

High-revenue Online Provisioning for Virtual Clusters in Multi-tenant Cloud Data Center Network

Shuaibing Lu^{1,2*}, Zhiyi Fang¹, and Jie Wu²

¹College of Computer Science and Technology,
Jilin University, China

[e-mail: lushuaibing11@163.com, fangzy@jlu.edu.cn]

²Department of Computer and Information Sciences, Temple University, USA
[e-mail: jiewu@temple.edu]

*Corresponding author: Shuaibing Lu

*Received April 29, 2018; revised August 15, 2018; accepted August 20, 2018;
published March 31 2019*

Abstract

The rapid development of cloud computing and high requirements of operators requires strong support from the underlying Data Center Networks. Therefore, the effectiveness of using resources in the data center networks becomes a point of concern for operators and material for research. In this paper, we discuss the online virtual-cluster provision problem for multiple tenants with an aim to decide when and where the virtual cluster should be placed in a data center network. Our objective is maximizing the total revenue for the data center networks under the constraints. In order to solve this problem, this paper divides it into two parts: online multi-tenancy scheduling and virtual cluster placement. The first part aims to determine the scheduling orders for the multiple tenants, and the second part aims to determine the locations of virtual machines. We first approach the problem by using the variational inequality model and discuss the existence of the optimal solution. After that, we prove that provisioning virtual clusters for a multi-tenant data center network that maximizes revenue is NP-hard. Due to the complexity of this problem, an efficient heuristic algorithm OMS (Online Multi-tenancy Scheduling) is proposed to solve the online multi-tenancy scheduling problem. We further explore the virtual cluster placement problem based on the OMS and propose a novel algorithm during the virtual machine placement. We evaluate our algorithms through a series of simulations, and the simulations results demonstrate that OMS can significantly increase the efficiency and total revenue for the data centers.

Keywords: Data center network, online resource provisioning, multi-tenancy scheduling, virtual cluster provision

A preliminary version of this paper appeared in IEEE INFOCOM 2018 workshop on CCSNA, April 15-19, Honolulu, USA. This version includes the NP-hardness proof of virtual clusters provision problem for the revenue maximization. It also includes the problem formulation by using the variational inequality model, and discuss the existence of the optimal solution. This work was done by the lead author during her stay as a visiting scholar of the China Scholarship Council at Temple University, Philadelphia, U.S. This research was supported in part by NSF grants CNS 1757533, CNS1629746, CNS 1564128, CNS 1449860, CNS 1461932, CNS 1460971, IIP 1439672, and CSC 20163100.

1. Introduction

Data Center Network has been widely used as the infrastructures are increasingly utilized to support the highly scaling of applications in many large enterprises. Virtualization techniques are normally utilized to achieve variable virtual architectures that co-exist without changing the topologies or protocols of the data center networks [1]. According to the feature of the tenant, we divide the resource provisioning problem into two cases: offline and online. For the offline resource provisioning problem, all information of the tenant is known in advance, including the processing time, deadline, and application topology. On the contrary, the online case stands for a theoretical framework for the resource allocation of multiple tenants. Thus, for online resource provisioning problem, all information is unknown ahead of time, and the virtual requests arrive dynamically and occupy the physical resources of the data center network for an arbitrary period of time before they depart [2, 3]. Therefore, finding an optimal solution to achieve the effectiveness and high revenue for the data center network under the online case has become a major work that focuses on research.

In this paper, we model the physical topology of the data center network as a multi-rooted tree. It consists of a group of physical machines and physical links. The capacity of the physical machine is slotted. One slot only holds one virtual machine. We suppose that each virtual request consists of a virtual cluster with properties of processing time and deadline. For one virtual cluster, it consists of one virtual switch and several virtual machines, so that the virtual machines are connected to the virtual switch through a bidirectional link with high communication capacity [4–6]. One challenge is to find an efficient scheduling scheme that can maximize the total revenue in the online case during the resource provisioning process for multiple tenants in data center networks. In order to illustrate this problem, we use an example which is shown in Fig. 1. We assume that $G_v^1 = \langle 1, 10, 3, 3 \rangle$, $G_v^2 = \langle 2, 10, 5, 10 \rangle$, and $G_v^3 = \langle 5, 10, 10, 10 \rangle$ are the virtual clusters of the tenants that request for the resource allocation and their arrival time are 0, 0, and 1. The amount of virtual machines of G_v^1 , G_v^2 , and G_v^3 are 1, 2, and 5, respectively, and the communication demand of each virtual machine is 10. The processing times of G_v^1 , G_v^2 , and G_v^3 are 3, 5, and 10, and their deadlines are 3, 10, and 10. The unit revenue of allocating one slot to each virtual machine is one-time frame. One scheduling scheme is processing the virtual clusters on a first come-first serve basis. It means that tenants with earlier arriving times have higher priorities. The execution order would be $G_v^1 \rightarrow G_v^2 \rightarrow G_v^3$. Since G_v^3 cannot be completed in the deadline, thus, the total revenue only consider the virtual clusters G_v^1 and G_v^2 , where $1a \cdot 3 + 2a \cdot 5 = 13a$. However, if we look ahead to one-time frame, and process the virtual clusters of tenants by the higher revenue first, the execution order would be $G_v^3 \rightarrow G_v^2 \rightarrow G_v^1$. The total revenue changes to $50 \cdot a = 50a$ as only G_v^3 can be scheduled. In this way, a good scheduling scheme can result in even better resource utilization. We summary our contributions as follows:

- We first formulate the online resource provisioning problem for virtual clusters in the data center network with the objective of maximizing the total revenue by using variational inequality. We prove the existence and uniqueness of the optimal solution for the online provision problem.
- We analyze the properties of our problem and propose an efficient resource allocation scheme which contains two parts: online multi-tenancy scheduling and virtual cluster

provisioning; We first prove that the virtual cluster provision problem for multi-tenant with revenue maximization is NP-hard. Then two efficient heuristic algorithms OMS and VMP-VC are proposed, which can guarantee the maximum total revenues and high efficiency.

- Extensive simulations are conducted to evaluate our proposed solutions. The results are presented from different perspectives to provide conclusions.

The remainder of this paper is organized as follows. We review related works and summarize the related research on the online resource provisioning problem for the data centers in Section 2. Section 3 describes the model and then formulates the problem. We analyze the properties of the resource allocation problem with maximizing the total revenue by using variational inequality in Section 4. Section 5 investigates the online virtual-cluster provisioning problem for multiple tenants and proposes an efficient heuristic algorithm OMS. Section 6 proposes a virtual machine placement method which is based on the physical machine clustering algorithm. Section 7 presents the experiments and the results. Section 8 presents the conclusion of this paper.

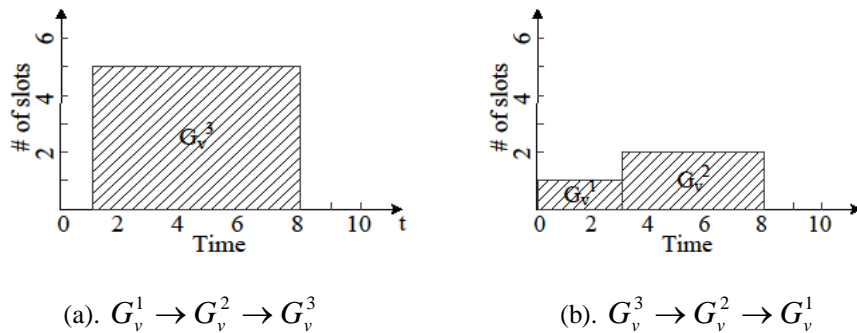


Fig. 1. An scheduling example for virtual clusters with multi-tenant.

2. Related Work

Online virtual cluster provisioning in data centers allows a great deal of flexible virtual requests for multiple tenants. This section provides a brief overview of the relevant methodologies proposed for the online resource allocation problem. We divide this problem into two parts: online multi-tenancy scheduling and virtual cluster provisioning. Online multi-tenancy scheduling determines the processing order of the virtual clusters from simultaneously many tenants in the data center network. Most studies on online scheduling only focus on the communication. [9] proposes an online virtual machine placement scheme based on re-allocation to improve the traffic distribution. This paper uses online migration to reduce the traffic congestion during the communication of the virtual machines, but online migration would produce a high cost and influence the placement of other users. [10] focuses on the scalability of network-aware virtual machine placement, which considers each pair of virtual machines of one user. It is inefficient of considering communication for each pair of virtual machines instead of the users as a whole. Cheng et al. [8] applied the Markov Random Walk (RW) model to rank the multi-tenant based on their resources and topological attributes. Markov RW is a mathematical formalization of a path that consists of a succession of random steps. This novel topology-aware ranking measure reflects the relative importance of the tenants and increases the long-term average revenue and acceptance ratio. However, since this solution is a backtracking algorithm and it has a high amount of data (traffic matrix) exchange,

the cost of this scheme is much higher than others.

Virtual cluster provisioning is another primary issue for the online resource allocation problem. For the virtual request embedding, some or all of the hardware (e.g., servers, routers, switches, and links) are virtualized [1, 12, 13, 31]. A key problem in the process of embedding is how to allocate the virtual resources to the infrastructures. This problem is also known as the virtual data center network embedding problem. Lischka et al. [15] proposed a back-tracking algorithm based on a subgraph isomorphism search method that maps physical machines and links during the same stage. This approach accounts for the link mapping constraints at each step of mapping, and it can revise a bad mapping decision by simply backtracking to the last valid mapping decision. This method produces better mapping and avoids wasting data center resources, but it requires a high time complexity for the backtracking procedure and ignores the topological structure of the data center network. Mansoor Alicherry et al. [7] developed efficient resource allocation algorithms for users in distributed clouds for the optimal selection of data centers in the distributed cloud, where the objective is to minimize the maximum distance between the selected data center networks. Li et al. [15] adopted a top-k dominating model based on GraphMap to rank the physical machines, aiming to balance these factors in order to improve resource allocation. The advantage of this method is its novel mapping algorithm TK-Match, which not only consists of two stages (the physical machine mapping stage and link mapping stage) but also maps the virtual machines in terms of physical machine ranking and the hops of the physical paths. This method focuses only on each virtual request and static data center during the mapping process. This class of algorithm, however, is not capable of supporting reconfiguration. Therefore, it cannot adapt to online requests or accommodate residual physical resources.

Another way to deal with the complexity of this problem is to formulate the embedding problem as an optimal mathematic model. For example, Papagiannithey et al. [17] provided a unified resource allocation framework for networked clouds. They first formulated the optimal networked cloud mapping problem as a mixed integer programming (MIP) problem, indicating objectives related to the cost efficiency of the resource mapping procedure while abiding by user requests for QoS-aware virtual resources. Zhang et al. [28] established two models for VN embedding: an integer linear programming model for a data center network that does not support path splitting and a MIP model when path splitting is supported. Sun et al. [20] modeled the virtual network resource allocation problem as a mathematical optimization problem with minimizing power consumption. Although the objective functions and effects of these models differ, the methods and principles are broadly similar. One of the great advantages of this class of model is the feasibility of solving the embedding problem by means of the theoretical mathematical method. However, these methods can only identify a locally optimal solution within a certain range, rather than the whole scope.

In this paper, we focus on the online virtual-cluster provisioning problem in multi-tenancy data center networks. Our objective is to maximize the revenue of the virtual clusters during the embedding process. Unlike existing algorithms, our solution deals with the requests in a real-time method for the multi-tenant and focuses on maximizing the overall revenue. Meanwhile, several constraints are jointly considered during the online resource allocation including the capacities (computation and communication) and multi-tenants' deadlines requested by multiple tenants.

3. Model and Problem Formulation

3.1 Virtual Cluster

For each user, we use one virtual cluster to represent its demands. One virtual cluster is an abstraction of a set of virtual machines and a virtual switch [2], as shown in the left part of Fig. 2. The i^{th} virtual cluster is described as $g_v^i = \langle N_v^i, L_v^i, p_i, d_i \rangle$. N_v^i and L_v^i refer to the sets of virtual machines and virtual links, where $N_v^i = \{n_v^i\}$ and $L_v^i = \{l_v^i\}$, respectively. Let G_v be the set of virtual clusters, where $G_v = \{g_v\}$. We assume that all virtual clusters need to be completely configured in their deadlines, and we use d_i to represent the deadline of virtual cluster g_v^i . Let p_i denote the processing time of g_v^i , which denotes the occupation time of the i^{th} virtual cluster. We use the same communication model of virtual machines of each virtual cluster in [4], where the link capacity is the sum of virtual machines between both communication sides, and the bandwidth requirement between virtual machines is a constant. As shown in Fig. 2, during the communication of each virtual cluster, each virtual machine can send and receive at rate B . Since one virtual switch connects n virtual machines for each virtual cluster, the bandwidth requirement of this virtual cluster will be $n \cdot B$. We suppose that the virtual cluster is unsplitable during the resource allocation. There will be no revenue on the virtual cluster if the provisioning of g_v^i is not completed before d_i .

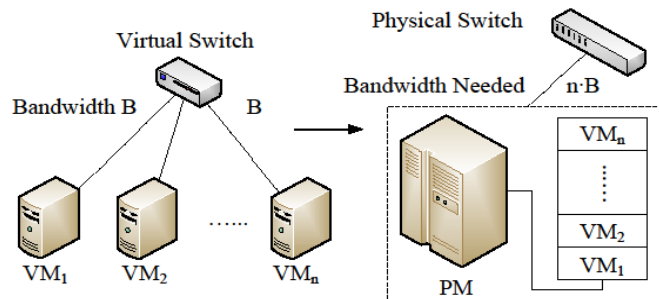


Fig. 2. Virtual cluster provisioning in the data center network.

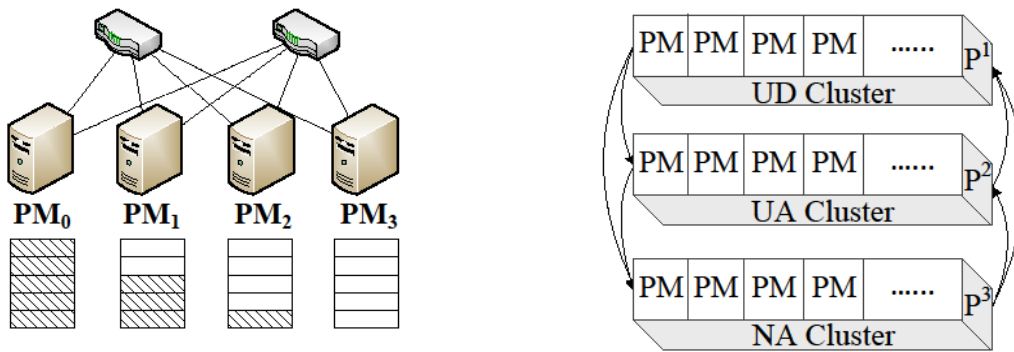
3.2 Data Center Network

We utilize a multi-rooted tree topology to represent the data center network in which each layer has the same capacities of the physical machines and physical links [7–9]. Let G_s denote the data center network, such that $G_s = \langle N_s, L_s \rangle$. N_s and L_s refer to the set of physical machines and the set of physical links, where $N_s = \{n_s\}$ and $L_s = \{l_s\}$, respectively. Let n_s^i denote the i^{th} physical machine in set N_s , i.e., $n_s^i \in N_s$, and $l_s^{i,j}$ denote the physical link between adjacent physical machines n_s^i and n_s^j in the set of L_s , i.e., $l_s^{i,j} \in L_s$. The physical machines provide storage capacity, computing capacity, or some other capacities. In this paper, however, the number of the computing units constitutes the capacity of each physical machine in this context, and the bandwidth as the capacities of physical links. Each physical machine

n_s^i , we use C_i and C_i^* to denote the initial and residual capacities of the i^{th} physical machine at time t . We use $B_{i,j}$ and $B_{i,j}^*$ to denote the initial and residual capacities of the physical link between adjacent physical machines n_s^i and n_s^j at time t .

3.3 Physical Machine Statements

In the data center network, as the arriving times and deadlines of the tenants are not the same, the remaining resources of physical machines are various at different time points. According to the remaining capacity, physical machines can be classified into three states: unutilized (UD), unutilized but still available (UA), and unavailable (NA). We define the states as follows: (i). Unutilized (UD), or $C_i^* = C_i$, indicates that all resources of the physical machine are unallocated or idle. Physical machines are initially in the UD state. In Fig. 3 (a), we use PM_i to represent the i^{th} physical machine in the data center network. Since there are no virtual machines allocated into PM_3 , the states of PM_3 is UD. (ii). Unavailable (NA), or $C_i^* = 0$, indicates that all resources of the physical machine have been occupied. In Fig. 3 (a), the state of PM_0 is NA until virtual cluster G_v release its resources. (iii). Unutilized but still Available (UA), or $C_i^* < C_i$, indicates that partial resources of physical machines have been occupied, however, the rest of the resources still can be used. In Fig. 3 (a), the states of PM_1 and PM_2 are UA, which indicates that the remaining resources of PM_1 and PM_2 can accommodate other virtual clusters that arrive at t .



(a) States of physical machines in the data center network. (b) Priorities of physical machines.

Fig. 3. Priorities of physical machines based on states.

3.4 Problem Formulation

1) Definitions

We first give a formal discussion, which is a provisioning scheme that depends on several desired resources. We define the provisioning process as $P_{G_v} : G_v \rightarrow G_s$, which indicates that the data center network allocates the resources to tenants on the basis of their requirements. We divided into two parts, virtual machine placement and bandwidth resource allocation, $P_{N_v} : N_v \rightarrow N_s$ and $P_{L_v} : L_v \rightarrow L_s$. During the provisioning process of virtual clusters, we use slots to denote the capacities of physical machines, and each slot can only hold one virtual

machine. We use the bandwidth for each link to denote the capacity of the physical link, which is measured in Gbps. Once the data center network allocates the resources to the virtual cluster, these resources are occupied by the virtual cluster throughout the processing time p_i until the end of execution (normal termination), or until the deadline d_i (forced termination). Dealing with the online resource provisioning problem can also help to solve the provisioning and scheduling problems for virtual clusters. The uncertainty depends on these properties: size (number of virtual machines), deadline (d_i), processing time (p_i), and order (processing virtual clusters for the multi-tenant at the same time point). Except for the processing order, all other information of the virtual clusters is known upon arrival. Thus, in order to maximize the total revenue obtained by the physical resources that are served, we need to find an efficient online scheme that allocates resources for tenants in the data centers.

2) Metrics

We focus on the total revenue of the data center network in the provisioning of virtual clusters, which consists of two main parts: the reward $r_v(g_v^i)$ and the cost $w_v(g_v^i)$. $r_v(g_v^i)$ is the reward of provisioning virtual cluster g_v^i , which is proportional to the processing time p_i . It is defined as

$$r_v(G_v^i) = \sum_{n_v^i \in N_v^i} C(n_v^i) + \alpha \sum_{l_v^i \in L_v^i} B(l_v^i) \quad (1)$$

Let $C(n_v^i)$ denote the one slot revenue on the physical machine that is allocated to the virtual cluster g_v^i . We suppose that the revenue of each slot grows exponentially relative to the processing time p_i of virtual cluster g_v^i in the physical machines, and the total reward of a virtual cluster is the total revenue of the allocated slots. Let $B(l_v^i)$ denote the revenue of physical links for the communication resource that is allocated to the virtual cluster i , and we suppose that the revenue of each physical link is an exponential growth function that relative to the processing time p_i . The reward $r_v(g_v^i)$ is the sum of total rewards $\sum_{n_v^i \in N_v^i} C(n_v^i)$ and $\sum_{l_v^i, j \in L_v^i} B(l_v^{i,j})$. The relative importance of the computation and communication resources are not the same for different applications; therefore, a relative factor α is defined to balance them, where $0 \leq \alpha \leq 1$. The other part is the provisioning cost of the virtual cluster g_v^i , and we use $w_v(g_v^i)$ to denote it. It contains two parts which are the energy cost of the physical machine and the communication cost of the physical link. Here, we use $E(n_v^i)$ to denote the energy cost of the physical machine. In addition, we use $H(l_v^i)$ to denote the communication cost of the physical link, which is measured in hop count. β is a relative factor that balances the communication cost and the energy cost, where $0 \leq \beta \leq 1$.

$$w_v(g_v^i) = \sum_{n_v^i \in N_v^i} E(n_v^i) + \beta \sum_{l_v^i \in L_v^i} H(l_v^i) \quad (2)$$

Let $R_v(g_v^i)$ be the total revenue of processing one virtual cluster g_v^i , which is measured by reward $r_v(g_v^i)$ excepting the provisioning cost $w_v(g_v^i)$.

$$R_v(g_v^i) = r_v(g_v^i) - \lambda \cdot w_v(g_v^i) \quad (3)$$

In order to formulate the online virtual-cluster provision problem, we use m to denote the amount of virtual clusters arriving at time frame t . Let $T(G_v, t)$ be the function for calculating the total revenue of the data center network that processes m virtual clusters during time frame t , and it is defined as

$$T(G_v, t) = \sum_{i=1}^m R(g_v^i, t) \quad (4)$$

3) Objective Function

Our objective is maximizing the total revenue T under the resource constraints of physical machines and physical links during the online resource allocation. Here are two challenges, one is the execution order of multiple virtual clusters at time t , and the other one is to maximize the revenue of the virtual clusters during the resource allocation.

$$\text{maximize } T(G_v, t) \quad (5)$$

$$\text{subject to } C(G_s) \geq 0, C(G_v) \geq 0 \quad (6)$$

$$B(G_s) \geq 0, B(G_v) \geq 0 \quad (7)$$

$$C(G_s) \geq C(G_v) \quad (8)$$

$$B(G_s) \geq B(G_v) \quad (9)$$

Equation (6) and Equation (7) show the constraints on the revenue of the computation resources and the communication resources of virtual clusters and data centers. Equation (8) shows that the revenue of the allocated computation resource of virtual clusters is limited by the revenue of the total available resources in physical machines, and Equation (9) shows that the allocated communication resource requirements of virtual clusters are under limited by the revenue of the total available resources of physical links. Our work is finding the optimal solution for the objective function when the entire network reaches equilibrium. All notations are shown in Table 1.

Theorem 1. The virtual cluster provisioning problem for multi-tenant with revenue maximization is NP-hard.

Proof: Given a set of virtual clusters $G_v = \{g_v^1, g_v^2, \dots, g_v^m\}$, and each virtual cluster $g_v^i = \langle N_v^i, L_v^i, p_i, d_i \rangle$. We assume the deadlines of the virtual clusters are the same, and the remaining available resources of the data center network are $\sum_i C_i^*$ and $\sum_{i,j} B_{i,j}^*$. Since the total amount of virtual clusters is m , the goal is to place the virtual clusters in G_v with maximum revenue under the rest available resource constraints. So we reduce the original problem to the so-called bin-packing problem [29], an NP-hard problem that needs an assignment using the fewest bins. Thus, the virtual cluster provision for multi-tenant with revenue maximization is NP-hard.

Table 1. Notations

G_S	Substrate data center network.
N_S	Set of physical machines in G_S .
L_S	Set of physical links in G_S .
n_S^i	Substrate physical machine i in N_S .
$l_S^{i,j}$	Link between adjacent n_S^i and n_S^j .
G_v	Set of virtual clusters.
g_v^i	The i^{th} virtual cluster.
N_v^i	Set of virtual machines of g_v^i .
L_v^i	Set of virtual links of g_v^i .
$p_S^{i,j}$	Path between two physical machines, n_S^i and n_S^j .
$C(n)$	Revenue function of physical machine.
$B(l)$	Revenue function of physical link.
$R(g_v, t)$	Revenue of accepting one virtual cluster g_v .
$T(G_v^m, t)$	Total revenue of accepting m virtual clusters during time frame t .
γ	Accommodation coefficient.

4. Properties Analysis

In order to solve the online resource allocation problem, we transform the online resource allocation problem to a convex optimization problem. Then we perform an equivalent conversion with the variational inequality. Finally, we illustrate that the result of our resource allocation model is deterministic and uniquely suited for utility maximization.

Theorem 2. The total revenue of the data center network can be reduced to a variational inequality in a finite dimension. The results can be obtained by solving a vector $X^* = (C(G_v)^*, B(G_v)^*)$, which satisfies $\langle \nabla T_R(X^*), X - X^* \rangle \leq 0$, $X \in K$. T_R is a continuous function from K to R_n , where K is a closed convex set. $\nabla T_R(X^*)$ refers to the gradient for each component of X^* , and $\langle *, * \rangle$ refers to the inner product in R_n .

Proof: The following assumption is relevant to the entire proof: We assume that the total revenue of a data center network can be expressed as $T(G_v^m, t)$ after receiving and handling all virtual network requests at time t . The bandwidth revenue (the first term in Equation (2)) is not affected by the physical links to which the virtual links are mapped when the value of $\alpha = 1$. In order to deal with the objective function, we can obtain

$$T(G_v^m, t) = \sum_{i=1}^m \sum_{j=1}^n \frac{\partial T(G_v^m, t)}{\partial C_i(G_v)} (C_i(G_v) - C_i^*(G_v)) + \sum_{i=1}^m \sum_{j=1}^n \frac{\partial T(G_v^m, t)}{\partial B_i(G_v)} (B_i(G_v) - B_i^*(G_v)) \quad (10)$$

According to the restrictions introduced in formula (3), we can verify that for any $X^* = (C(G_v)^*, B(G_v)^*)$ the following conditions are satisfied:

$$C_i \in \equiv \{KC_i \mid C_i(G_v) \geq 0, \forall i\} \quad (11)$$

$$B_i \in \equiv \left\{ KB_i \mid B_i(G_v) \geq 0, B_i(G_v) = \sum_{j=1}^n B_{ij}(L_v), \forall i, \forall j \right\} \quad (12)$$

Under the limit conditions, our objective is a convex function, and K is a closed convex set. Based on the variational inequalities and by optimizing the conversion relation theorem, we can determine that obtaining the optimal solution under restrictions is equal to finding the $C(G_v)^*$ and $B(G_v)^*$ that fulfill the inequality.

Corollary1. Assuming that there exist $C(G_v)^*$ and $B(G_v)^*$ that satisfy $\max T_R(X)$, then $C(G_v)^*$, $B(G_v)^*$ are the unique solution of $\langle \nabla T_R(X^*), X - X^* \rangle \geq 0$, $X \in K$.

Proof: According to Theorem 1, we can convert the optimization model into variational inequalities like equation (4). K is a bounded, closed convex set, and $T(G_v^m, t)$ is a continuous and differentiable function. As the number of virtual clusters that we accept grows, the total revenue of the data center network also increases. Since it is a monotonous increasing function, we can get $\langle T_R(G_v^{i+1}) - T_R(G_v^i), G_v^{i+1} - G_v^i \rangle$. According to the property of variational inequalities, if $C(G_v)^*$, $B(G_v)^*$ meet $\langle \nabla T_R(X^*), X - X^* \rangle \geq 0$, then $X \in K$ is the unique solution.

5. Online Virtual-Cluster Provision Scheme

5.1 Online Multi-tenancy Scheduling

1) Description: In this subsection, we consider the online multi-tenancy scheduling problem to maximize the total revenue over $[0, T]$. We first split the time period $[0, T]$ into equal units called frames, and the virtual clusters arriving during the same time frame are processed as one batch. We assume that there will be m_i virtual clusters g_v^i arriving at time frame t_i , the total number of arriving virtual clusters during $[0, T]$ will be $\sum_{i=0}^T m_i$. The main idea is finding an execution order for multiple tenants with a maximum total revenue. Before we describe the scheduling algorithm, a factor e_i was introduced to measure the importance of the virtual cluster. We define that the lower value of e_i has a higher priority, and the calculation is defined as:

$$e_i = \frac{d_i}{T} \cdot \frac{1}{R_v(g_v^i)} \quad (13)$$

Algorithm 1: Online Multi-tenancy Scheduling (OMS)

Input: Set of virtual clusters G_v in time period $[0, T]$;

Output: Execution order of m_i virtual clusters;

```

1: for  $i = 1$  to  $\varepsilon$  in  $[0, T]$  do
2:   for  $i = 1$  to  $m_i$  in  $G_v$  do
3:     Calculate  $e_i$  for  $g_v^i$ ;
4:     Sort  $g_v^i$  in the set  $G_v$  to  $G'_v$  with  $e_i$ ;
5:     for  $i = 1$  to  $m_i$  in  $G'_v$  do
6:       if  $C_v^i \leq C_s^i$  and  $B_v^i \leq B_s^i$  then
7:         Place  $g_v^i$  into  $G_s$ ;
8:         VMP-VC( $g_v^i$ )  $\rightarrow G_s$ ;
9:       else
10:        Reject virtual cluster;
11: Return the data center network occupation state for set  $G_v$ ;

```

2) Algorithm: Before allocating the resource for the virtual clusters, we first decide the processing order using multi-tenant scheduling. The main idea is finding an execution order of virtual clusters with maximizing the total revenue for the data center network of each time frame t_i in time period $[0, T]$. Our insight is to allocate the resources to virtual clusters with higher priority e_i (early deadlines and high resource demands). The Online Multi-tenancy Scheduling (OMS) is proposed in Algorithm 1. We take the set of virtual clusters G_v in time period $[0, T]$ to be the input, and the execution order of m_i virtual clusters to be the output. For each virtual clusters g_v^i , we initialize its available computing resource and the communication resource as n_v^i and l_v^i , respectively. The communication resource l_v^i is derived by the computing resource n_v^i , i.e., $l_v^i = n_v^i \cdot B$. In line 1, we first search all virtual clusters that income at t_i in G_v . Then, we calculate the factor e_i of each virtual cluster in lines 2 to 3, and sort g_v^i in set G_v to G'_v with the e_i by an increasing order in line 4. In lines 5 to 8, we start to place the virtual cluster by the priority order in G'_v . For each virtual cluster, the total requirement of the computing resource N_v^i should not exceed the capacity of the physical machine C_s^i . The requirement of the communication resource should not beyond the capacity of the physical link B_s^i . If the incoming virtual clusters meet the constraints of the resources, they will be allocated into physical machines with the maximum remaining available computation resource. After that, we direct it to G_s using the Virtual Machine Placement algorithm for the Virtual Clusters (VMP-VC) in Algorithm 2. Otherwise, we directly reject the virtual cluster in lines 9 and 10. The complexity of Algorithm 1 is analyzed in subsection 6.3.

6. Virtual Machine Placement of Virtual Cluster

6.1 Data Center Network Preprocess

1) Physical Machine Clustering

We propose a physical machine clustering method, which is a pretreatment for physical machines in the data center network. It significantly increases the efficiency and utilization of virtual cluster placement. We use G_s to denote our data center network topology, where $G_s = \langle N_s, L_s \rangle$. Let N_s denote the set of physical machines in G_s , and the states of physical machines are not the same which depend on the usage conditions of computing capacities. The states of physical machines are one of the major properties utilized in the physical machine clustering. As we mention in the last section, the states of the physical machines are UD, UA, and NA. We give the priorities to physical machines according to the states, which is shown in Fig. 3 (b). We use P^i to denote the priority of the physical machine, and for any i^{th} physical machine, where $G_s^i < G_s^j$, the priorities of these two physical machines are $P^i > P^j$. As the virtual clusters need to be completed before their deadlines, the state of physical machines in the data center network is dynamic. The priority varies with the condition of physical resource usage of the physical machine, as shown in Fig. 3. If the state of the physical machine is in UA at time t , it will be ignored directly during the virtual cluster provisioning process. Once the virtual machines release the resource they occupied, the partitions of clusters would be updated, and the state of the physical machine will be transferred to UA or NA. To ensure the high efficiency and accuracy of the resource allocation for the multiple tenants, we classify the physical machines into clusters according to their states by using cluster analysis in the data center network.

2) Algorithm

In this subsection, we propose a simple clustering algorithm for substrate networks. This algorithm is based on the techniques that are used for analyzing big data and the characteristics of data center networks. As shown in Algorithm 2, we first analyze the substrate network to complete feature extraction. Then we choose an optimum clustering method, which applies to the substrate nodes. It is based on the theory proposed in [11], combined with the idea of identifying density peaks and structural characteristics in the underlying network. The clustering algorithm divides the data objects into multiple classes. Objects in the same cluster have a high similarity, while objects in different clusters have great differences. This dramatically increases the efficiency and utilization of node embedding.

As shown in Algorithm 2, the input variables of the physical machine clustering algorithm are the nodes of the substrate data center network, and the outputs are clusters with different characteristics. First, we select two nodes randomly and calculate the distance between them in line 1, as shown in Algorithm 2. Let $degree(n)$ denote the state of the physical machine. The meaning of distance $dist(n_i, n_j)$ in our algorithm is the absolute value of $degree(n)$ between nodes n_i and n_j , which is calculated in line 2. If the distance value is 0, we can define either of the nodes as a cluster center. If the distance value exceeds 0, however, both are set as cluster centers. Then we perform the expansion from the new center, and each cluster absorbs nodes with the same attribute. During the process of clustering, the nodes with

nonzero distance can be seen as noise nodes. For each cluster, once a noise node is found, it is defined as a new center. According to the node construction, each node alternates among the three states UD, UA, and NA. Once the resource is assigned, the corresponding state of the node changes from UD to UA (or NA). The clustering information of the substrate nodes is dynamic. Since our clustering algorithm considers each pair of physical machines in the data center network, the complexity of Algorithm 2 is related to the number of physical machines in G_s . Let $|N_s|$ be the number of physical machines in the data center network, then we have that the complexity of Algorithm 2 is $O(|N_s|^3)$. Algorithm 2 depicts this process.

Algorithm 2: Physical Machine Clustering algorithm (PMC)

Input: Information of the substrate network G_s ;

Output: physical machine clusters;

- 1: Calculate the distance between two physical machines i, j in the substrate network;
 - 2: $dist(n_i, n_j) = abs(deg\ ree(n_i) - deg\ ree(n_j))$;
 - 3: **while** ($dist(i, j) == 0 \cap node(j) \in G_s$) **do**
 - 4: Divide nodes into different clusters and mark them according to their states value;
 - 5: Removed n_j from G_s ;
 - 6: **if** $G_s \neq \Phi$ **then**
 - 7: Select a node and calculate the degree in G_s ;
 - 8: Remove it from G_s , and set this node to the cluster center;
 - 11: Return physical machines clusters;
-

6.2 Virtual Machine Placement for Virtual Cluster (VMP-VC) Algorithm

As shown in Algorithm 3, we take the virtual request G_v^i as our input. The initialization in line 1 use $PMC(G_s)$ to classify the physical machines into several clusters. As mentioned above, the states of the physical machines in each cluster is the same. In lines 2 to 13, we begin to place the virtual clusters iteratively. In line 2, we first compare the total demands of the virtual cluster G_v^i with the rest of the available physical resource G_s , i.e., $G_s = \sum_i C_s^i$. If the rest of available physical resources cannot satisfy the demand of the virtual cluster, the algorithm will immediately reject this virtual cluster and go from G_v^i to G_v^{i+1} , as shown in line 13. Otherwise, it will start to place G_v^i . Then, we start to find an appropriate resource allocation scheme for each virtual cluster that maximizing the revenue. In line 3, we try to find the physical machine with the maximum available computation resource by using $i = \arg \max_i C_s^i$. If the virtual cluster G_v^i can be placed into the physical machine i , then we start to place G_v^i , as shown in lines 4 and 5. Otherwise, we prefer to place G_v^i into several different physical machines in the same cluster, as shown in lines 6 to 10. In line 8, we find the physical machine cluster with

maximum priority by using $i = \arg \max_i P_i$. In line 9, all the virtual machines of the virtual cluster are evenly placed into the physical machines in $cluster_i$, and we update g_v^i in line 10. We iteratively place the virtual machines into clusters until $G_v^i < 0$. Once the locations of the incoming m virtual machines has been determined into the physical machines, we search for all possible paths between them to find a path that can satisfy communication demands of them. As we discussed in this paper, we use the topology of multi-rooted tree as our data center network architecture. There exist multiple paths depending on the number of physical switches in the data center network. In line 11, the communication requirements of the placed virtual machines are evenly divided into paths that connect them.

Algorithm 3: Virtual Machine Placement for Virtual Cluster (VMP-VC)

Input: Virtual cluster G_v^i ;

Output: Data center network resource occupation for G_v^i ;

- 1: Initialization the data center network by using $PMC(G_s) \rightarrow G_s$;
 - 2: **if** $g_v^i \leq G_s$ **then**;
 - 3: Find the physical machine with maximum available capacity by setting $i = \arg \max_i C_s^i$;
 - 4: **if** $g_v^i \leq C_s^i$ **then**
 - 5: Place all virtual machines of G_s into physical machine N_s^i ;
 - 6: **else** $g_v^i > C_s^i$ **then**
 - 7: **while** $g_v^i > 0$ **do**
 - 8: Find the physical machine cluster with the maximum priority by setting $i = \arg \max_i P_i$;
 - 9: All virtual machines are evenly placed into physical machines in $cluster_i$;
 - 10: Update g_v^i ;
 - 11: Communication demands of placed virtual machines are evenly split into paths connecting them;
 - 12: **else if** $g_v^i > G_s$ **then**
 - 13: Reject virtual cluster g_v^i ;
-

6.3 Complexity Analysis

In this subsection, we discuss the complexity of our OMS algorithm. For each time frame t_i , the OMS algorithm recalls the VMP-VC algorithm to realize the virtual cluster provision for set G_v . Thus, we have that the time complexity for each time frame t_i is $O(m \cdot |N_s|^3)$. Since the time period $[0, T]$ can be divided into ε time frames, where $\varepsilon \geq 1$. In sum, the complexity of our OMS algorithm is $O(\varepsilon \cdot m \cdot |N_s|^3)$.

7. Performance Evaluation

7.1 Online Multi-tenancy scheduling

1) **Basic Setting:** In this subsection, we consider the relationship between the online multi-tenancy scheduling and the total revenue by comparing three evaluation groups. This paper implemented a virtual cluster placement simulator to evaluate our algorithms by utilizing the GT-ITM tool based on the NS2 for the data center network, which has been utilized in popular research requiring practical network topology generation [12–14]. We use a two-layer multi-rooted tree as our data center topology, which is built on the equal number of switches and physical machines [1]. Each physical machine in our topology is partially connected with switches. The data center network was constructed by 100, 200, and 300 physical machines and about 1,000 physical links. For each physical machine, we generate the number which is uniformly distributed between 50 and 100 to represent the CPU and bandwidth resources. Tenants can determine all the information of their requests, include the arriving times and the number of nodes. Since the arriving times of the virtual clusters were discretionary, we use the Perlin noise function with different parameters to formulate this process. The lifetimes of the requests were generated by the tenants with an average of 300 s. The parameters and symbols that we varied in our simulations were the acceptance ratio and the average revenue. The acceptance ratio is the percentage of successful provisioned virtual clusters to the total number of incoming virtual clusters in set G_v . The average revenue of the data center network is calculated by dividing the total revenue by the number of virtual clusters in set G_v . As the previous works did not focus on online multi-tenancy provisioning problem, therefore, we compare our algorithms with three baseline ones: First-Come First-service Algorithm (FCF), Earliest Deadline First (EDF) Algorithm [16], and High Capacity First Algorithm (HCF).

2) **Experimental Results:** We first consider the average revenue of the data center network, and the results of different algorithms are presented in Fig. 4. We deploy the same scheduling algorithms on the data centers, where the numbers of physical machines are 100, 200, and 300. We do the resource allocation for virtual clusters using the algorithm VMP-VC and trace the data in 0 to 600s for each group. The experimental results show that our algorithm can achieve significantly higher acceptance ratio and average revenue under different data center networks. As shown in Fig. 4 (a), as the number of virtual requests for multiple tenants is uncertain, the average revenue of FCF has changed drastically as the network topology has increased. When the data center network scale gets large ($N = 300$), the average revenue under the EDF and HCF are nearly the same. However, when the scales of the data center networks are $N = 100$ and $N = 200$, the average revenues under the EDF and HCF are fluctuating. With the increasing of the area, the gap between OMS and other algorithms also grows. The lifetime of the virtual cluster has a certain influence on the acceptance rate of the data center network. According to analysis of experimental results, we can see the usage condition of the data center network under different situations. As shown in Figs. 5 (a) and (b), since the number of virtual clusters for tenants is uncertain, the acceptance ratio of RE changes drastically as the topology of the data center network increases. The lifetimes of virtual clusters have a certain impact on the acceptance rate of the data center network. The results show in Figs. 5 (a), (b), and (c) indicate that with the resources are releasing, the acceptance condition and long-term average revenue of the data center network increase gradually. Fig. 5 shows that the virtual clusters using the clustering resource allocation algorithm have higher long-term average

revenues than the baseline algorithms in the data center network. We can see that the average revenue remains stable of OMS for different data centers. Experiment results show that our scheme generates a higher efficiency than the baseline methods. The acceptance ratios of virtual requests can improve on average by 16.5%, 12.1%, and 14.2% for the data center networks by using FCF, EDF, and HCF, respectively. The average revenues of virtual requests can improve on average by 11.3%, 13.4%, and 15.6% for the data center networks by using FCF, EDF, and HCF, respectively.

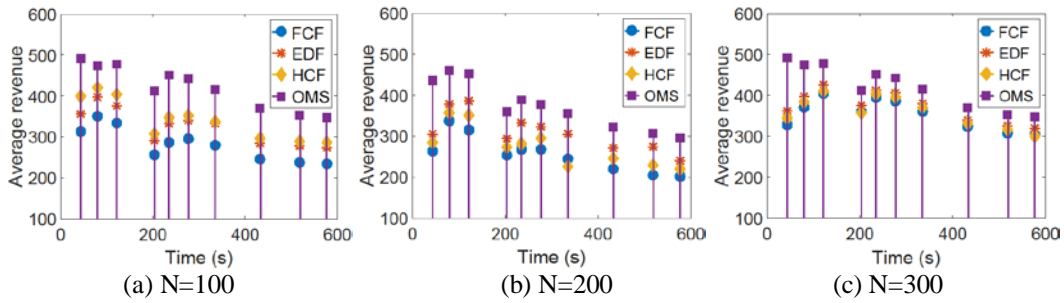


Fig. 4. Evaluation of the online multi-tenancy scheduling on various topologies of data center networks- Average revenue.

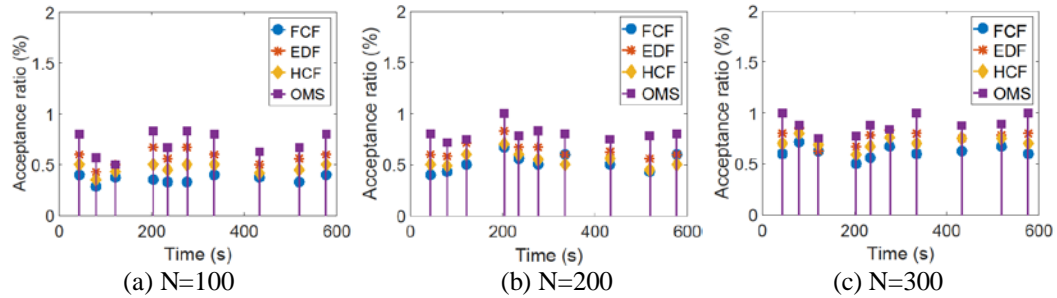


Fig. 5. Evaluation of the online multi-tenancy scheduling on various topologies of data center networks- Acceptance ratio of multi-tenant.

7.2 Virtual Machine Placement

In this section, we evaluate the performance of VMP-VC algorithm by focusing on the relationship between the total revenue and virtual machine placement for virtual clusters.

1) Basic Setting: We utilize the same dataset as in the online multi-tenancy scheduling problem. Since the former researches on the formulation of the multiple virtual cluster provisioning problem did not focus on the online case, we implemented the OMS algorithm according to the reference of the former experience and the requirements of our experiment. Three compared algorithms are utilized in our experiment, Random Embedding Algorithm (RE), Equally Distributed Embedding Algorithm (EDE), and Best Fit Embedding Algorithm (BFE).

2) Experimental Results: **Fig. 6** and **Fig. 7** present the acceptance ratio and average revenue of the data center network under the virtual request embedding algorithms. For each group, we have the VRE algorithm has a significantly higher acceptance ratio and average revenue for different data center networks. As shown in **Figs. 7 (a), (b), and (c)**, RE has the lowest acceptance ratio. Due to the lifetimes of the virtual requests are different, the available resources of physical machines will be dynamic in the same time frame. Since EDE and BFE

consider to balance the relationship between the allocation of virtual machines and the capacities of the physical machines during the virtual request embedding, we can see that EDE and BFE have better performances than RE. However, EDE disregards the communication demand of the virtual request, which involves the bandwidth resource allocation. The acceptance ratio fluctuates with the available physical resource, and the fluctuation decreases with the ascending scales of the data center networks. **Figs. 6 (a), (b), and (c)** show the average revenue for virtual requests of multi-tenancy. The value of average revenue varies with the acceptance ratio, in which a high acceptance ratio leads to large average long-term revenue. As the comparison of the experimental groups demonstrates, there is no obvious difference between EDE and BFE in terms of the acceptance ratios; Since the searching process in BFP is greedy at each iteration, its average long-term revenue is much higher than the other two algorithms. VRE achieves a better performance under the OMS on both acceptance ratio and average revenue. For VRE, both the acceptance ratio and average revenue remain stable under different data center networks. The acceptance ratios of the virtual requests improve on average by 15.7%, 13.3%, and 12.8% for data center networks with RS, EDF, and HCF. The average revenues of virtual requests improve on average by 15.4%, 13.2%, and 11.8% for data center networks with RS, EDF, and HCF.

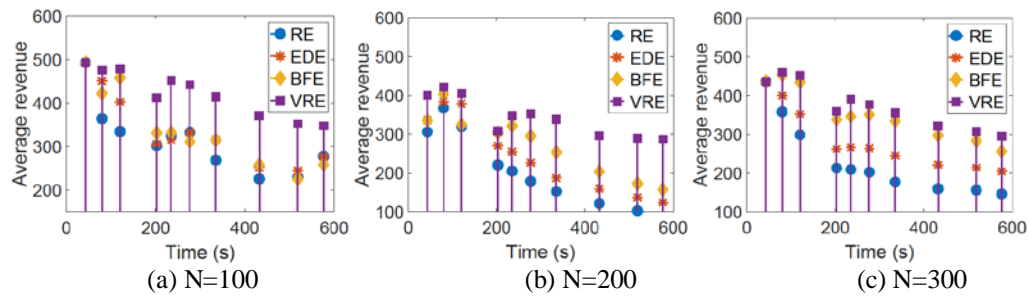


Fig. 6. Evaluation of virtual clusters provision on various topologies of data center networks-Average revenue.

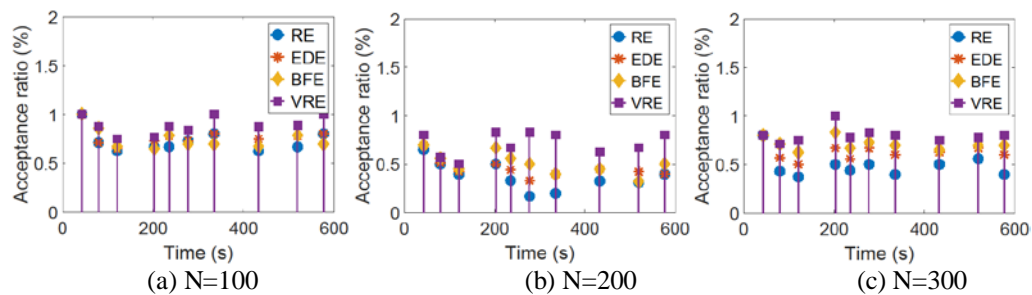


Fig. 7. Evaluation of virtual clusters provision on various topologies of data center networks-Acceptance ratio of multi-tenant.

8. Conclusion

In this paper, we study the online virtual cluster provision problem with multi-tenancy in the data center networks, including when and where virtual clusters should be placed in the data center. We use the virtual cluster as our communication model and the multi-rooted tree as our data center network model. In order to solve this problem, we divide it into two parts: online multi-tenancy scheduling and virtual cluster placement. Our objective is to find a provisioning

scheme that can maximize the revenue for the data center network under the constraints of computation and communication resource. We first formulate it by using the variational inequality model and discuss the existence of the optimal solution. After that, we prove that online virtual clusters provisioning for the revenue maximization problem is NP-hard. Due to the complexity of this problem, an efficient heuristic algorithm OMS is proposed. Depending on the OMS scheme, we propose a novel algorithm VMP-VC. A large number of simulations demonstrate that our algorithm outperforms existing approaches in maximizing the revenue of data center networks.

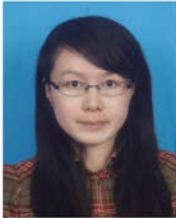
Acknowledgement

This work of the first author was done during her stay as a visitor scholar at Temple University. This research was supported in part by NSF grants CNS 1757533, CNS1629746, CNS 1564128, CNS 1449860, CNS 1461932, CNS 1460971, IIP 1439672, and CSC 20163100.

References

- [1] Bari, Md Faizul, et al. "Data center network virtualization: A survey," *IEEE Communications Surveys Tutorials*, 15.2, 909-928, 2013. [Article \(CrossRef Link\)](#)
- [2] Ballani, Hitesh, et al. "Towards predictable datacenter networks," *ACM SIG-view*, Vol. 41. No. 4. ACM, 2011. [Article \(CrossRef Link\)](#)
- [3] Al-Fares, Mohammad, Alexander Loukissas, and Amin Vahdat, "A scalable, commodity data center network architecture," *ACM SIGCOMM Computer Communication Review*, Vol. 38. No.4. ACM, 2008. [Article \(CrossRef Link\)](#)
- [4] Liu, Yang, et al. "Data center networks: Topologies, architectures and fault-tolerance characteristics," *Springer Science and Business Media*, 2013. [Article \(CrossRef Link\)](#)
- [5] Albers S., "Online algorithms: a survey," *Mathematical Programming*, 97(1-2):3-26, Jul 2003. [Article \(CrossRef Link\)](#)
- [6] Guo, Chuanxiong, et al. "BCube: a high performance, server-centric network architecture for modular data centers," *ACM SIGCOMM Computer Communication Review*, 39.4, 63-74, 2009. [Article \(CrossRef Link\)](#)
- [7] Alicherry M, Lakshman, "TV Network aware resource allocation in distributed clouds. In *Infocom*," *2012 proceedings IEEE*, pp. 963-971, Mar 25 2012. [Article \(CrossRef Link\)](#)
- [8] Cheng, Xiang, et al. "Virtual network embedding through topology-aware node ranking," *ACM SIGCOMM Computer Communication Review*, 41.2, 38-47, 2011. [Article \(CrossRef Link\)](#)
- [9] Dias DS, Costa LH., "Online traffic-aware virtual machine placement in data center networks," in *Proc. of Global Information Infrastructure and Networking Symposium (GIIS)*, pp. 1-8, Dec 17, 2012. [Article \(CrossRef Link\)](#)
- [10] Meng, Xiaoqiao, Vasileios Pappas, and Li Zhang. "Improving the scalability of data center networks with traffic-aware virtual machine placement." *INFOCOM, 2010 Proceedings IEEE. IEEE*, 2010. [Article \(CrossRef Link\)](#)
- [11] Edwards, Aled, Anna Fischer, and Antonio Lain, "Diverter: A new approach to networking within virtualized infrastructures," in *Proc. of the 1st ACM workshop on Research on enterprise networking*, ACM, 2009. [Article \(CrossRef Link\)](#)
- [12] Hao, Fang, et al. "Enhancing dynamic cloud-based services using network virtualization," in *Proc. of the 1st ACM workshop on Virtualized infrastructure systems and architectures*, ACM, 2009. [Article \(CrossRef Link\)](#)
- [13] Tsai, Linjiun, and Wanjiun Liao, *Virtualized Cloud Data Center Networks: Issues in Resource Management*, Springer, 2016. [Article \(CrossRef Link\)](#)

- [14] Guo, Chuanxiong, et al. "Secondnet: a data center network virtualization architecture with bandwidth guarantees," in *Proc. of the 6th International Conference*, ACM, 2010. [Article \(CrossRef Link\)](#)
- [15] Lischka, Jens, and Holger Karl., "A virtual network mapping algorithm based on sub-graph isomorphism detection," in *Proc. of the 1st ACM workshop on Virtualized infrastructure systems and architectures*, ACM, 2009. [Article \(CrossRef Link\)](#)
- [16] Li, Xiaoling, et al. "Resource allocation with multi-factor node ranking in data center networks," *Future Generation Computer Systems*, 32: 1-12, 2014. [Article \(CrossRef Link\)](#)
- [17] Papagianni, Chrysa, et al. "On the optimal allocation of virtual resources in cloud computing networks," *IEEE Transactions on Computers*, 62.6:1060-1071, 2013. [Article \(CrossRef Link\)](#)
- [18] Rodriguez, Alex, and Alessandro Laio, "Clustering by fast search and find of density peaks," *Science*, 344.6191 :1492-1496, 2014. [Article \(CrossRef Link\)](#)
- [19] Vdovin, P. M., et al. "Comparing various approaches to resource allocation in data centers," *Journal of Computer and Systems Sciences International*, 53.5: 689-701, 2014. [Article \(CrossRef Link\)](#)
- [20] Sun, Gang, et al. "Power-efficient provisioning for online virtual network requests in cloud-based data centers," *IEEE Systems Journal*, 9.2: 427-441, 2015. [Article \(CrossRef Link\)](#)
- [21] Fischer, Andreas, et al. "Virtual network embedding: A survey," *IEEE Communications Surveys and Tutorials*, 15.4:1888-1906, 2013. [Article \(CrossRef Link\)](#)
- [22] Yu, Minlan, et al. "Rethinking virtual network embedding: substrate support for path splitting and migration," *ACM SIGCOMM Computer Communication Review*, 38.2: 17-29, 2008. [Article \(CrossRef Link\)](#)
- [23] Eppstein, David, "Finding the k shortest paths," *SIAM Journal on computing*, 28.2: 652-673, Atlanta, Georgia ,1998. [Article \(CrossRef Link\)](#).
- [24] Zegura, Ellen W., Kenneth L. Calvert, and Samrat Bhattacharjee, "How to model an internetwork," in *Proc. of INFOCOM'96. Fifteenth Annual Joint Conference of the IEEE Computer Societies. Networking the Next Generation*, Vol.2, IEEE, 1996. [Article \(CrossRef Link\)](#)
- [25] Jiang, Joe Wenjie, et al. "Joint VM placement and routing for data center traffic engineering," *INFOCOM, 2012 Proceedings, IEEE*, 2012. [Article \(CrossRef Link\)](#)
- [26] Zhang, Zhongbao, Xiang Cheng, et al. "A unified enhanced particle swarm optimization-based virtual network embedding algorithm," *International Journal of Communication Systems*, 26, no. 8, 1054-1073, 2013. [Article \(CrossRef Link\)](#)
- [27] Rodriguez, Alex, and Alessandro Laio,"Clustering by fast search and find of density peaks," *Science*, 344.6191: 1492-1496, 2014. [Article \(CrossRef Link\)](#)
- [28] Issariyakul, Teerawat, and Ekram Hossain, *Introduction to network simulator NS2*. Springer Science and Business Media, 2011. [Article \(CrossRef Link\)](#)
- [29] Thomas, Megan, Elizabeth Edwards, and Samrat Bhattacharjee. "Modeling topology of large internetworks," *College of Computing, Georgia Institute of Technology*, May 1997. [Article \(CrossRef Link\)](#)
- [30] Zegura, Ellen W., Kenneth L. Calvert, and Samrat Bhattacharjee. "How to model an internetwork," in *Proc. of INFOCOM'96. Fifteenth Annual Joint Conference of the IEEE Computer Societies. Networking the Next Generation*, Vol.2, IEEE, 1996. [Article \(CrossRef Link\)](#)
- [31] Baccarelli, Enzo, et al. "Q*: Energy and delay-efficient dynamic queue management in TCP/IP virtualized data centers," *Computer Communications*, 102: 89-106, 2017. [Article \(CrossRef Link\)](#)



Shuaibing Lu received her M.S in computer science and technology from Jilin University, Changchun, China, and she is currently a Ph.D. student. She was a researcher at the distributed computing system and data center network. She is supported by the China Scholarship Council as a visiting scholar supervised by Prof. Jie Wu in the department of computer and information sciences at Temple University (2016-2018). She is a student member of the IEEE.



Zhiyi Fang received his PhD in computer science from Jilin University, Changchun, China, in 1998. He currently works there as a professor of computer science. He was a senior visiting scholar at the University of Queensland, Australia, from 1995 to 1996 and at the University of California, Santa Barbara, from 2000 to 2001. He is a member of the China Software Industry Association (CSIA) and a member of the Open System Committee of China Computer Federation (CCF). His research interests include distributed/parallel computing systems, mobile communication, and wireless networks.



Jie Wu is the Director of the Center for Networked Computing and Laura H. Carnell professor at Temple University. He also serves as the Director of International Affairs at College of Science and Technology. He served as Chair of Department of Computer and Information Sciences from the summer of 2009 to the summer of 2016 and Associate Vice Provost for International Affairs from the fall of 2015 to the summer of 2017. Prior to joining Temple University, he was a program director at the National Science Foundation and was a distinguished professor at Florida Atlantic University. His current research interests include mobile computing and wireless networks, routing protocols, cloud and green computing, network trust and security, and social network applications. Dr. Wu regularly publishes in scholarly journals, conference proceedings, and books. He serves on several editorial boards, including IEEE Transactions on Service Computing and the Journal of Parallel and Distributed Computing. Dr. Wu was general co-chair for IEEE MASS 2006, IEEE IPDPS 2008, IEEE ICDCS 2013, ACM MobiHoc 2014, ICPP 2016, and IEEE CNS 2016, as well as program co-chair for IEEE INFOCOM 2011 and CCF CNCC 2013. He was an IEEE Computer Society Distinguished Visitor, ACM Distinguished Speaker, and chair for the IEEE Technical Committee on Distributed Processing (TCDP). Dr. Wu is a CCF Distinguished Speaker and a Fellow of the IEEE. He is the recipient of the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award.