International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

Available at http://inpressco.com/category/ijcet

Research Article

A Geographical Region based Linear Representation to Minimize CO2 for Green Cloud Computing

Manish Kumar Singh* **and Raghav Yadav**

Department of Computer Science and Information Technology, SHIATS, India

Accepted 30 Nov 2016, Available online 07 Dec 2016, **Vol.6, No.6 (Dec 2016)**

Abstract

Cloud computing is a paradigm which offers a variety of new services. This service model involves cloud-based service providers providing a large pool of computational resources that comprise several data centers at different geographical locations. Recently, carbon emissions associated with powering data centers and linked networks have become an important issue. Here to reduce CO2 associated with a computational task in the cloud, a geographical partitions based decision policy has been proposed to compute on a greener data center. The proposed model accounts for CO2 at data centers as well as of networks in the processing of a particular task. Based on emitted CO2 level, the request should be routed to the data center where less carbon is produced for computation. Transport network and data center are taken into consideration to estimate carbon footprints. This paper contains matter which shows how a geographical partition based algorithm decides where to route computation request to make cloud computing greener

Keywords: *Cloud computing, CO2 emission, Green cloud computing, Voronoi partitions.*

1. Introduction

The increasing availability of high-speed Internet is enabling the delivery of new network-based services to end users. While Internet-based mail services have been operating for many years, service offerings have recently expanded to include network-based storage and network-based computing. These new services are being offered both to corporate and individual end users (Weiss, 2007). Services of this type have been called “cloud computing services”. Cloud computing offers potential, financial benefits to end users. In this service model end users can share a large, centrally managed pool of storage and computing resources, rather than owning and managing their own systems. This service model comprises several data centers at different geographical locations to service end users. But while its financial benefits have been widely discussed, the large shift in energy usage in a cloud computing model has received little attention. Cloud computing increase network traffic and associated energy consumption and in turn increase carbon footprint in the environment. Recently the carbon emissions associated with powering DCs have become important. Greenpeace report (Greenpeace, 2010) the carbon emissions of selected DCs and the percentage of their electricity generated by power plants that use fuels which emit a relatively large amount of carbon. In addition, it is also very important to consider the required energy to transport data to and from the end users. The CO2 emission of the network of a data center is modest part of the total CO2 of the data center (Tall A, Grosso P and Bomhof F, 2013).

From a user’s point of view, minimizing the load on the environment is equivalent to looking for a green data center, i.e a data center which has a low power usage effectiveness (PUE). It is found that many data centers come forward advertise their greenness based on PUE (Power Usage Effectiveness) factor. But to recognize the data center greenness, PUE is not the only factor to consider. To determine the amount of CO2 emitted for a given task, the energy sources powering a data center and the network used to move the data are also important. Many researchers have proposed algorithms to minimize carbon emission in cloud service model. There have been some proposals to use locally generated clean energy (Liu, Lin, Wierman, Low and Andrew, 2011) or employ load balancing based upon the carbon intensity of the electricity supplier (Doyle, O’Mahony, and Shorten, 2011). These proposals, however, do not consider the carbon emitted as a result of packets traveling across the network from the client to the server. While the energy consumed by the networking equipment as part of the cloud computing has been analyzed (Baliga, Ayre, Hinton and Tucker, 2011), additional analysis is required to examine the total carbon emission caused by a cloud computing system.

*Corresponding author: **Manish Kumar Singh**

It is presented here a geographical partitions based framework that decides where to perform a task based on overall emitted CO_2 level. The proposed framework is not going to consider the only CO_2 of data centers, but also estimates the CO_2 emission of the transport network between them. In the discussed framework, transporting of data to remote data center is preferred then transport network also plays an important role to count for carbon emission. This means that if the proposed decision framework introduced in this paper will be applied to all tasks of a data center, the total CO_2 emission will decrease.

In this paper, the computational scenario has been focused. Input data and output play an important role in estimating carbon cost in data centers and transport networks. For the computation in data centers, it is assumed that all data centers have same internal architecture.

2. Literature Review

There have been a number of proposals which consider the cost of carbon emission in our environment. Liu and Lin (2011) proposed the model to subtract locally generated clean energy from the energy cost calculation to allow data center which has clean energy generation facilities to service more load. Doyle and shorten (2011) describe an algorithm that minimizes a cost function containing the carbon intensity of the electricity supplier of the data center and average job time. Moghaddam and Cheriet (2011) attempt to use a genetic algorithm-based method with virtual machine migration to lower the carbon footprint of the cloud. Gao and Curtis (2012) use a flow optimization based framework to control the three-way trade-off between average job time, electricity cost, and carbon emissions. In this paper work proposed by Makkes, Taaal, Osseyran and Grosso (2013) has been discussed in brief and improved proposed method has been discussed in detail. Makkes *et al.*(2013) use a model to estimate carbon footprint to decide where to compute task. But this system does not count carbon cost from the source of the request to the local data center. Additionally, this system lacks to route requests at greener data center optimally.
According to network model proposed by Makkes *et al.*(2013) we have following network structure.

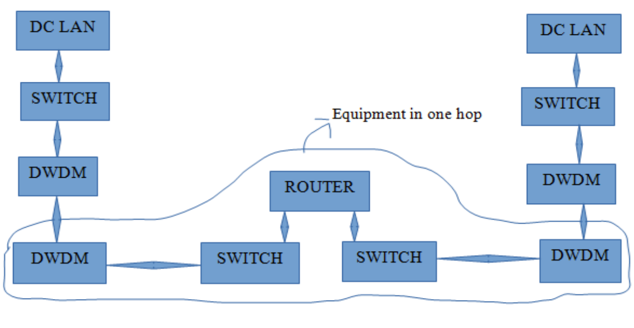

Fig. 1 Short distance Internet of 1 hop between two data centers

If the task is accompanied with Nin GByte of input data, this data will always be transferred through the transport network. So the transport cost would be

$$E_{transport\_internet\,(Nin)} = \frac{PUE_N}{U}\left\{\left(\frac{2P_{switch}}{C_{switch}} + \frac{2P_{dwdm}}{C_{dwdm}}\right) + \left(\frac{2P_{swich}}{C_{switch}} + \frac{P_{router}}{C_{router}} + \frac{2P_{dwdm}}{C_{dwdm}}\right) * n_{hops}\right\}\frac{8N_{in}}{3600}\ [KWH] \quad (1)$$

Transport cost of a task in LANs would be

$$E_{LAN(Nin)} = \frac{PUEd}{U}\left(\frac{P_{host}}{C_{host}} + 3\frac{P_{swich}}{C_{switch}} + \frac{P_{router}}{C_{router}} + \frac{2P_{firewall}}{C_{firewall}}\right) * \frac{8N_{in}}{3600}\ [KWH] \quad (2)$$

Processing Cost of a task at host would be

$$E_{processing} = PUE_d \times P_{computer\_host} . T_{processing} [KWH] \quad (3)$$

Where PUE_N = Power Usage Effectiveness of Network and equals to 2.2 (Makkes *et al.*, 2013)
PUE_d = Power Usage Effectiveness of Data Center
$T_{processing}$ = Processing Time in CPU
$P_{computer\_host}$ = Power Consumption of Computation Host
U = A Factor for the utilization of the network equipment equals to 0.5 .
P = Power Consumption by Equipment
C = Capacity of Equipment
n = Number of Hops

Makkes, Taaal, Osseyran and Grosso (2013) discuss the following metric.

Carbon cost local processing > Carbon cost network + Carbon cost remote processing (4)

According to proposed metric if carbon cost at local DC is less than remote DC then the request will not be transferred at remote DC otherwise it will and computation will be performed at the decided site and return back to the source of the request. Here Makkes *et al.* (2013) do not include carbon cost from the source of the request to the local data center . Also, next time if the same source of request sends computation request to local DC then the same procedure would be followed and hence takes more time to service a request. Here the only single path is assumed from the source of the request to DC.

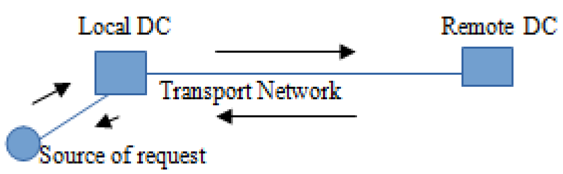

Fig. 2 Model proposed by Makkes *et al.*(2013)

In this paper work proposed by Makkes *et al*. (2013) has been improved by assuming that there are many paths from each source of the request to each data center. Here emitted carbon from the source of the request to local DC is also included.

3. Motivation

Cloud computing comprises many data centers to process user tasks. But to work all data centers need electricity from different sources. In this paradigm transport network connecting data centers and users also consume much electricity. To power, this cloud infrastructure different source of energy are used which in turn produce carbon in our environment. In 2014 the Intergovernmental Panel on Climate Change (IPCC) harmonized the carbon dioxide equivalent (CO2) findings of the major electricity generating sources used worldwide. Based on the survey of IPCC 2014, table 1 is mentioned here that show g CO2 eq/KWH for different energy sources.

Now to find out carbon emission cost C , relation given below is considered

i.e. 1kWh = Xgr. CO2 (5)

Where, the value of factor X depends on the type of energy source, e.g. X = 38 for Geothermal electricity production, and X = 490 for gas electricity production etc.
In this proposed work it is considered that data centers situated in India and China. So we have collected sources of electricity in India and China by installed capacity that is given below:

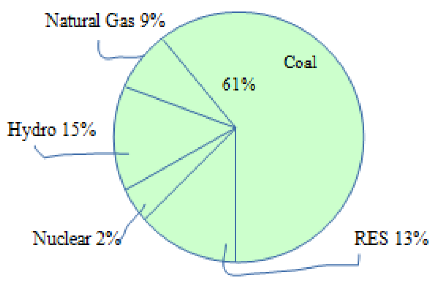

Fig. 3 Based on Govt. of India's Central Electrical Authority Report dated 31/1/16

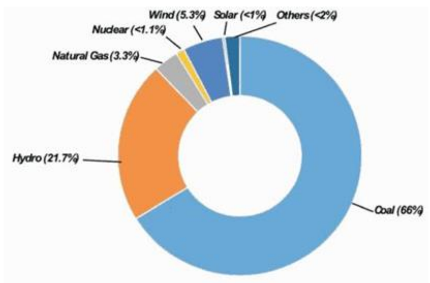

Fig. 4 Based on China Government released in 2015

Table 1 IPCC 2014 Report

Technology	Min	Median	Max
Coal – PC	740	820	910
Biomass–Confirming with coal	620	740	890
Gas – Combined Cycle	410	490	650
Biomass-Dedicated	130	230	420
Solar PV – Utility Scale	18	48	180
Solar PV - Rooftop	26	4	60
Geothermal	6	38	79
Concentrated Solar Power	8.8	27	63
Hydro Power	1	24	2200
Wind Offshore	8	12	35
Nuclear	3.7	12	110
Wind Onshore	7	11	56

4. Problem Formulation

Problem formulation is here. To do this, there is a need to have some background knowledge of what the graph is ; what the Voronoi partitions is; and how we can use all these in cloud computing.

Graph

A graph consists of a finite set of vertices, nodes or points which are connected by edges, arcs, or lines. A path is an ordered sequence of points such that any consecutive pair of points is linked by an edge in the graph. In an undirected graph, there is no direction associated with the edges meaning that there is no distinction between the two vertices associated with each edge. Hence, a path can be constructed with any edge in the graph. A weighted graph associates a label with each edge. Nodes are connected if a path exists between them.

Voronoi Partitions

Voronoi partitions are the decomposition of the plane into regions based on distance to points in a specific subset of the plane. These subsets are centered around points known as sites, generators or seeds. Each point in the set is added to a subset consisting of a site and all other points associated with this site. An abstract notion of distance between a point and the sites is used to determine which subset a point is associated with. A point is assigned to a subset if the distance to site is less than or equal to the distance to the other sites. In this paper, initial geographical partitions for the cloud is made of using the concept of Voronoi partitions. Later partitions are made by calculating the sum of carbon cost for transport network and data center that is discussed below.

Initial Geographical Partitions for Cloud

In this proposed work the set of points contains sources of requests and data centers which service end users. Initially, Voronoi partitions are used to find out where to service requests. A region represents which sources of requests a data center is servicing at a given time. The regions are made up of sources of requests which have paths available to them with lower weights than paths available to other data centers. Figure 5 shows how sources of requests are partitioned between two data centers DC_1 and DC_2 .

Problem Statement

It is supposed that |S| be a set of sources of requests distributed geographically and |D| be a data centers. Let |Q| be a fine set of points that represents either data centers or sources of requests. All theses nodes are connected by E edges in an undirected weighted manner.

Here we got , Graph G = (|Q|, |E|, |W|)

The weights are calculated as emitted carbon C to service a request.

$$\mathbf{W_k = C_k , \forall\ k \in |W|} \qquad (6)$$

Here it is assumed that the data centers have sufficient computational capacity to service requests of regions.

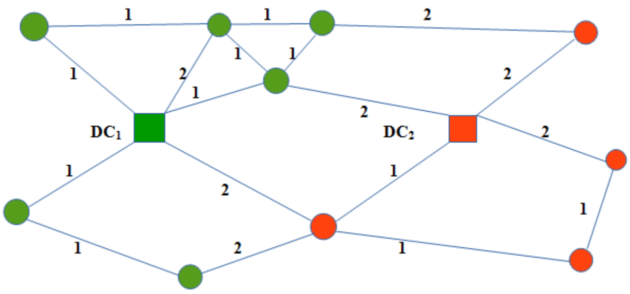

Fig. 5 Color indicates partition of regions

Here in figure 5, square boxes show data center and circles show source of request.

5. Proposed Method

The transport network plays an important role to transmit our data for computational purpose and hence it use huge energy to deliver. It is assumed that all data centers have the akin physical infrastructure to compute a task. But the transport networks and data centers may be powered by different energy sources. So in this discussion, there is a need only to calculate energy consumption by transport networks and data centers. Based on this model it will be decided where to service requests. To deliver computation through transport network two kinds of connections are available: the regular Internet and dedicated connections (light path). Regular Internet can be used by all users and dedicated connections are used for high end-users in scientific and corporate environments. In both the cases, data transfer can be very long or short. Here in this paper, we consider only Internet connection. Our proposed algorithm can also be used to find the best location to compute task in dedicated connection also.

In this proposed work it is assumed that initially, all data centers have prior knowledge in their knowledge base for regions associated with them by calculating the distance between the source of request and data center and all base information to calculate carbon emission. All data centers communicate with each other to know the partitions associated with them if there is the best route available between the source of request and data center. Suppose two data centers are i and j having initial partitions Ri and Rj and at a time t a request goes to data center i then based on following algorithm new partitions are made and the request is routed to the corresponding data center .

1. $\mathbf{S_1 \in R_i (t) \cup R_j (t)}$
2. **For** $\mathbf{S_1 \in |S|}$
3. **if** $\mathbf{(W_{i\_in} + W_{i\_out} + C_{lan\_i\_in} + C_{lan\_i\_out} + C_{processing\_i}) > (W_{j\_in} + C_{lan\_j\_in} + W_{j\_out} + C_{lan\_j\_out} + C_{processing\_j})}$
4. **Then** $\mathbf{S_1 \in R_j (t+1)}$
5. **if** $\mathbf{(W_{i\_in} + W_{i\_out} + C_{lan\_i\_in} + C_{lan\_i\_out} + C_{processing\_i}) < (W_{j\_in} + C_{lan\_j\_in} + W_{j\_out} + C_{lan\_j\_out} + C_{processing\_j})}$
6. **Then** $\mathbf{S_1 \in R_i (t+1)}$
7. **End For**

Here in this algorithm,
W_{i_in} = *carbon emitted during input data transport to data center i from source* S_1
W_{i_out} = *carbon emitted during output data transport from data center i to source* S_1
$C_{lan\_i\_in}$ = *carbon emitted through LAN of data center i with input data*
$C_{lan\_i\_out}$ = *carbon emitted through LAN of data center i with output data*
W_{j_in} = *carbon emitted during input data transport to data center j from source* S_1
W_{j_out} = *carbon emitted during output data transport from data center j to source* S_1
$C_{lan\_j\_in}$ = *carbon emitted through LAN of data center j with input data*
$C_{lan\_j\_out}$ = *carbon emitted through LAN of data center j with output data*
$C_{processing\_i}$ = *carbon emitted during computation of task at data center i*
$C_{processing\_j}$ = *carbon emitted during computation of task at data center j*

Here S_1 is a source of request that belongs to |S|. Initially, partitions are temporary and based on distance. When a request from source S_1 goes to a data center say i that cover initial region R_i then according to proposed algorithm W_{i_in}, $C_{lan\_i\_in}$, $C_{processing\_i}$, $C_{lan\_i\_out}$, W_{i_out} is calculated for source S_1. Data center i knows all available routes from source $S_1$to other data centers also as all data centers communicate and share their partitions information. So knowledge base present at data center i is used to calculating W_{j_in}, $C_{lan\_j\_in}$, $C_{processing\_j}$, $C_{lan\_j\_out}$, W_{j_out} for source $S_1$to other data center j. Here W_j is lowest among all weight for all available route to data center j. Then (W_{i_in} + W_{i_out} + $C_{lan\_i\_in}$ + $C_{lan\_i\_out}$ + $C_{processing\_i}$) is compared with value (W_{j_in} + $C_{lan\_j\_in}$ +W_{j_out} + $C_{lan\_j\_out}$ + $C_{processing\_j}$). If (W_{i_in} + W_{i_out} + $C_{lan\_i\_in}$ + $C_{lan\_i\_out}$ + $C_{processing\_i}$) is greater than (W_{j_in} + $C_{lan\_j\_in}$ +W_{j_out} + $C_{lan\_j\_out}$ + $C_{processing\_j}$) then request is computed at data center j and source of request S_1 now belong to region R_j. From next time if the same source of request sends data to calculate then request now go to data center j and reduce servicing time as well as carbon emission too.

If (W_{i_in} + W_{i_out} + $C_{lan\_i\_in}$ + $C_{lan\_i\_out}$ + $C_{processing\_i}$) is less than (W_{j_in} + $C_{lan\_j\_in}$ +W_{j_out} + $C_{lan\_j\_out}$ + $C_{processing\_j}$) then request is computed at data center i and source of request S_1 belong to region Ri.

The prediction of energy consumption in this paper is based on the work of Baliga *et al*. (2011). Now energy costs in kWh can be mapped given by Equations 1 into an equivalent carbon emission cost C in terms of grams of CO2 produced. For equation 5, the equivalent carbon emission cost is given below,

$$W = C_{transport\_internet} = X_{transport\_internet} \cdot E_{transport\_internet\ (Nin/out)} \quad (7)$$

$$C_{lan(Nin)} = X_{data\_center} . E_{LAN(Nin/out)} \quad (8)$$

$$C_{processing} = X_{data\_center} . E_{processing} \quad (9)$$

6. Analysis and Results

A comparative analysis between work discussed by Makkes *et al*. (2013) and proposed work in this paper is shown. Here one data center is situated in Mumbai, India, and the second one is situated in Guangzhou, China.

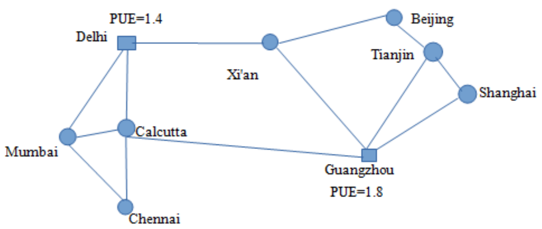

Fig. 6 Linear representation of data centers and source of requests

Before calculating carbon emission first, calculate how much energy is required from different sources to transport computation. To calculate there is a need of power per capacity record presented in table 2 for each equipment.

Table 2: Power per Capacity for the Different Components in our Model

Equipment	Power per capacity [kW/Gb/s]
Host data storage	0.28
Router	0.012
Ethernet switch	0.023
Firewall	0.016
DWDM terminal node	0.0034

For the comparative analysis suppose it is assumed that a request is originating from Calcutta and goes to data center situated in Delhi as the initial partition is based on distance covered. A number of hops between the source of the request and Delhi-based data center are 3, From Delhi to Guangzhou is 9 and from Calcutta to Guangzhou is 5 . PUE of both data centers is 1.4 and 1.8 as shown in Fig. 6 . It is assumed that Delhi-based data center is powered by electricity produced from coal (820 gr. CO2/kWh) and Guangzhou-based data center is powered by electricity produced from Hydro (24 gr. CO2/kWh). For the analysis, it is known that $T_{processing}$ = ¼ hours for N_{in}= 1 GByte and N_{out} = 1/2 GByte . The value $P_{computer\_host}$ = 0.3555 kW (Baliga, Ayre, Hinton and Tucker, 2011).

Existing work analysis

Now in the case of Makkes *et al*. (2013) when request goes to Delhi based data center values for $E_{transport\_internet}$, $E_{LAN(Nout)}$, $E_{LAN(Nin)}$ and $E_{processing}$ are $E_{transport\_internet}$ = 0.002417 KWH , $E_{LAN(Nout)}$ =0.001222 KWH, $E_{LAN(Nin)}$ = 0.002445 KWH and $E_{processing}$ = 0.12425 KWH. Data centers know the nearest remote data center to check for its greenness. So it considers shortest route available from local data center to remote data center. In fig. 6 the route is going through Xi'an city and 9 hops are there. The scheduler present at Delhi based data center finds values $E_{transport\_internet\_to\_remote}$= 0.006218 KWH/Gbyte, $E_{LAN(Nin)\_remote}$ = 0.003144 KWH/Gbyte and $E_{processing\_remote}$=0.15975 KWH. Now calculated carbon emission at local data center is compared with total carbon emission during relocation of computation at remote data center. Total carbon emission at local data center that is ($C_{local\_lan(Nin)}$ + $C_{local\_lan(Nout)}$ + $C_{processing}$) = 104.891 gr. CO2. Here carbon emission between source to local data center is not included as Makkes *et al*. (2013) assume source to be directly connected with local data center. Further carbon emission during computation at remote data center is equal to ($2 \times C_{local\_lan(Nout)}$ + $C_{local\_lan(Nin)}$ + $C_{transport\_internet(Nin)}$ + $C_{transport\_internet(Nout)}$ + $C_{remote\_lan(Nin)}$ + $C_{remote\_lan(Nout)}$ + $C_{processing}$) that yields 13.1876 gr. CO2. Here to calculate $C_{transport\_internet(Nin)}$ and $C_{transport\_internet(Nout)}$ it is assumed that from Delhi to Guangzhou 20% of transport network is in India and rest is in China. X factor for network is found by power mixes shown in fig. 3 and fig. 4. Now if request would go to Guangzhou then calculated $X_{transort\_internet}$ = $0.20 \times 550.09 + 0.80 \times 563.74 = 561.01$ gr. CO2/kWh . Here , ($C_{local\_lan(Nin)}$ + $C_{local\_lan(Nout)}$ + $C_{processing}$) > ($2 \times C_{local\_lan(Nout)}$ + $C_{local\_lan(Nin)}$ +$C_{transport\_internet(Nin)}$ + $C_{transport\_internet(Nout)}$ + $C_{remote\_lan(Nin)}$ + $C_{remote\_lan(Nout)}$ + $C_{processing}$)

Now from the analysis it is clear that request is computed at remote data center situated in Goungzhou, China as emitted carbon is 13.1876 gr. CO2.

Proposed work analysis

According to proposed algorithm to minimize carbon emission it is given that if (W_{i_in} + W_{i_out} + $C_{lan\_i\_in}$ + $C_{lan\_i\_out}$ + $C_{processing\_i}$) is greater than (W_{j_in} + $C_{lan\_j\_in}$ +W_{j_out} + $C_{lan\_j\_out}$ + $C_{processing\_j}$) then request is computed at data center j and source of request S_1 now belong to region R_j. The calculation for (W_{i_in} + W_{i_out} + $C_{lan\_i\_in}$ + $C_{lan\_i\_out}$ + $C_{processing\_i}$) yields 106.88 gr. CO2. Same calculation for(W_{j_in} + $C_{lan\_j\_in}$ +W_{j_out} + $C_{lan\_j\_out}$ + $C_{processing\_j}$) yields 7.047 gr. CO2 (it is assumed that from Calcutta to Guangzhou 20% of the transport network is in India and rest is in China). So here it is clear that computation is going to calculate at Guangzhou data center and the source of a request from Calcutta is included in a region serviced by data center present in Goungzhou. Next time if the same location generates request then it does not go to Delhi data center but serviced by Goungzhou data center and in turns save service time also.

From the analysis, it is clear that if the same source of the request is serviced by Makkes *et al*. (2013) model, it costs more carbon (13.1876 gr. CO2.) in the environment. In comparison proposed work shows that carbon emission will be less (7.047 gr. CO2) as shown in figure 7.

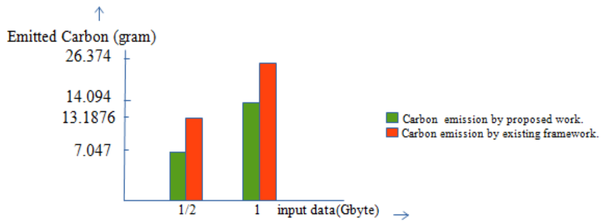

Fig. 7 Graph showing carbon emission by frameworks

Here there is a calculation of carbon emitted during transportation for Internet connection. In the same way, anyone can calculate carbon emission for dedicated connection also.

Conclusion

In this proposed work it is shown that how geographical partitions based framework is used to decide where to compute the task to minimize the carbon emission. It is shown that all data centers have information regarding partitions associated with them. The knowledge base present at each data center is used to compute weights for transporting paths and carbon emission at data centers . Based on these weights and emitted carbon the requesting source request should be routed. Here we have made target to minimize the carbon emission but so many factors are there to consider depending on the need to route the request to any data center. Additionally, here the network topology also matters to compute the cost. So in future due to rapid changes in networking, it's a challenging task to find the best way to minimize the costs.

References

Taal A, Grosso P, Bomhof F (2013). Transporting bitsor transporting energy: does it matter? Retrieved from http://www.surf.nl/

J. Baliga, R. W. A. Ayre, K. Hinton, and R. S. Tucker (2011). Green cloud computing: Balancing energy in processing, storage, and transport, Proceeding of the IEEE, pp. 149–167.

J. Doyle, D. O'Mahony, and R. Shorten (2011). Server selection for carbon emission control, in Proceeding of ACM SIGCOMM Workshop on Green Networking, Toronto, pp. 1–6.

P. X. Gao, A. R. Curtis, B. Wong, and S. Keshav (2012). It's not easy being green, in Proceedings of SIGCOMM, Helsinki, 13 – 17, pp. 221–222.

B. Hayes (2008). Cloud computing, Commun. ACM, pp. 9–11.

Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. Andrew(2011). Geographical load balancing with renewable, in Proceeding of GreenMETRICS, San Jose, 7 - 11 June 2011, pp. 1–5.

Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. Andrew (2011). Greening geographical load
balancing, in Proceeding of SIG- METRICS, San Jose, pp. 233–244.

Marc X Makkes, A. Taaal, A. Osseyran and P. Grosso (2013). A decision framework for placement of applications in clouds that minimizes their carbon footprint, Journal of cloud computing advances, Systems, and applications.

F. F. Moghaddam, M. Cheriet, and K. K. Nguyen (2011). Low Carbon Virtual Private Clouds, in Proceedings of IEEE International Conference on Cloud Computing, Washington DC, 4 – 9, pp. 259–266.

A. Weiss (2007) .Computing in the clouds, netWorker, pp. 16–25.

Greenpeace (2010). Make IT green cloud computing and its contribution to climate change, Retrieved from http://www.greenpeace.org/international/Global/international/planet-2/report/2010/3/make-it-green-cloud- computing.pdf.

https://en.wikipedia.org/wiki/Life-cycle_greenhouse-gas_emissions_of_energy_sources