SOON: self-optimizing optical networks with machine learning

YONGLI ZHAO,[1,*] BOYUAN YAN,[1] DONGMEI LIU,[2] YONGQI HE,[3] DAJIANG WANG,[4] AND JIE ZHANG[1]

[1]*State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2]*State Grid Information & Telecommunication Company, Beijing 100761, China*
[3]*State Key Laboratory of Advanced Optical Communication Systems Networks, Peking University, Peking 100871, China*
[4]*ZTE Corporation Co., Ltd, Beijing 100191, China*
*yonglizhao@bupt.edu.cn

Abstract: As optical networks undergo rapid development, the trade-offs among higher network service capability, and increasing operating expense (OPEX) about operations, administration and maintenance (OAM) become telecom operators' key obstacles. Intelligent and automatic OAM is considered to effectively satisfy service requirements, while dampening OPEX growth. In particular, machine learning (ML) has been investigated as a possible method of replacing human image recognition, nature language processing, automatic drive, and so forth. This is because of its essential feature extraction ability. ML application in optical networks was studied in a preliminary way recently. In ML-enabled optical networks, huge data storage and powerful computing resources are required to handle computer-intensive tasks performed in order to analyze features from big data sets. Integration of these two key resources into existing optical network architectures, in order to improve network performance, is an emerging challenge for ML-enabled optical networks. This article proposes a novel optical network architecture, which is based on software-defined networking (SDN), which is also named self-optimizing optical networks (SOON). First, we comb through intelligence development of optical networks, and introduce SOON as an OAM-oriented optical network architecture. Second, we demonstrate four typical applications within SOON, including tidal traffic prediction, alarm prediction, anomaly action detection, and routing and wavelength assignment. Finally, we discuss some open issues.

1. Introduction

Accompanied with the rapid development of information technology in the age of internet and information, the size of the optical networks increases rapidly. The number of heterogeneous devices in the network is increasing, and the service types, such as virtual reality (VR), the fifth generation wireless system (5G), are attracting more and more attention. Thus, inefficient traditional management for optical networks is gradually becoming the bottleneck for network development. Machine learning (ML) has been a potential method to solve this problem recently due to its breakthrough on applications about artificial intelligence. ML was coined in [1], whose studies mainly includes pattern recognition and computing learning theory. At present, ML has been widely used in various industries. For example, in the field of image classification, ML achieves and further exceeds human-level recognition capability [2–4]. In gaming, AlphaGo [5], developed by Google's DeepMind Research Group, beats the world's Go champion. Google also develops a new ML-based computer program, AlphaGo Zero [6], which beats AlphaGo without relying on human guidance and experience. And until now, some researchers have studied the application of ML in optical networks. A ML-based method is introduced for solving fiber

linear/nonlinear damage and estimating key signal parameters in optical networks [7]. In order to reduce the occurrence of faults in optical networks, the deep learning is used to predict faults, and reach 95% accuracy rate [8]. In Ethernet Passive Optical Networks (EPONs), in order to optimize video transmission, a nonlinear auto-regressive neural network model is proposed to predict the H.265 video bandwidth requirements, and the proposed model can reach accuracy rate more than 90% [9]. ML-based control plane intrusion detection schemes are also proposed in SDON. Moreover, results show that the accuracy of intrusion detection scheme can reach more than 85% [10].

By introducing ML in optical network management, optical networks are able to implement intelligence of control plane, dynamic resource self-adaptation and self-management for network events, smarter and optimized routing for spectrum resources. Based on software-defined optical networks (SDON) [11], Self-Optimizing Optical Networks (SOON) is proposed in this paper to improve performance of network management by introducing ML into optical networks. To explain it in detail, the rest of this paper is as follows. Section 2 describes intelligent evolution in optical networks. Section 3 introduces SOON architecture and talks about its key technologies. Section 4 shows four potential applications done by our team currently. Section 5 talks about the open issues with SOON. And finally, section 6 gives a conclusion.

2. Evolution of intelligent optical networking

In order to adapt to the emergence of various services, it's necessary for optical networks to achieve intelligence. With the technology development of optical networks in recent years, intelligence on different aspects is achieved in different network architectures. As shown in Fig. 1, we can summarize that optical networks have experienced three important stages of intelligence development.
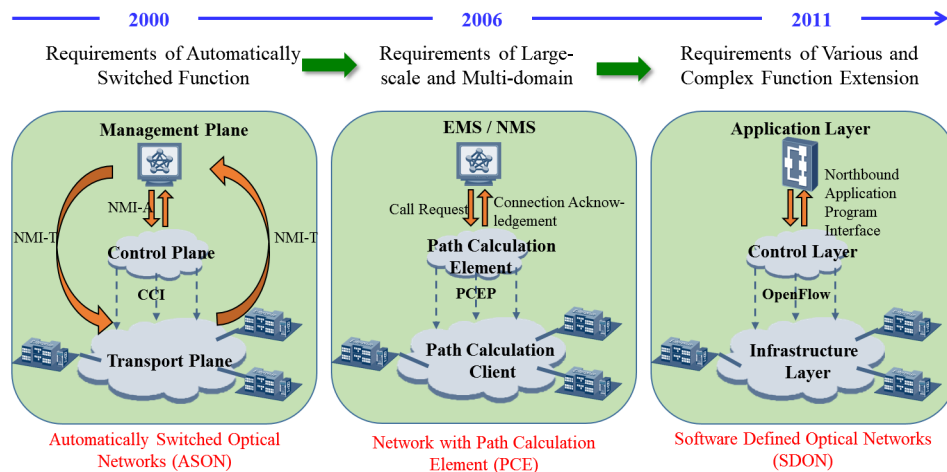

Fig. 1. The evolution of intelligent optical networks.

2.1 Automatic switched optical networks (ASON)

Most functions about optical network control and management are integrated in telecommunications management network (TMN) [12] before ASON era. With the development of optical networks, the questions about automatically switched function, and other resource control function become too complex for TMN. In Sept. 2000, ITU-T attempted to introduce control plane in optical networks to build a new optical network architecture on an ITU-T meeting, which is called ASON. In ASON, optical networks consist of management plane, control plane and transport plane from top to bottom. Transport plane provides connection control interface (CCI) to control plane, and network management

interface for the transport network (NMI-T) to management plane [13]. Control plane is the core of ASON, also the most important innovation. Control plane is responsible for topology discovery, connection set-up/tear-down, connection protection/restoration, traffic engineering, and etc. through CCI. Management plane execute network element management via NMI-T, and network resource management via network management interface for ASON control plane (NMI-A). The management plane can be implemented based on simple network management protocol (SNMP), or other protocols.

2.2 Path computation element (PCE) architecture

In large multi-domain, multi-region, or multi-layer optical networks, constraint-based path computation is complex. Especially, cross-domain path calculation may require special computational components and inter-domain cooperation. To address this problem, the internet engineering task force (IETF) published PCE model in 2006 [14]. PCE architecture is a generalized model for both ASON-capable networks and ASON-incapable networks, like optical transport network (OTN) that provide flexible bandwidth adjustment on IP layer and optical layer. PCE architecture separates the path computation function from network management system, and puts it on path computation element (PCE) entity. Path computation client (PCC) is the path calculation service requester to ask PCE. PCC could be network element, network manager, and others that need to get path information. PCE is actually a logical functional component, so that it could be centralized on a single machine or distributed on multiple machines. PCE architecture is mainly used in the path calculation at the beginning, and needs to be coordinated with other technologies.

2.3 Software defined optical networks (SDON)

SDON can be considered as the application of SDN in optical networks [15], which can provides a unified scheduling and control capability for various optical layer resources. According to the requirements of users or operators, SDON can be customized dynamically by software programming. SDON focus on quickly responding requests, efficiently utilizing resources and flexibly providing service to meet the diverse and complex needs. The technical characteristics of SDON are mainly shown as follows: programmable lightpath transmission adjustment, programmable flexible lightpath switching, and programmable lightpath automatic networking, etc. Programmable lightpath transmission adjustment is the online adjustment of the optical transceiver wavelength, input and output power, modulation format. The programmable flexible lightpath switching support programmable ROADM technology of flexible grid, which breaks out the traditional fixed-wavelength channel raster division and support all-optical aggregation and traffic grooming. SDON has achieved high spectral efficiency, flexible lightpath configuration and bandwidth management. Programmable lightpath automatic networking can achieve automatic networking of heterogeneous devices. Through programmable control of devices, algorithms, strategies and protocols, SDON completes the control of heterogeneous network resources and dynamic tuning of the overall network. The SDON opens the northern interface to support various flexible application of business. The key feature of SDON is to accomplish programmable tuning of optical network element states. Therefore, SDON is more suitable for multi-layer multi-constrained optical network control. SDON can improve efficiently operation and maintenance efficiency and reduce cost.

3. Self-optimizing optical networks (SOON)

SDON could provide logically centralized control mechanism, flexible network function via open northbound application program interface (API), and etc. Based on SDON, we propose a new architecture named self-optimizing optical networks (SOON), which integrates ML technology to SDON to improve intelligence of optical networks.
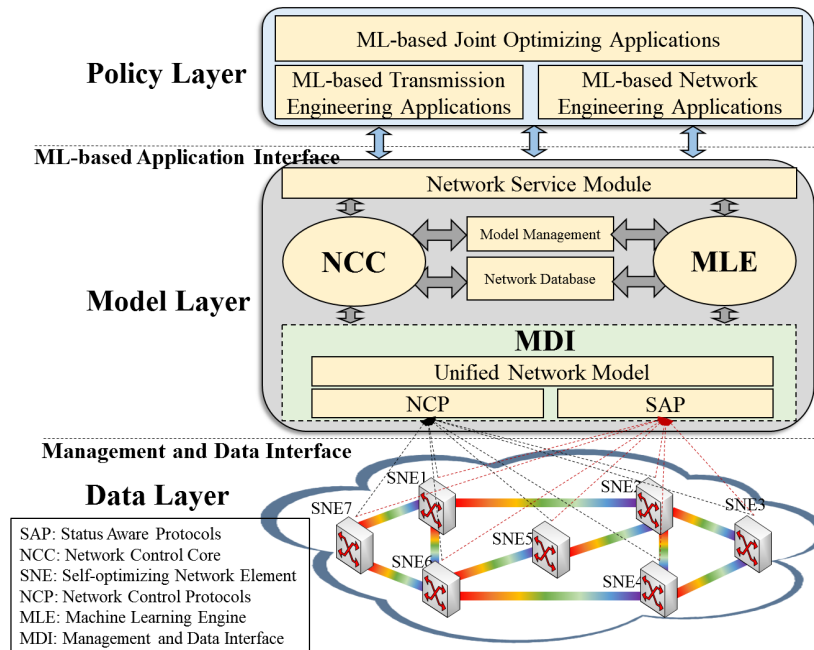
3.1 SOON architecture

Fig. 2. Architecture of SOON.

As shown in Fig. 2, the architecture of SOON consists of data layer, model layer, and policy layer, which is based on SDON architecture. Data layer contains physical/virtual optical networks. Each element of networks, named self-optimizing network element (SNE), would connect to model layer via management and data interface (MDI). MDI consists of two types of protocols and a unified network model on them. Network control protocols (NCP) are traditional network protocols, such as Generalized Multiprotocol Label Switching (GMPLS) protocol stack in ASON, path computation element communication protocol (PCEP) in PCE, OpenFlow protocol in SDON, etc. SNE reports network status and receive instruction through NCP. The main purpose of status-aware protocols (SAP) is to enable model layer to perceive detailed information of network element about physical components, for example, physical parameters like optical signal to noise ratio (OSNR), and environment parameters like temperature. Such an elaborative data acquisition method defined by SAP aims to provide mass data in as many dimensions as possible for ML applications. Besides, SAP should also provide direct control capability for model layer to adjust parameters in SNE. Currently, there is no SAP standards, but a related draft to define a SAP [16]. In MDI, unified network module aims to provide protocol-free interface for network control core (NCC) and machine learning engine (MLE), which could gather and filter all information from NCPs and SAPs, and reformat these data by following some unified network model format.

NCC implements the basic functions of SDN controller, such as topology discovery, routing calculation, bandwidth allocation, and so on. MLE is actually a built-in ML-enabled framework that contains ML algorithm library, and related lifecycle control interfaces. Algorithm library provides various ML algorithm implementations as much as possible. Control interfaces are composed of four fundamental functions, that is, algorithm initialization that configures parameters at beginning, training monitor that evaluates the progress of training, state control that manages if model training should start, pause, or stop, and parameter adjustment that is used to seek proper parameter setup for optimization. There are two modules that connect NCC and MLE indirectly. Network database module is the data

source for any usages in model layer, from which NCC could synchronize network status, and MLE could get the train/test/validation data set. Model management module is a well-trained model repository that is responsible for model creation, model delete, model replacement, and model search. NCC calls for model according to requirements from policy layer, and get final model from model management module. MLE executes instructions from policy layer through network service module to train model on specific data set. Network service module provides a unified interface for policy layer, named ML-based application interface, which enables open and secure access to NCC and MLE.

All ML-based applications are placed in policy layer. These applications execute specific policy with the help of ML model, and control network operations through NCC. Besides the applications about network engineering, benefited from deeper monitoring for SNE in data layer, there also exists such applications that could adjust physical parameters in time to improve the quality of transmission (QoT), named ML-based transmission engineering applications. Application also could handle network/service optimization problem by considering both network status and transmission status jointly, which is named ML-based joint optimizing application.

In reality, data layer could be physical optical networks or other entities that could provide history and real-time network status information. Model layer and policy layer may be located onto the deep integration between traditional SDON controller and ML-enabled platform, which could provide both network control and ML-related functions.

3.2 Key technologies in SOON

There are different technologies in the three layers of SOON respectively to enable intelligence.

a) **In data layer**, the physical parameters of traditional optical equipment are always fixed after deployment, which is too simple to guarantee that equipment could keep best performance as time goes on. Compared with traditional equipment, SNE contains bits of storage and computation resource (for example, special board with graphics processing unit), on which some simple computational tasks could be finished within a short time. The ML algorithm, such as support vector machine or neural network, is able to learn accurately the non-linear influence about modulation format, phase noise, transmission distance on transmission performance. However, the storage and computation resource requirement of such algorithm for different applications would be vastly different. For some scenarios that require huge computation resource, SNE is unable to cover the calculation in data layer, then model layer is the only choice to carry it. For some scenarios that require less computation resource, SNE could handle it inside. Then, how to evaluate task requirements, and make optimal task allocation on both model layer and data layer, is the key technology in SOON.

b) **In model layer**, the key technology is the efficient feature extraction function of networks and services by using ML technology. With the rapid development of communication networks, the complexity of networks and services increases exponentially, and traditional analytical methods are hard to solve survivability, resource allocation, and other problems efficiently. Abstracted features represent essence of networks and services well with little interference information. Based on these features, policy layer is able to judge states easily, and make a better decision than traditional methods. ML technology, such as deep learning, is the representative technique that has been proved to be able to extract features from data in pattern recognition. However, there is no sufficient evidence to prove that it can also be applied for problems in optical networks seamless. For example, convolution operation in convolutional neural network (CNN) is efficient in image recognition [17], because image structure, i.e. Euclidean structure has property of translation invariance. But topology structure of optical networks couldn't

keep this property because it belongs to non-Euclidean structure [18]. So how to apply ML technology to extract features in optical networks is the key technology of model layer.

c) **In policy layer**, the objective of this layer is to improve service capacity by using network and services features extracted by MLE from model layer. As a transport pipe, the fundamental requirement of optical networks is to support great load capacity while being satisfied with service requirements. Both the original mass data from data layer, and the ML models from model layer, could help policy layer to perceive the whole optical networks in details from different perspectives. Based on these information, ML-based transmission engineering applications should implement rapid fault location, accurate fault prediction, and spectrum resource adjustment. And the ML-based network engineering applications should implement intelligent service allocation, service rerouting, and service distribution prediction.

4. Applications with SOON

Based on the powerful feature extraction ability driven by model layer, SOON is able to handle many problems of operation, administration, and maintenance. From the view of application fields, SOON could be used in transmission engineering and network engineering as mentioned above. However, the selection of ML algorithm in SOON architecture depends on the problem to be solved. From the view of selection of ML algorithm, applications are composed of auxiliary information provision (AIP) applications and decision-making suggestion (DMS) applications. In the AIP application, ML model is responsible to provide additional information hidden in complex context to the application in policy layer. While in the DMS application, ML model is able to output part of solution for suggestion, or even a complete solution to make decision directly. Tidal traffic prediction, alarm prediction, and anomaly action detection listed below belong to AIP applications that use the prediction or classification function of ML algorithms. Actor-critic-based resource allocation (ACRA) algorithm in Section 4.4 belongs to DMS application, which is proposed to allocate wavelength resource automatically with the help of routing algorithm.
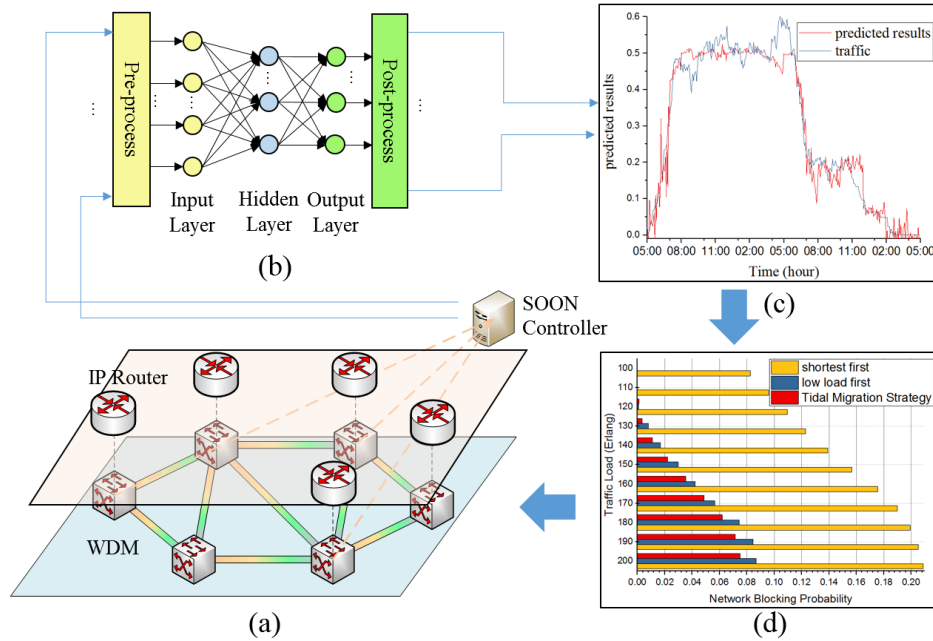
4.1 Tidal traffic prediction

Fig. 3. Procedure and result of traffic prediction, (a) IP-over-WDM scene, (b) ANN training, (c) traffic prediction, (d) traffic adjustment.

In metropolises, citizens regularly shuttle in residential and business areas every day [19]. On their way to work and off work, they also constantly connect to the network via their devices at different places. Driven by the concentrated period of residents' traveling, real-time traffic in metropolitan area network (MAN) appears regular fluctuation in spatial and time dimensions, named tidal phenomenon [20,21], which is analogy to the rise and fall of the sea levels. Dynamic changes of traffic bring huge load pressure for different network areas at different time. Figure 3(a) shows the overlay architecture of MAN, whose upper layer is composed of IP routers, and whose under layer is composed of wavelength division multiplexing (WDM) equipment. Network users access Internet via edge IP routers, which are connected to OTN equipment for traffic aggregation and transparent transmission. Hence, tidal phenomenon is more obvious in optical networks. In SOON, controller gathers the traffic historical data through SAP, reformat these data into a unified data set by following unified network model, and save them in network database.

If the traffic trends could be predicted in advance accurately via machine learning [22,23], it's helpful to relieve the load pressure in load-peak area, and utilize the network resource more efficiently. In SOON, the traffic prediction application in policy layer gets traffic data statistical characteristics from network database through NCC and network service module. After data analysis, this application decides to predict traffic distribution at time *t + 1* according to previous traffic distribution in *[t-n, t]*. To achieve this goal, this application divides traffic data set in network database into train subset and test subset, and invoke artificial neural network (ANN) algorithm implemented in MLE to train model by using the data set, as Fig. 3(b) shows. This ANN model uses back-propagation-based gradient descent strategy to update parameters, adjusts the threshold and the weights of connections according to the loss value across neurons in hidden layers. Loss value is calculated by squared hinge loss (L2 loss) that indicates the difference between training data and model output. After hundreds of epochs to traverse the whole train subset, loss value becomes less than 10^{-2}. Figure 3(c) shows the performance of trained model. The blue curve is the real traffic of test

data for 24 hours in one day, and the red curve is predicted traffic calculated by ANN model. These two curves denoted real traffic and prediction traffic respectively are very close, which means the ANN model can accurately predict the trends of dynamic changing traffic. Figure 3(d) shows the performance of traffic adjustment policy implemented in application according to tidal migration strategy that is implemented based on traffic prediction. The messages that control resource allocation from traffic prediction application is sent to NCC, and further translated to signalling in NCP. The benchmark algorithms are shortest-first algorithm and low-load-first algorithm, the former uses min-hop path calculation to calculate routing and first-fit strategy to allocate bandwidth, the latter uses min-metric path calculation and first-fit strategy. Compared with these two algorithms, tidal migration strategy algorithm uses predicted traffic distribution to judge future tidal-peak and tidal-valley areas, and adjust link metric respectively to change link priority for path calculation. In simulation, we use a MAN topology with 14-node and 22-link, where each link is equipped with 60 available wavelengths. There are 4 nodes located in business area, and the other 10 nodes located in residential area. In ANN model of traffic prediction, the input is composed of 45 neurons, and the output is composed of 15 neurons. The first 14 neurons are used to indicate specific edge. Each neuron represents a node in order, whose value is binary. 1 indicates that this edge connects to this node, and 0 has the opposite meaning. The 15th neuron is used to indicate current time point in one day. And the last 30 neurons are used to indicate the available wavelength number in the past 2 hours with 4 mins as a fixed period. The 15 neurons of output are predicted available wavelength number next hour. Figure 3(c) shows that performance of tidal migration strategy is obviously better than the benchmark algorithms, i.e. shortest first and low load first.

4.2 Alarm prediction

Failures of fiber and equipment in optical networks will cause network service interruption usually, which may result in huge economic loss. Failure accident is usually judged by root alarm reported by the optical equipment. Then, the prediction for network alarm would help telecom operators fix network problem and change parts to avoid service interruption in advance.

Figure 4 shows the demonstration of alarm prediction. In Fig. 4(a), a SOON controller manages a physical optical network through NCP. And each element is responsible to report real-time physical performance parameters to controller. There are two types of parameters roughly, standard-specific parameter and vendor-specific parameter. Standard-specific parameter contains the basic parameters used for almost all types of equipment by following specification/standard. For example, in ITU-G.783 [24] published by ITU-T, the frame of synchronous digital hierarchy (SDH) equipment should contains B1 byte for checking synchronous transport module level-1 (STM-1) errors of a regeneration section, B2 bytes for checking STM-1 errors of a multiplex section, B3 byte for checking virtual container-4 (VC-4) path errors, and so on. Vendor-specific parameter is defined privately by different vendors. 56901 alarms signals are collected on a large-scale commercial optical networks with 274 nodes and 487 links.
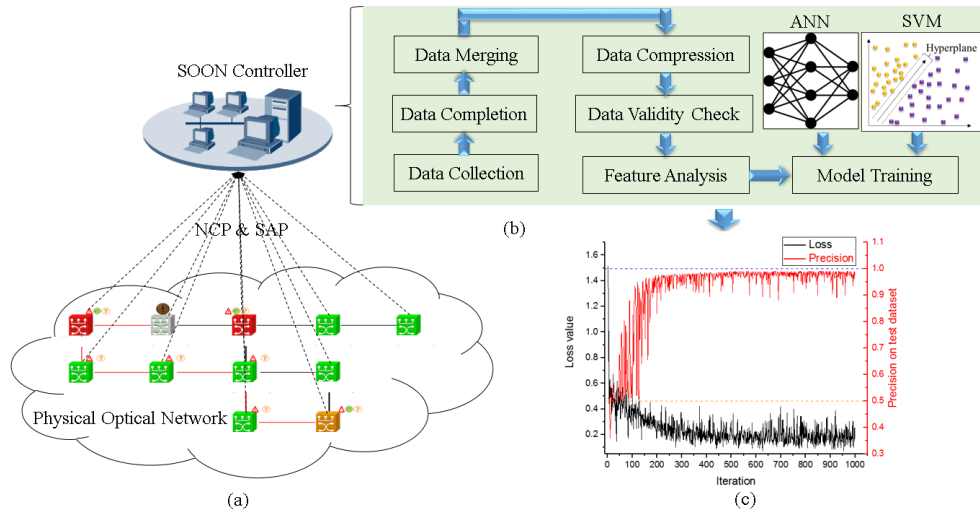
Fig. 4. Demonstration of alarm prediction in SOON, (a) alarm report, (b) ML model generation, (c) IN_PWR_LOW alarm prediction.

Figure 4(b) shows the data processing and model training for alarm-prediction ML model generation. Data pre-process is a fundamental and critical step for model training, especially in case that quantity of original data is quite dirty in completeness, consistency, entity identity, and other dimensions. Dirty data affects the accuracy of ML algorithm and even algorithm selection. In reality, the collected data of physical parameters are normally missing, inconsistent, and conflicting, which is hard to be utilized directly. Then, different from the pre-process in traffic prediction application, the process of alarm prediction application is composed of six steps, i.e., data collection, data completion, data merging, data compression, data validity check, and feature analysis, as Fig. 4(b) shows. Data-collection step gathers all historical data from the whole optical network as much as possible through SAP. Data-completion step completes lacking data according to the rules of performance log and valid data distribution. Data-merging step combines data with different types that has strong correlation with each other. Data-compression step reduce data size by removing redundant data. Data-validity-check step is designed to confirm that filtered data are credible enough to avoid misleading while training model. Finally, feature-analysis is the last step to test the relevance between input and output of data set to prevent invalid model training. After 6-step data processing, a useful data set is built, where input is performance data and output is the probability of alarm in future. We choose an appropriate ML algorithm from ANN, support vector machine (SVM), and other prediction algorithms, then execute model training as mentioned at Section 4.1.

Figure 4(c) shows the application for ML model where back-propagation ANN algorithm is used. After monitoring enough data, we use prediction model to predict when alarms may be triggered. Alarms are divided into warning alarm, minor alarm, major alarm and critical alarm according to the order of severity. Critical alarm is the most serious alarm that indicates failure occurs at very high rates. In the case mentioned in Fig. 4(c), the fault caused by power exception of optical interface is reported by IN_PWR_LOW critical alarm. And the related performance is optical input power. We extract a train data set with 800 records and a test data set with 200 records that includes the relationship between IN_PWR_LOW and optical input power. Input of each record is 12 values within 3 hours, and output is a binary value indicating whether this alarm happens in future 15 minutes. ANN algorithm is composed of three neural layers with rectified linear unit activation function, stochastic gradient descent optimizer, cross entropy loss function. The neural number of each layer is 12, 32, and 2. With

iteration number increases, loss value decreases, and model's predicting precision increases. The highest precision reaches 99.0% after about 400 iterations.

4.3 Anomaly action detection

For optical networks, anomaly network actions from tenants will cause wasting of network resource. The reason may be mis-operation from authorized user or malicious attack from ulterior intruder. Conventional anomaly action detection method is usually implemented based on fixed security rules. The intrusion rules are defined by network administrator to build a blacklist that can not adapt the dynamic network due to lacking of flexibility. Compared to the rule-based detection, data-analysis-based method is more intelligent and flexible because it allows controller to find hidden insights instead of fixed rules. Machine learning method can be applied to keep learning fuzzy rules to adapt dynamic network, which is the application in SOON.
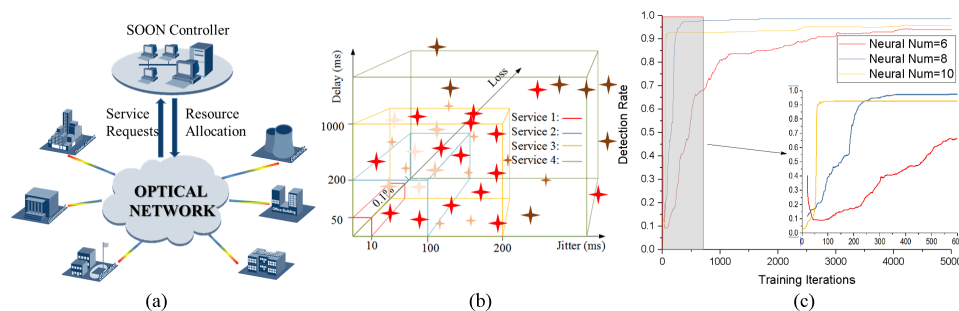

Fig. 5. Demonstration of anomaly action detection, (a) scene, (b) service requirements classification, (c) detection rate.

Figure 5(a) shows the scenario for anomaly action detection. In general, the users of optical networks may come from educational industry, government agency, electric power industry, trade centre, office building, enterprise, etc. SOON controller receives all connection requirements from these clients, then anomaly detection application detects if the requirement is legal and consistent with user behaviour according to detection model trained by MLE which mainly includes some legal performance parameter range. If so, detection application chooses to calculate resource allocation for the requirement, and execute it through NCC. If not, the application will report anomaly action to the administrator. In our simulation, bandwidth service could be divided into four types in requirement about delay, jitter and loss [25], as Fig. 5(b) shows. Besides, specific user's behaviour follows specific distribution on four service types usually. Based on this, network database records all historical requirements for all tenants, and anomaly detection application digs out potential user behaviour to build related ANN model with only one hidden layer. Then, SOON controller is able to detect abnormal operations. In the simulation, source and destination of bandwidth request are randomly selected on USNET topology. The data set contains 10000 instances for each type of request. Figure 5(c) shows three ML models of back-propagation neural network with different hidden neurons, where activation function in each neuron is rectified linear unit. The model with 6 neurons has the slowest performance improvement, and achieves the lowest detection rate finally. The model with 10 neurons has the fastest improvement, however, its final accuracy is lower slightly than the model with 8 neurons. The results of Fig. 5(c) indicates that the bigger model would reach the fastest improvement at the beginning, but may not reach the highest performance in the end.

4.4 Routing and wavelength assignment

Routing and wavelength assignment (RWA) is the key issue of resource allocation in WDM network, and has been studied by many heuristic algorithms [26–28]. However, most

heuristic algorithms could only reach approximately optimal performance under specific scenarios, which is limited by the way to make use of influence factors, and other potential influence factors that are not included in the algorithm. So, how to allocate resource in a proper way, and how to estimate if an allocation decision is optimal are too difficult to be integrated into current RWA algorithm. However, reinforcement learning (RL) [29] is a good choice to choose approximately optimal allocation, which could learn intelligently essential matching pattern under different conditions by exhausting possible conditions as much as possible. In reinforcement learning, there are only two parts, agent and environment. Agent observes the environment, and make decision to take an action according to its own policy. Influenced by the action, environment changes to new state, and would reward agent following reward and punishment mechanism (RPM). Then agent updates policy respectively, and take another action to environment again. Policy is the key part of reinforcement learning process, which is usually be implemented by machine learning algorithm.
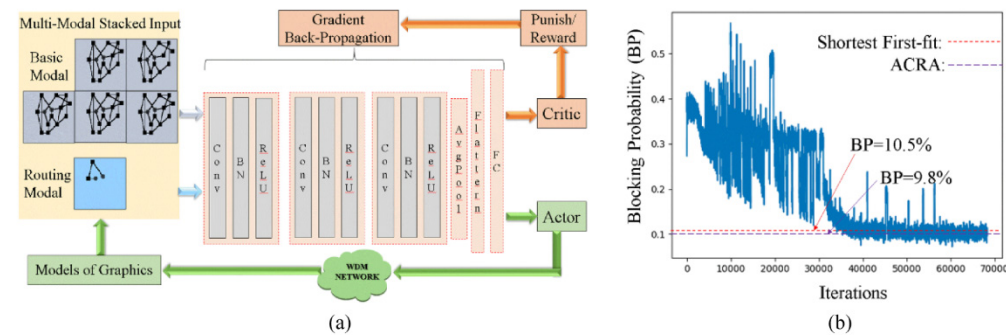

Fig. 6. Routing and wavelength assignment via ML, (a) multi-modal learning, (b) performance of multi-modal reinforcement learning model.

In this paper, SOON controller introduce actor-critic-based resource allocation (ACRA) algorithm to implement RWA automatically, where environment is the network status monitored by SAP and reformatted in network database. Agent is initialized and configured by RWA application, whose policy runs on MLE by calling RL algorithm. RPM is defined by RWA application, and policy update method is implemented in MLE. As the left part of Fig. 6(a) shows, we firstly introduce multi-modal learning to convert network status into graph form. Multi-modal learning is a ML method that use data sets with different modalities. For example, the wavelength allocation condition represents network status on bandwidth, and rouging represents a path from source to destination in network. These two information contain different content, and normally has different data format and data distribution. However, if we consider them together, we could get much more information. In RWA problem, the network wavelength allocation condition is called basic modal, and the routing for specific service request is called routing modal. Based on these two modal, simple RWA solution could be calculated.

We introduce a state-of-art reinforcement learning algorithm as processing architecture, named advantage actor-critic (A2C), which is the synchronous version of asynchronous advantage actor-critic algorithm (A3C) [30]. In SOON, network database converts network status and service requirement into multi-modal graph forms. Then graphs are put into a convolutional neural network (CNN) [31] designed in MLE, which is the most important innovation in machine learning in recent years. As Fig. 6(a) shows, this CNN is composed of three stacked modules, each of them contains convolution layer, batch normalization layer [32], and rectified linear unit (ReLU) layer [33].Then the output features are compressed through an average pooling (AvgPool) [34] layer, and flattened to one-dimension vector. The last layer for our CNN is a fully-connected neural network (FC). After that, the output would be handled by actor and critic. Actor is responsible to provide a recommended RWA solution

for optical networks. Critic is responsible to provide punishment and reward for taken action, and promote CNN's parameter updating to improve performance.

Figure 6(b) shows the blocking probability (BP) of network with different updating iterations under ACRA algorithm. The performance becomes higher with the growth of training iterations on the whole. The red line indicates the average blocking probability of benchmark algorithm, and the purple line indicates the final performance of ACRA. ACRA could reach about 6% improvement compared with benchmark.

5. Open issues with SOON

5.1 Computation capability requirement

With the development of mobile internet and Internet of Things (IoT), there has been billions of smart terminals all over the world. Edge computing is an emerging technology that proposes huge potential computing and storage resources supported by smart terminals, which is exactly able to meet SOON's requirements of computing and storage. However, there are two key issues on how to combine edge computing and SOON. The one is how to coordinate distributed resources of smart terminals and centralized resources of computing cloud (e.g., controller cluster), and the other one is how to accelerate computing speed while holding the performance of ML models on limited terminal resources. The former problem has become the hotspot on edge computing research, but the latter problem has not been studied in optical networks yet. In image recognition field, there are some research findings about applying deep learning algorithm to smartphone recently, which may inspire the acceleration research of SOON on terminals.

5.2 Response latency

The time cost of ML algorithm could be divided into two parts: offline training and online applying. Offline training is model generation or optimization, which means the process to train or iterate a ML model. The time scope of offline training ranges from several minutes to half a month according to selected ML algorithm, hyper-parameter initialization, and other factors. And time cost of online applying is the time when using specific model for specific application, which always ranges from a few hundred microseconds to a few seconds. In reality, it's a challenge to control time cost of offline training and online applying to reduce model response latency.

5.3 Model selection

The significant problem is to choose an optimal model while applying a model to specific application. There are many ML algorithms in MLE, many models for specific ML algorithm, even many models for specific application. Under different scenarios that require different model accuracy and response latency, it's important to select an optimal model to apply. Besides, it's also critical to confirm that present models is able to meet various requirements. Otherwise, we should consider to iterate current models to make them better, or train a model with different hyper-parameters over again.

6. Conclusion

A ML-based optical network architecture named SOON is described. SOON is inherited from SDON, but has different design objectives. In data layer, we replace foolish network element of SDON by SNE for the first time. SNE could adjust itself to improve transmission performance. Model layer is the brain of SOON. Apart from functions of SDON's controller, it is also able to process big data of network and services, extract essential features. Policy layer is the platform that carries intelligent applications. Applications could implement automatic adjustment on spectrum allocation, failure prediction, and other aspects of OAM by calling features from control plane.

References

1. D. N. L. Levy, "Computer games I," New York, NY, USA (Springer, 1988).
2. A. K. I. S, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," International Conference on Neural Information Processing Systems, Lake Tahoe, Nevada, USA, Dec. 2012.
3. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proceedings of the IEEE conference on computer vision and pattern recognition, Boston, Massachusetts, USA, June 2015.
4. C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17), San Francisco, California, USA, Feb. 2017.
5. D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of Go with deep neural networks and tree search," Nature **529**(7587), 484–489 (2016).
6. D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. Hassabis, "Mastering the game of Go without human knowledge," Nature **550**(7676), 354–359 (2017).
7. F. N. Khan, C. Lu, and A. P. T. Lau, "Machine learning methods for optical communication systems," Advanced Photonics 2017, New Orleans, USA, July 2017.
8. Z. Wang, M. Zhang, D. Wang, C. Song, M. Liu, J. Li, L. Lou, and Z. Liu, "Failure prediction using machine learning and time series in optical network," Opt. Express **25**(16), 18553–18565 (2017).
9. C. Daly, D. L. Moore, and R. J. Haddad, "Nonlinear auto-regressive neural network model for forecasting Hi-Def H.265 video traffic over Ethernet passive optical networks," SoutheastCon, Charlotte, NC, USA, March 2017.
10. H. Zhang, Y. Wang, H. Chen, Y. Zhao, and J. Zhang, "Exploring machine-learning-based control plane intrusion detection techniques in software defined optical networks," Opt. Fiber Technol. **39**, 37–42 (2017).
11. Y. Zhao, R. He, H. Chen, J. Zhang, Y. Ji, H. Zheng, Y. Lin, and X. Wang, "Experimental performance evaluation of software defined networking (SDN) based data communication networks for large scale flexi-grid optical networks," Opt. Express **22**(8), 9538–9547 (2014).
12. ITU-T Rec. M.3400, "TMN management functions," Feb. 2002.
13. ITU-T Rec. G.8080/Y.1304, "Architecture for the automatically switched optical network," Feb. 2012.
14. A. Farrel, J.-P. Vasseur, and J. Ash, "A path computation element (PCE)-based architecture," IETF RFC4655, 2006.
15. Y. Ji, J. Zhang, and Y. Zhao, "Development prospects of software defined optical networks," Telecommunications Science **30**(8), 19–22 (2014).
16. B. Yan, Y. Zhao, W. Wang, X. Yu, and J. Zhang, "An equipment parameter control architecture," *https://datatracker.ietf.org/doc/draft-epc-architecture/*.
17. F. Altenberger, and C. Lenz, "A non-technical survey on deep convolutional neural network architectures," arXiv, 1803.02129 (2018).
18. M. Brinstein, J. Bruna, X. Bresson, and Y. LeCun, "Geometric deep learning on graphs and manifolds," Proceedings of Conference on Neural Information Processing Systems, Los Angeles, California, USA, Sept. 2017.
19. C. Song, Z. Qu, N. Blumm, and A. L. Barabási, "Limits of predictability in human mobility," Science **327**(5968), 1018–1021 (2010).
20. S. Troia, G. Sheng, R. Alvizu, G. A. Maier, and A. Pattavina, "Identification of tidal-traffic patterns in metro-area mobile networks via matrix factorization based model," IEEE International Conference on Pervasive Computing and Communications, Kona, USA, May 2017.
21. Z. Zhong, N. Hua, H. Li, Y. Li, and X. Zheng, "Considerations of effective tidal traffic dispatching in software-defined metro IP over optical networks," Opto-Electronics and Communications Conference (OECC), Shanghai, China, June 2015.
22. M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," Science **349**(6245), 255–260 (2015).
23. R. Li, Z. Zhao, J. Zheng, C. Mei, Y. Cai, and H. Zhang, "The learning and prediction of application-level traffic data in cellular networks," IEEE Trans. Wirel. Commun. **16**(6), 3899–3912 (2017).
24. ITU-G.783, "Characteristics of synchronous digital hierarchy (SDH) equipment functional blocks," Sep. 2009.
25. Huawei QoS Technology White Paper, 2013.
26. S. Azodolmolky, M. Klinkowski, E. Marin, D. Careglio, J. S. Pareta, and I. Tomkos, "A survey on physical layer impairments aware routing and wavelength assignment algorithms in optical networks," Comput. Netw. **53**(7), 926–944 (2009).

27. N. Charbonneau and V. Vokkarane, "A survey of advance reservation routing and wavelength assignment in wavelength-routed WDM networks," IEEE Comm. Surv. and Tutor. **14**(4), 1037–1064 (2012).
28. A. Ding and G. Poo, "A survey of optical multicast over WDM networks," Comput. Commun. **26**(2), 193–200 (2003).
29. R. S. Sutton and A. G. Barto, "Reinforcement learning: an Introduction," Cambridge, MA, USA (MIT, 2011).
30. V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Harley, T. P. Lillicrap, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 2016.
31. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," NIPS Conference Proceedings, Lake Tahoe, Nevada, USA, December 2012.
32. S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," Proceedings of the 32 nd International Conference on Machine Learning, Lille, France, 2015.
33. V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," Proceedings of the 27th international conference on machine learning (ICML-10), Haifa, Israel, June 2010.
34. G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," arXiv preprint arXiv: 1207.0580 (2012).