2022 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS) | 978-1-6654-7340-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/ANTS56424.2022.10227763

Dynamic network simulation using DeepRMSA in Elastic Optical Networks

Boyang Hu, Mohammad Jafar Majid and Byrav Ramamurthy
School of Computing, University of Nebraska-Lincoln, Lincoln, NE 68588-0115, USA
Email: {bhu,mmajid,byrav}@cse.unl.edu

***Abstract*—In optical networks, simulation is a cost-efficient and powerful way for network planning and design. It helps researchers and network designers quickly obtain preliminary results on their network performance and easily adjust the design. Unfortunately, most optical simulators are not open-source and there is currently a lack of optical network simulation tools that leverage machine learning techniques for network simulation. Compared to Wavelength Division Multiplexing (WDM) networks, Elastic Optical Networks (EON) use finer channel spacing, a more flexible way of using spectrum resources, thus increasing the network spectrum efficiency. Network resource allocation is a popular research topic in optical networks. In EON, this problem is classified as Routing, Modulation and Spectrum Allocation (RMSA) problem, which aims to allocate sufficient network resources by selecting the optimal modulation format to satisfy a call request. SimEON is an open-source simulation tool exclusively for EON, capable of simulating different EON setup configurations, designing RMSA and regenerator placement/assignment algorithms. It could also be extended with proper modelings to simulate CapEx, OpEx and energy consumption for the network. Deep learning (DL) is a subset of Machine Learning, which employs neural networks, large volumes of data and various algorithms to train a model to solve complex problems. In this paper, we extended the capabilities of SimEON by integrating the DeepRMSA algorithm into the existing simulator. We compared the performance of conventional RMSA and DeepRMSA algorithms and provided a convenient way for users to compare different algorithms' performance and integrate other machine learning algorithms.**

***Keywords*—Simulator for elastic optical networks (SimEON), Deep Reinforcement Learning, Routing, Modulation and Spectrum Allocation (RMSA) and Dynamic Network Simulation.**

I. INTRODUCTION

Elastic Optical Networks (EON) descend from traditional Wavelength Division Multiplexing (WDM) networks and provide better spectrum allocation flexibility and scalability. EON is designed to handle heterogeneous bandwidth demands. The resource allocation problems are classified as Routing and Spectrum Allocation (RSA) and Routing, Modulation and Spectrum Allocation (RMSA) problems, similar to Routing and Wavelength Allocation (RWA) in the WDM network. These problems include finding a suitable route for a network lightpath to traverse while assigning a sufficient spectrum based on the selected modulation format. Spectrum allocation must abide by the spectrum continuity and contiguity constraints in EON. The spectrum continuity constraint ensures that the exact frequency slots of spectrum allocated for a lightpath must be the same for all network links associated with the lightpath. As for the spectrum contiguity constraint, all frequency slots of the spectrum allocated must be adjacent to the lightpath's network links. These constraints must be satisfied to allocate spectrum to a requested lightpath properly, or the request will be blocked.

Machine Learning techniques have been extensively studied and applied in different fields and proved efficient in solving complex problems. Deep learning (DL) is an area of machine learning that aims to mimic the behavior of the human brain using neural networks with three or more layers and "learn" from large quantities of data. At the same time, Reinforcement Learning (RL) uses agents to interact with the environment and learn optimal decisions based on continuous feedback to maximize a reward. Deep Reinforcement Learning (DRL) combines DL and RL, which leverages deep neural networks instead of the traditional Q-table used in regular RL to solve complex problems that cannot be dealt with using classic RL. Optical network researchers have used ML techniques in different topics, including optical network control and resource management, network monitoring and survivability.

Optical networking researchers have had extensive research on routing and resource management. There are mainly two types of conventional routing algorithms: adaptive and non-adaptive. The Minimum Hops (MH) and Shortest Path (SP) are two examples of non-adaptive algorithms derived from Dijkstra's minimum cost algorithm. The route is chosen based only on the topology (minimum number of nodes/minimum distance) and does not take into account the network conditions. The adaptive routing algorithms, however, make routing decisions based on topology and network traffic dynamically. The authors in [1] proposed an adaptive RSA algorithm that uses sequential search (sequential fitting) and adaptive routing to determine the near-optimal use of spectrum based on the history of established connections. Length and Occupation Routing (LOR) and Power Series Routing (PSR) proposed in [2] consider the network load and are capable of load balancing between the links, which effectively lower the blocking probability. Similar to RSA algorithms, RMSA algorithms are proposed to solve the resource allocation problem in an EON scenario.

SimEON [3] is an open-source EON simulation tool designed to simulate transparent, translucent and opaque elastic optical networks. It provides different routing and wavelength assignment algorithms by default and is capable of simulating the network blocking probability under dynamic traffic conditions with/without regenerators in the network. We intend to enable SimEON to implement machine learning algorithms in simulations and provide an integrated simulation interface with a seamless experience for users. Researchers can easily simulate and compare the performance of conventional RMSA algorithms and machine learning algorithms in one place. Our contributions are the following. 1) Integration of SimEON and DeepRMSA algorithm. 2) Provide an option for users to easily find out the best combination of algorithms for their network topology and network load in terms of blocking probability. 3) Run extensive experiments and compare different conventional and DRL algorithms' performance and 4) Identify the scenario where DRL outperforms the traditional algorithm.

The rest of the paper is organized as follows: In Section II, we provide a background of EON, DL and RMSA problems and discuss some EON simulators in the community. In Section III, we detail how we integrate SimEON and DeepRMSA. We present our experimental setup in Section IV and the results on different network scenarios in Section V. Section VI concludes our work and we discuss our future work in Section VII.

II. Background and Related Work

Simulation tools play an important role in optical network design as access to real testbeds is often limited. Building a physical testbed for experiments usually requires vast investments, is challenging to adjust and is inconvenient to test new features or algorithms. Currently, to our knowledge, no simulator provides both classical methods and machine learning methods to simulate the RMSA problem in EON.

There are few EON simulators available in the community. The work in [4] is an elastic networks OMNeT++ based simulator. The uniqueness of this tool is that it can increase or decrease the amount of spectrum allocated to a given light path. The work in [5], can work on both static and dynamic traffic scenarios. It solves the RSA, RMSA and regenerator placement problem. The work in [6] is capable of working with fragmentation and defragmentation in dynamic network scenarios. Net2Plan [7] is extensively used for WDM optical networks design and network performance simulation. While the work in [8] is an extension to Net2Plan, it aims to allocate resources equitably depending on service data rates, ensuring a similar blocking performance for different services in the EON context.

The RMSA algorithm was first presented by [9], which examined the routing problem in EON and showed the benefits of incorporating slices' status and spectral efficiency into the path computation function. Since then, numerous RMSA algorithms have been developed to solve the famous problem from different perspectives. The Joint RMSA algorithm and Separate RMSA algorithm [10] are proposed to support the traffic multicasting in EON, based on Integer Linear Programming (ILP) formulation. Though their accuracies are high, the computational complexities made these approaches not scalable to large problems. Thus, Heuristic RMSAs [11]–[13], which are based on a genetic algorithm (GA) [14] was proposed to make the computation feasible and support both static and dynamic traffic. The authors in [15] developed RMSA algorithms that combine the metric of path disaster availability with the metric of spectrum usage. Simulated results show that their proposed algorithms provide both efficient spectrum utilization and resilience in the event of multiple node failures. DeepRMSA [16] is the first RMSA framework in EON that leverages deep reinforcement learning. The Deep neural networks (DNN) in the framework are trained progressively with experiences from dynamic lightpath provisioning to learn the optimal online RMSA policies.

A. SimEON

SimEON [3] is a simulator for elastic optical networks written in C++. It generates dynamic traffic following Poisson process and builds physical device models including optical fibers, optical amplifiers, electronic regenerators, splitters and switching components. Physical impairments, capital expenditure (CapEx), operational expenditure (OpEx) and energy consumption are also considered in the network simulation.

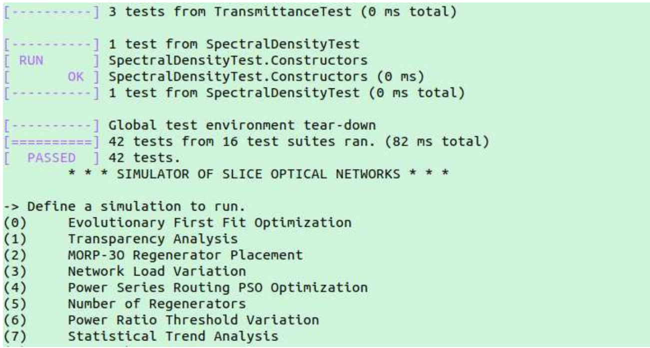
SimEON is designed to simulate transparent, translucent and opaque elastic optical networks. The transparent optical networks do not involve electronic regenerators. The translucent optical networks mean some nodes can convert an optical signal to an electronic domain and back to the optical domain after re-amplifying and reshaping. Opaque implies all network nodes have plenty of electronic regenerators. The transparent network scenario involves having one transparent segment route for all network links associated with the lightpath. The same principles of RMSA along with other functionality apply to the one segment used for the lightpath. A translucent optical network breaks the route into multiple transparent segments. In this case, each segment route conducts RMSA and other functionalities in the simulation.

SimEON is a versatile simulation tool which can be used for designing routing, spectrum assignment, regenerator placement and regenerator assignment algorithms. In addition, it can investigate the impacts of several network parameters on the network performance.

During a simulation, a call request will be blocked if there is a lack of resources or if the quality of transmission is deficient. The amount of call requests is defined by the user. SimEON's modular architecture design facilitates the process of extending its capabilities. Fig. 1 shows the original SimEON command line interface.

B. DeepRMSA

DeepRMSA [16] is the first RMSA algorithm that leverages deep reinforcement learning to solve the resource allocation problem in EON. It utilizes powerful DRL techniques upon a software-defined networking architecture. DRL employs agents and DeepRMSA uses actor-learners related to the agents used in DRL. Actor-learners interact with the environment and make decisions based on the policy generated by the Deep neural networks (DNN). DNN plays a significant role in DRL as it takes in input data featured engineered. The input data traverses through the hidden layers of the



```
[----------] 3 tests from TransmittanceTest (0 ms total)

[----------] 1 test from SpectralDensityTest
[ RUN      ] SpectralDensityTest.Constructors
[       OK ] SpectralDensityTest.Constructors (0 ms)
[----------] 1 test from SpectralDensityTest (0 ms total)

[----------] Global test environment tear-down
[==========] 42 tests from 16 test suites ran. (82 ms total)
[  PASSED  ] 42 tests.
        * * * SIMULATOR OF SLICE OPTICAL NETWORKS * * *

-> Define a simulation to run.
(0)     Evolutionary First Fit Optimization
(1)     Transparency Analysis
(2)     MORP-3O Regenerator Placement
(3)     Network Load Variation
(4)     Power Series Routing PSO Optimization
(5)     Number of Regenerators
(6)     Power Ratio Threshold Variation
(7)     Statistical Trend Analysis
```

Fig. 1: The Original Menu Option of SimEON interface

neural networks, where it would then output a policy. The policy is a probability distribution of an action given by the state. Policies dictate what action an actor-learner should take. The goal of DeepRMSA is to find the most efficient RMSA policy for each requested lightpath and maximizes the overall rewards received. The actor-learner continues to learn through a reward system. The modeling components of DeepRMSA include the state, action, reward and training. The state includes all the information related to a request and the current network environment; for example, the amount of spectrum frequency slots a request needs, the amount available, its position, possible routing paths, etc.

DeepRMSA will perform a specific action based on the received state data. The goal of the action is to satisfy the RMSA properties generated for a lightpath. The reinforcement framework allows actor-learners to learn by a reward received based upon the particular action taken. Rewards will enable actor-learners to make more suitable decisions. These modeling components have a significant role in the training phase. Essentially, the training phase improves the performance of DeepRMSA by tuning the DNN. The outcomes of the modeling components are all stored in a buffer. Once the experience buffer reaches a certain threshold, it uses the experience from previous sample requests, train the DNN further and improves the overall policy in a window-based method. The policy will continue to be optimized as more data from the experience buffer is applied to the DNN. When the experience buffer relinquishes the data, the process starts again.

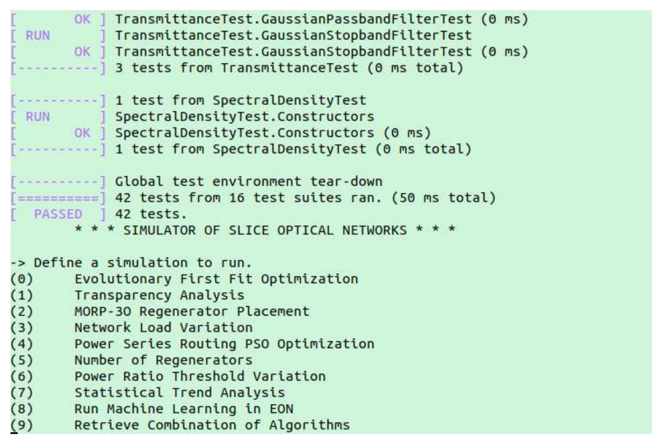
III. Integration of SimEON and DeepRMSA

In this section, we discuss how we integrate the DeepRMSA algorithm into SimEON and other features that we provide to enhance the user experience.

SimEON is written in C++ and engineered modularly, simplifying our process of extending its capabilities. Figure 2 shows the new menu options we created in SimEON. The new menu is engrained in SimEON and its compilation is adjusted to be the same as the previous menu options. The first new menu option (8) is the integration with DeepRMSA algorithm. We leverage the EON environments created by Optical-RL-Gym [17] and incorporated its DRL algorithms into SimEON. The components related to this new feature, such as the classes and functions we created for it are modified to undergo the DRL simulation. In order to use the enhanced SimEON tool, users need first to download the Optical-RL-Gym and install the necessary dependencies.

In our integration process, the initial challenge is to ensure that both SimEON and DeepRMSA utilize the same topology. SimEON and Optical-RL-Gym use different formats for the network topology files. To smooth the user experience, we created a script that allows users to define their network topology (nodes and links information) and automatically convert it into two topology files that SimEON and Optical-RL-Gym would understand. Since SimEON and Optical-RL-Gym use the same dynamic traffic generation method, there is no need to make changes. A sample DeepRMSA simulation output is shown in Fig. 3.

Another feature we added to SimEON is named Retrieve Combination of Algorithms based on the user-defined network topology, network load and the number of calls. The simulator will automatically try all the combinations of conventional routing and spectrum assignment algorithms and provide users with the best and worst possible combinations in terms of the blocking probability. Such an option enhances the user experience by saving the experiment setup time, organizing the results in structured folders and providing the users with the most interesting results at one time.

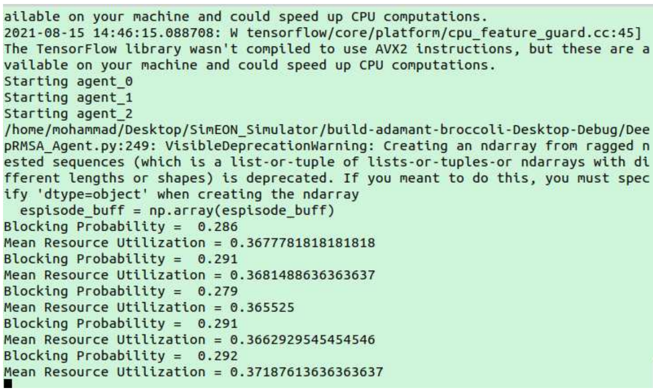


```
[       OK ] TransmittanceTest.GaussianPassbandFilterTest (0 ms)
[ RUN      ] TransmittanceTest.GaussianStopbandFilterTest
[       OK ] TransmittanceTest.GaussianStopbandFilterTest (0 ms)
[----------] 3 tests from TransmittanceTest (0 ms total)

[----------] 1 test from SpectralDensityTest
[ RUN      ] SpectralDensityTest.Constructors
[       OK ] SpectralDensityTest.Constructors (0 ms)
[----------] 1 test from SpectralDensityTest (0 ms total)

[----------] Global test environment tear-down
[==========] 42 tests from 16 test suites ran. (50 ms total)
[  PASSED  ] 42 tests.
        * * * SIMULATOR OF SLICE OPTICAL NETWORKS * * *

-> Define a simulation to run.
(0)     Evolutionary First Fit Optimization
(1)     Transparency Analysis
(2)     MORP-3O Regenerator Placement
(3)     Network Load Variation
(4)     Power Series Routing PSO Optimization
(5)     Number of Regenerators
(6)     Power Ratio Threshold Variation
(7)     Statistical Trend Analysis
(8)     Run Machine Learning in EON
(9)     Retrieve Combination of Algorithms
```

Fig. 2: New SimEON command line interface

IV. Experiment

In this paper, we consider two well-known network topologies depicted in Fig. 4 and Fig. 5. These refer to the 14-node NSF network topology and 11-node COST239 topology. For both SimEON and Optical-RL-Gym, we applied the same network settings and set the OSNR threshold to 14 dB. Each fiber link could accommodate 100 frequency slots (FS). For other network parameters, we use the default settings in DeepRMSA paper. In order to study the performance between different RMSA algorithms, we run extensive experiments with different number of call requests using the two topologies with different network loads (10,000 and 1,000,000 calls,



```
ailable on your machine and could speed up CPU computations.
2021-08-15 14:46:15.088708: W tensorflow/core/platform/cpu_feature_guard.cc:45]
The TensorFlow library wasn't compiled to use AVX2 instructions, but these are a
vailable on your machine and could speed up CPU computations.
Starting agent_0
Starting agent_1
Starting agent_2
/home/mohammad/Desktop/SimEON_Simulator/build-adamant-broccoli-Desktop-Debug/Dee
pRMSA_Agent.py:249: VisibleDeprecationWarning: Creating an ndarray from ragged n
ested sequences (which is a list-or-tuple of lists-or-tuples-or ndarrays with di
fferent lengths or shapes) is deprecated. If you meant to do this, you must spec
ify 'dtype=object' when creating the ndarray
  espisode_buff = np.array(espisode_buff)
Blocking Probability =  0.286
Mean Resource Utilization = 0.3677781818181818
Blocking Probability =  0.291
Mean Resource Utilization = 0.3681488636363637
Blocking Probability =  0.279
Mean Resource Utilization = 0.365525
Blocking Probability =  0.291
Mean Resource Utilization = 0.3662929545454546
Blocking Probability =  0.292
Mean Resource Utilization = 0.37187613636363637
```

Fig. 3: Running DeepRMSA algorithm in SimEON

respectively). Regenerator placement algorithms are also investigated to understand their impact in an optical network. The corresponding results are presented in the next Section.

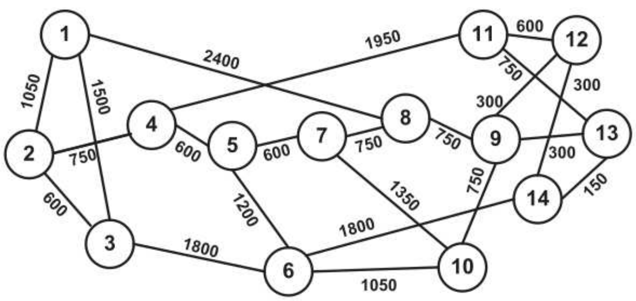


Fig. 4: 14-node NSFNET Topology (link length in kilometers)

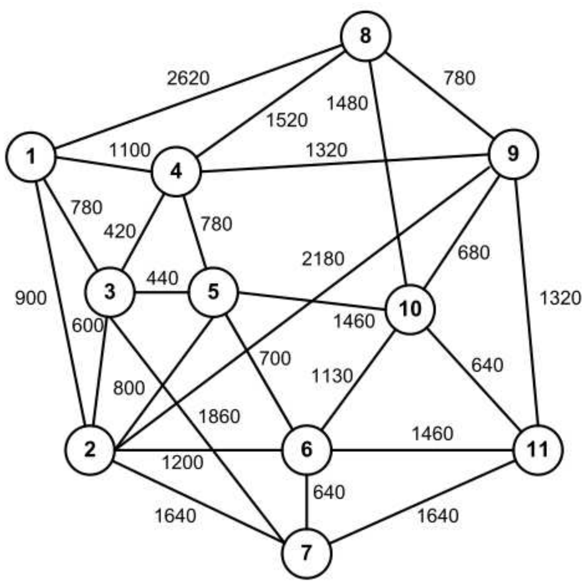


Fig. 5: 11-node COST239 Topology (link length in kilometers)

V. Results and Discussion

In this section, we discuss our findings when comparing the different RMSA algorithms in SimEON.

A. Comparison between conventional RMSA algorithms

We leverage our combination experiments feature to run different RMSA algorithm combinations simultaneously. We aim to find the overall best algorithm combination for our network topology. Fig. 6 and Fig. 7 show the blocking probability of different RMSA algorithms with 1,000,000 call requests, using COST239 topology and Dijkstra/BellmanFord algorithm fixed, respectively. We can see a clear performance distinction between different conventional algorithms. Furthermore, the FirstFit, ExactFit and BestFit algorithms seem to have the same level of performance in blocking probability. There is not much difference in using Dijkstra or Bellman-Ford for the routing algorithm for our experiment. Thus we consider these two have the same performance. As the network load increases, the blocking probability increases as well.

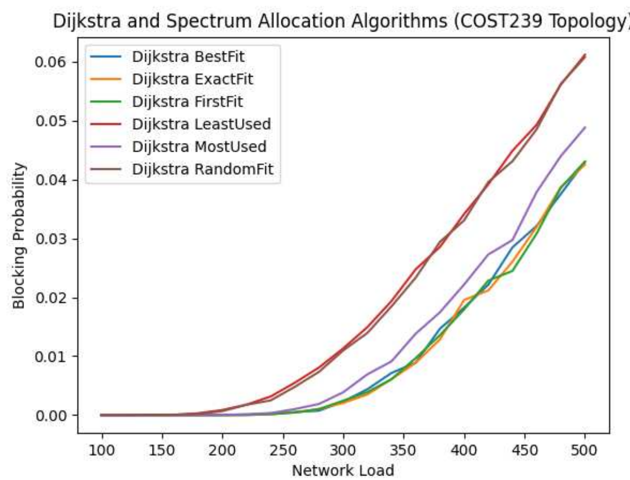


Fig. 6: Blocking probability of different RMSA algorithms (with Dijkstra algorithm fixed, COST239 topology, 1,000,000 calls)

B. Comparison between conventional RMSA and DeepRMSA algorithms

Fig. 8 shows the DeepRMSA agent training process. We use the same network parameter settings in the [16]. We can see that the blocking probability of the DeepRMSA algorithm is around 0.35 at the beginning and keeps dropping as the training process continues. After training with 500,000 requests, the DeepRMSA blocking probability is relatively stable and stays around 0.02. In our experiments that compare different conventional algorithms and DeepRMSA algorithms, we average the last 100 blocking probability results from DeepRMSA for different network loads and use that to plot our results. We run our experiments with the two topologies mentioned in Section IV.

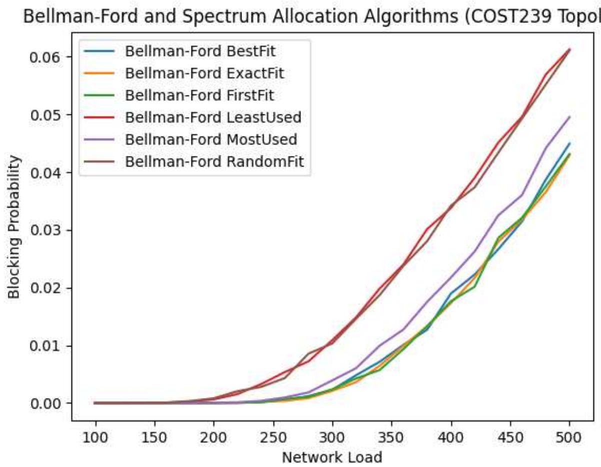


Fig. 7: Blocking probability of different RMSA algorithms (with BellmanFord algorithm fixed, COST239 topology, 1,000,000 calls)

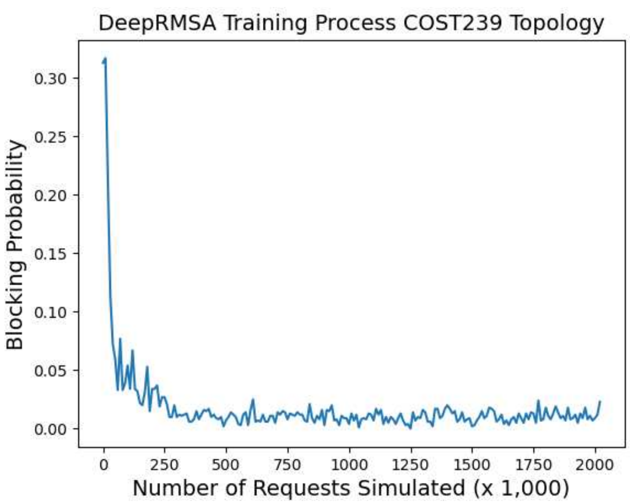


Fig. 8: Blocking probability of DeepRMSA algorithms (COST239 topology, first 2,00,000 calls in our experiment)

For COST239 topology, Fig. 9 and Fig. 10 show the blocking probability of different RMSA algorithms with 100,000 call requests and Dijkstra/BellmanFord algorithm fixed, respectively. We can see that the FirstFit and ExactFit algorithms work best for the conventional algorithms and the RandomFit and LeastUsed algorithms perform worst in terms of blocking probability. For all conventional algorithms, the blocking probability increases as the network load increases from 100 to 500 erlangs. On the other hand, the blocking probability of the DeepRMSA algorithm is around 0.035 regardless of the network load and it outperforms all the conventional algorithms when the network load is heavy.

For NSFNET topology, Fig. 11 and Fig. 12 show very similar trends as we get from COST239 topology. The FirstFit and ExactFit algorithms still perform best and the RandomFit and LeastUsed algorithms perform worst in terms of blocking

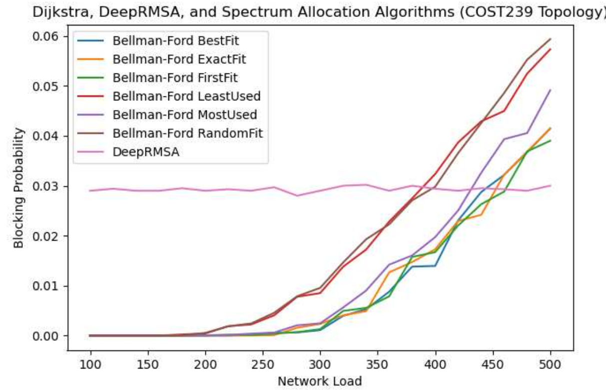


Fig. 9: Blocking probability of different RMSA algorithms (with Dijkstra algorithm fixed, COST239 topology, 100,000 calls)

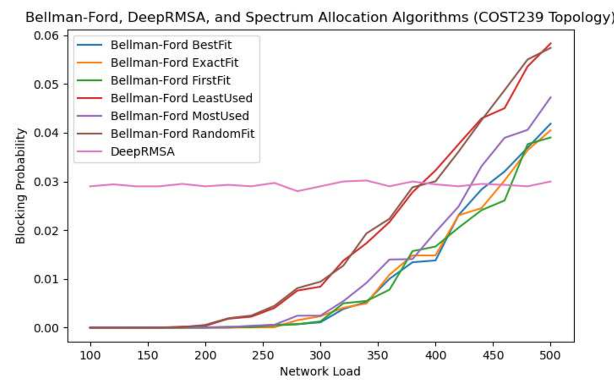


Fig. 10: Blocking probability of different RMSA algorithms (with BellmanFord algorithm fixed, COST239 topology, 100,000 calls)

probability. The blocking probability increases dramatically as the network load increases. The difference between these two topology results is that the overall blocking probability is much higher in the NSFNET topology. This is mainly because the NSFNET topology contains more nodes. The blocking probability of the DeepRMSA algorithm is around 0.05 regardless of the network load. It outperforms all the conventional algorithms when the network load is around 300 erlangs, which shows the superiority of the DeepRMSA algorithm for more complex topologies and heavy network loads.

VI. Conclusions

Our work enhances SimEON by integrating it with the DeepRMSA algorithm. We also provide a convenient way for users to test out different algorithms and get the best and worst algorithm combinations in terms of blocking probability. Such options allow users to evaluate different RMSA algorithms and create new reinforcement learning environments and algorithms for EON by leveraging the functionality pro-

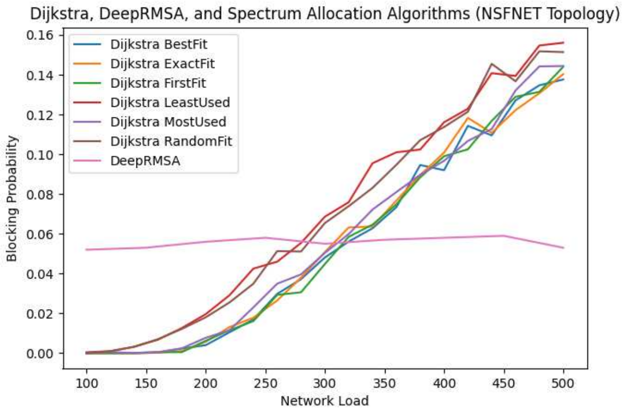


Fig. 11: Blocking probability of different RMSA algorithms (with Dijkstra algorithm fixed, NSFNET topology, 100,000 calls)

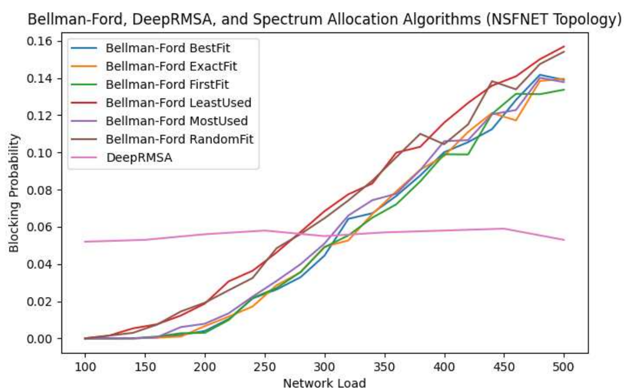


Fig. 12: Blocking probability of different RMSA algorithms (with BellmanFord algorithm fixed, NSFNET topology, 100,000 calls)

vided by the Optical-RL-Gym tool. Our experimental results show that for the conventional algorithms, the FirstFit and ExactFit algorithms perform best in blocking probability based on our network settings and the RandomFit and LeastUsed algorithms perform worst. The blocking probability increases dramatically as the network load increases. The DeepRMSA algorithms outperform the conventional algorithms when the network load is high, showing deep reinforcement learning's superiority in solving complex problems.

VII. Future Work

The current DeepRMSA simulation does not consider physical devices such as amplifiers and regenerators in the environment like SimEON does. Therefore, in our future work, we would like to create different environments for EON in the Optical-RL-Gym tool and propose our own DRL-enabled RMSA algorithm for further evaluation. We also want to use machine learning techniques to solve the regenerator placement problem in EON and compare the performance between the conventional algorithms and ML-enabled algorithms in the SimEON tool. Thus, users would have more freedom to choose from different RMSA algorithms and test their own topologies in different settings.

VIII. Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant Number CNS-1817105.

References

[1] A. Alyatama, I. Alrashed, and A. Alhusaini, "Adaptive routing and spectrum allocation in elastic optical networks," *Optical Switching and Networking*, vol. 24, pp. 12–20, 2017.

[2] M. Cavalcante, H. Pereira, D. Chaves, and R. Almeida, "Applying power series routing algorithm in transparent elastic optical networks," in *2015 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC)*. IEEE, 2015, pp. 1–5.

[3] M. A. Cavalcante, H. A. Pereira, and R. C. Almeida, "SimEON: an open-source elastic optical network simulator for academic and industrial purposes," *Photonic Network Communications*, vol. 34, no. 2, pp. 193–201, 2017.

[4] A. Asensio, A. Castro, L. Velasco, and J. Comellas, "An elastic networks OMNeT++-based simulator," in *2013 15th International Conference on Transparent Optical Networks (ICTON)*. IEEE, 2013, pp. 1–4.

[5] M. Aibin and M. Blazejewski, "Complex elastic optical network simulator (ceons)," in *2015 17th International Conference on Transparent Optical Networks (ICTON)*. IEEE, 2015, pp. 1–4.

[6] R. S. Tessinari, B. Puype, D. Colle, and A. S. Garcia, "ElasticO++: An elastic optical network simulation framework for OMNeT++," *Optical Switching and Networking*, vol. 22, pp. 95–104, 2016.

[7] P. Pavon-Marino and J.-L. Izquierdo-Zaragoza, "Net2plan: an open source network planning tool for bridging the gap between academia and industry," *IEEE Network*, vol. 29, no. 5, pp. 90–96, 2015.

[8] P. Pavon-Marino, M.-V. Bueno-Delgado, and J.-L. Izquierdo-Zaragoza, "Evaluating internal blocking in noncontentionless flex-grid ROADMs," *Journal of Optical Communications and Networking*, vol. 7, no. 3, pp. A474–A481, 2015.

[9] R. Casellas, R. Muñoz, J. M. Fàbrega, M. S. Moreolo, R. Martínez, L. Liu, T. Tsuritani, and I. Morita, "GMPLS/PCE control of flexi-grid DWDM optical networks using CO-OFDM transmission," *Journal of Optical Communications and Networking*, vol. 4, no. 11, pp. B1–B10, 2012.

[10] 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," *Journal of Optical Communications and Networking*, vol. 5, no. 8, pp. 836–847, 2013.

[11] 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 journal on selected areas in communications*, vol. 31, no. 1, pp. 26–38, 2012.

[12] Y. Wang, X. Cao, and Y. Pan, "A study of the routing and spectrum allocation in spectrum-sliced elastic optical path networks," in *2011 Proceedings Ieee Infocom*. IEEE, 2011, pp. 1503–1511.

[13] L. Gong, X. Zhou, W. Lu, and Z. Zhu, "A two-population based evolutionary approach for optimizing routing, modulation and spectrum assignments (RMSA) in O-OFDM networks," *IEEE Communications letters*, vol. 16, no. 9, pp. 1520–1523, 2012.

[14] J. R. Koza, "Genetic programming as a means for programming computers by natural selection," *Statistics and computing*, vol. 4, no. 2, pp. 87–112, 1994.

[15] F. Barbosa, A. de Sousa, A. Agra, K. Walkowiak, and R. Goścień, "RMSA algorithms resilient to multiple node failures in dynamic EONs," *Optical Switching and Networking*, vol. 42, p. 100633, 2021.

[16] X. Chen, B. Li, R. Proietti, H. Lu, Z. Zhu, and S. B. Yoo, "DeepRMSA: a deep reinforcement learning framework for routing, modulation and spectrum assignment in elastic optical networks," *Journal of Lightwave Technology*, vol. 37, no. 16, pp. 4155–4163, 2019.

[17] C. Natalino and P. Monti, "The Optical RL-Gym: An open-source toolkit for applying reinforcement learning in optical networks," in *2020 22nd International Conference on Transparent Optical Networks (ICTON)*. IEEE, 2020, pp. 1–5.