2023 International Conference on Optical Network Design and Modeling (ONDM)

Masked Deep Reinforcement Learning for Virtual Network Embedding on Elastic Optical Networks

Michael Doherty
Optical Networks Group
Electronic & Electrical Eng. Dept.
University College London
London, UK
michael.doherty.21@ucl.ac.uk

Yitao Zhang
Electronic & Electrical Eng. Dept.
University College London
London, UK
zceezhb@ucl.ac.uk

Alejandra Beghelli
Optical Networks Group
Electronic & Electrical Eng. Dept.
University College London
London, UK
alejandra.beghelli@ucl.ac.uk

***Abstract*—Deep reinforcement learning (DRL) with invalid action masking is applied to the optimization problem of virtual optical network embedding (VONE) over elastic optical networks (EON). Separate DRL agents are trained on the node-mapping task, link-mapping task, and overall VONE task. Their blocking probability performance is compared with a spectral fragmentation-aware VONE heuristic. All three DRL agents achieve lower blocking probability than the heuristic across low and high traffic loads.**

***Index Terms*—elastic optical network (EON), virtual network embedding (VNE), deep reinforcement learning (DRL)**

I. Introduction

A. Application Context

Network virtualisation is a key enabling technology for the continued expansion of cloud-based Internet services. Through the abstraction of network resources, e.g. optical bandwidth on links and compute at nodes, network virtualisation allows multiple heterogeneous virtual networks to be overlaid on the same substrate network.

The benefits of network virtualisation over the use of dedicated network hardware include reduced capital expenditure for network tenants, reduced network management complexity, improved scalability and architecture innovation [1]. These benefits derive from the flexibility of infrastructure as a service and the efficient use of network resources that virtualisation allows.

The process of allocating resources to virtual networks is known as virtual network embedding (VNE). Well-optimised VNE ensures that network tenants and operators get cost-effective use of the underlying resources.

In a network with fibre optic links, VNE is sometimes referred to as Virtual Optical Network Embedding (VONE) [2]. Depending on the optical substrate, the VONE problem must take into account different spectrum constraints. The division of available bandwidth into highly granular spectral frequency slot units (FSU) to form an elastic optical network (EON) greatly increases the potential network capacity [3]. Therefore, this work focuses on an elastic substrate network.

B. The VONE problem

A virtual network request comprises a set of requested node resources, bandwidth for interconnect, and virtual topology. To allocate the resources for a virtual network request, the VNE problem can be decomposed into the sub-problems of selecting substrate nodes for virtual nodes (node-mapping) and selecting one-or-more substrate links to comprise each virtual link (link-mapping). VNE over elastic optical networks has the additional requirement of FSU-selection for the virtual links, where the selected FSUs are subject to continuity and contiguity constraints. That is, the same FSUs must be used on each link and those FSUs must comprise an uninterrupted spectral 'block'.

The VONE problem can be further classified as static, where virtual network requests are known a priori, or dynamic. This work focuses on dynamic VONE, where requests must be served on-demand. The VNE problem is NP-hard [1] and thus, so is the VONE problem.

C. Previous work

As an NP-hard optimisation problem, VONE strategies have been attempted with integer and mixed integer linear programs [4] and heuristic algorithms [5]–[7].

An effective VONE heuristic is the Node-Switching-Capability k-Shortest-Path Fragmentation-Degree-Loss algorithm (NSC-kSP-FDL) [5]. This heuristic first ranks the substrate nodes by "node switching capability" (defined as the product of available node capacity, port count and sum of vacant FSUs on connected links) and maps virtual nodes to the highest ranked substrate nodes. Virtual links are then mapped by selecting from the available FSUs on the pre-computed shortest paths that least increase the metric of fragmentation degree loss. This combination of node-ranking and reduced spectral fragmentation allows NSC-kSP-FDL to make efficient use of network resources.

Most studies of the dynamic VONE problem have been restricted to heuristic algorithms, with a few recent developments in the use of deep reinforcement learning (DRL). The RDAM algorithm [8] utilises DRL to optimise node-mapping strategies and achieves greater acceptance ratio than two other algorithms but neglects to learn link-mapping,

Financial support from EPSRC Centre for Doctoral Training in Connected Electronic and Photonic Systems (CEPS CDT) and EPSRC Programme Grant TRANSNET (EP/R035342/1) is gratefully acknowledged.

978-3-903176-54-6 © 2023 IFIP

instead adopting breadth-first search. The multi-agent deep reinforced virtual network embedding algorithm (MADRVNE) [9] uses separate DRL agents for the node-mapping and link-mapping (including FSU-selection), with an interdependent reward structure to ensure their cooperation. It shows higher acceptance rate and resource utilization than RDAM and two heuristic algorithms, thereby demonstrating the effectiveness of DRL for this problem.

Masked reinforcement learning can be considered as a form of model-based reinforcement learning which incorporates domain knowledge about the problem to shape an agent's decision [10]. Invalid actions, e.g. selecting occupied nodes or FSUs, are masked out such that their probability of selection by the agent is zero. Training is consequently more efficient, as the agent does not need to learn which actions are invalid before learning which valid actions are most effective. This technique has been successfully applied to resource allocation problems in data centres [11], as well as routing-and-wavelength [12] and routing-and-spectrum [13] assignment in optical networks. The latter corresponds to the virtual link-mapping between nodes of the VONE problem here considered.

D. Motivation and Novel Contributions

Given the promising results of a multi-agent DRL approach for the dynamic VONE problem, this work aims to investigate if a simpler single agent DRL system can also outperform state-of-the-art heuristics. Therefore, this work presents the first study of DRL for the dynamic VONE problem that uses minimal feature engineering for the observation space compared to other approaches [9] and is the first to successfully use a single agent design, through the use of multi-step invalid action masking.

Three separate DRL models are evaluated, one each for the node-mapping and link-mapping sub-problems and the combined VONE problem, to further understand the influence of the problem structure on the effectiveness of DRL in finding optimised VONE policies. The agents are referred to as the Node Agent, Path Agent and Combined Agent, respectively.

II. Network and Traffic Model

The substrate network is an EON that is modelled as an undirected graph comprising N_s nodes and L_s bidirectional links. Each substrate node $n_s \in N_s$ is equipped with $C(n_s)$ capacity units. Each substrate link $l_s \in L_s$ has $B(l_s)$ FSUs. The substrate nodes may represent resources that are co-located or in geographically distinct datacentres. The physical interpretation of node capacity depends on the characteristics of the substrate network that is being modelled. Nodes may correspond to optical switches with capacity measured in number of ports/transceivers, or nodes may include computing resources, in which case capacity usually refers to available compute or memory [7]. In the rest of the paper, node capacity is referred to as compute units.

Virtual networks are also modelled as undirected graphs, consisting of N_v virtual nodes and L_v virtual links. Virtual nodes and links are assigned an integer identifier. The i-th virtual node n_v^i has a capacity requirement of $RC(n_v^i)$ compute units and the i-th virtual link l_v^i has a bandwidth requirement of $RB(l_v^i)$ FSUs.

Traffic is assumed to be dynamic. That is, virtual network requests arrive and depart following a stochastic process. Virtual network requests are triplets specifying: i) the list of virtual links as pairs of virtual nodes $[(n_v^i, n_v^j), ...]$, ii) the list of virtual node requirements ($[RC_1, RC_2, ..., RC_{N_v}]$) and iii) the list of bandwidth requirements for virtual links ($[RB_1, RB_2, ..., RB_{L_v}]$). The bandwidth requirements of virtual links can be expressed as the required number of FSUs directly, if the choice of modulation format based on link qualities and required data transmission rate has been pre-calculated.

III. Deep Reinforcement Learning Algorithm

Reinforcement learning describes a class of machine learning algorithms in which a policy is learned to perform actions in an environment to maximise some notion of 'reward'. The reward is always a scalar value that is returned by the environment in response to an action. The policy is a function that maps observations of the environment to a probability distribution of actions. In DRL, the policy function is approximated by a deep neural network.

DRL is distinct from other learning paradigms (e.g. supervised or unsupervised learning) in that the actions taken are evaluated and the policy is optimised based on the evaluation, rather than direct instruction of which actions are best [14]. This evaluative feedback during training is what allows DRL to find new policies that can outperform existing heuristics.

In this paper, the dynamic VONE problem is modelled as a sequence of discrete decision-making timesteps. At each timestep, a virtual network request is received and resources are allocated to service it. As the embedding decision made at each timestep is dependent only on the current state of the network, the entire VONE process can be considered as a Markov Decision Process (MDP). This MDP formulation allows DRL algorithms to be applied to learn optimised allocation strategies. As an MDP, the components of the DRL system are divided into the environment and the agent, as shown in Figure 1.

A. Environment

The environment (yellow box in Figure 1) corresponds to the substrate EON where virtual networks are established and released. It comprises the model of the substrate network, the generated virtual network requests, the reward function, and the machinery to interpret and effect the actions of the agent.

The interaction of the agent with the environment is mediated by the virtual network request and network state observation (arrow 1 in Figure 1), the agent's action (arrow 3) and reward (arrow 4).

The observation space contains the information on the state of the network's resources that is presented to the agent at each timestep, and includes the current virtual network

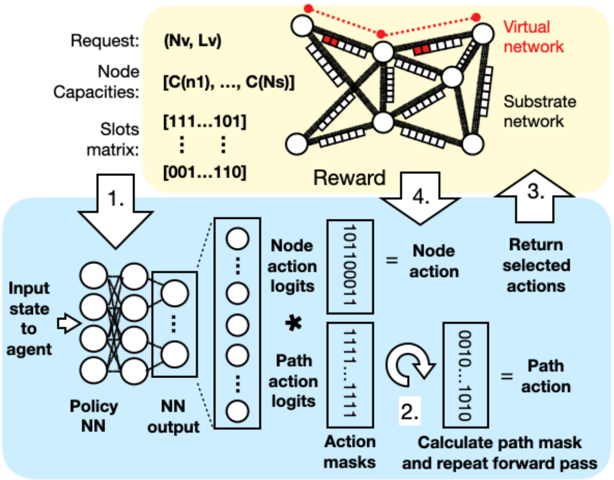

Fig. 1. Overview of the interaction between agent (blue) and environment (yellow) in the modelled VONE process, including the principle of multi-step invalid action masking for node-and-path selection by the agent.

request. In this work, the network state presented to the agent comprises a list of remaining node resource capacities and a two-dimensional binary array (slots matrix in Figure 1) representing the occupation of FSUs in each substrate link. Additionally, an agent that only performs the path-mapping task must receive the information on selected substrate nodes as part of the observation space.

The action space for the agent can be divided into node and path sections. The action space dimensionality must match that of the agent with which it is paired.

If selecting N_v virtual nodes per request from a possible N_s substrate nodes, and considering the request ordering to be immutable, the node action space dimensions are $(1 \times N_s^{N_v})$. These dimensions can be further reduced by considering the constraint that no virtual nodes in a request can share the same substrate node. A reduced action space is desirable to reduce the agent's training time taken to evaluate all actions.

The path action space in this work is simplified using the common technique of pre-calculating the k-shortest paths between all nodes. The action selection for each virtual link therefore becomes a choice between the FSUs on each of the k-paths. The selected slot serves as the initial slot to accommodate the requested bandwidth. For a request comprising L_v virtual links, the action space dimensions become $(L_v \times k * N_f)$, where N_f is the number of FSUs per link.

The reward returned by the environment is the signal on which the agent policy is optimised. Although reward functions that capture domain knowledge have been used in prior work to guide agent behaviour and improve performance [15], it is recommended in [14] that the reward function only provides information on whether the target outcome is achieved, without influencing how it is achieved. Consequently, the reward used in this work is not shaped to direct the learning experience of the agent by penalising or rewarding particular behaviour, other than successful servicing of a request, as shown by Equation 1.

$$R = \begin{cases} 10, & Success \\ 0, & Failure \end{cases} \qquad (1)$$

Success or failure is determined by the environment checking that the selected nodes have sufficient compute resources, the selected initial FSU's are the start of a sufficient vacant spectral block to accommodate the bandwidth request at every substrate link in the path, and that the selected slots for the different virtual links in the same request do not clash.

B. Agent

The DRL algorithm for the agent is based on Proximal Policy Optimization (PPO), a policy gradient method that compares favorably to other DRL algorithms for data efficiency and robustness to hyperparameter selection [16]. The algorithm is modified to apply invalid action masking prior to an action being sampled. The invalid action mask is a binary 1-D array that is calculated in the environment according to the current request and remaining node and link capacities. Each element of the array corresponds to an action and has value zero if the action attempts to utilise unavailable resources. The mask is then applied to the output probabilities of the policy network, which represent possible actions from the agent, such that only valid actions have a non-negligible probability of selection. The three agents investigated in this work (Node Agent, Path Agent, Combined Agent) all employ invalid action masking during training and evaluation.

The Node Agent and Path Agent perform only the node-mapping or link-mapping tasks of the VONE problem, respectively. That is, their policy networks only output probabilities for node and path selection, respectively, and the corresponding masks are applied in those cases. The remaining task is completed by the NSC-kSP-FDL heuristic [5] For example, once the Node Agent has selected the nodes, suitable paths to connect nodes are found by applying first-fit on the k-shortest paths between nodes. Analogously, once the physical nodes have been selected based on their ranking according to the heuristic, the Path Node agent selects a path to connect the selected nodes.

The Node Agent's output only comprises the node action, therefore the action is masked based on remaining node resource capacity. For the Path Agent, the nodes selected by the heuristic form part of the observation, such that the invalid action mask for the path action can be calculated on the basis of the selected nodes.

Invalid action masking for the Combined Agent requires a two-step approach, due to the sequential interpretation of its action as node-mapping followed by link-mapping. The process is illustrated in the blue region of Figure 1. The initial mask comprises the node mask, in which actions corresponding to nodes without sufficient capacity are masked to zero, and identity values for the path mask. The probability of each action logit is inferred by the policy network from the observation state and the node action is sampled. From the node action, the context for the path actions is established and the path mask is calculated to exclude occupied FSUs and FSUs that don't initiate a sufficient contiguous block. The combined node and path masks can then be applied to an additional forward pass of the policy network (arrow 2 in Figure 1).

As the action of the policy network is sampled stochastically according to the output probabilities of the action logits, the final node action must always be substituted with the original selection, to ensure the path mask context remains consistent.

IV. Results

A. Training

Each training episode starts with an unoccupied substrate network. A training episode comprises a set number of timesteps, during which the agent collects experiences and periodically updates the parameters of its policy network. Experiences are gathered in a rollout buffer of length n_{steps}, which is an important hyperparameter to tune to achieve peak agent performance. The batch size should be a factor of n_{steps}. Each training episode comprised 5000 timesteps and the agents were trained for 40 episodes, after which no improvement in acceptance ratio was observed.

Training was performed considering as a substrate network the NSFNet topology (14 nodes, 21 bidirectional links), equipped with 16 FSUs per link and 5 compute units per node. The virtual network topology is restricted to a 3-node ring topology, therefore requests are simplified to be a tuple of the requested node compute resources, RC_i, and FSUs, RB_i, for node interconnection: $([RC_1, RC_2, RC_3], [RB_1, RB_2, RB_3])$. The number of FSUs, node resources and virtual node topology-variants are restricted in order to facilitate the analysis of agent behaviour and performance on the tasks, while retaining the key challenges of judicious node-, link- and FSU-selection of the VONE problem.

Requested compute and bandwidth are randomly selected from $\{1,2\}$ units and $\{2,3,4\}$ FSUs, respectively. The mean service holding time is set to 10 time units and request arrival times are selected randomly from an exponential distribution of the ratio of traffic load to mean service holding time. The random number generator for traffic generation was reset at the beginning of each episode and seeded with the index of the training episode, to ensure non-identical traffic patterns across episodes.

The traffic load for training was set to 9 Erlangs, which was sufficient to require the agent to find strategies with good performance. The number of alternative routes was set to $k = 5$ and deemed to be a reasonable number for the size of substrate network to allow suitable diversity.

Hyperparameter optimisation was performed separately for each agent. To efficiently find hyperparameter values, sweeps were performed using a Bayesian optimisation [17] and Hyperband early stopping [18], with the Weights and Biases online monitoring platform [19].

Figure 2 shows the training results of the 3 agents, with the metric of mean acceptance ratio (accepted requests / total requests) per episode. The Node Agent rapidly converges on an effective policy, which is then refined over the course of training. The Path Agent follows a similar trend of rapid convergence but at a $\sim$20% lower overall acceptance ratio than the Node Agent. This disparity indicates the Path Agent is limited by the efficacy of the heuristic that determines node selection.

The Combined Agent shows lower acceptance ratio than the other agents in the first half of training. We hypothesise this is due to the much larger action space for the Combined Agent, which requires more timesteps to explore and to learn the interdependency of node and path selection. The Combined Agent converges to a policy with mean acceptance ratio between those of the other agents. The Combined Agent has access to the full range of actions in both node and path action spaces, so is capable of learning the same policy employed by the Node Agent, yet does not achieve the same level of performance. This suggests the Combined Agent converges to a local minimum, despite the invalid action masking, again due to its larger total action space.

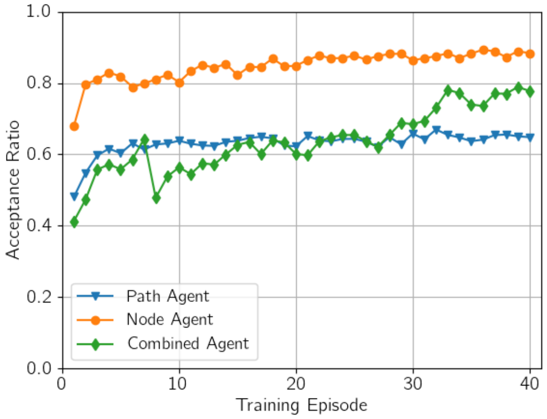

Fig. 2. Mean acceptance ratio during training under 9 Erlangs.

B. Evaluation

The agents were evaluated for 3 episodes at each traffic load from 1 to 15 Erlangs. The range of traffic loads selected produces blocking probabilities that are considered high for an operative network but are appropriate in this investigation due to the reduced number of FSUs in the substrate topology, and in order to test the performance of the different methods. For traffic loads of 1-3 Erlangs, sufficient evaluation episodes were run until 100 blocking events were observed before calculating the mean blocking probability (blocked requests / total requests). Figure 3 shows the evaluation results for each agent, compared with the results for the NSC-kSP-FDL heuristic VONE algorithm [5].

All three agents show lower blocking probability than the heuristic at low traffic loads despite only training at 9 Erlangs, which suggests good generalisation across traffic loads. At traffic loads of 4 Erlangs and above, the Path Agent performance converges with that of NSC-kSP-FDL.

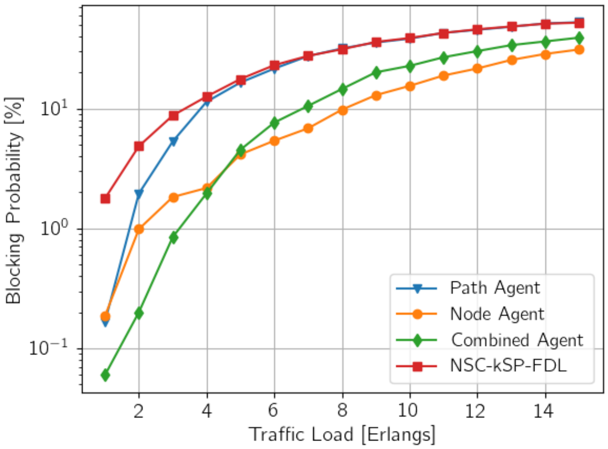

Fig. 3. Mean blocking ratio performance when evaluated across traffic loads.

This indicates that the Path Agent learned a link-mapping policy as successful as FDL that could also adapt to low traffic, but was limited by the node selection.

The Node Agent does not perform as well as the Combined Agent at low traffic loads, due to the rigid link-mapping heuristic's inability to adapt to a less-populated substrate. Above 4 Erlangs, the Node Agent gives the lowest blocking probability of all agents and the heuristic. This underscores the importance of node-mapping in dynamic VONE, which can outweigh link-mapping in determining overall performance.

The Combined Agent shows the best performance at low traffic loads but does not match the Node Agent's performance at higher loads. We hypothesise that the Combined Agent policy network converged to a local minimum due to insufficient exploration of its larger action space. The training efficiency of the Combined Agent may also have been impeded by requests that fail to be serviced due to virtual links that use the same substrate link and FSUs, when mapping links between different node pairs in the same request. This failure mode, referred to as slot clashing, is not masked and likely contributes to reduced performance compared to the Node Agent, which performs an exhaustive search in the link-mapping heuristic to avoid such clashes.

The greater freedom of action of the Combined Agent allows it more theoretical scope for optimization than the other agents, therefore strategies to optimise it further may be beneficial. Reward-shaping, such as awarding higher values for requests fulfilled when network utilisation is higher, could allow the agent to achieve better performance at higher traffic loads rather than lower. Additionally, the multi-step invalid action masking could be extended to re-mask the path actions after each virtual link is mapped. This would eliminate a common failure mode of slots clashing in fulfilling the same virtual request, thereby ensuring reduced blocking probability and/or training time, but at the expense of increased computational cost due to multiple forward passes of the policy network.

V. Conclusions

Three separate DRL agents: Node Agent, Path Agent and Combined Agent, were trained to find optimised policies for node-mapping, link-mapping and the complete VONE problem, respectively. All three outperform a heuristic algorithm, through the use of invalid action masking to improve training efficiency. The Node Agent achieved the lowest blocking probability, thus demonstrating the importance of node selection in VONE and the limited efficacy of the heuristic algorithm in this task.

The Combined Agent, which performs node-mapping and link-mapping simultaneously, demonstrates a novel application of multi-step invalid action masking and is the first single-agent architecture to outperform a heuristic in this problem. Although not as effective as the Node Agent in this work, it is expected that further optimisation of the training regime and additional action masking steps could allow the single-agent approach to achieve state-of-the-art blocking probability.

Future work on explainability of the best-performing agent would enable insight to its strategy, to inspire improved heuristics and highlight research directions for further policy optimisation.

References

[1] N. M. M. K. Chowdhury and R. Boutaba, “A survey of network virtualization,” *Computer Networks*, vol. 54, no. 5, pp. 862–876, Apr. 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128609003387

[2] L. Gong and Z. Zhu, “Virtual Optical Network Embedding (VONE) Over Elastic Optical Networks,” *Journal of Lightwave Technology*, vol. 32, no. 3, pp. 450–460, Feb. 2014. [Online]. Available: http://ieeexplore.ieee.org/document/6679238/

[3] O. Gerstel, M. Jinno, A. Lord, and S. B. Yoo, “Elastic optical networking: a new dawn for the optical layer?” *IEEE Communications Magazine*, vol. 50, no. 2, pp. s12–s20, Feb. 2012.

[4] K. D. R. Assis, S. Peng, R. C. Almeida, H. Waldman, A. Hammad, A. F. Santos, and D. Simeonidou, “Network virtualization over elastic optical networks with different protection schemes,” *Journal of Optical Communications and Networking*, vol. 8, no. 4, pp. 272–281, Apr. 2016.

[5] H. Wang, X. Xin, J. Zhang, Y. Sun, and Y. Ji, “Dynamic virtual optical network mapping based on switching capability and spectrum fragmentation in elastic optical networks,” in *2016 21st OptoElectronics and Communications Conference (OECC) held jointly with 2016 International Conference on Photonics in Switching (PS)*, Jul. 2016, pp. 1–3.

[6] M. Zhu, S. Zhang, Q. Sun, G. Li, B. Chen, and J. Gu, “Fragmentation-Aware VONE in Elastic Optical Networks,” *Journal of Optical Communications and Networking*, vol. 10, no. 9, p. 809, Sep. 2018. [Online]. Available: https://opg.optica.org/abstract.cfm?URI=jocn-10-9-809

[7] D. Bórquez-Paredes, A. Beghelli, A. Leiva, N. Jara, A. Lozada, P. Morales, G. Saavedra, and R. Olivares, “Agent-based distributed protocol for resource discovery and allocation of virtual networks over elastic optical networks,” *Journal of Optical Communications and Networking*, vol. 14, no. 8, p. 667, Aug. 2022. [Online]. Available: https://opg.optica.org/abstract.cfm?URI=jocn-14-8-667

[8] H. Yao, B. Zhang, P. Zhang, S. Wu, C. Jiang, and S. Guo, “RDAM: A Reinforcement Learning Based Dynamic Attribute Matrix Representation for Virtual Network Embedding,” *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 2, pp. 901–914, Apr. 2021. [Online]. Available: https://ieeexplore.ieee.org/document/8469054/

[9] G. Li, C. Xi, and R. Zhu, “Multi-Agent Deep Reinforced Virtual Network Embedding in Elastic Optical Networks,” in *2022 20th International Conference on Optical Communications and Networks (ICOCN)*. Shenzhen, China: IEEE, Aug. 2022, pp. 1–3. [Online]. Available: https://ieeexplore.ieee.org/document/9900943/

[10] S. Huang and S. Ontañón, "A Closer Look at Invalid Action Masking in Policy Gradient Algorithms," 2020. [Online]. Available: https://arxiv.org/abs/2006.14171

[11] Z. Shabka and G. Zervas, "Resource Allocation in Disaggregated Data Centre Systems with Reinforcement Learning," Nov. 2021, arXiv:2106.02412 [cs]. [Online]. Available: http://arxiv.org/abs/2106.02412

[12] J. W. Nevin, S. Nallaperuma, N. A. Shevchenko, Z. Shabka, G. Zervas, and S. J. Savory, "Techniques for applying reinforcement learning to routing and wavelength assignment problems in optical fiber communication networks," *Journal of Optical Communications and Networking*, vol. 14, no. 9, pp. 733–748, Sep. 2022.

[13] M. Shimoda and T. Tanaka, "Mask RSA: End-To-End Reinforcement Learning-based Routing and Spectrum Assignment in Elastic Optical Networks," in *2021 European Conference on Optical Communication (ECOC)*. Bordeaux, France: IEEE, Sep. 2021, pp. 1–4. [Online]. Available: https://ieeexplore.ieee.org/document/9606169/

[14] R. S. Sutton and A. G. Barto, *Reinforcement learning: an introduction*, ser. Adaptive computation and machine learning. Cambridge, Mass: MIT Press, 1998.

[15] B. Tang, Y.-C. Huang, Y. Xue, and W. Zhou, "Heuristic reward design for deep reinforcement learning-based routing, modulation and spectrum assignment of elastic optical networks," *IEEE Communications Letters*, vol. 26, no. 11, pp. 2675–2679, 2022.

[16] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," Aug. 2017, arXiv:1707.06347 [cs]. [Online]. Available: http://arxiv.org/abs/1707.06347

[17] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian Optimization of Machine Learning Algorithms," in *Advances in Neural Information Processing Systems*, vol. 25. Curran Associates, Inc., 2012. [Online]. Available: https://papers.nips.cc/paper/2012/hash/05311655a15b75fab86956663e1819cd-Abstract.html

[18] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: a novel bandit-based approach to hyperparameter optimization," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 6765–6816, Jan. 2017.

[19] L. Biewald, "Experiment tracking with weights and biases," 2020, software available from wandb.com. [Online]. Available: https://www.wandb.com/