IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Online User Distribution-aware Virtual Machine Re-Deployment and Live Migration in SDN-based Data Centers

JIN QIN[1,2], YIZHEN WU[3], YUTONG CHEN[1], KAIPING XUE[1,2], (Senior Member, IEEE), AND DAVID S.L. WEI[4], (Senior Member, IEEE)

[1]Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027 China (e-mail: hayato@mail.ustc.edu.cn, cyut@mail.ustc.edu.cn, kpxue@ustc.edu.cn)
[2]School of Cybersecurity, University of Science and Technology of China, Hefei 230027 China
[3]Huawei Shanghai Research Institute, Shanghai 201206, China (e-mail: yizhenwu@mail.ustc.edu.cn)
[4]Department of Computer and Information Science, Fordham University, NY 10458 USA (e-mail: wei@cis.fordham.edu)

Corresponding author: Kaiping Xue (kpxue@ustc.edu.cn)

This work is supported in part by the National Key Research and Development Plan of China under Grant No. 2017YFB0801702, the National Natural Science Foundation of China under Grant No. 61671420, and Youth Innovation Promotion Association CAS under Grant No. 2016394.

ABSTRACT Virtual machine (VM) re-deployment and migration has been proven to be a key technique for cloud data centers to implement resource optimization and load balance. Also, live VM migration aims to guarantee the continuity of the existing data flows. Though VM management has been well studied, it has so far mostly focused on optimising resource utilization, while the important issue of users' Quality of Experience (QoE) has been ignored. In mobile cloud computing (MCC), user distribution changes dynamically over time and it can significantly affect both the service latency and network resource utilization. In the traditional network, VM migration may cause a flash crowd of flow changing and a long service downtime due to VM's IP address changing. Software-defined networking (SDN) is an emerging paradigm to logically centralize the network control plane and automate the configuration of individual network elements, which can be ubiquitously deployed in data centers and serve as an effective means for flow handling. In this paper, we design a user Distribution-Aware virtual machine Re-Deployment (DARD) mechanism and propose a Traffic-Redirection virtual machine Migration (TRM) scheme to keep the active flows from being interrupted. We further provide simulations to show the effectiveness and superiority of the proposed approaches over existing schemes.

INDEX TERMS Cloud data centers, user distribution, live migration, re-deployment, software-defined networking

I. INTRODUCTION

Cloud computing (CC) is a new computing paradigm that enables cloud users to share pools of configurable resources and services in a cost-effective way. Through virtualization technology, virtual machines (VMs) are created according to users' demands, and users execute their applications on the VMs that are indeed running on physical servers [1]–[6]. In mobile cloud computing (MCC) [7], cloud data centers are dispersed across different geographic regions per service needs. To obtain their good quality service, users need to access some VMs in an effective way. To provide a better quality of experience (QoE) for users is thus the major mission for a cloud service provider (CSP). From a statistical viewpoint, the distances from online users to their corresponding servicing VMs usually directly influence the service delay and in turn influence QoE for the most online users. For this reason, VM(s) should be deployed at the place closer to most online users. However, as the user distribution varies over time, the initial VM deployment may not be suitable for its subsequent served network conditions. Therefore, VM redeployment in MCC with dispersed data centers is becoming a new means for efficient resource utilization to improve users' QoE [8]–[10]. As such, virtual machine management in cloud data centers should include two aspects: initial VM(s) deployment and VM(s) redeployment.

Firstly, it's necessary for a CSP to optimize initial VM(s) deployment across different geographic regions to improve service performance so that suitable VM(s) are deployed at

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2891115, IEEE Access

the right places to provide a specific service in consideration of resource constraints (e.g. CPU, Storage, and network bandwidth) of data centers [11], [12]. This raises the issues of different resource consumption [13]–[15] during the initial VM deployment.

Secondly, when initial VM deployment is no longer suitable due to the changes of user distribution, VM migration across Data Centers will lead to more efficient resource utilization and further improve users' QoE. For a CSP, VM redeployment generally requires to migrate a running VM to a new data center, and this needs to firstly copy a VM from its previous physical server to a new one, and then update the network to re-establish data paths from each user to the new location. As far as we know, the research of VM migration can be divided into three main categories: 1) The first one focuses on how to optimize copying CPU and Memory status to reduce service downtime [16]; 2) The second one is from the view of network resource optimization, e.g, selecting data copy path to optimize the overhead of the network updating [17]; 3) The third one is how to synchronize data copying phrase and network configuration [18] in order to reduce the migration time.

There are two kinds of factors causing service delay: 1) The first one is initial deployment of VMs without considering future online user distribution. In fact, this factor could not be controlled very well because it is difficult to accurately predict future user distribution. 2) The second factor is that user distribution always changes over time, because of many reasons, such as "tidal effect" [19]. This factor will cause the irrationality of initial deployment even if it was originally appropriate. For convenience, we collectively call this problem as "initial deployment unreasonable". Therefore, taking online user distribution into consideration, migrating VM(s) from their initial locations to new optimized locations can effectively reduce the average service delay and improve users' QoE.

Besides, in order to keep the continuity of active services, it is also important to implement a seamless online migration of VM [20], [21]. For this, we need to realize network layer mobility management to maintain the previously established flows without being interrupted because of the changes in IP address when a virtual machine is being migrated. Network layer mobility management has been proposed to keep the IP address unchanged when a user move from one network domain to another. By now, there have been extensive researches regarding the live VM migration based on the existing network layer mobility management schemes. There have been a large number of researches on live VM migration, and these works primarily focus on the migration cost in terms of the downtime, bandwidth, and power consumption [22]. For example, Deshpande et al. [23] proposed traffic-sensitive live migration of VMs. Nathan et al. [24] established a performance and energy model for live migration of VM. However, although how to effectively implement live VM migration for improving online users' QoE and reducing migration cost was studied over these years, the factor of online user distribution changing has attracted very little direct attentions.

Software-defined networking (SDN) is an emerging paradigm to logically centralize the network control plane and automate the configuration of individual network elements. Owing to SDN's centrality and network-wide abstraction of the control plane, it is much easier to implent fast service deployment and network virtulization, which is well-suited for dynamic environments such as cloud data centers. Based on the centralized control of SDN controller, it will be easier to maintain the states of the whole network, including bandwidth, storge, computing resource and even user distribution. Different from traditional distributed decision making, SDN controllers can make global optimal decisions for a CSP based on the collected network information. Meanwhile, SDN allows a single control protocol to implement a range of functions to provide flexible and unified control on routing. For live VM migration, the traffic path can be determined by the SDN controllers, ensuring the consistency and efficiency during the migration process while providing a guaranteed mobility support. However, the triangle routing problem still exists in such schemes, and it's also existing in mobility management schemes such as Mobile IP (IP) and Proxy Mobile IP (PMIP).

In this paper, we take both resource constraints and online user distribution into consideration, combine with the advantages of SDN to propose an online user distribution aware redeployment algorithm (DARD) to re-choose appropriate physical servers to carry the running VM(s). We further design a Traffic-Redirection virtual machine Migration (TRM) scheme to keep the active flows from being interrupted and solve the triangle touting problem at the same time. The performance analysis proves that our scheme can both improve the utilization of network resource and reduce the average service delay. The main contributions can be summarized as follows:

- We introduce online user distribution into optimizing VM re-deployment and formulate VM redeployment in SDN based data centers as an optimization problem. The proposed scheme can be implemented to choose a new physical server which is closest to the most online users to carry the running virtual machine, and the intelligent redeployment across geographically dispersed data centers can reduce both the service delay and the global bandwidth consumption.
- As only the SDN controller can generate flow table entries for each openflow switch, reducing the amount of entries distributed in the new path should be also taken into consideration in order to reduce the influence on core networks. We propose a Traffic-Redirection virtual migration scheme to minimizing the amount of flow table entries, and keep the active flows from being interrupted while avoiding the triangle routing problem.

The rest of this paper is organized as follows: The related work is briefly described in Section II. Section III gives the system model and the problem formulation of VM redeployment and migration. Our proposed DARD and TRM are presented in detail in Section IV, followed by the performance evaluation in Section V. Finally, we conclude the paper and give the future work in Section VI.

II. RELATED WORK

Live virtual machine migration has attracted increased attentions from both academia and industry since it was proposed in 2005 [20]. Existing works include Zhang et al. [9] proposed Network-aware virtual machine migration in an over-committed cloud. DeCusatis et al. [21] proposed an Openflow based network infrastructure to implement inter-domain VM(s) migration. Meanwhile, Keller et al. [22] proposed LIME (LIve Migration of Ensembles), which is an efficient solution to implement the live migration of the whole VM network consisting of multiple VMs. These existing research works indicate that SDN technology offers great advantages on solving VM migration in data centers. In general, there are three basic problems during the process of VM migration:

- ***Considering the optimization goal of VM migration:*** This kind of works include Traffic-sensitive live migration of virtual machines proposed by Deshpande et al. [23], and the work of Nathan et al. [24] that established a performance and energy model for live migration of VM(s). However, schemes mentioned above mainly focused on the migration cost in terms of the downtime, bandwidth, and power consumption, and just considered resource constraints. The influence of online user distribution changing has never been taken into account to enhance users' experience.
- ***Reducing the running VM's downtime:*** It is important to reduce the downtime of the running VM during migration. There have been a lot of approaches proposed to solve this problem, in which Pre-Copy is considered to be the most representative strategy [16], [18]. Although, with the Pre-Copy strategy in pages of memory, the contents are iteratively copied from the source physical server to the destination host without shutting down the execution of running VM, it is still not good enough to accomplish live VM migration.
- ***Keeping the running VM's IP address unchangeable:*** In traditional TCP/IP networks, a running VM with a specific IP address can be accessed by any online user through Internet. In general, the IP address of the VM has to be changed along with VM migrating across different geographic locations, so online users can not directly access the VM and keep the existing flows continuous without any changes. Some schemes have been proposed to deal with this continuity problem. For example, with extending proxy mobile IP, Silvera et al. [25] set the first hop switches respectively accessed by the physical servers that carry the running VM before and after migration to serve as the home-agent and the foreign agent. Then, an IP tunnel between these two agents can be established to maintain IP continuity. Although proxy Mobile IP has been proved to be a good solution to provide IP continuity in the network layer, it still has the problem of "triangle routing" which cannot be easily addressed. This problem is also introduced to the VM migration scenario.

As an enterprise usually deploys Data Centers in different geographic locations, there is a need to interconnect the dispersed Data Centers, and allow the seamless live VM migration among different physical servers in different locations. However, because of the bottleneck of the existing mobile IP based schemes, researchers turn to find new solutions using other network layer technologies. Raad et al. [26] attempted to solve live VM migration based on LISP (Locator/Identifier Separation Protocol), which is not compatible with the traditional TCP/IP architecture. Xie et al. [27] provided seamless live VM Migration via NDN (Named Data Networking) in Cloud Data Center.

As an emerging network paradigm, SDN has attracted much attention from both academia and industry over these years. There are many researches focusing on network virtualization and the design of virtual layer architecture, which lay the foundation for SDN's application in cloud computing and dynamic environments. Blenk et al. [28] proposed an SDN hypervisor architecture HyperFlex which relies on the decomposition of the hypervisor into functions that are essential for virtualizing SDN networks. And they further initiated the study of the network hypervisor placement problem in [29]. Sieber et al. [30] presented an extensible and distributed SDN hypervisor benchmarking framework based on flexible statistical request generators. Basta et al. [31] proposed a control path migration protocol for distributed hypervisors to provide a mobility support. Meanwhile, using the advantages of SDN to realize online VM management and live migration has also attracted more and more attentions. Satpathy et al. [32] proposed a two-layer VM placement algorithm using crow search and queuing structure. Rodrigues et al. [33] investigated an algorithm that utilizes VM migration and transmission power control, together with a mathematical model of delay in mobile edge computing and a heuristic algorithm called Particle Swarm Optimization, to balance the workload between cloudlets and consequently maximize cost-effectiveness. Liu et al. [34] and Mayoral et al. [35] both used global orchestrator approaches to implement the VM migration across different datacenters. Mandal et al. [36] proposed a multistage heterogeneous bandwidth provisioning scheme, which allocates optical network bandwidths at multiple stages for different phases of VM migrations. Sharma et al. [37] dealt with mulit-objective (network aware, energy efficient, and Service Level Agreement (SLA) aware) VMs migration at the cloud data center. Liu et al. [34] also proposed an NAT based solution to redirect the existing traffic to maintain online service continuity. Although to redirect traffic at the first hop switch accessed by the physical server before migration will lead to a minimum impact on the core network, it also introduces the problem of long triangle routing as that in mobile IP based schemes. As we know, a long

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2891115, IEEE Access

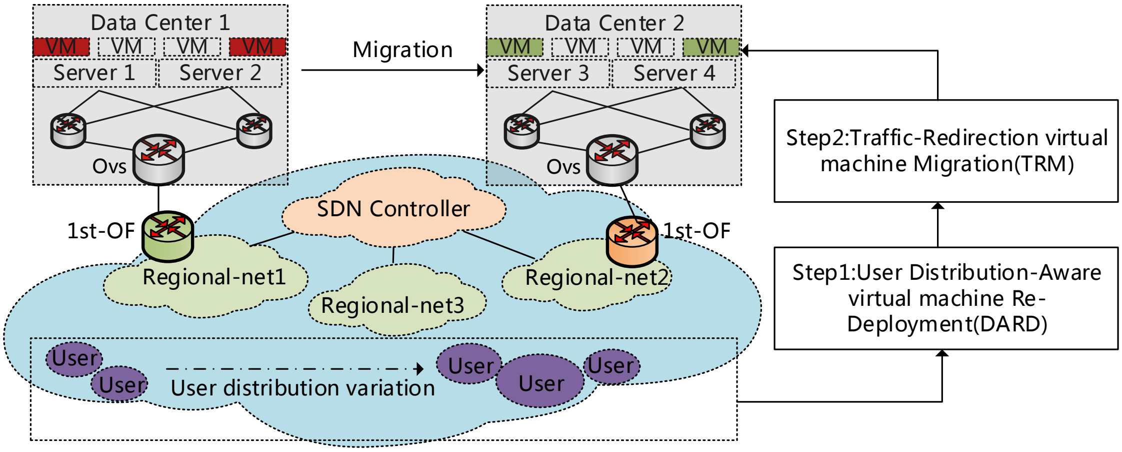


FIGURE 1. VM management: VM re-deployment and live migration

triangle routing will lead to a long delay and significantly reduce online users' experience. From this aspect, we can treat the traffic redirection as an optimization problem, which aims to find a tradeoff between providing optimal routing and minimizing the impact on the core network.

Although VM migration has been well studied over these years, the factor of online user distribution changing has attracted very little direct attentions. Different from those works only focus on the host based optimization, we decide to design a network based optimization scheme. Motivated by the advantages and challenges of virtual machine migration in SDN, we propose a user distribution aware redeployment (DARD) framework in consideration of user distribution, and design a traffic-redirection virtual machine migration scheme (TRM) to keep the active flows from being interrupted. With these two approaches, a cloud service provider can provide a higher QoE for the most online users, and keep the active flows uninterrupted while avoiding the triangle routing problem.

III. SYSTEM MODEL, ASSUMPTION, AND PROBLEM FORMULATION

A. ASSUMPTION

In this section, we formulate the virtual machine redeployment in SDN-based Data Centers as an optimization problem. Owing to its abstraction of the network control plane, SDN controller can timely collect network status, including bandwidth, computing resource and even the user distribution statistic. Then the controller can make a VM migration decision or select the specific migration path based on the changes in user distribution. Without loss of generality, and in order to simplify the subsequent derivation and experimentation, we assume that the whole set of data centers is supervised by a single SDN controller. Meanwhile, all the schemes we proposed can be further extended to multi-controllers scenario from the single controller scenario. In the scenario with multi-controllers, a VM may migrate from one controller's management domain to another controller's management domain. Compared with a single controller scenario, the major additional problem is the interaction and coordination between different controllers. Each data center usually has its own network controller, and different controllers need to interact with each other to update the routing path and ensure the consistency of VMs' information and their locations for data forwarding. There are already many related works aim to solve this problem, in which using a global orchestrator for controllers is considered to be an excellent choice. Liu et al. [34] used a global orchestrator for coordination among network controllers and cloud manage systems. The global orchestrator maintains all VMs' location information, help to select the best paths for transferring the migration traffic, and control the network update process. Mayoral et al. [35] also presented a network orchestration approach where several SDN controllers are directly orchestrated to implement seamless migration of VMs. And such strategies can also be applied to our schemes to change the control plane of SDN network into a two-layer structure, where all the controllers are orchestrated by a global orchestrator. Orchestrator can be used for information interaction and synchronization between different controllers, and the global optimal decision is made on the basis of the optimal decision of each controller. We will further investigate this problem in our future work, but in this paper all the problems were modeled and analyzed in a single controller scenario.

B. SYSTEM MODEL

Fig. 1 gives an example of initial deployment unreasonable problem, in which VMs have been initially deployed on the left data center (Data Center 1), which can be accessed from outside through a Openflow switch (we call this switch as a source 1st-OF). However, later when an online user move to a location far away from the initially deployed VMs, the user has to access the service through a long routing path. Therefore, the virtual machine management is also

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2891115, IEEE Access

responsible for VM redeployment. Firstly, we take user distribution into consideration, and formulate the virtual machine redeployment as an optimization problem, which contains not only resource constraints but also user-distribution distance constraint. Then, the next important work is how to seamlessly live migrate a VM from its old location to the new location, which should gurantee the online service's continuity and keep the IP address of VM unchanged during migration through traffic redirection. Different from Liu et al.'s work [34], we find an optimal position for redirecting traffic, which is a suitable tradeoff between providing optimal routing and minimizing the impact on the core network.

Fig. 1 briefly gives the main work that we complete in this paper. There are two steps to implement VM management. The first step is to implement user distribution aware virtual machine redeployment (DARD) to find an optimal location to carry the running VM by taking both online user distribution and resource constraints into consideration. The second step is to implement traffic-redirection virtual machine migration (TRM) to lively migrate VM to the new location found in Step 1. Later in this paper, we firstly formulate the problem of Step 1, and then formulate the problem of Step 2. The details of the two schemes are given in the next section (Section IV).

TABLE 1. Notions and Definitions in DARD

SYMBOL	DESCRIPTION
N	Total number of physical servers in a Data Center.
C_i	Resource constraint of memory on server s_i.
P_i	Resource constraint of CPU on server s_i.
U_i	Resource constraint of network bandwidth on server s_i.
c	The memory resource required by the VM v.
p	The CPU resource required by the VM v.
u	The network bandwidth required by the VM v.
$I(i)$	$I(i) \in \{0, 1\}$ where $I(i) = 1$ represent the VM should be deployed on the physical server s_i, and otherwise $I(i) = 0$. Furthermore, a VM can be deployed on only one physical server, hence, $\sum_{i=1}^{N} I(i) = 1$
$dist_{ij}$	Represent the length of the path between node i and j.

C. PROBLEM FORMULATION

1) Problem Formulation of DARD

DARD aims to solve initial deployment unreasonable through VM redeploying. As we have mentioned above, user distribution is a statistic and can be taken into consideration together with resource constraints. Therefore, how to reasonably quantify user distribution will be the first challenge in our scheme design. Besides, when user distribution is considered as a constraint in optimizing VM redeployment, we take service delay as the optimization object. TABLE 1 lists some notions and definitions used in DARD.

2) Problem Formulation of TRM

TRM aims to accomplish seamless live VM migration. After implementing DARD, one candidate physical server will be selected for redeploying the VM. To keep the same IP address when VM migrates across different subnets, we design an effective live migration scheme to redirect flows from source physical server to the destination one based on SDN architecture. To differentiate the old physical server that the VM resided before migration and the new physical server the VM will reside after migration, we assume that the VM migrates from s_n to s'_n . Besides, we give some additional notions and definitions used in TRM in Table II .

TABLE 2. Notions And Definitions In TRM

SYMBOL	DESCRIPTION
p_{old}	The path between user's location and the old attached physical server s_n. p_{old} is defined as a set of switches in the path, e.g, $p^1_{old} = \{s_1, s_2, \ldots, s_i, \ldots, s_n\}$
P_{old}	When there are n users access the service provided by the VM v in the old attached physical server s_n, the set of their paths are $P_{old} = \{p^1_{old}, p^2_{old}, \cdots, p^i_{old}, \cdots, p^n_{old}\}$
p_{new}	The path between user's location and the new attached physical server s'_n. p_{new} is defined as a set of switches in the path, e.g, $p^1_{new} = \{s_1, s_2, \ldots, s_i, \ldots, s'_n\}$
P_{new}	When there are n users access the service provided by the VM v in the new attached physical server s'_n, the set of their paths are $P_{new} = \{p^1_{new}, p^2_{new}, \cdots, p^i_{new}, \cdots, p^n_{new}\}$
t_{path}	The two tuple of path pair for a user is (p_{old}, p_{new}). All n path pairs make up a set$\{t^1_{path}, t^2_{path}, \cdots, t^n_{path}\}$

IV. OUR PROPOSED SCHEME

As mentioned above, the solution to address the problem of initial VM deployment unreasonable can be divided into two steps, respectively accomplished by DARD and TRM. These two steps constitute a complete scheme of VM redeployment and live migration. In this section, DARD and TRM are explained in detail.

A. DARD

By taking user distribution into account, DARD contains two sub-procedures: Firstly, it is necessary to solve the problem of how to quantize user distribution. Subsequently, by adding the factor of user distribution, we optimize the VM redeployment under both resource constraints and user distribution constraint. By solving this optimization problem, a physical server will be selected so that the most online users can access the VM with lower service delay. And as an extra note, DARD is not so sensitive to the real-time awareness. DARD only needs to obtain user distribution information, which is a statistic and the change is not fine-grained in time. Therefore, The movement law of individual user will not conflict with the current decision.

1) Specification and Analysis

As shown in Fig. 2, in the network topology of data centers network, nodes can be arranged in two categories: **Physical Servers** and **User Nodes**. Among these nodes, there is a Physical Server node selected for initial VM deployment. Through multiple core network switches along the path from each user node to the physical server, the corresponding users

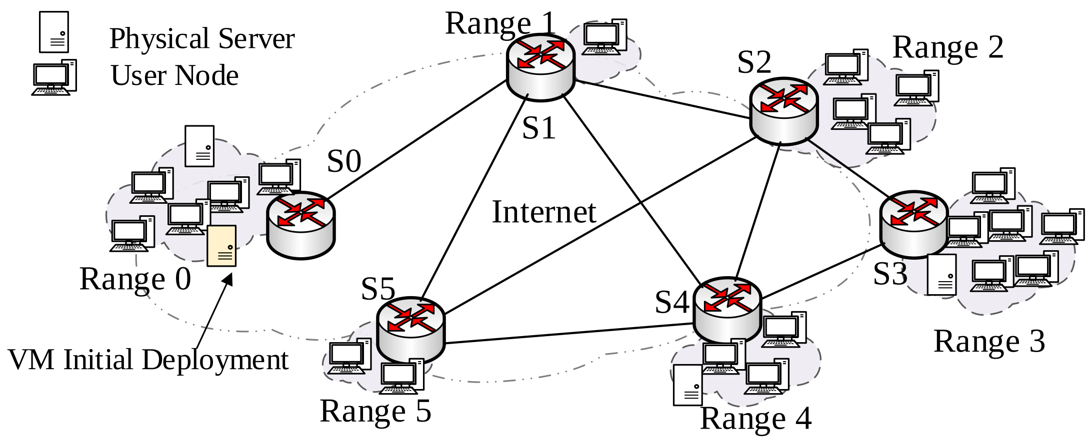


FIGURE 2. VM management: Range of user requirements and distribution

can access the service provided by the VM. Besides, **User Nodes** and **Physical Servers** in the topology can be divided into multiple ranges according to the switches to which they are accessing.

We use Fig. 3 as an example to introduce a statistical view to quantize user distribution. As shown in Fig. 3, we set the weight of each node, where each node represents a range and the weight represents the number of online users accessing the service in this range. Take the node $S3$ for example, $W_3 = 6$ represents there are 6 online users accessing the service provided by the VM in $Range0$. The weight of the edge represents the delay between the two core switches in two ranges. The blank node indicates that there is no physical server in this range, and the gray node means that there are one or more candidate physical servers in this range. In addition, without loss of generality, if a range has two or more physical servers, we still consider the range with only one physical server with sum of the capacity of these physical servers. Based on the above definitions, we can build a weighted undirected graph, which can be expressed as $G(V, E)$.

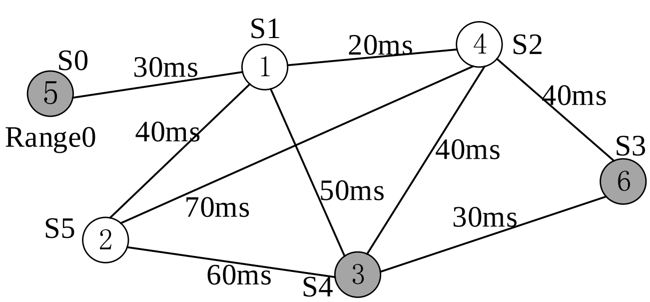


FIGURE 3. User distribution statistic evolution

After user distribution has been quantified when facing the problem of initial deployment unreasonable, we consider to optimize the average path length of all online users to make the most online users obtain the service with as lower latency as possible. In this formulation, $I(i)$ is the optimization variable, which is a binary indicator to denote whether the virtual machine is deployed at the specific physical server or not. As the optimization object, we concentrate on minimize the average path length for all online users with the consideration of both resource constraints and user distribution. Finally, by implementing DARD, one node in $G(V, E)$ will be selected for VM redeployment. The optimization object can be completely express in (1):

$$R_i = \sum_{j=1}^{N} I(i) \cdot W_j \cdot dist_{ij} \tag{1}$$

The followings are complete formulation. Equation (2a) is the minimum optimization goal, which represents the average delay of the system. Inequalities (2b)- (2d) respectively denote the constraints of the memory resource, the CPU computing capability, and the bandwidth capacity. The rationale behind our definition is that the resources consumed by the deployed server must not exceed the residual resources in each dimension.

$$\begin{aligned}
\underset{I(i)}{\text{minimize}} \quad & \sum_{j=1}^{N} I(i) \cdot W_j \cdot dist_{ij} / \sum_{j=1}^{N} W_j && \text{(2a)}\\
\text{subject to} \quad & I(i) \cdot c \leq C_i, \forall i \in N && \text{(2b)}\\
& I(i) \cdot p \leq P_i, \forall i \in N && \text{(2c)}\\
& I(i) \cdot u \leq U_i, \forall i \in N && \text{(2d)}\\
& \sum_{i=1}^{N} I(i) = 1, i \in \{1, N\} && \text{(2e)}
\end{aligned}$$

2) DARD Algorithm

Solving the problem of VM redeployment aims to find a vertex which can make the average path length be shortest while satisfying the resource constraints. Hence, the problem can be solved as a evolution of the shortest path problem. With the resource constraints in Inequality (2b)-(2e), the scale of problem can be largely reduced.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2891115, IEEE Access

Therefore, according to the definitions and analysis above, we can solve this problem in two steps:

Step 1: Select candidate physical servers which meet the condition of resource constraints;

Step 2: Among the servers selected in Step 1, using the following algorithm to further select the one with the best location in terms of the shortest average path length.

Algorithm 1: DARD algorithm.

Input: The graph of nodes and distance:$G(V, E)$; Neighbor node distance matrix:$matrix[i][j]$; The weight of each range: $W[i], i \in N$; Distance to other nodes: $dist[i][j]$.

Output: VM is on node i, $i \in N$; the minimum result R_i.

for $i \in V$ **do**
  $T = V - i$;
  $temp\ node\ t = i$;
  while $T \neq \varnothing$ **do**
    for $j \in T$ **do**
      if $dist[i][t] + dist[t][j] < dist[i][j]$ **then**
        $dist[i][j] = dist[i][t] + dist[t][j]$;
      end
    end
    $min = Inf$;
    for $m \in T$ **do**
      if $dist[i][j] < min$ **then**
        $min = dist[i][m]$;
        $t = m$;
      end
    end
    $T = T - t$;
  end
end
$min = Inf$;
for $i \in V$ **do**
  for $v \in V$ **do**
    $R_i = R_i + dist[i][v] * W[v]$;
  end
  if $R_i < min$ **then**
    $min = R_i$;
  end
end
return *the total shortest path is* R_i; *VM is on node* i.

B. TRM

By implementing DARD, a physical server will be selected for VM redeployment. For traffic redirection, there is still a tradeoff between providing optimal route and minimizing the impact on the core network. In SDN-based Data Centers, SDN controller computes optimal route and manage the whole network by distributing flow table entries to switches. To keep IP address unchanged during VM migration and minimize the impact on the core network, we design a traffic re-direction based VM migration scheme with which active data flows can be seamlessly migrated from the old server to the new one. To make it easier to understand, we give a SDN-based Data Center as Fig. 4 illustrated.

1) Specification and Analysis

As illustrated in Fig. 4, we assume that VM is migrated from the old location to the new one. As mentioned above, to keep the running service from being interrupted, there are two intuitive solutions for redirecting active flows from the old server to the new server (i.e, **redirection-1** and **redirection-4** in the figure). The traffic redirection solution in Liu et al.'s paper [34] is the same as **redirection-4** in Fig. 4. Then, we introduce these two basic solutions in detail and further give our TRM solution.

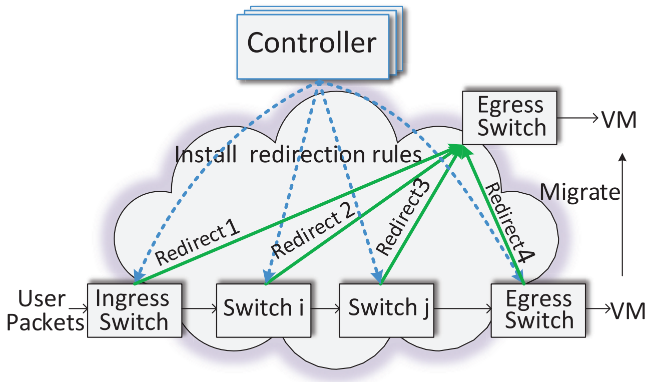


FIGURE 4. VM migration scenario in SDN-based cloud environment

redirection-1: For each data flow, the SDN controller needs to inform the switches in the path from from the Ingress Switch closest to the user to the Egress Switch closest to the new selected physical server (named Egress Switch new) to update their flow table entry. As we know, as a typical C/S (Client/Server) access pattern, one VM usually provides service to many clients at the same time. Therefore, implementation of informing switches for every data flow will lead to a massive impact on core network when VM migration happens, while every online user will have a shortest average route when accessing service provided by the VM in the new location.

redirection-4: Active flows are redirected from the Egress Switch of the old physical server. It becomes obvious that **redirection-4** brings the minimum influence on core network, but it causes a serious triangle routing problem and further decreases online users' experience.

redirection-1 and **redirection-4** are two extremes between impacting on core network and avoiding triangle routing. As shown in Fig. 4, **redirection-2** and **redirection-3** can be regarded as a tradeoff between these two factors. Therefore, in our TRM scheme, we will take these two factors of impacting on network and keeping optimal forwarding path into consideration in order to obtain a good tradeoff. To minimize flow table entries distribution per migration and to minimize the impact on network are equal to each other, we take minimizing flow table entries distribution as the main optimization object while keep the other one as a constraint.

2) TRM algorithm

On the basis of the definitions declared in Section III, some additional definitions in Fig. 5 are added. Assume that the old path from the location of user1 to the initial deployed

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2891115, IEEE Access

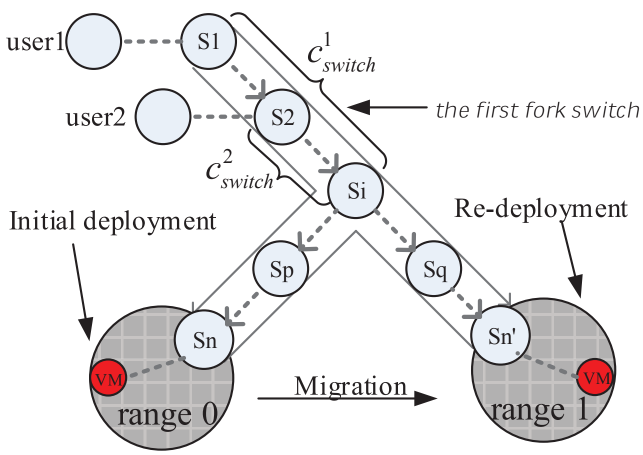


FIGURE 5. An example for better understanding TRM solution

range (range0) is p^1_{old}, and the new path to the redeployed range (range1) is p^1_{new}. $t^1_{path} = (p^1_{old}, p^1_{new})$ is named as a path pair, and $c^1_{switch} = p^1_{old} \cap p^1_{new}$ is the intersection of p^1_{old} and p^1_{new}. Define s as a redirect switch. We know that to redirect active flows on switch $s \in c^1_{switch}$ will not bring the problem of triangle routing, e.g$\{s_1, s_2, \ldots, s_i\}$, in Fig. 5. As the switch s_i is closest to the redeployed range, redirecting on s_i will minimize the impact on network while avoiding the problem of triangle routing. C_{switch} is defined as a set of c_{switch}, a small set of switches satisfying a set of path pair T_{path}, *i.e.* $\forall t_{path} \in T_{path}, \exists s \in c_{switch}$, s.t. s satisfy t_{path}.

To be noted that, there is a essential difference between TRM and the the traditional set-cover problem. If the intersection between any two sets c_{switch} is not null, they must share the same switches after intersecting on the first intersection switch, e.g, s_2 to s_i in Fig. 5.

To prove it, given any two c_{switch} sets, we assume that they are both part of the optimal paths. Assuming that one set is $c^1_{switch} = \{s_1, s_2, \ldots, s_k, \ldots, s_i\}$, and the other is $c^2_{switch} = \{s'_1, s'_2, \ldots, s_k, \ldots, s'_i\}$. Then,

- If $|\{s_k, \ldots, s_i\}| > |\{s_k, \ldots, s'_i\}|$, it leads to a suboptimal c^1_{switch} contradictory with assumption of a optimal path.
- If $|\{s_k, \ldots, s_i\}| < |\{s_k, \ldots, s'_i\}|$, it leads to a suboptimal c^2_{switch} contradictory with assumption of a optimal path.
- Therefore, $|\{s_k, \ldots, s_i\}| = |\{s_k, \ldots, s'_i\}|$ means that they share the same switches after intersecting on the first fork switch. Meanwhile, their last hop s_i will be the candidate switch to be selected to redirect traffic for VM migration.

Based on above definitions and analysis, we can give the mathematical description of TRM solution with two steps as follows:

Step 1: Given any path pair tuple t_{path}, find the satisfying set of switches c_{switch};

Step 2: Given a set n path pair tuple T_{path}, we find the smallest target-switch set TS s.t. $TS \subseteq C_{switch}$ and $\forall\ t_{path} \in T_{path}, TS \cap c_{switch} \neq \varnothing$.

Algorithm 2: TRM algorithm.

```
Input: T_path = {t^1_path, t^2_path, ..., t^n_path}
Output: The smallest Target-Switch set TS;
for each i ∈ [1, n] do
    computes finite satisfied switch set c^i_switch;
end
for i = 0 to n − 1 do
    for j = n − 1 to i do
        if c^i_switch ∩ c^j_switch ≠ ∅ then
            if |c^i_switch| < |c^j_switch| then
                keep the smaller c^i_switch, deletec^j_switch from
                C_switch;
            else
                keep the smaller c^j_switch, deletec^i_switch from
                C_switch;
            end
        else
            both c^j_switch and c^i_switch should be kept in
            C_switch;
        end
    end
end
enumerate the last element of collections contained in the
C_switch set, then add into TS;
return TS
```

C. REDIRECTION MECHANISM

This part will make TRM transparent to online users. A highest priority wildcard flow table as Fig. 6 illustrated will be added into a target switch selected by TRM. So, when data flows with specific destination address (VM old-addr) arrive at the target switch, these flows will match the wildcard flow table first and then be forwarded to Egress Switch-new hop-by-hop. Therefore, by adding such wildcard flow tables to each target switch, data flows can be effectively redirected to the VM's new location.

D. SYNCHRONOUS VM MIGRATION

While facing with the challenge of how to further reduce the impact from VM migration, SDN makes it much easier for a CSP to monitor the specific status of the whole Data Center network by taking the advantages of its abstraction of the centralized network control plane.

Fig. 7 shows a complete process of VM management in a SDN-based Data Center. In this figure, the complete process can be divided into three phases. In **Phase 1**, a VM might be initially deployed on a physical server to provide some online service. Because of initial deployment unreasonable problem, VM redeployment should be introduced to solve this problem in order to provide a good service for the most online users in **Phase 2**. After the accomplishment of VM redeployment in **Phase 3**, most online users will access the VM with a low delay and enjoy a better user experience.

Through the description of the above three phases, **Phase 2** is the most crucial phase in VM redeployment. However,

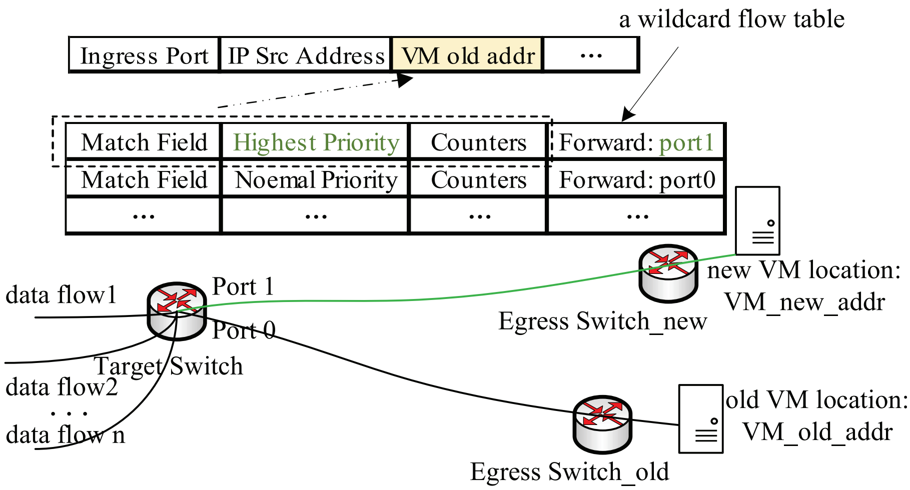


FIGURE 6. Matching the wildcard flow table

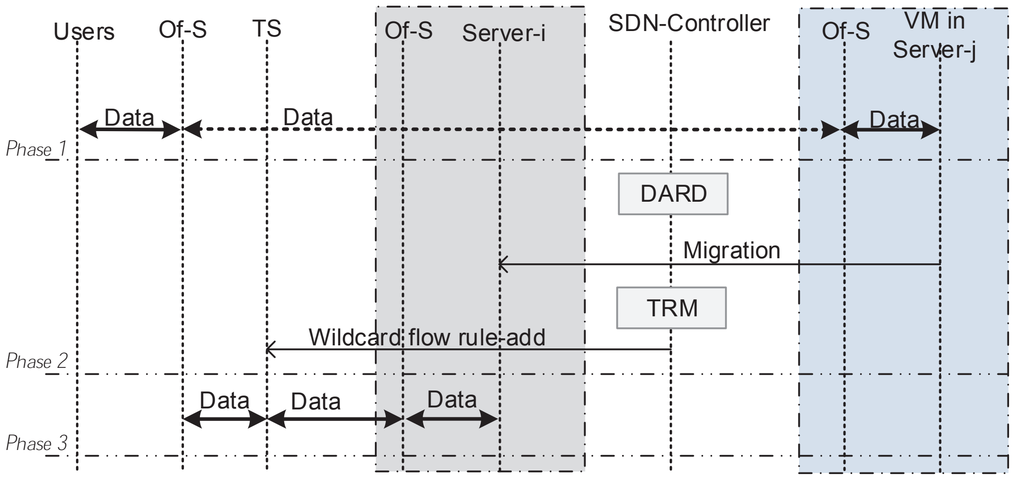


FIGURE 7. DARD and TRM in VM(s) management

operations serially executed in **Phase 2** sometimes will cause a long downtime when VM live migration happens. With the centralized feature of SDN controller, it will make more sense to concurrently implement the operations in **Phase 2**.

Furthermore, in Fig. 7, we introduce a synchronous VM migration scheme in SDN-based Data Center network. When a CSP finds a initial deployment unreasonable problem in its Data Center, DARD is needed for the CSP to find a suitable physical server. Once a destination server has been selected, a CSP should start using Pre-Copy strategy to copy VM states in **Live VM migration phase**. Meanwhile, it is adjudged wise to start **Network configuration phase**.

From Fig. 7, we can draw an obvious difference between synchronous processing with DARD and TRM process as shown in Fig. 8. This difference will change a lot, especially in VM migration. In the VM management framework, DARD is a method to find a user distribution aware optimal redeployment location. After finding the target destination server at $t1$, we synchronize **Live VM migration phase** with **Network configuration phase**. Different from the traditional TCP/IP networks, these two phases can be executed at the same time so as to help make live VM migration a more efficient method for VM management.

V. PERFORMANCE EVALUATION

In this section, we give the performance evaluation of our proposed schemes. We use MATLAB to numerically analysis the DARD scheme with different user amounts. We compare the average service delay for users between the initial VM deployment and our DARD redeployment. And then we use Mininet (version 2.3.0d4) as the emulator and Floodlight (version 1.2) as the SDN controller to simulate the TRM scheme. We use the path length and the number of distributed flow table entries as the performance metrics, and further we compare our scheme with other two solutions, with user amount varying.

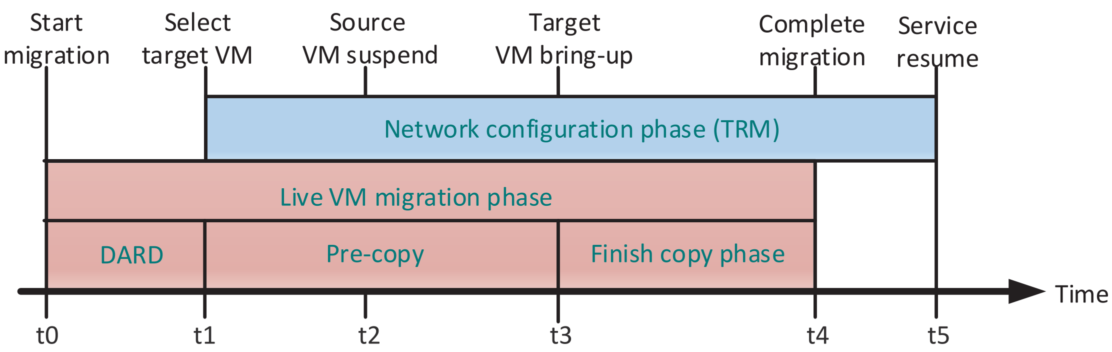


FIGURE 8. Synchronous VM(s) migration

As illustrated in Fig. 3, we give the network topology of a data center network. Then we randomly select a group of typical high dynamic users as samples, and here we use 64 users as an example. There are six ranges in the topology and three physical servers located in range 0, range 3, and range 4. The virtual machine was originally deployed in domain 0. The initial distribution of online users in each range is 19, 3, 11, 17, 9, 5, respectively. Then we set a 6-row 6-column transfer matrix to represent the dynamic user distribution transfer model, in which each row represents the transfer probability of an online user's moving to another range from the current range.

In the following evaluations, we analyse our proposed scheme from three perspectives with the dynamic distribution of users. Firstly, we analyze how user distribution impacts on average service delay. Secondly, we compare TRM solution with another two distinct solutions about the total routing path length (the total service delay for all users.). Finally, we evaluate how these three solutions influence flow table entries distribution. For the sake of simplicity of explanation, we still rename these two distinct solutions according to the switches they selected.

1) Impact on average service delay

To evaluate the impact of changes in user distribution, we periodically record the average service delay as shown in Fig. 9, that is the average time delay of users getting services from the VM. And we compare the average delay of initial deployment and redeployment for different numbers of users in the system as shown in Fig. 10. Simulation results show that our DARD solution is relatively stable in terms of performance fluctuation with respect to the number of users. In addition, our proposed DARD redeployment solution can reduce half of the average delay of that of the initial deployment regardless of the number of users.

2) Impact on path length

As shown in Fig. 11, we record the routing path length in network when the number of users varies to evaluate TRM with other two migration solutions. The path length refers to the total length of all users to the VM under one of

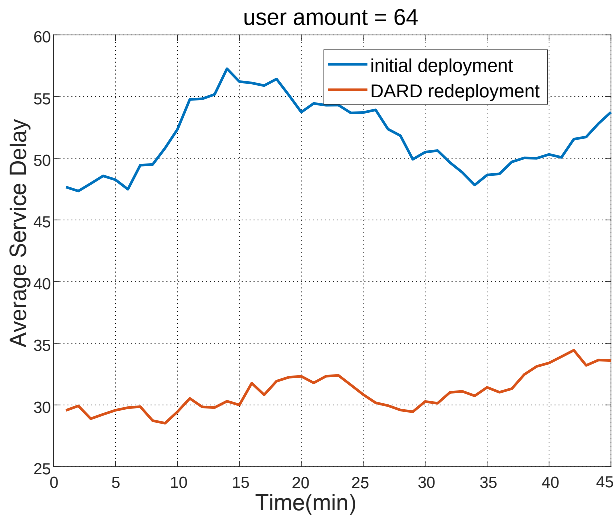


FIGURE 9. Impacts of user distribution on average service delay

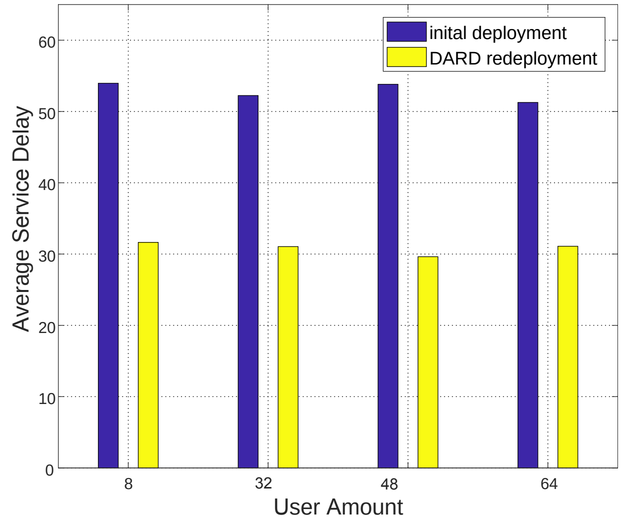


FIGURE 10. Impacts of user amount on average service delay

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2891115, IEEE Access

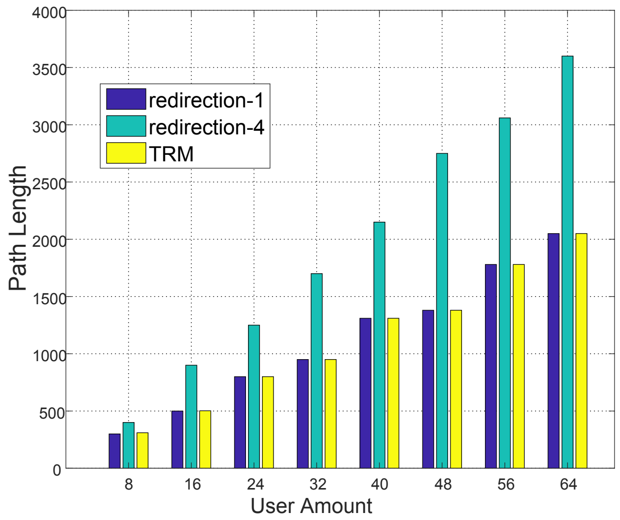


FIGURE 11. Impacts of user amount on path length

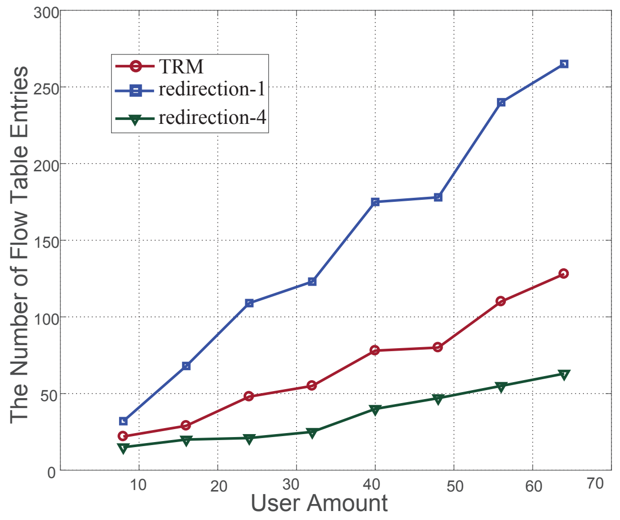


FIGURE 12. Impacts of user amount on the number of distributed flow table entries

the three different migration solutions. Because of the long triangle routing caused by redirection-4 solution, it is very obvious that the other two solutions (redirection-1 and our proposed TRM) are far superior to it. Moreover, TRM gets approximately the same performance in the total path length with redirection-1 solution, as the main constraint of TRM solution is to guarantee optimal routing. From the evaluation in the given scenario, the performance of our TRM solution is about two times better than redirection-4 solution.

3) Impact on flow tables

In a SDN-based Data Center network, flow table entries distribution directly influence the whole network performance. Therefore, it is reasonable to select flow table entry distribution as a metric to evaluate its impact on network performance. Fig. 12 declares that redirection-1 produces a huge amount of flow table entries for per VM migration, while the number is much smaller in TRM and redirection-4. This shows that flow table has significant influence on core network.

When taking both Fig. 11 and Fig. 12 into consideration, we can conclude that TRM has advantages in improving the overall performance. TRM can provide a lower service delay for online users, and also has slight positive influence on network performance. It will be a meaningful mechanism for a CSP to trade acceptable influence on network for a higher quality of service to online users.

VI. CONCLUSION

VM deployment and live VM migration are two very important issues for CSP. A CSP can address these two issues to achieve an efficient management operation, like hardware maintenances, load balancing, etc. However, without considering the variation of online user distribution, virtual machine deployment and migration may not guarantee QoE to users, especially when the number of online users is increasing rapidly.

In this paper, we introduce a new idea to support VM redeployment in consideration of online users distribution. We investigate existing approaches for virtual machine management and show how user distribution affect service quality. On the basis of this, we propose a user distribution aware virtual machine redeployment scheme. Based on the centralized control of SDN controller, it will be easier to maintain the states of the whole network, and make decisions for CSP to redeploy unreasonable VM deployments based on online user distribution. To keep the running services from being interrupted during VM migration, it is also necessary to implement a live VM migration scheme. On the basis of SDN architecture, a traffic redirecting migration (TRM) algorithm is proposed to determine the right tradeoff between influence on network and service quality. Furthermore, with our VM management scheme, we finally realize a synchronous VM migration framework which significantly decreases the service downtime during VM migration.

In our future work, we will further explore the multi-virtual machines migration issue in both single controller scenario and multi controllers scenario. Moreover, the deployment of controllers and the impact of flow table sizes will also be taken into our consideration.

ACKNOWLEDGMENT

The authors sincerely thank the anonymous referees for their invaluable suggestions that have led to the present improved version from the original manuscript.

REFERENCES

[1] M. A. Alsmirat, Y. Jararweh, I. Obaidat, and B. B. Gupta, "Internet of surveillance: a cloud supported large-scale wireless surveillance system," The Journal of Supercomputing, vol. 73, no. 3, pp. 973–992, 2017.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2891115, IEEE Access

[2] J. Pei, P. Hong, K. Xue, D. Li et al., "Efficiently embedding service function chains with dynamic virtual network function placement in geo-distributed cloud system," IEEE Transactions on Parallel and Distributed Systems, 2018.

[3] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, "Network function virtualization: State-of-the-art and research challenges," IEEE Communications Surveys & Tutorials, vol. 18, no. 1, pp. 236–262, 2016.

[4] A. M. Medhat, T. Taleb, A. Elmangoush, G. A. Carella, S. Covaci, and T. Magedanz, "Service function chaining in next generation networks: State of the art and research challenges," IEEE Communications Magazine, vol. 55, no. 2, pp. 216–223, 2017.

[5] Z. Usmani and S. Singh, "A survey of virtual machine placement techniques in a cloud data center," Procedia Computer Science, vol. 78, pp. 491–498, 2016.

[6] J. Pei, P. Hong, K. Xue, D. Li et al., "Resource aware routing for service function chains in sdn and nfv-enabled network," IEEE Transactions on Services Computing, 2018.

[7] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," Wireless Communications and Mobile Computing, vol. 13, no. 18, pp. 1587–1611, 2011.

[8] S. Secci, P. Raad, and P. Gallard, "Linking virtual machine mobility to user mobility," IEEE Transactions on Network and Service Management, vol. 13, no. 4, pp. 927–940, 2016.

[9] W. Zhang, S. Han, H. He, and H. Chen, "Network-aware virtual machine migration in an overcommitted cloud," Future Generation Computer Systems, vol. 76, pp. 428–442, 2017.

[10] B. Abali, C. Isci, J. O. Kephart, S. K. McIntosh, and D. Sarma, "Live virtual machine migration quality of service," 2017, uS Patent 9,619,258.

[11] F. Hao, M. Kodialam, T. Lakshman, and S. Mukherjee, "Online allocation of virtual machines in a distributed cloud," IEEE/ACM Transactions on Networking (TON), vol. 25, no. 1, pp. 238–249, 2017.

[12] D. Li, P. Hong, K. Xue, and J. Pei, "Virtual network function placement considering resource optimization and sfc requests in cloud datacenter," IEEE Transactions on Parallel and Distributed Systems, vol. 29, no. 7, pp. 1664–1677, 2018.

[13] A. Khosravi, A. Nadjaran Toosi, and R. Buyya, "Online virtual machine migration for renewable energy usage maximization in geographically distributed cloud data centers," Concurrency and Computation: Practice and Experience, vol. 29, no. 18, p. e4125, 2017.

[14] Y. Li, X. Tang, and W. Cai, "Play request dispatching for efficient virtual machine usage in cloud gaming," IEEE Transactions on Circuits and Systems for Video Technology, vol. 25, no. 12, pp. 2052–2063, 2015.

[15] J. Zhang, H. Huang, and X. Wang, "Resource provision algorithms in cloud computing: A survey," Journal of Network and Computer Applications, vol. 64, pp. 23–42, 2016.

[16] J. Zhang, F. Ren, R. Shu, T. Huang, and Y. Liu, "Guaranteeing delay of live virtual machine migration by determining and provisioning appropriate bandwidth," IEEE Transactions on Computers, no. 1, pp. 1–1, 2016.

[17] T. Wood, K. Ramakrishnan, P. Shenoy, and J. Van der Merwe, "Cloudnet: dynamic pooling of cloud resources by live wan migration of virtual machines," in ACM Sigplan Notices, vol. 46, no. 7. ACM, 2011, pp. 121–132.

[18] T. Kondo, R. Aibara, K. Suga, and K. Maeda, "A mobility management system for the global live migration of virtual machine across multiple sites," in 2014 IEEE 38th International Computer Software and Applications Conference Workshops (COMPSACW). IEEE, 2014, pp. 73–77.

[19] D. Pompili, A. Hajisami, and T. X. Tran, "Elastic resource utilization framework for high capacity and energy efficiency in cloud ran," IEEE Communications Magazine, vol. 54, no. 1, pp. 26–32, 2016.

[20] C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live migration of virtual machines," in Proceedings of the 2nd Conference on Symposium on Networked Systems Design & Implementation-Volume 2. USENIX Association, 2005, pp. 273–286.

[21] C. DeCusatis and R. B. Krishnamurthy, "Virtual machine mobility using openflow," 2017, uS Patent 9,609,086.

[22] E. Keller, S. Ghorbani, M. Caesar, and J. Rexford, "Live migration of an entire network (and its hosts)," in Proceedings of the 11th ACM Workshop on Hot Topics in Networks. ACM, 2012, pp. 109–114.

[23] U. Deshpande and K. Keahey, "Traffic-sensitive live migration of virtual machines," Future Generation Computer Systems, vol. 72, pp. 118–128, 2017.

[24] S. Nathan, U. Bellur, and P. Kulkarni, "Towards a comprehensive performance model of virtual machine live migration," in Proceedings of the Sixth ACM Symposium on Cloud Computing. ACM, 2015, pp. 288–301.

[25] E. Silvera, G. Sharaby, D. Lorenz, and I. Shapira, "Ip mobility to support live migration of virtual machines across subnets," in Proceedings of SYSTOR 2009: The Israeli Experimental Systems Conference. ACM, 2009, p. 13.

[26] P. Raad, S. Secci, D. C. Phung, A. Cianfrani, P. Gallard, and G. Pujolle, "Achieving sub-second downtimes in large-scale virtual machine migrations with lisp," IEEE Transactions on Network and Service Management, vol. 11, no. 2, pp. 133–143, 2014.

[27] R. Xie, Y. Wen, X. Jia, and H. Xie, "Supporting seamless virtual machine migration via named data networking in cloud data center," IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 12, pp. 3485–3497, 2015.

[28] A. Blenk, A. Basta, and W. Kellerer, "Hyperflex: An sdn virtualization architecture with flexible hypervisor function allocation," in 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), May 2015, pp. 397–405.

[29] A. Blenk, A. Basta, J. Zerwas, and W. Kellerer, "Pairing sdn with network virtualization: The network hypervisor placement problem," in 2015 IEEE Conference on Network Function Virtualization and Software Defined Network (NFV-SDN), Nov 2015, pp. 198–204.

[30] C. Sieber, A. Blenk, A. Basta, and W. Kellerer, "hvbench: An open and scalable sdn network hypervisor benchmark," in 2016 IEEE NetSoft Conference and Workshops (NetSoft), June 2016, pp. 403–406.

[31] A. Basta, A. Blenk, H. B. Hassine, and W. Kellerer, "Towards a dynamic sdn virtualization layer: Control path migration protocol," in 2015 11th International Conference on Network and Service Management (CNSM), Nov 2015, pp. 354–359.

[32] A. Satpathy, S. K. Addya, A. K. Turuk, B. Majhi, and G. Sahoo, "Crow search based virtual machine placement strategy in cloud data centers with live migration," Computers Electrical Engineering, vol. 69, pp. 334–350, July 2018.

[33] T. G. Rodrigues, K. Suto, H. Nishiyama, N. Kato, and K. Temma, "Cloudlets activation scheme for scalable mobile edge computing with transmission power control and virtual machine migration," IEEE Transactions on Computers, vol. 67, no. 9, pp. 1287–1300, Sept 2018.

[34] J. Liu, Y. Li, and D. Jin, "Sdn-based live vm migration across datacenters," in ACM SIGCOMM Computer Communication Review, vol. 44, no. 4. ACM, 2014, pp. 583–584.

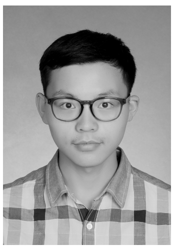
[35] A. Mayoral, R. Vilalta, R. Muoz, R. Casellas, and R. MartÍÍnez, "Experimental seamless virtual machine migration using an integrated sdn it and network orchestrator," in 2015 Optical Fiber Communications Conference and Exhibition (OFC), March 2015, pp. 1–3.

[36] U. Mandal, M. F. Habib, S. Zhang, P. Chowdhury, M. Tornatore, and B. Mukherjee, "Heterogeneous bandwidth provisioning for virtual machine migration over sdn-enabled optical networks," in OFC 2014, March 2014, pp. 1–3.

[37] N. K. Sharma, P. Sharma, and R. M. R. Guddeti, "Energy efficient quality of service aware virtual machine migration in cloud computing," in 2018 4th International Conference on Recent Advances in Information Technology (RAIT), March 2018, pp. 1–6.

JIN QIN received his B.S. degree from the Department of Automation, University of Science and Technology of China (USTC), Hefei, China, in 2015. He is currently working toward his Ph.D. degree in information security from the School of Cyber Security, USTC. His research interests include next-generation Internet architecture design and performance optimization.

YIZHENG WU received his B.S. degree from the School of Computer Science and Information Engineering, Hefei University of Technology, in 2013, and his M.S. degree in Information and Communication Engineering from the Department of Electrical Engineering and Information Science (EEIS), University of Science and Technology of China (USTC), Hefei, China, in 2016. Now he is a researches in Huawei Technology, Shanghai, China. His research interests include next-generation Internet and cloud computing.



YUTONG CHEN received his B.S. degree from the School of Electronics and Information, Jilin University, in 2017. He is currently working toward his M.S. degree in Information and Communication Engineering from the Department of Electronic Engineering and Information Science (EEIS), University of Science and Technology of China (USTC). His research interests include next-generation Internet architecture design and performance optimization.

KAIPING XUE (M'09-SM'15) received his B.S. degree from the Department of Information Security, University of Science and Technology of China (USTC), in 2003 and received his Ph.D. degree from the Department of Electronic Engineering and Information Science (EEIS), USTC, in 2007. From May 2012 to May 2013, he was a postdoctoral researcher with Department of Electrical and Computer Engineering, University of Florida. Currently, he is an Associate Professor in the School of Cybersecurity and Department of EEIS, USTC. His research interests include next-generation Internet, distributed networks and network security.

DAVID S.L. WEI (SM'07) received his Ph.D. degree in Computer and Information Science from the University of Pennsylvania in 1991. He is currently a Professor of Computer and Information Science Department at Fordham University. From May 1993 to August 1997 he was on the Faculty of Computer Science and Engineering at the University of Aizu, Japan (as an Associate Professor and then a Professor). He has authored and co-authored more than 100 technical papers in various archival journals and conference proceedings. He served on the program committee and was a session chair for several reputed international conferences. He was an associate editor of IEEE Transactions of Cloud Computing, a lead guest editor of IEEE Journal on Selected Areas in Communications for the special issue on Mobile Computing and Networking, a lead guest editor of IEEE Journal on Selected Areas in Communications for the special issue on Networking Challenges in Cloud Computing Systems and Applications, a guest editor of IEEE Journal on Selected Areas in Communications for the special issue on Peer-to-Peer Communications and Applications, and a lead guest editor of IEEE Transactions on Cloud Computing for the special issue on Cloud Security. He is currently an Associate Editor of Journal of Circuits, Systems and Computers, and a guest editor of IEEE Transactions on Big Data for the special issue on Trustworthiness in Big Data and Cloud Computing Systems. Currently, His research interests include cloud computing, big data, IoT, and cognitive radio networks.