Received March 31, 2019, accepted April 25, 2019, date of publication May 7, 2019, date of current version June 3, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2915333

Virtual Network Embedding in Fiber-Wireless Access Networks for Resource-Efficient IoT Service Provisioning

YEJUN LIU[1,2], YAHE YANG[2], PENGCHAO HAN[2], ZEYU SHAO[2], AND CHENGMING LI[3]

[1]School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
[2]School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China
[3]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518116, China

Corresponding author: Chengming Li (cm.li@siat.ac.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61775033, Grant 61501104, Grant 61801063, and Grant 61602462, and in part by the Fundamental Research Funds for the Central Universities under Grant N161608001.

ABSTRACT The rise of Industry 4.0 is pushing more machine devices into the connection of the communication network. Internet of Things (IoT) emerges as an enabling technique to support automatic information transmission and the universal interconnection among massive machine devices. Unlike the human type communication (HTC) in the conventional Internet, IoT features machine type communication (HTC) with small data granularity and massive connections, which impose new challenges on the network resource allocation earlier designed for HTC communication. One of the challenging issues is how we can manage the network resource for the IoT service provisioning. As a promising network architecture for the ''last-mile'' access in 5G communication, Fiber-Wireless (FiWi) access network compromises the high bandwidth capacity of the optical access network and the flexibility of wireless access network. Especially, network virtualization as an emerging technique provides FiWi with the feasibility for the coordinated bandwidth allocation of wireless access network and optical access network. The evolution of FiWi access network should be not only oriented to the technological breakthrough in regular traffic, but also be supportive of the emerging IoT traffic. In this paper, we focus on the resource allocation problem in FiWi access network supporting IoT service. We first analyze the dynamics of regular traffic and propose the traffic prediction method based on Q-Learning. Then, we propose a virtual network embedding algorithm to map exactly the virtual networks of IoT traffic to the idle resource of the virtual networks of regular traffic without degrading the performance of regular traffic. Thus, the proposed algorithm can improve the resource utilization more effectively. The simulation results demonstrate that the proposed algorithm can achieve a higher acceptance ratio and lower migration ratio of IoT traffic.

INDEX TERMS FiWi access network, IoT service, network virtualization, network resource allocation.

I. INTRODUCTION

Due to the widespread investigation in Industrial automation, smart city and intelligent transportation, IoT is gaining considerable attentions from both industrial and academic communities and has been widely recognized as a promising technique with dramatic change into our world. It is anticipated that the IoT devices around the world will reach up to 18 billion in 2022 [1]. Unlike the regular traffic in current Internet, which usually arises from the Human Type Communications (HTC) with large bandwidth granularity and sensitive QoS requirement, the IoT traffic features Machine Type Communications (MTC) with massive connections and small-sized data. The rapid growth of IoT devices brings a new issue with the network resource allocation in current Internet, particularly the ''last-mile'' access network that is recognized as a major bottleneck region of Internet for a long time.

As a promising network architecture for the fifth generation (5G) and beyond 5G (B5G), Fiber-Wireless (FiWi) converged access network combines Passive Optical Network (PON) with Wireless Mesh Network (WMN) and inherits the advantages of optical access and wireless access [2], [3]. Optical access can provide the satisfactory

The associate editor coordinating the review of this manuscript and approving it for publication was Jun Wu.

bandwidth, transmission quality and stability, while wireless access enables the ubiquitous connection with high flexibility and low cost. Therefore, from the perspective of access network architecture, FiWi is an advisable choice to provide communication connections for IoT service [4], [5].

Although the diverse technological superiority, the heterogeneity of optical subnetwork and wireless subnetwork in terms of topology, transmission mode and resource attributes makes both subnetworks have to be managed independently. The independence of network management is one of the primary reasons that restrict the interoperability between optical subnetwork and wireless subnetwork, which discourages the resource allocation in FiWi access network from being globally optimized. With the spread of various IoT applications, FiWi access network will have to undertake the diversity of service requirements for QoS, security, maintenance and functional upgrade, which inflicts the network with an ossified resource allocation [6].

Network virtualization emerges as an effective solution to address the network ossification issue and has received many research efforts. By pooling the network resource, network virtualization allows multiple Virtual Networks (VN) to exist simultaneously on the same physical network infrastructure. In the network virtualization scenario, the role of traditional Internet Service Provider (ISP) is divided into Infrastructure Provider (InP) and Service Provider (SP). As the owner of network infrastructure, InP just needs to take charge of the physical network resource and create the resource pool by means of virtualization. By renting the network resource from InP, the SPs can create and manage their own VNs in a customized way, independent of each other to support different IoT applications. Therefore, network virtualization is also recognized as a viable way to shield the heterogeneity of underlying facilities in FiWi access network and achieve the globally optimal network resource allocation. The virtualized FiWi access network will be a more eligible network paradigm to bear the various IoT applications [7].

Virtual network embedding is one of the key issues in network virtualization which addresses where the VNs should be embedded in the physical network and the resource allocation for them. Although VNE has been investigated in many works, few of them focus on this issue in the context of virtualized FiWi access network, especially taking into account the resource allocation for IoT service [8], [10], [11]. Because of the inherent difference between IoT service and regular service in traffic profile, we could not simply regard a VN of IoT service as that of regular service. If we embed the VN of IoT service into the physical network within dedicated resource, the resource may be underutilized for the small resource granularity of IoT service. Some of the related works on VNE focused on the dynamics of the traffic load of regular service and attempted to reallocate the idle resource that arises when traffic load is low for network resource reuse. However, the idle resource arising from low traffic load usually appears fragmentarily and it is hard to be reused by some types of regular service that demands network resource intensively. In the scenario of regular service coexisting with IoT service, the idle resource left by the VNs of regular service while low-loaded will be properly enough to carry the IoT service due to the complementary nature of their resource demands. This observation provides us with an opportunity to enable the resource reuse between the regular service and IoT service.

In this paper, we focus on an open issue of resource allocation in virtualized FiWi access network with the coexistence of regular service and IoT service, which remains less mentioned in previous works. We propose a VNE algorithm for the resource allocation of regular service. A Q-learning method is proposed to predict the network traffic load. Based on the predicted traffic load, another VNE algorithm is proposed for the resource allocation of VNs of IoT service. The idle resource arising from the VNs of regular service when they are not fully loaded will be preferentially allocated to the VNs of IoT service for the purpose of network resource reuse. The VNs of IoT service have to be migrated and return the resource to the VNs of regular service when the resource of the VNs of regular service is no longer enough to afford the growing load. We define the InP revenue models for both regular service and IoT service and a penalty factor is introduced into the InP revenue of IoT service to indicate the degraded QoS when the IoT traffic is migrated.

The remainder of this paper is organized as follows. In Section II, we describe the architecture of virtualized FiWi access network and formulate the VNE problem for both regular service and IoT service. In Section III, we present the proposed VNE algorithms. Performance evaluation is made in Section IV. Finally, we make the concluding words in Section V.

II. NETWORK ARCHITECTURE AND PROBLEM FORMULATION

A. VIRTUALIZED FIWI ACCESS NETWORK

Figure 1 shows the virtualized FiWi access network architecture, which mainly includes three layers, underlying physical network, InP layer and SP layer. The underlying physical network consists of PON at back-end and WMN at front-end. In WMN, the wireless routers act as access points for the users of both regular service and IoT service [12], [13]. The traffic from user-ends can be forwarded to wireless gateways via multi-hop wireless paths. In PON, the Optical Network Units (ONU) exist to aggregate the traffic from WMN and send it up to OLT through the distribution fiber links and then feeder fiber link. OLT manages the resource allocation in PON and serves as an interface between FiWi access network and Internet. The InP abstracts the underlying physical network resources (including the CPU resource at physical nodes and the bandwidth resource on physical links) to form a physical network topology, and stores resource information of physical nodes and links [14], [15]. Thereby, InP can realize a unified resource allocation based on a global network view across wireless subnetwork and optical subnetwork. To bear one type of service, regular service or IoT service, each SP

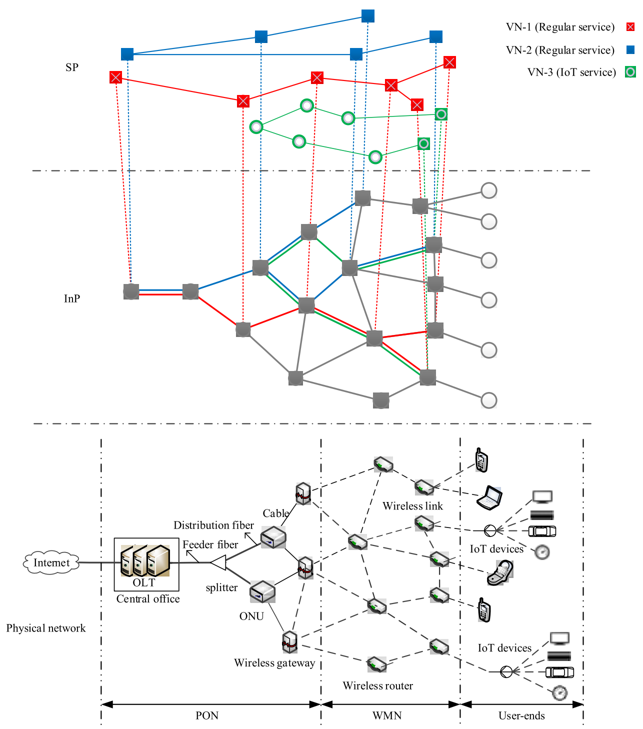


FIGURE 1. Virtualized FiWi access network architecture.

puts forward one VN request with the resource requirements including CPU resource requirements of virtual nodes and bandwidth resource requirements of virtual links. The VNs with their own service types can share the physical network resources such as the VN-1 and VN-2 in regular service and the VN-3 in IoT service shown in Fig. 1. The VN of regular service is generated with an aggregated topology with various number of virtual nodes and virtual links to reflect the access of HTC data to Internet, while the VN of IoT service is generated with a pair of virtual nodes to reflect the local connection of MTC data, e.g., from sensors to actuators. Due to the larger data packet size of regular service, the VNs of regular service require network resources within a larger scale than the VNs of IoT service [16], [17]. Each VN request arrives into and departs the network dynamically with the bandwidth resource demands of virtual links, the CPU resource demands and the expected locations of virtual nodes.

B. PHYSICAL NETWORK MODEL

The physical network can be represented as an undirected graph: $G^S = (N^S, L^S)$, where N^S denotes a set of physical nodes and L^S denotes a set of physical links, respectively. For each physical node $n_x^S \in N^S(x = 1, 2, \cdots, |N^S|)$, where $|N^S|$ represents the total number of nodes in the physical network, and n_x^S is characterized by the following elements.

- C_x^S : the residual CPU capacity of the physical node n_x^S.
- f_x^S : the CPU load of n_x^S.
- $type_x^S$: the type of n_x^S, $type_x^S \in \{OLT, ONU, WR\}$, where OLT denotes OLT, ONU denotes ONU, and WR denotes wireless gateway or wireless router.
- lct_x^S : the geographic location of n_x^S.

Each physical link $l_{(x,y)}^S \in L^S$ has a pair of end nodes n_x^S and $n_y^S(x, y = 1, 2, \cdots, |N^S|)$ and $l_{(x,y)}^S$ is characterized by the following elements.

- $B_{(x,y)}^S$: the residual bandwidth capacity of $l_{(x,y)}^S$.
- $type_{(x,y)}^S$: the type of the physical link $l_{(x,y)}^S$, $type_{(x,y)}^S \in \{FIBER, WCABLE, WIRLESS\}$ where $FIBER$ indicates fiber link, $WCABLE$ indicates cable and $WIRLESS$ indicates wireless link, respectively.

C. VIRTUAL NETWORK REQUEST OF REGULAR SERVICE

A virtual network of regular service can be represented as an undirected graph, $G_v^{VR} = (N_v^{VR}, L_v^{VR})$, where N_v^{VR} represents the set of virtual nodes, and L_v^{VR} represents the set of virtual links in the vth virtual network of regular service. The duration of the vth virtual network of regular service is T_v^{VR}. For the ith virtual node $n_{v,i}^{VR} \in N_v^{VR}(i = 1, 2, \cdots, |N_v^{VR}|)$, where $|N_v^{VR}|$ represents the total number of virtual nodes in the vth virtual network of regular service. $n_{v,i}^{VR}$ consists of the following elements.

- $c_{v,i}^{VR}$: the CPU resource request of $n_{v,i}^{VR}$.
- $type_{v,i}^{VR}$: the type of $n_{v,i}^{VR}$, $type_{v,i}^{VR} \in \{A, T\}$, where A represents the access node that should be embedded onto the OLT, and T represents the forwarding node that should be embedded onto the ONU, the wireless gateway and the wireless router in the physical network.
- $lct_{v,i}^{VR}$: the expected location of $n_{v,i}^{VR}$.
- $\Delta lct_{v,i}^{VR}$: the maximum allowable offset that $n_{v,i}^{VR}$ deviates from $lct_{v,i}^{VR}$ after it is embedded to the underlying physical network;
- $\xi_{v,i,x}^{VR}$: a binary variable, taking 1 if $n_{v,i}^{VR}$ is embedded to the physical node n_x^S, and 0 otherwise.

For the virtual link $l_{v,(i,j)}^{VR}$ between the virtual nodes $n_{v,i}^{VR}$ and $n_{v,j}^{VR}$, there is $l_{v,(i,j)}^{VR} \in L_v^{VR}(i, j = 1, 2, \cdots, |N_v^{VR}|)$. $l_{v,(i,j)}^{VR}$ consists of the following elements.

- $b_{v,(i,j)}^{VR}$: the maximum bandwidth request of $l_{v,(i,j)}^{VR}$.
- $w_{v,(i,j)}^{VR}$: the bandwidth resource that InP has already allocated to $l_{v,(i,j)}^{VR}$ but remains idle at the current time.
- $len_{v,(i,j)}^{VR}$: the length constraint of the path by which $l_{v,(i,j)}^{VR}$ is embedded onto the underlying physical network.
- $\zeta_{v,(i,j),(x,y)}^{VR}$: a binary variable, taking 1 if the virtual link $l_{v,(i,j)}^{VR}$ is embedded to the physical link $l_{(x,y)}^S$, and 0 otherwise.

D. VIRTUAL NETWORK REQUEST OF IOT SERVICE

The virtual network of IoT service can be represented as an undirected graph $G_v^{VI} = (N_v^{VI}, l_{v,(s,d)}^{VI})$, where N_v^{VI} represents a set of virtual nodes, and $l_{v,(s,d)}^{VI}$ represents the only virtual link of the vth virtual network of IoT service. The duration of the vth virtual network of IoT service is T_v^{VI}. For the vth virtual network of IoT service, there are only two virtual nodes, one of which is the source node $n_{v,s}^{VI}$ and another one is the destination node $n_{v,d}^{VI}$. Each virtual node of IoT service, e.g., $n_{v,s}^{VI}$ (or $n_{v,d}^{VI}$) consists of the following elements.

- $c_{v,s}^{VI}$ (or $c_{v,d}^{VI}$): the CPU resource request of $n_{v,s}^{VI}$ (or $n_{v,d}^{VI}$);
- $lct_{v,s}^{VI}$ (or $lct_{v,d}^{VI}$): the expected location of $n_{v,s}^{VI}$ (or $n_{v,d}^{VI}$);
- $\Delta lct_{v,s}^{VI}$(or $\Delta lct_{v,d}^{VI}$) : the maximum allowable offset that $n_{v,s}^{VI}$ (or $n_{v,d}^{VI}$) deviates from $lct_{v,s}^{VI}$(or $lct_{v,d}^{VI}$) when it is embedded to the underlying physical network.
- $\xi_{v,s,x}^{VI}$ (or $\xi_{v,d,x}^{VI}$): a binary variable, taking 1 if $n_{v,s}^{VI}$ (or $n_{v,d}^{VI}$) is embedded to the physical node n_x^S, and 0 otherwise.

The virtual link $l_{v,(s,d)}^{VI}$ represents the link between the virtual nodes $n_{v,s}^{VI}$ and $n_{v,d}^{VI}$. $l_{v,(s,d)}^{VI}$ consists of the following elements.

- $b_{v,(s,d)}^{VI}$: the bandwidth request of $l_{v,(s,d)}^{VI}$.
- $len_{v,(s,d)}^{VI}$: the length constraint of the path by which $l_{v,(s,d)}^{VI}$ is embedded onto the underlying physical network.
- $\zeta_{v,(s,d),(x,y)}^{VI}$: a binary variable, taking 1 if $l_{v,(s,d)}^{VI}$ is embedded to the physical link $l_{v,(s,d)}^{S}$, and 0 otherwise.

E. CONSTRAINTS OF NODE EMBEDDING

For any VN of regular service or IoT service, each virtual node can only be embedded to one physical node [18], and the virtual nodes in the same virtual network must be embedded to different physical nodes, as in following equations:

$$\sum_{x=1}^{|N^S|} \xi_{v,i,x}^{VR} = 1, \quad \forall v, i \tag{1}$$

$$\sum_{x=1}^{|N^S|} \xi_{v,s,x}^{VI} = \sum_{x=1}^{|N^S|} \xi_{v,d,x}^{VI} = 1 \quad \forall v, i \tag{2}$$

$$\sum_{i=1}^{|N_v^{VR}|} \xi_{v,i,x}^{VR} \leq 1, \quad \forall v, x \tag{3}$$

$$\xi_{v,s,x}^{VI} + \xi_{v,d,x}^{VI} \leq 1 \quad \forall v, x \tag{4}$$

The location of the physical node in which a virtual node is embedded should satisfy the geographic location constraint of this virtual node, as follows:

$$\left| lct_{v,i}^{VR} - lct_x^S \right| \leq \Delta lct_{v,i}^{VR} \quad \forall v, i, x \tag{5}$$

$$\left| lct_{v,s}^{VI} - lct_x^S \right| \leq \Delta lct_{v,s}^{VI} \quad \forall v, x \tag{6}$$

$$\left| lct_{v,d}^{VI} - lct_x^S \right| \leq \Delta lct_{v,d}^{VI} \quad \forall v, x \tag{7}$$

Furthermore, a physical node should have the residual CPU resource enough to afford the CPU resource request of each virtual node embedded on this physical node [19].

$$\sum_{i=1}^{|N_v^{VR}|} \xi_{v,i,x}^{VR} \cdot c_{v,i}^{VR} \leq C_x^S \quad \forall v, x \tag{8}$$

$$\xi_{v,s,x}^{VI} \cdot c_{v,s}^{VI} + \xi_{v,d,x}^{VI} \cdot c_{v,d}^{VI} \leq C_x^S \quad \forall v, x \tag{9}$$

F. CONSTRAINTS OF LINK EMBEDDING

The physical path on which a virtual link is embedded should satisfy the flow conservation and the flow split is not allowed [20]. Thus, each virtual link is embedded onto only one physical path. The corresponding constraint is as follows

$$\sum_{x=1}^{|N^S|} \zeta_{v,(i,j),(x,y)}^{VR} - \sum_{x=1}^{|N^S|} \zeta_{v,(i,j),(y,x)}^{VR} = \begin{cases} -1 & \text{if} y = i \\ 1 & \text{if} y = j \\ 0 & \text{otherwise} \end{cases} \quad \forall y, i, j \tag{10}$$

For each physical link $l_{(x,y)}^S$, it should have the residual bandwidth capacity enough to accommodate the bandwidth demand of each virtual link that is embedded onto $l_{(x,y)}^S$, as formulated as follows

$$\sum_{i=1}^{|N_v^{VR}|} \sum_{j=1}^{|N_v^{VR}|} \zeta_{v,(i,j),(x,y)}^{VR} \cdot b_{v,(i,j)}^{VR} \leq B_{(x,y)}^S \quad \forall v, x, y \tag{11}$$

If the current VN request comes from IoT service, the residual bandwidth capacity together with the idle bandwidth capacity of the VNs of regular service can be used to afford the bandwidth demands of the virtual links of the current VN request. Thus, we can represent the constraint of link bandwidth capacity as follows.

$$\zeta_{v,(s,d),(x,y)}^{VI} \cdot b_{v,(s,d)}^{VI} \leq B_{(x,y)}^S + \sum_{m=1}^{v-1} \sum_{i=1}^{|N_v^{VR}|} \sum_{j=1}^{|N_v^{VR}|} \zeta_{m,(i,j),(x,y)}^{VR} w_{m,(i,j)}^{VR} \quad \forall v, x, y \tag{12}$$

G. INP REVENUE MODEL

The InP obtains revenue by leasing the physical network resource to the SPs who put forward the VN requests of regular service and IoT service. A larger amount of revenue arises when the physical network accepts more VN requests by using less network resource. Generally, we can define the InP revenue as:

$$Revenue = \sum_v R(G_v^{VR}) + \sum_v R(G_v^{VI}) \tag{13}$$

where $R(G_v^{VR})$ refers to the revenue arising from the vth VN request of regular service and $R(G_v^{VI})$ refers to the revenue arising from the vth VN request of IoT service, respectively. We can calculate the revenue by subtracting the cost from the income for each VN request. Therefore, $R(G_v^{VR})$ can be further represented as

$$\begin{aligned} R(G_v^{VR}) = T_v^{VR} \cdot \left[\beta_C \sum_{i=1}^{|N_v^{VR}|} c_{v,i}^{VR} + \beta_B \sum_{i=1}^{|N_v^{VR}|} \sum_{j=1}^{|N_v^{VR}|} b_{v,(i,j)}^{VR} \right] \\ - T_v^{VR} \cdot \left[\alpha_C \sum_{i=1}^{|N_v^{VR}|} c_{v,i}^{VR} + \alpha_B \sum_{i=1}^{|N_v^{VR}|} \sum_{j=1}^{|N_v^{VR}|} \sum_{x=1}^{|N^S|} \sum_{y=1}^{|N^S|} \right. \\ \left. \times \zeta_{v,(i,j),(x,y)}^{VR} b_{v,(i,j)}^{VR} \right] \end{aligned} \tag{14}$$

where β_C, β_B, α_C and α_B represent the income per unit of CPU resource and the income per unit of bandwidth resource, the cost per unit of CPU resource and the cost per unit of bandwidth resource, respectively. These four weight factors should make sure $\beta_C > \alpha_C$ and $\beta_B > \alpha_B$ such that the income that InP acquire from one unit of CPU or bandwidth resource is higher than the cost and thus the positive revenue. Therefore, in Eq. (14) the first item and the second item are the total income and the total cost from the vth VN request of regular service, respectively.

Similarly, we can formulate the revenue $R(G_v^{VI})$ from the vth VN request of IoT service as follows.

$$R(G_v^{VI}) = T_v^{VI} \cdot \left[\beta_C \left(c_{v,s}^{VI} + c_{v,d}^{VI}\right) + \beta_B \cdot b_{v,(s,d)}^{VI}\right] - T_v^{VI} \cdot \left[\alpha_C \left(c_{v,s}^{VI} + c_{v,d}^{VI}\right) + \gamma_B \cdot \alpha_B \sum_{x=1}^{|N^S|} \sum_{y=1}^{|N^S|} \times \zeta_{v,(s,d),(x,y)}^{VI} b_{v,(s,d)}^{VI}\right] \tag{15}$$

where the first item and the second item are the total income and the total cost from the vth VN request of IoT service, respectively. Unlike the bandwidth resource allocation for the virtual link of regular service, the virtual link of IoT service can share the idle bandwidth resource that have been allocated to the VNs of regular service but remain idle. Therefore, we introduce a penalty factor γ_B into the item of resource cost to indicate the status of the bandwidth resource allocated to the VN of IoT service. For any VN of IoT service, the value of γ_B is decided depending on the following three cases: 1) $\gamma_B = 1$ if the VN of IoT service is allocated the dedicated bandwidth resource, 2) $\gamma_B = 0$ if the VN of IoT service is allocated the idle bandwidth resource from one VN of regular service and the idle bandwidth resource stays available for the VN of IoT service over its whole service duration, and 3) $\gamma_B \in (0, 1)$ if the VN of IoT service is allocated the idle bandwidth resource, while it has to migrate its own virtual link and give the corresponding bandwidth resource back to the VN of regular service because the traffic load of the VN of regular service grows. In the last case, the VN of IoT service has to suffer from the service degradation even interruption during the virtual link migration, thus it is necessary to introduce a revenue reduction according to the actual service time on the idle bandwidth resource. A larger value of γ_B indicates a smaller revenue reduction.

III. HEURISTIC ALGORITHMS

A. OVERVIEW OF THE VNE ALGORITHM

We introduce the following symbols to facilitate the algorithmic description.

- Γ_t : The set of physical links on which the idle bandwidth resource of the VNs of regular service is no longer enough to afford the requested bandwidth resource of the existing VNs of IoT service.
- $\Psi_{(x,y),t}^{VI,ID}$: The set of VNs of IoT service that have been using the idle bandwidth resource of the VNs of regular service on the physical link in the tth time window.
- $W_{(x,y),t}^{VR,ID}$: The bandwidth resource of the VNs of regular services on $l_{(x,y)}^S$ that remains idle due to low load in the tth time window.
- $W_{(x,y),t}^{VI}$: The idle bandwidth resource of the VNs of regular service on $l_{(x,y)}^S$ that has been allocated to the VNs of IoT service in the tth time window.
- $W_{v,(x,y),t}^{VI}$: The idle bandwidth resource of the VNs of regular service on $l_{(x,y)}^S$ that has been allocated to the vth VN of IoT service in the tth time window.
- $f_{(x,y),t}^S$: the load of $l_{(x,y)}^S$ in the tth time window.

The VNE algorithm is usually operated in the VN embedding server. The VN requests are processed periodically on basis of time window. In each time window, e.g., the t^{th} time window, InP collects the VN requests of both regular service and IoT service that have arrived in current time window, where the set of VN requests of regular service and the set of VN requests of IoT service are represented by Ψ_t^{VR} and Ψ_t^{VI}, respectively. As shown in the **Algorithm 1**, in order to increase the revenue from embedding VNs, InP calculates the income for each VN request in Ψ_t^{VR} and Ψ_t^{VI} and sorts these VN requests in the order of decreasing income. The traffic load $f_{(x,y),t}^S$ on each physical link in the t^{th} time window will be recorded and used to update the idle bandwidth resource $W_{(x,y),t}^{VR,ID}$ of the VNs of regular service. When $W_{(x,y),t}^{VR,ID}$ is no longer enough due to the increase of traffic load, the resource contention occurs between the VNs of regular service and the VNs of IoT service. In this case, the VNs of IoT service will have to migrate to other physical links and return the bandwidth resource to the VNs of regular service. In order to find out the set of physical links Γ_t with the VNs of IoT service migrated, InP goes through all physical links whose traffic load increases in the t^{th} time window. For each physical link $l_{(x,y)}^S \in \Gamma_t$, the VNs of IoT service $\Psi_{(x,y),t}^{VI,ID}$ that have been using the idle bandwidth of the VNs of regular service on $l_{(x,y)}^S$ will be sorted in the order of increasing revenue. The VNs of IoT service that brings lower revenue for InP will be migrated prior to the ones with higher revenue, such that the penalty incurred by VN migration is minimized. The VNs of IoT service in $\Psi_{(x,y),t}^{VI,ID}$ will be released from the idle bandwidth of the VNs of regular service one by one until the idle bandwidth of the VNs of regular service is enough to afford the resource requests of the VNs of IoT service. The released VNs of IoT service are migrated and re-embedded using the **Algorithm 3** onto the physical network based on the updated traffic load in current time window. After the re-embedding of all the released VN requests, the VN requests of IoT service and the VN requests of regular service that have arrived in current time window will be embedded using the **Algorithm 2** and **Algorithm 3**, respectively.

B. VIRTUAL NETWORK EMBEDDING OF REGULAR SERVICES

The embedding of the VNs requests of regular service consists of two stages, first the embedding of virtual nodes including access virtual node and forward virtual nodes, and

Algorithm 1 VNE Algorithm for Hybrid Services

Input: G^S, C_x^S, $f_x^S lct_x^S$ and $B_{(x,y)}^S$.
Output: $\xi_{v,i,x}^{VR}$, $\zeta_{v,(i,j),(x,y)}^{VR}$, $\xi_{v,s,x}^{VI}$, $\xi_{v,d,x}^{VI}$ and $\zeta_{v,(s,d),(x,y)}^{VI}$.
1: Collect all VN requests of regular service and IoT service that have arrived in current time window: Ψ_t^{VR} and Ψ_t^{VI};
2: Calculate the income for each VN request in Ψ_t^{VR} and Ψ_t^{VI} and sort the VN requests in the order of decreasing income;
3: Record the traffic load $f_{(x,y),t}^S$ observed in current time window for each physical link, and update: $W_{(x,y),t}^{VR,ID} \leftarrow W_{(x,y),t-1}^{VR,ID} - \left(f_{(x,y),t}^S - f_{(x,y),t-1}^S\right)$;
4: Go through all the physical links with $f_{(x,y),t}^S > f_{(x,y),t1}^S$ and find out the set of physical links $\Gamma_t = \{l_{(x,y)}^S \,|\, l_{(x,y)}^S \in L^S,\ W_{(x,y),t}^{VI} > W_{(x,y),t}^{VR,ID}\}$;
5: for each physical link $l_{(x,y)}^S \in \Gamma_t$, **then**
6: Sort the VNs of IoT service in $\Psi_{(x,y),t}^{VI,ID}$ in the order of increasing revenue;
7: for the vth VN of IoT service in $\Psi_{(x,y),t}^{VI,ID}$, **then**
8: Release the bandwidth resource of the vth VN of IoT service: $W_{(x,y),t}^{VI} \leftarrow W_{(x,y),t}^{VI} - W_{v,(x,y),t}^{VI}$;
9: if $W_{(x,y),t}^{VI} \leq W_{(x,y),t}^{VR,ID}$, **then break**;
end if
10: end for
11: end for
12: Re-embed the released VNs of IoT service from $\Psi_{(x,y),t}^{VI,ID}$ using the Algorithm 3;
13: Embed the VN requests of regular service in Ψ_t^{VR} using the Algorithm 2;
14: Embed the VN requests of IoT service in Ψ_t^{VI} using the Algorithm 3.

then the embedding of virtual links. For each VN request of regular service in Ψ_t^{VR}, e.g., the vth VN request, its access virtual node is first embedded to OLT and allocated the requested CPU resource. Then, the forward virtual nodes are ranked in the order of increasing CPU resource request. The forward virtual node with lower CPU resource request will be more easily accepted, especially in the resource-hungry situation. We observed from our previous works [13], [14] that, the selection of physical nodes not only determines the embedding of virtual nodes but also impacts the embedding of the virtual links associated with the corresponding virtual nodes. The impact shows much more obviously when the bandwidth resource is distributed more unfairly among physical links, which is a typical feature in FiWi access network where optical links and wireless links are allocated bandwidth on basis of time division multiplexing. In the VNE algorithm for regular service, we define a weighted degree for each physical node as the number of associated physical links multiplied by the residual CPU capacity to indicate the

Algorithm 2 VNE Algorithm for Regular Service

Input: G^S, C_x^S, f_x^S, lct_x^S and $B_{(x,y)}^S$.
Output: $\xi_{v,i,x}^{VR}$ and $\zeta_{v,(i,j),(x,y)}^{VR}$.
1: for the vth VN request of regular service, **then**
2: Embed the access virtual node $n_{v,i}^{VR}(type_{v,i}^{VR} = A)$ to OLT and update the residual CPU resource of OLT;
3: Rank the forward virtual nodes $\left\{n_{v,i}^{VR} \middle| type_{v,i}^{VR} = T\right\}$ in the order of increasing CPU resource request;
4: Calculate the weighted degree for each physical node and rank them in the order of decreasing weighted degree;
5: for each forward virtual node $n_{v,i}^{VR}$ and each physical node n_x^S, **then**
6: Embed $n_{v,i}^{VR}$ to the physical node n_x^S which first fits the CPU resource request and location constraint;
7: if none of the physical nodes is eligible to carry $n_{v,i}^{VR}$, **then** $\sigma_{RJ} \leftarrow 1$;
8: else $\sigma_{RJ} \leftarrow 0$, and $\xi_{v,i,x}^{VR} \leftarrow 1$;
9: Update the residual CPU resource: $C_x^S \leftarrow C_x^S - c_{v,i}^{VR}$;
10: end if
11: end for
12: for each virtual link $l_{v,(i,j)}^{VR}$, **then**
13: Embed $l_{v,(i,j)}^{VR}$ to the physical links $\left\{l_{(x,y)}^S\right\}$ with the bandwidth resource enough to afford $b_{v,(i,j)}^{VR}$ for the fewest hops;
14: if no physical path is eligible to carry $n_{v,i}^{VR}$, **then** $\sigma_{RJ} \leftarrow 1$;
15: else $\sigma_{RJ} \leftarrow 0$ and $\zeta_{v,(i,j),(x,y)}^{VR} \leftarrow 1$;
16: Update the residual bandwidth resource on the physical links: $B_{(x,y)}^S \leftarrow B_{(x,y)}^S - b_{v,(i,j)}^{VR}$, $\forall x, y : \zeta_{v,(i,j),(x,y)}^{VR} = 1$;
17: end if
18: end for
19: if $\sigma_{RJ} = 1$, **then** reject the vth VN request;
20: else accept the vth VN request;
21: end if
22: end for

potential impact on the subsequent embedding of the virtual links if a forward virtual node is embedded on this physical node.

According to the weighted degree of physical node, the physical node with a larger weighted degree provides higher probability for the successful embedding of the associated virtual links. The physical nodes are ranked in the order of decreasing weighted degree. To embed a forward virtual node, e.g., $n_{v,i}^{VR}$, the ranked physical nodes are examined in sequence and the one n_x^S which first fits the CPU resource request and location constraint is selected to carry $n_{v,i}^{VR}$. As a

Algorithm 3 VNE Algorithm for IoT Service

Input: G^S, C_x^S, f_x^S, lct_x^S and $B_{(x,y)}^S$.

Output: $\xi_{v,s,x}^{VI}$, $\xi_{v,d,x}^{VI}$ and $\zeta_{v,(s,d),(x,y)}^{VI}$.

1: for each physical link $l_{(x,y)}^S$, **then**

2: Calculate the migration ratio of the VNs of IoT service on $l_{(x,y)}^S$ that occurred in the tth time window;

3: Calculate the reward/penalty value according to Eq.(16) for $l_{(x,y)}^S$;

4: Update the Q-table according to Eq.(17);

5: Take the action that makes the maximum Q-value for the current state and generate a new state;

6: Resolve the state and obtain the idle bandwidth resource $W_{(x,y),t+1}^{VR,ID}$ expected in the next time window;

7: end for

8: Rank the physical nodes $n_x^S \in N^S$ in the order of decreasing weighted degree;

9: for the vth VN request of regular service, **then**

10: Embed the virtual node pair $n_{v,s}^{VI}$ and $n_{v,d}^{VI}$ to the physical node pair n_x^S and n_y^S which first fit their CPU resource requests and location constraints;

11: **if** none of the physical nodes is eligible to carry $n_{v,s}^{VI}$ or $n_{v,d}^{VI}$, **then** $\sigma_v^{VI} \leftarrow 1$;

12: **else** $\sigma_v^{VI} \leftarrow 0$, $\xi_{v,s,x}^{VI} \leftarrow 1$ and $\xi_{v,d,y}^{VI} \leftarrow 1$;

13: Update the residual CPU resource of n_x^S and n_y^S: $C_x^S \leftarrow C_x^S - c_{v,s}^{VI}$ and $C_y^S \leftarrow C_y^S - c_{v,d}^{VI}$;

14: **end if**

15: Embed $l_{v,(s,d)}^{VI}$ to the physical links $\left\{l_{(x,y)}^S\right\}$ with the bandwidth resource enough to afford $b_{v,(s,d)}^{VI}$ for the minimum occupied bandwidth resource;

16: **if** no physical path is eligible to carry $l_{v,(s,d)}^{VI}$, **then** $\sigma_v^{VI} \leftarrow 1$;

17: **else** $\sigma_v^{VI} \leftarrow 0$, $\zeta_{v,(s,d),(x,y)}^{VI} \leftarrow 1$;

18: Update the residual bandwidth resource on each link in $\left\{l_{(x,y)}^S\right\}$: $B_{(x,y)}^S \leftarrow B_{(x,y)}^S - b_{v,(s,d)}^{VI} + W_{(x,y),t+1}^{VR,ID} \forall x, y:$ $\zeta_{v,(s,d),(x,y)}^{VR} = 1$;

19: **end if**

20: **if** $\sigma_v^{VI} = 1$, **then**reject the vth VN request;

21: **else** accept the vth VN request;

22: **end if**

23: end for

result, the residual CPU resource of n_x^S is updated. If none of the physical nodes is eligible to carry $n_{v,i}^{VR}$, the vth VN request will be rejected.

After all virtual nodes of the vth VN requests are embedded, the algorithm goes into the stage of virtual link embedding. For each virtual link $l_{v,(i,j)}^{VR}$, the shortest routing algorithm is used to calculate a target physical path between the physical nodes where $n_{v,i}^{VR}$ and $n_{v,j}^{VR}$ are embedded with the fewest hops. Because each link on the target physical path needs to allocate the requested bandwidth resource to $l_{v,(i,j)}^{VR}$, the fewest hops help reduce the occupation of bandwidth resource and thus raise the revenue of InP. Upon the successful embedding of $l_{v,(i,j)}^{VR}$, the physical links on the target physical link will be updated in terms of residual bandwidth resource. If there is no any physical path with enough bandwidth resource to carry $l_{v,(i,j)}^{VR}$, the vth VN request will be rejected.

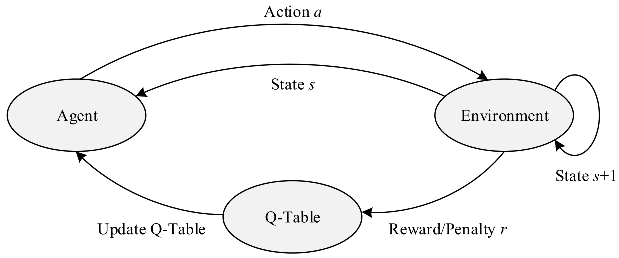


FIGURE 2. Working flow of Q-learning.

C. TRAFFIC LOAD PREDICTION WITH Q-LEARNING

Reinforcement learning is a branch of machine learning, which aims to enable an agent to adjust the decision-making on its action by constantly perceiving the environmental change and learning from the historical dataset. Since the excellent compromise between effectiveness and complexity, Q-learning distinguishes itself from others as one of most studied algorithms in the family of reinforcement learning [21].

A typical working flow of Q-learning is illustrated in Fig. 2 as a circular system. There are three models included, *agent*, *environment* and *Q-table*. The *environment* is characterized by a set of serial states. The change occurs to the environment when it is transferred from one state to another. As a reaction to the new state, the *agent* perceives the changes in environment and learns from Q-table and then makes the decision on the next action of environment. The action has an impact more or less on the state change of *environment*. The Q-table records the reward and penalty to evaluate the effectiveness of the agent's decision on action, which in turn acts on the decision-making of agent on next action. Hence, the definition of the reward and penalty functions dominates the decision of agent on action significantly.

Since InP has an overview of the resource information of the whole physical network, Q-learning is a viable way to predict the dynamics of traffic load on each physical link. Based on the working flow aforementioned, we can define the models for the Q-earning algorithm to predict the traffic load in FiWi access network as follows:

1) ENVIRONMENT

The environment here refers to the idle bandwidth resource of physical links over the whole network. We define a set of states $\{s^p_{(x,y)}\}$ to quantify the amount of idle bandwidth resource on $l^S_{(x,y)}$ observed in any time window into P levels, where $p = 1, 2, 3, \cdots, P$. Thus, the transition among different states relies on the dynamics of the traffic load on the virtual links of the VNs of regular service.

2) AGENT

InP plays the role of *agent* who acts to predict the transition of the state of environment and decide which of the physical links will have the appropriate idle bandwidth resource from the VNs of regular service to carry the virtual links of the VNs of IoT service. Thus, the action that the agent needs to take refers to predicting the state transition and allocating resource for the virtual links of the VNs of IoT service. We define a set of actions to represent the change of idle bandwidth resource on each physical link, including to increase, to decrease and stay unchanged.

3) Q-TABLE

Because of the unavoidable prediction inaccuracy, it is possible that the traffic load of regular service grows out of expectation and the idle bandwidth resource of the VNs of regular service on a physical link is no longer enough to meet the requests of the VNs of IoT service. To avoid the resource contention, the VNs of IoT service have to migrate its virtual links such that the bandwidth resource can be released back to the VNs of regular service. During the virtual link migration, the service performance of the VNs of IoT service will be degraded and more severely when migration occurs more frequently and the scale of affected traffic is larger. A reward and penalty function is defined to reflect the impact of the prediction accuracy on the service performance, as follows

$$r^p_{(x,y)} = -\tan(p_M - p_0)\ 0 \le p_M \le 1 \tag{16}$$

where $r^p_{(x,y)}$ is the reward/penalty value for $l^S_{(x,y)}$, and p_M is the migration ratio of the VNs of IoT service, which is defined as the ratio of the migrated VNs to the total VNs of IoT service. p_0 is the migration ratio boundary for the reward area (i.e., $r^p_{(x,y)} \ge 0$) and penalty area (i.e., $r^p_{(x,y)} < 0$). When $p_0 = 0.5$, the reward has a same area as the penalty.

A Q-table is maintained to record the Q-value for each physical link. In the Q-table for $l^S_{(x,y)}$, the element at the pth row and the qth column represents the Q-value for the state $s^p_{(x,y)}$ when the action $a^q_{(x,y)}$ occurs on $s^p_{(x,y)}$. For example, a Q-table with three states and three actions is shown in Table 1. The Q-value is updated in each time window once the idle bandwidth resource transits from one state to another by taking an action. Assuming the idle bandwidth resource of $l^S_{(x,y)}$ is in the state $s^p_{(x,y)}$ currently, it transits to the state $s^l_{(x,y)}$ when an action $a^q_{(x,y)}$ is taken. Accordingly, the Q-value of the element at the pth row and the qth column in the Q-table for $l^S_{(x,y)}$ should be updated as

$$Q^{t+1}_{(x,y),p,q} = (1-\alpha)\cdot Q^t_{(x,y),p,q} + \alpha\left[r^p_{(x,y)} + \gamma \max\left\{Q^t_{(x,y),p,q}\right\}\right] \tag{17}$$

TABLE 1. Example of Q-table.

	a_1	a_2	a_3
s_1	5	0	2
s_2	-2	2	1
s_3	0	1	-3

where $Q^t_{(x,y),p,q}$ is the Q-value of the element at the pth row and qth column in the tth time window. α denotes the learning rate, where $0 < \alpha < 1$. The larger α indicates the agent learns more from the historical experience. γ is the discount factor to control the proportion of $r^p_{(x,y)}$ and $\max\left\{Q^t_{(x,y),p,q}\right\}$.

According to Eq. (17), we can use the Q-value $Q^{t+1}_{(x,y),p,q}$ to estimate the probability that the action $a^q_{(x,y)}$ occurs when the idle bandwidth resource of $l^S_{(x,y)}$ is in the state $s^p_{(x,y)}$. The action with the highest Q-value will be chosen for the state $s^p_{(x,y)}$ and as a result the next state is derived. As the example in Table, if the current state is s_1, the action a_1 will be taken because $Q^t_{(x,y),1,1}$ is larger than $Q^t_{(x,y),1,2}$ and $Q^t_{(x,y),1,3}$. Thereafter, $Q^t_{(x,y),1,1}$ will be updated with $Q^{t+1}_{(x,y),1,1}$ as in Eq. (17).

With the new state derived, we can predict the idle bandwidth resource on each physical link, based on which the VN requests of IoT service that have arrived in the $(t+1)$th time window will be embedded.

D. VIRTUAL NETWORK EMBEDDING OF IOT SERVICE

The detailed procedure of the VNE algorithm for IoT service is described in **Algorithm 3**. For each physical link $l^S_{(x,y)}$, InP first calculates the migration ratio of the VNs of IoT service on $l^S_{(x,y)}$ that occurred in current time window and the reward/penalty value according to Eq.(16). Then, given the current state of the idle bandwidth resource on $l^S_{(x,y)}$, the Q-table can be updated according to Eq.(17). The action that can make the maximum Q-value is taken on the basis of current state to generate a new state, from which the idle bandwidth resource expected in the next time window is resolved out.

As done in the VNE algorithm for regular service, the embedding of the VN requests of IoT service is also divided into the stage of virtual node embedding and the stage of virtual link embedding. For the embedding of the vth VN request of IoT service, all of the physical nodes are first ranked in the order of decreasing weighted degree. The virtual node pair $n^{VI}_{v,s}$ and $n^{VI}_{v,d}$, are embedded to the physical node pair n^S_x and n^S_y which first fit their CPU resource requests and location constraints. If none of the physical node is eligible to carry $n^{VI}_{v,s}$ or $n^{VI}_{v,d}$, the vth VN request of IoT service will be rejected. Then, a physical path that occupies the minimum bandwidth resource excluding the idle bandwidth resource from the VNs of IoT service is calculated between the nodes n^S_x and n^S_y to carry the virtual link $l^{VI}_{v,(s,d)}$. Each link on this physical path should have enough bandwidth resource (including both residual bandwidth resource and idle bandwidth resource) to meet the request of the vth VN request of IoT service. If there is no such a physical path with enough bandwidth resource to carry $l_{v,(s,d)}^{VI}$, the vth VN request of IoT service will be rejected. To save the occupancy of bandwidth resource, the idle bandwidth resource is always allocated to the VN requests of IoT service prior to the residual bandwidth resource.

IV. PERFORMANCE EVALUATION

A. SETTINGS

Settings of physical network: The physical network consists of 1 OLT, 4 ONUs and 49 wireless routers in a square area of 500m $*$ 500m. Each of the ONUs drives two gateways via cable [22], [23]. Each wireless router is equipped with 4 radio interfaces and both transmission range and interference range are set to 100 m. The CPU capacity and bandwidth capacity for different types of physical nodes and links are set as in Table 2.

TABLE 2. Resource settings of physical network [9]–[11].

Parameter	Value
OLT CPU capacity	(500, 1000)
ONU CPU capacity	(200, 500)
CPU capacity of wireless router	(50, 100)
Fiber bandwidth capacity	1000 Mbps
Cable bandwidth capacity	100 Mbps
Wireless link bandwidth capacity	54 Mbps

Settings of virtual network: The time is measured on basis of number of time window. Both the VN requests of regular service and IoT service arrive into network according to Poisson distribution with the average arrival rate of 2 per time window [24], [25]. The average durations of the VN requests of regular service and IoT service follow the exponential distribution with the average of 20 time windows and 5 time windows, respectively. The number of virtual nodes in each VN request of regular service takes in 2 to 4 randomly, while the number of virtual nodes in each VN request of IoT service is fixed to 2. Each pair of virtual nodes in a VN of regular service are connected by virtual link with the probability of 0.5. The requests of virtual node and virtual link for regular service and IoT service are shown in Table 3 [22], [26]–[29]. The weights for InP's revenue $\beta_C = \beta_B = 5$ and $\alpha_C = \alpha_B = 1$.

Settings of benchmarks: We set the following two algorithms in order to have a comprehensive evaluation of each component of the proposed algorithm, 1) VNE-EACH, in which resource sharing is not allowed between the VNs of regular service and the VNs of IoT service, and 2) VNE-USE, in which the VNs of IoT service is allowed to use the idle bandwidth resource from the VNs of regular service, but without the traffic load prediction. For convenience of comparison, we use VNE-USE-Q to represent the VNE algorithm (i.e., Algorithm 1) proposed in this paper.

TABLE 3. Requests settings of virtual node and virtual link.

Requests	Regular service	IoT service
CPU of virtual node	[10, 20]	[5, 10]
Bandwidth of virtual link	[4,9] Mbps	[1,3] Mbps
Virtual node location offset	100 m	100 m
Path length of virtual link	6 hops	6 hops

B. RESULT ANALYSIS

Figure 3 shows the comparison of the VNE-USE algorithm and the VNE-USE-Q algorithm in the migration ratio of IoT service VNs. As aforementioned, the VNs of IoT service which use the idle bandwidth resource from the VNs of regular service will have to migrate its virtual link and release the bandwidth resource back to the VNs of regular service when the VNs of regular service needs the idle bandwidth resource to carry higher load. In the VNE-USE algorithm, the idle bandwidth resource of the VNs of regular service is allocated to the VNs of IoT service just based on current traffic load and without the consideration of the following change in idle bandwidth resource. Thus, we can observe a 10.8% to 23.2% higher migration ratio in the VNE-USE algorithm than the proposed VNE-USE-Q algorithm. In the VNE-USE-Q algorithm, the Q-learning based resource prediction mechanism can provide a forward-looking resource allocation which helps mitigate the resource contention. With the time going, a richer experience about the state transition of idle bandwidth resource will be collected from which the Q-learning can make a more accurate prediction. Therefore, the migration ratio of IoT service VNs shows a gradually decreasing tendency over time. It is proved that the proposed VNE-USE-Q algorithm is more effective in holding down the performance degradation of IoT service when bandwidth resource is shared between IoT service and regular service.

Figure 4 and Figure 5 display the acceptance ratio of total VNs and the acceptance ratio of IoT service VNs, respectively. Since the VN requests are generated randomly in terms of topology and resource request, both the acceptance ratios

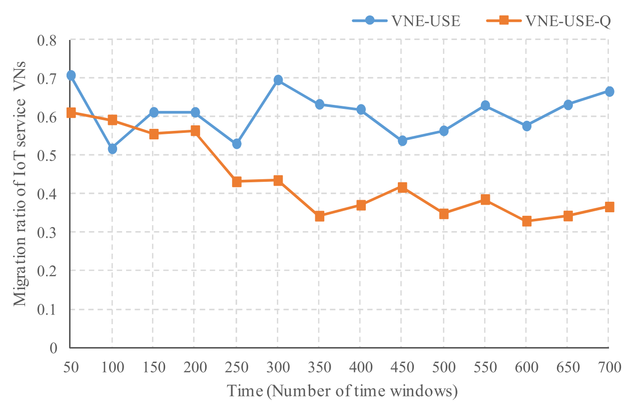


FIGURE 3. Migration ratio of IoT service VNs over time.

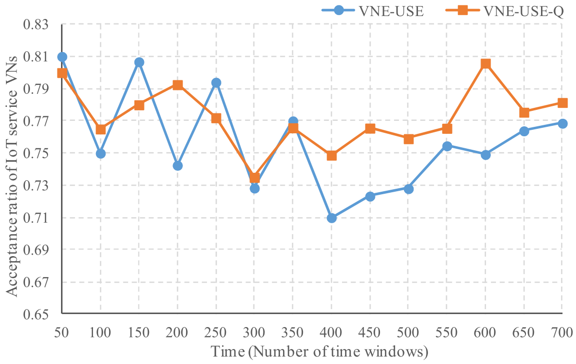


FIGURE 4. Acceptance ratio of IoT service VNs over time.

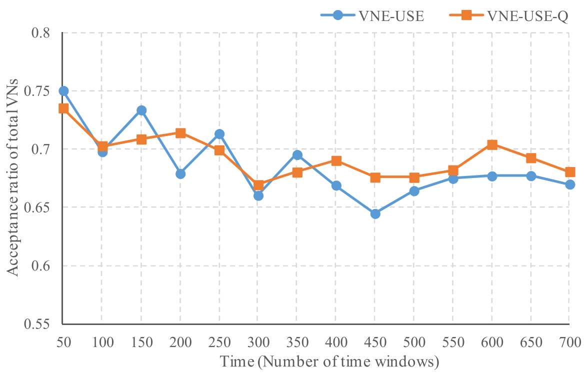


FIGURE 5. Acceptance ratio of total VNs over time.

undergo the irregular fluctuation and the fluctuation is more intensive when the number of time windows is low. When the time goes over around 400 time windows, the acceptance ratios in both Figs. 4 and 5 begin to go into a comparatively steady state. We observe that the outperformance of the VNE-USE-Q algorithm over the VNE-USE algorithm appears more clearly. This indicates that the traffic load prediction mechanism in the VNE-USE-Q algorithm has entered into a stable learning process. For this observation, we choose the Q-table at the 400th time window as the initial Q-table for the VNE-USE-Q algorithm in the following results.

The comparison of total InP's revenue among the VNE-EACH, VNE-USE and VNE-USE-Q algorithms is made in Fig. 6. The VN requests of regular service have the same arrival rate as the VN requests of IoT service, ranging from 0.5 to 6. With the VN arrival rate increasing, all three algorithms display the gradually rising InP revenue. In the VNE-EACH algorithm, the VNs of regular service and the VNs of IoT service are allocated the network resource separately and no any resource is shared between them. Thus, we can regard the InP revenue of the VNE-EACH algorithm as a performance benchmark for the two-stage embedding proposed in this paper. Owing to the effective reuse of idle bandwidth resource, the VNE-USE and VNE-USE-Q algorithms enable InP to accept more VN requests and thus higher revenue. Particularly, the VNE-USE-Q algorithm can reduce the migration of the VNs of IoT service that is incurred by

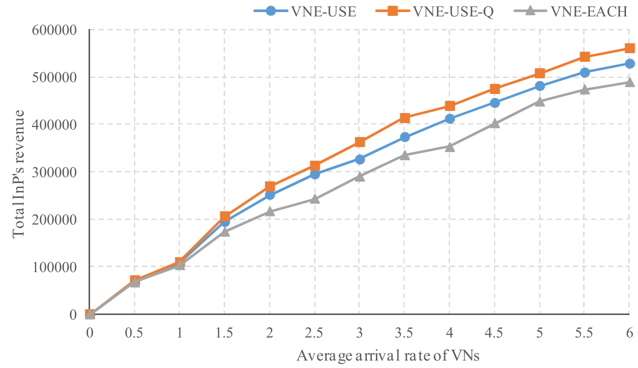


FIGURE 6. Total InP's revenue with different VN arrival rates.

bandwidth resource contention by predicting the change of idle bandwidth resource. Compared to the VNE-USE algorithm, the VNE-USE-Q algorithm suffers from much lower penalty from IoT service VNs migration and thus provides InP with higher revenue. For example, when the average VN arrival rate is 4, the VNE-USE-Q algorithm and the VNE-USE algorithm can bring InP with 17.1% and 22.8% more revenue, respectively.

We can further explain the composition of InP's revenue from regular service and IoT service in Fig. 7 and Fig. 8, respectively. We observe that, compared to the VNE-EACH algorithm, the VNE-USE and VNE-USE-Q algorithms bring InP with not only more revenue from the IoT service but also the regular service. This is because the reuse of idle bandwidth resource in both algorithms helps InP reduce the extra occupancy of bandwidth resource for carrying the VNs of IoT service, more residual bandwidth resource is available to accept more VNs of regular service. In the VNE-USE algorithm, the additional revenue from IoT service just comes from the saved bandwidth resource occupancy, while the VNE-USE-Q algorithm creates this part of additional revenue by both saving bandwidth resource occupancy and reducing the migration penalty. Therefore, by comparing the VNE-USE-Q and VNE-USE algorithms, we can observe a notable gain in the revenue from IoT service over the revenue from regular service. For example, when the average arrival rate of

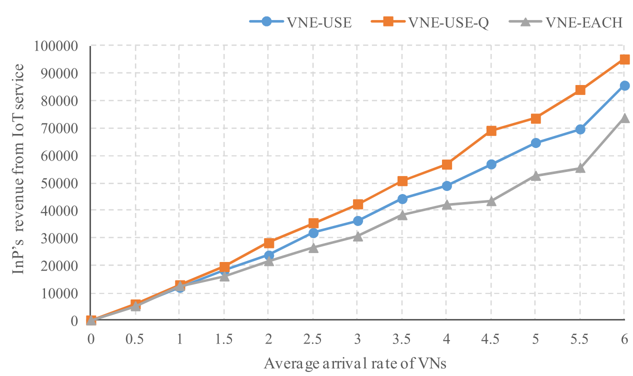


FIGURE 7. InP's revenue from IoT service with different VN arrival rates.

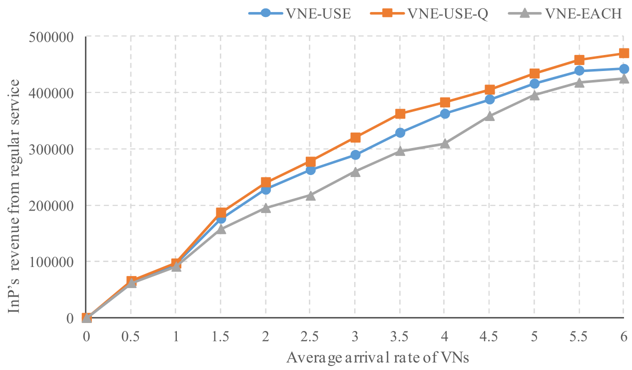


FIGURE 8. InP's revenue from regular service with different VN arrival rates.

VNs varies from 4 to 6, the VNE-USE-Q algorithm outperforms the VNE-USE algorithm with 11.17% more revenue from IoT service, and 5.61% more revenue from regular service.

V. CONCLUSION

In this paper, we addressed the resource allocation issue for the virtual networks in FiWi access network with the coexistence of regular service and IoT service. A Q-learning method was proposed to predict the traffic load of virtual networks of regular service. The idle resource arising from regular service while it is low-loaded is reallocated to bear the virtual networks of IoT service, such that the network resource can be utilized more effectively. The InP revenue models were designed for both regular service and IoT service taking into account the penalty of IoT service VN migration due to the resource contention. The VNE algorithms were proposed to embed the VNs of regular service and IoT service aiming at the maximum InP revenue. Performance results demonstrated that the proposed VNE algorithms are effective in increasing the InP revenue, reducing IoT migration ratio and improving VN acceptation ratio.

REFERENCES

[1] Z. Guan *et al.*, "APPA: An anonymous and privacy preserving data aggregation scheme for fog-enhanced IoT," *J. Netw. Comput. Appl.*, vol. 125, pp. 82–92, Jan. 2019.

[2] L. Guo *et al.*, "A secure mechanism for big data collection in large scale Internet of Vehicle," *IEEE Internet Things J.*, vol. 4, no. 2, pp. 601–610, Feb. 2017.

[3] Z. Ning, X. Wang, J. Rodrigues, and F. Xia, "Joint computation offloading, power allocation, and channel assignment for 5G-enabled traffic management systems," *IEEE Trans. Ind. Informat.*, to be published. doi: 10.1109/TII.2019.2892767.

[4] A. Mostafa and Y. Gadallah, "A statistical priority-based scheduling metric for IOT communications in LTE networks," *IEEE Access*, vol. 5, pp. 8106–8117, 2017.

[5] J. Wu, S. Luo, S. Wang, and H. Wang, "NLES: A novel lifetime extension scheme for safety-critical cyber-physical systems using SDN and NFV," *IEEE Internet Things J.*, to be published. doi: 10.1109/JIOT.2018.2870294.

[6] X. Wang *et al.*, "Optimizing content dissemination for real-time traffic management in large-scale Internet of Vehicle systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1093–1105, Feb. 2019.

[7] J. Wu, M. Dong, K. Ota, J. Li, and Z. Guan, "Big data analysis-based secure cluster management for optimized control plane in software-defined networks," *IEEE Trans. Netw. Service Manag.*, vol. 15, no. 1, pp. 27–38, Mar. 2018.

[8] X. Wang *et al.*, "Privacy-preserving content dissemination for vehicular social networks: Challenges and solutions," *IEEE Commun. Surveys Tuts.*, to be published. doi: 10.1109/COMST.2018.2882064.

[9] Z. Fan, R. Haines, and P. Kulkarni, "M2M communications for E-health and smart grid: An industry and standard perspective," *IEEE Wireless Commun.*, vol. 21, no. 1, pp. 62–69, Feb. 2014.

[10] D. Van, B. Rimal, and S. Andreev, "Machine-to-machine communications over FiWi enhanced LTE networks: A power-saving framework and end-to-end performance," *IEEE/OSA J. Lightw. Technol.*, vol. 34, no. 4, pp. 1062–1071, Feb. 2016.

[11] S. Liao, Y. Ding, J. Dong, X. Wang, and X. Zhang, "125-GHz microwave signal generation employing an integrated pulse shaper," *J. Lightw. Technol.*, vol. 35, no. 13, pp. 2741–2745, Jul. 1, 2017.

[12] Z. Ning, J. Huang, and X. Wang, "Vehicular fog computing: Enabling real-time traffic management for smart cities," *IEEE Wireless Commun.*, vol. 26, no. 1, pp. 87–93, Jun. 2019.

[13] P. Han, L. Guo, and Y. Liu, "A new virtual network embedding framework based on QoS satisfaction and network reconfiguration for fiber-wireless access network," in *Proc. IEEE ICC*, May 2016, pp. 1–7.

[14] P. Han, Y. Liu, and L. Guo, "QoS satisfaction aware and network reconfiguration enabled resource allocation for virtual network embedding in fiber-wireless access network," *Comput. Netw.*, vol. 143, pp. 30–48, Oct. 2018.

[15] Z. Ning, J. Huang, X. Wang, J. Rodrigues, L. Guo, "Mobile edge computing-enabled Internet of vehicles: Toward energy-efficient scheduling," *IEEE Netw.*, doi: 10.1109/MNET.2018.1800309 .

[16] S. Su *et al.*, "Energy-aware virtual network embedding," *IEEE/ACM Trans. Netw.*, vol. 22, no. 5, pp. 1607–1620, Oct. 2014.

[17] S. Liao, Y. Ding, J. Dong, S. Yan, X. Wang, and X. Zhang, "Photonic arbitrary waveform generator based on Taylor synthesis method," *Opt. Exp.*, vol. 24, no. 21, pp. 24390–24400, 2016.

[18] J. Wu, M. Dong, K. Ota, J. Li, and Z. Guan, "FCSS: Fog computing based content-aware filtering for security services in information centric social networks," *IEEE Trans. Emerg. Topics Comput.*, to be published. doi: 10.1109/TETC.2017.2747158.

[19] M. Chowdhury, M. R. Rahman, and R. Boutaba, "ViNEYard: Virtual network embedding algorithms with coordinated node and link mapping," *IEEE/ACM Trans. Netw.*, vol. 20, no. 1, pp. 206–219, Feb. 2012.

[20] W. Hou, Z. Ning, L. Guo, and M. S. Obaidat, "Service degradability supported by forecasting system in optical data center networks," *IEEE Syst. J.*, to be published. doi: 10.1109/JSYST.2018.2821714.

[21] F. Martignon, S. Paris, and I. Filippini, "Efficient and truthful bandwidth allocation in wireless mesh community networks," *IEEE/ACM Trans. Netw.*, vol. 23, no. 1, pp. 161–174, Feb. 2015.

[22] S. Liao *et al.*, "Arbitrary waveform generator and differentiator employing an integrated optical pulse shaper," *Opt. Express*, vol. 23, no. 9, pp. 12161–12173, 2015.

[23] J. Wu, K. Ota, M. Dong, and C. Li, "A hierarchical security framework for defending against sophisticated attacks on wireless sensor networks in smart cities," *IEEE Access*, vol. 4, pp. 416–424, 2016.

[24] Z. Ning, X. Kong, F. Xia, W. Hou, and X. Wang, "Green and sustainable cloud of things: Enabling collaborative edge computing," *IEEE Commun. Mag.*, vol. 57, no. 1, pp. 72–78, Dec. 2018.

[25] Z. Ning, X. Wang, and J. Huang, "Mobile edge computing-enabled 5G vehicular networks: Toward the integration of communication and computing," *IEEE Veh. Technol. Mag.*, vol. 14, no. 1, pp. 54–61, Dec. 2018.

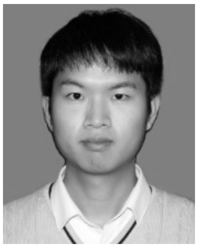
[26] Z. Ning, P. Dong, X. Kong, and F. Xia, "A cooperative partial computation offloading scheme for mobile edge computing enabled Internet of Things," *IEEE Internet Things J.*, to be published. doi: 10.1109/JIOT.2018.2868616.

[27] S. Liao, Y. Ding, C. Peucheret, T. Yang, J. Dong, and X. Zhang, "Integrated programmable photonic filter on the silicon-on-insulator platform," *Opt. Express*, vol. 22, no. 26, pp. 31993–31998, 2014.

[28] Z. Ning, P. Dong, X. Wang, J. Rodrigues, and F. Xia, "Deep reinforcement learning for vehicular edge computing: An intelligent offloading system," *ACM Trans. Intell. Syst. Technol.*, to be published. doi: 10.1145/3317572.

[29] X. Wang *et al.*, "A city-wide real-time traffic management system: Enabling crowdsensing in social Internet of vehicles," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 19–25, Sep. 2018.

YEJUN LIU was an Associate Professor with the School of Computer Science and Engineering, Northeastern University, China. He is currently a Professor with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, China. His research interests include converged fiber-wireless access networks and the Internet of Things (IoT).



YAHE YANG is currently pursuing the M.S. degree with the School of Computer Science and Engineering, Northeastern University, Shenyang, China. His research interests include the industrial Internet of Things (IoT) and network optimization.

PENGCHAO HAN is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, Northeastern University, Shenyang, China. Her research interests include fiber-wireless access networks and network virtualization.

ZEYU SHAO received the M.S. degree from the School of Computer Science and Engineering, Northeastern University, Shenyang, China. Her research interests include fiber-wireless access networks and network virtualization.

CHENGMING LI received the B.S. and M.S. degrees in computer application technology from the Dalian University of Technology, in 2009 and 2011, respectively, and the Ph.D. degree from the Graduate School of Information Science and Electrical Engineering, Kyushu University, in 2015. He is currently an Associate Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Science. His research interests include information-centric networking, software defined networking, the Internet of Things, and big data.