Adaptive Traffic Grooming Using Reinforcement Learning in Multilayer Elastic Optical Networks

Takafumi Tanaka
NTT Network Innovation Laboratories, NTT Corporation, Japan
takafumi.tanaka.mg@hco.ntt.co.jp

Abstract: We introduce a traffic grooming technique for multilayer networks that uses reinforcement learning. We confirm its superior performance over heuristic methods as regards its ability to meet several key requirements such as blocking probability and energy consumption.

1. Introduction

The IP over WDM architecture is gaining attention for metro core networks again. This concept was proposed over 20 years ago, but only now is feasible in terms of hardware, as the emergence of pluggable digital coherent transceivers enables high-capacity, long-distance communications directly from IP routers. From a software perspective, Telecom Infra Project (TIP) has identified the integration of IP and WDM operational control as its next target, following the activities for disaggregation, softwarization, and open sourcing [1].

One of the essential techniques for multilayer network planning is traffic grooming, which accommodates IP, Optical Data Unit (ODU), and other electrical layer paths or connections into optical paths, as well as the Routing and Spectrum Assignment (RSA) algorithms that determine the route and frequency slots of optical paths at the optical layer. Against these problems, heuristic methods have generally been used to find the best solution for networks of realistic size. However, in recent years, especially for RSA, research has been conducted on methods that use reinforcement learning (RL) to autonomously train and select optical paths and frequency slots, and it has been reported that these methods can achieve lower blocking probabilities than the benchmark RSA algorithm, the First-Fit method [2].

This work describes a method for applying RL to traffic grooming, in which electrical paths are accommodated into optical paths in dynamic multilayer networks. We show the superiority of the proposed method over heuristic methods, and the challenges and solutions to planning multilayer paths in an integrated manner using electrical and optical layer reinforcement learning will be discussed.

2. Traffic Grooming in Multilayer Path Planning

2.1. *Multilayer Path Planning Framework*

Figure 1 illustrates the multilayer path planning process [3]. Once a new (electrical) path demand is generated, one or more optical paths are assigned to the paths connecting the source and destination nodes. This optical path may groom its electrical paths to use free resources (e.g. tributary slots) of an already established optical path, or it may establish a new optical path for accommodating the electrical paths. One heuristic approach to this multilayer path planning problem is known to be based on auxiliary graphs [3,4]. The planning output is obtained by constructing an auxiliary graph consisting of existing optical paths that can be groomed and candidate optical paths that can be newly established, and then performing a shortest route search on the graph.

The process includes two planning items. The first is the routing, spectrum, and operational mode assignment to be performed for the newly established optical path. When a new optical path is established as a result of multilayer planning, the parameters of the optical path, such as routing, frequency slot, modulation format, symbol rate, etc., are determined. The other is edge weight assignment, which determines whether priority is given to establishing an optical path or grooming an existing optical path.

In this work, we discuss how to apply RL to optimize the latter's edge weights. The weights of the edges $e \in G^{ECG}(E)$ constituting the graph Establish Candidate Graph (ECG), which is composed of candidate optical paths to be established, are set to $W_e^{ECG} = W^{ECG} + h * W_h^{ECG}$, the weight of the groomable established optical edge $e \in G^{GCG}(E)$ of the graph Grooming-Capable Graph (GCG) composed by paths, is $W_e^{GCG} = W^{GCG} + h * W_h^{GCG}$. These four parameters W^{ECG}, W_h^{ECG}, W^{GCG}, and W_h^{GCG} are determined arbitrarily in the heuristic method, but this work determines them based on RL.

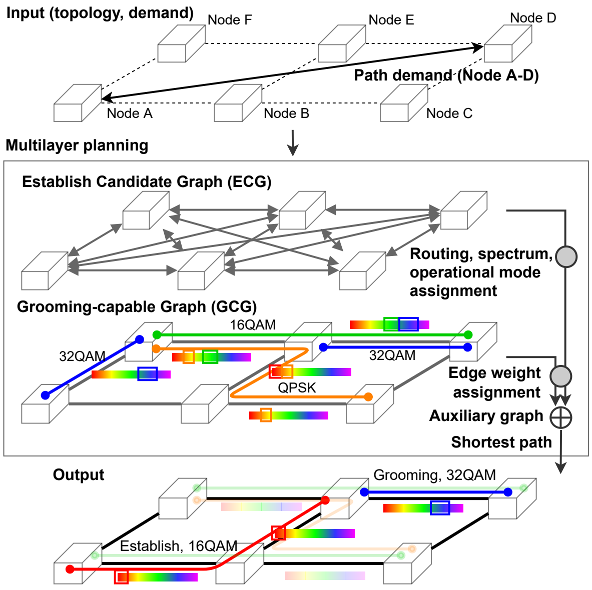


Fig. 1: Multilayer path planning

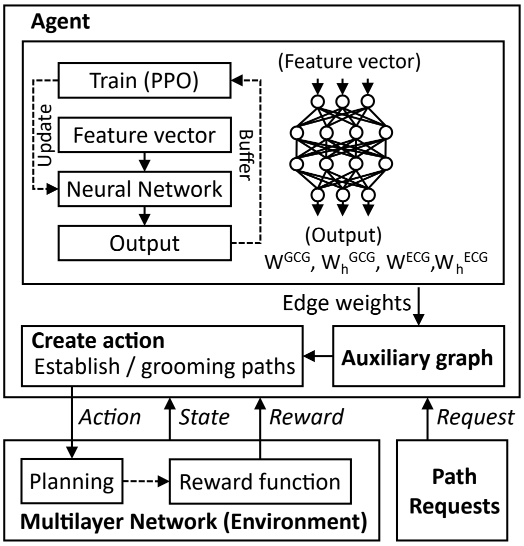


Fig. 2: Reinforcement learning scheme for traffic grooming

2.2. *Traffic Grooming using Reinforcement Learning*

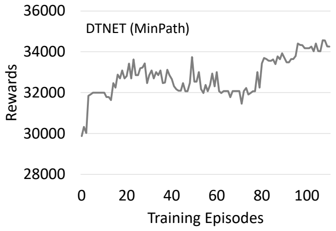
RL is a type of machine learning that iterates interaction between the *agent* and its *environment*. The *agent* acts as the brain that plans multilayer paths, and develops the *agent's* strategies that maximize the *reward*, a measure of how good or bad the planned result is. Figure 2 shows the RL model for determining the edge weights of the auxiliary graph. When a path demand arrives, the *agent* generates a feature vector from the path demand and the current multilayer network for input to the neural network. The feature vector assumed in this work includes a one-hot vector representing the source and destination nodes of path demand and information on the number of hops required per operational mode (0 if unreachable) for each GCG and ECG for all node pairs. In addition, this work employs Proximal Policy Optimization (PPO) [5], a proven RL algorithm using the policy gradient method, as it can directly output four edge weights by means of policy functions processed by neural networks. Based on the output values, an auxiliary graph is constructed and the *action* is determined by running the shortest route algorithm along the graph. After reflecting this *action* in the multilayer network, the reward function evaluates the planning results. Then, the weights of the neural network of the *agent* are updated in accordance with the result of the reward function.

Here, we discuss two advantages of RL over heuristic methods. The first is that it realizes edge weight optimization based on flexible parameter settings. As mentioned earlier, edge weights are determined according to an arbitrarily determined policy, but if this policy is made more flexible, the rule setting becomes more complex. In contrast, in RL, edge weights are automatically optimized for feature vectors that change according to path demand and network conditions. Second, the reward function can directly specify whether the *reward* is good or bad. In addition to blocking probability, delay and power consumption may also need to be taken into account when planning multilayer paths. In the heuristic method, the design considering delay and power consumption was indirectly implemented by adjusting the edge weights, but in RL, these evaluation measures can be directly reflected in the reward function.

3. Simulation Setup and Results

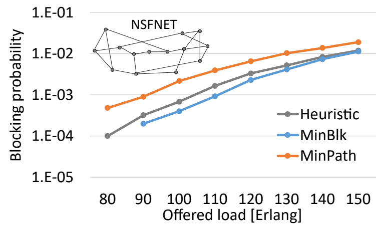
The superior effectiveness of the proposed method using RL over heuristic methods was evaluated in simulations of two topologies: NSFNET (14 nodes, 12 links) and DTNET (14 nodes, 23 links). The number of dynamic path demands evaluated was set to 40000, and bit rates in increments of 100 Gbit/s from 100 Gbit/s to 400 Gbit/s were assumed to arrive between randomly selected node pairs in equal proportions, with the occurrence and duration following Poisson and exponential distributions, respectively.

In the optical layer, the frequency slots available for each link were 12.5 GHz/slot * 160 slots. In the optical layer, k-shortest paths (k=3) for path selection and first-fit for frequency assignment were employed for creating new optical paths. The operational modes that offer bandwidth and transmission distance of optical paths can be selected from the 15 options described in a previous work [3], From among them, the optimal operational mode adopted required determining the bit rate and distance with the highest bandwidth and groomable capacity. These assumptions at the optical layer have been shown to attain lower blocking probabilities in the previous work on

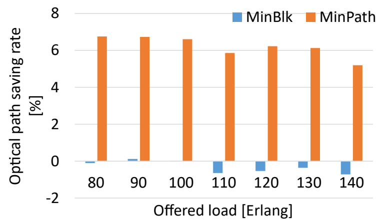


Path requests per episode = 10000
Optimizer = Adam
Optimizer parameters:
  Learning rate = 1e-4, Epsilon = 0.3,
  Weight decay = 1e-3
PPO parameters:
  Batch size = 32, Buffer size = 256,
  Num epoch = 10
Hidden layers: 8192 x 2 layers

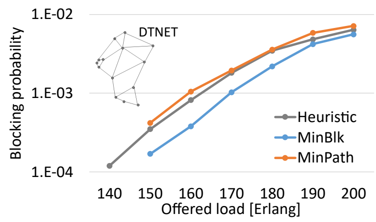
Fig. 3: Training process example and parameter settings



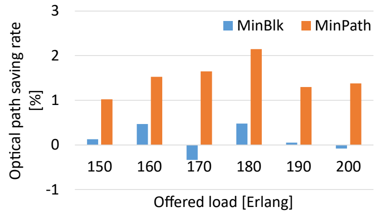
(a) Blocking probability (NSFNET)



(b) Optical path saving rate against heuristic (NSFNET)



(c) Blocking probability (DTNET)



(d) Optical path saving rate against heuristic (DTNET)

Fig. 4: Simulation results

heuristic methods, and the same policy is adopted in this work for the establishment of optical paths for both heuristic methods and proposed RL method.

For the RL model, we consider two reward functions: 1) MinBlk: a policy that minimizes the blocking probabilities, earning +1 for a successful path planning and -1 when blocking occurs; 2) MinPath: a policy that minimizes the number of paths required per path demand, earning 10 / (hop count of established path + hop count of groomed path) for a successful path planning and -1 when blocking occurs. The hyperparameters used for training and an example of how training progresses when MinPath is used in DTNET are shown in Fig. 3. Note that even better results may be obtained in the future by optimizing the hyperparameters and by fully implementing the training process.

Figures 4(a) and 4(c) show the blocking probability against traffic intensity. MinBlk can reduce the blocking probability against the heuristic method, for example, if a path is accommodated with a blocking probability of 1e-3, MinBlk can accommodate about 5 % more path demands in both two topologies, respectively. On the other hand, MinPath is not optimized for blocking probability and has the same or higher blocking probability than the heuristic method. Figures 4(b) and 4(c) show the total number of optical paths used to accommodate dynamically generated path demands. This is a parameter closely related to power consumption and delay. MinPath saves up to 6.7 % and 2.1 % of the number of paths in the two topologies, respectively, versus the heuristic method.

4. Conclusion

This work introduced a dynamic traffic grooming method for multilayer networks that uses RL. One of the advantages of RL is its ability to design multilayer paths with various characteristics by setting the reward function. Our simulations demonstrated that the method outperforms conventional methods in metrics such as blocking probability and power consumption. As mentioned earlier, the optical path planning algorithm, which was a heuristic method used in this work, could be changed to a RL-based method and further performance improvement could be achieved by cooperating the RL models in electrical and optical layers. How to coordinate the optical layer path planning, which aims at efficient use of frequency resource, and the electrical layer path planning, which prefers grooming, is a future challenge.

References

1. A. Mayoral et al., "Unified SDN control and management of disaggregated multi-vendor IP over open optical networks," ECOC 2022.
2. M. Shimoda et al., "Mask RSA: end-To-end reinforcement learning-based routing and spectrum assignment in elastic optical networks," ECOC 2021.
3. T. Tanaka et al., "Impact of operational mode selection and grooming policies on auxiliary graph-based multi-layer network planning," ECOC 2021.
4. S. Zhang et al., "Dynamic traffic grooming in elastic optical networks," IEEE J. Select. Areas Commun., vol. 31, no. 1, pp. 4-12, Jan. 2013.
5. J. Schulman et all, "Proximal policy optimization algorithms," arXiv preprint, arXiv:1707.06347, 2017.