

Highly-Efficient and Automatic Spectrum Inspection based on AutoEncoder and Semi-Supervised Learning for Anomaly Detection in EONs

Siqi Liu, Jiawei Kong, Xiaoqin Pan, Deyun Li, and Zuqing Zhu, *Senior Member, IEEE*

Abstract—To closely monitor the performance of lightpaths in an elastic optical network (EON), people need to rely on real-time and fine-grained spectrum monitoring. This, however, will generate tremendous telemetry data, which can put great pressure on both the control and data planes. In this work, we design and experimentally demonstrate AutoSpeck, which is a DL-assisted network automation (DLaNA) system that can realize highly-efficient and automatic spectrum inspection for anomaly detection in EONs. Specifically, we architect AutoSpeck based on the software-defined EON (SD-EON) architecture, and propose techniques to greatly reduce the loads of data reporting (in the data plane) and data analyzing (in the control plane). To reduce the loads of data reporting, we leverage the AutoEncoder (AE) technique to design a spectrum data compression method. To improve the efficiency of data analytics, we first design a coarse filtering module (CFM) to let the control plane filter out most of the normal data before invoking the DL-based anomaly detection. Then, to address the difficulty of labeling massive spectrum data, we develop a DL-based anomaly detection based on semi-supervised learning. Our experimental demonstrations consider two representative intra-channel anomalies (*i.e.*, the filter drifting and in-band jamming), and the results confirm that AutoSpeck can achieve highly-efficient and automatic spectrum inspection for anomaly detection in EONs.

Index Terms—Software-defined networking (SDN), Elastic optical networks (EONs), Network automation, Anomaly detection, Deep learning (DL), AutoEncoder (AE), Spectrum inspection.

I. INTRODUCTION

NOWADAYS, the rapid deployment of cloud infrastructures and fast emergence of bandwidth-hungry applications (*e.g.*, Big Data analytics and live streaming) have driven network traffic to skyrocket in volume and become much more dynamic [1–4]. Hence, flexible-grid elastic optical networks (EONs) have recently attracted intensive interests and have been considered as a promising technology for the optical layer [5–7]. This is because EONs get rid of the restrictions due to fixed spectral grids, and enable adaptive spectrum allocation at a granularity of 12.5 GHz or even narrower [8–10].

However, the rising of EONs also brings challenges to network control and management (NC&M), especially for anomaly detection in the optical layer [11–13]. Specifically, the challenges are introduced due to two reasons. Firstly, the lightpaths in EONs use various spectral widths and more

modulation formats, while the significant increase of combinations makes it much more difficult to detect anomalies. For instance, the anomaly detection based on comparing lightpaths’ spectra to a standard spectral shape [14] will be inapplicable. Secondly, as they use sophisticated modulation formats, such as quadrature phase shift keying (QPSK) and 16-quadrature amplitude modulation (16-QAM), the lightpaths in EONs are more prone to be affected by quality-of-transmission (QoT) degradations. This means that the anomaly detection in EONs has to be more sensitive and precise than its antecessors.

The anomalies concerned by an NC&M system can be categorized into hard and soft failures. Hard failures refer to those that will cause severe performance degradations or even disconnect lightpaths completely, and they can be detected with traditional optical performance monitoring techniques [15]. On the other hand, soft failures are much less harmful than the hard ones, which makes them hard to detect, and they would still cause lightpath disfunction if having been ignored for a long time [14]. Note that, in addition to the reactive technique to detect soft failures timely and precisely, their impacts can also be mitigated proactively by estimating and reserving sufficient QoT margins [16]. Although people have already proven the effectiveness of the proactive schemes, soft failure detection is still necessary and important for EONs. This is because the highly dynamic traffic in today’s Internet makes reconfigurations common in the optical layer [17], and the QoT of lightpaths can become unstable due to frequent reconfigurations [18]. Hence, we have to face the dilemma that if reserving excessive margins, network resources will be underutilized during normal operation, but reducing the margins will lead to higher failure probabilities during reconfigurations.

One can hardly realize timely and precise anomaly detection without real-time and fine-grained network monitoring [13], which has recently made promising advances due to the introduction of network telemetry [19, 20] and disaggregation of optical networking systems [21]. Specifically, we can deploy many optical performance monitors (OPMs) in a disaggregated optical networking system, and then leverage network telemetry techniques to convey the collected status data to the NC&M system for deep learning (DL) based data analytics [22]. Note that, many OPMs are essentially simplified optical spectrum analyzers, due to the reason that most of the key performance metrics of a lightpath (*e.g.*, power level, optical signal-to-noise ratio (OSNR), and channel width) can be obtained through spectrum analysis [11]. Meanwhile, the current commercially-available OPMs can quickly scan the whole C-band using a

S. Liu, J. Kong, X. Pan, D. Li, and Z. Zhu are with the School of Information Science and Technology, University of Science and Technology of China, Hefei, Anhui 230027, P. R. China (email: zqzhu@ieee.org).

X. Pan is also with the Engineering Technology Center, Southwest University of Science and Technology, Mianyang, Sichuan 621010, P. R. China

Manuscript received on July 3, 2020.

fine resolution of 312.5 MHz within one or two seconds [23].

Nevertheless, the aforementioned real-time and fine-grained spectrum monitoring will generate tremendous status data, which will flood the control channels between the NC&M system and an EON, and put great pressure on the DL-based data analytics in it. To the best of our knowledge, how to improve the efficiency of spectrum inspection for anomaly detection in EONs still has not been fully explored. Hence, in this work, we design and experimentally demonstrate AutoSpecheck, which is a DL-assisted network automation (DLaNA) system that can realize highly-efficient and automatic spectrum inspection for anomaly detection in EONs. Specifically, we architect AutoSpecheck based on the software-defined EON (SD-EON) architecture [24, 25], propose new techniques to greatly reduce the loads of data reporting (in the data plane) and data analyzing (in the control plane), and implement our proposals in a real SD-EON testbed for experimental demonstrations.

Compared with the existing schemes for DL-based anomaly detection in EONs, our innovations and contributions can be summarized as follows. Firstly, we leverage the AutoEncoder (AE) technique [26] to design a spectrum data compression method, which not only significantly reduces the load of data reporting from the data plane, but also ensures that the correlations buried in spectrum data will be kept through the compression. Secondly, we design a coarse filtering module (CFM) for the control plane to filter out most of the normal data before invoking the DL-based spectrum inspection for anomaly detection. Note that, during network operation, the QoT of a lightpath stays normal in most of the time [27], and thus the CFM can greatly relieve the load of data analyzing.

Thirdly, to address the difficulty of labeling massive spectrum data, we develop the DL-based anomaly detection based on semi-supervised learning. Specifically, we first propose an unsupervised learning model based on clustering to detect anomalies in and label massive data automatically by utilizing only a small amount of labeled data, and then design a supervised learning model that can leverage the labeled data to train its deep neural network (DNN) for accurate anomaly detection. Fourthly, we implement our proposals in an SD-EON testbed to consider two representative intra-channel anomalies (*i.e.*, filter drifting and in-band jamming [28]). Finally, we conduct experiments to verify that AutoSpecheck can achieve highly-efficient and automatic spectrum inspection for anomaly detection in EONs, and close the loop of “observe-analyze-act” for the DLaNA to 1) detect and locate anomalies timely and accurately and 2) implement recovery plans quickly.

The rest of paper is organized as follows. Section II briefly surveys the related work. We present the overall design of AutoSpecheck in Section III. Sections IV and V elaborate on our proposals for the data and control planes, respectively, to realize highly-efficient and automatic spectrum inspection. Then, the numerical results regarding AutoSpecheck are discussed in Section VI, and we show its experimental demonstrations in Section VII. Finally, Section VIII summarizes the paper.

II. RELATED WORK

Recently, the promising advances on DL have promoted the idea of integrating DL models in NC&M systems to real-

ize the so-called knowledge-defined networking (KDN) [29]. Following this trend, people have proposed various DLaNA systems for optical networks, to facilitate QoT estimation [30], anomaly detection [27, 31, 32], physical-layer security monitoring [28, 33], network function virtualization [34, 35], datacenter management [36, 37], *etc.* For comprehensive information about the applications of DL in optical networks, one is suggested to check the surveys and tutorials in [38–40].

Since each fiber link can carry a tremendous volume of data traffic, survivability and availability are always important topics in the design and operation of EONs. Therefore, the previous studies in [41–43] have considered how to allocate backup resources to protect lightpaths against hard failures such as fiber cuts. However, they did not address how to detect and locate anomalies, especially when they are soft failures. In the proactive manner, the impacts of soft failures can be relieved or even avoided by estimating and reserving enough QoT margins [16]. Hence, people have designed different machine learning models to achieve precise QoT estimation for lightpaths [44–46]. Although it can compensate for future QoT degradations to ensure the performance of lightpaths, the proactive scheme also results in under-utilization of spectrum resources in EONs during normal operation. Hence, timely and accurate anomaly detection and corresponding service recovery are vital for realizing low-margin operation [47].

For anomaly detection, the authors of [48] took the bit-error-rate (BER) of lightpaths as the input to detect anomalies related to optical filtering. Nevertheless, BER measurement or estimation requires complicated implementations related to the time-domain (*e.g.*, the coherent detection and digital signal processing (DSP) [11]), which are expensive and only available at the receiver end. Therefore, the anomaly detection might have difficulty to precisely locate where the anomalies happen on lightpaths. Moreover, it is known that BER might not always be available in EONs, since certain cross-domain lightpaths may end in other domains [49]. Regarding the anomaly detection in filterless optical networks (FONs), Shariati *et al.* [50–52] proposed several interesting optical signal tracking approaches to detect small laser drifts. There are two major differences between these studies and our work. Firstly, as laser drifting can only happen at the transmitter of a lightpath, the studies in [50–52] only needed to address anomaly detection, while we consider anomaly detection and location. Secondly, because the studies only focused on tracking the central frequency of each optical transmitter accurately, they did not need to use supervised learning. However, in our case, we need to address two different types of anomalies, and thus the mapping between spectrum features and anomalies is not that straightforward. Therefore, we have to leverage semi-supervised learning for the anomaly detection and location.

On the other hand, spectrum analysis can be realized at any point on a lightpath, does not need sophisticated time-domain processing, and is much cheaper. Hence, it would be more promising to detect anomalies with spectrum inspection [53]. However, as the spectrum inspection needs to operate on tremendous spectrum data from real-time and fine-grained monitoring, the loads of data reporting and analyzing will cause severe scalability issues. The studies in [54, 55] de-

veloped a few machine learning based approaches to analyze optical spectrum for soft-failure detection and location (with the focus on filter drifting and filter tightening). Although these studies considered a similar problem as ours, we make new contributions over them and thus the techniques proposed in this work can be used to improve the efficiency of the soft-failure detection and location in [54, 55]. Specifically, as they leveraged the classification based on support vector machine (SVM) or decision tree (DT) for detecting and locating soft-failures, our CFM can also help their systems filter out most of the normal data in advance and thus reduce the load of data analyzing in the classification, and our GMM-based data clustering and labeling can be leveraged to avoid the hassle of labeling data manually too. Note that, the proposals in [54, 55] were based on the node-based approach in the data plane. In other words, the soft-failure detection is mainly accomplished by local OPMs. The benefits of this approach are: 1) decisions can be made locally and timely, and 2) the need of sending spectrum data to the control plane can be minimized. However, the involvement of the control plane is still needed, and thus the authors designed algorithms for the control plane to obtain a global view for anomaly detection and location.

Hence, we consider the controller-based approach in this work. Nevertheless, for both the node-based approaches in [54, 55] and our controller-based one, there are pros and cons (*i.e.*, there does not exist a universal winner). Therefore, we should select the right one to implement according to its actual characteristics of an optical network. In [56, 57], the authors discussed the learning life cycle to speed up the network automation in optical networks, while the techniques used for anomaly detection and location were based on the designs in [54, 55]. Hence, the studies also assumed that the anomaly detection and location are based on supervised learning with the node-based approach. Note that, with supervised learning, the classification models have to be trained with massive labeled data that tells the ground-truth about anomalies before they can be put into operation, and thus data labeling will be an issue. Finally, we would like to mention that our AE-based spectrum data compression can also be leveraged to improve the efficiency of the node-based systems proposed in [50–52, 54–57], because it can reduce the storage used to record the standard spectra and/or correction masks in local OPMs.

Previously, the study in [27] considered how to detect anomalies with unlabeled data, to avoid the hassle of labeling massive data. Nevertheless, the proposed algorithm could only differentiate the normal and abnormal data points, but could not classify the anomalies or locate them in the data plan. On the other hand, Christodoulopoulos *et al.* [58] investigated the method to realize accurate anomaly detection and location with as few OPMs as possible. However, reducing the number of OPMs can only partially address the scalability issues, and cannot resolve the difficulty on data labeling. To the best of our knowledge, highly-efficient spectrum inspection for anomaly detection in EONs is still under-explored.

III. OVERALL SYSTEM DESIGN

Fig. 1 shows the overall system design of AutoSpecheck, which includes new functional modules in both the data and

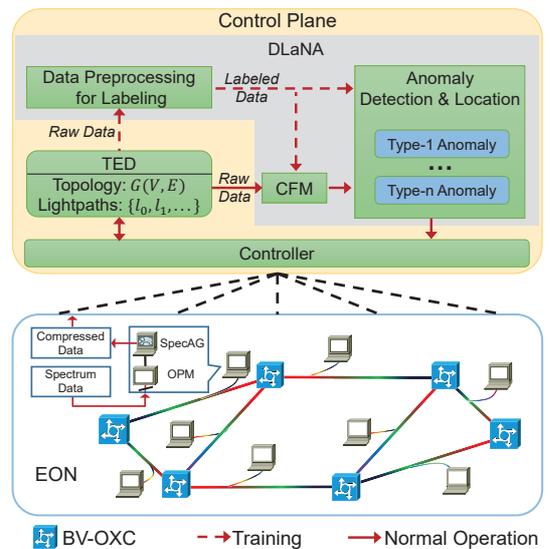


Fig. 1. Network architecture and system design of AutoSpecheck.

control planes of an SD-EON. The data plane of the SD-EON consists of bandwidth-variable optical cross-connects (BV-OXCs) interconnected by optical fibers. By leveraging built-in bandwidth-variable wavelength selective switches (BV-WSS'), each BV-OXC de-multiplexes lightpaths from input fibers, terminates those that mark the BV-OXC as their destinations, and optically grooms the remaining ones with the locally-generated lightpaths to send to the corresponding output fibers. For network monitoring, we place at least one OPM on each fiber link, which taps a small portion of optical power from the fiber and performs spectrum analysis on the signal.

A. Major Components

In our AutoSpecheck, the OPM is a commercial product that can scan the whole spectrum of C-band with a fine resolution of 312.5 MHz in two seconds [23]. Hence, the spectrum of each lightpath can be represented with a series of power levels at the spectral points in its channel, which can be modeled as $P_{l,m} = \{p_{l,m}^{(1)}, \dots, p_{l,m}^{(n)}\}$, where l and m are the indices of the lightpath and the OPM, respectively, and n is the number of spectral points collected for the lightpath. To realize real-time monitoring, OPMs in the data plane need to report all the spectrum data (*i.e.*, $\{P_{l,m}, \forall l, m\}$) to the control plane frequently, which will generate a large volume of traffic in the control channels. Hence, as shown in Fig. 1, we design a spectrum data agent (SpecAG) to compress the data received from each OPM with an AutoEncoder (AE) based technique [26]. The SpecAG transforms each $P_{l,m}$ to $\hat{P}_{l,m}$, which is also a vector but containing much fewer elements (*i.e.*, from 160 elements to 6 or less), and it ensures that the correlations buried in $P_{l,m}$ will be kept through the compression. We will explain the details about SpecAG in Section IV.

In the control plane, the SDN controller is developed based on the well-known ONOS platform [59], and it communicates with the network elements in the data plane with the OpenFlow protocol. The controller collects compressed spectrum data from the data plane, stores the data in the traffic engineering

database (TED), leverages the DLaNA module for timely and accurate anomaly detection, calculates the recovery plan when encountering anomalies, and reconfigures the data plane accordingly. In the TED, the data plane is modeled as a graph $G(V, E)$, where V and E are the sets of BV-OXCs and fiber links, respectively, while the information about each lightpath l is recorded as a tuple $\langle R_l, \{\hat{P}_{l,m}, \forall m \in R_l\} \rangle$, where R_l denotes the set of fiber links on its routing path, and each OPM m corresponds to a fiber link. Then, the set of lightpaths is denoted as $L = \{l_0, l_1, \dots\}$, and the TED organizes the information regarding each lightpath as an entry.

The DLaNA in the control plane takes lightpaths' spectrum data from the TED for anomaly detection and location. Note that, to precisely detect and classify anomalies, the DLaNA needs to be trained with labeled data, but labeling massive spectrum data manually will be time-consuming and inconvenient. Therefore, we design the data preprocessing module for data labeling, which leverages unsupervised learning together with only a small amount of labeled data to detect anomalies in and label massive spectrum data automatically. Then, the labeled data is used to train the coarse filtering module (CFM), such that it can filter out most of the normal data before invoking the DL-based spectrum inspection, *i.e.*, reducing the load of data analyzing significantly.

The anomaly detection and location module also gets trained with the labeled data, to detect and classify different types of anomalies based on the compressed spectrum data. Next, during network operation, the anomaly detection and location module return the information about anomalies to the controller, which comes up with proper recovery plans. This work considers two representative intra-channel anomalies, *i.e.*, filter drifting and in-band jamming. Our DLaNA does not assume a fixed channel width to adapt to the lightpaths in EONs, and its detailed design will be discussed in Section V.

B. Data Collection and Preparation

In this work, we collect spectrum data from a real EON system and apply our proposal on the data to demonstrate its practicalness. Fig. 2 shows the experimental setup, where the lightpaths are generated and received by the bandwidth variable transponders (BV-Ts) in our Juniper optical packet platform (BTI-7800). With three pairs of BV-Ts, we can set up three lightpaths, each of which uses a channel width of 50 GHz to achieve the data-rate of 100 Gbps with QPSK modulation (*i.e.*, 31.2 Gbaud with forward-error correction (FEC)). In the experiments, we configure the 50-GHz channels of the three lightpath as adjacent, only concern the lightpath that locates in the middle of the three, and use the remaining two as possible background lightpaths. Then, we can obtain four lightpath scenarios by configuring the BV-Ts (*i.e.*, only the concerned lightpath is on, the concerned lightpath and a background one are on, and all the three lightpaths are on).

Next, we consider fiber transmission and optical switching. As shown in Fig. 2, there is a fiber link with inline erbium-doped fiber amplifier (EDFA) before the receiver, while before this final fiber link, we can insert [2, 10] stages of fiber transmission and optical switching. Here, the fiber transmission

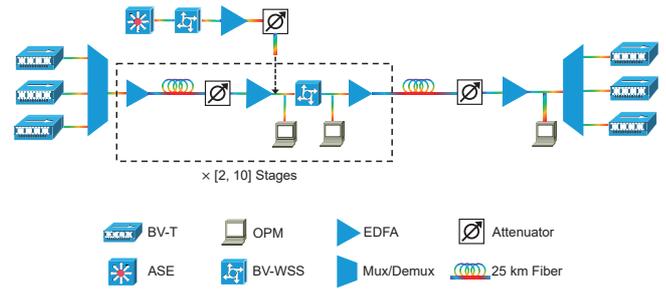


Fig. 2. Experimental setup for spectrum data collection.

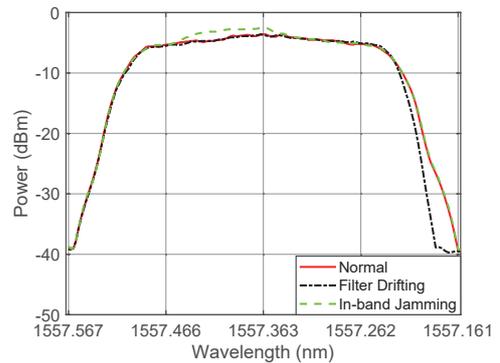


Fig. 3. Spectra of normal and abnormal lightpaths.

is also realized with a fiber link with inline EDFA, and the optical switching is realized by a BV-WSS. Therefore, we can get nine transmission/switching scenarios. Meanwhile, as we consider two types of anomalies (*i.e.*, filter drifting and in-band jamming) and there are 8 severities of filter drifting, we totally have 10 anomaly scenarios including the normal cases. Here, filter drifting is emulated by changing the central frequency of the BV-WSS in a stage, while in-band jamming is realized by using a BV-WSS to make the noise from an amplified spontaneous emission (ASE) noise generator span the same channel of the concerned lightpath, and injecting it in one of the stages with a fiber coupler. As we only address soft failures in this work, we make sure that the anomalies on the concerned lightpath only cause minor BER degradations at its receiver, but will not disconnect its data transmission. Fig. 3 gives several examples on the spectra of lightpaths. It can be seen that the spectrum of a lightpath with in-band jamming will not be significantly different from that without.

At this moment, we have four lightpath scenarios, nine transmission/switching scenarios, and ten anomaly scenarios, which totally result in 360 combinations. For each combination, we can further change its setup (*e.g.*, changing the location of anomaly insertion, slightly adjusting the magnitude of the concerned anomaly, and varying the attenuations before inline EDFAs) to obtain 15 lightpath states. Finally, we can use the setup in Fig. 2 to get 5,400 types of lightpaths.

Meanwhile, we have to explain that not all the data used in this work is purely from experiments, and we do leverage simulations to obtain certain part of it. This is because we can only use a BV-WSS to emulate a filter drifting of 12.5 GHz, and the drifting can be toward both directions, while

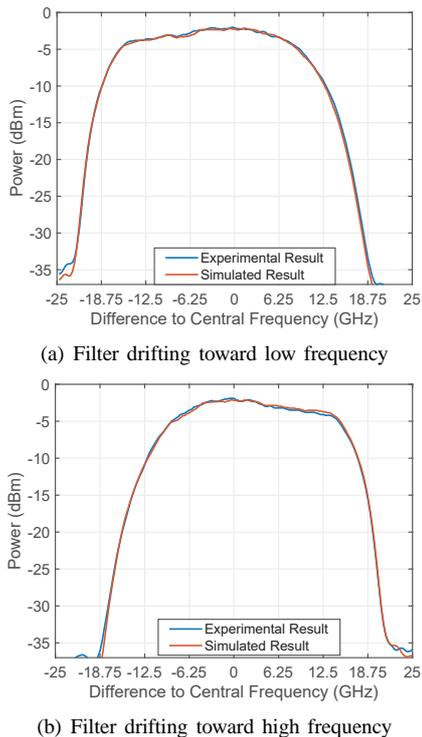


Fig. 4. Comparison of measured and simulated spectra of a lightpath going through a BV-WSS with 12.5 GHz filter drifting.

a smaller filter drifting is not feasible. Hence, in order to generate the filter drifting that is smaller than 12.5 GHz, we measure the transfer function of the BV-WSS on a 50-GHz channel, change the central frequency of the transfer function with a numerical method to get a new one that can induce a desired filter drifting, and apply the obtained transfer function to the spectra collected in experiments to emulate the smaller filter drifting. Note that, the numerical method will not make our data collection inaccurate or impractical, because it is still based on real-world measurements. To prove this, we first conduct an experiment to measure the spectrum of a lightpath that goes through a BV-WSS with 12.5 GHz drifting, and then use the method mentioned above to simulate the spectrum. The results are shown in Fig. 4, which indicates that the measured and simulated spectra do not have noticeable differences.

IV. AE-BASED SPECTRUM DATA COMPRESSION IN DATA PLANE

The AE-based data compression in an OPM needs to transform spectrum vector $P_{l,m}$ to a vector $\hat{P}_{l,m}$ that contains fewer elements. Note that, we concern more about the shape of the spectrum in anomaly detection, and thus each OPM normalizes the power levels in $P_{l,m}$ according to the peak power of each lightpath. As shown in Fig. 5, the data compression module follows the operation principle of AE [26] to include an encoder and an decoder. Here, the encoder compresses spectrum data, while the decoder is included to assist the training of the encoder and guarantee that useful information will not be lost through the compression.

In Fig. 5, the encoder first uses a convolutional neural network (CNN) after the input layer. This is because the

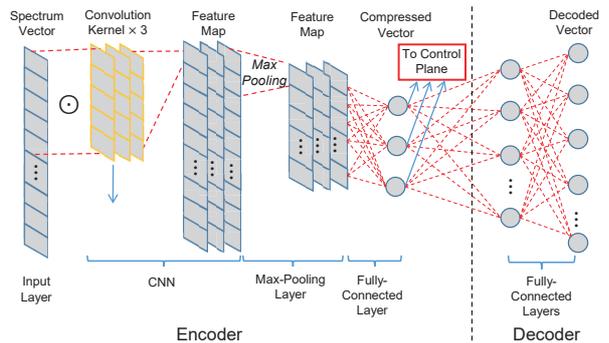


Fig. 5. AE-based data compression module.

CNN can accurately detect the presence of specific features in spectrum vectors (e.g., the shapes of rising and falling edges and the ripples in the pass-band), with the convolution kernels that will be optimized in training. The CNN can extract important features in the spectrum data, but the size of its output is the same as that of the input spectrum vector. Hence, we insert a max-pooling layer after it to compress the extracted features, whose output will be further compressed with a fully-connected layer to minimize the number of elements. The output of the fully-connected layer is just the compressed spectrum vector $\hat{P}_{l,m}$, which will be sent to the control plan for data analytics. Meanwhile, $\hat{P}_{l,m}$ also gets fed into the decoder to evaluate the performance of the encoder, and the decoder leverages two fully-connected layers to reconstruct the original spectrum vector $P_{l,m}$ based on $\hat{P}_{l,m}$.

In the training, we input spectrum vectors to the encoder to get compressed vectors, use the decoder to reconstruct the original vectors, and then update the parameters of the encoder and decoder according to the mean squared error (MSE) between the original and reconstructed vectors.

$$\text{MSE}(P_{l,m}, P'_{l,m}) = \frac{1}{n} \cdot \sum_{i=1}^n (p_{l,m}^{(i)} - p'_{l,m}^{(i)})^2, \quad (1)$$

where $P'_{l,m} = \{p_{l,m}^{(1)}, \dots, p_{l,m}^{(n)}\}$ is the reconstructed spectrum vector from the decoder. To show the effectiveness of our AE-based data compression, we train and test one for lightpaths with a channel width of 50 GHz as follows¹. For the 5,400 lightpaths considered in Section III-B, we use OPMs to measure the optical spectra at the inputs/outputs of fiber links along their routing paths, and get 54,000 spectrum vectors, among which the vectors for normal and abnormal lightpaths are equal. Hence, each vector $P_{l,m}$ represents a lightpath's spectrum at a certain location, and since the lightpath's channel width is 50 GHz and an OPM scans with a resolution of 312.5 MHz, $P_{l,m}$ is a vector with 160 elements. Next, in the 54,000 spectrum vectors, we randomly select 80% to put in the training set, and use the remaining ones as the testing set.

We set the length of the compressed spectrum vector $\hat{P}_{l,m}$ as $\{2, 3, 6\}$, and Table I shows the results on the MSE in Eq. (1)

¹Note that, our AE-based data compression can also operate on the spectrum data of lightpaths whose channel widths are various, as long as each of their spectrum vector has the same length. This can be achieved by including down-/up-sampling after spectrum scanning in each OPM.

TABLE I
RESULTS OF AE-BASED DATA COMPRESSION

Output Lengths	6	3	2
Data Compression Ratio	26.67	53.33	80
Average MSE ($\times 10^{-2}$)	3.103	3.105	4.268
Training Time (sec)	1747.74	2119.41	3138.69

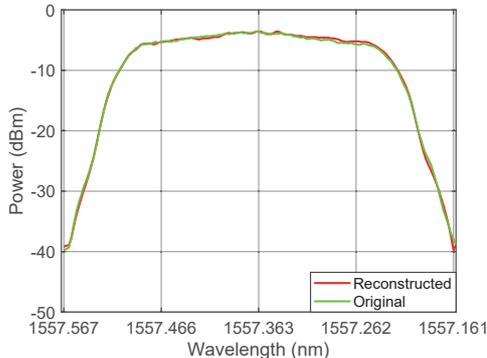


Fig. 6. Comparison of original and reconstructed spectra of a lightpath, when AE-based data compression output 3-element vectors.

and training time. Here, the MSE is calculated and averaged over all the spectrum vectors in the testing set. We can see that the average MSE almost stays the same ($\sim 3.1\%$) when we reduce output length from 6 to 3, but it increases noticeably if we keep reducing the output length to 2. Meanwhile, the results on the training time suggest that it takes more time to get an AE for compressing the spectrum vectors to those that contain fewer elements. However, the training time only increases by 21% when the output length gets decreased from 6 to 3. Therefore, we set the output length of the AE-based data compression as 3 in the rest of the paper, and in this case, the comparison of the original and reconstructed spectra of a lightpath can be seen in Fig. 6.

V. SEMI-SUPERVISED LEARNING BASED ANOMALY DETECTION IN CONTROL PLANE

This section describes the control plane design of AutoSpeckcheck, *i.e.*, the data preprocessing for automatic labeling, the CFM, and the DL-based anomaly detection and location. As lightpaths in an EON do not have a standard spectral shape, we compare the compressed spectrum vectors collected along the routing path of a lightpath for anomaly detection. To see how the optical spectrum of a lightpath changes after cascaded BV-WSS', we use the system configuration in Section III-B, and measure the optical spectra of a 100 Gbps QPSK lightpath from back-to-back to after 6 cascaded BV-WSS', where each BV-WSS has a channel width of 50 GHz. The results are shown in Fig. 7, which indicates that the spectra only shrink slightly and the additional shrinking on them becomes more ignorable, when the number of cascaded BV-WSS' increases. Specifically, the 6-dB bandwidth of the lightpath only decreases less than 1 GHz after 6 cascaded BV-WSS'. Therefore, we do not consider the spectrum narrowing effect due to cascaded BV-WSS' in the algorithm design below. Meanwhile, we need to admit that the effect can become more significantly, if the

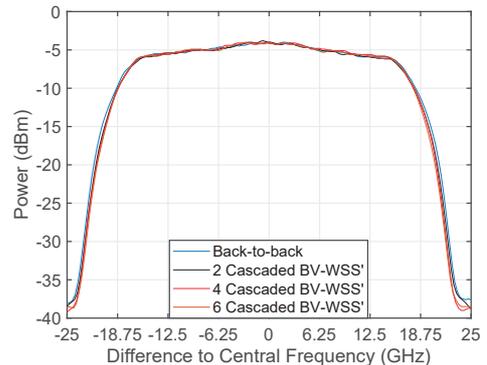


Fig. 7. Optical spectra of 100 Gbps lightpath after cascaded BV-WSS'.

lightpath is operating in the low-margin scenario. To address such cases, we can leverage the correction masks discussed in [54–57] to compensate for the spectrum narrowing effect.

A. Automatic Data Labeling

As explained in Section III, we need to design a data preprocessing technique to get rid of the hassle of labeling massive spectrum data manually. We assume that there are $|R_l|$ fiber links in the routing path of lightpath l , m_1 is the OPM that is the nearest to the source of l , and the OPMs on links from source to destination are indexed with $\{m_1, \dots, m_{|R_l|}\}$, respectively. After receiving the compressed spectrum vectors for a lightpath from the data plane, we first organize the vectors into pairs. In each pair, the two vectors correspond to the first and other hops of the lightpath, respectively. Then, we denote such a vector pair as a sample $(S_{l,m_i} = \{\hat{P}_{l,m_1}, \hat{P}_{l,m_i}\}, i \in [2, |R_l|])$. Next, we manually label a small portion of the samples (*i.e.*, one for each anomaly category), and use an unsupervised learning model based on clustering to classify anomalies based on the labeled data. Therefore, the data labeling can be accomplished automatically.

We consider two types of anomalies (*i.e.*, filter drifting and in-band jamming), and each type of anomalies can have a few severities. Hence, we define each anomaly category to include all the anomalies whose types and severities are both the same. Then, if the numbers of severities for the two anomaly types are predefined as M_1 and M_2 , respectively, we totally have $M_1 + M_2 + 1$ anomaly categories (one for the normal case).

Algorithm 1 shows the procedure of the unsupervised learning for automatic data labeling. As the numbers of severities for the two anomaly types are M_1 and M_2 , respectively, Line 1 uses $M_1 + M_2 + 1$ pre-labeled samples to represent all the anomaly categories, which are the only ones that need to be labeled manually. We use Lines 2-3 to initialize the parameters for the sample classification that tells anomaly types and severities. Here, N' is defined to limit the maximum number of iterations in the while-loop that covers Lines 4-9, and its value is determined empirically (Line 3).

Next, the while-loop tries to classify samples in \mathbf{S} into different numbers of clusters until there have been $(N' - N)$ iterations. In each iteration, we first leverage the Gaussian mixture model (GMM) [60], which tries to describe a set of data with a few Gaussian probability density functions

Algorithm 1: Unsupervised Learning for Data Labeling

Input: Number of severities for filter drifting M_1 ,
 number of severities for in-band jamming M_2 , set
 of data samples $\mathbf{S} = \{S_{l,m_i}, \forall l, i\}$.

- 1 use $M_1 + M_2 + 1$ pre-labeled compressed spectrum vector pairs (*i.e.*, samples) to represent all the anomalies;
 - 2 insert the pre-labeled samples in \mathbf{S} ;
 - 3 $N = M_1 + M_2 + 1$, $N' = 3 \cdot N$, $B = \emptyset$;
 - 4 **while** $N \leq N'$ **do**
 - 5 apply GMM-based unsupervised learning to classify vectors in \mathbf{S} into N clusters;
 - 6 calculate BIC over the N clusters to obtain b ;
 - 7 insert tuple $\{b, N\}$ in B ;
 - 8 $N = N + 1$;
 - 9 **end**
 - 10 **find** the smallest b in B and its corresponding N ;
 - 11 merge the N clusters into N' ones based on Euclidean distance between cluster centers and pre-labeled samples;
 - 12 label all the samples in \mathbf{S} according to obtained clusters;
-

(GPDFs) whose means and covariance matrices are different, to classify the vectors in \mathbf{S} into a fixed number of clusters (*Line 5*). Note that, even though GMM can adapt to clusters with arbitrary shapes, it needs to know the number of clusters to classify in advance. Hence, we fix the number of clusters in each iteration, get the clusters with GMM, and calculate the Bayesian information criterion (BIC) [61] b to evaluate the quality of the obtained clusters (*Line 6*). According to [61], a smaller b indicates a better clustering. *Line 7* records the value of b and number of clusters in the current iteration in set B . After the while-loop, *Line 10* finds the smallest b and its corresponding number of clusters N in B , which represents the best clustering. Then, *Line 11* merges the clusters according to the Euclidean distance between the center of each cluster to the pre-labeled samples. Specifically, if a cluster does not include any pre-labeled sample, we will merge it to the cluster whose pre-labeled sample has the smallest Euclidean distance to its center. Finally, we label all the samples in \mathbf{S} according to the pre-labeled sample in each obtained cluster (*Line 12*).

B. Efficient Anomaly Detection and Location

Note that, the QoT of a lightpath usually stays normal in most of its network operation time [27], and thus even after compression, abnormal vectors are still very sparse in the spectrum data received by the control plane. Hence, we propose a CFM to filter out most of the normal data before invoking the DL-based spectrum inspection for anomaly detection, such that the load of data analyzing can be greatly reduced. We denote an anomaly as $a = \{l, m, tp, sv\}$, where l refers to the abnormal lightpath, m represents the OPM that reports the abnormal spectrum vector and it actually indicates the location of the anomaly, and tp and sv tell the type and severity of the anomaly, respectively. For each lightpath l , its compressed spectrum data includes a series of vectors as

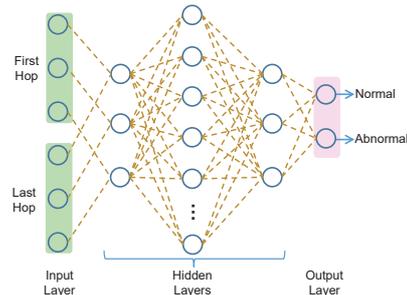


Fig. 8. Design of CFM.

TABLE II
 ACCURACY ON TESTING SET (45% OF SAMPLES ARE ABNORMAL)

Samples in Training Set	10%	30%	50%	70%	90%
GMM (%)	98.42	99.54	99.82	99.89	99.90
CFM w/ GMM (%)	99.58	99.76	99.90	99.91	99.97
CFM w/ Ground-truth (%)	99.75	99.83	99.93	99.93	99.97

$\{\hat{P}_{l,m}, \forall m \in R_l\}$, where each concerned OPM m corresponds to a fiber link on the lightpath's routing path R_l .

We design CFM based on the observation that when a lightpath is transmitted all-optically, the shape of its spectrum should stay similar from one link to another when it does not experience any anomaly, and even in the low-margin scenario, the spectrum narrowing due to cascaded BV-WSS' can be compensated with correction masks [54–57]. Specifically, for a lightpath l that has $|R_l|$ hops, the CFM can compare the spectra from OPM $m_{|R_l|}$ with that from OPM m_1 to filter out most of the normal data. The architecture of the CFM is shown in Fig. 8. Specifically, for each lightpath l , the CFM takes the compressed spectrum vectors of the first and last hops as inputs (*i.e.*, \hat{P}_{l,m_1} and $\hat{P}_{l,m_{|R_l|}}$, respectively), compares their shapes with its hidden layers, and determines whether the lightpath is abnormal with its output layer. We train the CFM with the labeled data from the data preprocessing module in the offline manner. If the CFM determines that a lightpath l is abnormal, it forwards all the related spectrum data $\{\hat{P}_{l,m}, \forall m \in R_l\}$ to the anomaly detection and location module.

As shown in Fig. 9(a), we leverage a neural network that uses exactly the same architecture of the CFM to compare the spectrum vectors in $\{\hat{P}_{l,m_1}, \dots, \hat{P}_{l,m_{|R_l|-1}}\}$, and utilize the binary search to quickly locate where the anomaly happens. Note that, the trained GMM that is for automatic data labeling can also be reused to determine the type and severity of each anomaly, as shown in Fig. 9(b). Before being putting into operation, the CFM in the anomaly detection and location module is also trained in the offline manner with the labeled data from the data preprocessing module.

VI. NUMERICAL RESULTS

In this section, we present numerical results about the DL models proposed for AutoSpecheck to show their performance.

A. Performance of Data Preprocessing for Labeling and CFM

With the 5,400 lightpaths in Section III-B, we measure the optical spectra at different locations on their routing paths,

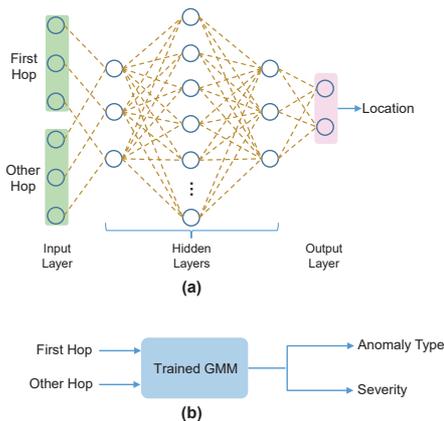


Fig. 9. Design of anomaly detection and location module.

and get 37,800 spectrum vectors, among which the vectors of normal samples take up 45% and the rest are anomalies, and the abnormal ones are evenly distributed over the 9 anomaly scenarios (*i.e.*, 8 severities of filter drifting and in-band jamming). Then, we use the AE-based approach to compress each spectrum vector to a 3-element sample, and insert all the obtained samples in a data set \mathbf{S} , which is used to evaluate *Algorithm 1*.

We have $M_1 = 8$ and $M_2 = 1$, divide \mathbf{S} into the training and testing sets that include different portions of samples, use the training set to train the GMM in the data preprocessing module, and test the accuracy of its data labeling with the testing set. Note that, when the ratio of training to testing samples is selected, we randomly allocate the samples in \mathbf{S} to them. The results on the GMM's labeling accuracy on the testing set are listed in Table II. It can be seen that even if the GMM is only trained with 10% of samples in \mathbf{S} , it can deliver a labeling accuracy of 98.42% on the remaining 90% of samples in the testing set. This confirms the effectiveness of *Algorithm 1*. Fig. 10 shows how BIC changes with the number of clusters, when we train the GMM with 10% of samples in \mathbf{S} . We observe that BIC reaches its minimum when there are 16 clusters. As we finally need to classify the samples into 10 categories, the 16 clusters are merged into 10 clusters by *Line 9* in *Algorithm 1*. Table II also suggests that the labeling accuracy on the testing set increases with the portion of samples in the training set, which is well expected because the GMM gets trained better with a larger training set.

Then, we use the labeled data obtained by the data preprocessing module to train the CFM, also with different ratios of training to testing samples. Table II also shows the classification accuracy of the CFM, where the "CFM w/ GMM" means that the CFM is trained with the data labeled by the GMM, and the "CFM w/ Ground-truth" refers to the one trained with the ground-truth data. It can be seen that due to its sophisticated architecture, the CFM always provides higher classification accuracy than the GMM even if it is trained with the data labeled by the GMM. This observation confirms two important features of our design. Firstly, the small percentage of incorrect labeling in the training set for the CFM will not affect its classification accuracy. Secondly, even if the GMM

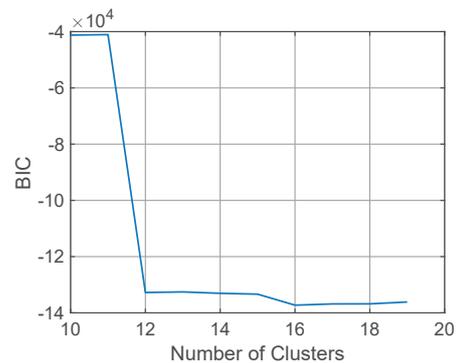


Fig. 10. BIC versus number of clusters (trained with 10% of samples in \mathbf{S}).

TABLE III
ACCURACY ON TESTING SET (10% OF SAMPLES ARE ABNORMAL)

Samples in Training Set	10%	30%	50%	70%	90%
GMM (%)	98.24	99.50	99.82	99.86	99.89
CFM (%)	98.90	99.75	99.89	99.91	99.93

already provides accurate labeling, the CFM is still necessary, because it can further improve the classification accuracy and does not need to simultaneously consider a large number of samples to accomplish the data filtering. Meanwhile, between the two CFMs, the one trained with the ground-truth data can achieve a higher accuracy, which is expected. However, the relative difference is really small (*i.e.*, with a maximum value of 0.17%), and it decreases with the size of the training set. This further proves that our semi-supervised learning can perform well regardless of the small portion of labeling errors induced by the GMM, and also suggests that training the CFM with a larger number of samples can compensate for the negative effect due to the labeling errors.

Note that, the results in Table II are obtained based on the assumption that the ratio of abnormal samples is 45% in \mathbf{S} . This, however, might not be the case in practical situations, since the QoT of a lightpath usually stays normal in most of its operation time [27]. Therefore, we include more lightpaths of normal states and exclude ones of anomalies so as to reduce the percentage of abnormal samples in \mathbf{S} from 45% to 10% and redo the simulations. The new results are listed in Table III. We can see that the general trends of the results stay unchanged, and the classification accuracies of the GMM and CFM either manage to be the same or only decrease slightly. This further verify the performance and practicalness of our design.

B. Performance of Anomaly Detection and Location

In the anomaly detection and location module, we locate anomalies by leveraging a neural network that has the same structure of the CFM and the binary search. Hence, there is no need to analyze its accuracy again, and we will show the effectiveness of the anomaly location with the experimental demonstrations in Section VII. The anomaly detection reuses the GMM in the data preprocessing module to determine the anomaly type and severity. We consider two scenarios to test the performance of the anomaly detection, *i.e.*, the

0	98.04	0	0	0	0	0	0	0	0	1.96
1	0	100	0	0	0	0	0	0	0	0
2	0	0	100	0	0	0	0	0	0	0
3	0	0	0	100	0	0	0	0	0	0
4	0	0	0	0	100	0	0	0	0	0
5	0	0	0	0	0	100	0	0	0	0
6	0	0	0	0	0	0	100	0	0	0
7	0	0	0	0	0	0	0	100	0	0
8	0	0	0	0	0	0	0	0	100	0
9	0	0	0	0	0	0	0	0	0	100
	0	1	2	3	4	5	6	7	8	9

Output of Anomaly Detection

(a) Training set includes 10% of all the samples

0	99.81	0	0	0	0	0	0	0	0	0.19
1	0	100	0	0	0	0	0	0	0	0
2	0	0	99.54	0.46	0	0	0	0	0	0
3	0	0	0	99.52	0.48	0	0	0	0	0
4	0	0	0	0	100	0	0	0	0	0
5	0	0	0	0	0	100	0	0	0	0
6	0	0	0	0	0	0	100	0	0	0
7	0	0	0	0	0	0	0	100	0	0
8	0	0	0	0	0	0	0	0	100	0
9	0	0	0	0	0	0	0	0	0	100
	0	1	2	3	4	5	6	7	8	9

Output of Anomaly Detection

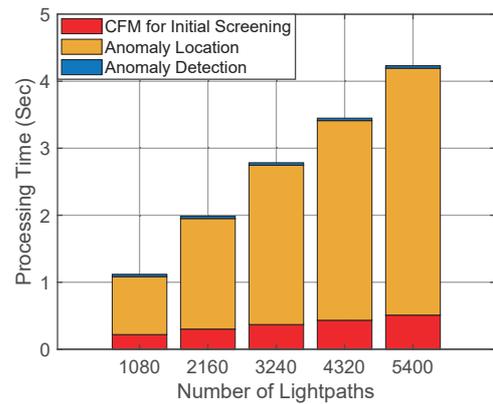
(b) Training set includes 50% of all the samples

Fig. 11. Confusion matrices of anomaly detection.

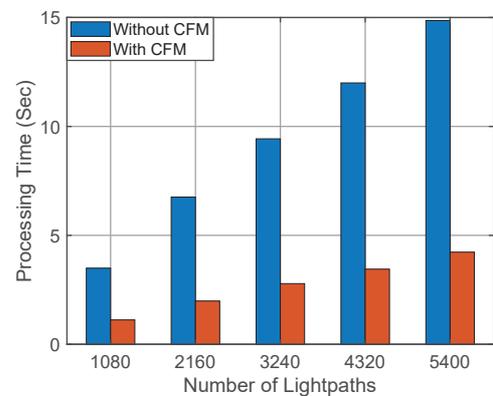
sizes of training and testing sets have ratios of 1:9 and 1:1, respectively. Meanwhile, to emulate practical situations, we set the percentage of abnormal samples in \mathbf{S} as 10%. Fig. 11 plots the confusion matrices that describe the anomaly detection's performance on the testing sets, *i.e.*, the breakdown of classification errors. The numbers on the x- and y-axes indicate the anomaly categories, each of which corresponds to an anomaly with a certain severity. More specifically, *Category-0* is for the normal case, *Category-1* to *Category-8* are for filter drifting with different severities, and *Category-9* is for in-band jamming. The results in Fig. 11 confirm that the anomaly detection can accurately tell the types and severities of anomalies in both scenarios.

C. Performance of Overall AutoSpecheck System

Finally, we perform numerical simulations to verify the overall performance of AutoSpecheck, especially its time-efficiency. We still use the DL models that are trained in the aforementioned subsections to build our AutoSpecheck system. Note that, an optical network should operate normally in most of the time, and thus in a practical scenario, the samples about abnormal lightpaths should be much less than those about normal ones [27]. Hence, to emulate such a practical scenario, we recollect spectrum vectors of the 5,400



(a) Breakdown of processing time with CFM



(b) Comparison of processing time with and without CFM

Fig. 12. Processing time used by AutoSpecheck.

lightpaths in Section III-B and keep the ratio of abnormal lightpaths as 10%. Then, we leverage AutoSpecheck to check the spectrum vectors, record its processing time, and plot the results in Fig. 12. Although its processing time increases with the number of lightpaths, the AutoSpecheck system only uses 4.23 seconds to process all the spectrum samples of the 5,400 lightpaths. On average, it only takes 783 μ s to process one lightpath, which proves the time efficiency of AutoSpecheck.

We also show the breakdown of the processing time in Fig. 12(a). It can be seen that the anomaly location takes the majority of the processing time, which is because for each abnormal lightpath, it needs to invoke the CFM multiple times in the binary search. Note that, the anomaly location happens after the initial screening of spectrum samples in another CFM, when most of the normal samples have been removed. Hence, we can expect that the time used for anomaly location could be much longer if we do not have the initial screening with the CFM. Moreover, with the initial screening, the anomaly detection only takes 37.4 msec in total to process the spectrum data of 5,400 lightpaths, which only contributes to less than 1% of the total processing time. We also remove the CFM from AutoSpecheck and measure the total processing time. The comparison of the total processing time used by the AutoSpecheck systems with and without the CFM is shown in Fig. 12(b), which indicates that the total processing time is much longer when the CFM is removed. This confirms the benefit of our CFM on improving the system's time-efficiency.

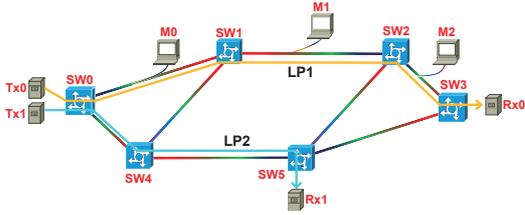


Fig. 13. Experiment setup for detecting and recovering filter drifting.

VII. EXPERIMENTAL DEMONSTRATIONS

In this section, we implement our AutoSpeck system in a real EON testbed that is built with commercial products, to further verify its performance and practicalness. The EON testbed consists of 6 nodes and 8 fiber links, as shown in Fig. 13. Each node is built with Finisar 1×9 BV-WSS' to realize flexible-grid optical switching.

A. Detection and Recovery of Filter Drifting

With the experimental setup in Fig. 13, we first conduct an experiment to detect and recover filter drifting with AutoSpeck. Here, we connect two BV-Ts on our Juniper optical packet platform to the optical ports on SW0, and the BV-Ts generate 100 Gbps QPSK optical signals centered at 1557.36 and 1558.17 nm, respectively. The optical signals are transmitted according to the routing paths of LP1 and LP2 in Fig. 13 to form two lightpaths. On the fiber links of LP1, the OPMs collect spectrum vectors of the lightpath, compress them with AE-based data compression, and send compressed spectrum vectors to the control plane every 3.5 seconds.

We capture control messages in the control plane and illustrate them in Fig. 14(a). Here, the first message block in Fig. 14(a) is for normal operation. The SpecAGs in the OPMs first report compressed spectrum vectors to the controller. Then, the controller forms the end-to-end spectrum data for the lightpath by organizing compressed spectrum vectors according to the order of OPMs that the lightpath passes through (*i.e.*, $M0 \rightarrow M1 \rightarrow M2$), and sends the lightpath data to the DLaNA. Next, the DLaNA leverages the CFM and anomaly detection and location module to check status of the lightpath, and returns anomaly reports to the controller. As shown in Fig. 14(b), because the DLaNA does not detect any anomaly, it returns the anomaly reports, in which the anomaly type is set as 0 (*i.e.*, the normal case) and the lightpath ID is also set as 0 to suggest that there is no lightpath to worry about.

We then introduce filter drifting on LP1, and to emulate the filter drifting, we configure the pass-band on SW2 for LP1 as [1557.262, 1557.566] nm during operation. The second message block in Fig. 14(a) indicates that the DLaNA quickly detects the anomaly, classifies it correctly, and sends an anomaly report to the controller. The details on the anomaly report are illustrated in Fig. 14(c), which includes the ID of the abnormal lightpath, and the position, type and severity of the detected anomaly. With the anomaly report, the controller sends a *FlowMod* message to SW2 to reconfigure it back to the normal state, as shown in the third message block in Fig. 14(a), which also indicates that the status of the lightpath returns to

No.	Time	Source	Destination	Protocol	Length	Info
823	*REF	SpecAG	Controller	Collection	589	Compressed Spectrum Vectors
825	0.002605164	Controller	DLaNA	Lightpath	591	Lightpath data
828	0.115607752	DLaNA	Controller	Anomaly	115	Anomaly: 0
1145	3.468988796	SpecAG	Controller	Collection	580	Compressed Spectrum Vectors
1147	3.469144857	Controller	DLaNA	Lightpath	582	Lightpath data
1149	3.528289607	DLaNA	Controller	Anomaly	115	Anomaly: 0
1561	7.078291880	SpecAG	Controller	Collection	589	Compressed Spectrum Vectors
1563	7.078567768	Controller	DLaNA	Lightpath	591	Lightpath data
1568	7.225292593	DLaNA	Controller	Anomaly	115	Anomaly: 0
...						
4505	35.872998001	SpecAG	Controller	Collection	583	Compressed Spectrum Vectors
4507	35.873149609	Controller	DLaNA	Lightpath	585	Lightpath data
4512	35.932732671	DLaNA	Controller	Anomaly	115	Anomaly: 1
...						
4518	35.936477982	Controller	SW2	OpenFlow	130	Type: OFPT_FLOW_MOD
4822	39.432619775	SpecAG	Controller	Collection	593	Compressed Spectrum Vectors
4824	39.482821274	Controller	DLaNA	Lightpath	600	Lightpath data
4827	39.543264751	DLaNA	Controller	Anomaly	115	Anomaly: 0

(a)

Anomaly Report	
Lightpath_ID:	0
Position:	0
Type:	0
Severity:	0

(b)

Anomaly Report	
Lightpath_ID:	1
Position:	2
Type:	1
Severity:	8

(c)

Fig. 14. Messages for detecting and recovering filter drifting.

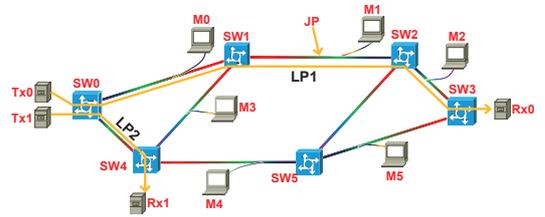


Fig. 15. Experiment setup for detecting and recovering in-band jamming.

normal at the next network monitoring time. Note that, the filter drifting is actually a soft failure that will not disconnect LP1. We measure pre-FEC-BER on the receiver end of LP1. The results indicate that when the lightpath is in its normal state, its pre-FEC-BER is $\sim 8.5 \times 10^{-5}$, and the pre-FEC-BER increases to $\sim 1.2 \times 10^{-3}$ when the filter drifting happens.

B. Detection and Recovery of In-band Jamming

We adjust the experimental setup to that in Fig. 15 to show how to detect and recover in-band jamming with AutoSpeck. This time, when the experiment first starts, LP1 still uses the same routing path as before. We still capture control messages in the control plane, and they are shown in Fig. 16(a), where the first message block is still for normal operation. Then, during operation, we inject in-band jamming into the fiber link between SW1 and SW2. The second message block in Fig. 16(a) indicates that the DLaNA in our AutoSpeck system detects the anomaly quickly, and sends an accurate anomaly report to the controller (as shown in Fig. 16(b)). Upon receiving the anomaly report, the controller decides to reroute LP1 to go through SW0 \rightarrow SW4 \rightarrow SW5 \rightarrow SW3, for staying away from the in-band jamming. As illustrated in the third message block in Fig. 16(a), the controller sends *FlowMod* messages to the related optical switches to accomplish the rerouting. Finally, the last message block in Fig. 16(a) shows that the status of LP1 goes back to normal at the next network monitoring time. When the in-band jamming is present, the pre-FEC-BER of LP1 increases from $\sim 8.5 \times 10^{-5}$ to $\sim 1.67 \times 10^{-3}$.

No.	Time	Source	Destination	Protocol	Length	Info
719	*REF*	SpecAG	Controller	Collection	610	Compressed Spectrum Vectors
721	0.002369464	Controller	DLANA	Lightpath	612	Lightpath data
725	0.114660010	DLANA	Controller	Anomaly	115	Anomaly: 0
1080	3.550373398	SpecAG	Controller	Collection	602	Compressed Spectrum Vectors
1082	3.550545651	Controller	DLANA	Lightpath	604	Lightpath data
1086	3.609115081	DLANA	Controller	Anomaly	115	Anomaly: 0
1444	7.175369832	SpecAG	Controller	Collection	595	Compressed Spectrum Vectors
1446	7.175546555	Controller	DLANA	Lightpath	597	Lightpath data
1498	7.304882234	DLANA	Controller	Anomaly	115	Anomaly: 0
...						
3102	25.142932604	SpecAG	Controller	Collection	610	Compressed Spectrum Vectors
3104	25.143101359	Controller	DLANA	Lightpath	612	Lightpath data
3131	25.200887848	DLANA	Controller	Anomaly	115	Anomaly: 1
...						
3136	25.202739797	Controller	SW4	OpenFlow	162	Type: OFPT_FLOW_MOD
3137	25.202745294	Controller	SW5	OpenFlow	162	Type: OFPT_FLOW_MOD
3140	25.202876781	Controller	SW3	OpenFlow	162	Type: OFPT_FLOW_MOD
3141	25.202914911	Controller	SW1	OpenFlow	162	Type: OFPT_FLOW_MOD
...						
3408	28.751652671	SpecAG	Controller	Collection	608	Compressed Spectrum Vectors
3490	28.751827346	Controller	DLANA	Lightpath	611	Lightpath data
3494	28.943736738	DLANA	Controller	Anomaly	115	Anomaly: 0

(a)

Anomaly Report	
Lighpaht ID:	1
Position:	1
Type:	2
Severity:	1

(b)

Fig. 16. Messages for detecting and recovering in-band jamming.

VIII. CONCLUSION

In this paper, we considered how to improve the efficiency of spectrum inspection for anomaly detection in EONs, and designed and experimentally demonstrated AutoSpecheck. Specifically, we architected AutoSpecheck based on the SD-EON architecture, proposed new techniques to greatly reduce the loads of data reporting (in the data plane) and data analyzing (in the control plane), and implemented it in a real network testbed for experimental demonstrations. To reduce the loads of data reporting in the data plane, we used the AE technique to design a spectrum data compression method, which not only significantly reduces the load of data reporting from the data plane, but also ensures that the correlations buried in spectrum data will be kept through the compression. To improve the efficiency of data analytics in the control plane, we made two major innovations. Firstly, we designed a CFM for the control plane to filter out most of the normal data before invoking the DL-based spectrum inspection for anomaly detection. Secondly, to address the difficulty of labeling massive spectrum data, we developed a DL-based anomaly detection based on semi-supervised learning. Our experimental demonstrations considered two representative intra-channel anomalies (*i.e.*, the filter drifting and in-band jamming), and the results verified that AutoSpecheck can achieve highly-efficient and automatic spectrum inspection for anomaly detection in EONs.

ACKNOWLEDGMENTS

This work was supported in part by the NSFC projects 61871357, 61771445 and 61701472, Zhejiang Lab Research Fund 2019LE0AB01, CAS Key Project (QYZDY-SSW-JSC003), SPR Program of CAS (XDC02070300), and Fundamental Funds for Central Universities (WK3500000006).

REFERENCES

- [1] Cisco visual networking index: Forecast and methodology, 2017-2022. [Online]. Available: <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>
- [2] P. Lu *et al.*, "Highly-efficient data migration and backup for Big Data applications in elastic optical inter-datacenter networks," *IEEE Netw.*, vol. 29, pp. 36–42, Sept./Oct. 2015.
- [3] L. Gong, H. Jiang, Y. Wang, and Z. Zhu, "Novel location-constrained virtual network embedding (LC-VNE) algorithms towards integrated node and link mapping," *IEEE/ACM Trans. Netw.*, vol. 24, pp. 3648–3661, Dec. 2016.
- [4] J. Liu *et al.*, "On dynamic service function chain deployment and readjustment," *IEEE Trans. Netw. Serv. Manag.*, vol. 14, pp. 543–553, Sept. 2017.
- [5] O. Gerstel, M. Jinno, A. Lord, and S. Yoo, "Elastic optical networking: A new dawn for the optical layer?" *IEEE Commu. Mag.*, vol. 50, pp. s12–s20, Feb. 2012.
- [6] L. Gong *et al.*, "Efficient resource allocation for all-optical multicasting over spectrum-sliced elastic optical networks," *J. Opt. Commun. Netw.*, vol. 5, pp. 836–847, Aug. 2013.
- [7] Y. Yin *et al.*, "Spectral and spatial 2D fragmentation-aware routing and spectrum assignment algorithms in elastic optical networks," *J. Opt. Commun. Netw.*, vol. 5, pp. A100–A106, Oct. 2013.
- [8] K. Christodoulopoulos, I. Tomkos, and E. Varvarigos, "Elastic bandwidth allocation in flexible OFDM-based optical networks," *J. Lightw. Technol.*, vol. 29, pp. 1354–1366, May 2011.
- [9] Z. Zhu, W. Lu, L. Zhang, and N. Ansari, "Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing," *J. Lightw. Technol.*, vol. 31, pp. 15–22, Jan. 2013.
- [10] L. Gong and Z. Zhu, "Virtual optical network embedding (VONE) over elastic optical networks," *J. Lightw. Technol.*, vol. 32, pp. 450–460, Feb. 2014.
- [11] Z. Dong *et al.*, "Optical performance monitoring: A review of current and future technologies," *J. Lightw. Technol.*, vol. 34, pp. 525–543, Jan. 2016.
- [12] B. Niu *et al.*, "Visualize your IP-over-optical network in realtime: A P4-based flexible multilayer in-band network telemetry (ML-INT) system," *IEEE Access*, vol. 7, pp. 82 413–82 423, 2019.
- [13] S. Tang, J. Kong, B. Niu, and Z. Zhu, "Programmable multilayer INT: An enabler for AI-assisted network automation," *IEEE Commun. Mag.*, vol. 58, pp. 26–32, Jan. 2020.
- [14] A. Vela *et al.*, "Soft failure localization during commissioning testing and lightpath operation," *J. Opt. Commun. Netw.*, vol. 10, pp. A27–A36, Jan. 2018.
- [15] D. Kilper *et al.*, "Optical performance monitoring," *J. Lightw. Technol.*, vol. 22, pp. 294–304, Jan. 2004.
- [16] W. Mo *et al.*, "ANN-based transfer learning for QoT prediction in real-time mixed line-rate systems," in *Proc. of OFC 2018*, pp. 1–3, Mar. 2018.
- [17] S. Liu, W. Lu, and Z. Zhu, "On the cross-layer orchestration to address IP router outages with cost-efficient multilayer restoration in IP-over-EONs," *J. Opt. Commun. Netw.*, vol. 10, pp. A122–A132, Jan. 2018.
- [18] L. Barletta, A. Giusti, C. Rottondi, and M. Tornatore, "QoT estimation for unestablished lighpaths using machine learning," in *Proc. of OFC 2017*, pp. 1–3, 2017.
- [19] C. Kim *et al.*, "In-band network telemetry (INT)," *Tech. Spec.*, Jun. 2016. [Online]. Available: <https://p4.org/assets/INT-current-spec.pdf>
- [20] S. Tang *et al.*, "Sel-INT: A runtime-programmable selective in-band network telemetry system," *IEEE Trans. Netw. Serv. Manag.*, vol. 17, pp. 708–721, Jun. 2020.
- [21] F. Paolucci, A. Sgambelluri, F. Cugini, and P. Castoldi, "Network telemetry streaming services in SDN-based disaggregated optical networks," *J. Lightw. Technol.*, vol. 36, pp. 3142–3149, Aug. 2018.
- [22] M. De Leenheer, Y. Higuchi, and G. Parulkar, "An open controller for the disaggregated optical network," in *Proc. of ONDM 2018*, pp. 230–233, May 2018.
- [23] Flexgrid High Resolution Optical Channel Monitor (OCM). [Online]. Available: <https://optical.communications.ii-vi.com/roadms-wavelength-management/focm01fxc1mn>
- [24] C. Chen *et al.*, "Demonstrations of efficient online spectrum defragmentation in software-defined elastic optical networks," *J. Lightw. Technol.*, vol. 32, pp. 4701–4711, Dec. 2014.
- [25] Z. Zhu *et al.*, "Demonstration of cooperative resource allocation in an OpenFlow-controlled multidomain and multinational SD-EON testbed," *J. Lightw. Technol.*, vol. 33, pp. 1508–1514, Apr. 2015.
- [26] G. Hinton, A. Krizhevsky, and S. Wang, "Transforming auto-encoders," in *Proc. of ICANN 2011*, pp. 44–51, Jun. 2011.
- [27] X. Chen *et al.*, "Self-taught anomaly detection with hybrid unsupervised/supervised machine learning in optical networks," *J. Lightw. Technol.*, vol. 37, pp. 1742–1749, Apr. 2019.

- [28] C. Natalino *et al.*, “Experimental study of machine-learning-based detection and identification of physical-layer attacks in optical networks,” *J. Lightw. Technol.*, vol. 37, pp. 4173–4182, Aug. 2019.
- [29] A. Mestres *et al.*, “Knowledge-defined networking,” *ACM Comput. Commun. Rev.*, vol. 47, pp. 2–10, Sept. 2017.
- [30] M. Salani, C. Rottondi, and M. Tornatore, “Routing and spectrum assignment integrating machine-learning-based QoT estimation in elastic optical networks,” in *Proc. of INFOCOM 2019*, pp. 1738–1746, Apr. 2019.
- [31] D. Rafique *et al.*, “Cognitive assurance architecture for optical network fault management,” *J. Lightw. Technol.*, vol. 36, pp. 1443–1450, Apr. 2018.
- [32] S. Liu *et al.*, “DL-assisted cross-layer orchestration in software-defined IP-over-EONs: from algorithm design to system prototype,” *J. Lightw. Technol.*, vol. 37, pp. 4426–4438, Sept. 2019.
- [33] M. Furdek *et al.*, “Machine learning for optical network security monitoring: A practical perspective,” *J. Lightw. Technol.*, vol. 38, pp. 2860–2871, Jun. 2020.
- [34] B. Li, W. Lu, S. Liu, and Z. Zhu, “Deep-learning-assisted network orchestration for on-demand and cost-effective vNF service chaining in inter-DC elastic optical networks,” *J. Opt. Commun. Netw.*, vol. 10, pp. D29–D41, Oct. 2018.
- [35] B. Li, W. Lu, and Z. Zhu, “Deep-NFVorch: Leveraging deep reinforcement learning to achieve adaptive vNF service chaining in EON-DCIs,” *J. Opt. Commun. Netw.*, vol. 12, pp. A18–A27, Jan. 2020.
- [36] H. Fang *et al.*, “Predictive analytics based knowledge-defined orchestration in a hybrid optical/electrical datacenter network testbed,” *J. Lightw. Technol.*, vol. 37, pp. 4921–4934, Oct. 2019.
- [37] Q. Li *et al.*, “Scalable knowledge-defined orchestration for hybrid optical/electrical datacenter networks,” *J. Opt. Commun. Netw.*, vol. 12, pp. A113–A122, Feb. 2020.
- [38] F. Musumeci *et al.*, “An overview on application of machine learning techniques in optical networks,” *IEEE Commun. Surveys Tuts.*, vol. 21, pp. 1383–1408, Second Quarter 2018.
- [39] D. Rafique and L. Velasco, “Machine learning for network automation: overview, architecture, and applications,” *J. Opt. Commun. Netw.*, vol. 10, pp. D126–D143, Oct. 2018.
- [40] F. Khan, Q. Fan, C. Lu, and A. Lau, “An optical communication’s perspective on machine learning and its applications,” *J. Lightw. Technol.*, vol. 37, pp. 493–516, Jan. 2019.
- [41] M. Liu, M. Tornatore, and B. Mukherjee, “Survivable traffic grooming in elastic optical networks - shared protection,” *J. Lightw. Technol.*, vol. 31, pp. 903–909, Mar. 2013.
- [42] F. Ji *et al.*, “Dynamic p-cycle protection in spectrum-sliced elastic optical networks,” *J. Lightw. Technol.*, vol. 32, pp. 1190–1199, Mar. 2014.
- [43] X. Chen, F. Ji, and Z. Zhu, “Service availability oriented p-cycle protection design in elastic optical networks,” *J. Opt. Commun. Netw.*, vol. 6, pp. 901–910, Oct. 2014.
- [44] E. Seve *et al.*, “Learning process for reducing uncertainties on network parameters and design margins,” *J. Opt. Commun. Netw.*, vol. 10, pp. A298–A306, Feb. 2018.
- [45] C. Rottondi, L. Barletta, A. Giusti, and M. Tornatore, “Machine-learning method for quality of transmission prediction of unestablished lightpaths,” *J. Opt. Commun. Netw.*, vol. 10, pp. A286–A297, Feb. 2018.
- [46] J. Yu *et al.*, “Model transfer of QoT prediction in optical networks based on artificial neural networks,” *J. Opt. Commun. Netw.*, vol. 11, pp. C48–C57, Oct. 2019.
- [47] Y. Pointurier, “Design of low-margin optical networks,” *J. Opt. Commun. Netw.*, vol. 9, pp. A9–A17, Jan. 2017.
- [48] A. Vela *et al.*, “BER degradation detection and failure identification in elastic optical networks,” *J. Lightw. Technol.*, vol. 35, pp. 4595–4604, Nov. 2017.
- [49] R. Proietti *et al.*, “Experimental demonstration of cognitive provisioning and alien wavelength monitoring in multi-domain EON,” in *Proc. of OFC 2018*, pp. 1–3, Mar. 2018.
- [50] B. Shariati *et al.*, “Real-time spectrum surveillance in filterless optical networks,” in *Proc. of OFC 2018*, pp. 1–3, Mar. 2018.
- [51] —, “Optical signal tracking for robust PAM4 deployment in filterless metro network scenarios,” in *Proc. of OFC 2019*, pp. 1–3, Mar. 2019.
- [52] —, “Real-time optical spectrum monitoring in filterless optical metro networks,” *Photon. Netw. Commun.*, vol. 40, pp. 1–13, Jan. 2020.
- [53] S. Varughese *et al.*, “Identification of soft failures in optical links using low complexity anomaly detection,” in *Proc. of OFC 2019*, pp. 1–3, Mar. 2019.
- [54] L. Velasco *et al.*, “Learning from the optical spectrum: Soft-failure identification and localization,” in *Proc. of OFC 2018*, pp. 1–3, Mar. 2018.
- [55] B. Shariati, M. Ruiz, J. Comellas, and L. Velasco, “Learning from the optical spectrum: failure detection and identification,” *J. Lightw. Technol.*, vol. 37, pp. 433–440, Jan. 2019.
- [56] B. Shariati, M. Ruiz, and L. Velasco, “Out-of-field generic ML training with in-field specific adaptation to facilitate ML deployments,” in *Proc. of OFC 2019*, pp. 1–3, Mar. 2019.
- [57] L. Velasco *et al.*, “Learning life cycle to speed up autonomic optical transmission and networking adoption,” *J. Opt. Commun. Netw.*, vol. 11, pp. 226–237, May 2019.
- [58] K. Christodouloupoloulos, N. Sambo, and E. Varvarigos, “Exploiting network kriging for fault localization,” in *Proc. of OFC 2016*, pp. 1–3, Mar. 2016.
- [59] Open network operating system (ONOS). [Online]. Available: <https://onosproject.org/>
- [60] Z. Zivkovic, “Improved adaptive Gaussian mixture model for background subtraction,” in *Proc. of ICPR 2004*, pp. 28–31, Aug. 2004.
- [61] S. Watanabe, “A widely applicable Bayesian information criterion,” *J. Mach. Learn. Res.*, vol. 14, no. Mar., pp. 867–897, 2013.