# DeepCoop: Leveraging Cooperative DRL Agents to Achieve Scalable Network Automation for Multi-Domain SD-EONs

**Baojia Li[1] and Zuqing Zhu[1]**

*1. University of Science and Technology of China, Hefei, Anhui 230027, China, Email: zqzhu@ieee.org*

**Abstract:** We design DeepCoop to realize service provisioning in multi-domain software-defined elastic optical networks (SD-EONs) with cooperative deep reinforcement learning (DRL) agents.

**OCIS codes:** (060.1155) Software-defined optical networks; (060.4251) Networks, assignment and routing algorithms.

## 1. Introduction

The emergence of new network paradigms (*e.g.*, cloud computing and 5G) has pushed backbone networks undergone dramatic changes to adapt to the ever-increasing traffic and various quality-of-service (QoS) demands [1]. Software-defined elastic optical networks (SD-EONs) are therefore widely considered as a promising solution for future backbone networks [2]. This is because SD-EON makes the network control and management (NC&M) more adaptive and programmable with a centralized control plane, and thus it further promotes the agility of EON. Meanwhile, since a backbone network can cover a relatively large geographical area, it is usually owned and operated by different carriers. Hence, it would be inevitable to consider multi-domain SD-EONs [3]. Specifically, the inter-domain service provisioning scheme, which considers the autonomy of each domain, should be developed with reasonably good scalability.

The basic problem of service provisioning in an EON is the routing and spectrum assignment (RSA) [4], which is known to be NP-hard. Therefore, even though numerous heuristic RSA algorithms have been designed, they can hardly guarantee the optimality of their solutions or even ensure the approximation gap to optimal ones. To this end, people tried to leverage deep reinforcement learning (DRL) to solve RSA [5], because DRL can work with statistical models to solve complex problems without being explicitly programmed. Although the DRL-based approach in [5] had outperformed a few well-known heuristic algorithms, it still suffers from limited scalability and universality. This is because the DRL model was designed to select the actual RSA scheme for each lightpath. Hence, the action space size will increase fast, when more routing path candidates and more available frequency slot (FS) blocks on each path are considered for each lightpath. This would make the model hard to converge, and thus cause severe scalability issues. Moreover, as the “goodness” of RSA schemes is tightly related to the topology of the concerned EON, a DRL model that has been trained in one EON would be inapplicable in another one. Meanwhile, for the inter-domain provisioning in a multi-domain SD-EON, the study in [6] considered a DRL-based approach that involves multiple DRL agents. Each DRL agent corresponds to a broker that can observe all the domains, and it accomplishes two tasks: 1) selecting the whole inter-domain provisioning scheme for each connection request, and 2) competing with other agents for service provisioning opportunities. For the first task, the DRL agent operates similarly as the one in [5]. Hence, the scalability and universality issues still exist. For the second task, since the agents compete, instead of cooperate with each other, the overall computation complexity would increase other than decrease with the number of agents.

The limitations of existing approaches motivate us to propose an inter-domain service framework that leverages cooperative DRL agents to achieve scalable network automation, namely, DeepCoop. For each connection request, DeepCoop first utilizes a domain-level path computation element (PCE) to get the domain sequence to go through, and then counts on the DRL agent allocated to each domain for intra-domain RSA calculation and inter-domain link selection. By exchanging little information among each other (*i.e.*, sharing state parameters and feeding back rewards collaboratively), the DRL agents calculate inter-domain provisioning schemes distributedly. Moreover, each DRL agent is designed to select a proper RSA algorithm (*i.e.*, from a few well-known heuristics) for each connection request, and thus the size of its action space is significantly reduced and will be fixed for different domain topologies. Therefore, DeepCoop effectively relieves both the scalability and universality issues. We evaluate the performance of DeepCoop with numerical simulations, and the results indicate that DeepCoop outperforms several existing benchmarks.

## 2. System Architecture and Design of DeepCoop

Fig. 1(a) shows the system architecture of DeepCoop. We place a domain-level PCE on top of all the domain managers, and use it to determine the domain-level routing path for each connection request, which is the domain sequence from its source to its destination. Each domain manager consists of two components, *i.e.*, a DRL agent and an SDN controller. The controller reports current intra-domain status (*i.e.*, the spectrum usage on each intra-domain fiber link) and pending connection requests to the DRL agent, and configures the intra-domain lightpath segment for each request according to the service provisioning scheme returned by the agent. The DRL agent is trained in the online manner to learn how to choose the best RSA algorithm based on the current intra-domain status, to calculate the lightpath segment in the domain for a connection request. Meanwhile, the DRL agents of all the domain along the inter-domain routing path work cooperatively to determine the inter-domain RSA schemes.

We use the example in Fig. 1(b) to explain the cooperation between two DRL agents, which belong to adjacent domains. Let us assume that for a connection request, the domain-level PCE has already determined the domain sequence as *Domain* 1→*Domain* 2. Then, *DRL Agents* 1 and 2 work cooperatively to figure out the inter-domain provisioning scheme, which is done by taking their actions sequentially. Firstly, *DRL Agent* 1 obtains the RSA scheme within its domain (*i.e.*, including the egress node to *Domain* 2), and sends the egress node to *DRL Agent* 2. Secondly, *DRL Agent* 2 uses the egress node to first get the ingress node and then the RSA scheme in its domain. Here, when calculating the intra-domain RSA schemes, each DRL agent first selects a proper RSA algorithm from several present heuristics, and then computes RSA with the algorithm. Finally, after finalizing the intra-domain provisioning schemes in both domains, DeepCoop obtains the spectrum assignment on the inter-domain link with the first-fit scenario. In this work, we assume that there are mandatory optical-electrical-optical (O/E/O) conversions at border nodes to ensure domain privacy. Hence, there is no need to consider the spectrum continuous constraint when provisioning a connection across border nodes (*i.e.*, from an intra-domain link to an inter-domain one, or *vice versa*).

Next, the SDN controllers deploy the inter-domain provisioning scheme of the request in the multi-domain SD-EON. After this, each controller feeds back an evaluation of the RSA scheme implemented in its domain to the reward system, where the reward of its DRL agent's previous action (*i.e.*, the selection of a RSA algorithm) gets calculated (as shown in Fig. 1(b)). The reward system pushes the reward and its corresponding action and state into the experience buffer as a training sample, which will be used to update the deep neural networks (DNNs) in the DRL agent on-the-fly. Note that, the DRL agents' actions are correlated and affect each other's performance. For instance, *DRL Agent* 1 determines the ingress node to *Domain* 2, which will in turn affects the performance of *DRL Agent* 2. Hence, the state observed by each DRL agent should also include the status of its adjacent domains, and its reward calculation should take the feedbacks from the agents in adjacent domains into account. As illustrated in Fig. 1(b), we design the DRL agents to share state parameters and calculate rewards collaboratively, and in the meantime, we only allow very necessary interactions among the agents to restrict overheads and thus ensure the cost-effectiveness of NC&M.

We model the multi-domain SD-EON as a graph $G = \{G_i(V_i, E_i), i \in [1, N]\}$, where $i$ is the domain index, $N$ is the number of domains, and $V_i$ and $E_i$ are the sets of nodes and links in *Domain* $i$. A dynamic connection request is represented as $\chi^t(s, d, BD, \tau)$, where $t$ is the arrival time, $s$ and $d$ are the source and destination nodes, respectively, $BD$ is the number of required FS', and $\tau$ is the service duration. Upon the arrival of $\chi^t$, domain-level PCE calculates the domain sequence for its provisioning, *i.e.*, $\psi_t = [G_{k_1}, \cdots, G_{k_m}]$, where $G_{k_1}$ and $G_{k_m}$ denote the source and destination domains, respectively, and the intermediate ones are in between of them. Then, the DRL agents in the selected domains, *i.e.*, $\{DRL_j,\ j \in \psi_t\}$, utilize the procedure explained above to determine the inter-domain provisioning scheme of $\chi^t$ through cooperation. The four basic elements of each agent's DRL model are designed as follows.

**State**: The state $s_j^t$ observed by $DRL_j$ includes: request $\chi^t$, status of *Domain* $j$, and status of the next domain. The status of *Domain* $j$ is represented as the average number and average size of available FS blocks on each candidate path. Here, we calculate $K$ shortest paths through *Domain* $j$ as the candidates to provision $\chi^t$. The status of the next domain is defined similarly, but to keep domain privacy, the next domain will only disclose the average number and average size of available FS blocks between each of its border node pair, but hide the path information. For the destination domain, the status of its next domain is empty (*i.e.*, filled with zeros).

**Action**: The action $a_j^t$ of $DRL_j$ is to select a proper RSA algorithm to calculate the intra-domain RSA in *Domain* $j$, based on the observed state $s_j^t$. This work considers three famous RSA algorithms, which are the shortest path and first-fit (SP-FF), $K$ shortest paths and first fit (KSP-FF), and $K$ shortest paths and load balancing (KSP-LB) [4].

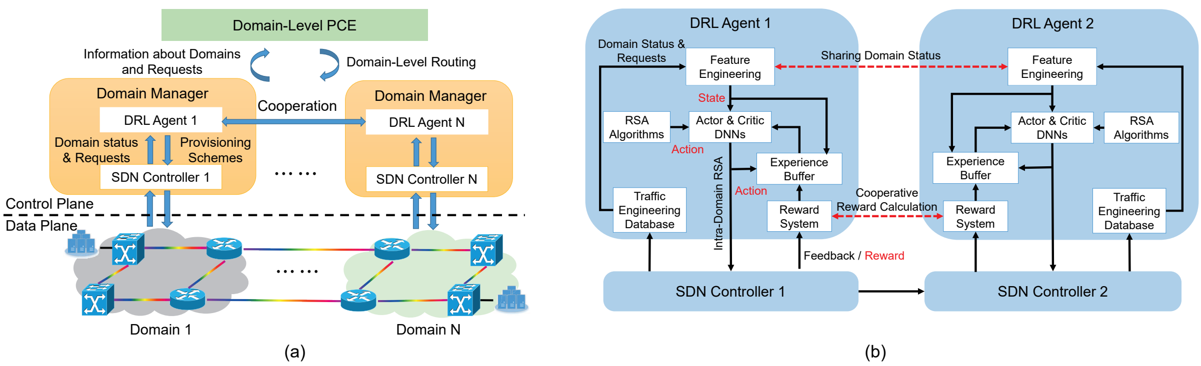


Fig. 1. (a) System architecture of DeepCoop, and (b) Cooperation between two DRL agents for inter-domain provisioning.

**Reward**: The objective of $DRL_j$ is to minimize the long-term blocking probability of inter-domain provisioning. Hence, we calculate the instant reward $r_j^t$ after serving request $\chi^t$ in *Domain j* as follows. The value of $r_j^t$ is initialized as 0, and it gets increased by 1 if the intra-domain provisioning in the next domain is successful. Otherwise, we decrease $r_j^t$ by the total number of remaining domains (*i.e.*, the domains from the next one to the destination).

**Agent**: Each DRL agent is based on advantage actor critic (A2C) [7], which uses two DNNs to ensure good performance on dynamic decision making. Specifically, it uses the actor DNN to choose an action in a state, while the critic DNN evaluates the action's performance. The DRL agents are trained in the online manner, with the common training procedure of A2C, and each DRL agent is trained independently using its own experience buffer.

## 3. Performance Evaluation

We evaluate the performance of DeepCoop with a multi-domain SD-EON whose topology is shown in Fig. 2(a), which consists of three domains whose topologies are all the 14-node NSFNET. Here, we purposely design the multi-domain topology with relatively large numbers of nodes and links (*i.e.*, 42 nodes and 142 directional links) to explore the scalability of our proposal. We assume that each intra-domain link can accommodate 100 FS', while the capacity of each inter-domain link is set as 300 FS' to prevent inter-domain bottlenecks. Connection requests are dynamically generated according to the Poisson traffic model. Their source and destination nodes are randomly selected in *Domains* 1 and 3, respectively, and their bandwidth demands are uniformly distributed within $[2,9]$ FS'. We consider four benchmarks, which are named as all-SP-FF, all-KSP-FF, all-KSP-LB and DeepRSA. For the first three benchmarks, they always use the same RSA algorithm to calculate the intra-domain provisioning schemes. For instance, all-SP-FF refers to the scenario that uses SP-FF for intra-domain provisioning in all the three domains. DeepRSA utilizes the same system architecture as DeepCoop, but its DRL agents work independently without any cooperation or information sharing.

Fig. 2(b) shows the evolution of blocking probability along training episodes, when the traffic load is fixed at 70 Erlangs. We observe that as the training goes on, the blocking probability from DeepCoop quickly converges to $2.3\times10^{-3}$ (after 60 episodes), which achieves $\sim$96%, $\sim$90%, and $\sim$91% reduction on blocking probability related to all-SP-FF, all-KSP-FF and all-KSP-LB, respectively. This confirms that the cooperative DRL agents in DeepCoop can efficiently learn how to serve connection requests in a large-scale multi-domain SD-EON. Meanwhile, we notice that the blocking probability from DeepRSA fluctuates along training episodes and is difficult to converge. This is because the agents in DeepRSA do not cooperate, which further verifies the superiority of our cooperative design for DeepCoop. Finally, we plot the results on blocking probability from the five algorithms at different traffic loads in Fig. 2(c). It can be see that DeepCoop outperforms all the benchmarks to provide the lowest blocking probability.

## 4. Summary

We leveraged cooperative DRL to propose DeepCoop, which can achieve scalable network automation for inter-domain provisioning in multi-domain SD-EONs. Simulation results confirmed that DeepCoop can quickly learn how to provision lightpaths in a large-scale network and achieve lower blocking probability than the existing benchmarks.

## References


[1] P. Lu *et al.*, "Highly-efficient data migration and backup for Big Data applications in elastic optical inter-datacenter networks," *IEEE Netw.*, vol. 29, pp. 36-42, Sept./Oct. 2015.

[2] L. Liu *et al.*, "OpenSlice: an OpenFlow-based control plane for spectrum sliced elastic optical path networks," *Opt. Express*, vol. 21, pp. 4194-4204, Feb. 2013.

[3] Z. Zhu *et al.*, "Demonstration of cooperative resource allocation in an OpenFlow-controlled multidomain and multinational SD-EON testbed," *J. Lightw. Technol.*, vol. 33, pp. 1508-1514, Apr. 2015.

[4] Z. Zhu *et al.*, "Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing," *J. Lightw. Technol.*, vol. 31, pp. 15-22, Jan. 2013.

[5] X. Chen *et al.*, "DeepRMSA: A deep reinforcement learning framework for routing, modulation and spectrum assignment in elastic optical networks," *J. Lightw. Technol.*, vol. 37, pp. 4155-4163, Aug. 2019.

[6] X. Chen *et al.*, "Multi-agent deep reinforcement learning in cognitive inter-domain networking with multi-broker orchestration," *in Proc. of OFC*, paper M2A.2, Mar. 2019.

[7] V. Mnih *et al.*, "Asynchronous methods for deep reinforcement learning," in Proc. of ICML, pp. 1928-1937, Jun. 2016.


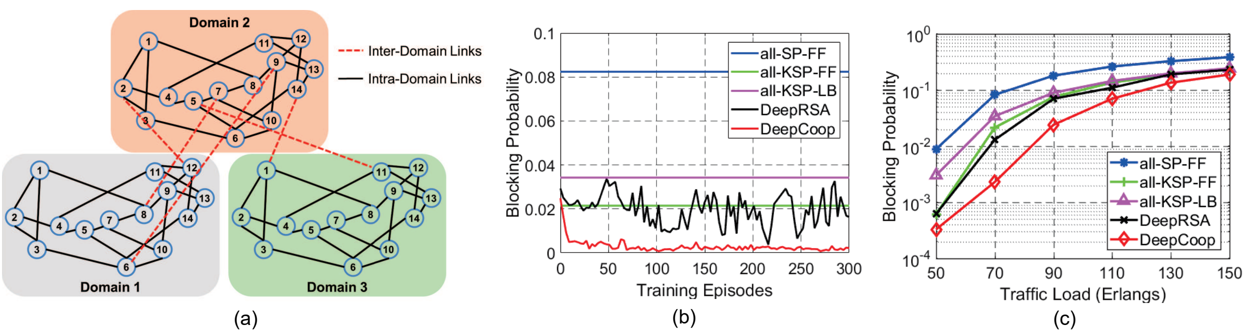


Fig. 2. Simulation setup and results, (a) Topology of the multi-domain SD-EON, (b) Evolution of blocking probability during training (traffic load at 70 Erlangs), and (c) Blocking probability at different traffic loads.