This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.3030957, IEEE Open Journal of the Computer Society

Self-optimizing optical network with cloud-edge collaboration: architecture and application [Invited]

Zhuotong Li, Yongli Zhao, Yajie Li, Mingzhe Liu, Zebin Zeng, Xiangjun Xin, Feng Wang, Xinghua Li, and Jie Zhang

Abstract As an important bearer network of the fifth generation (5G) mobile communication technology, the optical transport network (OTN) needs to have high-quality network performance and management capabilities. Proof by facts, the combination of artificial intelligence (AI) technology and software-defined networking (SDN) can improve significant optimization effects and management for optical transport networks. However, how to properly deploy AI in optical networks is still an open issue. The training process of AI models depends on a large amount of computing resources and training data, which undoubtedly increases the carrying burden and operating costs of the centralized network controller. With the continuous upgrading of functions and performance, small AI-based chips can be used in optical networks as on-board AI. The emergence of edge computing technology can effectively relieve the computation load of network controllers and provide high-quality AI-based networks optimization functions. In this paper, we describe an architecture called self-optimizing optical network (SOON) with cloud-edge collaboration, which introduces control-layer AI and on-board AI to achieve intelligent network management. In addition, this paper introduces several cloud-edge collaborative strategies and reviews some AI-based network optimization applications to improve the overall network performance.

***Index Terms*—OTN, SDN, control-layer AI, on-board AI, cloud-edge collaboration.**

I. Introduction

With the continuous popularization and promotion of 5G technology, the emerging services in the network puts forward new requirements on the underlying transport network, such as low latency and large bandwidth transmission [1]. OTN combines the advantages of both optical domain transmission and electrical domain processing. It provides not only end-to-end rigid transparent pipe connection and strong networking capabilities, but also long-distance and high-capacity transmission [2]. OTN has become an important bearer solution for 5G technology, which also requires OTN to have flexible management capabilities and high-quality network performance [3]-[4].

In recent years, the rise of SDN and AI makes it almost inevitable to combine these two promising technologies for an unprecedented level of network automation [5]-[6]. The introduction of SDN into optical networks, i.e., software-defined optical network (SDON), is used to trigger unified control and orchestration, allowing for separation of control and data planes in various degrees of centralization [7]-[8]. At the same time, the openness and programmability of SDN provide the perspectives for adopting AI-based optimization algorithms to improve network performance [9]-[10]. With the great analysis and fitting performance for multi-dimensional data, AI has been demonstrated to play an important role in the optimization of optical networks [11]-[12]. It has been used as an advanced tool to deal with complex problems in optical networks from the following two perspectives. In terms of optical transmission, AI is mostly used to tackle fiber linear/nonlinear impairments [13]. For example, detectors based on Parzen Windows are used to mitigate both deterministic fiber nonlinearities and stochastic nonlinear signal-amplified spontaneous emission (ASE) noise interactions [14]. Moreover, AI is also used to estimate crucial signal parameters. A simple artificial neural network (ANN) is used to estimate the quality of transmission (QoT) of unestablished lightpaths [15]. A convolutional neural network (CNN) and constellation diagrams-based method is proposed to estimate the optical signal-to-noise ratio (OSNR) accurately [16]. In the aspect of optical networks, AI algorithms are mostly used for network-level optimization and improving network reliability. Some studies use AI algorithms to allocate network resources to achieve single/multi-objective optimization [17]-[19]. Other studies analyze network traffic

Manuscript submitted September 9, 2020. This work was supported in part by National Natural Science Foundation of China (NSFC) Project (Grant No. 61901053, 61822105) and China Postdoctoral Science Foundation (2019M650588)

Zhuotong Li, Yongli Zhao, Yajie Li, Mingzhe Liu, Zebing Zeng, Xiangjun Xin, and Jie Zhang are with State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications, Beijing, 100876, China (email: yonglizhao@bupt.edu.cn).

Feng Wang, Xinghua Li are with Ningxia Grid Information &Telecommunication Company, Yinchuan, 750001, China.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.3030957, IEEE Open Journal of the Computer Society

and device performance data to predict network failures via AI algorithms [20]-[21]. These studies indicate that the AI-based optimization techniques and methods are becoming more and more mature in optical networks.

However, the proper deployment of AI in SDON is still an open issue. A few AI-based network control and management schemes are designed to facilitate AI-assisted network automation in software-defined elastic optical network [22]-[23]. These schemes analyze network data to predict network status and implement automated management decisions. Futhermore, the workflow of AI model needs to be considered in the design of the network architecture based on AI, including the training, testing and application. There are several challenges:

- *Unified control*: To introduce AI in the SDON control plane, it is necessary to design the workflow of the overall AI functions. The control plane should be able to implement unified control operations on the storage and call of the AI model. In addition, since the data required by the AI model may be different from the data required by traditional algorithms, it is necessary to redesign the interface to facilitate the data collections.
- *Computing resources*: In the training process, the AI engine needs to be fed with a large amount of data, which means that the AI-related components require storage resources to save data sets and computing resources to update model parameters. AI modules are usually located on a resource-rich device such as the centralized controller in SDON. This will undoubtedly increase the burden on the centralized controller. However, for the testing process, since the testing data set is much smaller than the training data set, the testing process does not require high storage or computing resources. Therefore, it is necessary to allocate appropriate computing resources for AI engine.
- *Hierarchical optimization*: In some cases, the AI engine needs to support a real-time response. However, due to the delay in reporting network element data, it is difficult for the AI engine in the centralized controller to handle local problems on the equipment-side in real time. Since such many-to-one synchronization causes heavy workload to the centralized controller, the controller-side AI is not an ideal solution for equipment-side problems. Thus, it is necessary to combine the characteristics of AI for hierarchical control and optimization.

To improve the capabilities of network control and management, a novel optical network architecture, i.e., self-optimizing optical networks (SOON) was proposed [24], which integrates AI and SDON. The network architecture integrates AI technology to improve network intelligent control and management capabilities. In this architecture, the efficient transmitting massive data benefits from an AI-oriented southbound extension protocol. Moreover, SOON can uniformly manage AI models, and quickly processes and trains data for different service requirements. In addition, in order to solve the problem of uneven distribution of computing resources and hierarchical optimization, we introduced on-board AI to SOON [25], and proposed several cloud-edge collaboration modes to improve the network control capability. The collaboration of control layer AI and multiple on-board AI can effectively improve efficiency of model training and testing, and rationally use computing resources to provide rapid response to different application requirements. This paper provides an integrated review of the evolution of SOON with cloud-edge collaboration. We begin with the evolution of SOON and elaborate on the key module functions that need to be implemented in the combination of AI and SDN. We also introduce the idea of introducing on-board AI and achieving the cloud-edge collaboration in this architecture. Then, the SOON testbed based on cloud-edge collaboration and several cloud-edge collaborative strategies are introduced and validated. Finally, some AI-based optimization applications are reviewed.

The rest of this paper is organized as follows. Section II presents an overview of SOON. In section III, on-board AI is introduced and deployed in SOON to achieve the cloud-edge collaboration. In section IV, we introduce the SOON testbed and cloud-edge collaborative strategies. Section V reviews some innovative AI-based network optimization applications. Finally, we summarize this paper.

II. Self-optimizing Optical Networks

Due to the diversification of optical network services, network management needs to be gradually intelligent. The rapid development of optical network technology has spawned a series of intelligent optical network architectures, such as automatically switched optical networks (ASON) and path computation elements (PCE). ASON introduced the control plane in optical networks for the first time to solve the problem of manual and complicated resource allocation and management capabilities of telecommunications management network (TMN) [24]. The control plane can collect and diffuse network topology information, quickly and effectively configure service connections, and reconfigure or modify service connections. In order to solve the problem of complex path calculation in large multi-domain and multi-layer optical networks, the internet engineering task force (IETF) proposed the PCE model in 2006 [27]. The PCE architecture separates the path calculation function from the network management system and is carried by dedicated resources. PCE is actually a logical functional component, which achieves the optimization of inter-domain routing by sharing part of the inter-domain information.

The PCE architecture can solve the problem of multi-domain optical network interconnection under the existing heterogeneous transmission system. However, the path calculation and connection control process of the architecture are highly related to the transmission system. Driven by new switching equipment or methods, a solution is needed to support the smooth network upgrade. SDN provides a good idea for this. In 2009, the concept of SDN was proposed based on OpenFlow [28]. The idea of SDN is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.3030957, IEEE Open Journal of the Computer Society

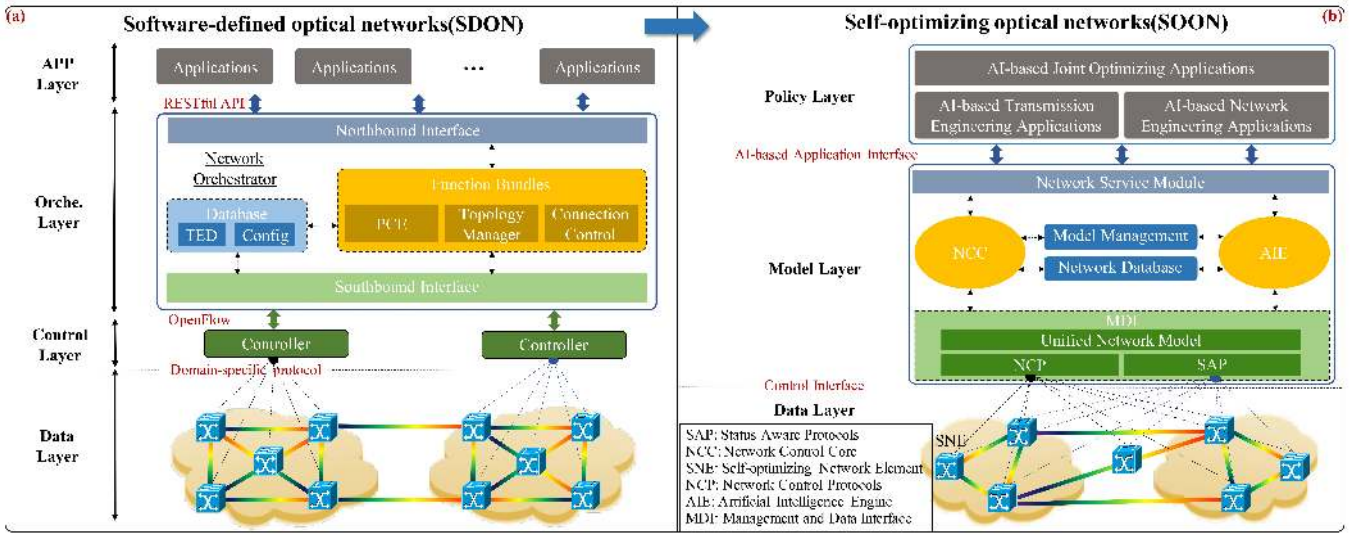

Fig. 1. Comparison of two network architectures: (a) SDON; (b) SOON.

to separate the control plane and data plane of network equipment. Introducing SDN into optical networks, i.e., SDON (as shown in Fig. 1(a)), can solve the problems of scalability, flexibility and smooth upgrade of optical networks. The programmability of network functions and protocols in SDON is beneficial to potentially promote the coordination and orchestration of network services [29]. SDON abstracts the lower layer resource information into the common application programming interface (API) functions of upper layer applications. The network devices in each domain in the data layer are managed by the local controller. The main components in the orchestration layer, the network orchestrator, collects network information from the optical domains and stores it in a database through the southbound interface based on the OpenFlow protocol. This information is used for functional bundles to construct various applications through the northbound interface.

TABLE I
COMPARISON BETWEEN CONTROL-LAYER AI AND ON-BOARD AI

Evaluation	Control-layer AI	On-board AI
Model training	support	**partial support**
Model testing	support	support
Computing resource	huge	small
Power consumption	high	**low**
Delay for network-level application	low	high
Survivability	weak	strong
Price	high	low

With the development of AI technology, more and more AI-based algorithms are used to optimize optical networks. As shown in Fig. 1(b), a new network architecture, i.e., SOON, was proposed for deploying AI in SDON [24]. Compared to SDON, SOON is a three-layer network architecture. The traditional network control core (NCC) and AI engine (AIE) are concentrated in the model layer. The model layer collects data from the underlying network and uses unified AI model management and network database to support the AI-based applications in the policy layer. It is worthy that the training of AI functions requires massive multi-dimensional data, which affects the design of the interface protocol for data transmission between the data layer and the model layer. SOON performs data transmission through the management and data interface (MDI), which contains two types of protocols and a unified network model. The network control protocols (NCP) that include traditional network protocols, such as the general multi-protocol label switching (GMPLS) protocol stack in ASON, the path computing element communication protocol (PCEP) in PCE, the OpenFlow protocol in SDON, etc. In addition, MDI utilizes the state aware protocols (SAP) to perceive massive amounts of detailed information of network elements (named self-optimizing network element (SNE)) about physical components, such as optical signal-to-noise ratio (OSNR) and environment temperatures, etc. The unified network model collects and filters data information for NCC and AIE, which could gather and filter all information from networks, and reformat these data by following some unified network model format. The model management module provides a well-trained model storage library for AIE to perform unified operations on models. The network database is a data source in the model layer, which contains traffic engineering database, data plane status database, etc. In this way, the modules in the model layer cooperate with each other to achieve the upper-layer AI-based optimization applications, including optical transmission-oriented, optical network-oriented, and joint optimization applications.

III. SOON WITH CLOUD-EDGE COLLABORATION

Actually, the data layer in the SOON architecture can be a physical optical network or other entities that can provide historical records and real-time network status information. The model layer and policy layer may be located on the deep integration between the traditional SDN controller cluster (such as ONOS and Opendaylight) and the platform that supports AI (such as Tensorflow and Pytorch). In this way, the training of AI models requires a large amount of computing resources, which will undoubtedly bring a computing burden to the central cloud controller. In addition,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.3030957, IEEE Open Journal of the Computer Society

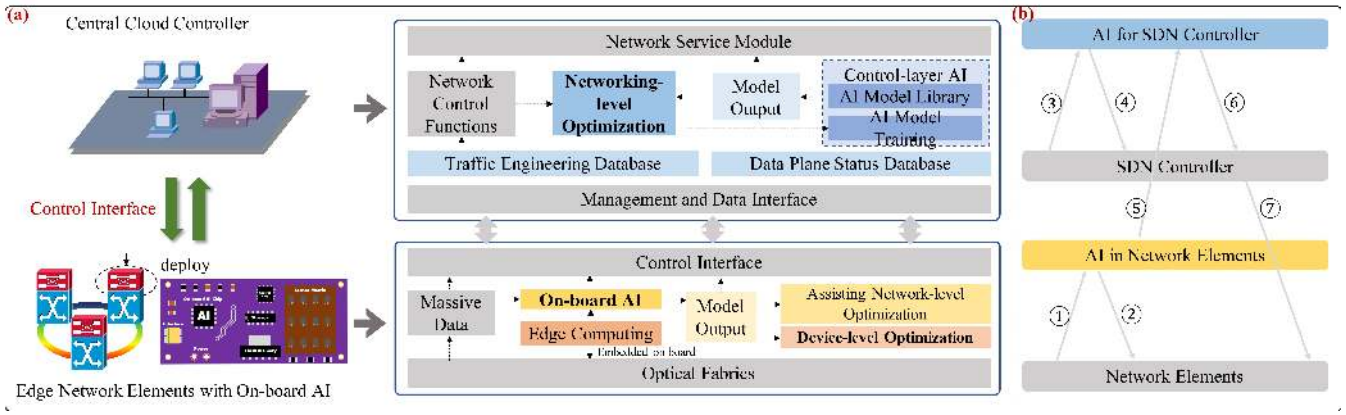

Fig. 2. SOON with cloud-edge collaboration: (a)function modules; (b)collaborative workflow.

the deployment of AI engine only on the central controller cannot realize the device-side optimization quickly. Therefore, the concepts of control-layer AI and on-board AI are proposed and introduced into SOON to construct an intelligent optical network with cloud-edge collaboration [30].

The control-layer AI is the aforementioned AI engine deployed in the SDN-based central controller, which can achieve network-level optimization. On-board AI is deployed on network devices in the data layer. Therefore, on-board AI can provide a faster response for device-side optimization and data processing than control-layer AI. On-board AI exists in the form of an embedded AI board that can be inserted into an expandable slot of an optical network device (as shown in Fig. 2(a)). Table I compares the different performances of the control-layer AI and the on-board AI. According to the product of embedded AI boards of multiple vendors (e.g., Xilinx, Cambricon Technologies, Horizon Robotics, etc.), on-board AI has the characteristics of low cost and low power consumption, which also limits its computing power and training functions. Thus, deploying on-board inside the network devices will not bring a large power consumption and computing burden to networks. In this way, multiple AI boards can be deployed in a network transmission device to achieve scalable and flexible data processing capabilities. Meanwhile, such on-board AI can also be distributed to deploy on multiple network devices in the data layer, which means that it has stronger survivability than the control-layer AI. Due to the characteristics of centralized control, the functions of network devices in the data layer have been simplified, which means that the device side lacks the ability to support AI engines. Therefore, while deploying AI on network devices, edge computing needs to be introduced to enhance the data processing and storage capabilities for on-board AI. On-board AI can access all data of local devices, including network performance and device status. With these data, on-board AI can not only quickly solve device-level optimization problems, but also process original data and collaborate with the control-layer AI to assist the network-level optimization.

Fig. 2(a) shows the collaboration between functional modules after deploying on-board AI in SOON. It is worth noting that the training and testing of AI models depends on on-board AI performance and service requirements. Table I shows the computing capabilities and functions of the current embedded AI board are limited. Only few embedded AI products support the training of complex AI models, and most products only support the testing of models or the training of simple model. Therefore, the training process of complex AI models still needs to be completed at the control-layer AI. For on-board AI supporting model training, different strategies of distributed model training need to be adopted according to the delay requirements of services on the device side and the state of available computing resources in edge computing nodes.

Fig. 2(b) shows the collaborative workflow of the control layer AI and the on-board AI. In the data plane, the on-board AI collects network device status data for analysis and optimization to solve local optimization problems. The control layer AI in the policy layer can learn through the massive data reported by the network to solve the network-level optimization problem. In addition, the on-board AI can also interact with the control layer AI to solve the network layer optimization problem.

IV. Cloud-edge Collaboration Strategy

Faced with different performances of on-board AI, service requirements, and computing resource status in network, different cloud-edge collaboration strategies can be used to train and apply AI models. In this section, several implementation strategies based on cloud-edge collaboration for AI applications are discussed and verified. We will start with the SOON testbed based on cloud-edge collaboration and introduce the cross-layer optimization with a single edge node. Then the distributed model training and inference strategy based on the collaboration of central cloud and multiple edge computing nodes will be discussed.

A. SOON testbed with cloud-edge collaboration

As shown in Fig. 3, the SOON platform is constructed by the ONOS controller and Tensorflow. Tensorflow is integrated in the ONOS controller to provide AI algorithm and service development support. In addition, a unique graphical user interface (GUI) is developed to perform unified operations on the AI model library and service requirements, and display application optimization effects. The platform can call the corresponding model in the AI algorithm library according to the needs of the AI-based

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.3030957, IEEE Open Journal of the Computer Society

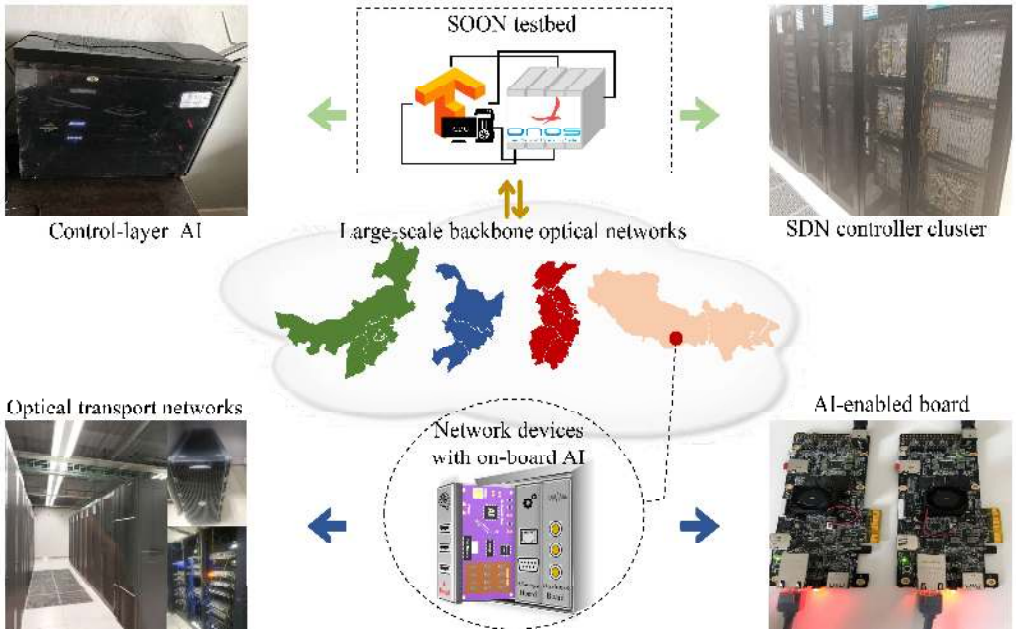

Fig. 3. SOON testbed with cloud-edge collaboration.

applications, and use the data collected in the network for offline feature extraction and model training. The model manager calls the trained model for online testing and application according to requirements. The control layer AI is supported by a high-performance computer with two powerful GTX1080Ti GPUs in this experiment. The SDN controller cluster shown in the figure cooperates with the GPUs to control the network intelligently. All kinds of training data come from the real large-scale backbone optical networks in multiple regions. On-board AI is implemented using multiple AI embedded boards, which are connected to each other through wireless communication.

B. Cross-layer Optimization based on Optimal Model

SOON with cloud-edge collaboration can achieve cross-layer optimization. The central cloud controller can train the AI model according to the data in the database and store the AI model. On-board AI can directly perform data processing and AI model application locally according to optimization requirements. In the model training process, the choice of hyperparameters has a greater impact on the performance of the AI algorithm, such as the number of neurons in neural networks and learning rate. The rapid implementation of multi-model training/testing and the selection of the best model can improve the efficiency and effect of network cross-layer optimization.

A general cross-layer optimization strategy was proposed to solve this problem [30]. As shown in Fig. 4(a), cloud AI in the central controller is responsible for training the model on the training data set. After each period or specific iteration, the model needs to be saved in the database. Then, on-board AI downloads the model and executes the test on the test data set. The downloaded models need to be compressed, compiled and run to adapt the on-board AI. With the same hyperparameters to complete multi-stage training, according to the type of application, choose the best model to deploy on the central controller or optical equipment.

During the verification of the strategy, two DP-8020 boards developed by Xilinx were used as on-board AI. Fig. 4(b) shows the training effect of 100 models with various combinations of two hyperparameters, the input neuron number and learning rate. This strategy can quickly select

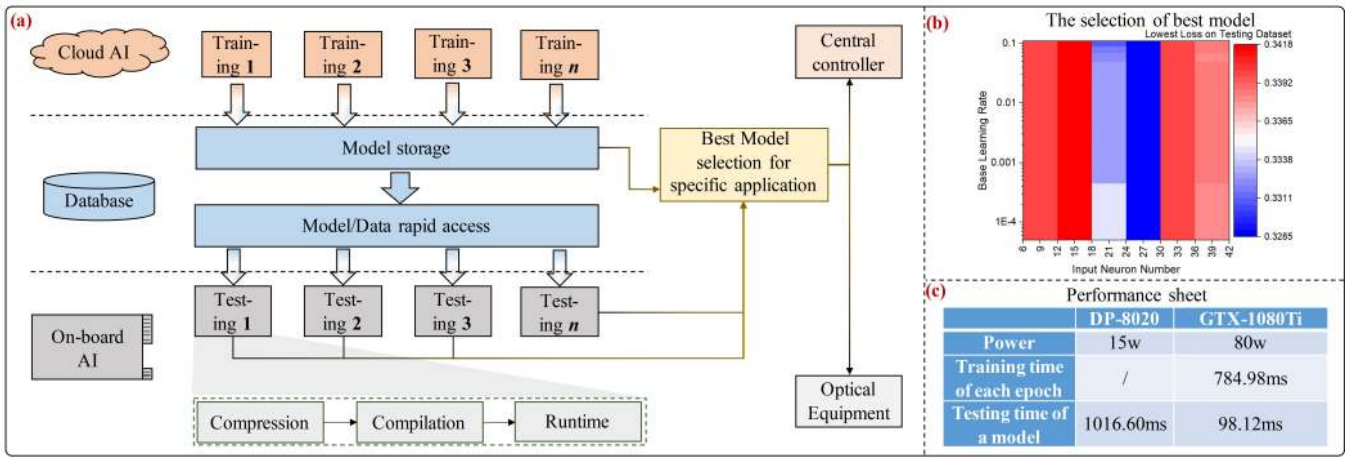

Fig. 4. Cross-layer optimization: (a)collaborative training and testing of models; (b) the selection of best model; (c)the performance sheet of control-layer AI and on-board AI.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.3030957, IEEE Open Journal of the Computer Society

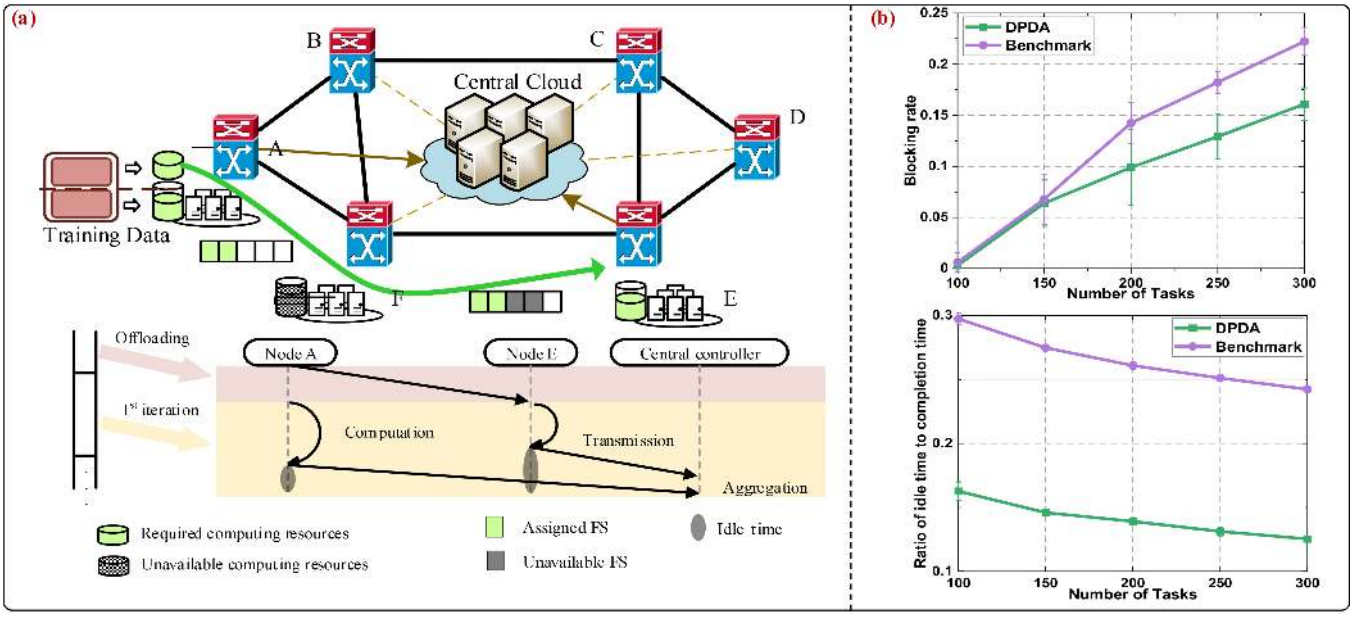

Fig. 5. Distributed training of AI models based on data parallelism: (a) model training process based on data parallelism; (b) performance results of DPDA.

the best model. Fig. 4(c) shows the actual performance collected from the experiment. The power of DP-8020 is much smaller than GTX 1080Ti. In addition, we also reported the time consumption of the control-layer AI and the on-board AI when processing an epoch data. The average time consumption of a training epoch in the controller is about 784.98ms, and the average time consumption of testing on the on-board AI is about 1016.60ms. Since two boards are used at the same time in this experiment, the test time consumption is reduced to 508.30ms. The test time consumption is lower than the training consumption in cross-layer collaboration mode, which means that testing with on-board AI will not block the training of the control-layer AI. This strategy saves the additional 98.12ms test time consumption than the case where AI models are only trained and tested using the control-layer AI.

C. Distributed Training of AI Models based on Data Parallelism

Some AI embedded boards support the training of simple AI models, which makes it possible to perform distributed training of AI models in SOON. Due to the limitations of the computing and data caching capabilities of on-board AI, the collaboration of edge nodes and cloud node is required to implement the model training process. The collaboration can shorten training time and reduce the computing resource requirements on a single node. In the research of AI model distributed training, synchronous training has been validated by splitting and distributing the training data on multiple edge nodes [31]-[32]. As shown in Fig. 5(a), during each iteration, all edge nodes independently train the model and send model parameters to the cloud. The model will be returned to the edge node after the AI in cloud controllers summarizes and update parameters. Once the accuracy of model is reached, the training process is stopped. In this process, there is still the problem of how to achieve the dynamic allocation and deployment of training data for multiple training tasks. Specifically, the scheme of data partition and training deployment will affect the use of computing and transmission resources in the network. Given a batch of training tasks, the cloud controller needs to find the best data partition and deployment to perform as many training tasks as possible.

A data parallelism deployment algorithm (DPDA) is proposed to solve the training tasks deployment problem. DPDA first searches candidate offload edge nodes for each training task and calculates the resource occupancy factor of each candidate offload node. Secondly, DPDA performs routing and spectrum allocation (RSA) for the shortest path between task source node and candidate nodes with low resource occupancy to transmit training data. If there are not enough resources to support the training task, the request is terminated. Finally, divide and deploy training data according to the proportion of available computing resources in edge nodes. The DPDA algorithm uses ILP to model the deployment problem. In the process of partitioning data and selecting transmission path, the algorithm designs resource constraints and time consumption based on the training task (including the size of the training data, the maximum tolerable task delay requirement, etc.) and the available computing and frequency slot resources in the network. The time consumption factors include the time consumption for offloading data edge nodes, the calculation, and the data transmission to the cloud controller. The objective is to jointly minimize the resources cost and the average time to complete a training task. The complexity of DPDA depends on network size, FS number per link and the offloading request of tasks.

In the verification of the algorithm, a benchmark algorithm is designed for comparison, which selects the closest edge node to the source node for each request to offload training data. Fig. 5(b) shows the performance results of DPDA. The comparison with the benchmark algorithm illustrates that DPDA can deploy more training tasks under limited network resources. As the number of tasks increases, the task blocking rate of DPDA is about 5% lower than the benchmark algorithm. In addition, the ratio of idle time to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.3030957, IEEE Open Journal of the Computer Society

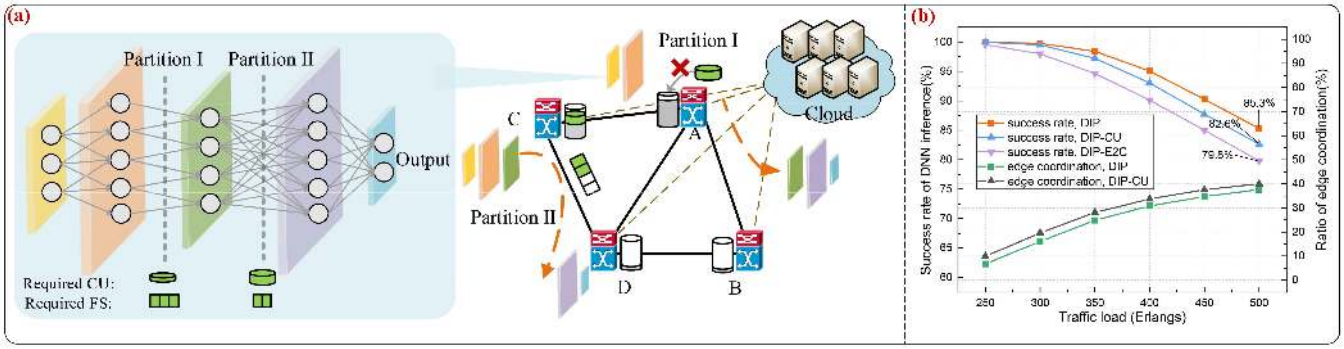

Fig. 6. DIaaS: (a) DNN inference process based on cloud-edge collaboration; (b) performance results of DIP.

task completion time of the algorithm is relatively low, which means that the computing resources of each node can be effectively used.

D. DNN Inference as a Service

The inference of deep neural networks (DNN) can also benefit from SOON with cloud-edge collaboration. In order to reduce the computational burden of a single node or satisfy the delay requirements of different services in SOON, a concept of flexibly adjusting the DNN inference process was proposed, which called DNN inference as a service (DIaaS) [33]. DIaaS refers to the on-demand provisioning of DNN inference based on flexible model partition and distribution according to service requirements and network resources. Specifically, as shown in Fig. 6(a), a DNN model with multiple layers can offload some layers to edge or cloud nodes to complete the overall inference process of DNN. This partition of DNN model can effectively reduce the inference delay and the calculation burden of a single node. This process involves two issues: the partitioning of multi-layer DNN and the deployment of computing nodes, which requires comprehensive consideration of service delay requirements, node computing resources and spectrum resources used for data transmission.

We designed the DNN inference provisioning (DIP) algorithm to realize the DIaaS, with the aim of maximizing the inference provisioning [30]. According to the requirements of inference delay and network resource availability, the DIP algorithm can select the most suitable DNN partition and inference deployment between the edge and the cloud for each task. In the process of DNN partitioning and offloading, two network resources need to be considered: i) the available computing units (CU) of each node for data caching, and ii) the available frequency slots (FS) for data transmission. The DIP algorithm is constrained to the edge node where the inference task originates, the input data size and the maximum tolerable delay of the task. The sizes of intermediate data and transmission route selection all affect the task delay. Network resource metrics (CU and FS metrics) are used as load balance metrics to evaluate candidate solutions for each model partition deployment. Finally, DIP will select the candidate with the lowest load balance metric as the best solution for model partition and deployment.

To verify the effectiveness of the DIP algorithm, ResNet-18 was used as the DNN model. DIP-CU and DIP-E2C are designed as comparison algorithms. The former only considers the usages of computing resources and ignores the transmission resources during model partitioning. The latter only studies the DNN partition between edge nodes and cloud nodes, without considering the coordination of edge nodes. As shown in Fig. 6(b), higher traffic load leads to higher resource utilization, which makes it more difficult to meet the inference delay requirements. Compared with the other two algorithms, DIP has the highest success rate. When the traffic load is set to 500 Erlangs, the success rate of DIP reaches 85.3%. Moreover, the ratio of edge coordination is the ratio of inference tasks completed by edge coordination to the total inference of services. Fig. 6(b) shows that DIP and DIP-CU are close in terms of the ratio of edge coordination.

V. INNOVATIVE AI-BASED APPLICATION

Various innovative AI-based optimization applications are developed within the SOON testbed. These applications are designed to use AI technology to provide users with network optimization services. In this section, several important use cases are reviewed.

(1) Alarm prediction. A single fiber/node failure in optical networks may cause massive service interruption and heavy economic loss, even for a few seconds. Alarms are the most direct manifestation of network failures. Therefore, predicting alarm information can provide advantages for network administrators to deal with faults in a timely and effective manner. The alarm prediction use case can perform data preprocessing and data enhancement on a large amount of dirty data reported in networks, combining AI algorithms and knowledge-based collective self-learning methods to extract the features of performance data in multi-domain networks to predict the next time series of alarms [34]-[35].

(2) Resource allocation. Allocating resources for network services in optical networks has always been the focus of research. There are many heuristic algorithms for routing and wavelength assignment (RWA) that can only achieve approximately optimal performance under certain circumstances. The resource allocation optimization application in SOON uses reinforcement learning (RL) to make resource allocation decisions for multi-modal optical networks to maximize the utilization of network resources [36]. In addition, the application also considers the

constraints of other resources such as the number of device ports [37].

(3) Fault localization. A major difficulty in dealing with faults in optical networks is that the complex relationship between the alarms will interfere with the identification of root-cause alarms. A single point of fault in networks may cause the reporting of massive alarms, and one alarm will generate multiple alarms. Fault localization use case proposes the concept of alarm knowledge graphs (KGs). According to the alarm knowledge in the equipment manual, the knowledge graph is automatically constructed. And graph neural network (GNN) is used to infer the location of network faults [38]-[39].

VI. Conclusion

The deployment of AI in the optical network is conducive to improving network control capabilities. This paper reviews the evolution of self-optimizing optical network architecture that implements AI services in SDON. In addition, on-board AI is introduced to SOON to achieve the cloud-edge collaboration. Based on this architecture, a SOON testbed and several collaborative strategies have been proposed and verified to improve the efficiency of AI applications in the network and balance computing resources. Finally, we summarize several innovative AI-based use cases. The current AI model training and testing methods based on cloud-edge collaboration are limited by the performance of on-board AI. In the future, as the performance of AI embedded boards improves, more AI service strategies based on cloud-edge collaboration will be proposed, which means that AI services can be provided better in optical networks.

References

[1] Y. Choi, J. H. Kim and C. K. Kim, "Mobility management in the 5G network between various access networks," in *2019 Eleventh International Conf. on Ubiquitous and Future Networks (ICUFN)*, Zagreb, Croatia, 2019, pp. 751-755.

[2] *Whiter Paper on 5G Vision and Requirements*, MT-2020(5G) PG, 2014.

[3] Y. Ji, J. Zhang, Y. Xiao and Z. Liu, "5G flexible optical transport networks with large-capacity, low-latency and high-efficiency," *China Communications*, vol. 16, no. 5, pp. 19-32, May 2019.

[4] J. Montalvo, M. Arroyo, J. A. Torrijos, J. Lorca and I. Berberana, "Fixed-mobile convergence and virtualization in 5G optical transport networks," in *2015 17th International Conf. on Transparent Optical Networks (ICTON)*, Budapest, 2015, pp. 1-4.

[5] A. Mestres, A. Rodriguez-Natal, J. Carner, P. Barlet-Ros, E. Alarcón, M. Solé, et al, "Knowledge-defined networking," *ACM SIGCOMM Computer Communication Review*, vol. 47, no. 3, pp. 2-10, 2017.

[6] M. Wang, S. Liu, Z. Zhu, "Can you trust AI-assisted network automation? A DRL-based approach to mislead the automation in SD-IPoEONs," in *Optical Fiber Communication Conf.*, Optical Society of America, March 2020, pp. Th1F-6.

[7] V. Lopez, J. M. Gran, J. P. Fernandez-Palacios, D. Siracusa, F. Pederzolli, O. Gerstel, et al, "The role of SDN in application centric IP and optical networks," in *2016 European Conf. on Networks and Communications (EuCNC)*, IEEE, 2016, pp. 138-142.

[8] J. Santos, "On the impact of deploying federated SDN controllers in optical transport networks," in *Optical Fiber Communication Conf.*, Optical Society of America, 2016, pp. Th1A-5.

[9] M. Garrich, F. Moreno-Muro, M. Bueno Delgado and P. Pavón Mariño, "Open-source network optimization software in the open SDN/NFV transport ecosystem," *Journal of Lightwave Technology*, vol. 37, no. 1, pp. 75-88, 1 Jan.1, 2019.

[10] H. Zhang, Y. Wang, H. Chen, Y. Zhao, J. Zhang, "Exploring machine-learning-based control plane intrusion detection techniques in software defined optical networks," *Optical Fiber Technology*, vol. 39, pp. 37-42, 2017.

[11] J. Wang, C. Jiang, H. Zhang, et al, "Thirty years of machine learning: the road to pareto-optimal next-generation wireless networks," arXiv preprint arXiv:1902.01946, 2019.

[12] J. Wang, C. Jiang, H. Zhang, et al, "Learning-aided network association for hybrid indoor LiFi-WiFi systems," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 4, pp. 3561-3574, 2017.

[13] F. N. Khan, C. Lu, and A. P. T. Lau, "Machine learning methods for optical communication systems," in *Signal Processing in Photonic Communications*, Optical Society of America, 2017, pp. SpW2F-3.

[14] A. Amari, X. Lin, O. A. Dobre, R. Venkatesan and A. Alvarado, "A Machine Learning-Based Detection Technique for Optical Fiber Nonlinearity Mitigation," *IEEE Photonics Technology Letters*, vol. 31, no. 8, pp. 627-630, 15 April, 2019.

[15] M. Zhang, D. Fu, B. Xu, B. Wu and K. Qiu, "QoT estimation for unestablished lightpaths using artificial neural networks," in *2018 Confer. on Lasers and Electro-Optics (CLEO)*, San Jose, CA, 2018, pp. 1-2.

[16] Z. Wang, A. Yang, P. Guo, L. Feng and P. He, "CNN based OSNR estimation method for long haul optical fiber communication systems," in *2018 Asia Communications and Photonics Conference (ACP)*, Hangzhou, 2018, pp. 1-3.

[17] M. Balanici and S. Pachnicke, "Machine Learning-Based Traffic Prediction for Optical Switching Resource Allocation in Hybrid Intra-Data Center Networks," in *2019 Optical Fiber Communications Conference and Exhibition (OFC)*, San Diego, CA, USA, 2019, pp. 1-3.

[18] W. Mo, C. L. Gutterman, Y. Li, G. Zussman and D. C. Kilper, "Deep Neural Network Based Dynamic Resource Reallocation of BBU Pools in 5G C-RAN ROADM Networks," in *2018 Optical Fiber Communications Conference and Exposition (OFC)*, San Diego, CA, 2018, pp. 1-3.

[19] A. Yu, H. Yang, W. Bai, L. He, H. Xiao and J. Zhang, "Leveraging Deep Learning to Achieve Efficient Resource Allocation with Traffic Evaluation in Datacenter Optical Networks," in *2018 Optical Fiber Communications Conference and Exposition (OFC)*, San Diego, CA, 2018, pp. 1-3.

[20] G. Choudhury, G. Thakur and S. Tse, "Joint Optimization of Packet and Optical Layers of a Core Network using SDN Controller, CD ROADMs and Machine-Learning-Based Traffic Prediction," in *2019 Optical Fiber Communications Conference and Exhibition (OFC)*, San Diego, CA, USA, 2019, pp. 1-3.

[21] L. Cui, Y. Zhao, B. Yan, D. Liu, J. Zhang, "Deep-learning-based failure prediction with data augmentation in optical transport networks," in *17th International Conference on Optical Communications and Networks (ICOCN2018)*, International Society for Optics and Photonics, 2019, vol. 11048, pp. 110482I.

[22] D. Rafique, L. Velasco, "Machine learning for network automation: Overview, architecture, and applications [invited tutorial]," *Journal of Optical Communications and Networking*, vol.10, no.10, pp. D126-D143, 2018.

[23] S. Liu, B. Niu, D. Li, M. Wang, S. Tang, J. Kong, et al., "DL-assisted cross-layer orchestration in software-defined IP-over-EONs: from algorithm design to system prototype," *Journal of Lightwave Technology*, vol. 37, no. 17, pp. 4426-4438, 2019.

[24] Y. Zhao, B. Yan, D. Liu, Y. He, D. Wang, J. Zhang, "SOON: self-optimizing optical networks with machine learning," *Optics express*, vol. 26, no. 22, pp. 28713-28727, 2018.

[25] Y. Zhao, B. Yan, W. Wang, Y. Lin and J. Zhang, "On-board artificial intelligence based on edge computing in optical transport networks," in *2019 Optical Fiber Communications Conference and Exhibition (OFC)*, San Diego, CA, USA, 2019, pp. 1-3.

[26] ITU-T Rec. G.8080/Y.1304, "Architecture for the automatically switched optical network," Feb. 2012.

[27] A. Farrel, J.-P. Vasseur, and J. Ash, "A path computation element (PCE)-based architecture," IETF RFC4655, 2006.

[28] S. Das, G. Parulkar, N. McKeown, "Unifying packet and circuit switched networks," in *Proceedings of 2009 IEEE GLOBECOM Workshops*, Honolulu, HI, 2009.
[29] Y. Ji, J. Zhang, Y. Zhao, X. Yu, J. Zhang, X. Chen, (2016). "Prospects and research issues in multi-dimensional all optical networks," *Science China Information Sciences*, vol. 59, no. 10, pp.101301, 2016.
[30] Y. Zhao, B. Yan, Z. Li, W. Wang, Y. Wang and J. Zhang, "Coordination between control layer AI and on-board AI in optical transport networks [Invited]," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 12, no. 1, pp. A49-A57, January 2020.
[31] Q. Ho, J. Cipar, H. Cui, S. Lee, J. K. Kim, P. B. Gibbons, et al., "More effective distributed ml via a stale synchronous parallel parameter server," in *Advances in neural information processing systems*, 2013, pp. 1223-1231.
[32] Y. Lin, S., Han, H. Mao, Y. Wang, W. J. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," arXiv preprint arXiv:1712.01887, 2017.
[33] M. Liu, Y. Li, Y. Zhao, H. Yang, J. Zhang, J, "Adaptive DNN model partition and deployment in edge computing-enabled metro optical interconnection network," in *Optical Fiber Communication Conf.*, Optical Society of America, March 2020, pp. Th2A-28.
[34] X. Xing, Y. Zhao, Y. Li and J. Zhang, "Knowledge-Based Collective Self-learning for Alarm Prediction in Real Multi-Domain Autonomous Optical Networks," in *2020 16th International Confer. on the Design of Reliable Communication Networks DRCN 2020*, Milano, Italy, 2020, pp. 1-5.
[35] B.Yan, Y. Zhao, S. Rahman, Y. Li, X. Yu, D. Liu, et al., "Dirty-data-based alarm prediction in self-optimizing large-scale optical networks," *Optics express*, vol. 27, no. 8, pp. 10631-10643, 2019.
[36] B. Yan, et al., "Actor-Critic-Based Resource Allocation for Multi-Modal Optical Networks," in *2018 IEEE Globecom Workshops (GC Wkshps)*, Abu Dhabi, United Arab Emirates, 2018, pp. 1-6.
[37] H. Ma et al., "Demonstration of Image Processing Based on Reinforcement Learning in Multi-Modal Optical Transport Networks," in *2019 18th International Conf. on Optical Communications and Networks (ICOCN)*, Huangshan, China, 2019, pp. 1-3.
[38] Z. Li et al., "Demonstration of alarm knowledge graph construction for fault localization on ONOS-based SDON platform," in *2020 Optical Fiber Communications Conf. and Exhibition (OFC)*, San Diego, CA, USA, 2020, pp. 1-3.
[39] Z. Li, Y. Zhao, Y. Li, S. Rahman, X. Yu and J. Zhang, "Demonstration of fault localization in optical networks based on knowledge graph and graph neural network," in *2020 Optical Fiber Communications Conf. and Exhibition (OFC)*, San Diego, CA, USA, 2020, pp. 1-3.

Zhuotong Li received his B.S. degree in Communication Engineering from Southwest Jiaotong University (SWJTU) in 2018. He is currently working toward the Ph.D. in Information and Communication Engineering at Beijing University of Posts and Telecommunications (BUPT). His research focuses on Software Defined Optical Networks and Artificial Intelligence.

Yongli Zhao is currently a professor of institute of information photonics and optical communications at Beijing University of Posts and Telecommunications (BUPT). He received the B.S. degree in communication engineering and Ph.D. degree in electromagnetic field and microwave technology from BUPT, respectively. During Jan. 2016 to Jan. 2017, he was a visiting associate professor at UC Davis. Since 2018, he has become a full professor of BUPT. He has published more than 300 international journal and conference papers. Since 2015, he became a senior member of IEEE. His research focuses on software defined optical networks, elastic optical networks, machine learning in optical networks, and optical network security.

Yajie Li received his Ph.D. degree in communication and information system from Beijing University of Posts and Telecommunications (BUPT), in 2018. He is currently working as a post doctor in BUPT. He was a visiting doctoral student at KTH Royal Institute of Technology from Oct. 2016 to Dec. 2017. His research interests include software defined optical networks, edge computing, artificial intelligence and 5G optical transport networks.

Mingzhe Liu received his B.S. degree (2018) from University of Electronic Science and Technology of China (UESTC), Chengdu, China. He is a M.S. candidate in Electronics and Communication Engineering at Beijing University of Posts and Telecommunications (BUPT). His research interests include edge computing, data center optical networks, and artificial intelligence.

Zebin Zeng received his B.S. degree (2019) from Nanjing University of Posts and Telecommunications (NJUPT), Nanjing, China. He is a M.S. candidate in Electronics and Communication Engineering at Beijing University of Posts and Telecommunications (BUPT). His research interests include edge computing, data center optical networks, and distributed machine learning.

Xiangjun Xin received the Ph.D. degrees from the School of Electric Engineering, Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2004. He is currently a Professor with the School of Electric Engineering, BUPT. He is a member of the State Key Laboratory of Information Photonics and Optical Communications, BUPT. His main research interests focus on broadband optical transmission technologies, optical sensor, and all-optical network. Within this area, he has authored or co-authored over 100 SCI papers.

Jie Zhang is currently a professor and the dean of Information Photonics and Optical Communications Institute at Beijing University of Posts and Telecommunications (BUPT), China. He received his bachelor's degree in communication engineering and a Ph.D. in electromagnetic field and microwave technology from BUPT in 1993 and 1998, respectively. He has published more than 300 technical papers, authored 8 books, and submitted 17 ITU-T recommendation contributions, 10 IETF drafts. Also, he holds more than 40 patents. He has served as TPC members for a number of conferences such as ACP, OECC, PS, ONDM, COIN and ChinaCom. His research focuses on architecture, protocols and standards of optical transport networks.