

# DeepDefrag: Spatio-Temporal Defragmentation of Time-Varying Virtual Networks in Computing Power Network based on Model-Assisted Reinforcement Learning

Huangxu Ma<sup>(1)</sup>, Jiawei Zhang<sup>(1)\*</sup>, Zhiquan Gu<sup>(1)</sup>, Hao Yu<sup>(2)</sup>, Tarik Taleb<sup>(2)</sup>, and Yuefeng Ji<sup>(1)\*</sup>

<sup>(1)</sup> State Key Lab of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications (BUPT), Beijing, China, *Corresponding author*: \*[zjw, yfj@bupt.edu.cn](mailto:zjw, yfj@bupt.edu.cn).

<sup>(2)</sup> Center for Wireless Communications, University of Oulu, Oulu, Finland.

**Abstract** We propose DeepDefrag, a model-assisted reinforcement learning for spatio-temporal defragmentation of time-varying virtual networks in a cross-layer optical network testbed, which realizes the efficient utilization of computing nodes and lightpaths by co-optimizing scheduling and embedding with fragment matching, reduces >13.5% cost of computing power network. ©2022 The Author(s)

## Introduction

In the metaverse era, computing power as the core of service forms will be ubiquitous in the network. The computing power network (CPN) integrates the ubiquitous computing resources through an IP/optical cross-layer network supporting wavelength bypass and traffic grooming, provides ultra-low latency, flexible adjustment, and green energy savings computing power services<sup>[1]</sup>. The services usually encode multiple stages into virtual networks composed of virtual nodes and virtual links. Embedding virtual networks into optical substrates (VONE)<sup>[2]</sup> has become a major problem in CPN. Traditional VONE researches mostly assume that virtual networks keep fixed allocated resources during the lifetime<sup>[2-5]</sup>. In practice, however, virtual networks' computing and bandwidth resources requirements exhibit time-varying characteristics due to the dynamic trend of the actual rate at which terminals collect data to be processed<sup>[6]</sup>. The embedding of time-varying virtual network requests (TVNRs) causes serious resource fragmentation, which will bring challenges to the efficient utilization of resources in CPN.

The resource fragmentation of CPN consists of temporal fragments and spatial fragments, which refer to some remaining computing or bandwidth resources that are isolated and cannot meet the demands of TVNRs due to the discontinuity in the spatio-temporal dimension. Recent researches related to VONE of TVNRs, however, have been limited to resource redistribution<sup>[6,7]</sup> or traffic-aware migration<sup>[8,9]</sup>, failing to consider the spatio-temporal resource fragments (ST-RF), which account for a large proportion of resource waste. To this end, we innovatively optimize the cost of VONE (i.e., the energy consumption of activated computing nodes and lightpaths) by promoting the efficient utilization of ST-RF.

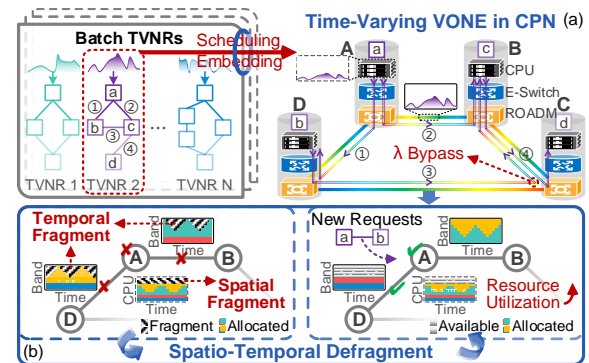
In this paper, we first propose the metric of ST-RF. Then, we analyze the principal causes of ST-RF lie in: for one thing, the embedding of

mismatched nodes and lightpaths; for another, the inappropriate scheduling sequence of TVNRs. An auxiliary graph (AG)<sup>[10]</sup> model-assisted edge-featured graph attention network<sup>[11]</sup>-enabled reinforcement learning (EGAT-RL), DeepDefrag, is proposed for spatio-temporal defragmentation in CPN. Its core highlight is co-optimizing batch TVNRs scheduling and embedding with fragment matching. We demonstrate DeepDefrag in a cross-layer optical network testbed, which reduces the number of activated computing nodes and lightpaths for >13.5% cost reduction.

## Spatio-Temporal Fragmentation of CPN

Fig. 1(a) shows the CPN composed of: the computing layer where CPU supplies computing resources, IP layer where E-Switch aggregates traffic, and the optical layer where ROADMs provides wavelength bypass. We model CPN as a dynamic directed graph  $G_s$  with CPU nodes  $N_s$ . Its dynamically connected edges  $E_s$  represent lightpaths established during cross-layer routing.

As shown in Fig. 1(b), the remaining bandwidth of edge AB/AD is discontinuous during the period. These temporal fragments cannot further carry virtual links, thereby causing the remaining computing resources of node A to be isolated in the spatial dimension, forming spatial fragments. In CPN, increased ST-RF will lead to lower resource utilization, and hence result in



**Fig. 1:** (a) Time-Varying VONE in CPN; (b) Spatio-Temporal Defragmentation.

more activated CPU and lightpaths, bring larger costs. To promote the efficient utilization of ST-RF, we define the metric  $\psi$  of fragment resources for each node or edge in the temporal dimension as Eq. (1)-(3), where  $\mu$  and  $\sigma$  denote the mean and standard deviation of the time-varying remaining computing or bandwidth resources  $L$  during the period  $T$ , and  $C$  is the resource capacity.  $\gamma$  is a weighting factor, indicating that with the decrease of the minimum value of  $L$ , the impact of  $\sigma$  on  $\psi$  increases.  $\phi$  represents the temporal fragment availability of each node or edge.

$$\psi = \gamma \cdot \sigma(L), L = \{l_t\}_{t \in T} \quad (1)$$

$$\gamma \propto 1 - \text{Min}(L)/C \quad (2)$$

$$\phi = \mu(L) - \psi \quad (3)$$

Further, we extend the centrality in graph theory to dynamically calculate the availability  $\theta$  of ST-RF of each node by combining the state of adjacent nodes and edges as Eq. (4), where  $\theta = \{\theta_n\}_{n \in N_S}$ .  $W$  is the weighted adjacency matrix of  $G_S$  whose weight is defined as the sum of  $\phi$  of edges between every two nodes.  $R$  is a row vector representing  $\phi$  of each node.

$$\theta = R \times (W + W^T) \quad (4)$$

### DeepDefrag Algorithm

The overall structure of DeepDefrag is shown in Fig. 2, where the core components lie in:

#### • Interface Design

(1) **State**: The optical substrate state includes the time-varying remaining computing and bandwidth resources. For each TVNR, its state includes the time-varying requirements and the 0/1 flag of whether the virtual node or virtual link is about to be embedded by breadth-first search. When one TVNR has been completed, the flags of unscheduled TVNRs' first virtual nodes are set to 1. TVNRs are divided into batches in order of arrival, and each batch is concatenated into a disconnected graph—the batch TVNRs state. The action mask state is used to dynamically mask inappropriate actions: 1) When continuing the embedding of the current TVNR, only the nodes with sufficient resources for the next step in the current action area are allowed, and the rest are masked. 2) When selecting the next TVNR, only the nodes that can host the first virtual node of each unscheduled TVNR are allowed. (2) **Action**: The action space is a flattened 2-D matrix, in

which each row represents the action area of each TVNR, and the columns are  $N_S$ . (3) **Reward**: When a virtual node is embedded along with several virtual links, the reward  $r$  set as Eq. (5) is fed back. The additional cost of VONE is the weighted sum of the number of newly activated CPU  $\omega^n$  and lightpaths  $\omega^e$ , where  $\alpha$ ,  $\beta$  are their respective energy consumption (i.e., 2500 W and 1500 W<sup>[12]</sup>). The feedback from the temporal fragment metric  $\psi$  of the selected CPU  $n_i$  and the selected lightpaths  $E_i$  will promote the matching of fragment resources. The optimization of  $\theta$  will promote the integration of ST-RF and then reduce the additional cost of future TVNRs.

$$r = -(\alpha\omega^n + \beta\omega^e + \psi_{n_i}\theta_{n_i} + \text{Max}(\{\psi_e\}_{e \in E_i})) \quad (5)$$

#### • EGAT-Enabled Agent

For the efficient graph state extraction of the optical substrate and the batch TVNRs with the features of nodes and edges as time-series, we propose to introduce EGAT into the RL agent for better optimization. EGAT especially strengthens the contribution of edge features to the attention scores of adjacent nodes<sup>[11]</sup> as shown in Fig. 2(a). Specifically, EGAT modifies the attention mechanism of the widely used GAT<sup>[13-15]</sup> by concatenating the edge message  $F_{ji}$  to the node message  $H_i$  and its adjacent node message  $H_j$ . The attention score  $\varepsilon_{ji}$  of node  $j$  to node  $i$  is then calculated as Eq. (6), and the messages of nodes and edges are updated as Eq. (7)(8), where  $\mathcal{N}_i$  is neighborhood of node  $i$ ,  $A$  is a trainable matrix.

$$\varepsilon_{ji} = \text{Softmax}(F_{ji}) \quad (6)$$

$$F'_{ji} = \text{LeakyReLU}(A[H_i || F_{ji} || H_j]) \quad (7)$$

$$H'_i = \sigma(\sum_{j \in \mathcal{N}_i} \varepsilon_{ji} H_j) \quad (8)$$

Respectively, after multiple rounds of message updates by EGAT, the nodes' messages of the optical substrate and the batch TVNRs are concatenated after averaging and fed into hidden layers. The action with the highest output probability after masking is chosen.

#### • AG-Assisted Environment

After the node's embedding by the RL agent, AG, a model that can route in the cross-layer optical network<sup>[10]</sup>, is introduced into the RL environment for solving two sub-problems: 1) when and how to establish new lightpaths; 2) how to route within a combination of existing and new lightpaths. An AG example is shown in Fig. 2(b).

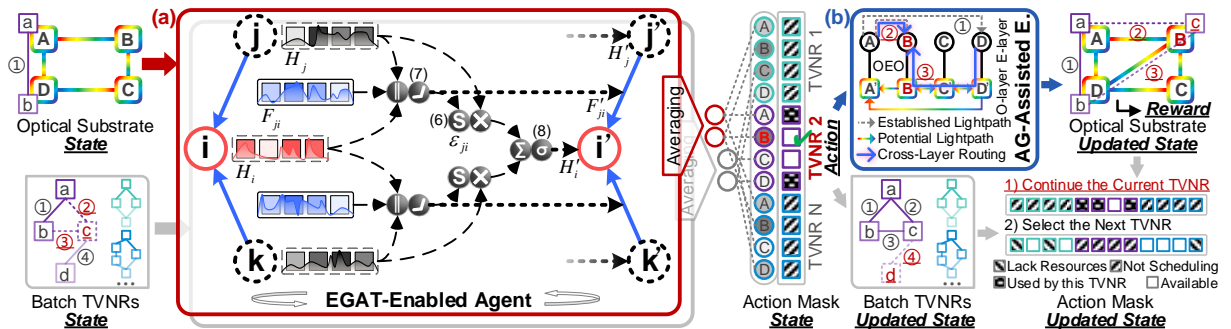


Fig. 2: The overall structure of DeepDefrag. (a) EGAT-Enabled Agent; (b) AG-Assisted Environment.

We trained the RL agent of DeepDefrag with Proximal Policy Optimization (PPO)<sup>[16]</sup>. The agent includes two sets of EGAT models for the optical substrate and the batch TVNRs, which both adopt 4 EGAT layers with 3 attention heads and 32 hidden features. The following are 3 hidden layers with (128, 128, 90) hidden neurons respectively. The parameters of AG are set to minimize the number of established lightpaths<sup>[2]</sup>.

### Experimental Setup and Results

To verify the performance and feasibility of DeepDefrag, we built a cross-layer optical network testbed reported in our previous work<sup>[17]</sup>, including nine hybrid optical-electrical switching nodes as shown in Fig. 3(a)(b). ROADM is bidirectionally interconnected with three 10 Gbps wavelengths. Correspondingly, six 10 Gbps optical transceivers are deployed between each ROADM and the line ports of E-Switch. The client ports of E-Switch are connected to the traffic generator and analyzer, which is responsible for generating variable traffic of virtual links after being processed by the simulated 10 TFLOPS CPU nodes. Topologies of TVNRs are randomly generated directed acyclic graphs with 2-5 virtual nodes. The virtual nodes and virtual links' resource requirements in 4-hour of each TVNR are generated by multiplying the same time-varying characteristic by random factors. The batch size is set to 10.

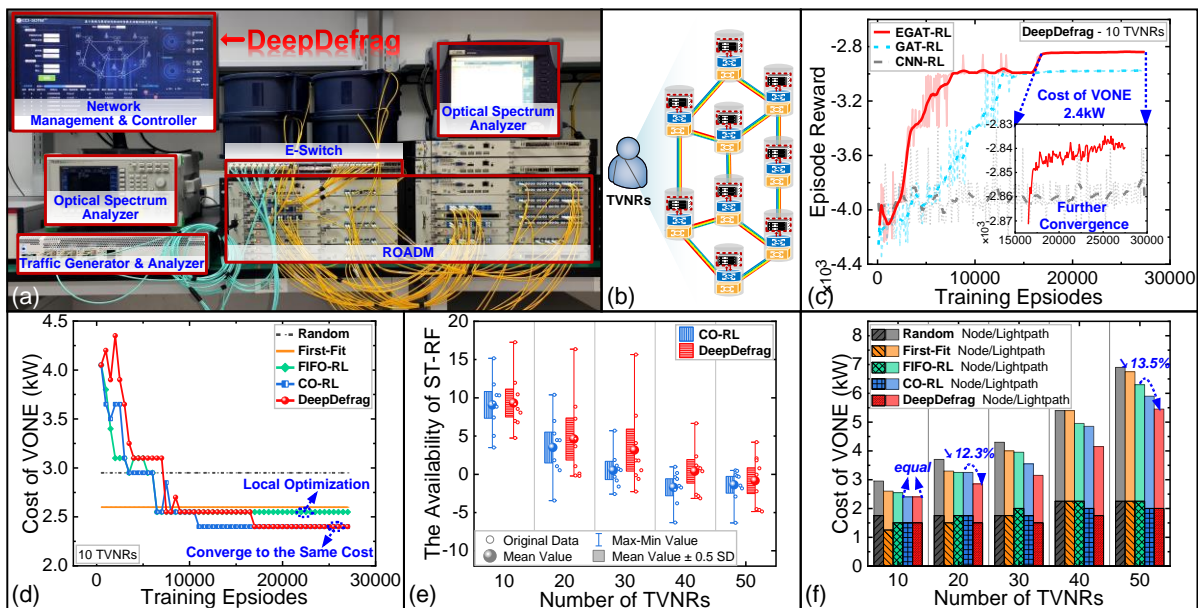
Fig. 3(c) shows the convergence of DeepDefrag with agents enabled by EGAT, GAT, and traditional convolutional neural network (CNN). The CNN agent can hardly converge as it's hard to extract the graph state from the dynamical adjacency matrix of the optical substrate. Compared with GAT, the EGAT agent achieves better convergence due to its enhanced capability to perceive edge features. To verify the

core highlight of DeepDefrag, we compared DeepDefrag with Random, First-Fit, FIFO-RL (TVNRs are sequentially embedded by EGAT-RL), and CO-RL (co-optimizing scheduling and embedding batch TVNRs by EGAT-RL without fragments feedback). With the convergence of the above algorithms, the costs of VONE for 10 TVNRs are shown in Fig. 3(d). FIFO-RL can only achieve local optimization because the scheduling sequence of TVNRs cannot be adjusted. Both CO-RL and DeepDefrag can converge to the optimal cost, but the further convergence of DeepDefrag shown in the subgraph of Fig. 3(c) can achieve better defragmentation with longer training, which is illustrated in Fig. 3(e) that DeepDefrag has better ST-RF availability  $\theta$  than CO-RL. As can be seen in Fig. 3(f), 12.3% cost reduction compared with CO-RL for 20 TVNRs is achieved by DeepDefrag, the key reason for which is the better availability  $\theta$  of ST-RF brought by defragmentation. Finally, DeepDefrag reduces the cost by 13.5% and 7.6% respectively compared with FIFO-RL and CO-RL for 50 TVNRs.

### Conclusions

We proposed DeepDefrag for VONE of TVNRs. The experimental results validated that DeepDefrag has better performance in reducing the cost of activated computing nodes and lightpaths by co-optimizing scheduling and embedding with fragment matching. 13.5% cost reduction can be achieved by DeepDefrag compared with FIFO-RL for 50 TVNRs.

**Acknowledgements:** This work is supported by the National Nature Science Foundation of China Projects (61871051, 61971055), the BUPT Innovation and Entrepreneurship Support Programs (2022-YC-T006, 2022-YC-A004).



**Fig. 3:** (a) DeepDefrag testbed; (b) Topology; (c) Convergence process of DeepDefrag with EGAT/GAT/CNN agent; (d) Cost of VONE vs. training episodes; (e) The availability of ST-RF vs. the number of TVNRs; (f) Cost of VONE vs. the number of TVNRs.



## References

- [1] B. Lei and G. Zhou, "Exploration and practice of Computing Power Network(CPN) to realize convergence of computing and network," in *2022 Optical Fiber Communications Conference and Exhibition (OFC)*, pp. 1-3, 2022, DOI: <https://doi.org/10.1364/OFC.2022.M4A.2>.
- [2] J. Zhang, Y. Ji, M. Song, H. Li, R. Gu, Y. Zhao, and J. Zhang, "Dynamic virtual network embedding over multilayer optical networks," *Journal of Optical Communications and Networking*, vol. 7, no. 9, pp. 918-927, 2015, DOI: <https://doi.org/10.1364/JOCN.7.000918>.
- [3] M. Lu, Y. Gu, and D. Xie, "A Dynamic and Collaborative Multi-Layer Virtual Network Embedding Algorithm in SDN Based on Reinforcement Learning," *IEEE Transactions on Network and Service Management*, vol. 17, no. 4, pp. 2305-2317, 2020, DOI: <https://doi.org/10.1109/TNSM.2020.3012588>.
- [4] Z. Yan, J. Ge, Y. Wu, L. Li, and T. Li, "Automatic Virtual Network Embedding: A Deep Reinforcement Learning Approach With Graph Convolutional Networks," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 6, pp. 1040-1057, 2020, DOI: <https://doi.org/10.1109/JSAC.2020.2986662>.
- [5] H. Yu, T. Taleb, J. Zhang, and H. Wang, "Deterministic Latency Bounded Network Slice Deployment in IP-Over-WDM Based Metro-Aggregation Networks," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 2, pp. 596-607, 2022, DOI: <https://doi.org/10.1109/TNSE.2021.3127718>.
- [6] H. Liu, M. Wen, Y. Chen, C. Tang, J. Hu, and H. Chen, "Virtual optical network embedding of time-varying traffic in elastic optical networks," *Optics Communications*, vol. 508, p. 127693, 2022, DOI: <https://doi.org/10.1016/j.optcom.2021.127693>.
- [7] S. Ding, S. K. Bose, and G. Shen, "Spectrum trading between virtual optical networks with time-varying traffic in an elastic optical network," *Journal of Optical Communications and Networking*, vol. 12, no. 3, pp. 24-37, 2020, DOI: <https://doi.org/10.1364/JOCN.377462>.
- [8] H. Yu, F. Musumeci, J. Zhang, M. Tornatore, L. Bai, and Y. Ji, "Dynamic 5G RAN slice adjustment and migration based on traffic prediction in WDM metro-aggregation networks," *Journal of Optical Communications and Networking*, vol. 12, no. 12, pp. 403-413, 2020, DOI: <https://doi.org/10.1364/JOCN.403829>.
- [9] H. Yu, T. Taleb, and J. Zhang, "Deterministic Latency/Jitter-aware Service Function Chaining over Beyond 5G Edge Fabric," *IEEE Transactions on Network and Service Management*, Early Access, 2022, DOI: <https://doi.org/10.1109/TNSM.2022.3151431>.
- [10] H. Zhu, H. Zang, K. Zhu, and B. Mukherjee, "A novel generic graph model for traffic grooming in heterogeneous WDM mesh networks," *IEEE/ACM Transactions On Networking*, vol. 11, no. 2, pp. 285-299, 2003, DOI: <https://doi.org/10.1109/TNET.2003.810310>.
- [11] Z. Wang, J. Chen, and H. Chen, "EGAT: Edge-Featured Graph Attention Network," in *30th International Conference on Artificial Neural Networks (ICANN)*, pp. 253-264, 2021, DOI: [https://doi.org/10.1007/978-3-030-86362-3\\_21](https://doi.org/10.1007/978-3-030-86362-3_21).
- [12] Y. Xiao, J. Zhang, and Y. Ji, "Energy-efficient DU-CU deployment and lighthouse provisioning for service-oriented 5G metro access/aggregation networks," *Journal of Lightwave Technology*, vol. 39, no. 17, pp. 5347-5361, 2021, DOI: <https://doi.org/10.1109/JLT.2021.3069897>.
- [13] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *2018 International Conference on Learning Representations (ICLR)*, pp. 1-12, 2018, DOI: <https://doi.org/10.48550/arXiv.1710.10903>.
- [14] Y. Shao, R. Li, Z. Zhao, and H. Zhang, "Graph Attention Network-based DRL for Network Slicing Management in Dense Cellular Networks," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1-6, 2021, DOI: <https://doi.org/10.1109/WCNC49053.2021.9417321>.
- [15] H. Ma, J. Zhang, and Y. Ji, "Graph Sequence Attention Network-Enabled Reinforcement Learning for Time-Aware Robust Routing in OSU-Based OTN," in *2022 Optical Fiber Communication Conference and Exhibition (OFC)*, pp. 1-3, 2022, DOI: <https://doi.org/10.1364/OFC.2022.Th2A.18>.
- [16] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017, DOI: <https://doi.org/10.48550/arXiv.1707.06347>.
- [17] Z. Chen, J. Zhang, B. Zhang, R. Wang, H. Ma, and Y. Ji, "ADMIRE: Demonstration of Collaborative Data-Driven and Model-Driven Intelligent Routing Engine for IP/Optical Cross-Layer Optimization in X-Haul Networks," in *2022 Optical Fiber Communications Conference and Exhibition (OFC)*, pp. 1-3, 2022, DOI: <https://doi.org/10.1364/OFC.2022.M3F.4>.