AI-Assisted Resource Advertising and Pricing to Realize Distributed Tenant-Driven Virtual Network Slicing in Inter-DC Optical Networks

Wei Lu, Hongqiang Fang, Zuqing Zhu†
School of Information Science and Technology, University of Science and Technology of China, Hefei, China
†Email: {zqzhu}@ieee.org

***Abstract*—We propose a novel artificial intelligence (AI) assisted framework to realize virtual network (VNT) slicing in an inter-datacenter optical network (IDCON), where the infrastructure provider (InP) performs resource advertising and pricing based on deep reinforcement learning (DRL) and grants the virtual network embedding (VNE) schemes calculated distributedly by the tenants. Simulation results confirm that compared with the traditional centralized VNT slicing framework, our proposal can not only make the InP more profitable but also relieve its computation complexity effectively.**

***Index Terms*—Inter-DC optical networks, Virtual network slicing, Knowledge-defined networking, Artificial intelligence.**

I. Introduction

Recently, the omnipresent requirements of cloud computing are demanding an unprecedented amount of data to be transferred among datacenters (DCs) [1]. Therefore, the architecture of inter-DC optical networks (IDCONs) [2] and the network virtualization schemes in them [3] have received intensive research interests. With network virtualization, service providers (SPs) (*i.e.*, tenants) are allowed to lease substrate network (SNT) resources from an infrastructure provider (InP) and build various virtual networks (VNTs) in a "pay as you use" manner [4, 5]. This is extremely useful in an IDCON, since the InP can allocate bandwidth and IT resources dynamically and adaptively to slice VNTs for the tenants and help them satisfy the time-varying and diversified demands from their services [6]. Hence, a win-win situation can be achieved, *i.e.*, the InP's substrate resource utilization can be improved and the tenants' time-to-market can be reduced.

Note that, for VNT slicing, the InP of an IDCON usually needs to 1) select a substrate DC node to host each virtual node (VN) of the VNT for satisfying the IT resource requirement (*i.e.*, the node mapping), and 2) reserve sufficient optical spectra on a substrate path to carry each virtual link (VL) between a VN pair for satisfying the bandwidth requirement (*i.e.*, the link mapping), which is also known as virtual network embedding (VNE) [7]. Previously, the problem of VNE has already been studied intensively in various network scenarios and with different optimization objectives [7–10], and related network system prototypes have been experimentally demonstrated in [11–13]. However, all these previous investigations assumed that the InP is in charge of VNT slicing solely without any involvement of the tenants, and it calculates

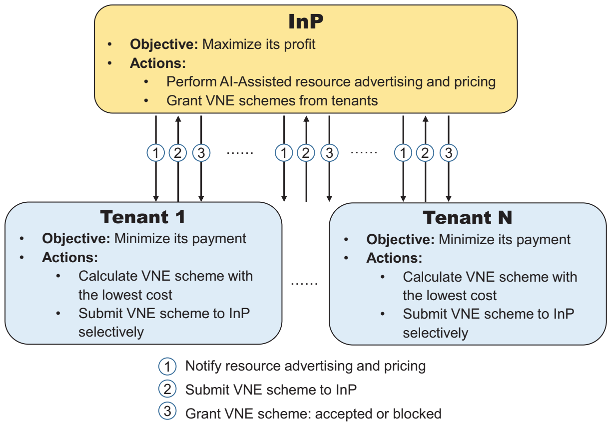


Fig. 1. Proposed framework for distributed tenant-driven VNT slicing.

VNE schemes based on current network status without the intelligence for forecasting. This would not only complicate the network control and management (NC&M) of the InP but also limit the cost-effectiveness of VNT slicing. For instance, it is known that the energy efficiency or the cost-effectiveness of an IDCON can be improved with network-wide resource consolidation, *i.e.*, consolidating computing tasks on fewer DCs and grooming inter-DC traffic to fewer fiber links [14, 15]. Nevertheless, in case of VNT slicing, the InP can hardly realize the most effective resource consolidation, if it cannot directly forecast future VNT requests from the tenants or indirectly affect their behaviors on submitting VNT requests.

The aforementioned issue with existing approaches motivates us to revisit the problem of VNT slicing in IDCONs. Specifically, inspired by the idea of knowledge-defined networking (KDN) [16], we propose to add three new mechanisms into the framework of VNT slicing to make it operate in a distributed tenant-driven manner and more profitable:

- The InP performs resource advertising and pricing to tell the tenants about the DCs and fiber links that can be used to embed their VNTs and the cost of using the corresponding IT and bandwidth resources[1].

[1]Here, the advertised resources might not be all the available ones in the IDCON. For the purpose of resource consolidation, the InP may choose to hide certain resources from advertising.

- Based on the advertisement from the InP, each tenant distributedly calculates the VNE scheme for its VNT with the lowest cost, and determines whether the price is affordable. If yes, it will submit the scheme to the InP.
- The InP collects all the requests from the tenants, grants them based on current network status, calculates the profit from the VNT slicing, and feeds all the information into an artificial intelligence (AI) module based on deep learning to obtain the strategy of the next round of resource advertising and pricing for maximizing its profit.

As shown in Fig. 1, with the new mechanisms, VNT slicing is realized in a distributed and thus much more time-efficient way, and the InP would not be directly involved in the computation of VNE schemes. Hence, the InP's intelligence lies in being able to maximize the profit of VNT slicing by leveraging the AI-assisted resource advising and pricing. In this work, based on the framework in Fig. 1, we first lay out the network model and design an integer linear programming (ILP) model for each tenant to distributedly calculate the VNE scheme for its VNT with the lowest cost. Then, we study how to perform AI-assisted resource advertising and pricing in the InP for profit maximization. Specifically, we design a deep reinforcement learning (DRL) based algorithm to help the InP learn the relation between the strategy of resource advertising and pricing and the profit from VNT slicing. In other words, the DRL-based algorithm enables the InP to analyze the tenants' behaviors on distributed VNE computation for making wise decisions on resource advertising and pricing.

The rest of the paper is organized as follows. We formulate the problem in Section II. The DRL-based resource advertising and pricing algorithm is proposed in Section III. Section IV evaluates the performance of our proposal. Finally, we summarize the paper in Section V.

II. Problem Formulation

A. Network Model of IDCON

We model the topology of an IDCON as $G(V, E)$, where V and E denotes the sets of nodes and fiber links in it, respectively. Note that, there are actually two types of nodes in the IDCON, as shown in Fig. 2(a). Each of the first type ones consists of a local DC and an optical switch (OXC), which is referred to as an edge node and included in set V^E. The second type ones are intermediate nodes, each of which only includes an OXC and is included in set V^I. Apparently, we have $V^E \cap V^I = \emptyset$ and $V^E \cup V^I = V$. In the IDCON, each DC offers IT resources and each fiber link provides bandwidth, for VNT slicing. To facilitate distributed tenant-driven VNT slicing, the InP needs to perform resource advertising and pricing periodically. An example of the resource advertising is illustrated in Fig. 2(b), where for cost saving, the InP only turns on partial of the network elements in the IDCON and advertises the resources on them. Meanwhile, in order to maximize its profit and encourage the tenants to use substrate resources in a balanced manner, the InP needs to price the advertised resources properly. In the next section, we will design a DRL-based algorithm to help the InP achieve this.

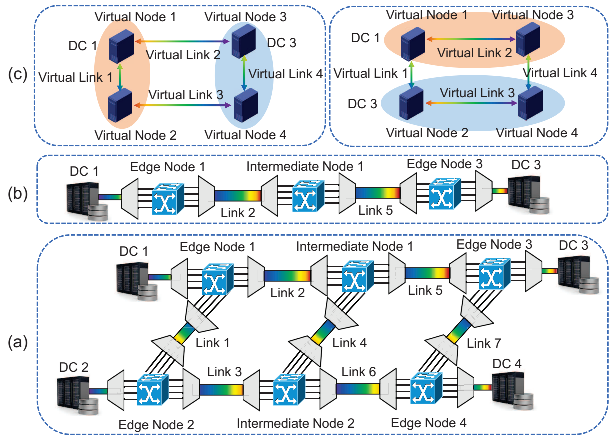


Fig. 2. Example on distributed tenant-driven VNT slicing, (a) IDCON, (b) Resource advertisement from InP, and (c) VNE schemes computed by tenants.

TABLE I
Notations for Resource Cost Model

Notation	Explanation
IDCON:	
v_i^E	the i-th edge node in V^E
v_i^I	the i-th intermediate node in V^I
$E(v)$	the set of links that connect to node $v \in V$
R_i^c	the amount of available IT resources in the DC of v_i^E
R_e^b	the available bandwidth on a fiber link $e \in E$
Edge Node $v_i^E \in V^E$:	
$\dot{C}_{o,i}^E$	the base cost of the OXC in v_i^E if it is working
$C_{o,i}^E$	the unit cost of switching capacity of the OXC in v_i^E
$\dot{C}_{d,i}^E$	the base cost of the DC in v_i^E if it is working
$C_{d,i}^E$	the unit cost of IT resources in the DC in v_i^E
Intermediate Node $v_i^I \in V^I$:	
$\dot{C}_{o,i}^I$	the base cost of the OXC in v_i^I if it is working
$C_{o,i}^I$	the unit cost of switching capacity of the OXC in v_i^I
Fiber Link $e \in E$:	
$\dot{C}_e$	the base cost of e if it is active with traffic
C_e	the unit cost of bandwidth usage on e
$\tilde{C}_e$	the merged unit cost of bandwidth usage on e

To assist the resource advertising and pricing, we define a few notations for the cost model of resources, which are listed in Table I. Here, for each network element in the IDCON (*i.e.*, a DC, an OXC or a fiber link), we assume that the cost of using it consists of a static component (*i.e.*, the base cost of turning it on) and a dynamic component (*i.e.*, the one that increases linearly with the actual resource usage on it). Note that, since the data transmission on fiber link e uses both the link and the two OXCs in its end-nodes, we combine the unit costs of bandwidth usage on them to get the merged unit cost $\tilde{C}_e$ as

$$\tilde{C}_e = C_e + \sum_{\{v_i^E:\ e \in E(v_i^E)\}} C_{o,i}^E + \sum_{\{v_i^I:\ e \in E(v_i^I)\}} C_{o,i}^I. \quad (1)$$

With this cost model, the InP needs to determine its strategy of resource advertising and pricing for profit maximization, and the strategy can be denoted with the variables defined in Table II. Here, for simplicity, we also get the merged unit price of bandwidth usage on fiber link e, which is

$$\tilde{P}_e = P_e + \sum_{\{v_i^E:\ e\in E(v_i^E)\}} P_{o,i}^E + \sum_{\{v_i^I:\ e\in E(v_i^I)\}} P_{o,i}^I. \tag{2}$$

TABLE II
VARIABLES DEFINED FOR RESOURCE ADVERTISING AND PRICING

Variable	Definition
$x_{o,i}^E$	Boolean variable that equals 1 if the OXC in edge node v_i^E is advertised (*i.e.*, turned on) by the InP, and 0 otherwise.
$x_{d,i}^E$	Boolean variable that equals 1 if the DC in edge node v_i^E is advertised by the InP, and 0 otherwise.
$x_{o,i}^I$	Boolean variable that equals 1 if the OXC in intermediate node v_i^I is advertised by the InP, and 0 otherwise.
y_e	Boolean variable that equals 1 if fiber link e is advertised by the InP, and 0 otherwise.
$P_{o,i}^E$	Positive real variable that represents the unit price of switching capacity of the OXC in edge node v_i^E.
$P_{d,i}^E$	Positive real variable that represents the unit price of IT resources in the DC in edge node v_i^E.
$P_{o,i}^I$	Positive real variable that represents the unit price of switching capacity of the OXC in intermediate node v_i^I.
P_e	Positive real variable that represents the unit price of bandwidth usage on fiber link e.
$\tilde{P}_e$	Positive real variable that represents the merged unit price of bandwidth usage on fiber link e.

B. Distributed Tenant-driven VNT Slicing

We assume that there are K pending VNT requests from the tenants. The k-th VNT request can be represented as $G_k^r(V_k^r, E_k^r, \hat{P}_k^r)$, where V_k^r and E_k^r are the sets of virtual nodes (VNs) and virtual links (VLs), respectively, and $\hat{P}_k^r$ is the highest cost that the tenant can afford. Here, each VN $v_{k,i}^r \in V_k^r$ has an IT resource requirement of $R_{k,i}^r$, and it should be mapped onto an edge node in V^E with sufficient IT resources in its DC. Note that, for node mapping, a tenant may have a location constraint from its services, *i.e.*, its VNs should only be mapped onto a subset of edge nodes in the IDCON to ensure certain access latency and/or coverage of its services [10]. We denote the subset of the edge nodes that VN $v_{k,i}^r$ can be mapped onto as $V_{k,i}^E$ and have $V_{k,i}^E \subseteq V^E$. Each VL $e \in E_k^r$ has a bandwidth requirement of $R_{k,e}^r$, and it should be mapped onto a substrate path with sufficient bandwidth.

Based on the resource advertisement from the InP, the tenant calculates the VNE scheme for its VNT request with the lowest cost, as shown in Fig. 2(c). This can be done by leveraging the ILP model listed in Table III. Note that, in Eq. (9), $E(v)^-$ and $E(v)^+$ mean the sets of egress and ingress links to node v, respectively. After obtaining the VNE scheme by solving the ILP, the tenant checks whether the scheme's cost is affordable (*i.e.*, not exceeding $\hat{P}_k^r$). If yes, the tenant will submit the VNE scheme and the corresponding payment to the InP. Otherwise, it will cancel its VNT request temporarily.

For the VNT requests submitted to the InP, we denote the set of their indices as K'. Then, based on the payments from the tenants and the corresponding resource costs, the InP calculates the profit from each VNT request (*i.e.*, payment minus total resource cost), sort the requests in descending order of the profits from them, and grant them one-by-one in sorted order. Note that, in this process, a VNT request can be blocked due to insufficient resources in the IDCON. Hence, the granted VNT requests may be a subset of the submitted ones, and the set of their indices can be denoted as K''. Finally, with K'', the InP can calculate its profit from this round of VNT slicing, which is denoted as $\mathcal{P}$. According to Table II, the strategy of resource advertising and pricing can be represented with the advertisement matrix $\mathbf{A} = [\{x_{o,i}^E\}, \{x_{d,i}^E\}, \{x_{o,i}^I\}, \{y_e\}]$ and the price matrix $\mathbf{P} = [\{P_{o,i}^E\}, \{P_{d,i}^E\}, \{P_{o,i}^I\}, \{P_e\}, \{\tilde{P}_e\}]$. We can see that the profit $\mathcal{P}$ is actually a function of $\mathbf{A}$ and $\mathbf{P}$, *i.e.*, $\mathcal{P} = f(\mathbf{A}, \mathbf{P})$. In the next section, we will design a DRL-based algorithm to let the InP learn $\mathcal{P} = f(\mathbf{A}, \mathbf{P})$ intelligently.

TABLE III
ILP MODEL FOR TENANT TO CALCULATE VNE SCHEME OF THE k-TH VNT REQUEST

Variable	Definition
$x_{i,i'}$	Boolean variable that equals 1 if the i-th VN $v_{k,i}^r$ in V_k^r is mapped onto the i'-th edge node $v_{i'}^E$ in V^E, and 0 otherwise.
$y_{e,e'}$	Boolean variable that equals 1 if VL $e \in E_k^r$ goes through fiber link $e' \in E$, and 0 otherwise.

Objective:

$$\text{Minimize} \left(\sum_{v_{k,i}^r \in V_{k,i}^E} \sum_{v_{i'}^E \in V^E} P_{d,i'}^E \cdot x_{i,i'} \cdot R_{k,i}^r + \sum_{e \in E_k^r} \sum_{e' \in E} \tilde{P}_{e'} \cdot y_{e,e'} \cdot R_{k,e}^r \right). \tag{3}$$

Node Mapping Constraints:

$$x_{i,i'} \le x_{o,i'}^E, \quad \forall v_{k,i}^r \in V_k^r,\ v_{i'}^E \in V_{k,i}^E, \tag{4}$$

$$x_{i,i'} = 0, \quad \forall v_{k,i}^r \in V_k^r,\ v_{i'}^E \notin V_{k,i}^E, \tag{5}$$

$$\sum_{v_{i'}^E \in V_{k,i}^E} x_{i,i'} = 1, \quad \forall v_{k,i}^r \in V_k^r, \tag{6}$$

$$x_{o,i'}^E + x_{i,i'} - 1 \le x_{d,i'}^E, \quad \forall v_{k,i}^r \in V_k^r,\ v_{i'}^E \in V_{k,i}^E. \tag{7}$$

Link Mapping Constraints:

$$y_{e,e'} \le y_{e'}, \quad \forall e \in E_k^r,\ e' \in E, \tag{8}$$

$$\sum_{e' \in E(v_{i'}^E)^-} y_{e,e'} - \sum_{e' \in E(v_{i'}^E)^+} y_{e,e'} = x_{i,i'} - x_{j,i'}, \quad \{e : e = (v_{k,i}^r, v_{k,j}^r), e \in E_k^r\},\ \forall v_{i'}^E \in V^E. \tag{9}$$

Resource Constraints:

$$\sum_{v_{k,i}^r \in V_k^r} x_{i,i'} \cdot R_{k,i}^r \le R_{i'}^c, \quad \forall v_{i'}^E \in V^E, \tag{10}$$

$$\sum_{e \in E_k^r} y_{e,e'} \cdot R_{k,e}^r \le R_{e'}^b, \quad \forall e' \in E. \tag{11}$$

III. AI-ASSISTED RESOURCE ADVERTISING AND PRICING

We first design an evaluate function $\widehat{Q}(\cdot)$ that can rank network elements in the IDCON to get the advertisement matrix

$\mathbf{A}$, and then propose a DRL-based algorithm to parameterize $\widehat{Q}(\cdot)$ such that the price matrix $\mathbf{P}$ can be learned iteratively.

A. Design of Evaluation Function $\widehat{Q}(\cdot)$

The evaluation function $\widehat{Q}(\cdot)$ should be able to rank network elements in the IDCON such that if the InP turns down them in the sorted order (*i.e.*, maximizing $\widehat{Q}(\cdot)$ each time), its profit can be maximized. We formulate $\widehat{Q}(\cdot)$ as $\widehat{Q}(\mathbf{A}, n_d; \Theta, \mathbf{P})$, which is a function of $\mathbf{A}$ and n_d with parameters Θ and $\mathbf{P}$, and n_d is the network element (*i.e.*, a DC, an OXC or a fiber link) to be shut down and removed from the upcoming advertisement. Supposing that $\widehat{Q}(\cdot)$ has already been parameterized with known Θ and $\mathbf{P}$, the InP can use the simple procedure in *Algorithm* 1 to obtain the advertisement matrix $\mathbf{A}$.

Algorithm 1: Determining Advertisement Matrix $\mathbf{A}$ with Evaluation Function $\widehat{Q}(\mathbf{A}, n_d; \Theta, \mathbf{P})$

```
initialize A as turning on all the elements in G(V, E);
calculate the InP's profit 𝒫 based on A and P with the
  approach in Section II;
𝒫' = 0;
while 𝒫 > 𝒫' do
    𝒫' = 𝒫, n_d = argmax_{n_d ∈ A} Q̂(A, n_d; Θ, P);
    shut down n_d and update A accordingly;
    calculate the InP's profit 𝒫 based on A and P;
end
```

We design a recursive structure [17] for $\widehat{Q}(\cdot)$ to capture the features of each network element, by considering both the characteristics of the IDCON's topology $G(V, E)$ and the element's relation with other elements in the IDCON. Specifically, at the t-th recursion, the features of n_d are represented by a $(2|V^E|+|V^I|+|E|)$-dimensional vector $\varpi_{n_d}^{(t)}$, and the recursive relations are defined as follows.

$$\varpi_{n_d}^{(t)} = \begin{cases} f_0(\theta_1 \cdot x_{o,i}^E + \theta_2 \cdot A_1 + \theta_3 \cdot f_0(B_1)), & n_d \text{ is the OXC in } v_i^E, \\ f_0(\theta_1 \cdot x_{d,i}^E + \theta_2 \cdot \varpi_{O(v_i^E)}^{(t-1)}), & n_d \text{ is the DC in } v_i^E, \\ f_0(\theta_1 \cdot x_{o,i}^I + \theta_2 \cdot A_2 + \theta_3 \cdot f_0(B_2)), & n_d \text{ is the OXC in } v_i^I, \\ f_0(\theta_1 \cdot y_e + \theta_2 \cdot A_3 + \theta_3 \cdot f_0(B_3)), & n_d \text{ is fiber link } e, \end{cases}$$

where $f_0(x) = x$ if $x \geq 0$, and 0 otherwise, and the parameters $\{A_m, B_m : m \in [1, 3]\}$ are calculated as follows

$$\begin{cases} A_1 = \varpi_{D(v_i^E)}^{(t-1)} + \sum\limits_{n_d \in N(v_i^E)} \varpi_{n_d}^{(t-1)} + \sum\limits_{n_d \in E(v_i^E)} \varpi_{n_d}^{(t-1)}, \\ B_1 = \sum\limits_{v_j^E \in N(v_i^E)} \theta_4 \cdot \left(P_{o,j}^E + P_{d,j}^E\right) + \sum\limits_{v_j^I \in N(v_i^E)} \theta_5 \cdot P_{o,j}^I + \sum\limits_{e \in E(v_i^E)} \theta_6 \cdot P_e, \\ A_2 = \sum\limits_{n_d \in N(v_i^I)} \varpi_{n_d}^{(t-1)} + \sum\limits_{n_d \in E(v_i^I)} \varpi_{n_d}^{(t-1)}, \\ B_2 = \sum\limits_{v_j^E \in N(v_i^I)} \theta_4 \cdot \left(P_{o,j}^E + P_{d,j}^E\right) + \sum\limits_{v_j^I \in N(v_i^I)} \theta_5 \cdot P_{o,j}^I + \sum\limits_{e \in E(v_i^I)} \theta_6 \cdot P_e, \\ A_3 = \sum\limits_{\{v_i^E :\, e \in E(v_i^E)\}} \varpi_{O(v_i^E)}^{(t-1)} + \sum\limits_{\{v_i^I :\, e \in E(v_i^I)\}} \varpi_{O(v_i^I)}^{(t-1)}, \\ B_3 = \sum\limits_{\{v_i^E :\, e \in E(v_i^E)\}} \theta_4 \cdot \left(P_{o,i}^E + P_{d,i}^E\right) + \sum\limits_{\{v_i^I :\, e \in E(v_i^I)\}} \theta_5 \cdot P_{o,i}^I, \end{cases}$$

where $N(v)$ returns the set of OXCs in adjacent nodes of node v, and $D(v)$ and $O(v)$ return the DC and OXC in node v, respectively. With T recursions, the features of each network element are spread to those that are T hops away from it. Then, the evaluation function $\widehat{Q}(\mathbf{A}, n_d; \Theta, \mathbf{P})$ can be formulated as

$$\widehat{Q}(\mathbf{A}, n_d; \Theta, \mathbf{P}) = \theta_7^\top \cdot f_0\left(\left[\theta_8 \cdot \sum_{n_d' \in G} \varpi_{n_d'}^{(T)},\ \theta_9 \cdot \varpi_{n_d}^{(T)}\right]\right), \quad (12)$$

where $\Theta = \{\theta_i : i \in [1, 9]\}$.

B. DRL-based Algorithm to Parameterize $\widehat{Q}(\cdot)$

We propose a DRL-based algorithm with the following principle to parameterize $\widehat{Q}(\cdot)$, *i.e.*, determining Θ and $\mathbf{P}$.

- *States*: each state corresponds to a feasible $\mathbf{A}$.
- *Actions*: an action is to shut down one network element n_d at the current state $\mathbf{A}$.
- *Rewards*: the reward of an action at the current state $\mathbf{A}$ is calculated as:

  $$f_r(\mathbf{A}, n_d) = f(\mathbf{A}/n_d, \mathbf{P}) - f(\mathbf{A}, \mathbf{P}), \quad (13)$$

  where $f(\mathbf{A}, \mathbf{P})$ calculates the InP's profit, and $\mathbf{A}/n_d$ means to shut down n_d at state $\mathbf{A}$.

Based on Eq. (13), we define an n-step-forward function

$$y = \sum_{i=0}^{n-1} f_r(\mathbf{A}^{(t+i)}, n_d^{(t+i)}) + \beta \cdot \max_{n_d} \left[\widehat{Q}(\mathbf{A}^{(t+n)}, n_d; \Theta, \mathbf{P})\right], \quad (14)$$

where t is the index of the current iteration, $\mathbf{A}^{(t+i)}$ and $n_d^{(t+i)}$ are the state and action at the $(t+i)$-th iteration, respectively, and β is a constant coefficient. Then, in the DRL, we try to minimize the squared regression loss defined as

$$\left[y - \widehat{Q}(\mathbf{A}^{(t)}, n_d^{(t)}; \Theta, \mathbf{P})\right]^2. \quad (15)$$

Algorithm 2 shows the procedure of the proposed DRL-based algorithm. In each round of training, we first create two sets Γ and Δ (*Lines* 2-3). The former is to store all the valid training samples, and the latter is the training set with a fixed size for an iteration. *Line* 4 initializes $\mathbf{A}^{(1)}$, and the for-loop covering *Lines* 5-21 tries to shut down a network element in each iteration. Here, to diversify the training samples, we generate a random number $\epsilon \in [0, 1]$ (*Line* 6), and test whether it is smaller than a preset threshold T_h. If yes, the action $n_d^{(t)}$ is randomly selected within $\mathbf{A}^{(t)}$ (*Line* 8). Otherwise, the action is determined according to the policy in *Line* 10. Then, we get $\mathbf{A}^{(t+1)}$ accordingly (*Line* 12) and calculate the corresponding reward in *Line* 13. Due to the n-step-forward function in Eq. (14), only when the iteration number is larger than n, $\{\mathbf{A}^{(t-n)}, n_d^{(t-n)}, f_r(\mathbf{A}^{(t-n)}, n_d^{(t-n)})\}$ becomes a valid sample. Hence, it is added into Γ in *Line* 15. Once there are more than $|\Delta|$ samples in Γ (*Line* 16), the training set Δ can be formed by selecting $|\Delta|$ samples from Γ randomly (*Line* 17), and then the values of $\{\Theta, \mathbf{P}\}$ are updated by performing stochastic gradient descent (SGD) over Eq. (15) for Δ (*Line* 18). Finally, after M rounds of training, *Algorithm* 2 determines and returns the values of $\{\Theta, \mathbf{P}\}$.

Algorithm 2: DRL-based Algorithm to Parameterize $\widehat{Q}(\cdot)$

for *each round* $j \in [1, M]$ **do**
  create a set Γ to store valid training samples;
  create a training set Δ with a fixed size;
  initialize $\mathbf{A}^{(1)}$ as turning on all elements in $G(V, E)$;
  for *each iteration* $t \in [1, 2|V^E| + |V^I| + |E|]$ **do**
    generate a random real number $\epsilon \in [0, 1]$;
    if $\epsilon \leq T_h$ **then**
      select $n_d^{(t)}$ randomly in $\mathbf{A}^{(t)}$;
    else
      $n_d^{(t)} = \underset{n_d \in \mathbf{A}^{(t-1)}}{\text{argmax}}\ \widehat{Q}(\mathbf{A}^{(t)}, n_d; \Theta, \mathbf{P})$;
    end
    get $\mathbf{A}^{(t+1)}$ from $\mathbf{A}^{(t)}$ by removing $n_d^{(t)}$;
    calculate $f_r(\mathbf{A}^{(t)}, n_d^{(t)})$ with Eq. (13);
    if $t \geq (n + 1)$ **then**
      $\{\mathbf{A}^{(t-n)}, n_d^{(t-n)}, f_r(\mathbf{A}^{(t-n)}, n_d^{(t-n)})\} \rightarrow \Gamma$;
      if $|\Gamma| \geq |\Delta|$ **then**
        select $|\Delta|$ samples from Γ randomly to form Δ;
        update $\{\Theta, \mathbf{P}\}$ by performing SGD over Eq. (15) for Δ;
      end
    end
  end
end
return $\{\Theta, \mathbf{P}\}$;

IV. Performance Evaluation

We conduct numerical simulations to evaluate the proposed framework with DRL-based resource advertising and pricing, which run on a computer with 4.0 GHz Inter Core i7-6700K CPU, 16 GB RAM and 11 GB NVIDIA GTX 1080Ti GPU. The DRL-based algorithm is implemented with TensorFlow 1.4.1. The topology of the IDCON can take either the 8-node or the NSFNET topologies in Fig. 3. The cost of resources are uniformly distributed within $[10, 30]$ units in both topologies. Here, unit stands for a general currency unit. We generate each VNT request in a way as: 1) the number of VNs is uniformly distributed within $[1, 5]$, 2) the subset of edge nodes that each VN is location-constrained within is randomly selected, 3) each VN pair is connected by a VL with a probability of 0.6, and 4) the highest cost that a tenant can afford is linearly proportional to the total number of VNs and VLs, with a slope uniformly distributed within $[16, 116]$ and $[50, 150]$ units in the 8-node and NSFNET topologies, respectively.

When training the proposed DRL-based algorithm, we use $M = 200$ as the maximum number of training rounds, set the number of VNT requests in each round as uniformly distributed within $[5, 45]$, and have $|\Delta| = 5$ as the number of training samples. After the training is done, we compare the proposed DRL-based algorithm with a centralized benchmark. The benchmark prices resources according to a normal distribution with mean μ and standard deviation σ equal to $\{65.7, 26.8\}$ and $\{123, 37.4\}$ for the 8-node and NSFNET topologies, respectively, and performs resource advertising in a greedy manner. Specifically, in the benchmark, by estimating the network element that can be shut down to bring in the maximum profit gain, the InP removes selected network elements one-by-one from the upcoming resource advertisement until its profit is maximized, and the whole process does not consider any inputs from the tenants'. The simulations average the results from 5 independent runs to get each data point.

Fig. 4 shows the results on the InP's profit and the algorithms' average running time for the 8-node topology. We can see that the proposed DRL-based algorithm achieves $\sim 14.56\%$

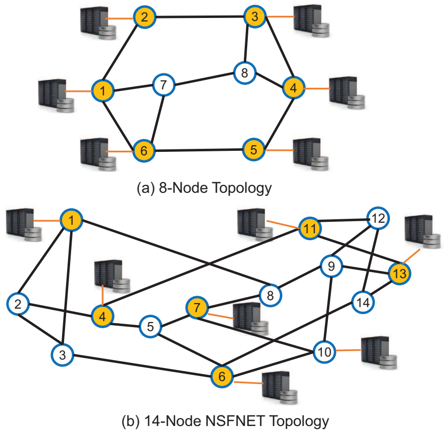


Fig. 3. IDCON topologies used in simulations.

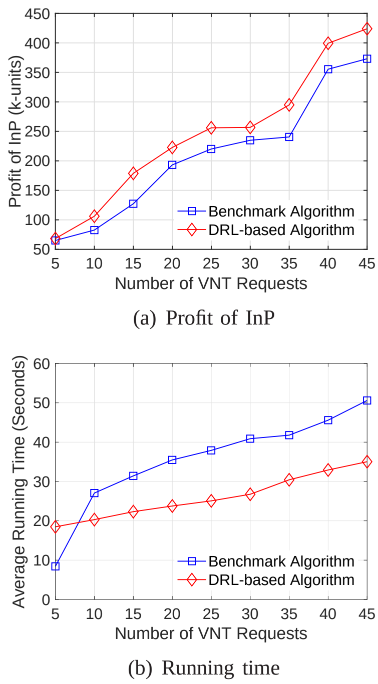
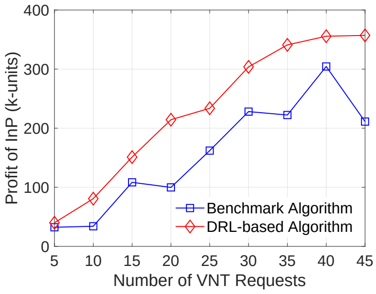
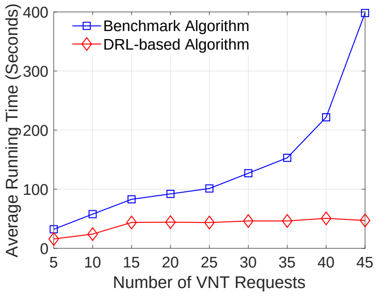


Fig. 4. Results in 8-node topology.



(a) Profit of InP



(b) Running time

Fig. 5. Results in the 14-node NSFNET topology.

more profit than the benchmark in all the cases. Meanwhile, the DRL-based algorithm also consumes less running time than the benchmark. Here, to achieve fair comparisons, the running time of the DRL-based algorithm includes the training time. This is because with the benchmark, the InP has to calculate the VNE schemes for the VNT requests and determine the resource advertising scheme in a centralized manner, while with the DRL-based algorithm, it only needs to perform network advertising according to the trained evaluation function and grant the VNE schemes from the tenants based on resource availability. Hence, the results in Fig. 4 verify that our proposed framework can not only make the InP more profitable but also relieve its computation complexity effectively.

Fig. 5 illustrates the results on the InP's profit and average running time in the NSFNET topology. We observe the similar trends as in Fig. 4. Actually, as the IDCON's size is larger, the proposed DRL-based algorithm achieves a larger profit increase over the benchmark, *i.e.*, $\sim 58.40\%$ on average. Since with the benchmark, the InP does not consider the tenants' inputs when determining resource advertising and pricing schemes, its profit in Fig. 5(a) does not exhibit a stable trend. Moreover, the centralized scheme of the benchmark does not scale well with the problem's size, and thus its running time increases exponentially with the number of VNT requests in Fig. 5(b). This makes our proposed framework's advantage on reduced time complexity much more significant.

V. Conclusion

We proposed a novel framework to realize VNT slicing in an IDCON, where the InP performs AI-assisted resource advertising and pricing and grants the VNE schemes calculated distributedly by the tenants. Then, for the InP, we designed a DRL-based resource advertising and pricing algorithm for profit maximization. Simulation results confirmed that compared with the traditional centralized VNT slicing framework, our proposal can not only make the InP more profitable but also relieve its computation complexity effectively. In the future, we need to further study both the timing and methods for predicting the tenant demands and their affordable prices accurately.

Acknowledgments

This work was supported in part by the NSFC Projects 61701472 and 61771445, CAS Key Project (QYZDY-SSW-JSC003), NGBWMCN Key Project (2017ZX03001019-004), China Postdoctoral Science Foundation (2016M602031), and Research Funds for Central Universities (WK2100060021).

References

[1] "Cisco global cloud index: Forecast and methodology, 2016-2021," 2017, [Online]. Available: https://www.cisco.com/c/en/us/solutions/service-provider/visual-networking-index-vni/index.html.
[2] P. Lu *et al.*, "Highly-efficient data migration and backup for big data applications in elastic optical inter-datacenter networks," *IEEE Netw.*, vol. 29, pp. 36–42, Sept./Oct. 2015.
[3] H. Jiang, Y. Wang, L. Gong, and Z. Zhu, "Availability-aware survivable virtual network embedding in optical datacenter networks," *J. Opt. Commun. Netw.*, vol. 7, pp. 1160–1171, Dec. 2015.
[4] M. Chowdhury and R. Boutaba, "Network virtualization: State of the art and research challenges," *IEEE Commun. Mag.*, vol. 47, pp. 20–26, Jul. 2009.
[5] L. Gong, Y. Wen, Z. Zhu, and T. Lee, "Toward profit-seeking virtual network embedding algorithm via global resource capacity," in *Proc. of INFOCOM 2014*, pp. 1–9, Apr. 2014.
[6] M. Bari *et al.*, "Data center network virtualization: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, pp. 909–928, Second Quarter 2013.
[7] L. Gong and Z. Zhu, "Virtual optical network embedding (VONE) over elastic optical networks," *J. Lightw. Technol.*, vol. 32, pp. 450–460, Feb. 2014.
[8] X. Cheng *et al.*, "Virtual network embedding through topology-aware node ranking," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, pp. 38–47, April 2011.
[9] M. Chowdhury, M. Rahman, and R. Boutaba, "Vineyard: Virtual network embedding algorithms with coordinated node and link mapping," *IEEE/ACM Trans. Netw.*, vol. 20, pp. 206–219, Feb. 2012.
[10] L. Gong, H. Jiang, Y. Wang, and Z. Zhu, "Novel location-constrained virtual network embedding (LC-VNE) algorithms towards integrated node and link mapping," *IEEE/ACM Trans. Netw.*, vol. 24, pp. 3648–3661, Dec. 2016.
[11] R. Munoz *et al.*, "Integrated SDN/NFV management and orchestration architecture for dynamic deployment of virtual SDN control instances for virtual tenant networks," *J. Opt. Commun. Netw.*, vol. 7, pp. B62–B70, Nov. 2015.
[12] J. Yin *et al.*, "Experimental demonstration of building and operating QoS-aware survivable vSD-EONs with transparent resiliency," *Opt. Express*, vol. 25, pp. 15 468–15 480, 2017.
[13] Z. Zhu *et al.*, "Build to tenants' requirements: On-demand application-driven vSD-EON slicing," *J. Opt. Commun. Netw.*, vol. 10, pp. A206–A215, Feb. 2018.
[14] W. Fang *et al.*, "Joint defragmentation of optical spectrum and IT resources in elastic optical datacenter interconnections," *J. Opt. Commun. Netw.*, vol. 7, pp. 314–324, Mar. 2015.
[15] F. Tso, S. Jouet, and D. Pezaros, "Network and server resource management strategies for data centre infrastructures: A survey," *Comput. Netw.*, vol. 106, pp. 209–225, Sept. 2016.
[16] A. Mestres *et al.*, "Knowledge-defined networking," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 47, pp. 2–10, Jul. 2017.
[17] H. Dai, B. Dai, and L. Song, "Discriminative embeddings of latent variable models for structured data," in *Proc. of ICML 2016*, pp. 2702–2711, Jun. 2016.