Carbon-efficient Virtual Machine Placement in Cloud Datacenters over Optical Networks

Jiaxiang Zhang[1], Wei Wang[1], Yanran Xiao[1], Liyazhou Hu[2], Yajie Li[1], Yongli Zhao[1], Jie Zhang[1]
[1]School of Electronics Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]Macau University of Science and Technology, Macau 999078, China
jie.zhang@bupt.edu.cn

***Abstract*—We propose an ILP scheme for virtual machine (VM) placement in cloud datacenters(DCs) to minimize the overall carbon emissions of cloud DCs while satisfying the communication demands and latency requirements of VM services.**

Keywords—VM placement, datacenters, carbon emissions

I. INTRODUCTION

In recent years, cloud DCs have undertaken the important task of storing and processing massive amounts of data, and have become one of the most important infrastructures in today's society. Many large IT companies (such as Google and Microsoft) have multiple geographically distributed DCs to meet their customers' cloud service requirements. These DCs can host from a few servers to thousands of servers. However, when users enjoy high-quality and convenient cloud services, DCs are consuming a significant amount of energy (1.5% of global energy consumption) and generating a lot of carbon emissions [1].

Cloud DCs utilize a technology known as virtualization [2], which makes it possible to share the physical resources of a server among various VMs. VM placement is an important issue in cloud DCs, and it involves how to reasonably place VMs on physical servers to achieve optimal resource utilization, performance, and service quality. Many studies have noticed that carbon emissions can be reduced through effective VM deployment methods[3][4]. They basically focus on carbon emissions and electricity prices, resource utilization, and so on. For example, [5][6] consider the trade-off of electricity cost and emission reduction budget in virtual machine deployment, and use Lyapunov optimization technique to design and analyze carbon-aware control framework. The heuristic algorithm proposed in [7] focuses on minimizing the carbon emissions of federated cloud DCs through efficient scheduling and migration of VMs. [8] introduces Google's system for global Carbon-Intelligent Compute Management, which actively minimizes electricity-based carbon footprint and power infrastructure costs by delaying temporally flexible workloads. The above studies also effectively reduce carbon emissions, but they do not consider the cross-datacenter bandwidths and network latency requirements of virtual machines deployed in remote locations.

In this paper, we develop an integer linear programming model to find an optimal solution to reduce carbon emissions, taking into account the constraints of bandwidth capacity and network latency in inter-datacenter optical. Numerical results show that the latency requirements, the bandwidth demand of VMs, and the hardware capacity of DCs have a significant impact on overall carbon emissions.

II. CLOUD DATACENTER CARBON EMISSION MODEL

In this section, we introduce the fundamental components of cloud DCs and discuss their impact on the carbon emissions of VM placements.

A. Basic Components of Cloud Datacenter

Cloud DCs consist of many geographically distributed DCs and inter-datacenter optical networks. Each DC contains many server devices for computing and storage and many network devices for communications. DCs are interconnected through optical networks, for long-haul communications.

In this article, we simplify the resources of the DC infrastructures into three types: CPU, memory, and storage. The bandwidth and latency of optical connectivity between different DCs are considered as fixed.

B. Datacenter Carbon Emissions

The DC mainly includes three subsystems: IT equipment, cooling system, and power distribution system. Power usage effectiveness (PUE) is an industry-standard to measure the energy efficiency of DC. Its definition is shown in Eqn. (1). The PUE value of each data center is different. A DC with a large PUE value tends to consume more energy for powering the same amount of computing resources.

$$PUE = \frac{Total\ energy\ consumption\ of\ DC}{energy\ consumption\ of\ IT\ equipment} \tag{1}$$

The energy grid uses different types of energy sources in different regions, leading to differences in the cleanliness of the supplied energy. As a result, the cleanness of the energy that is consumed by DCs in different regions might also be heterogeneous. Here, we use the carbon effectiveness (CE, carbon emissions per kWh of electricity) as a metric for measuring electricity cleanliness. By calculating the weighted contribution of each fuel type used to generate electricity, we can calculate CE at different locations, as Eqn. (2).

$$CE = \sum c_k \times p_k \tag{2}$$

In Eqn. (2), p_k represents the percentage of type k fuel in total power generation, and c_k represents the carbon emission of type k fuel. Its specific value is given in Table Ⅰ, which takes g/kWh as the calculation unit.

TABLE I. CARBON DIOXIDE EMISSION PER KILOWATT-HOUR FOR THE MOST COMMON FUEL TYPES [2].

Fuel Type	Nuclear	Coal	Gas	Oil	Hydro	Wind
CO2 g/kWh	15	968	440	890	13.5	22.5

The PUE and CE are different for each DC. The carbon emissions generated by the same VM service mainly depend on the PUE and CE. Therefore, we define carbon intensity (CI) as Eqn. (3):

$$CI = CE \times PUE \quad (3)$$

Therefore, if VM can be placed in DC with low CI while meeting service requirements, carbon emissions will be significantly reduced.

C. Virtual Machine Placement

In this article, our primary consideration in terms of VM placement is to minimize the total carbon emissions of the cloud DCs. In Fig. 1, we can see the DCs in three different colors. Green DCs are low carbon intensity DCs, orange DCs have slightly higher carbon intensity, and red DCs have the highest carbon intensity.

The VM is used to serve users' APP requests and requires that the client and VM always maintain a certain amount of communication bandwidth. As shown in Fig. 1, all APP requests of the user are first aggregated to the user-side access point and the local DC. We call this DC the source point of the requests for the target VM. The local DC will further redirect the requests to the target VMs, which could be placed in a local DC or deployed to a remote DC. We call the DC that hosts the VM the destination DC. For example, the VM corresponding to the yellow APP request in Fig. 1 is deployed in a remote, greener DC. Note that if the VM is placed in a remote DC, there is network latency between the source point to the destination DC.

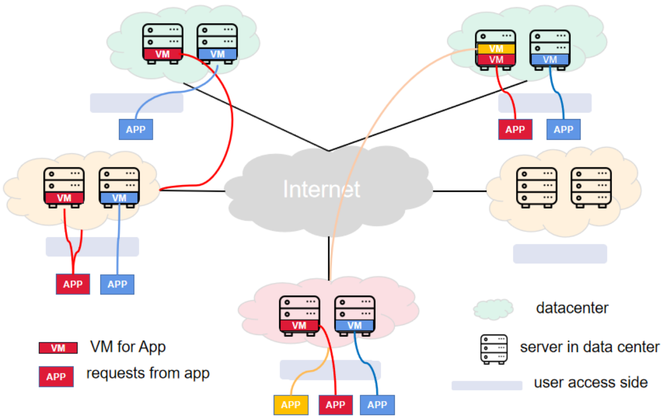

Fig. 1. VM placement in the cloud datacenter

III. Integer Linear Program Model for Carbon Emission Rate Minimization

This section formulates an ILP model that assigns a target data center to each VM, with the goal of reducing the carbon emissions of deploying VMs, a goal that infrastructure operators strive to accomplish. Carbon emissions mean more carbon taxes for infrastructure operators, but more importantly, more harm to the environment.

A. Problem Statement:

Given a group of DCs (including the CPU, memory, and storage capacity of each DC), The PUE and CE of each DC, the network propagation latency and bandwidth capacity of each pair of DCs, a group of VMs (including the CPU, memory, and storage demand of each VM), the source point of each VM, network latency threshold of each VM, the cross-DC bandwidth demand of each VM, the power consumption of each VM, we determine the destination DC for each VM.

B. Input Parameters

V: set of VM

D: set of DC

CPU_d: CPU capacity of DC d, $d \in D$

MEM_d: memory capacity of DC d, $d \in D$

STO_d: storage Capacity of DC d, $d \in D$

PUE_d: PUE of DC d, $d \in D$

CE_d: CE of DC d, $d \in D$

CPU_v: CPU required by VM v, $v \in V$

MEC_v: memory required by VM v, $v \in V$

STO_d: storage required by VM v, $v \in V$

T_v: latency threshold by VM v, $v \in V$

BW_v: cross-DC bandwidth required by VM v, $v \in V$

SRC_v: the source DC of the VM v, $v \in V$

P_v: power consumption of a VM v, $v \in V$

$NT_{s,d}$: network latency between DC s and d, $s \in D, d \in D$

$C_{s,d}$: bandwidth capacity between DC s and DC d, $s \in D$, $d \in D$

C. Variables:

X_v^d: binary, whether VM v is deployed in DC d, $d \in D$

D. Objective: Minimize Carbon Emission:

$$\text{Min} \sum_{d \in D} \sum_{v \in V} X_v^d \times P_v \times PUE_d \times CE_d \quad (4)$$

E. Constraints:

$$\sum_{d \in D} X_v^d = 1, \forall v \in V \quad (5)$$

$$\sum_{v \in V} X_v^d \times CPU_v \le CPU_d, \forall d \in D \quad (6)$$

$$\sum_{v \in V} X_v^d \times MEM_d \le MEM_d, \forall d \in D \quad (7)$$

$$\sum_{v \in V} X_v^d \times STO_d \le STO_d, \forall d \in D \quad (8)$$

$$\sum_{d \in D} X_v^d \times NT_{SRC_v,d} \le T_v, \forall v \in V \quad (9)$$

$$\sum_{v \in V} X_v^d \times BW_v + X_v^d \times BW_v \le C_{s,d}, \forall s, d \in D \quad (10)$$

Equation (5) enforces that A VM can only be deployed in one DC .Constraint (6)(7)(8)guarantee that all the VMs placed at each DC cannot exceed the host's hardware capacity. Constraint (9) enforces that the latency required for VM is less than the latency between the source DC and the destination DC. Constraint (10) enforces that the total bandwidth occupied by VM communication cannot exceed the total communication bandwidth capacity between DCs

IV. Simulation Setup and Results Analysis

In this section, we introduce the settings of our simulation and analyze the factors affecting carbon emissions in details.

A. Simulation Setup

The topology used in the simulation is the NSFNet 14-node topology. We obtained 2020 carbon emissions data for each state's electricity supply from the US Energy Information Administration website [9]. It measures kilograms of energy-related carbon dioxide per million British thermal units. We take the carbon emission data of the state where each data center is located as the CE value of the DC. The specific data are shown in Table II. We obtained the time latency between different data center locations from the website [10], and the time latency between topological nodes has been marked on the link in Fig. 2, in unit of ms.

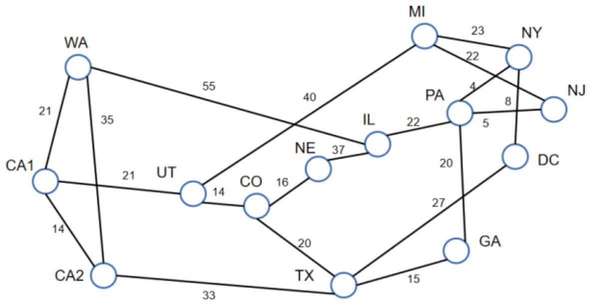

Fig. 2. Overview of a NSFNet topology

In the simulation, we consider two types of VMs (type-A and type-B). The capacity and energy consumption of type-B VM is twice of that for type-A VM. And set the power consumption of the VM type A as 1 unit power consumption. A total of 14,000 VM requests are generated, and they are evenly distributed in cloud DCs.

TABLE II. CE VALUES OF DIFFERENT DATACENTERS

DC-Name	**WA**	**CA1**	**CA2**	**UT**	**CO**	**TX**	**NE**
CE	33.6	49.2	49.2	66.7	56.9	46.8	51.5
DC-Name	**IL**	**MI**	**GA**	**PA**	**NY**	**DC**	**NY**
CE	43.2	54.4	46.9	46.8	43.5	49.6	48.2

B. *Result analysis*

First, we explore the relationship between DC carbon emissions and latency threshold of VMs. Here we uniformly require the latency threshold of all VMs. We plot three lines in Fig. 3, where the cross-DC bandwidth requirements for VM are fixed as 1 unit, and the bandwidth capacities between DCs for the three lines are set as 100, 120, and 140 units, respectively. When the VM service latency threshold is 0, all VMs must be deployed locally. In this case, carbon emissions are high. We can see from the figure that in all three cases, the total amount of carbon emissions decreases as the delay requirement decreases. Because when latency requirements are reduced, VM can find remote but greener data centers.

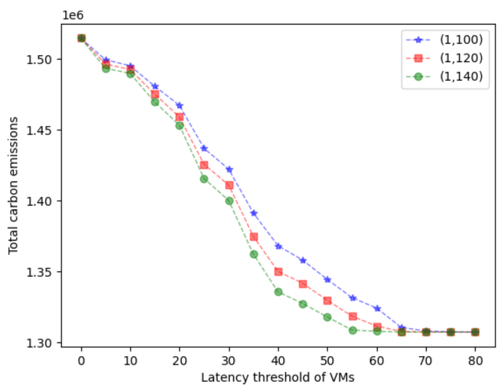

Fig. 3. Carbon emission variation with latency requirement reduction

We can also see that the latency requirement is in the range of 10-40ms, and the carbon emission drops rapidly. Because most of the network latency between DCs are within this range. When the latency requirement becomes above 50, the carbon emissions change very slowly. At this point, there is no further need for the VMs to look for other green DC.

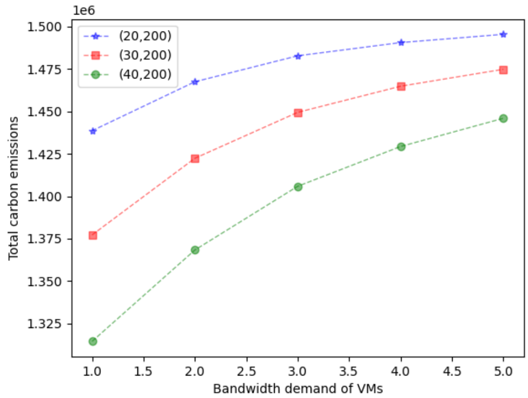

Fig. 4. Carbon emission variation with VMs bandwidth demand

In Fig 4, we plot the carbon emission with the increase of VMs bandwidth demand under the condition that the bandwidth capacity between DCs is constant. We discuss three situations in the figure, the bandwidth capacity between DCs is 200 units, and the VM latency requirements are 20, 30, and 40 ms. As can be seen from the figure, carbon emissions increase as the VM bandwidth demand increases. We can understand it this way, for example, on a point-to-point link, When the bandwidth requirements of VMs increase, the number of VM services carried by the same link will decrease Therefore, the number of VMs allowed to migrate to other DCs will be reduced at this time.

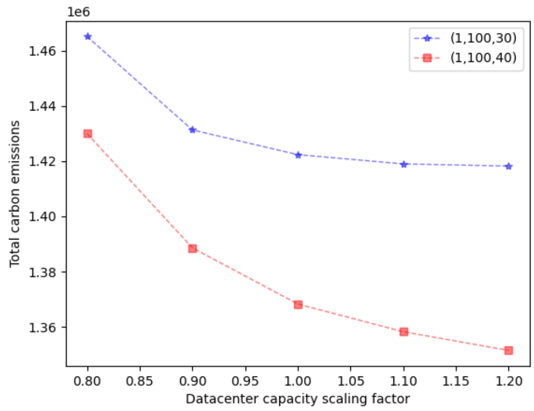

Fig. 5. The relationship between carbon emissions and datacenter capacity

In Fig. 5, we explore the impact of DC capacity on carbon emissions. We set the scaling factor of the DC capacity to be 0.8, 0.9, 1, 1.1, and 1.2 respectively. We also conducted experiments under two conditions. In the legend, the three parameters are VM bandwidth demand, DC bandwidth capacity, and VM latency threshold. As can be seen from the figure, when the capacity shrinks, the carbon emission will increase; when the capacity increases, the carbon emission will decrease. This is mainly because the change of the capacity of the green DC will make more rooms on the greener DC to take more VMs.

V. Conclusion

Our work examined VM placement problem in cloud DCs to minimize carbon emissions. An integer linear programming model was established to optimize the placement of VMs while meeting the latency threshold and bandwidth demand requirements of VM services. The simulation results showed that the VM latency and communication requirements, and DC capacity all affect the carbon emissions in VM deployment.

Acknowledgment

This work is supported in part by the National Natural Science Foundation of China (under grants 62101063 and 62021005), and in part by the Shenzhen Virtual University Park (under grant Szvup010).

References

[1] Avgerinou, M., Bertoldi, P. and Castellazzi, L., 2017. Trends in datacenter energy consumption under the European code of conduct for datacentre energy effificiency. Energies, 10(10), p.1470.

[2] Tom Brey, Larry Lamers,"Using Virtualization to improve Data Center Efficiency," The Green Grid, Whitepaper 19, 2009

[3] B. Wadhwa and A. Verma, "Carbon efficient VM placement and migration technique for green federated cloud datacenters," 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Delhi, India, 2014, pp. 2297-2302, doi: 10.1109/ICACCI.2014.6968597.

[4] K. Kinkar, P. Bhosale, A. Kasar and V. Gutte, "Carbon Footprint Analysis: Need for Green Cloud Computing," 2022 International Conference on Electronics and Renewable Systems (ICEARS), Tuticorin, India, 2022, pp. 1-6, doi: 10.1109/ICEARS53579.2022.9752341.

[5] Z. Zhou et al., "Carbon-Aware Load Balancing for Geo-distributed Cloud Services," 2013 IEEE 21st International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems, San Francisco, CA, USA, 2013, pp. 232-241, doi: 10.1109/MASCOTS.2013.31.

[6] Z. Zhou, F. Liu, R. Zou, J. Liu, H. Xu and H. Jin, "Carbon-Aware Online Control of Geo-Distributed Cloud Services," in *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 9, pp. 2506-2519, 1 Sept. 2016, doi: 10.1109/TPDS.2015.2504978.

[7] B. Wadhwa and A. Verma, "Carbon efficient VM placement and migration technique for green federated cloud datacenters," 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Delhi, India, 2014, pp. 2297-2302, doi: 10.1109/ICACCI.2014.6968597.

[8] A. Radovanović et al., "Carbon-Aware Computing for Datacenters," in IEEE Transactions on Power Systems, vol. 38, no. 2, pp. 1270-1280, March 2023, doi: 10.1109/TPWRS.2022.3173250.

[9] (2013). U.S. Energy Information Administration. [Online]. Available: https://www.eia.gov/environment/emissions/state/

[10] Wondernetwork, [online] Available: https://wondernetwork.com/pings