure 1 shows a 4-pod fat tree with four hierarchical layers which includes 4 core switches, 8 aggregate switches, 8 edge switches and 16 servers. Data transfer takes place between the servers. Hence many network devices have been provisioned to support the data transferring between servers. Main aim is to selectively deactivate parts of these devices without violating the quality of services(QOS) in order to save the amount of energy consumption. On the other hand, the hierarchical multi-rooted network topology makes multiple path possible between any two servers in the data centers. Hence it is highly recommended to take advantage of multi-path routing. Recently, Software Defined Networking (SDN) emerges as a new networking model which leads to a new opportunity towards DCN energy optimization. Software-defined data centers can be built by incorporation of SDN into DCN.

There are some inherent characteristics of SDN such as ternary content addressable memory (TCAM), which stores the forwarding rules is considerably expensive and size-limited. Each switch in the given 4-pod fat tree topology consist of its own TCAM memory which enables the multi-path routing for a flow in SD-DCN. Each switch in the selected path must allocate one forwarding rule. Unlike traditional DCN, multi-path routing in SD-DCN is additional constrained by the limited TCAM size.

* 1. **WHAT IS THE PROJECT**

The project focuses on a heuristic approach towards energy-efficient software-defined data centers. The goal is to develop an energy-efficient network management strategy with guaranteed satisfaction of network traffic demand in Software Defined Data Center Networks. In a typical data center, energy consumption mainly consists of two parts: servers and network devices e.g. switches. Since the network energy consumption accounts for the 20% of the total energy hence it cannot be ignored. Therefore, in this paper, it is investigated how to minimize the switch energy consumption by jointly considering switch activation, multi-path flow routing scheduling and forwarding rule placement, while guaranteeing the data center traffic demand satisfaction.

* 1. **DEFINATION AND PURPOSE OF THE PROJECT**

The explosive demands on cloud services imposes a heavy burden on modern data centers, this have raised the big concerns over the high energy consumption. In order to satisfy the ever-increasing data traffic needs, the energy consumption of data center network(DCN) also take a significant proportion. The main and important issue which is to be considered is the energy consumption by this data centers, solution to this problem need to be considered in order to optimize the energy consumption as well as to satisfy the huge traffic demand.

Therefore, the purpose of the project is to design an energy-efficient network management strategy which guarantees the network traffic demand in software-defined Data Centers. So, in order to provide a promising solution to DCN energy optimization the switch activation, multi-path flow routing scheduling and forwarding rule placement are jointly considered.

* 1. **SCOPE OF THE PROJECT**

Since there are huge traffic demand for the cloud services, so in order to meet this demands several Data Centers are deployed which lead to large amount of energy consumption. Due to huge energy consumption, a recent study shows that the electricity cost has become the dominant operational expenditure even surpassing the capital expenditure such as hardware cost. According to National Resources Defense Council in 2013 US data centers consumed 91 billion kilowatt-hours amount of power which is sufficient to power all the household of New York city over twice, therefore Data center energy consumption optimization has become an emergent issue. Since Software defined networking(SDN) allows more control over the network devices and leads to decoupling of control plane and data plane so it gives a more feasible and promising solution for DCN energy optimization.

SDDCs are truly revolutionary and packed with the capabilities enterprises need in the cloud. They free the application layer from the physical infrastructure layer and allow for a wide scope of uses including deploying, managing, storing, computing and networking myriad business applications in a cloud environment. SDDCs are adopted across various industry verticals such as telecom & IT, retail, healthcare, government and BFSI, manufacturing and others.

* 1. **SALIENT CONTRIBUTIONS OF THE PROJECT**

The main contributions of the project are, first to study the minimum-energy switch activation for multi-path routing in SD-DCN with the consideration of the limitations of SDN itself. Specially, first the energy optimization problem is formulated as integer linear programming(ILP) problem.

Secondly, a heuristic algorithm is proposed to deal with the high computational complexity of solving ILP.

Thirdly, through extensive simulation based studies, the high efficiency of the proposed ILP-based algorithm is shown and it is proved that the algorithm produces near-optimal solutions with much less scheduling time.

* 1. **OUTLINE OF THE PROJECT**

In order to design an energy-efficient network management strategy with guaranteed satisfaction of network traffic demands in Software Defined Data Center Networks the following are considered-

* the switch activation, multipath flow routing scheduling and forwarding rule placement are jointly considered and this energy optimization problem is formulated as integer linear programming problem.
* A heuristic algorithm is proposed to deal with its high computational complexity.
* Extensive simulation-based evaluations are conducted to validate the high efficiency of the algorithm.

The simulation-based results proved that the solution obtained are near to optimal solutions with much less scheduling time.

**2) REVIEW OF LITERATURE**

**2.1 THEORETICAL FRAMEWORK**

Data centers are physical or virtual infrastructure used by enterprises to house computer, [server](http://www.webopedia.com/TERM/S/server.html) and [networking](http://www.webopedia.com/TERM/N/networking.html) systems and components for the company's [information technology](http://www.webopedia.com/TERM/I/IT.html) (IT) needs, which typically involve storing, processing and serving large amounts of mission-critical data to clients in a [client/server architecture](http://www.webopedia.com/TERM/C/client_server_architecture.html). It is a large scale physical environments where an extensive number of computers and servers are housed or are interconnected in order to facilitate processing, storing and serving data.

It often requires extensive redundant or backup power supply systems, cooling systems, redundant networking connections and policy-based [security](http://www.webopedia.com/TERM/S/security.html) systems for running the enterprise's core applications.

DATA CENTER NETWORKING

Data center networking is the process of establishing and interconnecting the entire physical and network-based devices and equipment within a data center facility. It enables a digital connection between data center infrastructure nodes and equipment to ensure that they can communicate and transfer data between each other and on an external network or Internet.

SOFTWARE-DEFINED NETWORKING (SDN)

The software-defined networking basically involves decoupling of control plane and data place, it is a newly emerging technology which provides more control over the network devices. [Software-defined networking](https://en.wikipedia.org/wiki/Software-defined_networking) (SDN) includes [network virtualization](https://en.wikipedia.org/wiki/Network_virtualization) which is the process of merging hardware and software resources and networking functionality into a software-based [virtual network](https://en.wikipedia.org/wiki/Virtual_network). Originally, SDN focused on separation of the [control plane](http://searchsdn.techtarget.com/definition/control-plane-CP) of the network, which makes decisions about how [packets](http://searchnetworking.techtarget.com/definition/packet) should flow through the network from the [data plane](http://searchsdn.techtarget.com/definition/data-plane-DP) of the network, which actually moves packets from place to place.

SOFTWARE-DEFINED DATA CENTER NETWORK (SD-DCN)

Software-defined data center can be built by the incorporation of SDN into DCN. Software defined data center (SDDC) is basically a software based data storage facility where all the resources which includes storage, networking, and security are combined and provided as a software service. SDDC provides a secured user portal which utilizes web-based servers and deliver the data effectively. It enables the end-user to access the information by virtualization and cloud technology. Moreover, SDDC provides an advanced data management solution which monitors and creates a backup of the information. It is the phrase used to refer to a [data center](http://www.webopedia.com/TERM/D/data_center.html) where all infrastructure is [virtualized](http://www.webopedia.com/TERM/V/virtualization.html) and delivered [as a service](http://www.webopedia.com/TERM/S/SaaS.html). Control of the data center is fully automated by software, meaning hardware configuration is maintained through intelligent software systems. This is in contrast to traditional data centers where the infrastructure is typically defined by hardware and devices.

In software defined data center networks all the element of infrastructure such as networking, storage, CPU and security are visualized and delivered as a service. But in traditional data centers the infrastructure is typically defined by hardware and devices.

**2.2 REVIEW OF THE EXISTING LITERATURE RELATED TO**

**THIS PROJECT**

[1] Data centers aim to provide reliable and scalable computing infrastructure for massive Internet services. To achieve these properties, they consume huge amounts of energy, and the resulting operational costs have been dominant over the capital cost (hardware cost). Most efforts have been focused on the reducing the energy consumed by the cooling system and the servers. Improvement basically includes better components as well as better software products. Since several data centers have deployed so the amount of energy consumption has by these data centers have been increasing rapidly. Networks are a shared resource connecting critical IT infrastructure, and the general practice is to always leave them on.

Hence the energy can be saved by improving a network’s ability to scale up and down as the requirements of the traffic demand. Any dynamic energy management system that attempts to achieve energy proportionality by powering off a subset of idle components must demonstrate that the active components can still meet the current offered load, as well as changing load in the immediate future. The main aim is that the power savings must be worthwhile, performance effects must be minimal, and fault tolerance must not be sacrificed. In this paper, they present an Elastic Tree- a network wide power manager which dynamically adjusts the set of active network elements such as links and switches to satisfy changing data center traffic loads. They first compare multiple strategies for finding minimum-power network subsets across a range of traffic pattern. They implement and analyze Elastic Tree on a prototype testbed built with production OpenFlow switches from three network vendors. Further, we examine the trade-offs between energy efficiency, performance and robustness, with real traces from a production e-commerce website.

Their results demonstrate that for data center workloads, Elastic Tree can save up to 50% of network energy, while maintaining the ability to handle traffic surges. They fast heuristic for computing network subsets enables Elastic Tree to scale to data centers containing thousands of nodes. We finish by showing how a network admin might configure Elastic Tree to satisfy their needs for performance and fault tolerance, while minimizing their network power bill. Elastic Tree can maintain the robustness and performance, while lowering the energy bill.

[2] Today’s data centers, containing tens of thousands of switches and servers, run data-intensive applications from cloud services such as search, web email, to infrastructural computations such as GFS, Cloud Store, and MapReduce. The goal of data center network (DCN) is to interconnect the massive number of data center servers, and provide efficient and fault-tolerant routing service to upper-layer applications. It is well known that the current practice of tree architecture in data centers suffers from the problems of low scalability, high cost as well as single point of failure. To overcome the problem of tree architecture in current practice, many new network architectures are proposed, represented by Fat-Tree, BCube, and etc. A clear diurnal pattern emerges: traffic peaks during the day and falls at night. Therefore, a great number of network devices work in idle state in these richly-connected data center networks.

At the same time, the energy consumed by power-hungry devices now becomes a headache for many data center owners.

In this paper, they discuss how to save energy consumption in high-density data center networks in a routing perspective. They call this kind of routing as energy-aware routing. The key idea is to use as few network devices to provide the routing service as possible, with no/little sacrifice on the network performance. Meanwhile, the idle network devices can be shut down or put into sleep mode for energy saving. They then formally establish the model of energy-aware routing problem, and prove that it is NP-Hard. Then they proposed a heuristic routing algorithm to achieve their design goal. The algorithm works in the following way. First, the network throughput is computed, which is the most important performance metric for data-intensive computations, according to the routing on all data center switches. The corresponding routing is called basic routing. Second, they gradually remove switches from the basic routing, until when the network throughput decreases to a predefined performance threshold. Third, switches not involved in the final routing are powered off or put into sleep mode. Extensive simulations in typical data center networks is conducted to validate the effectiveness of our energy-aware routing algorithm. The results show that the energy-aware routing algorithm is a feasible and efficient method for saving energy consumed by network devices in data center network, especially under low network loads.

1. **OBJECTIVE**

There are a lot of applications, e.g., data analysis applications using MapReduce, running in a data center. These applications may frequently require data transferring between servers. These data are generally called as intra-data-center trafﬁc, imposing a huge burden on the underlying DCN. A large amount of energy consumption takes place by these data centers.

So, the main objective is to design an energy-efficient network management strategy with guaranteed satisfaction of network traffic demands in software-defined data center networks.

In order to design an energy-efficient software-based data centers three main issues are tackled

* The subset of switch which is to be activated (switch activation).

In order to save the energy, it is not always important to activate all the switches, especially when the traffic demands are extremely low at night. This provides an opportunity to save energy consumption by deactivating reductant switches without violating the Quality of Services (QOS).

* Multipath routing scheduling for all flows.

Since hierarchical multi-rooted network topology such as fat tree is considered for the given problem, hence multiple path exists between any two servers. So, it is important to choose a particular path for the given pair of servers which requires minimum number of switches to be activated in order to save energy. Hence the flow routing should be carefully scheduled.

* Forwarding rule placement in SDN switches.

Since each switch is associated with the ternary content addressable memory (TCAM) which stores forwarding rules, is considerably expensive and size-limited. Hence TCAM constraint must be kept into consideration while choosing a path from all existing path in order to accommodate a flow.

All these three issues are jointly considered and should be formulated as linear integer programming problem. An algorithm must be defined to solve the above problem. Several simulation-based experiments must be conducted to validate the efficiency of the proposed algorithm.

**4) RESEARCH METHODOLOGY**

**4.1 METHOD OF RESEARCH**

In order to design an energy-efficient software defined data centers the a) subset of switch that shall be activated b) multi-path routing scheduling for all flows c) forwarding rule placement in SDN switches. All these issues are jointly considered and formulated as integer linear programming (ILP) problem.

A hierarchical DCN topology as an undirected graph G = (N, E), where N includes both SDN-enabled switch set V and the host set H, i.e., N = V ∪ H and edge set E represents the links between the nodes in N. the switches and the host are interconnected according to the adopted topology e.g. fat tree

**TABLE 1: SUMMARY OF NOTATIONS**

|  |  |  |
| --- | --- | --- |
| **S NO.** | **SYMBOL** | **DESCRIPTION** |
| 1 | Cuv | Link capacity between nodes u and v |
| 2 | Sv | TCAM size of switch v |
| 3 |  | Demand of flow k from source s to destination d |
| 4 |  | Flow k from source s to destination d passing through link (u,v) |
| 5 |  | Whether the flow k from s to d passing through v or not |
| 6 |  | Binary variable to denote whether switch v is on or not |
| 7 | A | Arbitrary large number |

**PROBLEM FORMULATION INTO INTEGER LINEAR PROGRAMMING PROBLEM**

Hence the following constraint are considered

1. **Traffic Demand Satisfaction Constraints**

To fully explore the available transmission potential in a DCN, multi-path routing is widely applied. A flow therefore may be split into multiple sub flows, which go through different paths from source node to the indented destination.

 …..(1)

Since in constraint (1) the first flow is over the link (s, v) and second flow is over the link (v, s) which denotes the same link, hence the subtraction of it is equal to the demand of flow k from source s to destination d.

**** .….(2)

Consider the constraint (2) the first flow is over the link (v, w) and the second flow is over the link (u, v), since the destination for both the flows are same, hence their subtraction will be equal to 0. Both v and w represents the intermediate switch.

 …..(3)

Since for both the flows the destination is same and u belongs to the either switch or server, hence their subtraction represents the demand of flow k from source s to destination d.

1. **Link Capacity Constraints**

In multi-path routing, no matter how a ﬂow is split or merged, the total data that can ﬂow along a link is limited by its capacity. As an undirected link capacity model is considered, the overall ﬂows, either from u to v or from v to u, on link (u, v) shall not exceed its capacity Cuv.

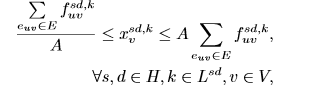
 ...…(4)

1. **Forwarding Rule Placement Constraints**

According to the SDN rule policy, for any given ﬂow going through a switch, there must be one forwarding rule in the TCAM to describe the ﬂow handling action. Note that no matter how a ﬂow is split or merged at a switch, we only need one rule to describe the multi-path routing action. We therefore deﬁne a binary variable x pow(sd, k) to the base v to denote whether ﬂow k from s to d passing v or not. Obviously, we have the below variable x as 1 if and only if there exists a flow k from source s to destination d which passes through the link (u, v).

 …..(5)

the relationship described above can be represented in linear form as given below

 .….(6)

The number of rules that can be stored in switch v ∈ V is constrained by its TCAM size Sv, provided the switch is activated. Otherwise, no rule can be stored in the switch. Therefore, ﬁrst deﬁne a binary variable Yv to denote whether switch v is on or not as given in constraint (7).

 ..…(7)

Constraint (8) is explained as follows. On one hand, when switch v is powered on, i.e., Yv =1, it describes the TCAMsize constraint. On the other hand, when Yv =0, equivalently the available TCAM size can be regarded as 0.

 …..(8)

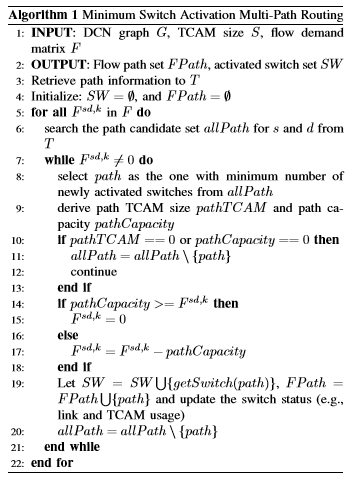
1. **An ILP Formulation**

The main objective is to minimize the number of switch which are activated in order to save energy saving and is represented by the equation (9). By summarizing all constraints discussed above, the problem can be formulated as an ILP problem as follow,

**ILP:**  .….(9)

**ALGORITHM DESIGN**

It is computationally prohibitive to solve the ILP problem and get the optimal solution for large-scale cases. We therefore propose a computation-efﬁcient heuristic algorithm in this section.



The main idea of our algorithm is to select paths for each ﬂow demand under the link capacity and TCAM size constraints, aiming at minimizing the current number of activated switches. In our algorithm

* we ﬁrst retrieve the path information to T.
* For each server pair, we can then ﬁnd out the candidate path set allPath from T.
* To ensure the trafﬁc demand satisfaction, we then start to schedule the routing paths for ﬂow from allPath until flow gets fully satisﬁed. In order to activate switches as less as possible, we always choose a path p with the minimum number of newly activated switches for each sub-ﬂow.
* Unfortunately, not all the chosen paths can be explored directly with the consideration of TCAM size and link capacity. We deﬁne the smallest TCAM size and the link capacity in the path as pathTCAM and pathCapacity. If there exists one switch whose TCAM or one link whose capacity has been used up, the whole path cannot be chosen and we shall consider the next path candidate.
* Otherwise, path can be allocated for sub-ﬂow, If the path capacity is sufﬁcient to satisfy current ﬂow demand, then it shall be routed through path.
* If the ﬂow demand exceeds the available path capacity, we greedily use up the capacity of current path ﬁrst and let the next path to handle the residual ﬂow.
* In either case, we shall update the path candidates allPath, ﬂow path setFPath, activated switch set SW, and the switch statuses (e.g., available link capacity, TCAM size and forwarding rules).

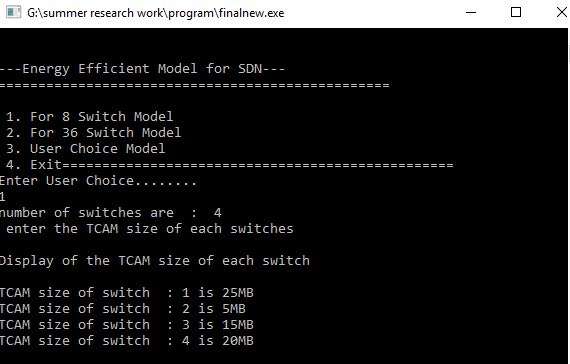
I have implemented the following algorithm in C++ programming language with the data set provided above.

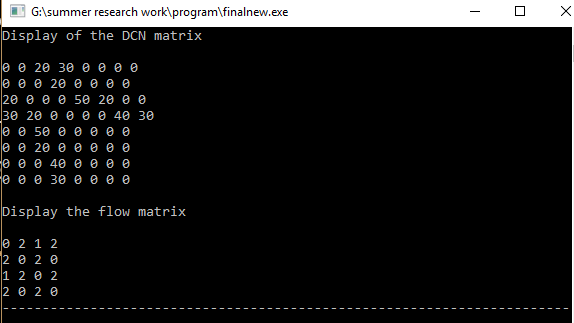
Fat tree is considered as a base data center topology. DCN graph is represented in terms of matrix where the nodes represent the servers or switches and the edges represent the link between nodes with weight as link capacity, the default link capacity is set as 1GBPS. The switches TCAM sizes and the flow traffic demands are generated in [250,1250] and [100KBPS, 100MBPS] uniformly at random. The source and the destination of each flow is randomly chosen from hosts in the network.

The screenshots of the input and output of the implemented code have displayed in the below section.

**5) DATA ANALYSIS, RESULTS AND INTERPRETATION**

**5.1. RESULTS OF THE IMPLEMENTED CODE**

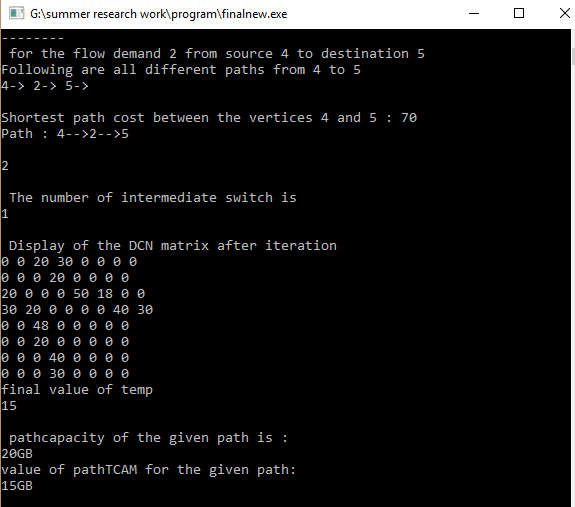
****

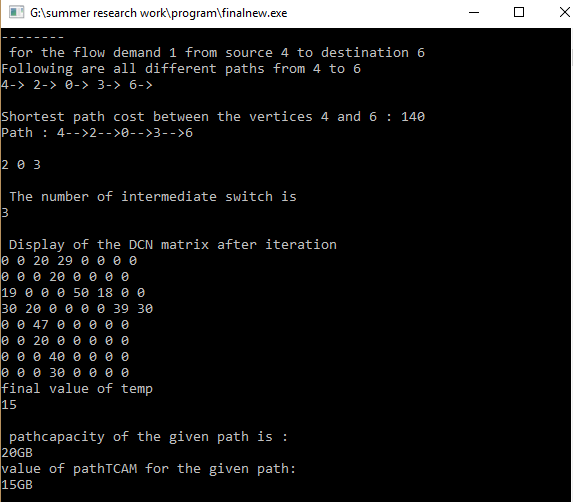
****

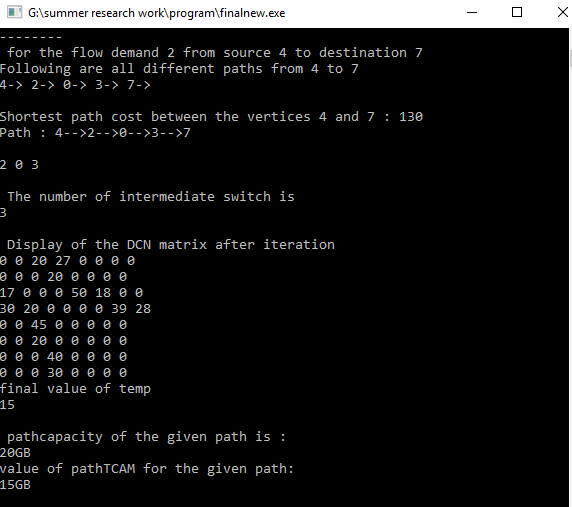
The above are the user inputs. The DCN graph in the form of matrix, the flow demand and TCAM of each switches are taken as an input from user.

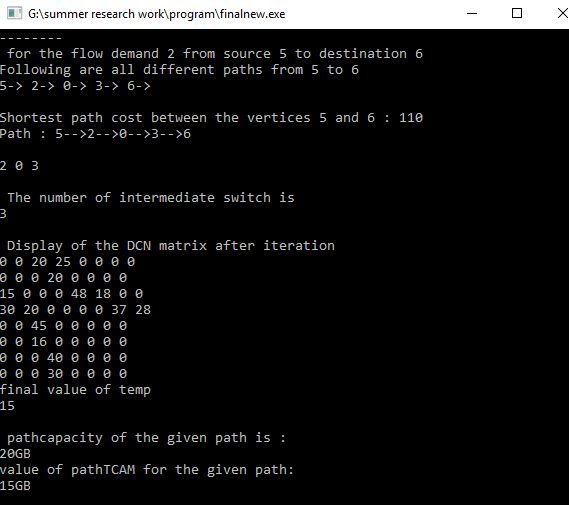
The below figure displays the flow demand of 2GB which is to be transferred from server 4 to server 5. First all the that exist between server 4 and server 5 is displayed and the shortest path which require minimum number of switch activation is choosen.

Pathcapacity and TCAM size of the path choosen is displayed in order to check whether the flow demand can be satisfied or not.

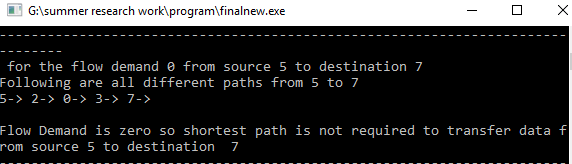
****

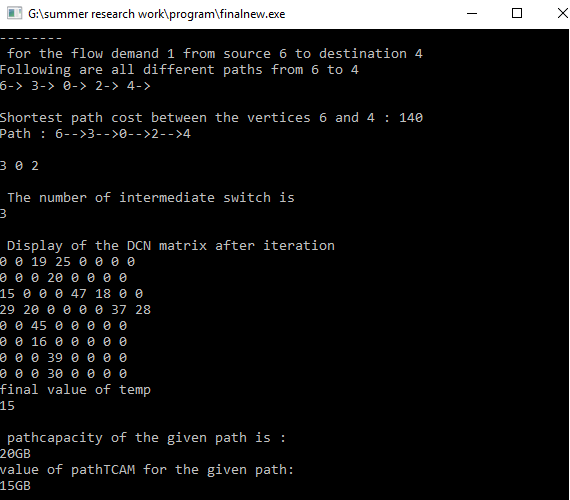
****

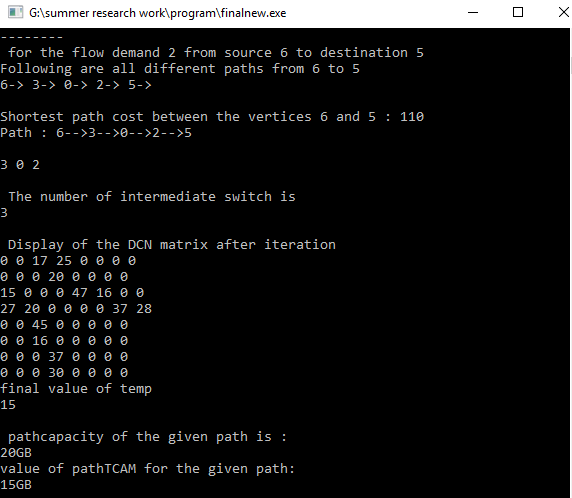
****

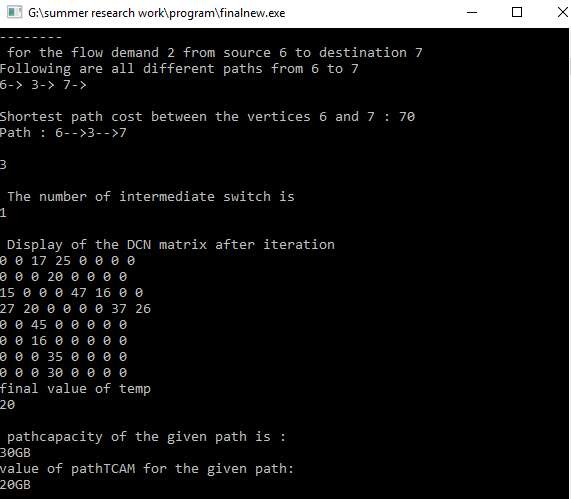
****

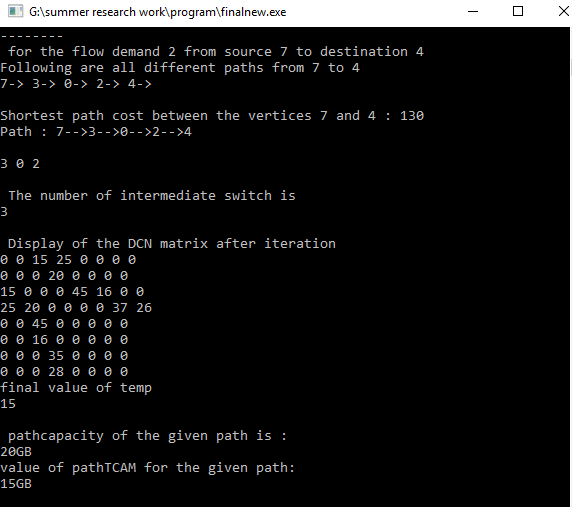
Since the flow demand is 0 hence no data need to be transferred so no shortest path is required for the given flow from server 5 to server 7

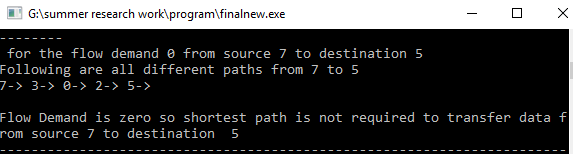
****

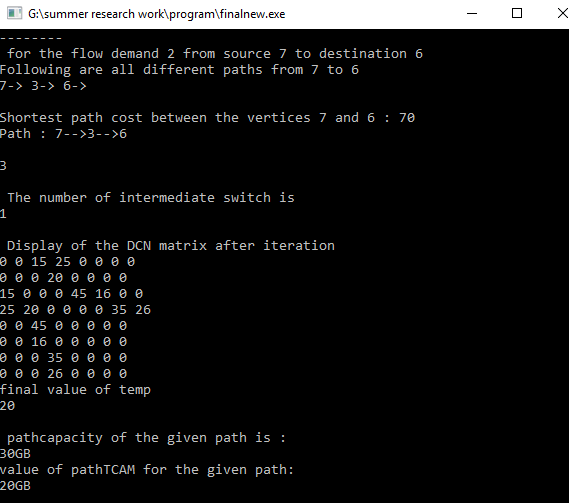
****

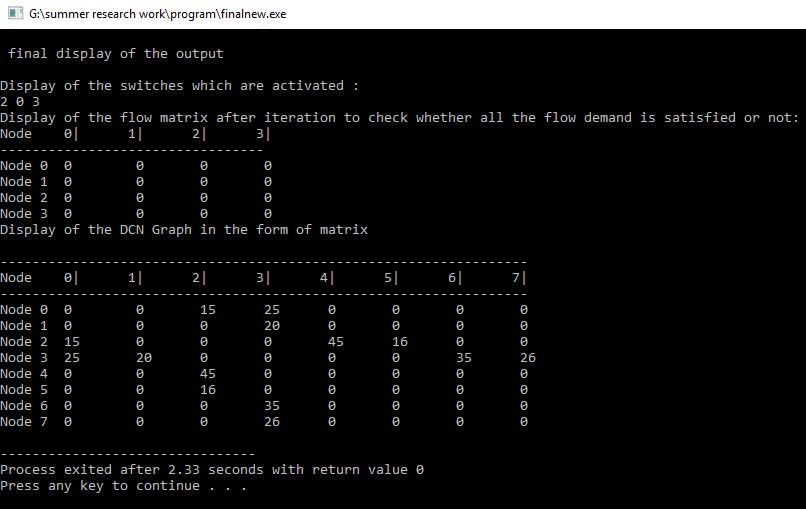
****

****

****

****

****

****

The above figure gives the final output in which displays the switches which are activated to order to satisfy all the flow demand. The flow matrix is displayed all satisfying all the demands, since the value of the flow matrix is 0 which means that all the flow demand has been satisfied with minimum number of switch activated.

The final DCN matrix is displayed which gives the information about the path-capacity of each edge left after satisfying all the flow demands.

**5.2 DATA ANALYSIS AND INTERPRETATION**

Several simulations based experimental results are considered in order to investigate the efficiency of the proposed algorithm. Investigation is done on the following parameters

1. ON THE SCHEDULING TIME

First the investigation is done on the scheduling time of both the proposed algorithm and the optimal solution using commercial Gurobi optimizer.

Two different data center scales, i.e., 6-pod with 54 hosts and 10-pod with 250 hosts, are considered, respectively. The average number of ﬂows emitted from a host is set as 1, 10, 20, 50 and 100 in different simulation instances.

The following are the observations-

* The number of activated switches obtained by the proposed algorithm is almost the same as the optimal solution, especially in the small-scale case with 6 pods.
* The computation time of the proposed algorithm is much lower than the one using Gurobi and the advantage becomes more signiﬁcant with larger network size or number of ﬂows.

1. ON THE ENERGY SAVING

Investigation is done how the proposed algorithm can save the energy consumption by deactivating the unnecessary switches and routing the flow with the given activated switches. The proposed algorithm is compared with the shortest path algorithm which always activates the switches on the shortest for each flow.

The no. of flows varies from 10 to 12,000 totals 20 instances are run to obtain the average energy saving percentage.

The average energy saving percentage decline with increase of flow number. This is due to the fact that more switches must be activated to accommodate the huge traffic demand incurred by large number of flows. Still the proposed algorithm shows obvious advantages over shortest path algorithm.

-effect of link capacity on the energy saving percentage

The TCAM size is fixed to 750 and the link capacity is varied between 500 to 1600. It is proved that the proposed algorithm performs better than the shortest path algorithm. Higher the link capacity indicates higher energy saving percentage. This is because more flow can go through a switch without violating the link capacity constraint and thus less switch need to be activated.

-effect of TCAM size on energy saving percentage

The link capacity is fixed to 1GB and the TCAM size is varied from 350 to 1050. From the given graph below it can be observed that the energy saving percentage increases with increase in TCAM size, higher TCAM size allows more flows to get through a switch resulting in less switches to be activated.

1. ON THE TCAM UNTILIZATION

In order to study the TCAM utilization the link capacity is fixed to 1GBPS and TCAM size as 750 to obtain a cumulative distribution function (CDF) of the TCAM utilization.

the proposed algorithm always exhibits a higher utilization ratio than shortest-path algorithm.

**6) CONCLUSION**

The main objective of this project is to investigate how to minimize the energy consumption in Software-Defined Data Center Networks by selectively activating the switches and carefully scheduling the multi-path routing according to the data center traffic demands.

So, in order to meet the requirements, the three main issues a) the subset of the switch which is to be activated (switch activation) b) multipath routing scheduling for all the flows c) forward rule placement in SDN switches, are jointly considered. This energy optimization problem is formulated as linear integer programming problem.

In order to address the computation complexity on solving the integer linear programming further a heuristic minimum switch activation multi-path routing algorithm is proposed.

Hence through extensive simulation high efficiency of the algorithm is proved and the results of the simulation-based experiment shows that the proposed algorithm produces near to optimal solution with much less scheduling time.

**7) LIMITATIONS AND SCOPE FOR THE FUTURE**

In the proposed minimum switch activation multi-path routing algorithm, for the given flow the path with the minimum number of newly activated switch is chosen.

So, the following might be the limitation of the proposed model-

* Since the path with minimum number of newly activated switch is considered hence the path chosen for the given flow might be the longest path then that of the other hence3 it would affect the quality of service(QOS).
* In the proposed algorithm, first all path for the given flow is calculated, so it is difficult to search all path at the runtime for the large-scale data centers.
* As the number of flows increases, the energy saving percentage decreases, this is due to the fact that more switch much be activated to accommodate the huge traffic demands incurred by large number of flows. Especially when the number of flows exceeds a certain value no energy saving can be obtained because the network traffic demand already saturates the base topology and all switch must be activated to ensure the quality of service(QOS) at this point, leaving no optimization space.

**8) REFERENCES**

[1] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and N. McKeown, “Elastictree: Saving energy in data center networks.,” in NSDI, vol. 10, pp. 249–264, 2010

[2] Y. Shang, D. Li, and M. Xu, “Energy-aware routing in data center network,” in Proceedings of the ﬁrst ACM SIGCOMM workshop on Green networking, pp. 1–8, ACM, 2010.

[3] D. Huang, D. Yang, and H. Zhang, “Energy-aware virtual machine placement in data centers,” GLOBECOM 2012, 2012.

[4] D. Li, Y. Shang, and C. Chen, “Software deﬁned green data center network with exclusive routing,” in INFOCOM, 2014 Proceedings IEEE, pp. 1743–1751, IEEE, 2014

[5] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, “The cost of a cloud: research problems in data center networks,” ACM SIGCOMM computer communication review, vol. 39, no. 1, pp. 68–73, 2008