DeepDRAMA: Deep Reinforcement Learning-based Disaster Recovery with Mitigation Awareness in EONs

Rujia Zou*, Nathaniel Bury*, Hiroshi Hasegawa†, Masahiko Jinno‡, Suresh Subramaniam*
*The George Washington University, {rjzou,nbury,suresh}@gwu.edu
†Nagoya University, hasegawa@nuee.nagoya-u.ac.jp
‡ Kagawa University, jinno@eng.kagawa-u.ac.jp

GLOBECOM 2021 - 2021 IEEE Global Communications Conference | 978-1-7281-8104-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/GLOBECOM46510.2021.9685680

***Abstract*—Elastic Optical Networks (EONs) have become a promising solution to satisfy the dramatic growth of bandwidth demand due to 5G and cloud applications. Due to the flexibility of resource allocation, EONs provide high spectrum utilization efficiency, and because of this, developing efficient policies to ensure the survivability of EONs is a challenging problem. A well-designed disaster management plan is needed to prevent data loss during network failures and large-scale disasters. The bottleneck problem caused by disabled parts of the network causes difficulties for disaster recovery. Depending on the disaster, even traffic that may be far away from the disaster may be impacted by it. In this paper, we propose a new approach to disaster management using machine learning to facilitate efficient recovery. In addition to traffic immediately affected by the disaster, all traffic which is "close to" the disaster is re-routed and re-assigned with possibly degraded service, while requests "far from" the disaster are left unaffected. A deep reinforcement learning disaster recovery algorithm with mitigation awareness (DeepDRAMA) is proposed for recovery. A novel deep reinforcement learning agent is designed and trained for the agent to select the appropriate level of service degradation for re-assigned traffic. Simulation results show the performance improvement with DeepDRAMA.**

***Index Terms*—Disaster management, degraded service, machine learning**

I. Introduction

Elastic optical networks (EONs) have arisen as an efficient solution to satisfy growing bandwidth demands due to their flexibility in resource allocation and spectrum assignment [1]. In EONs, network traffic is allocated bandwidth in terms of frequency slots (FS), each of which is 12.5 GHz. The FSs must be continuous and contiguous throughout the entirety of the lightpath [2]. Further, the development of coherent transmission and high-level modulation formats means that an efficient modulation format can be chosen for a lightpath depending on its length. Therefore, the routing and wavelength assignment problem in Wavelength Division Multiplexing (WDM) optical networks has evolved into the Routing, Modulation, and Spectrum Assignment (RMSA) problem.

Survivability is regarded as an important aspect for optical networks. Survivability mechanisms can be divided into protection and recovery [3]. In protection strategy, backup resources are reserved and will be used immediately upon failure. For instance, p-cycle protection is considered to be particularly promising due to high protection efficiency [4] [5] [6]. In restoration or recovery strategy, resources are assigned to affected lightpaths after a network failure happens. Affected lightpaths are re-routed and re-assigned based on the surviving network.

A special case of survivability is disaster management. Disasters, such as earthquakes and hurricanes, may cause large-scale damage to network infrastructure. In this case, the protection strategy is not appropriate because of the excessive amount of redundant resources required to protect against an unlikely large-scale network failure. Therefore, a recovery-based policy is the best solution in this case. Disaster recovery in optical networks has been previously studied. A heuristic traffic recovery algorithm is proposed in [7], where a genetic operator is used to optimize the serving order for failed services. A capacity-constrained maximally spatial disjoint lightpath algorithm is proposed in [8] for EONs. However, in these papers, services far away from the disaster center are inevitably affected by the recovery.

With the rapid development of machine learning, deep reinforcement learning (DRL) has begun to play an important role in network management, and has been shown to provide better performance compared with conventional heuristic algorithms for EONs. The use of machine learning in EONs has been previously studied. [9] proposed Deep-RMSA, a deep Q learning (DQL) agent which was used for path selection in standard RMSA. In [10], a transfer learning framework is proposed, which uses multiple agents that share DNN parameters among each other. In [11], a DRL is proposed to enhance network survivability by providing protection, in which two agents are used to select primary and backup paths. However, DRL has not been applied to disaster management in the literature.

In this paper, we propose a new approach to disaster recovery using a mitigation zone and machine learning. All the traffic inside the mitigation zone is re-accommodated with potentially degraded service (i.e., decreased bit rate) to solve the bottleneck problem caused by a large-scale disaster. This new approach is also based on the intuition that services that are far away from the disaster zone should not be affected during the recovery process. The benefit of mitigation zone has been demonstrated in [12], where we proposed a heuristic recovery algorithm called DRAMA. This work expands upon the use of mitigation zone by using a DQL agent to select a degradation factor for rerouted lightpaths affected by a disaster. A new recovery algorithm called DeepDRAMA is proposed and a DQL agent is designed to select the degradation for each

978-1-7281-8104-2/21/$31.00 ©2021 IEEE

traffic inside the mitigation zone. To the best of our knowledge, this is the first paper that leverages machine learning for disaster management in optical networks. The contributions of our work can be summarized as follows:

- A DQL agent is designed for disaster management.
- A novel deep neural network (DNN) is designed and trained to select the appropriate degradation for each lightpath in different mitigation cases.
- The DeepDRAMA algorithm is proposed for the recovery of affected traffic and re-assignment of traffic inside the mitigation zone with proposed DQL agent.
- Simulation results show the effectiveness of DeepDRAMA.

The rest of the paper is organized as follows. The disaster recovery problem is defined in Section II. The DQL agent is designed in Section III, and the DeepDRAMA algorithm is presented in Section IV. Sample simulation results are given in Section V, and the paper is concluded in Section VI.

II. Problem Statement

A. Network and traffic model

We define the problem as follows. Consider a network $G(N, E)$, where N denotes the node set and E denotes the link set. On each link, there is a pair of fibers in opposite directions. At the time of disaster, there is a set of ongoing lightpaths, T, with pre-assigned resources (routes, spectrum, modulation). Each lightpath is denoted as $t(s, d, w)$, where s and d represent the source and destination nodes, and w represents the lightpath data rate. There are several modulation formats for different spectrum efficiencies and different distance limitations. A lightpath is assigned the highest level modulation possible for the length of its path, and assigned spectrum according to continuity and contiguity constraints.

Consider a circular *disaster zone* $D(C_d, R_d)$, where C_d denotes the center and R_d denotes the radius. We assume the disaster causes any node that lies in the disaster zone and any link with either end node in the disaster zone to fail. If there is no possible path from a lightpath's source node to its destination node, or either source node or destination node is failed after the disaster, the lightpath is considered to be *unrecoverable*.

A *mitigation zone* $M(C_m, R_m)$ is established as the annulus bounded by the circular region with center $C_m = C_d$ and radius $R_d + R_m$, and the disaster zone. The area excluding the disaster and the mitigation zones is denoted as U.

Every lightpath $t \in T$ is considered be in one of the three zones – D, M, or U – determined by where their source/destination nodes lie. If the source and/or destination node lies within D, then we say $t \in D$. In this case, the lightpath is not recoverable. For all recoverable lightpaths, if either source or destination node lies within M, then we say $t \in M$; else, $t \in U$.

Based on different areas, all the lightpaths can be divided into 5 types and managed in different ways (examples are shown in Fig. 1):

- If $t \in M$ is affected by the disaster, the lightpath will be recovered with a new path and frequency slots with degradation. This case is shown as t_5.
- If $t \in M$ is not affected by the disaster, the lightpath will be re-assigned with degradation in order to re-organize the spectrum to solve the bottleneck issue. This case is shown as t_2.
- If $t \in U$ is affected by the disaster, the lightpath will be recovered with its original data rate without degradation. This case is shown as t_3.
- If $t \in U$ is not affected by the disaster, the lightpath will not be touched at all. This case is shown as t_1.
- If the lightpath is unrecoverable, the lightpath will be dropped without recovery attempt. This case is shown as t_4.

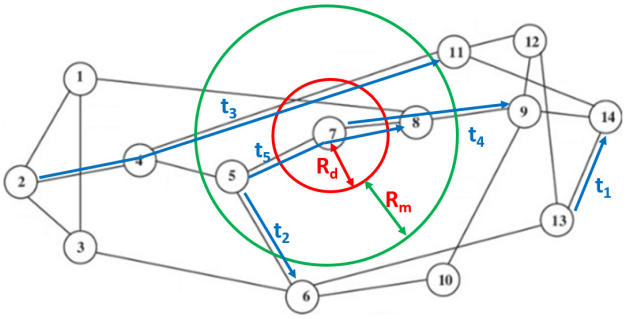

Fig. 1. Examples of LPs inside and outside the disaster and mitigation zones.

B. Penalty function

Since each lightpath (LP) brings revenue to the network, we assume that the revenue is equal to the data rate of the LP (arbitrary units) in this paper. If the data rate of a LP is degraded (or dropped) after disaster, we assume a penalty function to model the lost revenue [13]. The penalty function $P(df)$ is a non-decreasing function of the degradation factor df. In this paper, we assume the following function [12], which is also plotted in Fig. 2.

$$P(df) = \frac{\log(1 - 0.9 \times df)}{\log(1 - 0.9 \times 1)}. \tag{1}$$

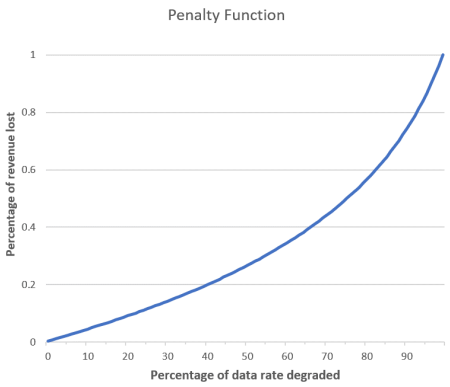

Fig. 2. The penalty function.

The penalty function shows the relationship between the percentage of the revenue lost and the percentage of reduction

in the LP's data rate. In (1), df is the percentage of the data rate degraded. The value of the penalty function is 0 for no degradation, while the value is 1 for full degradation. The absolute value of the penalty is the value of the penalty function times the total revenue. For example, suppose the original data rate of a LP is 400 Gbps and the LP is recovered with 300 Gbps. In this case, the percentage of data rate degraded is 25%. Then the value of the penalty function is 0.11 and the absolute penalty is 0.11 * 400 = 44. If a LP is blocked/dropped during the recovery or re-assignment, all the revenue is considered to be lost and the absolute penalty is equal to the revenue.

The objective of the disaster management problem is to accommodate the recoverable LPs (LPs that need recovery or re-assignment) and minimize the total penalty.

This problem is challenging since it is very hard to find the appropriate degradation for every single lightpath. In [12], we presented a heuristic algorithm to find the appropriate degradation factors and do the Routing and Spectrum Assignment. In this work, we leverage deep reinforcement learning to solve this problem.

III. Deep Q learning design

Reinforcement learning is widely used in scenarios that need to interact with the environment. The general process of reinforcement learning is: given the state of an environment, the agent selects a corresponding action according to some policy. The state will be updated to a new state after executing this action. The agent will get a reward after executing the action, and will adjust its policy according to the value of the reward. The objective is to maximize the sum of the rewards when the state reaches the terminal state. Q-learning is a type of reinforcement learning in which the agent stores the reward value of state-action pairs in a table (Q-table). However, conventional Q-learning is limited by the space of states and actions. When the number of state-action pairs becomes very large, using a Q-table becomes impractical.

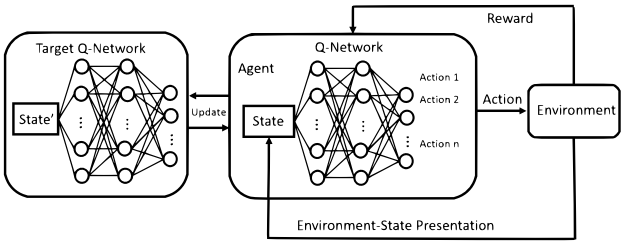

Fig. 3. The structure of DQL agent.

Deep Q Learning (DQL), which is a typical case of deep reinforcement learning and a combination of deep learning and Q learning, replaces the Q-table with a deep neural network (Q-network), and acts as the policy for action selection. Fig. 3 shows the structure of DQL. The environment is presented as the state, and the Q-network is used to determine the action. Once the action is selected, the environment is changed according to the action execution. Meanwhile, a reward is generated and fed back to the Q-network for training. A target Q-network is used to stabilize learning and establish an optimal policy. The purpose of the target neural network is to calculate the target Q values, which are used as the training label of the main Q-network. The target Q-network is updated with the parameters of the main Q-network periodically.

DQL is well-suited to the disaster recovery problem because it does not require a labeled training data set, and attempts to obtain training samples through continuous interaction between the agent and the environment.

A. Q-network design

We now describe the structure of the Q-network inside the DQL agent, including the presentation of the environment and the DNN design.

The observation space $state$ includes three sub-states: network state s, lightpath state t and mitigation zone state mz. The network sub-state s is represented by a 2-D array, where the information of the FS usage is carried. The row represents the FS index and the column represents the fiber index. For example, if FS 3 on fiber 1 is assigned and occupied by a lightpath, then the value of row 1 and column 3 is set to 1. The value is set to 0 if the corresponding FS is not occupied. The lightpath sub-state t is represented as a 1-D array of 4-tuples, where the source node S, destination node D, data rate R and modulation factor M are included. The modulation factor M is the normalized spectrum efficiency of the highest modulation format available (defined as data rate per FS (in Gbps) / 12.5, e.g., the value of M for 8QAM is 3) to be used for the shortest path in the surviving network. The mitigation zone sub-state mz is represented as a 1-D array. The value is set to 1 if the corresponding node is inside the mitigation zone, otherwise it is set to 0. For example, the value of index 2 is set to 1 if node 2 is inside the mitigation zone.

Layer 1 of the DNN is generated by convolution of t and each unit in s as follows:

$$l_2^{i,j} = f(w \cdot [l_1^{i,j}, t_S, t_D, t_R, t_M]^T + b) \tag{2}$$

where $l_2^{i,j}$ is the value of row i and column j in layer 2 and $l_1^{i,j}$ is the value of the position in layer 1. t_S, t_D, t_R, t_M are the lightpath's source node, destination node, data rate and modulation factor, respectively. w and b are the weight of the kernel and bias. Function f is the activation function.

After convolution between s and t, Layer 2 is generated by standard convolution with n_k kernels and biases. The kernel size is 3×3 in this step. Then, layer 3 is generated by doing max pooling to layer 2 with strides size 2×2. Layer 4 is generated with one more convolution and max pooling calculation, same as layers 1 through 3. Layer 5 is generated by the convolution of each column in layer 4 with n_k kernels. Then, the dense layer D1 is generated by the convolution of all the elements on each channel and flatten reshaping.

Dense layer D1 and D2 are full-connected while D2 and the upper part of D3 are full-connected. The mitigation zone sub-state mz is flatten reshaped and concatenated to layer D3 as the bottom part. Then, dense layer D4, D5 and action layer are full-connected.

In this work, the size of the action space is 21×1, corresponding to 21 degradation options (0 % to 100 % in 5% steps). Action value 1 represents 0% degradation and action 21 is 100% degradation (i.e., drop the lightpath). The action with the highest value is selected as the degradation. For example,

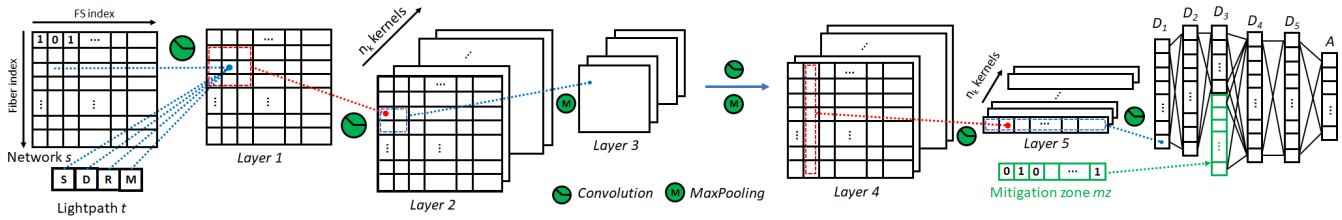

Fig. 4. The structure of Q-network in DQL agent.

if the original data rate of a lightpath is 400 Gbps and action value 15 has the highest value in the action space, then the value the lightpath is recovered with is $400 \times (1-70\%) = 120$ Gbps.

B. Training design

The training of the DNN aims to adjust the parameters to minimize the gap between Q-value of the action generated by the state and the training label. The Q-value is defined as follows:

$$Q(state, A) = r + \gamma E\left[\max_{A'} Q'(state', A')\right], \qquad (3)$$

where $state$ is the current state, including the network sub-state, lightpath sub-state and mitigation zone sub-state. The action selected by the DNN is denoted as A, and r is the immediate reward of the action. The reward is defined as the remaining revenue of each lightpath, which is the total revenue of the lightpath minus the absolute penalty due to the degradation. γ is the discount factor, which is 0.95 in this paper. Q' represents the target Q network. $state'$ is the updated state after the recovery operation according to action A. A' is the future action selected based on future states. In this work, we randomly select 5 of the lightpaths waiting to be recovered or assigned as the future lightpaths. The average Q-value of these 5 lightpaths is used.

The mean squared error is used as the loss function and the Adam optimizer is used [14]. $ReLU$ is used as the activation function. The memory replay mechanism is implemented. Instead of training the Q-network immediately after the recovery or re-assignment, the training data is restored in the memory pool as a group of $state$, $action$, $reward$, $state'$, where $state'$ represents the next state after the $action$ is applied on $state$. Then the training data is periodically randomly selected from the memory pool and used for training. The reason for adding experience replay is because the states are obtained from consecutive environments in the recovery procedure [11]. Compared with conventional reinforcement learning problem, the samples are much more relevant. If there is no experience replay, the training basically does the gradient descent in the same direction for a continuous period of time, so the direct calculation of the gradient may not converge under the same training times. The design of the memory replay is presented in Section V.

IV. DeepDRAMA

We now describe the DeepDRAMA algorithm. The pseudocode of DeepDRAMA is shown in Algorithm 1.

In lines 1-8, we release the spectrum that was assigned to unrecoverable LPs as well as LPs that are waiting to be recovered or re-assigned. The LPs that can be recovered and re-assigned are added to set T'.

In lines 9 -12, all the LPs in T' and U are recovered first. These LPs are sorted in decreasing order of data rate, and each LP is recovered one by one in this order. The shortest path (SP) and first fit (FF) slot assignment is used to accommodate the lightpath.

In lines 13-16, we recover and reassign the lightpaths inside the mitigation zone. The $Agent$ is used to select the degraded data rate of the lightpath.

Algorithm 1 DeepDRAMA Algorithm

Require: $G(N, E)$, T, $D(C_d, R_d)$, $M(C_m, R_m)$, U, $Agent$
Ensure: Degradation factor and RMSA for recovered LPs

1: Initialize an empty LP set T'
2: **for** each $t \in T$ **do**
3: **if** $t \in D$ (i.e., t is unrecoverable) **then**
4: Release the spectrum of t
5: **else if** $t \in M$ **or** t's path is disrupted by disaster **then**
6: Release the spectrum of t, add t to T'
7: **end if**
8: **end for**
9: Sort all $t \in T'$ in decreasing order of data rate
10: **for** each $t \in T'$ **do**
11: **if** $t \notin M$ **then**
12: Assign t with SP-FF RSA without degradation; block t if FSs not available
13: **else**
14: Determine the modulation format with SP
15: Select the degradation option with the $Agent$
16: Assign t with SP and FF with selected degradation; block t if FSs not available
17: **end if**
18: **end for**

V. Simulation Results

In this section, we present and analyze the performance of DeepDRAMA and the training results of DQL. The network topologies used are the COST239 network (11 nodes and 26 links [6]) and the NSF network (14 nodes and 21 links [15]). There is a pair of fibers in opposite directions and there are 100 FSs on each fiber.

Three different disasters are tested, as shown in Table I. Before the disaster, a set of unidirectional traffic requests is

TABLE I
DISASTER SCENARIOS FOR EXPERIMENTS.

Center	Affected links
Node 7 in NSF	1-8, 5-7, 7-8, 4-11
Node 2 in NSF	1-2, 1-3, 2-3, 2-4
Node 6 in COST	0-3, 3-6, 5-6, 8-6, 9-6 ,3-9

generated first. The source and destination nodes for each traffic request are uniformly randomly selected from the nodes. We assume there are three different types of requests with rate 40/100/400 Gbps with probability 0.3, 0.5, and 0.2, respectively. Four types of modulation formats are used: 16-QAM, 8-QAM, QPSK and BPSK. The number of required FSs for a LP is determined by its data rate and modulation format. The number of FSs corresponding to different data rates and different modulation formats are shown in Table II. For each modulation format, the physical distance limitations are also shown in Table II.[1]

TABLE II
REQUIRED FSS AND DISTANCE LIMITATIONS [16].

Modulation \ Date Rate	40G	100G	400G
16-QAM (500 km)	1	2	8
8-QAM (1000 km)	2	3	11
QPSK (2000 km)	2	4	16
BPSK (>2000 km)	4	8	32

The required numbers of FSs for a given modulation format is calculated as follows:

$$F = \left\lceil \frac{w}{ModE_m} \right\rceil, \tag{4}$$

where w is the data rate of the lightpath, $ModE_m$ is the spectrum efficiency of modulation format m (defined as data rate per FS, e.g., BPSK is 12.5 Gbps and 16QAM is 50 Gbps) used for the lightpath.

A. Training and testing

For the training, in each episode, 200 lightpaths are generated and assigned a path and FSs (if available) with shortest path-first fit (SP FF). The DeepDRAMA algorithm is operated with the current $Agent$ for different mitigation zone cases once (from the smallest mitigation zone case to the case of the mitigation zone being the entire network). For each lightpath, the group of $state$, $action$, $reward$, $state'$ is stored in the memory pool. We randomly select 10 batches with batch size 16 and do the batch training of DNN in the $Agent$ [9]. The target Q network is updated after the batch training at the end of each episode. We train the $Agent$ with 1000 episodes. The total number of lightpaths used for training is approximately 160k.

During the training, the ϵ-greedy strategy is used. The action with highest Q value is selected with probability $1-\epsilon$; else the agent selects a random action for exploration. The value of ϵ is set to 1 at the beginning of the training and updated with $\epsilon = \epsilon \times \epsilon_{\text{decay}}$ after each batch training until $\epsilon < \epsilon_{\min}$. The value of ϵ_{decay} and $\epsilon_{\min}$ are set to 0.99 and 0.001, respectively.

In the testing phase, as in training, the number of lightpaths before the disaster is set to 200. DeepDRAMA with trained agent is used to recover and re-assign lightpaths. Average results over 30 trials are shown with 95% confidence interval.

[1]We assume that there is no physical distance limitation for BPSK.

B. Training results

Before the training process starts, 20 traffic sets of 200 lightpaths each are generated; these sets are not used in the training process. As the training progresses, we apply (the partially trained) DeepDRAMA to recover these lightpaths and observe the penalty. The evolution of the average penalty (over these 20 traffic sets) as the training progresses (i.e., number of episodes increases) is shown in Fig. 5 for different mitigation zone cases. As we can see, the penalty rapidly decreases and becomes stable after about 100 episodes of training. (We only show the first 400 episodes here since since the penalty has already stabilized. The actual training is done for 1000 episodes. It takes about 30 hours to finish all the training.)

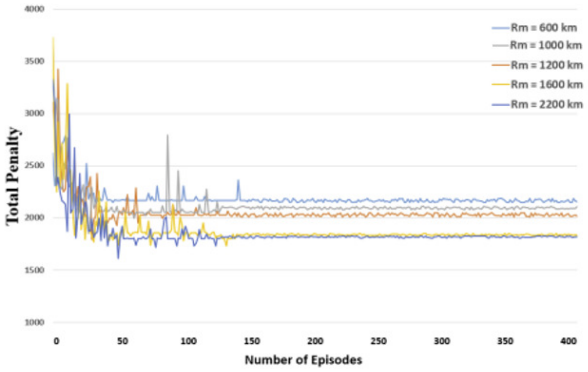

Fig. 5. Total penalty when disaster happens at node 7 in NSF network.

C. Testing results

The testing results for NSFnet are shown in Fig. 6 and 7. The penalty when there is no mitigation zone is shown as a dashed line. In this case, only affected lightpaths are recovered by SPFF without degradation. In the no degradation case, lightpaths inside the mitigation zone are reassigned but no degradation is applied. Compared with baseline cases, DeepDRAMA provides the lowest penalty. The penalty decreases as the mitigation zone expands, which shows that DeepDRAMA is able to exploit the flexibility offered by the mitigation zone.

The testing results for COS239 network are shown in Fig. 8. As we can see, DeepDRAMA is better than baseline cases but the performance improvement is lower than in NSF network. The reason is that COS239 is a smaller network and all the lightpaths select a higher level modulation, which means the average number of slots required is lower. In this case, it is hard to gain advantage from lightpath re-accommodation due to spectrum segmentation. Interestingly, the total penalty for the no degradation case is larger than that for the no mitigation case when the mitigation zone is small. This provides evidence that appropriate degradation selection is necessary even the mitigation zone is implemented.

VI. CONCLUSION

Disaster management is an important aspect of elastic optical networks. Conventional disaster recovery algorithms

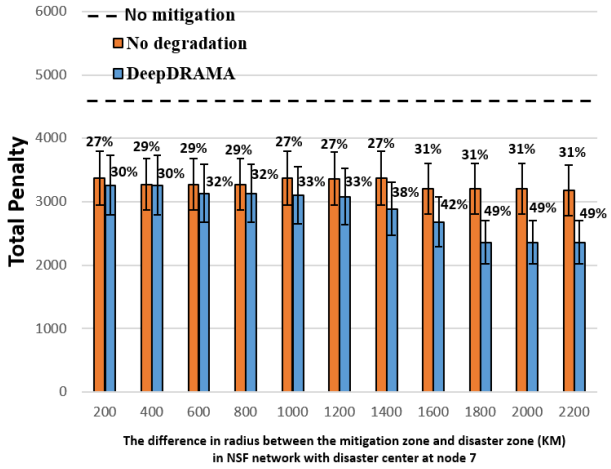

Fig. 6. Total penalty when disaster happens at node 7 in NSF network.

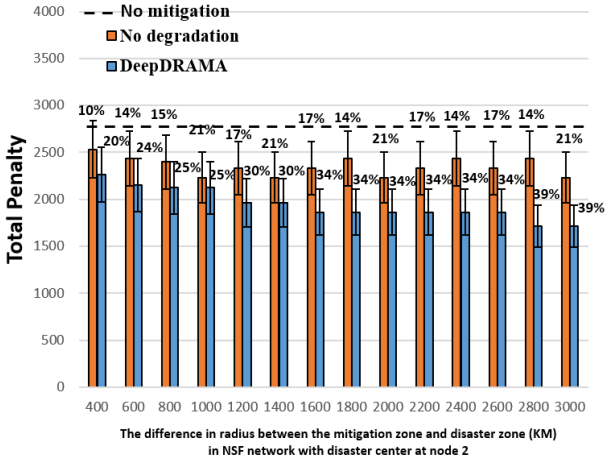

Fig. 7. Total penalty when disaster happens at node 2 in NSF network.

attempt to recover the affected traffic using the resources of the surviving network. In this paper, we use the previously proposed idea of a mitigation zone which defines an area around the disaster zone within which service may be recovered in a degraded manner. We leverage deep Q learning (DQL) to select the appropriate level of service degradation. Results demonstrate the performance improvement that can be achieved with DQL. In the future, we intend to apply DQL to the joint problem of re-assignment of path and spectrum in addition to selecting the degradation level for affected traffic.

ACKNOWLEDGEMENT

This work was supported in part by NSF grant CNS-1818858.

REFERENCES

[1] M. Jinno, H. Takara, B. Kozicki, Y. Tsukishima, Y. Sone, and S. Matsuoka, "Spectrum-efficient and scalable elastic optical path network: architecture, benefits, and enabling technologies," *IEEE communications magazine*, vol. 47, no. 11, pp. 66–73, 2009.

[2] Y. Hirota, H. Tode, and K. Murakami, "Multi-fiber based dynamic spectrum resource allocation for multi-domain elastic optical networks," in *2013 18th OptoElectronics and Communications Conference held jointly with 2013 International Conference on Photonics in Switching (OECC/PS)*, June 2013, pp. 1–2.

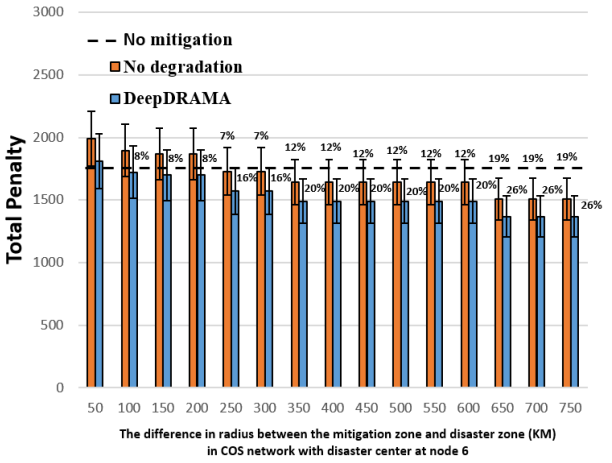

Fig. 8. Total penalty when disaster happens at node 6 in COS239 network.

[3] G. Shen, H. Guo, and S. K. Bose, "Survivable elastic optical networks: survey and perspective," *Photonic Network Communications*, vol. 31, no. 1, pp. 71–87, 2016.

[4] R. Zou and S. Subramaniam, "Novel p-cycle selection algorithms for elastic optical networks," in *ONDM 2019 - 23rd International Conference on Optical Network Design and Modeling (ONDM 2019)*, Athens, Greece, May 2019.

[5] R. Zou, S. Subramaniam, H. Hasegawa, and M. Jinno, "P-cycle design for translucent elastic optical networks," in *2019 IEEE Global Communications Conference*, Waikoloa, USA, Dec 2019.

[6] R. Zou, H. Hasegawa, M. Jinno, and S. Subramaniam, "Link-protection and FIPP p-cycle designs in translucent elastic optical networks," *Journal of Optical Communications and Networking*, vol. 12, no. 7, pp. 163–176, 2020.

[7] S. Li, R. Gu, G. Zhang, Y. Wang, Y. Wang, and Y. Ji, "Order aware service recovery algorithm in elastic optical network with multiple failures," in *2019 International Conference on Networking and Network Applications (NaNA)*. IEEE, 2019, pp. 135–141.

[8] A. Waqar, S. Idrus, R. Butt, and F. Iqbal, "Post-disaster least loaded lightpath routing in elastic optical networks," *International Journal of Communication Systems*, vol. 32, no. 8, p. e3920, 2019.

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

[10] X. Chen, R. Proietti, C.-Y. Liu, Z. Zhu, and S. B. Yoo, "Exploiting multi-task learning to achieve effective transfer deep reinforcement learning in elastic optical networks," in *Optical Fiber Communication Conference*. Optical Society of America, 2020, pp. M1B–3.

[11] 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, no. 6, pp. 7896–7911, 2019.

[12] R. Zou, H. Hasegawa, and S. Subramaniam, "DRAMA: disaster management algorithm with mitigation awareness for elastic optical networks," in *2021 17th International Conference on the Design of Reliable Communication Networks (DRCN 2021)*, Apr. 2021.

[13] J. Zhao and S. Subramaniam, "QoT-and SLA-aware survivable resource allocation in translucent optical networks," in *2015 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2015, pp. 1–6.

[14] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[15] X. Chen, S. Zhu, L. Jiang, and Z. Zhu, "On spectrum efficient failure-independent path protection p-cycle design in elastic optical networks," *Journal of Lightwave technology*, vol. 33, no. 17, pp. 3719–3729, 2015.

[16] C. Wang, G. Shen, and S. K. Bose, "Distance adaptive dynamic routing and spectrum allocation in elastic optical networks with shared backup path protection," *Journal of Lightwave Technology*, vol. 33, no. 14, pp. 2955–2964, 2015.