JOURNAL OF
Optical Communications
and Networking

Reconfiguring multicast sessions in elastic optical networks adaptively with graph-aware deep reinforcement learning

XIAOJIAN TIAN,[1] BAOJIA LI,[1] RENTAO GU,[2] AND ZUQING ZHU[1,*]

[1]School of Information Science and Technology, University of Science and Technology of China, Hefei, Anhui 230027, China
[2]School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China
*Corresponding author: zqzhu@ieee.org

Received 11 May 2021; revised 26 June 2021; accepted 7 July 2021; published 29 July 2021 (Doc. ID 431225)

With the fast deployment of datacenters (DCs), bandwidth-intensive multicast services are becoming more and more popular in metro and wide-area networks, to support dynamic applications such as DC synchronization and backup. Hence, this work studies the problem of how to formulate and reconfigure multicast sessions in an elastic optical network (EON) dynamically. We propose a deep reinforcement learning (DRL) model based on graph neural networks to solve the sub-problem of multicast session selection in a more universal and adaptive manner. The DRL model abstracts topology information of the EON and the current provisioning scheme of a multicast session as graph-structured data, and analyzes the data to intelligently determine whether the session should be selected for reconfiguration. We evaluate our proposal with extensive simulations that consider different EON topologies, and the results confirm its effectiveness and universality. Specifically, the results show that it can balance the trade-off between the number of reconfiguration operations and blocking performance much better than existing algorithms, and the DRL model trained in one EON topology can easily adapt to solve the problem of dynamic multicast session reconfiguration in other topologies, without being redesigned or retrained.

https://doi.org/10.1364/JOCN.431225

1. INTRODUCTION

In recent years, the increase in cloud services and live video streaming has made multicast services more and more popular on the Internet [1]. This trend has become even more remarkable since 2020, because of the surge in demands for video conferencing and online classroom services during the pandemic. Due to the fast deployment of datacenters (DCs) all over the world, the popularity of multicast services can also be seen in metro and wide-area networks [2], especially for bandwidth-intensive applications such as DC synchronization and backup, distributed scientific computing, etc. [3]. This has put great pressure on DC interconnects (DCIs) and made multicast provisioning in DCIs an attractive research topic.

With the tremendous bandwidth in each optical fiber, optical networking plays an important role in DCIs, and a recent study [4] even suggested that an optical-circuit-switched architecture could be more scalable and cost effective for regional DCIs than a natural packet-switched network. More promisingly, advances on flexible-grid elastic optical networks (EONs) can further improve the performance of optical switching on spectrum efficiency, adaptivity, and application awareness [5–7]. Note that, for bandwidth-intensive and long-lasting applications (e.g., DC backup), realizing multicasting directly in the optical domain has benefits such as less bandwidth/protocol overhead and ease of obtaining large throughput [8]. The agility of EONs would further promote these benefits, which motivated researchers to study how to provision multicast services in EONs and propose various algorithms [9–14].

The semi-permanent optical layer in telecommunication networks might not adapt to dynamic applications and traffic in DCIs [15]. Therefore, a dynamic optical layer with fast reconfiguration speed is desired. For instance, the standardization effort in [16] suggested that to properly support inter-DC communications, a dynamic optical network should be reconfigurable within a few milliseconds. Following this trend, researchers have considered different dynamic operation scenarios for EONs, e.g., the reconfiguration to accommodate time-varying unicast traffic [17,18], spectrum defragmentation [19], lightpath restoration [20], and spectrum retuning for bulk data transfers [21]. The dynamic nature of multicast services in DCIs determines that each multicast session might also need to be updated consistently to maintain the optimality of its service provisioning scheme (i.e., the one that consumes the

fewest spectrum resources) [22]. For example, during a one-to-many DC backup, each destination DC joins the multicast session when the data of interest start to be transferred, and it will leave the session when its data transfer is done.

The problem of how to formulate and reconfigure multicast sessions in EONs dynamically was previously studied in [22]. Specifically, the authors divided the problem into two sub-problems, i.e., session selection and session reconfiguration, and designed algorithms to solve them. The session selection algorithm finds the most "critical" multicast sessions whose provisioning schemes waste the most spectrum resources when compared with the optimal ones (i.e., off their optima the most), to reconfigure. After the sessions have been selected, they can be reconfigured with either full or partial rearrangements in the session reconfiguration, to free up unnecessary spectrum usages. Note that the reconfiguration of multicast sessions should be evaluated from two perspectives, i.e., the number of reconfiguration operations and overall blocking probability of multicast sessions. Specifically, by invoking more reconfiguration operations, we generally can readjust the provisioning schemes of multicast sessions better to save more spectrum resources, and thus a lower blocking probability will be obtained in the future. Hence, to maximize the efficiency of reconfiguration, we should use the fewest reconfiguration operations to achieve the largest reduction on blocking probability. However, to the best of our knowledge, how to optimize this trade-off has not been fully explored yet.

We can see that in the reconfiguration of multicast sessions, the sub-problem of session selection is more relevant to the aforementioned trade-off. Nevertheless, the heuristic approaches developed in [22] (i.e., the D-/Q-value-based selection strategies) cannot universally adapt to dynamic EON environments, and the problem of how to select between them and determine their key parameters can be tackled only in an empirical manner. This motivates us to revisit the sub-problem in this work. Note that deep reinforcement learning (DRL) can obtain statistically optimal solutions for complex and time-varying problems without explicit programming [23]. Hence, we try to replace the heuristic approaches for session selection with a DRL-based algorithm, and expect that it can balance the trade-off between the number of reconfiguration operations and blocking probability better.

Note that to select multicast sessions in an EON to reconfigure, we need to process data in graph structure, which can hardly be handled well by neural networks (NNs) in linear structures. This is because certain important information buried in the graph-structured data can be lost, and DRL models with NNs in linear structures need to be redesigned and retrained when the EON's topology changes. Fortunately, graph NNs (GNNs) [24] can fulfill the requirements much better, as they can operate directly on graph-structured data to understand the complex relations in it for applications related to networks [25].

In this work, we propose a DRL model based on GNNs to solve the sub-problem of multicast session selection in a more universal and adaptive way. The DRL model takes the topology information of the EON and the current provisioning scheme of a multicast session as the input, abstracts them as graph-structured data, and analyzes the data to intelligently determine whether the multicast session should be selected for reconfiguration. We evaluate the proposed graph-aware DRL model with extensive simulations that consider different EON topologies. The simulation results confirm the effectiveness and universality of our proposal, and show that it can balance the trade-off between the number of reconfiguration operations and blocking probability much better than existing heuristic approaches, without empirical parameter adjustments.

The rest of the paper is organized as follows. Section 2 briefly surveys related work. We describe the network model and operation principle of the dynamic reconfiguration of multicast sessions in EONs in Section 3. The graph-aware DRL model for session selection is designed in Section 4, and we discuss its performance evaluations in Section 5. Finally, Section 6 summarizes the paper.

2. RELATED WORK

Multicasting in the optical domain has been studied since the inception of wavelength-division-multiplexing (WDM) networks, and Sahasrabuddhe and Mukherjee [8] first came up with the concept of a light-tree for it. One can refer to the survey in [26] for a complete review of optical multicasting in fixed-grid WDM networks. The proposals of flexible-grid EONs [5–7] were considered to leverage bandwidth-variable transponders (BV-Ts) and bandwidth-variable switches (BV-WSSs) to manage the spectrum allocation in the optical layer with a fine granularity of 12.5 GHz or even less, and thus make the optical layer more spectrum efficient and adaptive. Flexible spectrum management in EONs transforms the well-known routing and wavelength assignment (RWA) problem in WDM networks into a more complex one, i.e., routing and spectrum assignment (RSA) [27]. Hence, the provisioning of optical multicasting should be revisited for EONs.

In [9], the authors proposed two multicast-capable RSA (MC-RSA) algorithms for EONs and analyzed their performance. Liu *et al.* [12] improved the performance of MC-RSA by leveraging layered auxiliary graphs. Nevertheless, these two studies did not consider the adaptive modulation-level selection in EONs. The multicast provisioning with impairment-aware routing, modulation, and spectrum assignment (RMSA) was addressed in [11], where the authors designed two integer linear programming (ILP) models and a few heuristics. Then, the MC-RMSA algorithms to support distance-adaptive transmissions were developed in [13]. The authors of [14] introduced a light-forest to further improve the performance of MC-RMSA and proposed a polynomial-time approximation algorithm. In addition to algorithmic contributions, researchers have also leveraged the idea of software-defined EON (SD-EON) to experimentally demonstrate the control plane operations for optical multicasting in [28].

However, the aforementioned studies all assumed that optical switches are MC (i.e., supporting light splitting). Note that MC optical switches usually have complicated architectures and thus can be relatively expensive [26]. Therefore, it might not be cost effective to build an EON with them, since the majority of the communications in the EON will

still be for unicast services. This issue can be addressed by realizing multicast with multicast-incapable (MI) optical switches, i.e., establishing a logic light-tree for each multicast session with multiple unicast lightpaths [10]. Specifically, the study in [10] proposed a spectrum-flexible member-only relay (OL-M-SFMOR) scheme for this purpose.

Another benefit of realizing multicasting with MI optical switches is that multicast sessions can be reconfigured in a localized and easier manner. This is because multicasting with MC optical switches has the restriction that all the branches of a light-tree should have the same spectrum assignment, while this is not required by the OL-M-SFMOR scheme [10]. In [22], the authors studied how to formulate and reconfigure multicast sessions dynamically, assuming that OL-M-SFMOR is used in an EON built with MI optical switches. Nevertheless, as we have already explained, the algorithms proposed in [22] for multicast session selection still have a few drawbacks, which motivates us to revisit the sub-problem in this work and try to solve it better with a novel graph-aware DRL model.

Previously, Li *et al.* [29] designed a deep NN (DNN) to predict the performance of multicast light-trees. However, the DNN still uses a linear architecture, which is not good at processing graph-structured data, and the topic was not on a multicast reconfiguration. Due to its promising performance on processing graph-structured data, GNN has attracted great attention [25], especially for the complex optimizations in networks [30,31].

3. PROBLEM DESCRIPTION

In this section, we explain the network model and operation principle of the dynamic multicast reconfiguration in EONs.

A. Network Model

The topology of an EON for DCI is modeled as a directed graph $G(V, E)$, where V and E are the sets of DCs and fiber links, respectively. Here, similar to the case in [22], we assume that the EON is built with MI optical switches. On each link $e \in E$, there are F frequency slots (FSs), each of which has a bandwidth of 12.5 GHz. The BV-Ts that terminate each fiber link are assumed to be sliceable [32], which means that as long as there are sufficient spectrum resources on a link, its BV-Ts can always be sliced to facilitate the requested lightpath transmissions.

We model each multicast session as $MR(s, D, b, t)$, where $s \in V$ denotes the source, D represents the set of destinations, b is the bandwidth demand in Gbps, and t stands for its lifetime. In this work, we consider a dynamic EON environment in which each multicast session $MR(s, D, b, t)$ can come and leave on-the-fly, and during its lifetime t, the DCs in D can change over time too. Hence, when a new multicast session first comes in, we leverage the OL-M-SFMOR scheme in [10] to set up several lightpaths for establishing a logic light-tree, such that each destination in D can receive b Gbps from the source s through one or more lightpaths. Here, each lightpath for serving the multicast session can start and end only at its member nodes (i.e., those in $s \cup D$) for saving BV-Ts, according to the principle of OL-M-SFMOR [10]. As the optical signal is transmitted only all-optically on each lightpath, the RSA schemes of different lightpaths in the logic light-tree are independent, i.e., the spectrum assignments on different branches of the light-tree can be different.

After the initial provisioning of the multicast session, the DCs in D can change over time. Then, when a DC leaves the session or a new DC joins in, the lightpaths in the logic light-tree are updated, still with OL-M-SFMOR. Nevertheless, this might gradually degrade the optimality of the logic light-tree; the RSA schemes of certain lightpaths in it can be sub-optimal and waste spectrum resources. Therefore, we need to reconfigure the multicast session adaptively from time to time.

B. Dynamic Reconfiguration of Multicast Sessions

We use Algorithm 1 to explain the operation principle of dynamic formulation and reconfiguration of multicast sessions [22]. Lines 2–10 explain how to formulate multicast sessions dynamically. Then, the reconfiguration of multicast sessions is triggered periodically to maintain the optimality of the logic light-trees of in-service multicast sessions. Here, we need to solve two sub-problems for the reconfiguration, i.e., session selection (line 12) and session reconfiguration (line 13). The session selection needs to find the most "critical" multicast sessions whose logic light-trees are off their optima the most, to reconfigure. The session reconfiguration rearranges the logic light-trees of the selected sessions to save spectrum resources, which can be done with either full or partial rearrangements [22]. Specifically, the full rearrangement recalculates the logic light-tree of each selected session with OL-M-SFMOR, while the partial rearrangement chooses only certain lightpaths in the logic light-tree of each selected session to reconfigure, according to the average cost of the lightpaths in it. (Here, the cost of a lightpath was defined in [22], which depends on the lightpath's spectrum usage and the number of hops of its routing path.)

Throughout the aforementioned process, we need to balance the trade-off between the number of lightpath reroutings and overall blocking probability of multicast sessions. It can be seen that the sub-problem of session selection is more relevant to

Algorithm 1. Dynamic Provisioning of Multicast Sessions

1. **while** the EON is operational **do**
2. **for** each newly arrived session $MR_i(s, D, b, t)$ **do**
3. try to set up a logic light-tree for it with OL-M-SFMOR
4. **if** the light-tree cannot be established **then**
5. mark MR_i as blocked
6. **for** each existing session $MR_j(s, D, b, t)$ **do**
7. **if** t has been expired **then**
8. remove MR_j and free its resources
9. **if** D has changed **then**
10. update its logic light-tree with OL-M-SFMOR
11. **if** it is time to reconfigure multicast sessions **then**
12. select existing multicast sessions to reconfigure
13. reconfigure the selected multicast sessions

this tradeoff. Hence, in the following, we first review the D-/Q-value-based selection strategies designed in [22], analyze their drawbacks, and then explain the principle of our graph-aware DRL-based selection algorithm.

The D-value of a logic light-tree is actually the hop-count of its longest-destination branch [22]:

$$\mathcal{D}(\mathcal{T}) = \max\left[hops(s \rightarrow d),\ \forall d \in D\right], \tag{1}$$

where $\mathcal{T}$ is the light-tree for multicast session $MR(s, D, b, t)$, and $hops(\cdot)$ returns the hop-count of a routing path. With the definition in Eq. (1), the D-value-based selection (DTS) strategy first calculates the average D-value of all the in-service multicast sessions, and then selects those whose D-values are larger than the average value to reconfigure. As DTS considers only the branch lengths of each logic light-tree but does not address the overall tree structure or the spectrum assignment on the links, it might not always select the most critical sessions to reconfigure.

The Q-value of a logic light-tree considers the overall tree structure and the spectrum assignment on its links [22]:

$$\mathcal{Q}(\mathcal{T}) = \frac{hops(\mathcal{T}^*) \cdot hidx(\mathcal{T}^*)}{hops(\mathcal{T}) \cdot hidx(\mathcal{T})}, \tag{2}$$

where $\mathcal{T}^*$ is the logic light-tree calculated with OL-M-SFMOR based on the current network status, and $hidx(\cdot)$ returns the highest index of the used FSs on a light-tree. With Eq. (2), the Q-value-based selection (QTS) strategy first chooses a threshold $\mathcal{Q}_{lb}$, and then selects those whose Q-values are smaller than $\mathcal{Q}_{lb}$ to reconfigure. Although QTS considers more information of a logic light-tree than DTS, the information is still somehow limited, and the value of $\mathcal{Q}_{lb}$ can be determined only empirically, which is rather difficult in a dynamic EON or for EONs with various topologies.

To address the issues of DTS and QTS, this work proposes to select multicast sessions in a self-adaptive manner with graph-aware DRL. More specifically, the DRL model takes the topology information of the EON and the current provisioning scheme of a multicast session as the input, abstracts them as graph-structured data, and analyzes the data with GNNs to intelligently determine whether the multicast session should be selected for reconfiguration. After offline training, the DRL model is also trained in the online manner to make sure that it can optimize its decision making automatically and adaptively according to the reward feedbacks from a dynamic EON environment, i.e., its effectiveness and universality can be guaranteed without empirical parameter adjustments.

4. GRAPH-AWARE DRL-BASED APPROACH

This section elaborates on our graph-aware DRL model for multicast session selection. Note that to determine whether a multicast session should be selected for reconfiguration or not, we need to process graph-structured data (i.e., the tree topology and the spectrum usages on its links). This task is suitable for GNNs, because NNs in linear structures normally deal only with the data in Euclidean domains well [25].

A. System Architecture

We still assume that the EON for DCI is operated by leveraging SD networking (SDN), which means that the control plane consists of a centralized controller to handle the tasks for network control and management (NC&M). Our graph-aware DRL model obtains the information about the EON and the multicast sessions in it from the controller, and selects the most critical sessions to reconfigure.

Figure 1 explains the operation principle of the graph-aware DRL model, and its work-flow is illustrated with step numbers. Dynamic requests regarding multicast sessions (i.e., new multicast sessions and changes on in-service sessions) are first processed by the request handler, which dispatches them to both the traffic engineering database (TED) and the service provisioning module. As explained in Algorithm 1, the service provisioning module serves the dynamic requests and updates their provisioning results in the TED. Then, the reconfiguration of multicast sessions is triggered periodically, and it starts from TED sending the current network status to the feature engineering module, which abstracts the network status to a state that consists of graph-structured data. The DRL agent uses two GNNs, i.e., the actor GNN (A-GNN) and critic GNN (C-GNN), to analyze the state of each in-service multicast session and selects certain sessions to reconfigure.

Next, the action from the DRL agent (i.e., the selected multicast sessions) is forwarded to the session reconfiguration module, which works with the service provisioning module to reconfigure the selected sessions. After this, the TED sends the new network status to the reward calculation module to obtain the reward of the last action conducted by the DRL agent. Then, we organize the state, action, and reward as a training sample, and store it in the experience buffer. When enough entries of experience have been accumulated, the online training module invokes a training process to update the global

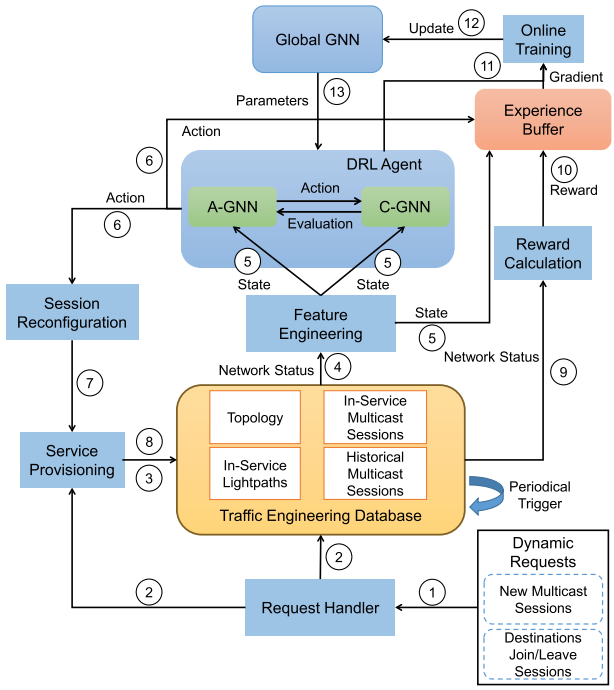

Fig. 1. Architecture and operation principle of our graph-aware DRL model.

GNN, which in turn updates the parameters of the A-GNN and C-GNN in the DRL agent accordingly.

B. Preprocessing of Data

To prepare the input to the graph-aware DRL model, we abstract the topology information of the EON and the current provisioning scheme of a multicast session as graph-structured data $\mathcal{G}(V, \tilde{V}, E, \tilde{E})$, where V and E still represent the sets of DC nodes and fiber links in the EON, respectively, while $\tilde{V}$ and $\tilde{E}$ denote the features of the nodes and links in V and E, respectively, regarding the current provisioning scheme a multicast session $MR(s, D, b, t)$. Specifically, according to the current logic light-tree $\mathcal{T}$ of MR, we classify the nodes in V into five categories, i.e., the source s, destinations in D, intermediate node on $\mathcal{T}$ that used to be a destination, normal intermediate node, and nodes that are not on $\mathcal{T}$. Then, the feature of a node $v \in V$ can be described with a corresponding vector $\tilde{v} \in \tilde{V}$, with one-hot coding. Here, we use five bits to represent the aforementioned node categories. For instance, if we have node $v = s$, its feature vector $\tilde{v}$ should be [1,0,0,0,0], or if node v is a destination, its feature vector $\tilde{v}$ should be [0,1,0,0,0]. On the other hand, the feature of a link $e \in E$ is defined as $\tilde{e} = \frac{f}{F}$, where f is the number of unused FSs on link e, and F is the total number of FSs there.

Here, for simplicity, we do not consider distance-adaptive modulation selection, and assume that all the lightpaths in each logic light-tree use the lowest modulation level (i.e., binary phase shift keying). Note that if we need to consider distance-adaptive modulation selection, the only difference is that we should let our DRL model learn the relation between the transmission distance of a lightpath and the number of FSs that it uses. Hence, when preprocessing the graph-structured data of $\mathcal{G}(V, \tilde{V}, E, \tilde{E})$, we need to include the length of each fiber link as an attribute, modify the feature of each link in E and $\tilde{E}$ accordingly, and redesign the GNNs in the DRL model to accommodate the changes. This will be considered in our future work.

C. Structure of GNNs

We design the GNNs used in our DRL model based on a graph convolutional network (GCN) [33]. The GCN takes the graph-structured data $\mathcal{G}(V, \tilde{V}, E, \tilde{E})$ as the input, and performs two types of operations on the data, i.e., message transfer and information reduction. For the two types of operations, we define two functions as follows. The message function calculates the message to be sent from node v to node u:

$$msg(v, u) = \tilde{v} \cdot \tilde{e}, \quad v, u \in V,\ e = (v, u) \in E, \tag{3}$$

where nodes v and u are connected with a link $e = (v, u)$ in $\mathcal{G}(V, \tilde{V}, E, \tilde{E})$. The reduction function reduces the messages that each node in V receives from its neighbors:

$$rdu(v) = \sum_{\{u:(u,v)\in E\}} msg(u, v), \quad v \in V. \tag{4}$$

Then, we send $\{rdu(v),\ \forall\, v \in V\}$ through a linear network in the GCN to obtain the new feature vector of v, and the transfer function from layer-l to layer-$(l+1)$ is defined as

$$\tilde{v}^{(l+1)} = \sigma(W \cdot rdu^{(l)}(v) + b), \tag{5}$$

where W and b denote the weight matrix and bias of the linear network, respectively, $\sigma(\cdot)$ is the nonlinear transfer function, and $rdu^{(l)}(v)$ represents the reduced information for node v obtained in layer-l of the linear network.

After several layers of GCNs, we introduce a pooling layer to aggregate the processed graph-structured data and get a vector for representing it. Specifically, we select the pooling layer that averages the feature vectors of each node as

$$\tilde{\mathcal{G}} = \frac{1}{|V|} \sum_{v \in V} \tilde{v}^{(k)}, \tag{6}$$

where $\tilde{\mathcal{G}}$ is the obtained vector, $|V|$ is the number of nodes in V, and k is the number of GCN layers. Finally, we send $\tilde{\mathcal{G}}$ to go through several linear layers to get the final output.

D. Design of the Graph-Aware DRL Model

We design the four basic elements of the DRL model:

- **Agent**. The DRL agent is based on the asynchronous advantage actor–critic (A3C) framework [34], which uses multiple pairs of A-GNN and C-GNN for parallel online training in several threads. For each pair of A-GNN and C-GNN, the A-GNN provides an action policy $\pi(S)$ based on the state S in graph structure, and chooses the appropriate action according to the policy $\pi(S)$. The C-GNN is responsible for learning the value of state S and evaluating the action from A-GNN based on it.
- **State**. The state S contains the topology information of the EON and the current provisioning scheme of a multicast session, and it is just the graph-structured data $\mathcal{G}(V, \tilde{V}, E, \tilde{E})$ obtained by the data preprocessing.
- **Action**. The action is modeled with a binary variable a, i.e., if the multicast session should be selected for reconfiguration, we have $a = 1$, and $a = 0$, otherwise.
- **Reward**. We define the reward as follows:

$$\begin{aligned} r = &-k_1 \cdot N_{re} + k_2 \cdot \left[slots(\mathcal{T}) - slots(\mathcal{T}^*)\right] \\ &+k_3 \cdot \left[cuts(\mathcal{T}) - cuts(\mathcal{T}^*)\right], \end{aligned} \tag{7}$$

where k_1, k_2, and k_3 are the positive coefficients for normalization; N_{re} represents the number of lightpath reroutings to reconfigure the multicast session; $\mathcal{T}$ and $\mathcal{T}^*$ denote the logic light-trees for the multicast session before and after the reconfiguration, respectively; $slots(\cdot)$ returns the number of FSs used by a logic light-tree; and $cuts(\cdot)$ returns the number of spectrum cuts [19] caused by a logic light-tree. Hence, the reward in Eq. (7) decreases with the number of lightpath reroutings, and increases with the spectrum usage and spectrum cuts saved by the reconfiguration. In other words, by maximizing the reward, our graph-aware DRL model tries to invoke the smallest number of lightpath reroutings on a multicast session to achieve the largest savings on spectrum usage and spectrum cuts.

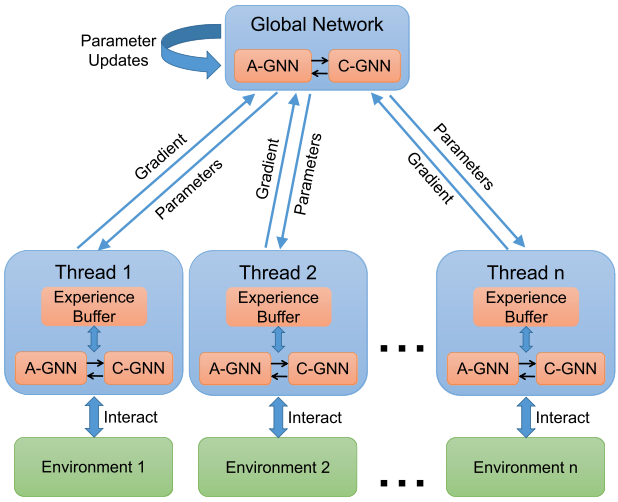

Fig. 2. Training of the DRL model in the A3C framework.

As shown in Fig. 2, we duplicate the A-GNN and C-GNN into several copies, use one copy as the global GNN, and put each of the others in a training thread to expedite the training process. Specifically, each training thread uses its A-GNN and C-GNN to interact with an EON environment independently to obtain training samples. In the iterative manner, the global GNN collects the gradients generated by the training threads, leverages them to update the parameters of its A-GNN and C-GNN, and synchronizes the updated parameters to the A-GNNs and C-GNNs in the training threads. As each thread is trained independently to obtain the gradients, the major benefit of this approach is that it effectively reduces the correlations among training samples; meanwhile, the multi-thread training can make full use of available computing resources to accelerate the online training.

Algorithm 2 explains the training process in a thread in detail, where we use T to record the number of training iterations, and $T_{\max}$ is the upper limit on training iterations. Lines 3–9 use the local A-GNN and C-GNN of the thread to interact with its own EON environment to collect training samples. Then, when enough training samples have been collected, lines 11–17 perform one iteration of the training. Specifically, the gradients are first calculated locally with the obtained training sample (lines 11–13), then they are forwarded to the global GNN (line 14), and finally the thread updates the parameters of its A-GNN and C-GNN according to the feedback from the global GNN and prepares itself for the next iteration of training (lines 15–17).

Algorithm 2. Training Process of a Thread

1. $T = 0$
2. **while** $T < T_{\max}$ **do**
3. **if** it is the time to reconfigure multicast sessions **then**
4. **for** each in-service multicast session MR_i **do**
5. get state S_i of MR_i
6. put S_i into the A-GNN to get an action a_i
7. apply a_i to the EON environment
8. calculate reward r_i
9. push $\{S_i, a_i, r_i\}$ to experience buffer
10. **if** experience buffer is full **then**
11. reset the gradients as zero
12. calculate the loss with A-GNN and C-GNN using the training samples in experience buffer
13. get the gradients with the loss
14. send the gradients to the global GNN
15. update the parameters of A-GNN and C-GNN according to the feedback from the global GNN
16. empty the experience buffer
17. $T = T + 1$

5. PERFORMANCE EVALUATION

In this section, we conduct extensive numerical simulations to evaluate our proposed approach based on graph-aware DRL.

A. Simulation Setup

The simulations use the four topologies in Fig. 3 for the EONs for DCIs, to confirm the universality of our proposal in terms of topologies. The capacity of each fiber link is assumed to be $F = 100$ FSs, where each FS has a bandwidth of 12.5 GHz to deliver 12.5 Gbps throughput. For each multicast session $MR(s, D, b, t)$, s, and D are randomly selected from the nodes in the EON; D contains [2,5] destinations initially; the bandwidth demand b is uniformly distributed within [50,200] Gbps; and the lifetime t follows the exponential distribution with an average of 500 time units. As the multicast sessions are dynamic, we generate new multicast sessions according to the Poisson traffic model, and for each in-service multicast, destinations can join or leave dynamically during its lifetime. Specifically, the service time of each destination follows exponential distribution, and it leaves its multicast session when the service time expires, while new destinations are generated with the Poisson distribution. In Subsection 5.E, we will change the settings mentioned above and run more simulations to further verify the universality of our proposal.

The reconfiguration of multicast sessions is invoked every 100 time units, and this interval is empirically set. The simulations compare our proposal based on graph-aware DRL with the heuristics for session selection in [22] (i.e., DTS and QTS), and consider both partial and full rearrangements for session reconfiguration. To ensure sufficient statistical accuracy, we average the results from five independent runs to obtain each data point.

B. Training Performance

We first evaluate the training performance of our DRL model. Note that the DRL model needs to first go through the offline training that optimizes its parameters initially, to make it suitable for being put into online operation/training. Hence, we study the performance of the offline training in this subsection, and will consider that of the online operation/training in subsequent ones. Figures 4(a) and 4(b) show how the average number of lightpath reroutings per session and blocking probability change in the training process, respectively, for the case in the NSFNET topology with the traffic load at 25 Erlangs. For comparison, we also plot the results from DTS-based and QTS-based algorithms in Fig. 4. Here, all the algorithms

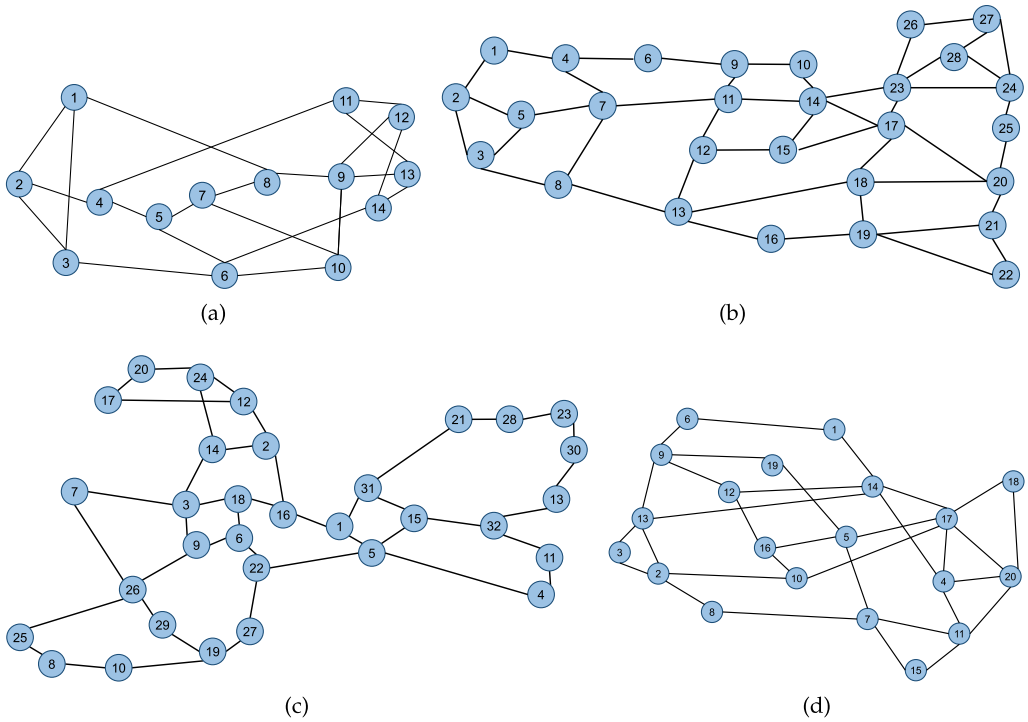

Fig. 3. EON topologies used in simulations. (a) 14-node NSFNET topology. (b) 28-node US Backbone (USB) topology. (c) 32-node European Backbone (EUB) topology. (d) 20-node random topology (RT).

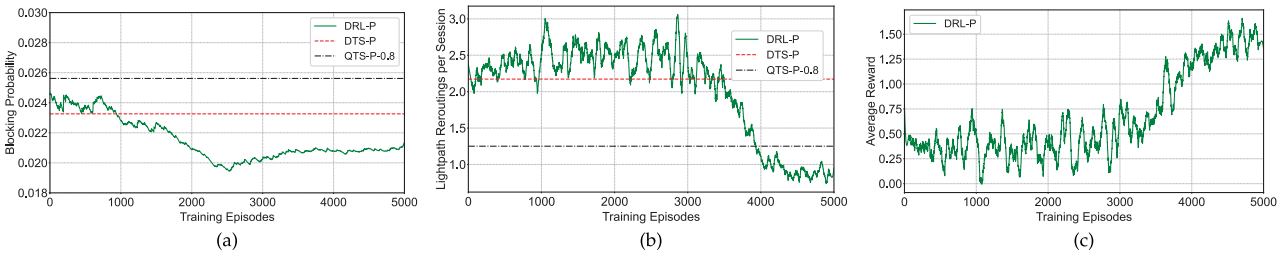

Fig. 4. Training performance (NSFNET, 25 Erlangs). (a) Overall blocking probability. (b) Average number of lightpath reroutings per session. (c) Average reward.

assume that full rearrangement is used to reconfigure the selected multicast sessions, and thus they are labeled with "-F." In the following, the algorithms labeled with "-F" and "-P" mean that they accomplish multicast session reconfiguration with the full and partial rearrangements in [22], respectively. For QTS-based algorithms, we can choose their thresholds on Q-values for session selection (i.e., Q_{lb}), and thus they are also labeled with their Q_{lb} values. For instance, the QTS-P-0.8 in Fig. 4 means that the multicast reconfiguration uses QTS to select multicast sessions with $Q_{lb} = 0.8$, and reconfigures them with partial rearrangement.

The results in Fig. 4(a) show that compared with QTS-P-0.8, DTS-P achieves a lower blocking probability by invoking many more lightpath reroutings per session. After being trained with 5000 episodes, our DRL-P can obtain a blocking probability that is lower than that of QTS-P-0.8, while its average lightpath reroutings per session are fewer than that of QTS-P-0.8 in Fig. 4(b). In other words, by utilizing its graph-aware intelligence, our DRL-P can balance the trade-off between overall blocking probability and average lightpath reroutings per session much better than the two benchmarks that use deterministic strategies.

Moreover, to clearly see how the average value of DRL-P's reward correlates with the metrics in Figs. 4(a) and 4(b) in the training, we plot it in Fig. 4(c). Here, we empirically set the positive coefficients in Eq. (7) as $k_1 = 6.0$, $k_2 = 1.0$, and $k_3 = 2.0$. It can be seen that the average reward generally increases with the decrease in overall blocking probability and average number of lightpath reroutings per session in Figs. 4(a) and 4(b), respectively. Note that we also check other traffic loads in NSFNET and the cases with full rearrangement, and confirm that our DRL-based approach can always achieve training performance similar to that in Fig. 4. Hence, the results are omitted due to the page limit.

Table 1 lists the running time of the offline training that makes our DRL model suitable for online operation/training. We observe that for EONs with the NSFNET, US Backbone (USB), European Backbone (EUB), and random topologies

Table 1. Average Running Time of Offline Training (Seconds)

Topology	NSFNET	USB	EUB	RT
DRL-P	25,167.35	34,698.01	34,944.14	32,342.45
DRL-F	31,541.22	40,423.58	45,508.86	36,843.80

(RT), the running time actually increases with the size of the topology. The DRL model for the full rearrangement scheme usually has a longer offline training time than that for the partial rearrangement scheme, regardless of the topology. These trends are expected, because when the topology of the EON becomes larger or the multicast reconfiguration changes from a partial rearrangement to full rearrangement, the problem of multicast reconfiguration actually becomes more complex.

C. Performance in Dynamic Network Environments

Next, we evaluate the performance of our DRL-based approach by putting the graph-aware DRL model, which has passed the offline training, in a dynamic network environment with the NSFNET topology, and compare its performance with DTS-based and QTS-based algorithms. Figures 5 and 6 show the simulation results for the cases using partial and full rearrangements, respectively. Here, "NR" denotes the case without multicast session reconfiguration. Note that in Figs. 5(a) and 6(a), when the traffic load is above 35 Erlangs, the blocking probabilities from the algorithms with a multicast session reconfiguration can actually exceed the practical range of the blocking probability in a real-world EON. Although the traffic loads exceed what should be considered in a real-world EON, we still simulate them to get a complete picture about how the algorithms will perform at various traffic loads. The DTS-based algorithm still provides the lowest overall blocking probability with the largest number of lightpath reroutings per session. By combining the results in the figures, we can conclude that to keep the overall blocking probabilities comparable to those of DTS-based and QTS-based algorithms, our DRL model always requires the smallest number of lightpath reroutings per session effectively, for all the simulation scenarios considered in Figs. 5 and 6. Hence, our graph-aware DRL-based approach can effectively reduce the operational complexity of dynamic multicast session reconfiguration, without sacrificing much performance on request blocking.

Moreover, we notice that the QTS-based algorithm can change the value of Q_{lb} to balance the trade-off between blocking probability and average lightpath reroutings per session. Hence, we change Q_{lb} to obtain different sets of blocking probabilities and average lightpath reroutings per session, and plot the results in Fig. 7, when the traffic load is set as 40 Erlangs. Here, we take average lightpath reroutings per session and blocking probability as the x axis and y axis, respectively, to illustrate the trade-off more clearly. It can be seen that no matter whether a partial or full rearrangement is used, the data point for the results from the DRL model is always below the curve for the results from the QTS-based algorithm. This verifies that the DRL model balances the trade-off better than QTS, regardless of the choice of Q_{lb}. In addition to 40 Erlangs, the simulations also check other traffic loads, and similar trends can be obtained.

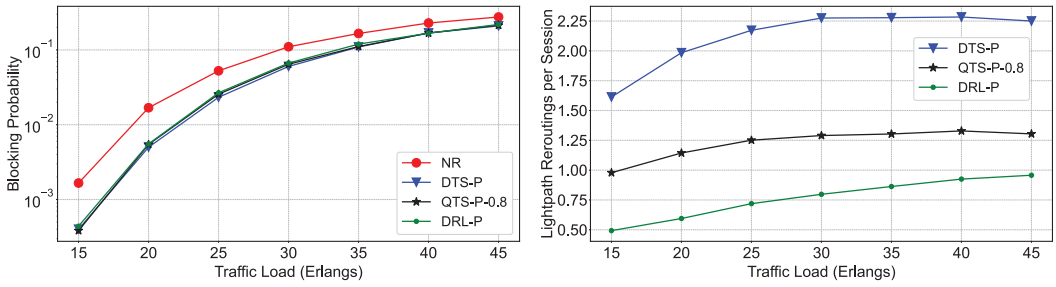

Fig. 5. Results of dynamic operations (NSFNET, partial rearrangement). (a) Overall blocking probability. (b) Average number of lightpath reroutings per session.

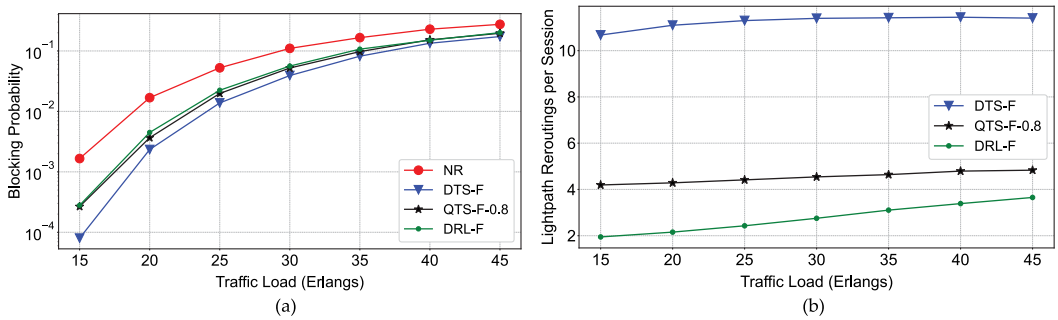

Fig. 6. Results of dynamic operations (NSFNET, full rearrangement). (a) Overall blocking probability. (b) Average number of lightpath reroutings per session.

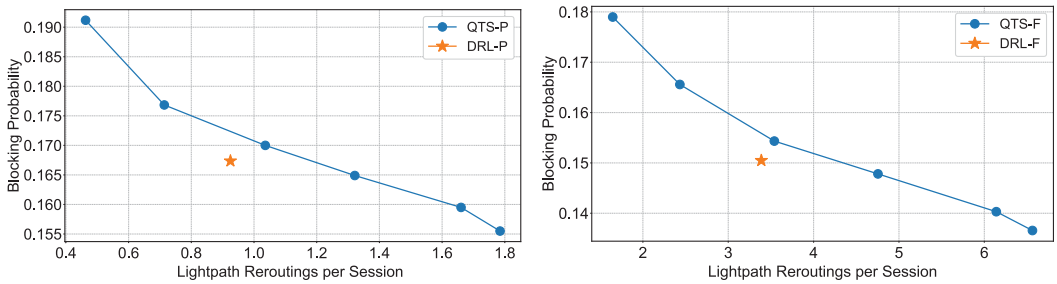

Fig. 7. Trade-off between blocking probability and average lightpath reroutings per session (NSFNET, 40 Erlangs). (a) Partial rearrangement. (b) Full rearrangement.

D. Universality across Different Topologies

We then evaluate the universality of our graph-aware DRL-based approach across different topologies. The operation principle of our graph-aware DRL model ensures that the DRL model trained in one EON topology can be directly applied to solve the problem of dynamic multicast session reconfiguration in others. Specifically, we need to abstract only the new topology information of the EON and the provisioning scheme of each multicast session as graph-structured data $\mathcal{G}(V, \tilde{V}, E, \tilde{E})$ and input the data to the trained DRL model, while the DRL model does not need to be redesigned or retrained. To verify this, the simulations apply the DRL model trained in NSFNET to solve the problem of dynamic multicast session reconfiguration in the other topologies in Fig. 3.

Figure 8 shows the results for the dynamic operations in USB, when a partial rearrangement is considered. We can see that the results follow trends similar to those in Fig. 5. To further clarify the adaptability of our DRL model, we take the case of traffic load at 25 Erlangs in USB as an example, and plot how the performance metrics change over the simulation time in Fig. 9. As we directly apply the DRL model trained in NSFNET to the EON with the USB topology, a zero-shot transfer learning (i.e., applying a trained DRL model to an unseen environment for the same task [35]) is actually considered. It can be seen that due to the superior adaptability of our DRL model, it achieves relatively good performance on the performance metrics at the beginning of the online operation/training, and both the overall blocking probability and average number of lightpath reroutings per session changes only slightly afterwards.

Figures 10 and 11 illustrate the results obtained by directly applying the DRL model trained in NSFNET to the EONs with EUB and RT topologies, respectively. The results still

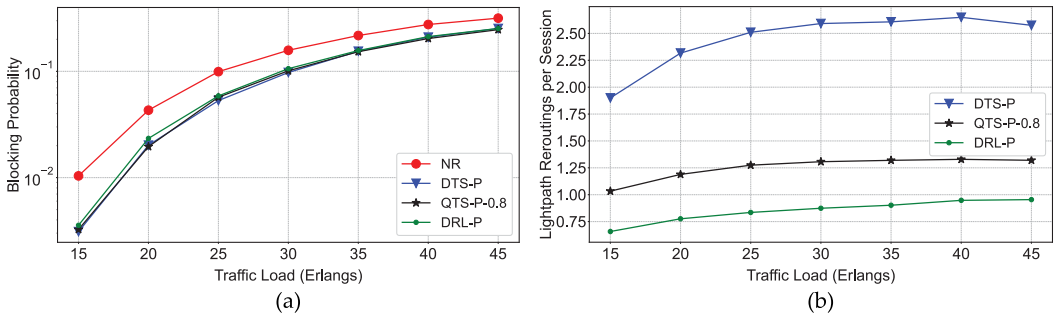

Fig. 8. Results of dynamic operations (USB, partial rearrangement). (a) Overall blocking probability. (b) Average number of lightpath reroutings per session.

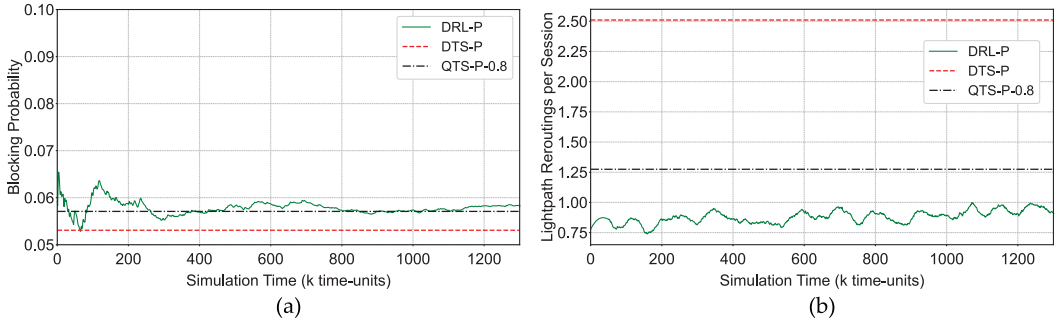

Fig. 9. Performance on zero-shot transfer learning (USB, 25 Erlangs). (a) Overall blocking probability. (b) Average number of lightpath reroutings per session.

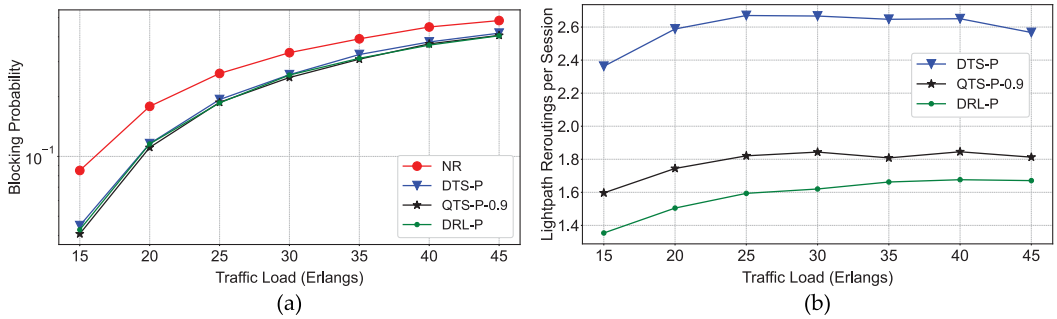

Fig. 10. Results of dynamic operations (EUB, partial rearrangement). (a) Overall blocking probability. (b) Average number of lightpath reroutings per session.

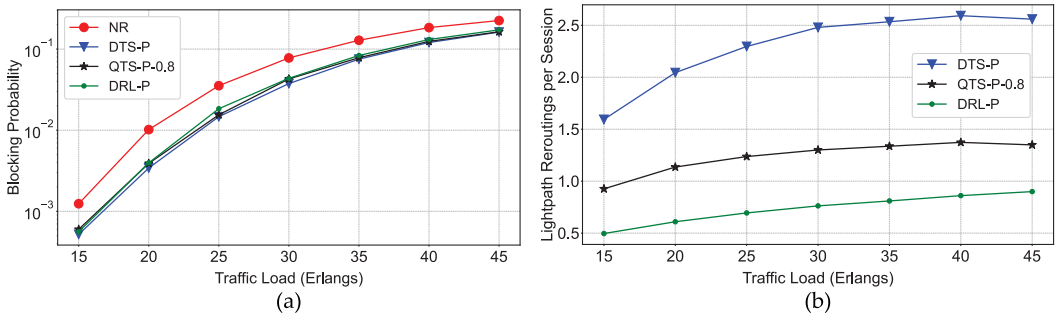

Fig. 11. Results of dynamic operations (RT, partial rearrangement). (a) Overall blocking probability. (b) Average number of lightpath reroutings per session.

follow trends similar to those in Fig. 5. Note that when the EON topology changes, we might need to change the value of Q_{lb} (i.e., the threshold on Q-value for session selection) for QTS-based algorithms empirically. This is the reason that we simulate QTS-P-0.9 in EUB (as shown in Fig. 10). On the other hand, with its graph-aware intelligence, our DRL model can adapt to different topologies without such manual adjustments. Although the results in Figs. 8–11 are all about cases that use partial rearrangement, we also check those with full rearrangement and confirm that our DRL-based approach achieves similar performances in them too. Therefore, we prove the universality of our DRL model across different topologies.

Table 2 lists the average running time per multicast session reconfiguration of the algorithms. Here, for our DRL model, the running time is only for its online operation/training, because the offline training should be finished before the DRL model can be put into operation, and its running time has already been summarized in Table 1. The results in Table 2 suggest that the running time of all the algorithms is comparable and short enough to adapt to dynamic operations. The running time of our DRL model is less than that of the QTS-based algorithm in all the simulation scenarios, while as the DTS-based algorithm makes decisions only according to the depth of each logical light-tree, it runs the fastest. The running time of each algorithm generally increases with the size of the topology, or from using a partial rearrangement to using full rearrangement.

Table 2. Average Running Time per Multicast Reconfiguration (Seconds)

Topology	NSFNET	USB	EUB	RT
DRL-P	0.1943	0.2494	0.2383	0.2908
QTS-P	0.2964	0.3941	0.4040	0.3986
DTS-P	0.0693	0.0900	0.1016	0.0867
DRL-F	0.2104	0.2823	0.3022	0.3110
QTS-F	0.3620	0.4648	0.4927	0.4448
DTS-F	0.1995	0.2723	0.2882	0.2687

E. Generalization to Various EON Settings

Finally, we consider more simulation settings to verify that our proposed graph-aware DRL model can adapt to various EON settings. First of all, we notice that the assumption of a Poisson traffic model might not hold in today's Internet. Hence, we design a new simulation scenario, in which the multicast sessions are generated dynamically in a bursty manner, i.e., they come in according to the realistic on–off pattern for bursty Internet traffic [36]. Note that we still quantify the traffic load of the multicast sessions with Erlangs, i.e., the production of the average number of new sessions per unit time and the average lifetime of each session in time units. The results of the simulations with NSFNET are shown in Fig. 12, and by comparing them with those in Fig. 5, we can see similar trends. As the bursty traffic model is more likely to cause session blocking, the blocking probability of each algorithm in Fig. 12 is higher. Nevertheless, our DRL model still retains its advantage of significantly reducing the number of reconfiguration operations without sacrificing the performance on blocking probability. With the bursty traffic model, we also simulate other EON topologies and test the algorithms with full rearrangement, while the results always follow trends similar to those in Fig. 12.

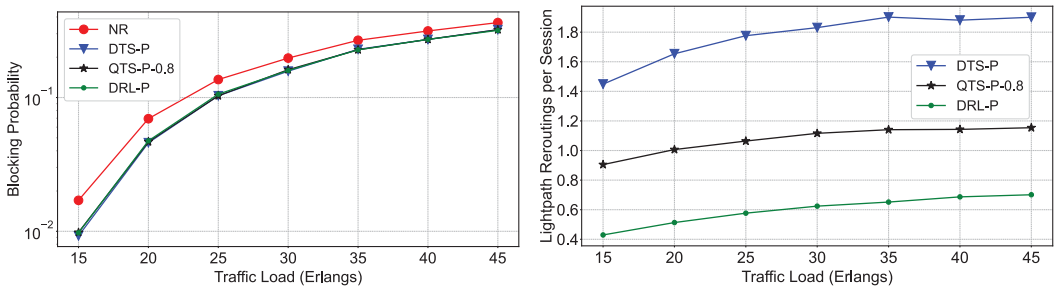

Fig. 12. Results of dynamic operations with bursty multicast sessions (NSFNET, partial rearrangement). (a) Overall blocking probability. (b) Average number of lightpath reroutings per session.

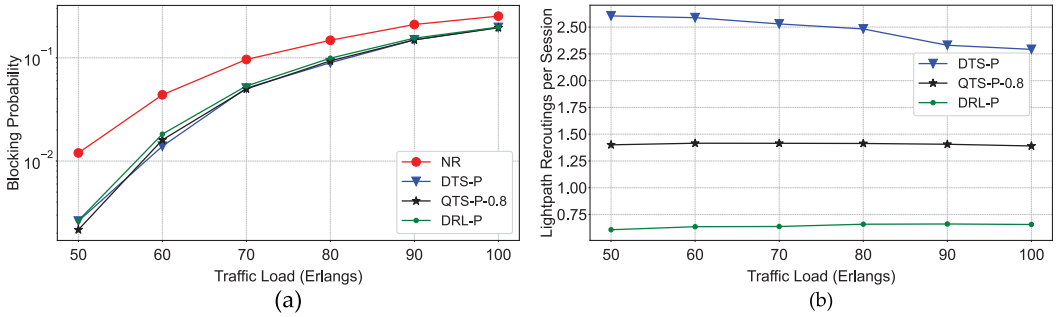

Fig. 13. Results of dynamic operations with 200 FSs per fiber link (NSFNET, partial rearrangement). (a) Overall blocking probability. (b) Average number of lightpath reroutings per session.

Second, we increase the number of FSs on each fiber link to 200 to simulate the EONs with more spectrum resources. The results of the simulations with NSFNET are shown in Fig. 13, and by comparing them with those in Fig. 5, we still see similar trends. Since there are more spectrum resources in the EON, we need to increase the traffic load to see the same blocking probability. Our DRL model still exhibits advantages over the heuristics, which suggests that its performance is not affected by the change of spectrum resources in the EON. With the new setting of spectrum resources, we also simulate other EON topologies and test the algorithms with full rearrangement, and the results always follow trends similar to those in Fig. 13.

Finally, considering the fact that in a real-world EON, there are unicast and anycast lightpaths coexisting with multicast sessions, we design a realistic simulation scenario in which unicast and anycast lightpaths are used as the background traffic of multicast sessions. Specifically, to create a stressful scenario for our DRL model, we make the total bandwidth demands of unicast, anycast, and multicast account for 25%, 25%, and 50% of the overall bandwidth usage in the EON, respectively. The results of the simulations with NSFNET are shown in Fig. 14, and by comparing them with those in Fig. 5, we can see that the blocking probability of multicast sessions becomes lower. This is because, for the same traffic load, unicast and anycast lightpaths generally require fewer spectrum resources than multicast sessions, and thus the total spectrum usage is actually smaller. For the same reason, the gap on blocking probability between the multicast reconfiguration algorithms and the case without multicast reconfiguration (i.e., NR) becomes smaller too. Note that, compared with the QTS-based and DTS-based algorithms, our DRL model still invokes a smaller number of lightpath reroutings per session to maintain almost the same blocking probability. This verifies

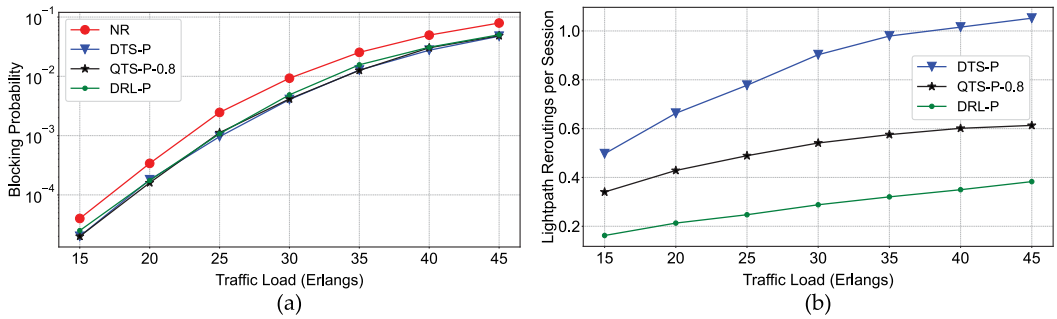

Fig. 14. Results of dynamic operations with unicast/anycast background traffic (NSFNET, partial rearrangement). (a) Overall blocking probability. (b) Average number of lightpath reroutings per session.

the effectiveness of our DRL model in the EON environment that contains mixed types of traffic demands.

6. CONCLUSION

In this work, we revisited the problem of how to formulate and reconfigure multicast sessions in an EON, and proposed a DRL model based on GNNs that can solve the sub-problem of multicast session selection in a more universal and adaptive way. Specifically, we abstracted the state information of each multicast session as graph-structured data, which can be directly analyzed by our graph-aware DRL model. Then, the graph-based reasoning capability of our proposal made sure that the state information of each multicast session can be analyzed in depth for dynamic reconfiguration, and facilitated the universality across different topologies. Hence, an important takeaway is that our graph-aware design of the DRL model made its architect and operation independent of the EON's topology, and thus avoided the hassle of redesigning its architecture to adapt to different EON topologies.

Simulation results verified that, compared with the existing deterministic algorithms based on DTS and QTS, our graph-aware DRL-based approach can significantly reduce the average lightpath reroutings per multicast session while maintaining the overall blocking probability approximately at the same level. This suggests that our proposal can balance the trade-off between the number of reconfiguration operations and blocking performance much better than existing algorithms. Moreover, our simulations also confirmed that the DRL model trained in one EON environment can easily adapt to solve the problem of dynamic multicast session reconfiguration in EONs with various settings (e.g., different topologies, spectrum resources, traffic models, and request types). Therefore, the universality of our proposal helps to effectively save the time and effort needed to adjust the DRL model according to an EON's setting, and provides a more realistic solution for network automation.

Funding. National Key Research and Development Program of China (2020YFB1806400); National Natural Science Foundation of China (61871357); Chinese Academy of Sciences (XDC02070300); Fundamental Research Funds for the Central Universities (WK3500000006).

REFERENCES

1. Cisco, "Cisco Visual Networking Index, 2017–2022."
2. P. Lu, L. Zhang, X. Liu, J. Yao, and Z. Zhu, "Highly efficient data migration and backup for big data applications in elastic optical inter-data-center networks," IEEE Netw. **29**, 36–42 (2015).
3. N. Laoutaris, M. Sirivianos, X. Yang, and P. Rodriguez, "Inter-datacenter bulk transfers with NetStitcher," in *Proceedings of ACM SIGCOMM 2011* (2012), pp. 74–85.
4. V. Dukic, C. Gkantsidis, T. Karagiannis, F. Parmigiani, A. Singla, M. Filer, J. Cox, A. Ptasznik, N. Harland, W. Saunders, and C. Belady, "Beyond the mega-data center: networking multi-data center regions," in *Proceedings of ACM SIGCOMM* (2020), pp. 765–781.
5. O. Gerstel, M. Jinno, A. Lord, and B. Yoo, "Elastic optical networking: a new dawn for the optical layer?" IEEE Commun. Mag. **50**(2), s12–s20 (2012).
6. Z. Zhu, W. Lu, L. Zhang, and N. Ansari, "Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing," J. Lightwave Technol. **31**, 15–22 (2013).
7. L. Gong and Z. Zhu, "Virtual optical network embedding (VONE) over elastic optical networks," J. Lightwave Technol. **32**, 450–460 (2014).
8. L. Sahasrabuddhe and B. Mukherjee, "Light trees: optical multicasting for improved performance in wavelength routed networks," IEEE Commun. Mag. **37**(2), 67–73 (1999).
9. Q. Wang and L. Chen, "Performance analysis of multicast traffic over spectrum elastic optical networks," in *Optical Fiber Communication Conference (OFC)* (2012), paper OTh3B.7.
10. X. Liu, L. Gong, and Z. Zhu, "On the spectrum-efficient overlay multicast in elastic optical networks built with multicast-incapable switches," IEEE Commun. Lett. **17**, 1860–1863 (2013).
11. L. Gong, X. Zhou, X. Liu, W. Zhao, W. Lu, and Z. Zhu, "Efficient resource allocation for all-optical multicasting over spectrum-sliced elastic optical networks," J. Opt. Commun. Netw. **5**, 836–847 (2013).
12. X. Liu, L. Gong, and Z. Zhu, "Design integrated RSA for multicast in elastic optical networks with a layered approach," in *Proceedings of GLOBECOM* (2013), pp. 2346–2351.
13. K. Walkowiak, R. Goscien, M. Klinkowski, and M. Wozniak, "Optimization of multicast traffic in elastic optical networks with distance-adaptive transmission," IEEE Commun. Lett. **18**, 2117–2120 (2014).
14. Z. Zhu, X. Liu, Y. Wang, W. Lu, L. Gong, S. Yu, and N. Ansari, "Impairment- and splitting-aware cloud-ready multicast provisioning in elastic optical networks," IEEE/ACM Trans. Netw. **25**, 1220–1234 (2017).
15. A. Mahimkar, A. Chiu, R. Doverspike, M. Feuer, P. Magill, E. Mavrogiorgis, J. Pastor, S. Woodward, and J. Yates, "Bandwidth on demand for inter-data center communication," in *Proceedings of HOTNETS* (2011), article no. 24.
16. A. Malis, B. Wilson, G. Clapp, and V. Shukla, "Requirements for very fast setup of GMPLS label switched paths (LSPs)," IETF RFC 7709, 2015.
17. A. Castro, L. Velasco, M. Ruiz, M. Klinkowski, J. Fernandez-Palacios, and D. Careglio, "Dynamic routing and spectrum (re)allocation in future flexgrid optical networks," Comput. Netw. **56**, 2869–2883 (2012).
18. M. Klinkowski, M. Ruiz, L. Velasco, D. Careglio, V. Lopez, and J. Comellas, "Elastic spectrum allocation for time-varying traffic in flexgrid optical networks," IEEE J. Sel. Areas Commun. **31**, 26–38 (2013).
19. Y. Yin, H. Zhang, M. Zhang, M. Xia, Z. Zhu, S. Dahlfort, and S. Yoo, "Spectral and spatial 2D fragmentation-aware routing and spectrum assignment algorithms in elastic optical networks," J. Opt. Commun. Netw. **5**, A100–A106 (2013).
20. Y. Sone, A. Watanabe, W. Imajuku, Y. Tsukishima, B. Kozicki, H. Takara, and M. Jinno, "Bandwidth squeezed restoration in spectrum-sliced elastic optical path networks (SLICE)," J. Opt. Commun. Netw. **3**, 223–233 (2012).
21. W. Lu and Z. Zhu, "Malleable reservation based bulk-data transfer to recycle spectrum fragments in elastic optical networks," J. Lightwave Technol. **33**, 2078–2086 (2015).
22. M. Zeng, Y. Li, W. Fang, W. Lu, X. Liu, H. Yu, and Z. Zhu, "Control plane innovations to realize dynamic formulation of multicast sessions in inter-DC software-defined elastic optical networks," Opt. Switching Netw. **23**, 259–269 (2017).
23. R. Gu, Z. Yang, and Y. Ji, "Machine learning for intelligent optical networks: a comprehensive survey," J. Netw. Comput. Appl. **157**, 102576 (2020).
24. F. Scarselli, M. Gori, A. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," IEEE Trans. Neur. Netw. **20**, 61–80 (2009).
25. J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: a review of methods and applications," AI Open **1**, 57–81 (2021).
26. A. Ding and G. Poo, "A survey of optical multicast over WDM networks," Comput. Commun. **26**, 193–200 (2003).
27. K. Christodoulopoulos, I. Tomkos, and E. Varvarigos, "Elastic bandwidth allocation in flexible OFDM-based optical networks," J. Lightwave Technol. **29**, 1354–1366 (2011).

28. P. Zhu, J. Li, Y. Chen, X. Chen, Z. Wu, D. Ge, Z. Chen, and Y. He, "Experimental demonstration of EON node supporting reconfigurable optical superchannel multicasting," Opt. Express **23**, 20495–20504 (2015).
29. X. Li, L. Zhang, J. Wei, and S. Huang, "Deep neural network based OSNR and availability predictions for multicast light-trees in optical WDM networks," Opt. Express **28**, 10648–10669 (2020).
30. K. Rusek, J. Suarez-Varela, A. Mestres, P. Barlet-Ros, and A. Cabellos-Aparicio, "Unveiling the potential of graph neural networks for network modeling and optimization in SDN," in *Proceedings of Symposium on SDN Research (SOSR)* (2019), pp. 140–151.
31. P. Sun, J. Lan, J. Li, Z. Guo, Y. Hu, and T. Hu, "Efficient flow migration for NFV with graph-aware deep reinforcement learning," Comput. Netw. **183**, 107575 (2020).
32. N. Sambo, P. Castoldi, A. D'Errico, E. Riccardi, A. Pagano, M. Moreolo, J. Fabrega, D. Rafique, A. Napoli, S. Frigerio, E. Salas, G. Zervas, M. Nolle, J. Fischer, A. Lord, and J. Gimenez, "Next generation sliceable bandwidth variable transponders," IEEE Commun. Mag. **53**(2), 163–171 (2015).
33. M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," arXiv:1606.09375 (2017).
34. V. Mnih, A. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *Proceedings of the International Conference on Machine Learning (ICML)* (2016), pp. 1928–1937.
35. Z. Zhang and V. Saligrama, "Zero-shot learning via semantic similarity embedding," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (2015), pp. 4166–4174.
36. X. Yang, "Designing traffic profiles for bursty Internet traffic," in *Proceedings of GLOBECOM* (2002), pp. 2149–2154.