Energy-efficient Deep Reinforced Traffic Grooming in Elastic Optical Networks for Cloud-Fog Computing

Ruijie Zhu, *Member, IEEE,* Shihua Li, *Student Member, IEEE,* Peisen Wang, *Student Member, IEEE,* Mingliang Xu, *Member, IEEE,* and Shui Yu, *Senior Member, IEEE*

***Abstract*—Cloud-fog computing emerges to satisfy the low latency and high computation requirements of Internet of Things (IoTs) services. Elastic Optical Networks (EONs) are excellent substrate communication networks between fog datacenters and cloud datacenters. However, the uneven traffic of massive cloud-fog services incurs many spectrum fragments, leading to high extra energy consumption. To solve this problem, we propose an Energy-efficient Deep Reinforced Traffic Grooming (EDTG) algorithm based on deep reinforcement learning. Unlike existing manually network features extracting methods, we convert the traditional network modal and the service routing path into colored network images to represent their states, and extract the features automatically by MobilenetV3 according to these images. With the extracted features, we implement an Advantage Actor-Critic (A2C) algorithm, whose actor module and critic module share an Artificial Neural Network (ANN) to get optimal grooming actions. Additionally, after repeated attempts and experiments, we set up an objective reward and punishment mechanism to evaluate the grooming actions. We conduct extensive simulations for performance evaluation, and the results have shown that EDTG can significantly reduce energy consumption compared with two well-performed traffic grooming algorithms.**

***Index Terms*—Elastic Optical Networks (EONs), Deep Reinforcement Learning, Traffic Grooming, Cloud-fog Computing, Energy Efficient.**

I. INTRODUCTION

CLOUD computing is good at providing substantial computing resources to process requests [1]. With the explosive growth of global Internet of Things (IoTs) services, traditional cloud computing cannot satisfy their low latency requirement [2]. Fog computing emerges as a complementation of cloud computing to decrease the latency [3]. As shown in Fig. 1, the cooperation of cloud and fog computing can provide high quality services to the customers [4]. Elastic Optical Networks (EONs) are the promising substrate communication networks between fog datacenters and cloud datacenters [5]. It can adaptively allocate spectrum and flexibly select modulation format. With the rapid development of 5G, Augmented Reality (AR), Virtual Reality (VR), and other emerging services, EONs have been widely deployed as substrate networks carrying 5G services. As many mice traffic are generated in EONs, these incur spectrum fragmented, and extra energy consumption [6]. How to groom these mice traffic onto the same channel to save spectrum resource and to decrease energy consumption is extremely important [7]-[10].

Previous studies have investigated the traffic grooming problem intensively, which are dedicated to the bandwidth-efficient utilization problem [11], the energy-saving problem [12], and the expense cost-saving problem [13]. Especially, a State-aware Modification Grooming Algorithm (SGA) is proposed to solve the traffic grooming problem. SGA has been applied to balance network indicators with survivable EONs to achieve high network performance [14]. Although heuristic algorithms similar to SGA and traditional optimization algorithms perform well in traffic grooming, they need to extract features manually. The features dimension cannot cover the whole EONs state, and they cannot intelligently perceive the EON state to groom the services. Deep Reinforcement Learning (DRL) algorithm has been applied in some large-scale missions [15]-[17]. It can make decisions by perceiving the state, and can evaluate the decisions to maximize cumulative rewards and get better strategies [18]. To solve the problem of temporal and spatial imbalance caused by network load, DRL stateful grooming algorithm is proposed to accommodate

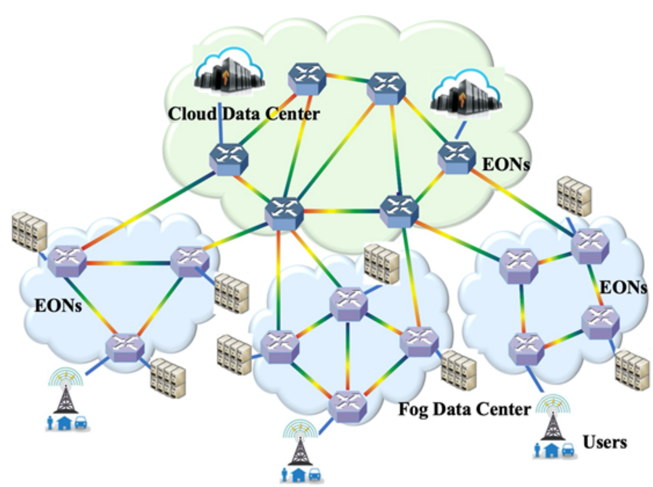


Fig. 1: The structure of cloud-fog elastic optical networks.

Manuscript received September 24, 2020; revised January 6, 2021 and February 9, 2021; accepted February 28, 2021. This work was supported in part by National Science Foundation of China under Grants No. 62001422, 61971380, and Open Fund of IPOC (BUPT) No. IPOC2019A008. (Corresponding author: Mingliang Xu.)

Ruijie Zhu, Shihua Li, Peisen Wang, Mingliang Xu are with School of Information Engineering, Zhengzhou University, Zhengzhou, China. Shui Yu is with School of Computer Science, University of Technology Sydney, Ultimo, Australia. (e-mail: zhuruijie@zzu.edu.cn; iexumingliang@zzu.edu.cn; Shui.Yu@uts.edu.au;)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3063471, IEEE Internet of Things Journal

the variability according to the network state [19]. However, these previous works all rely on manually extracting network features, and how to apply DRL to solve the traffic grooming problem automatically in EONs for cloud-fog computing has not been studied.

In this work, we design an Energy-efficient Deep Reinforced Traffic Grooming (EDTG) algorithm to generate the optimal traffic grooming strategy in EONs for cloud-fog computing. The main contributions of this work include the following aspects:

- To optimize energy consumption in traffic grooming, we design an EDTG algorithm based on the Advantage Actor-Critic (A2C) algorithm. We provide a tractable solution that the actor module and the critic module share an ANN to complete tasks, which significantly reduces the number of parameters and the complexity of the network.
- Unlike existing methods that manually extract network features from traditional network modal, we convert the network modal into colored pixel images to represent the network states and adopt pixel features to represent the network information. This not only considers the quantitative information of resources in the process of traffic grooming but also includes other network-wide context information.
- We deeply analyze various situations that in traffic grooming and set up a detailed reward and punishment mechanism with 12 rewards and punishments through repeated attempts and experiments. We use it to evaluate the effect of grooming actions to optimize EDTG.
- We conduct comprehensive simulations, and simulation results show that the EDTG algorithm performs better than several well-performed algorithms, such as State-aware Modification Grooming Algorithm (SGA) [14], and traditional Deep Reinforcement Learning (DRL) algorithm [19].

The rest of this paper is organized as follows. In Section II, the related works are investigated. In Section III, the motivation of this work is presented and an energy model of EONs is designed. In Section IV, the network states are converted into images, and an EDTG algorithm is proposed. Simulations are comprehensively conducted, and the comparisons are taken with several baseline algorithms in Section V. Finally, Section VI gives a succinct conclusion and future work.

II. RELATED WORK

The related works are investigated from three aspects, energy-efficient EONs, energy-aware traffic grooming, and reinforced resource assignment.

A. Energy-efficient EONs

In this subsection, we introduce the previous researches on energy consumption issues in EONS. Due to the tremendous growth of energy consumption in substrate networks, many efforts have been taken towards energy-efficient solutions [20]. Minimizing the number of activated network elements, a manycast routing and spectrum assignment algorithm was designed by A. Fallahpour et al. [21] to save the energy for cloud computing. To save the energy in the delay-constrained passive optical network, an energy-aware framework was presented, optimizing the number of working wavelengths. Y. Zhao et al. [22] developed a static linear programming and a dynamic virtual optical network embedding algorithm to minimize the energy consumption with sliceable multi-flow transponders. Focusing on the predictable tidal traffic phenomenon, R. Alvizu et al. [23] proposed a suite of on-line mathematics to decrease the power in the mobile metro-core networks. Using the network coding technique, Musa et al. [24] analyzed the bounds of energy efficiency of 1+1 survivable electric-optics dual layers networks. For enhancing the energy efficiency in software-defined networks, Xiong et al. [25] utilized traffic prediction methods to manage the lightpath and reduce the number of lightpath termination and re-establishment. To cope with the traffic spikes, Zhong et al. [26] introduced a lightpath splitting approach to support more traffic services through improving the modulation levels of the split lightpaths.

Most of these works applied multicast scheme or network virtualization method to improve energy efficiency. Different with them, this work grooms the mice traffic onto the same network elements as many as possible to reduce the energy consumption.

B. Energy-aware Traffic Grooming

In this subsection, we introduce the advantages of traffic grooming in reducing energy consumption. Enabled by the sliceable optical transponders, optical-layer traffic all grooming can decrease the number of power consumption elements. Zhang et al. [27] investigated three kinds of transponders with different slicing levels. For each transponder, a power-minimized integrated model and the corresponding heuristic algorithm were developed. To solve the problem of mixed-electrical-optical grooming with dynamic traffic, Zhang et al. [28] put forward the concept of a three-layered auxiliary graph, and designed different traffic grooming strategies by adapting the edge weights of the auxiliary graph. To maintain the survivability issue, Wu et al. [29] studied the survivable grooming routing and spectrum assignment in software defined elastic optical networks. To provision mission-critical optical wireless datacenter networks, Celik et al. developed a fast efficient grooming solution that grooms mice flows, mission-critical flows, and forwards on scheduled rack-to-rack lightpaths. To solve the problems of low spectrum efficiency and large network energy consumption, J. Zhang et al. [30] proposed a traffic grooming method for elastic optical networks with distributed data centers. Hadi et al. [31] designed a two-stage energy-efficient resource assignment algorithm, which minimized the number of adopted amplifiers and transponders in the first stage, and optimized the parameters of the transponders to minimize power consumption. Jin et al. [32] analyzed both delay and energy issues in traffic grooming scenarios.

All the above studies adopt the traditional heuristic algorithms for traffic grooming to reduce energy consumption. In this paper, we aim to reduce the energy consumption by minimizing the utilized ports, transponders, regenerators, and spectrum resources. Different from the previous works, we utilize reinforcement learning algorithm to generate optimal grooming scheme to decrease energy consumption.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3063471, IEEE Internet of Things Journal

C. Reinforced Resource Assignment

In this subsection, we introduce some research on the application of reinforcement learning in resource allocation. To support the machine learning based optical networks, Zhao et al. [33] designed a new architecture of the optical network, named self-optimizing optical networks (SOON). Several classical applications have been demonstrated with SOON. Francesco et al. [34] provided a comprehensive exploration of the applications of machine learning in elastic optical networks, and proposed several potential research directions. To accommodate the reinforcement learning algorithm for optical network resource assignment, Yan et al. [35] introduced an interesting definition of multi-modal optical networks to replace different optical network. A resource allocation algorithm based on actor-critic reinforcement learning was proposed, and simulation results showed that it could achieve resource optimization. The deep reinforcement learning was first adapted to the routing, modulation, and spectrum assignment (RMSA) process in EONs, and a Deep-RMSA algorithm was proposed by X. Chen et al. [19]. To improve network performance under survivable EONs, a deep reinforcement learning based RMSA algorithm was investigated by X. Luo et al. [36]. Simulation results showed that it could significantly improve the overall network performance while maintaining the network survivability. To mitigate the crosstalk and fragmentation issue in spatial division multiplexing EONs, Xiong et al. [37] presented a spectrum partition scheme to suppress the crosstalk and utilized a two-dimensional spectrum resource packing method to improve spectrum efficiency. R. Proietti et al. [38] introduced a machine-learning based quality of transmission estimation scheme for lightpath provisioning with intradomain and interdomain traffic. Considering crosstalk and physical layer impairments, Yao et al. [39] designed an intelligent resource allocation algorithm based on dynamic unsupervised fuzzy clustering in SDM-EONs.

These previous works are based on traditional network modal, which make it difficult to extract all useful features. We design a new method to convert the network state into several colorful pixels images, so that we can extract network features easily and groom services accurately.

III. MOTIVATION AND ENERGY MODEL

In this section, we first present the motivation of our work by visualizing the basic idea of general traffic grooming framework, and then describe the energy model for calculating network energy consumption.

A. Motivations

Figure 2 illustrates the difference in resource occupation between grooming and no grooming. If there is no traffic grooming, there are many fragmented resources after allocating massive services [6]. For example, service A with 10Gbps traffic rate is provisioned with 25GHz resources, and then the surplus resources are wasted. Supposing service B with 20Gbps traffic rate also passes through the same routing path with service A, it will be provisioned with other 25GHz resources. But if we groom these two services together as

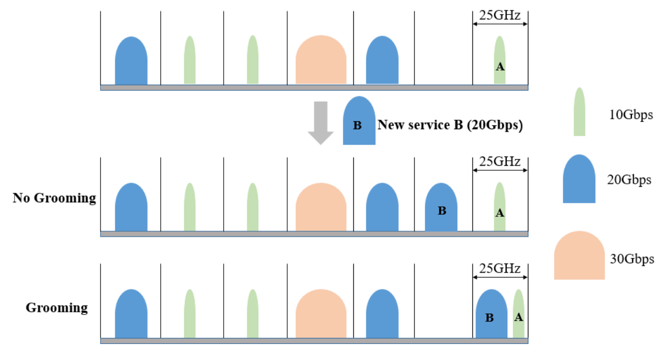


Fig. 2: The grooming process.

the total traffic rate is 30Gbps, which is less than 40Gbps, resource occupancy and energy consumption will be reduced. The traditional network modal cannot perceive the whole state of the network, but the image can contain all the information and we can extract features easily from it. In addition, traffic grooming is a step-by-step decision-making process, which is just right to utilize DRL.

Motivated by these two findings, we propose an Energy-efficient Deep Reinforced Traffic Grooming (EDTG) algorithm. It converts the traditional network modal into images to extract features, and introduces reinforcement learning to groom services. We define the energy consumption model of the network. As the spectrum resources are much more sufficient relative to the requested bandwidths, and each node is equipped with regenerators. Thus the spectrum continuity is not considered [23].

B. Energy consumption model

A given network contains many energy-consuming elements, the most important of which is the energy consumption of ports, transponders and regenerators. The resources of a port, transponder, and regenerator are limited. How to develop a grooming strategy so that these elements can support more services is very important, which can reduce the number of ports, transponders and regenerators, and reduce network energy consumption. In this paper, we only consider the 40Gbps and 100Gbps line rates to define the energy consumption model of the network. Their corresponding energy consumption parameters are shown in Table I [27], which is proved to be feasible. We only consider DP-QPSK as the modulation format for the substrate network EONs. In the experiments, we convert spectrum resources into spectrum slots. A 40Gbps line rate service requires 25GHz spectrum bandwidth, which corresponds to one spectrum slot. A 100Gbps line rate service requires 50GHz spectrum bandwidth, which corresponds to two spectrum slots.

IP Port: We consider a 400Gbps IP port for traffic grooming in the electrical layer, which connects an optical transponder and an IP router. The energy consumption of each port is 560W, and the sum of energy consumption of all IP ports is E_{IPT}, which is calculated by

$$E_{IPT} = 560 \times n, \tag{1}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3063471, IEEE Internet of Things Journal

TABLE I: The energy consumption of different line rates [27].

Line Rate (Gbps)	Spectrum Width (GHz)	Modulation Formart	Reachability (km)	Power of Transponder (W)
40	25	DP-QPSK	1800	158.65
100	50	DP-QPSK	2000	259.63

where n represents the number of IP port.

Optical Transponder: The energy consumption parameters corresponding to different line rates are shown in Table I, which is proved to be feasible. The energy consumption for an elastic optical transponder is given by

$$E_{OPT}^{i} = 1.683 \times TR(GB/s) + 91.333(W), \tag{2}$$

where i represents the i-th transponder, TR denotes the traffic rate of a service. The sum of energy consumption of all optical transponders is,

$$E_{OPT} = \sum_{i=1}^{N_{OPT}} E_{OPT}^{i}, \tag{3}$$

where N_{OPT} represents the number of occupied optical transponders. There are two types of line rates, 40Gbps and 100Gbps. For example, a transponder is occupied by service A, whose line rate is 40Gbps, the energy consumption calculated according to formula 2 is $158.65W$.

Optical Regenerator: For an optical regenerator, we consider two parts of its energy consumption. One part depends on the fundamental power of the optical regenerator, for every increase of one regenerator, the energy consumption increases by $\mu = 100W$. The other part θ is the energy consumption according to the different modulation formats which have different line rates. The θ for line rate 40 is 25W, the θ for line rate 100 is 50W. The calculation formulas are as follows,

$$E_{OPR}^{i} = \mu + \theta, \tag{4}$$

$$E_{OPR} = \sum_{i=1}^{N_{OPR}} E_{OPR}^{i}, \tag{5}$$

where θ is the corresponding energy consumption with different line rates. The energy consumption of 40Gbps is 25W, and the energy consumption of 100Gbps is 50W. N_{OPR} represents the number of optical regenerators. Thus, for line rate 40Gbs and 100Gbps, the corresponding energy consumption of a transponder is 125W and 150W, respectively.

Therefore, the optimization goal is to minimize the total energy consumption E_{TG},the formula of calculating E_{TG} is as follows,

$$E_{TG} = E_{IPT} + E_{OPT} + E_{OPR}. \tag{6}$$

IV. DEEP REINFORCEMENT TRAFFIC GROMMING

How to perceive the state of a heterogeneous network for traffic grooming algorithms is complicated and vitally important. The information contained in the network state is intricate, and it is challenging to perceive traditional network modal features. In this part, we first convert the network states into different images and extract the features of these images by MobilenetV3. Then we design an Energy-efficient Deep Reinforced Traffic Grooming (EDTG) algorithm based on the Advantage Actor-Critic (A2C) reinforcement learning algorithm.

A. Converted to Images

The network information is discontinuous, such as the number of elements, the locations of different elements, and whether the spectrum resources are occupied. All the information cannot be easily extracted based on the traditional network modal. However, the pixels are continuous in an image, and we can extract useful information easily through a Convolutional Neural Network (CNN). Motivated by the explosion of information and the excellent performance of image in extracting features, we implement a mechanism of converting the traditional network modal into colorful network images.

As shown in Fig. 3, we take the NSFNET as an example. We set up NSFNET to have five wavelengths according to the resource simulation of the optical network, and there are ten spectrum slots on each link of each wavelength. We aim to groom as many services as possible to the same wavelength, so as to maximize resource utilization. We divide NSFNET into five parts. The information presented by the traditional network modal is limited. It is impossible to know where resources in the network are occupied and where resources are available. Even the network matrix vector with this information can hardly cover all the information. Driven by these limitation, we use different shapes with 11 colors to represent the different resources occupancy states of ports, transponders, regenerators, and links. For a regenerator, we use a triangle to represent it. When service arrives one after another, whenever the 25GHz spectrum resources (a spectrum slot) of regenerator A is occupied, the triangle changes the color followed by black, purple, green, cyan, red, blue, gray, orange, yellow, pink and white. When all the spectrum slots of regenerator A are occupied, it means that regenerator A can no

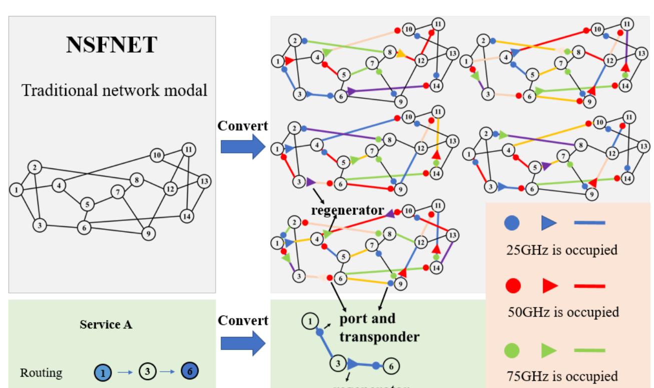


Fig. 3: Convert to images.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3063471, IEEE Internet of Things Journal

longer support more services, and we use a white triangle to represent it. Similarly, we use solid circles to represent ports and transponders, and line segments to represent links. In this way, we convert the states of the five parts of the NSFNET into five images, namely G_1, G_2, G_3, G_4, G_5. The service routing path is calculated by the shortest path algorithm, and it is also converted into a image S_r. We extract features from G_1, G_2, G_3, G_4, G_5 and S_r, these six images are as the input of the CNN, rather than treated as an image. Then we groom the service based on these features.

B. EDTG algorithm

Traffic grooming is a process of perceiving the current state of the network and grooming services to a specific part where resources are available. Deep learning algorithms perform well in perceiving the environment, but they lack the specific decision-making ability to determine service grooming problems. Reinforcement learning algorithms make actions according to the states and continuously optimize the action strategy. Therefore, Deep Reinforcement Learning (DRL) is introduced in this paper, it integrates the advantages of these two techniques, and it can learn decision-making methods from high-dimensional raw data directly.

The Actor-Critic (AC) algorithm is a classic DRL algorithm composed of the actor module and critic module. The actor module selects actions based on the calculated probability, and the critic module evaluates the effect of actions. In the AC algorithm, the strategy network π_θ deployed by the actor module for decision-making is an ANN, whose parameter is θ. During the whole process of the task, each step can get a reward value r, and the final reward obtained by the whole task is R. The function $Q(s,a)$ is used instead of R, and the critic module is adopted to calculate the function value Q. To maximize the expected reward, the gradient strategy is used as follows,

$$\bigtriangledown R_\theta = \frac{1}{N}\sum_{n-1}^{N}\sum_{t-1}^{T} Q^{\pi_\theta}(s,a) \bigtriangledown logp_\theta(s,a), \tag{7}$$

where N represents the number of iterations, p_θ represents the action policy of AC. The critic value is updated according to the mean square error between the estimated Q value and the actual Q value. The calculation formula of the loss is as follows,

$$loss = \frac{1}{N}\sum_{n-1}^{N}\sum_{t-1}^{T}(r + maxQ^{\pi_\theta}(s_{t+1},a_{t+1}) - Q^{\pi_\theta}(s,a))^2, \tag{8}$$

where T is the total time, t represents a time unit.

However, AC algorithm is difficult to converge because some actions cannot be sampled. In order to balance the actions not taken in the algorithm, a baseline needs to be added to make the feedback positive and negative. The Advantage Actor-Critic (A2C) algorithm is proposed to optimize the AC algorithm. A2C uses the advantage function to replace the original reward in the Critic network, which can be used as an indicator to measure the selected action value and the average value of all actions. The specific formula is as shown in the formula 10-12.

The Asynchronous Advantage Actor-Critic (A3C) algorithm uses an asynchronous update method, and does not require a large amount of memory like Deep Q Network (DQN) for experience playback. And it breaks the correlation between data in neural network training. In fact, each asynchronous worker of A3C is an independent A2C, but its structure is more complicated. Therefore, we designed a multi-process A2C algorithm, which is comparable in speed to A3C, but its structure is simpler.

By introducing the A2C algorithm, we propose a traffic grooming algorithm named EDTG, as shown in Fig. 4. The

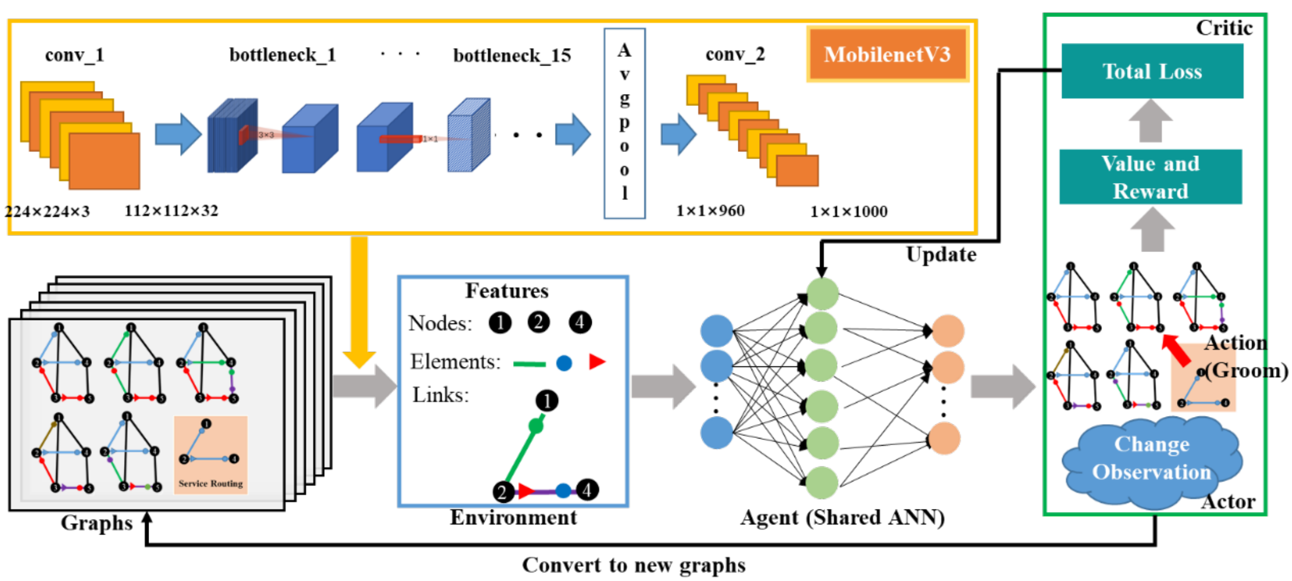


Fig. 4: EDTG Algorithm.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3063471, IEEE Internet of Things Journal

Algorithm 1 EDTG Algorithm

Initialize the network, all resources are available.
$i = 1$.
while $i < I$ **do**
  Convert current network state s_i and service i routing to image.
  Observe state s_i to check the resources required by service i.
  Take an action a_i to groom i according to strategy $\pi(a_i|s_i;\theta)$.
  Observe state s_{i+1} and get reward r_i.
  $i = i + 1$.
  if $i \in U$ **then**
    $R_i \leftarrow V(s_{i,\theta_v})$.
    Calculate loss l_v,l_a,and entropy e.
    Calculate total loss $l_t = l_v \cdot c_v + l_a - e \cdot c_e$
    Update θ_v and θ_a according to loss l_t.
  end if
  if $i \in E$ **then**
    Clean all occupation of resources.
    $i = 1$.
  end if
end while

Actor module uses the MobilenetV3 network to extract the characteristics of the six input images, and uses the ANN network to select which wavelength to groom the service represented by the routing image S_r according to the characteristics. The Critic network is also an ANN, it evaluates the impact of actions on the network and updates the parameters of ANN network to modify the action selection strategy of the Actor module. In order to formulate the process of EDTG, we define some parameters and functions, i is the service index, I is the total number of steps, s_i represents the state of the network. The strategy function $\pi(a_i|s_i;\theta)$ with parameter θ_a in the actor module and value function $V(s_{i,\theta_v})$ with parameter θ_v are used by EDTG. U is a sign of network update, E is a sign of completion of all services grooming. Algorithm 1 describes the detailed procedures of EDTG.

State: The extracted 1×1×1000 features are used as the state of EDTG. We divide the NSFNET into five parts and initialize each part. There are no services in the network, and all resources are available. Perceiving the state of the network is the first step of EDTG. We utilize the method mentioned in subsection IV-A to convert the five divided network and current service i routing path into six images as the network state s_i. There are two major advantages of adopting the images as the state of the network. One is that s_i can comprehensively contain the information in the network, and the other is that we can more easily extract features from s_i.

Then a lightweight convolutional neural network named MobilenetV3 is utilized to extract the features, including information on nodes, links, ports, transponders, regenerators, etc. As shown in Fig. 4, the input is six images of 224×224×3, and has not been treated as an image. The output is the features of 1×1×1000. In this process, MobilenetV3

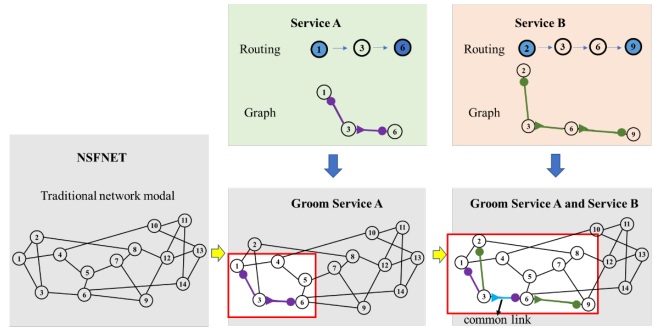


Fig. 5: Resource allocation during the grooming process.

decomposes standard convolutional layers into the form of deep convolution and points convolution, which dramatically improves the running speed. In addition, it is necessary to remove the activation function of the last layer, which is because the activation function after dimensionality reduction will destroy the extracted features. MobilenetV3 adopts the H-Swish activation function instead of ReLU6. As shown in formula 9, where x represents the input of the active layer. H-swish can effectively improve network accuracy.

$$H - Swish[x] = x\frac{ReLU6(x+3)}{6} \tag{9}$$

Action: After extracting features from G_1, G_2, G_3, G_4, G_5 and S_r, EDTG observes the traffic of the current service i and the network state s_i, then takes an action a_i according to strategy $\pi(a_i|s_i)$ to groom the service into G_1 or G_2 or G_3 or G_4 or G_5 . It calculates the probability distribution $P = [p_1, p_2, p_3, p_4, p_5]$ of various actions through the softmax function, where $p_1 + p_2 + p_3 + p_4 + p_5 = 1$, the various actions represent which part of the resources of the five parts are occupied by the current service. Finally, EDTG selects specific actions according to the probability distribution. The essence of the DRL algorithm is how to make an action, and the action of EDTG is to determine which part of the current service should be groomed by observing the network state.

As shown in Fig. 5, there are two services A and B.When service A that needs a unit of resources is successfully groomed, the color states of the corresponding links, ports, transponders and regenerators change to purple, and the new network state is shown in the image. When service B that needs two units of resources arrives, and it has a common link with service A. If EDTG makes an action to groom service B to the same part as service A, and the new network state is shown in the image. Then the energy consumption can be reduced by serving the requests with a common link.

Reward: After the service is groomed completely, the state of the network will become s_{i+1} by adding ports, transponders, and regenerators. In the meantime, those six images will also adjust accordingly. Then EDTG will calculate a reward or a punishment value r_i by judging the grooming effect to get better grooming strategies. If the service occupies fewer resources and consumes less energy after traffic grooming, EDTG will get a larger reward. If the service cannot be groomed, EDTG will get a punishment value. The reward and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3063471, IEEE Internet of Things Journal

TABLE II: Rewards and Punishments

Condition	Action							Reward
	Blocking			No Blocking				
	Active Blocking	Passive Blocking	Switch to First-Fit	Add Port	Add Transponder	Add Regenerator	Grooming	
Unavailable Resource	✓							-3
		✓						-4
Available Resource	✓							-5
			✓					-7
							✓	3.5
								3
						✓	✓	2.5
					✓		✓	2
						✓		1.5
				✓	✓			1
				✓	✓	✓	✓	0.5
				✓	✓	✓		0

punishment mechanism affects the optimization direction of the grooming strategy, and each value of which is inappropriate will guide the grooming strategy to update in the incorrect direction. We determine the reward and punishment values shown in Table II through a large number of experiments. We divide the different occupancy states of resources and the results of the grooming into 12 categories. When a service arrives, the resources required by the service may or may not exist in the network at the moment. If there is no available resource, it should be actively blocked. However, this service will be passively blocked when the grooming strategy insists on grooming the service to a wavelength without resources. In addition, if there are not resources required by the service in the wavelength I, but the service is groomed to wavelength I. In order to reduce the blocking rate, we have added an option to utilize the First-Fit strategy to groom the blocked service to wavelength with resources. This will greatly increase energy consumption, which is contrary to our original intention. Therefore, we increase the punishment for this situation. If the necessary resources required for this service are available, the service should be groomed according to the required resources of ports, transponders, and regenerators. In general, the reward value is inversely proportional to the number of ports, transponders and regenerators added. Then a reward or a punishment value shall be got.

Whenever five services are groomed successfully, U is activated. The states, actions, and reward values are used to update the EDTG. It is worth noting that we adopt the same one ANN as the actor module and critic module, which reduces the complexity of the EDTG network structure. This ANN network undertakes the tasks of the actor module and the critic module, and gets two losses respectively. We calculate the total loss to update the ANN network. Therefore, it can calculate the total loss of the ANN according to formula 10, 11 and 12,

$$l_v = \frac{1}{n}\sum_{i=1}^{N}(R_i - V(s,\theta))^2, \tag{10}$$

$$l_a = \frac{1}{n}\sum_{i=1}^{N}(R_i - V(s,\theta)) \times log(\pi(a|s),\theta), \tag{11}$$

$$l_t = l_v \times c_v + l_a \times c_a + e \times c_e, \tag{12}$$

where l_a represents the mean square error of the value function and the total reward, l_v represents the cross-entropy of the strategy function and the difference between the value function and the total reward, and l_t represents the total loss. c_v, c_a,c_e are the coefficients of l_v, l_a and e. R_i represents the reward. Then EDTG updates the ANN parameter θ through the gradient descent method. Entropy (e) is introduced to express the difference in action selection probability. When the system learns policies that can groom all services effectively, e will converge. When all services are groomed successfully, the system activates E and resets the network.

V. SIMULATION AND RESULTS ANALYSIS

In this section, we evaluate the performance of EDTG with extensive simulations using 14-node and 21-link NSFNET topology, as shown in Fig. 6. NSFNET has 50 spectrum slots on each physical link. We randomly generate 100 services in advance, and the source node and destination node of each service are randomly selected from all nodes. The traffic requirements of each service are uniformly distributed within [10,100] Gbps. The simulation parameters are shown in Table III. We calculate $K = 3$ candidate paths for each service and set $W = 8$ processes to train at the same time. For the training of EDTG, we set the parameters α and ϵ of RMSprop optimizer, and the base learning rate as 0.99, $1e-5$, and $7e-6$. The total number of training steps is $10e6$, the number of forward steps in A2C is 5. We test different CNN such as Alexnet, Fcnet, MobilenetV2, and MobilenetV3, and we finally find that MobilenetV3 is the best. The server we use to run the simulation is configured with Intel Core i9-9900K CPU, 32GB memory, and Nvidia GTX 1080Ti graphics card.

We analyze the performance of AC, A2C, and A3C algorithms in traffic grooming. Then we conduct experiments on the following two well-performed grooming algorithms to evaluate EDTG.

State-aware Modification Grooming Algorithm (SGA) [14]: SGA is a heuristic algorithm, and it grooms the services with the same source node and destination node together to occupy fewer resources.

Deep Reinforcement Learning Algorithm (DRL) [19]: DRL is an intelligent algorithm for distributing services, and it uses a matrix containing network-specific information as

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3063471, IEEE Internet of Things Journal

TABLE III: Parameters description

Parameters	Definitions	Value
Net	Simulation network	NSFNET
Node	Number of nodes	14
Link	Number of links	21
Wavelength	Number of network divide	5
Spectrum Resources	Total resources of the network	50
K	Number of service routing	3
W	Number of multiple processes	8
α	RMSprop optimizer alpha	0.99
ϵ	RMSprop optimizer epsilon	1e-5
learning rate	Base learning rate value	7e-6
steps	Number of training steps	10e6
num-steps	Number of forward steps in A2C	5

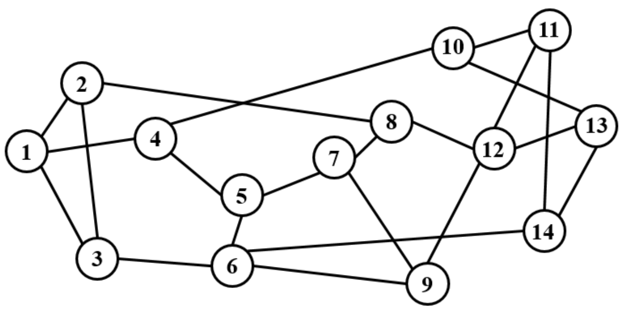


Fig. 6: NSFNET topology.

a feature set, and then distributes services to reduce energy consumption.

We aim to reduce the energy consumption of various elements in EON, such as ports, transponders, and regenerators through traffic grooming. Among them, SGA judges whether the current service is the same with the source and destination nodes of the service previously allocated. If they are same, they will be groomed. If they are different, only the available resources will be allocated. EDTG identifies the links that need to be occupied according to the converted images and whether it should be unblocked. Therefore, we ignore the minor energy consumption and only focus on the energy consumption of the port, transponder, and regenerator in the network.

A. Comparison of Different Traffic Grooming Algorithms

Figure 7 illustrates the energy consumption of different update algorithms corresponding to AC, A2C, and A3C algorithms. We can observe that the convergence values of each algorithm are almost indistinguishable, and this is because the framework of these three algorithms is the Actor-Critic mechanism. There is little difference in the final convergence of energy consumption in our work. The A2C algorithm uses the advantage function instead of the original reward in the AC algorithm to measure the quality of the selected action and the average value of all actions. Therefore, the variance of A2C results is smaller. A3C adds an asynchronous update method on the basis of A2C. It is divided into multiple worker networks and a global network. Each worker is an independent update network and obtains parameters from the global network and uses its own gradient to update the parameters of the global network. In fact, every worker is an A2C, the update time of each round of A2C and A3C with the same number of processes is the same, but we can observe that the A3C algorithm converges 2000 iterations earlier than A2C algorithm from Fig. 7. So A3C only speeds up the convergence speed, and there is no significant reduction in energy consumption. However, we start eight A2C processes, train at the same time, calculate the average loss to update the same network, which also speeds up the process of convergence, but it is not as complicated as the asynchronous update of A3C. Based on the above reasons, EDTG adopts the A2C framework.

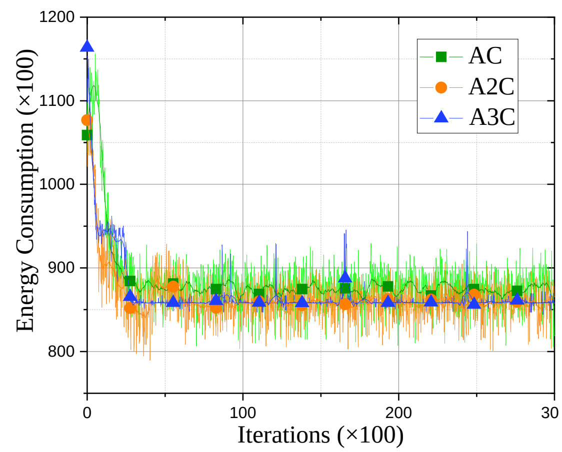


Fig. 7: Energy consumption of different AC frameworks.

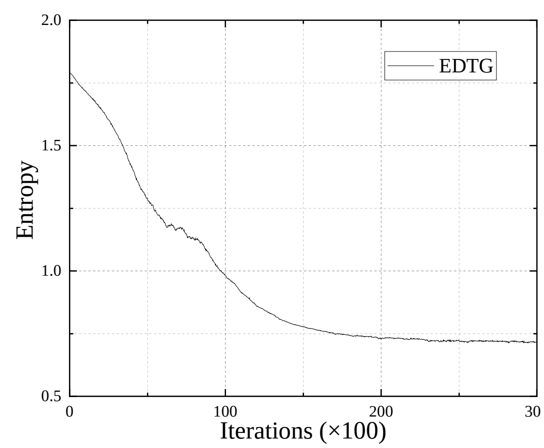


Fig. 8: The entropy of EDTG.

Figure 8 illustrates the entropy of EDTG. The entropy value represents the probability distribution of action selection. We can observe that the entropy value is inversely proportional to the number of iterations, and it starts to converge after 20000 iterations. As the entropy value changes smaller, the EDTG's selection action strategy changes more stable. When the entropy value is converged, it means that with the continuous learning of the network, EDTG has learned a stable strategy and it can groom all services in an energy-efficient manner.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3063471, IEEE Internet of Things Journal

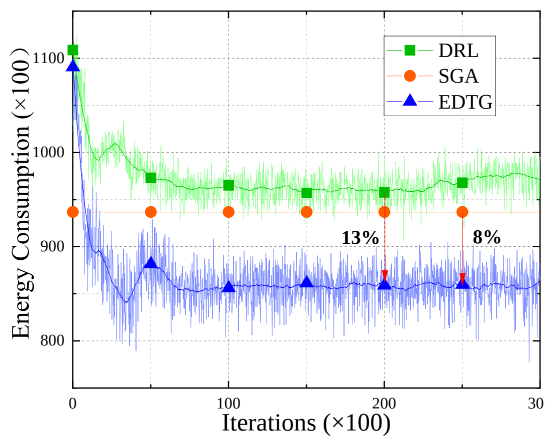


Fig. 9: Energy consumption of different algorithms.

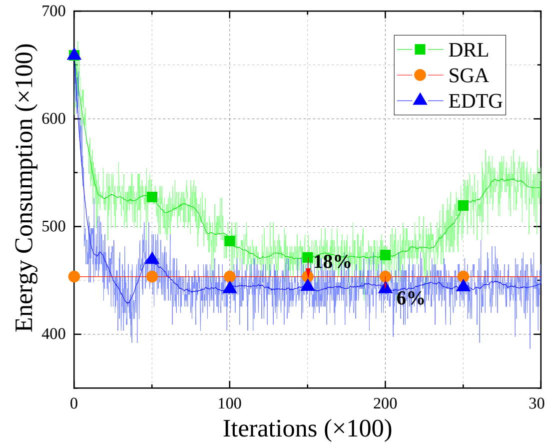


Fig. 11: Energy consumption of ports.

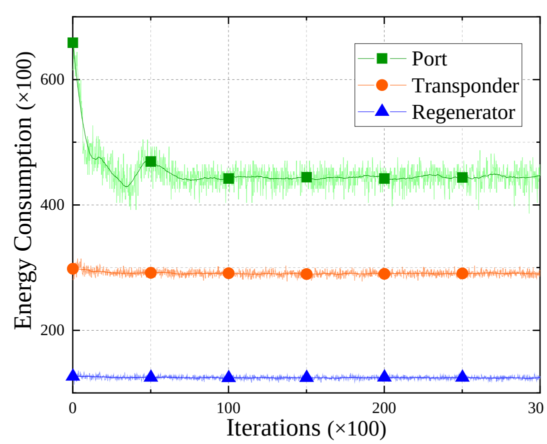


Fig. 10: Energy consumption of different elements.

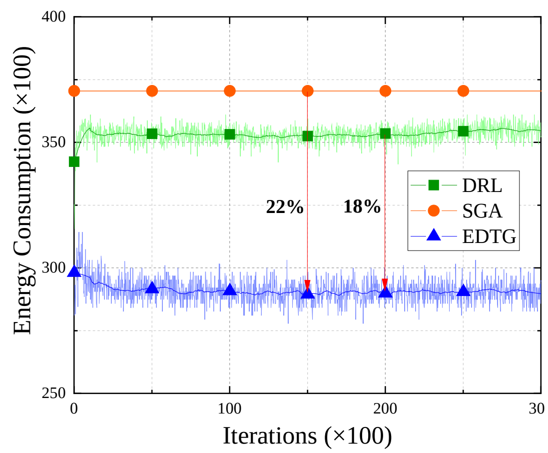


Fig. 12: Energy consumption of transponders.

Figure 9 shows the energy consumption of different traffic grooming algorithms. The EDTG achieves 13% and 8% improvement on energy consumption compared with the DRL algorithm and SGA algorithm. The reason is that DRL only allocates services to the part where resources are available, but it cannot aggregate mice traffic. SGA only grooms services with the same source node and destination node, and it wastes a lot of fragmented resources. Compared with these two algorithms, after thousands of training iterations, EDTG can groom all services with the same link to the appropriate part by constantly updating the reinforcement learning strategy so that fewer elements are occupied in the whole network, and the energy consumption decreases rapidly.

B. Energy consumption of Different Elements

Figure 10 illustrates the energy consumption of ports, transponders, and regenerators in the EDTG algorithm. Fig. 11, Fig. 12, and Fig. 13 compare the variation of energy consumption of ports, transponders, and regenerators with three algorithms DRL, SGA, and EDTG, respectively. Compared with DRL algorithm and SGA algorithm, EDTG achieves 18% and 6% improvement on energy consumption of ports, and achieves 18% and 22% improvement on energy consumption of transponders, and achieves 13% and 22% improvement on energy consumption of regenerators. Due to imperfect parameter settings, the energy consumption convergence trend of the port obtained by the DRL algorithm in Fig. 11 is not good. The simulation results show that the energy consumption obtained by the EDTG algorithm is lower and can be reduced significantly. This is because EDTG can perceive complete information in the network from the converted images and learn an intelligent groom strategy.

C. Energy comparison of Different learning rates

Figure 14 depicts the energy consumption with different l_r. EDTG needs to continuously optimize the strategic network for grooming services to ensure that each device can support more services. When the learning rate gets lower, the convergence speed of the loss function will be slower. However, if the learning rate is too high, the gradient explosion is likely to occur, the loss of vibration amplitude is large, and the model is difficult to converge. From Fig. 14, we can clearly observe that the learning rate is 7e-6 and 1e-6. When the learning rate is 7e-6, the energy consumption converges faster, and the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3063471, IEEE Internet of Things Journal

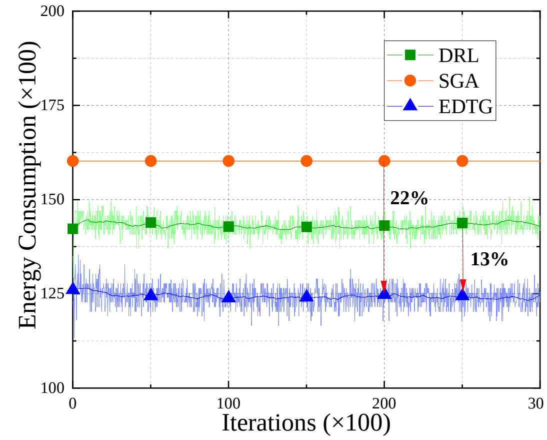


Fig. 13: Energy consumption of regenerators.

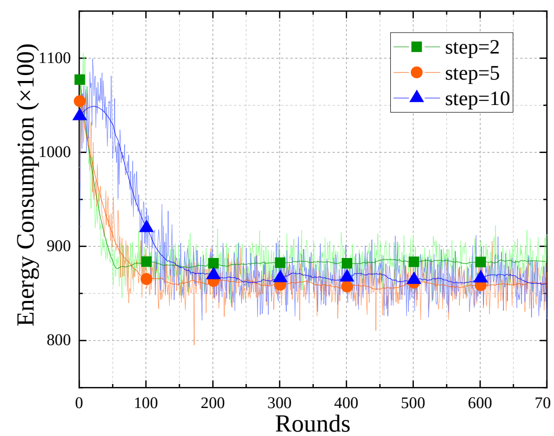


Fig. 15: Energy consumption with different steps.

vibration amplitude is larger, which meets the characteristics of the learning rate in deep learning. But with the two learning rates, the energy consumption convergence effect is the same. Thus, l_r is set to be $7e-6$ in the experiment.

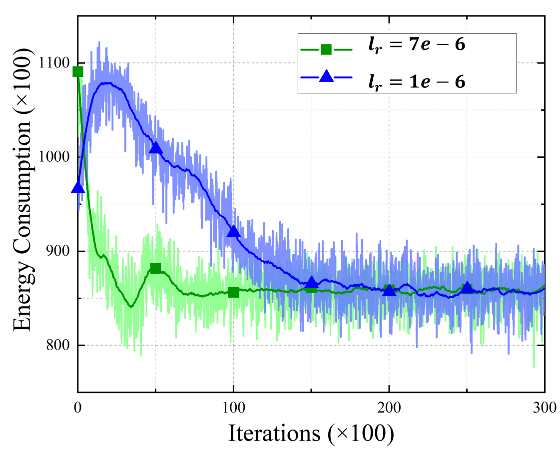


Fig. 14: Energy consumption with different l_r.

D. Energy comparison of Different steps

Figure 15 depicts the energy consumption curves corresponding to different update steps. The update step setting of the network in the learning process is essential. Different update steps mean that the network weight of each update is different, which will affect the update of network parameters. A smaller step size will reach the local optimum, and a larger step size will result in slower convergence. For the convenience of observation, the horizontal ordinate represents the number of update rounds, and each round means all the services are groomed once, and the ordinate represents the energy consumption of the entire network. The green, orange, and blue lines represent step sizes of 2, 5, and 10, respectively. We can observe that the convergence rate becomes slower as the step size increases. The effect of step sizes 5 and 10 is the same, and the energy consumption of a length of 2 is reduced by about 3%, so the step size of 5 is more appropriate.

VI. Conclusion and Future Work

In this paper, we mainly study the traffic grooming problem in elastic optical networks for Cloud-Fog Computing. We convert the topological network into several images by converting the network elements into colorful solid circles, triangles, and line segments, making it simple to extract features. We design an Energy-efficient Deep Reinforced Traffic Grooming algorithm to groom services with minimizing energy consumption. Simulation results show that, compared with two well-performed traffic grooming algorithms, our algorithm can reduce energy consumption significantly.

The design and implementation of EDTG, a traffic grooming algorithm based on deep reinforcement learning, involve many theories, methods, and technologies. EDTG still has some new problems to be solved, which need to be continuously accumulated and improved further. Further research is needed in the following aspects. EDTG converts the whole network state into several colorful images, which is bound to contain redundant information. In the future, we will consider simplifying the network and only convert the network structure composed of service-related nodes and links. In addition, Graph Neural Network (GNN) has a powerful modeling function for the dependence between the points of the graph, which has made breakthrough progress in the research field related to graph analysis. In the future, we can use GNN to directly extract the graph structure features of associated nodes instead of the complex information of the whole network.

References

[1] J. Doyle, V. Giotsas, M. A. Anam, and Y. Andreopoulos, “Dithen: A computation-as-a-service cloud platform for large-scale multimedia processing,” IEEE Transactions on Cloud Computing, vol. 7, no. 2, pp. 509-523, 2019.

[2] H. Yang, Y. Liang, J. Yuan, Q. Yao, A. Yu, and J. Zhang, “Distributed Blockchain-based Trusted Multi-domain Collaboration for Mobile Edge Computing in 5G and beyond,” IEEE Transactions on Industrial Informatics, pp.99:1-1, 2020

[3] Z. Ning, X. Kong, F. Xia, W. Hou, and X. Wang, "Green and sustainable cloud of things: Enabling collaborative edge computing," IEEE Communications Magazine, vol. 57, no. 1, pp. 72-78, 2019.

[4] H. Yang, J. Yuan, H. Yao, Q. Yao, A. Yu, and J. Zhang, "Blockchain-Based Hierarchical Trust Networking for Joint Cloud," IEEE Internet of Things Journal, vol. 7, no. 3, pp. 1667-1677, 2020.

[5] J. Yuan, R. Zhu, et al., "A Spectrum Assignment Algorithm in Elastic Optical Network With Minimum Sum of Weighted Resource Reductions in all Associated Paths," IEEE Journal of Lightwave Technology, vol. 37, no. 21, pp. 5583-5592, 1 Nov.1, 2019.

[6] R. Zhu, S. Li, P. Wang, and J. Yuan, "Time and spectrum fragmentation-aware virtual optical network embedding in elastic optical networks," Optical Fiber Technology, vol. 54, pp. 102117, 2020.

[7] A. Celik, A. AlGhadhban, B. Shihada, and M. Alouini, "Design and provision of traffic grooming for optical wireless data center networks," IEEE Transactions on Communications, vol. 67, no. 3, pp. 2245–2259, 2019.

[8] R. Zhu, S. Li, P. Wang, Y. Tan, and J. Yuan, "Gradual Migration of Co-Existing Fixed/Flexible Optical Networks for Cloud-Fog Computing," IEEE Access, vol. 8, pp. 50637-50647, 2020.

[9] R. Zhu, A. Samuel, S. Li, P. Wang, B. Kham Oun, L. Li, P. Lv, M. Xu, and S. Yu, "Protected Resource Allocation in Space Division Multiplexing-Elastic Optical Networks (SDM-EONs) with Fluctuating traffic," Journal of Network and Computer Applications. vol. 174, 2021.

[10] R. Zhu, A. Samuel, P. Wang, S. Li, L. Li, P. Lv, and M. Xu, "Survival Multipath Energy-aware Resource Allocation in SDM-EONs during Fluctuating Traffic," Journal of Lightwave Technology, pp.1-1, 2020.

[11] W. Jin, R. Gu, Y. Tan, and Y. Ji, "Proactive grooming with delay optimization in sliceable elastic optical network," IEEE Access, vol. 7, pp. 105030-105040, 2019.

[12] E. E. Moghaddam, H. Beyranvand, and J. A. Salehi, "Routing, spectrum and modulation level assignment, and scheduling in survivable elastic optical networks supporting multi-class traffic," IEEE Journal of Lightwave Technology, vol. 36, pp. 5451-5461, 2018.

[13] B. Gangopadhyay, J. Pedro, and S. Spaelter, "5g-ready multi-failure resilient and cost-effective transport networks," Journal of Lightwave Technology, vol. 37, no. 16, pp. 4062-4072, 2019.

[14] Z. Zhong, N. Hua, M. Tornatore, Y. Li, H. Liu, C. Ma, Y. Li, X. Zheng, and B. Mukherjee, "Energy efficiency and blocking reduction for tidal traffic via stateful grooming in ip-over-optical networks," IEEE/OSA Journal of Optical Communications and Networking, vol. 8, no. 3, pp. 175-189, 2016.

[15] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "Applications of deep reinforcement learning in communications and networking: A survey," IEEE Communications Surveys and Tutorials, pp. 1-1, 2019.

[16] F. Wang, C. Zhang, F. wang, J. Liu, Y. Zhu, H. Pang, and L. Sun, "Intelligent edge-assisted crowdcast with deep reinforcement learning for personalized QoE," IEEE INFOCOM 2019 - IEEE Conference on Computer Communications, pp. 910-918, 2019.

[17] X. Chen, R. Proietti, H. Lu, A. Castro, and S. J. B. Yoo, "Knowledge based autonomous service provisioning in multi-domain elastic optical networks," IEEE Communications Magazine, vol. 56, no. 8, pp. 152-158, 2018.

[18] J. Suz-Varela, A. Mestres, J. Yu, L. Kuang, H. Feng, P. Barlet-Ros, and A. Cabellos-Aparicio, "Routing based on deep reinforcement learning in optical transport networks," Optical Fiber Communications Conference and Exhibition (OFC), pp. 1-3, 2019.

[19] X. Chen, J. Guo, Z. Zhu, R. Proietti, A. Castro, and S. J. B. Yoo, "Deeprmsa: A deep-reinforcement-learning routing, modulation and spectrum assignment agent for elastic optical networks," Optical Fiber Communications Conference and Exposition (OFC), pp. 1-3, 2018.

[20] Y. Zhang, P. Chowdhury, M. Tornatore, and B. Mukherjee, "Energy efficiency in telecom optical networks," IEEE Communications Surveys Tutorials, vol. 12, no. 4, pp. 441–458, 2010.

[21] A. Fallahpour, H. Beyranvand, and J. A. Salehi, "Energy-efficient manycast routing and spectrum assignment in elastic optical networks for cloud computing environment," IEEE Journal of Lightwave Technology, vol. 33, no. 19, pp. 4008-4018, 2015.

[22] Y. Zhao, B. Chen, J. Zhang, and X. Wang, "Energy efficiency with sliceable multi-flow transponders and elastic regenerators in survivable virtual optical networks," IEEE Transactions on Communications, vol. 64, no. 6, pp. 2539-2550, 2016.

[23] R. Alvizu, X. Zhao, G. Maier, Y. Xu, and A. Pattavina, "Energy efficient dynamic optical routing for mobile metro-core networks under tidal traffic patterns," Journal of Lightwave Technology, vol. 35, no. 2, pp. 325-333, 2017.

[24] M. Musa, T. Elgorashi, and J. Elmirghani, "Bounds for energy-efficient survivable ip over wdmnetworks with network coding," IEEE/OSA Journal of Optical Communications and Networking, vol. 10, no. 5, pp. 471-481, 2018.

[25] Y. Xiong, J. Shi, Y. Yang, Y. Lv, and G. N. Rouskas, "Lightpath management in sdn-based elastic optical networks with power consumption considerations," Journal of Lightwave Technology, vol. 36, no. 9, pp. 1650-1660, 2018.

[26] Z. Zhong, N. Hua, M. Tornatore, J. Li, Y. Li, X. Zheng, and B. Mukherjee, "Provisioning short-term traffic fluctuations in elastic optical networks," IEEE/ACM Transactions on Networking, vol. 27, no. 4, pp. 1460-1473, 2019.

[27] J. Zhang, Y. Zhao, X. Yu, J. Zhang, M. Song, Y. Ji, and B. Mukherjee, "Energy-efficient traffic grooming in sliceable-transponder-equipped improver-elastic optical networks [invited]," IEEE/OSA Journal of Optical Communications and Networking, vol. 7, no. 1, pp. A142-A152, 2015.

[28] J. Zhang, Y. Ji, M. Song, Y. Zhao, X. Yu, J. Zhang, and B. Mukherjee, "Dynamic traffic grooming in sliceable bandwidth-variable transponder enabled elastic optical networks," Journal of Lightwave Technology, vol. 33, no. 1, pp. 183-191, 2015.

[29] J. Wu, Z. Ning, and L. Guo, "Energy-efficient survivable grooming in software-defined elastic optical networks," IEEE Access, vol. 5, pp. 6454-6463, 2017.

[30] J. Zhang, Y. Lei, B. Chen, M. Gao, L. Xiang, and Q. Zhang, "Traffic grooming approaches in flexible bandwidth optical networks with distributed data centers," Opto-Electronics and Communications Conference (OECC), pp. 1-2, 2018.

[31] M. Hadi and M. R. Pakravan, "Energy-efficient fast configuration of flexible transponders and grooming switches in ofdm-based elastic optical networks," IEEE/OSA Journal of Optical Communications and Networking, vol. 10, no. 2, pp. 90-103, 2018.

[32] W. Jin, R. Gu, Y. Tan, and Y. Ji, "Proactive grooming with delay optimization in sliceable elastic optical network," IEEE Access, vol. 7, pp. 105030-105040, 2019.

[33] Y. Zhao, B. Yan, D. Liu, Y. He, D. Wang, and J. Zhang, "Soon: self-optimizing optical networks with machine learning," Optics Express, vol. 26, pp. 28713-28726, 2018.

[34] F. Musumeci, C. Rottondi, A. Nag, I. Macaluso, D. Zibar, M. Ruffini, and M. Tornatore, "An overview on application of machine learning techniques in optical networks," IEEE Communications Surveys Tutorials, vol. 21, no. 2, pp. 1383-1408, 2019.

[35] B. Yan, Y. Zhao, Y. Li, X. Yu, J. Zhang, Y. Wang, L. Yan, and S. Rahman, "Actor-critic-based resource allocation for multi-modal optical networks," in 2018 IEEE Globecom Workshops (GC Wkshps), pp. 1-6, 2018.

[36] X. Luo, C. Shi, L. Wang, X. Chen, Y. Li, and T. Yang, "Leveraging double-agent-based deep reinforcement learning to global optimization of elastic optical networks with enhanced survivability," Optics Express, vol. 27, pp. 7896-7911, 2019.

[37] Y. Xiong, Y. Yang, Y. Ye, and G. N. Rouskas, "A machine learning approach to mitigating fragmentation and crosstalk in space division multiplexing elastic optical networks," Optical Fiber Technology, vol. 50, pp. 99-107, 2019.

[38] R. Proietti, X. Chen, K. Zhang, G. Liu, M. Shamsabardeh, A. Castro, L. Velasco, Z. Zhu, and S. J. Ben Yoo, "Experimental demonstration of machine-learning-aided qot estimation in multi-domain elastic optical networks with alien wavelengths," IEEE/OSA Journal of Optical Communications and Networking, vol. 11, pp. A1-A10, 2019.

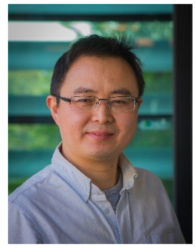
[39] H. Yang, Q. Yao, A. Yu, Y. Lee, and J. Zhang, "Resource assignment based on dynamic fuzzy clustering in elastic optical networks with multi-core fibers," IEEE Transactions on Communications, vol. 67, no. 5, pp. 3457-3469, 2019.

Ruijie Zhu is currently an associate professor in the School of Information Engineering, Zhengzhou University. He is also a member of the Center for interdisciplinary Information Science Research. He received the PhD degree in the State Key Lab of Information Photonics and Optical Communication at Beijing University of Posts and Telecommunications (BUPT), China. He was a visiting student at the University of Texas at Dallas under the supervision of Prof. Jason P. Jue. His research interests include elastic optical networks, network virtualization, and machine learning.

Shihua Li is currently pursuing Ph.D. degree in the School of Information Engineering, Zhengzhou University. He received his Bachelor degree in Electronic Science Computing in Zhengzhou University. His research interests include elastic optical networks, network virtualization, and machine learning.

Peisen Wang is currently pursuing Ph.D. degree in the School of Information Engineering, Zhengzhou University. He received his Bachelor degree in Electronic Science Computing in Henan University. His research interests include elastic optical networks, network virtualization, and machine learning.

Mingliang Xu received the Ph.D. degree from the State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China. He is currently a Professor with the School of Information Engineering, Zhengzhou University, Zhengzhou, China. He has authored more than 60 journal articles and conference papers in these areas, including the ACM Transaction on Graphics (TOG), the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (TPAMI), IEEE TRANSACTIONS ON IMAGE PROCESSING (TIP), IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (TCSVT), the ACM Special Inter-est Groupon Computer Graphics (ACM SIGGRAPH) Asia/ACM Multimedia Conference (MM), and the IEEE International Conference on Computer Vision (ICCV). His current research interests include computer graphics, multimedia, and artificial intelligence.



Shui Yu is currently a full Professor of School of Computer Science, University of Technology Sydney, Australia. Dr Yu's research interest includes Security and Privacy, Networking, Big Data, and Mathematical Modelling. He has published two monographs and edited two books, more than 200 technical papers, including top journals and top conferences, such as IEEE TPDS, TC, TIFS, TMC, TKDE, TETC, ToN, and INFOCOM. Dr Yu initiated the research field of networking for big data in 2013. His h-index is 49. Dr Yu actively serves his research communities in various roles. He is currently serving the editorial boards of IEEE Communications Surveys and Tutorials, IEEE Communications Magazine, IEEE Internet of Things Journal, IEEE Communications Letters, IEEE Access, and IEEE Transactions on Computational Social Systems. He has served many international conferences as a member of organizing committee, such as publication chair for IEEE Globecom 2015, IEEE INFOCOM 2016 and 2017, TPC chair for IEEE BigDataService 2015, and general chair for ACSW 2017. Dr Yu is a final voting member for a few NSF China programs in 2017. He is a Senior Member of IEEE, a member of AAAS and ACM, the Vice Chair of Technical Commuittee on Big Data of IEEE Communication Society, and a Distinguished Lecturer of IEEE Communication Society.