This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JLT.2017.2747223, Journal of Lightwave Technology

BER Degradation Detection and Failure Identification in Elastic Optical Networks

Alba P. Vela, Marc Ruiz, Francesco Fresi, Nicola Sambo, Filippo Cugini, Gianluca Meloni, Luca Potì,
Luis Velasco and Piero Castoldi

***Abstract*—Optical connections support virtual links in MPLS-over-optical multilayer networks and therefore, errors in the optical layer impact on the quality of the services deployed on such networks. Monitoring the performance of the physical layer allows verifying the proper operation of optical connections, as well as to detect bit error rate (BER) degradations and anticipate connection disruption. In addition, failure identification facilitates localizing the cause of the failure by providing a short list of potential failed elements and enables self-decision making to keep committed service level. In this paper, we analyze several failure causes affecting the quality of optical connections and propose two different algorithms: one focused on detecting significant BER changes in optical connections, named as BANDO, and the other focused on identifying the most probable failure pattern, named as LUCIDA. BANDO runs inside the network nodes to accelerate degradation detection and sends a notification to the LUCIDA algorithm running on the centralized controller. Experimental measures were carried out on two different setups to obtain values for BER and received power and used to generate synthetic data used in subsequent simulations. Results show significant improvement anticipating maximum BER violation with small failure identification errors.**

***Index Terms*— BER Degradation Detection, Failure Identification, Elastic Optical Networks.**

I. Introduction

SERVICE layer connections are usually set up on virtual network topologies, where virtual links are supported by lightpaths in the optical layer. Thus, errors in optical transmission translate into packet losses and retransmissions leading to unacceptable Quality of Service (QoS) and to Service Level Agreements (SLA) violations. Such violations represent money losses for the network operator.

Although commercially available optical equipment is able to correct degraded optical signals by means of Forward Error Correction (FEC) algorithms, a value of pre-FEC Bit Error Rate (BER) over the pre-defined limit (max BER) would imply a non-error-free post-FEC transmission and, as a result, communication would be disrupted. Therefore, a prompt detection of optical connections with excessive pre-FEC BER can greatly reduce SLA violations.

Monitoring the physical layer is essential to verify the fulfillment of SLAs and, in the case of faults or degradations, e.g. transmitter laser drift or filters misconfiguration, to localize the failed elements, and to take actions for preserving the services. Information retrieved by commonly used power monitors measuring received power (P_{Rx}) can be combined with monitoring information accessible through emerging transponders based on coherent detection [2], [3]. In particular, such transponders offer the possibility to monitor several parameters associated with connections or to the traversed links, e.g., pre-FEC BER or linear dispersion.

In fact, coherent detection has posed new possibilities for monitoring [4], [5]. Monitoring is attracting increasing interest for several reasons such as: *i*) the reduction of system margins (which derives in reducing capital expenditures) might induce more frequent degradations at the optical layer [6], [7]; *ii*) a more accurate estimation of the quality of transmission and an optimization of transmission parameters, routing, and spectrum assignment [8].

Besides optical power, thanks to the digital signal processing (DSP) module of coherent receivers, it is possible to monitor several end-to-end performance parameters. Above all, pre-FEC BER, Optical Signal to Noise Ratio (OSNR), Q-factor, and also electrical SNR can be monitored by already available commercial transponders. Moreover, other parameters can be monitored: e.g. chromatic dispersion through equalizer taps [9], the central frequency of the signal thanks to an automatic frequency control [10], polarization channel characteristics and the state of polarization [11]. Such information can be collected by commercial cards with a time period of 10 ms and can also be used for failure prediction applications allowing operators to pre-empt outages [11]. However, we have to consider that since linear impairments (e.g., dispersion) can be compensated by the DSP itself, signal degradations in coherent systems are mainly dominated by amplified spontaneous emission, non-linear effects including also interference, and filters introducing signal distortion. Thus, the identification of such most relevant impairments is mandatory. A challenge is the analysis of monitoring data with the objective of identifying the nature of a problem (e.g., decide if a reduction of the OSNR is due to an amplifier malfunction or some other issue) and such topic still requires to be investigated to reach an adequate maturity. Besides the aforementioned parameters, studies are ongoing to determine the effects of non-linearities and filters. Non-linear effects can be compensated –thus estimated– through digital back propagation techniques [12] or maximum likelihood sequence estimator [13]. Regarding filtering effects, studies in [14], [15]

Manuscript received April 12, 2017. This work was presented in part at OFC [1].

Alba P. Vela (apvela@ac.upc.edu), Marc Ruiz, and Luis Velasco are with the Optical Communications Group (GCO) at Universitat Politècnica de Catalunya (UPC), Barcelona, Spain. Francesco Fresi, Nicola Sambo, and Piero Castoldi are with Scuola Superiore Sant'Anna, Pisa, Italy. Filippo Cugini, Gianluca Meloni, and Luca Potì are with CNIT, Pisa, Italy.

evaluated the related induced penalties. However, work is still needed to correlate information related to end-to-end parameters such as OSNR and non-linear or filtering effects to identify the type of failure.

Finally, as stated in [6], failure localization or quality of transmission estimators [16] based on monitoring information typically require link-level characteristics while coherent receivers provide end-to-end information. However, link-level metrics may be obtained via correlation techniques such as network kriging when the characteristics are linearly additive (e.g., chromatic dispersion) [17], or via more advanced techniques e.g., based on machine learning algorithms, which perform better for non-linear impairments [18].

However, quality of transmission estimation, computed as a function of the links and nodes traversed by each optical connection, is a useful information since it can be used to configure a BER threshold at connection set-up, which would help to detect BER degradation by comparing the actual measured BER against it. However, if the threshold value is set to a value too close to the actual BER, many threshold-crossing notifications would be raised because of small BER changes, which, in addition to add control overhead, do not give useful information. On the contrary, if the threshold value is relaxed, e.g., closed to the equipment max BER, degradation detection could not be anticipated early enough the transmission is totally disrupted.

Regarding failure localization, several works in the literature have proposed methods for localization of hard link failures that affect a number of established connections, focused on reducing restoration times (see, e.g., [19]-[21]). All the proposed methods basically consist on computing and establishing a number of auxiliary connections (m-trails or m-cycles). In the event of a link failure, one single connection would be affected, thus localizing the failed link. Few works in the literature, however, have been focused on soft-failure localization that might affect a single or a reduced set of optical connections. In our previous work in [22], we focused on identifying the most probable cause of failure after its detection; a probabilistic failure localization algorithm based on Bayesian Networks (BN) [23] was proposed. The proposed BN needed to be trained to locate different causes of failures, which entail having previous data available. Note that since failures occur very infrequently, previous data availability is frequently unavailable.

In addition, it is hard to discern the real cause of soft failures, since, transmitter laser shift and filters misconfiguration could lead to similar evidence. However, discovering and identifying a failure pattern reduces remarkably the subsequent failure localization effort by providing a short list of potential failed elements (e.g. filters used by a certain connection). Moreover, failure identification enables self-decision making to keep committed service level, e.g., by triggering rerouting of that traffic using a connection where a gradual BER degradation has been identified. For this very reason, in this paper, we focus on BER degradation detection and failure identification. Specifically, the contribution is three-fold:

- Section II analyzes four different failures affecting the signal quality of an optical connection and motivates the definition of two different algorithms: *i*) the BER Anomaly Detection (BANDO) algorithm focused on detecting significant BER changes in optical connections, and *ii*) the Failure Identification Algorithm (LUCIDA) algorithm that identifies the most probable failure pattern.
- In Section III the proposed BANDO and LUCIDA algorithms are described in detail. BANDO algorithm is defined as a finite state machine (*fsm*) to follow the metered BER and to raise notifications in case of abrupt BER changes. LUCIDA is a probabilistic algorithm that analyzes time series from monitoring and notifications and returns the most probable failure class together with its probability.
- Experimental measures for BER and P_{Rx} obtained from two different setups are reported in Section IV. Based on measured values, realistic scenarios are generated, and exhaustive simulations are run, where obtained results show the performance of the proposed algorithms.

II. BER Degradation Detection and Failure Identification

Fig. 1 illustrates four failures affecting the signal quality of an optical connection: a) *signal overlap* (Fig. 1a) happens when the spectrum allocation of an optical connection invades that of a neighboring one. This might be caused by the inaccuracy in the central frequency of the laser and/or the filters of one of the connections; b) *tight filtering* (Fig. 1b) appears when there exists a central frequency misalignment or a width inaccuracy in the filters along the route of an optical connection. Fig. 2 presents four different causes of tight filtering, where filter F2 is misaligned in Fig. 2a, filter F2 width is narrower than the required frequency slot width in Fig. 2b, filters F2 and F3 are misaligned in Fig. 2c, and the central frequency of the signal is misaligned in Fig. 2d. Note that the consequence on the optical signal is similar for all four cases; c) *gradual drift* (Fig. 1c) appears when either the optical signal or the filter gradually deviate from the central frequency determined at set-up time; and d) *cyclic drift* (Fig. 1d) occurs when a gradual drift describes a cyclical movement with time.

For illustrative purposes, Fig. 3 plots the evolution with time of pre-FEC BER and P_{Rx} monitoring data metered at the receiver side of a connection affected by each of the failures above-described. In the case of signal overlap (Fig. 3a), the allocation of a neighboring optical connection results in a sudden increment in both, BER and P_{Rx}, of the previously established connection. In the case of the new connection, high pre-FEC BER and within limits P_{Rx} values can be measured just after its set-up. As for tight filtering (Fig. 3b), similarly as for newly established connection in the previous case, high pre-FEC BER and P_{Rx} values within limits can be

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JLT.2017.2747223, Journal of Lightwave Technology

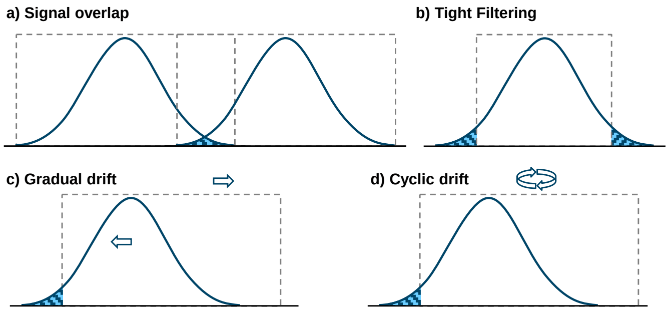

Fig. 1. Four failures affecting the signal of an optical connection.

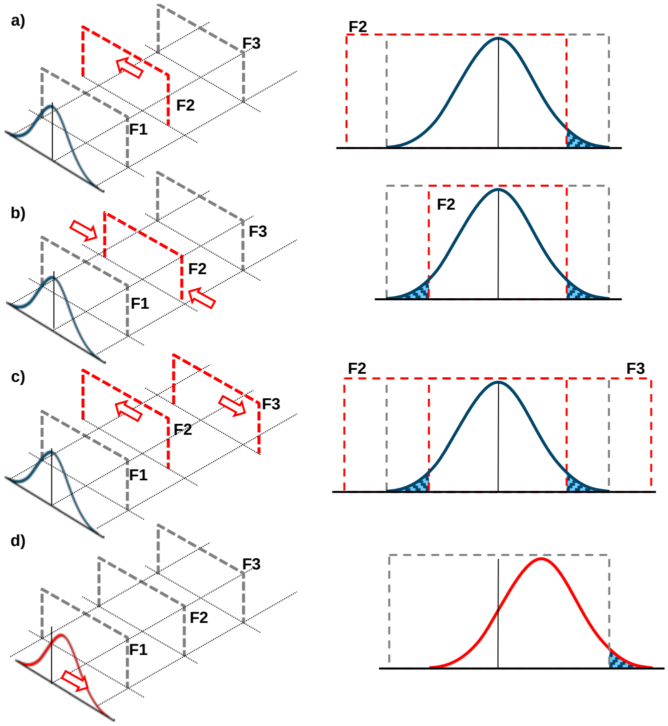

Fig. 2. Causes of tight filtering.

measured in the receptor. In the case of gradual drift (Fig. 3c), pre-FEC BER shows a gradual deterioration with time, while measured P_{Rx} reduction is almost imperceptible. Finally, in the case of cyclic drift (Fig. 3d), high pre-FEC BER and slight P_{Rx} reduction periods when part of the signal is out filters' bandwidth are followed by normal values when the signal is inside them. Note that any combination of the previous failures might happen, e.g., a *gradual cyclic drift* would produce increasingly higher pre-FEC BER periods. These cyclic failures are especially difficult to identify due to its periodic nature.

It is hard to discern the real cause of the above failures since transmitter laser degradation, and filters misconfiguration could lead to similar evidence. In this paper, we concentrate in the prompt detection of pre-FEC BER degradation and in the identification of the failure pattern as presented in Fig. 3.

For the BER degradation detection, we propose the BANDO algorithm that can be placed inside network nodes, closer to the monitoring points, to reduce the amount of monitoring data to be conveyed to the control/management plane [24]. BANDO detects changes in the monitored BER measured in the receptor of an optical connection.

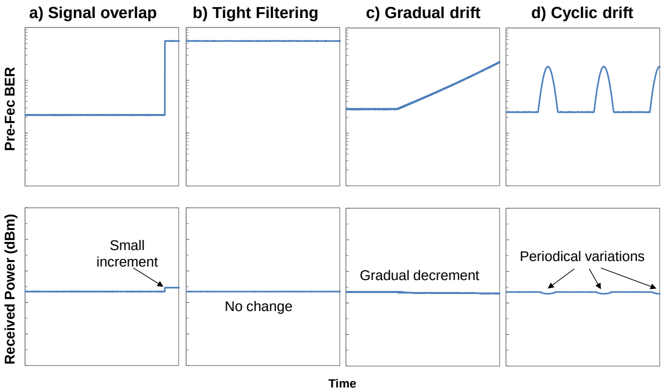

Fig. 3. Example of pre-FEC BER and P_{Rx} monitoring time series for the considered BER degradation failures.

As for the failure identification, we propose an algorithm named as LUCIDA that analyses monitoring time series and, based on the expected patterns of the considered failure causes obtained in our experiments, identifies the most probable cause of failure affecting a given set of optical connections. Because of its target, LUCIDA needs to be placed on the network manager, where monitoring data from different nodes, as well as operational data regarding the optical connections are available.

Fig. 4 presents the suggested architecture and placement to run the proposed algorithms. The BANDO algorithm runs inside the optical nodes and has access and fine-granular monitoring data to accelerate BER degradation detection. Once BER variation is detected, a notification is sent towards the network controller for further analysis, which triggers LUCIDA for failure identification. Main features of the proposed algorithms are also summarized in Fig. 4. Depending on the particular case, different reconfiguration algorithms can be triggered after the failure has been identified.

III. Algorithms for BER Degradation Detection and Failure Identification

In this section, we define BANDO and LUCIDA algorithms.

A. BER Anomaly Detection (BANDO) Algorithm

We assume that metered pre-FEC BER and P_{Rx} data for every connection is received at a given rate (e.g., every minute) and stored in a vector M of fixed capacity n in the node. BANDO algorithm analyzes pre- FEC BER data to detect gradual changes with time that might derive into BER degradation and intolerable BER values, as well as sudden anomalous BER values.

Fig. 5 illustrates three cases of BER evolution with time, where the dark continuous line represents monitored BER. Besides, two different limits are presented: *i*) *BER max* is the maximum pre-FEC BER that equipment can correct; and *ii*) a

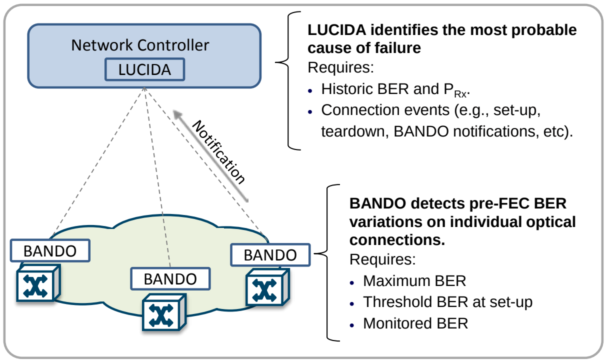
Network Controller
LUCIDA
Notification
BANDO
BANDO
BANDO
LUCIDA identifies the most probable cause of failure
Requires:
Historic BER and P_Rx.
Connection events (e.g., set-up, teardown, BANDO notifications, etc).
BANDO detects pre-FEC BER variations on individual optical connections.
Requires:
Maximum BER
Threshold BER at set-up
Monitored BER

Fig. 4. Proposed architecture and algorithm features

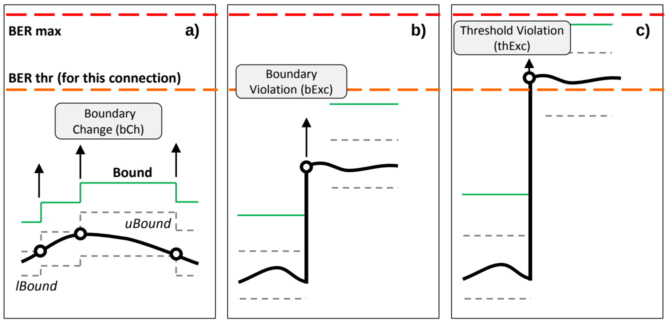

Fig. 5. BER and boundaries evolution with time

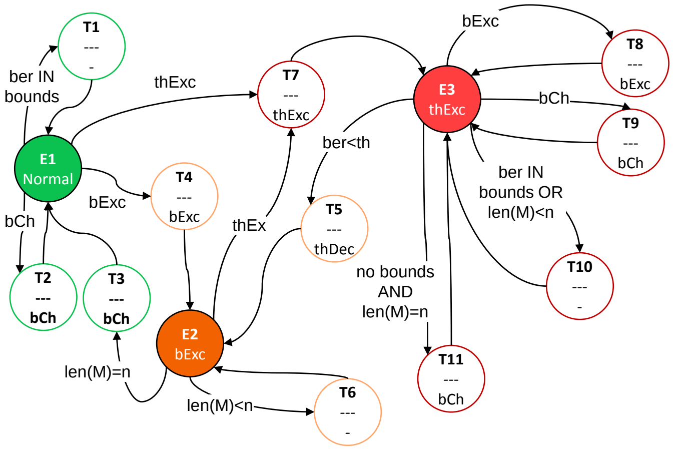

Fig. 6. BANDO finite state machine

BER threshold for the current connection computed as a function of the estimated BER (e.g., 5*estimated BER) and represents the maximum tolerable BER for such connection.

To follow BER evolution with time, an *outer boundary* is used to anticipate BER threshold violation and to detect sudden BER variations. In addition, two *inner boundaries*, named as a lower boundary (*lBound*) and upper boundary (*uBound*), are used to trigger boundary re-estimation when measured BER reaches, exceeds or falls below one of them. Inner and outer boundaries are estimated as *bound* = $\mu(M.ber)$ +/- $k{\cdot}\sigma(M.ber)$, where $\mu(M.ber)$ and $\sigma(M.ber)$ are the mean and the standard deviation computed on the last *n* BER measures and *k* is a multiplicative factor different per each boundary. Every time an event occurs, a notification is sent to the controller and analyzed by LUCIDA; defined events include: *i*) the *boundary* is re-estimated (*bCh*), *ii*) the boundary is exceeded (*bExc*), *iii*) BER exceeds the threshold (*thExc*) and *iv*) BER falls below the threshold (*thDec*).

Fig. 5a presents an example of monitored BER evolution with time causing boundary changes. As soon as monitored BER crosses one of the inner bounds, a boundary re-estimation is triggered, and a notification is sent toward the controller. Note that such boundary changes do not necessarily entail excessive BER, so the notification has an INFO severity level. Fig. 5b) and Fig. 5c) present two examples of sudden BER variation where the bound and the BER threshold is exceeded, respectively. In such cases, boundaries are reset, and notifications are sent to the controller with WARNING and MAJOR severity levels, respectively. Note that, in case pre-FEC BER exceeds maximum BER, a notification will be sent with a CRITICAL severity level.

The BANDO algorithm has been designed as an fsm with three main states and 11 transient states (Fig. 6); main states are used to store whether BER status is normal or has exceeded either the boundary or the threshold, whereas transient states are used to produce notifications and actions (i.e., boundaries re-estimation or reset). Every time a sample arrives, two fsm transitions are performed, one to obtain the output and action, and another to move to the new main state.

State E1 (*normal BER*) is reached when the last BER value falls below the boundary and the threshold. Transitions to transient state T1 follow BER within boundaries, while transitions from transient states T2 and T3 re-estimate the boundaries (as in Fig. 5a). State E2 (*boundary exceeded*) is reached when the last BER value has exceeded the boundary, but it is still below the threshold (as in Fig. 5b). Transitions from transient states T4 and T5 reset boundaries, so *n*-1 new samples are needed to arrive to re-compute new boundaries. Finally, state E3 (*threshold exceeded*) is reached when the last monitored BER is above the threshold (as in Fig. 5c). Transitions from transient states T7 and T8 reset boundaries, whereas from transient states T9 and T11 re-estimate them.

B. Failure Identification (LUCIDA) Algorithm

Regarding failure identification, we propose LUCIDA as a probabilistic algorithm that returns the most likely failure among a set *Q* of failure classes. Firstly, LUCIDA computes the probability of a set of relevant features *H* that can be observed on collected monitoring time series. In view of the failures described in Section II, three relevant features that can be identified and quantified in time series are: 1) *P_{Rx} above the reference level* (*P_{RX}high*); 2) *BER positive trend* (*BERTrend*); and 3) *BER periodicity* (*BERPeriod*). Secondly, LUCIDA maps feature probabilities to failure probabilities by means of predefined combination functions.

Upon the reception of a BANDO notification with a relevant BER change, i.e., either *bExc* or *thExc*, the algorithm in Table I is triggered. After retrieving useful data from the received notification, it is stored in a notification database (DB) for further analysis (lines 1-4 in Table I). Then, the ratio between the last monitored BER value and the connection BER threshold is computed and compared to parameter δ.

TABLE I LUCIDA ALGORITHM

```
INPUT: notif, $\alpha$, $\beta$, $\delta$
OUTPUT: <class, prob, timeToMaxBER>
 1:  connId ← notif.conn_id
 2:  lastBer ← notif.getLast().ber
 3:  threshold ← notif.threshold
 4:  storeNotif(connId, notif)
 5:  if (lastBer / threshold) < $\delta$ then return <'no',-,->
 6:  M.D={<t,ber,$P_{Rx}$>} ← getData(connId)
 7:  M.N={<t,type,data>} ← getNotif(connId)
 8:  $P_H$ ← computeFeatureProbs(M, $\alpha$) (Table II)
 9:  $P_Q$ ← computeFailureProbs($P_H$, $\beta$)
10:  failureClass ← arg $\max_{(q \in Q)}(P_Q)$
11:  tmax ← -
12:  if $P_H$(BERTrend) > 0 then
13:      tmax ← computeTMax (M)
14:  return <failureClass, $P_Q$(failureClass), tmax>
```

In case the ratio does not exceed δ, we assume that no failure is evinced (line 5); otherwise, failure detection is positive, and the identification procedure is started (lines 6-14).

Failure identification is based on processing historical BER and P_{Rx} time series obtained from a monitoring DB, as well as historical notifications stored in the notification DB. The first step consists in retrieving P_{Rx} and notification time series that are stored in the local structure M and computing feature probabilities from data (lines 6-8). Once all feature probabilities have been computed, failure probabilities are evaluated (line 9). For each of the failures q, a score is computed by means of the product of feature probabilities (eq. (1)), where β_{qh} coefficients are defined in the interval [0,1].

$$S(q)=\prod_{h\in H}\left[\beta_{qh}\cdot P_H(h)+\left(1-\beta_{qh}\right)\cdot\left(1-P_H(h)\right)\right] \qquad (1)$$

For example, if β_{qh}=1 the partial score of feature h equals $P_H(h)$; if $P_H(h)$=0, the partial score will be 0, thus discarding the evidence of failure q. Finally, to obtain a failure probability in the range [0,1], the score of every failure is normalized (eq. (2)).

$$P_Q(q)=S(q)\Big/\sum_{q\in Q}S(q) \qquad (2)$$

The failure class with the maximum probability is retrieved (line 10). Additionally, if the probability of feature BERTrend is non-zero, the time when maximum BER would be reached is estimated by means of linear extrapolation computed from monitoring data (lines 11-13).

Table II details the algorithm to compute feature probabilities. Input time series are firstly split into two segments: *i*) the *stationary* segment (D_S) containing the oldest samples which average and standard deviation remain near to a constant value, and *ii*) the *non-stationary* segment (D_{NS}) that contains the most recent samples where meaningful changes of mean and/or standard deviation with respect to the stationary segment are observed (line 1 in Table II). The rationale behind this division is based on the assumption that monitored signals behave stationary in time under normal conditions and that stationary behavior is severely altered in the event of a failure.

TABLE II COMPUTEFEATUREPROBS ALGORITHM

```
INPUT: M, $\alpha$
OUTPUT: $P_H$
 1:  <$D_s$, $D_{ns}$> ← splitSegments(M.D)
 2:  F ← $\mathcal{N}(\mu(D_s.P_{Rx}), \sigma(D_s.P_{Rx}))$
 3:  x ← M.N.$P_{Rx}$.getLast()
 4:  $P_H[P_{RX}high] = p(x)$ ← computeTruncatedProb(F, x, $\alpha$)
 5:  $P_H$[BERTrend]←0; $P_H$[BERPeriod] ← 0
 6:  $D_{notif}$ ← extractDataSeries(M.N)
 7:  for D in {$D_{ns}$.BER, $D_{notif}$.BER} do
 8:      max_model ← linearRegression(D,'max')
 9:      x ← max_model.slope.mean
10:      F ← $\mathcal{N}$(0, max_model.slope.std)
11:      p(x) ← computeTruncatedProb(F,x,$\alpha$)
12:      $P_H$[BERTrend] ← max($P_H$[BERTrend], p(x))
13:      min_model ← linearRegression(D,'min')
14:      D' ← normalize(D, min_model, max_model)
15:      SP={<period, density>} ← spectrogram(D')
16:      SPhigh← {sp∈SP | sp.density≥ mean(SP.density)}
17:      x ← 1-(|SPhigh|/|SP|)/0.5
18:      F ← $\mathcal{N}(0, (|SPhigh|)^{-1/2})$
19:      p(x) ← computeTruncatedProb(F,x,$\alpha$)
20:      $P_H$[BERPeriod] ← max($P_H$[BERPeriod], p(x))
21:  return $P_H$
```

To compute feature probabilities, we obtain the probability distribution function F that returns high probabilities when the evidence of the desired feature is significant. To give emphasis to significantly high feature values (x), we use the truncated probability distribution defined in eq. (3), where $F(x)$ is the cumulated probability when the feature takes the value x, $\alpha\in[0,1)$ is the minimum allowed probability in F, and $F^{-1}(\alpha)$ is the inverse of the distribution function and returns the value with a cumulative probability equal to α.

$$p(x)=\begin{cases} 0, & x<F^{-1}(\alpha) \\ \dfrac{F(x)-\alpha}{1-\alpha}, & x\geq F^{-1}(\alpha) \end{cases} \qquad (3)$$

The probability of feature $P_{RX}high$ is computed by characterizing the probability distribution of P_{Rx} in the stationary segment, i.e., the P_{Rx} reference level (lines 2-4). Without loss of generality, we assume a Gaussian distribution function defined by the mean and standard deviation of the samples in that stationary segment ($\mathcal{N}(\mu,\sigma)$).

In the case of features related to BER, the non-stationary time series segment D_{ns} is used. Since D_{ns} time series could be noisy, we consider another time series D_{notif}, created from the notifications M.N time series that could reflect more clearly the desired features of trend and periodicity; D_{notif} data is completed with intermediate data points computed by linear interpolation (line 6). In the algorithm, we compute BER-related feature probabilities in both time series and return the highest probability for each feature (lines 7-21).

For the *BERTrend* feature, the linear model that represents the evolution of the maximum BER with time is found; time series are split into several chunks, and the model is obtained by applying linear regression to the pairs <time, maximum value>. Note that this model collects trend independently of whether the time series has a meaningful period or not (line 8).

The mean and the standard deviation of the slope of the model allow evaluating whether that mean slope is significantly higher than 0 (lines 9-10). Finally, the feature probability is computed eq. (3) and P_H is updated (lines 11-12).

As for the BERPeriod feature, we compute a linear model for the evolution of the minimum BER that it is used, together with that for the maximum, to normalize the selected time series D as specified in eq. (4) (lines 13-14).

$$D'(t) = \frac{D(t)\text{-}min\_model(t)}{max\_model(t)\text{-}min\_model(t)} \tag{4}$$

Next, the spectrogram of D' is computed to obtain the density value for every possible period interval (line 15) [25]. To detect periodicity, we look for periods with densities clearly higher than the majority of the densities; hence, we find the set of periods with a density over the mean and its proportion over the total of periods is compared to the expected proportion in case of no periodicity, i.e., 0.5 (lines 16-17). Since x tends to be 0 when no meaningful period is observed, we use a Gaussian distribution centered in 0 and with a standard deviation inversely proportional to the number of periods over the mean (line 18). Feature probability is eventually computed and P_H updated (lines 19-20).

As a final remark, it is worth noting that the accuracy to detect and identify failures is subject to various factors, including the configuration of BANDO and LUCIDA parameters. The next section presents illustrative results to find the best configuration leading to fast and accurate detection and identification of failures.

IV. Results

In this section, we first present the experimental setup needed to evaluate the performance of the proposed algorithms for early pre-FEC BER degradation detection and failure identification and then, illustrative simulation results are presented.

A. Experimental Measurements for BER and P_{RX}.

In this subsection, we experimentally reproduce the BER degradation failures presented in Section II aiming at retrieving data that will be used to generate synthetic data for the simulations in the next subsection. A comprehensive set of measurements is carried out in a network testbed employing two types of 100Gb/s transmission systems and different configurations of traversed filters, channels spacing, and optical spans.

Fig. 7 presents the first experimental setup used for the measurements. In this setup, the considered 100Gb/s transmission system is based on Nyquist wavelength division multiplexing (NWDM) technology, derived from the lab implementation utilized in [10]. A digital-to-analog converter (DAC) is used to periodically output pulse-shaped electrical signals which drove the Mach-Zehnder based IQ-modulators. A root raised cosine (RRC) with a roll-off of 0.2, and a bandwidth of 15 GHz is used to confine signal bandwidth.

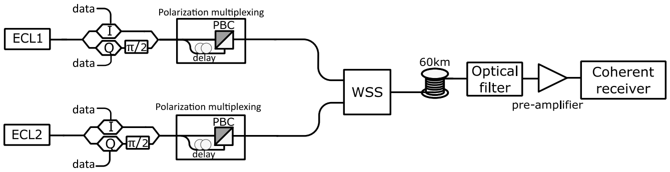

Fig. 7 Data plane experimental testbed (first setup)

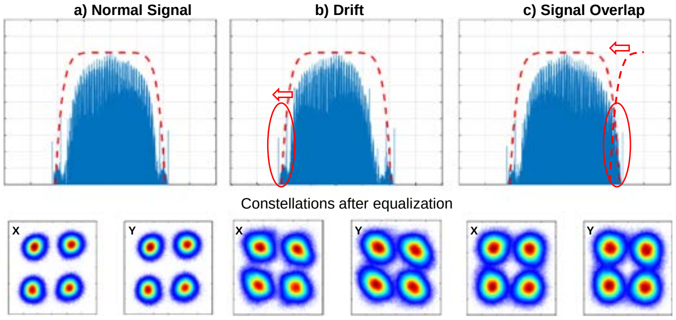

Fig. 8. Experimental results for the normal conditions and considered failures

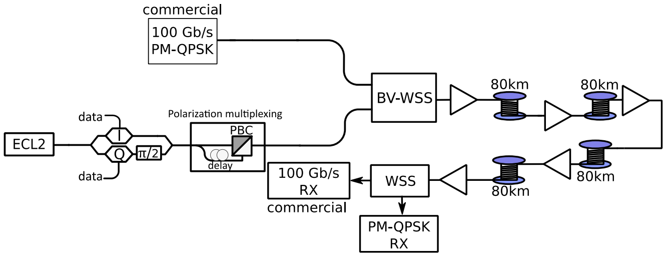

Fig. 9. Data plane experimental testbed (second setup)

Two single polarization IQ-modulators are used to modulate two external cavity lasers (ECL) and generate two quadrature phase-shift keying (QPSK) at a gross baud rate of 30 Gbaud (i.e., 60 Gb/s gross bit rate). Next, the bit rate is doubled by a polarization multiplex emulation stage thus, obtaining two 120 Gb/s polarization multiplexed (PM)-QPSK signals. The two modulated lasers are then multiplexed by means of a bandwidth variable wavelength selective switch (BV-WSS) configured to reserve a 37.5 GHz frequency slot for each channel. Measurements are reported for signal 1.

In a first experiment, drift effects are applied by inducing frequency drift to signal 1. In a second experiment, signal overlap is introduced by applying laser drift to signal 2. In this second case, the channel spacing among the signals decreases, inducing an increase of interference. The spectrum related to signal 1 is reported in Fig. 8 for both experiments. Fig. 8a shows signal 1 spectrum under normal conditions. Fig. 8b reports on the first experiment, showing the slight shift in frequency due to the laser drift. Fig. 8c reports on the second experiment, showing that part of signal 2 falls within the bandwidth of signal 1.

Additional experiments have been performed on a second setup (Fig. 9) exploiting, as signal 1, a commercial 100Gb/s transmission system based on polarization multiplexing quadrature phase shift keying (PM-QPSK) and coherent detection. In this second setup, four 80km-spans are also introduced to assess the system performance under different conditions of OSNR.

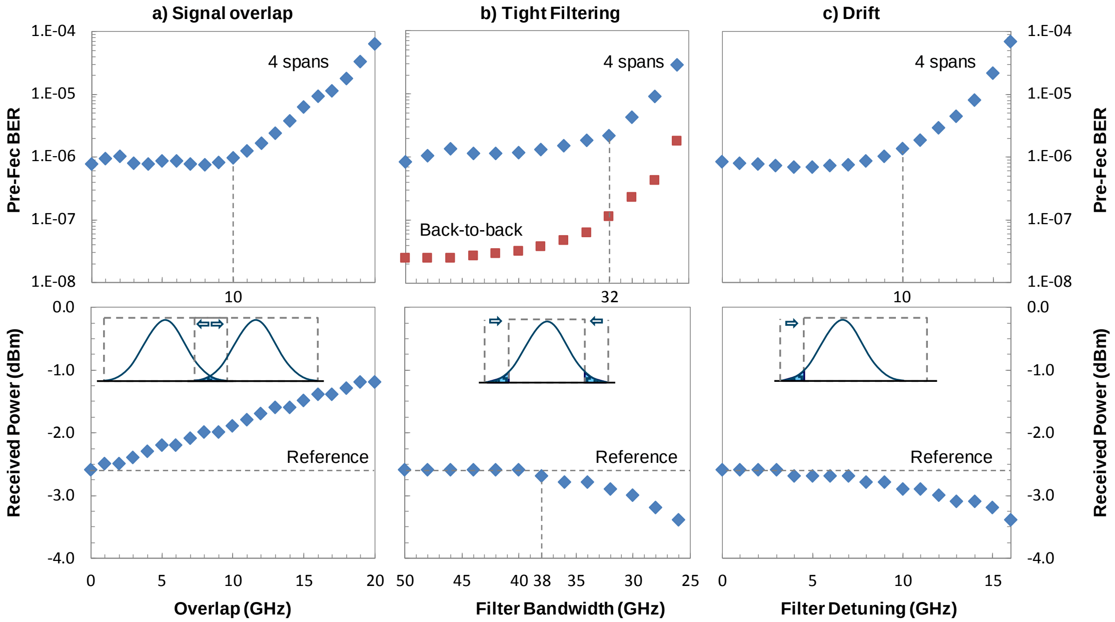

Fig. 10. Experimental BER and P_{Rx}

Fig. 10 reports pre-FEC BER and P_{Rx} provided by the commercial 100Gb/s system. It is worth noting the differences between these plots and those in Fig. 3, where historical time series are plotted. Plots in Fig. 10a show the measured values in the case of signal overlap. In particular, the 100Gb/s signal used in the first setup is now utilized to induce overlap the commercial 100Gb/s signal (*x*-axis in Fig. 10a reports such overlap). Results show that the pre-FEC BER starts to increase when channel overlapping goes above 10GHz, while received power starts increasing for small channel overlapping values since part of signal 2 enters in signal 1 bandwidth.

In the case of tight filtering (Fig. 10b) *x*-axis reports the actual bandwidth configured on the traversed BV-WSS. The central frequency of the signal has been aligned with the center of the filter, i.e., both sides of the signal are equally affected when tight filtering is applied. For comparison, Fig. 10b also reports the measurements for the first experimental setup (back-to-back configuration). Results show that up to 32GHz can be configured without significant penalties, whilst further reduction of the actual frequency slot drives signal degradations. Note that post-FEC performance is error free in all the reported plots. The minimum supported filter configuration is 26GHz (lower values would tear down the connection since post-FEC error free condition can be no longer guaranteed). Regarding received power, results show a clear deviation from the reference value, starting from a frequency slot of 38GHz.

A similar behavior was observed in the case of drift (Fig. 10c), where *x*-axis reports filter detuning. Pre-FEC BER increases when filtering effects become more relevant because of filter detuning. Results show that the pre-FEC BER starts to increase when filter detuning goes above 10GHz. Obviously, received power decreases when part of the power is cut by the filter; a clear deviation from the reference value is shown for the received power when filter detuning goes above 10GHz. Another case of drift is that of the laser. 48-hours monitoring was performed with the bandwidth set to 30GHz and, because laser drift of the commercial card, pre-FEC BER was observed as if the bandwidth was 28GHz.

As a conclusion, although the behavior of the pre-FEC BER looks similar for all three failure cases, that of the received power is different. Indeed, the proposed LUCIDA algorithm is based on the identification of such different behaviors to discern between failures.

B. Degradation Detection and Failure Identification

According to the experimental measurements in the previous section, we generated synthetic monitoring time series at a rate of one sample per minute by means of a generator implemented in R v3.2.5. Each monitoring sample includes a synthetic measure of pre-FEC BER and P_{Rx}. The generator allows reproducing realistic monitoring activity of a set of optical connections with different characteristics, such as route, spectrum allocation, and slot width. Based on such characteristics and those of the underlying optical network topology, signal behavior in the absence of failures is generated. Besides, a per-connection BER threshold is computed based on an estimated BER value computed as a function of the links' OSNR in its route [16].

The generator allows reproducing any of the failures described in Section II. According to the selected failure, one or more connections become affected at a given time, when some of their relevant physical properties are altered, e.g., filter bandwidth is narrowed; in the case of gradual changes, the magnitude of the alteration increases linearly with time following a predefined rate. Varying optical connection properties, failure class, failure magnitude, and gradual variation rate, we generated more than 100 distinct configurations. For each configuration, five 60-day instances (each generating 86,400 monitoring samples per optical

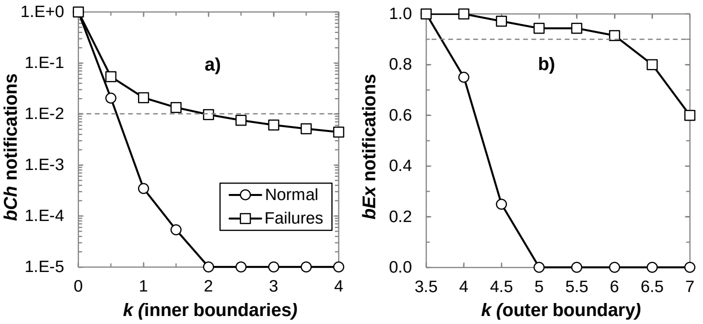

Fig. 11. Tuning of BANDO parameters.

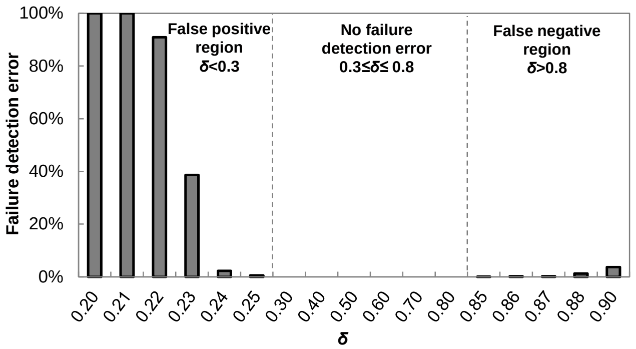

Fig. 12. Failure detection errors

connection) were randomly generated. Some of these configurations produced instances where BER never exceeded connection's BER threshold (we call this as the *lowBER* set), whereas the rest contain at least one monitoring sample exceeding the connection's BER threshold (we call this as the *highBER* set).

Both BANDO and LUCIDA algorithms were implemented in R and integrated into a simulator following the architecture presented in Fig. 4. Aiming at finding the best configuration for BANDO parameters (to avoid an excessive number of notifications being sent to the controller while keeping it informed of meaningful BER changes), we set n=15 and perform several tests with a wide range of k values for inner and outer boundaries; results are reported in Fig. 11 were values are normalized to those for the minimum k.

Starting with inner boundaries, Fig. 11a shows a number of *bCh* notifications for different values of k for connections affected by a failure and for those normal. Hence, configuring k equal to 3 allows keeping boundaries constant when normal BER behavior is monitored. In the event of connections with failure, less than 1% of all monitoring samples generate a *bCh* notification, which is enough to keep track of BER evolution with time as it will be shown in the following results.

Regarding the outer boundary, Fig. 11b shows the amount of *bExc* notifications as a function of k. Fixing k equal to 6 eliminates those notifications caused by atypical BER measures that do not entail failures, as well as keeps more than 90% of those notifications raised in the event of a failure. It is worth noting that *bExc* notifications are much less frequent than *bCh* ones and consequently, its impact on total notification overhead is negligible.

Once BANDO has been properly configured, simulations including failure detection and identification were run. We configured LUCIDA parameters α=0.7 and β_{qh} = 1 if failure q must present evidence of feature h (β_{qh} = 0, otherwise).

In the simulations, LUCIDA was triggered in two distinct modes: only upon the reception of a *thExc* notification (*Major* mode) and upon the reception of any notification (*Info* mode). It is worth noting that only the *Info* mode allows detecting failures in the lowBER set, which confirms the need of BANDO and LUCIDA collaboration.

For the *lowBER* instance set and the *Info* triggering mode, Fig. 12 analyses δ parameter tuning, where the percentage of decision errors is plot as a function of its value. Since BER threshold is set as 5 * estimated BER, we assume δ=0.2 as starting point. When δ<0.3, some normal optical connections cross the failure detection condition and are classified as one of the failure classes thus, producing a false positive detection. On the other end, δ>0.8 produces that some actual failures never reach the detection limit and hence, they are wrongly classified as normal (false negatives). In the middle, failure detection has no error and hence, we assume δ=0.5 for the ongoing results.

Let us now focus on the identification of the detected failures. Table III details the identification error upon the reception of the first triggering notification. Note that no identification error is observed for signal overlap and tight filtering failures, which is a good result since these failures generate very few notifications and need to be identified as soon as they are detected.

As anticipated above, the *Info* mode allows LUCIDA to detect all signal overlap and tight filtering failures, even when they do not produce BER samples over the threshold, which enables detecting soft failures hidden below a too high threshold.

In the case of gradual and cyclic drift failures, the first identification is not correct in most of the cases since they are related to BER trend and periodicity features, and time is needed to ensure their presence or absence. However, both failures produce many and various notifications compared to signal overlap and tight filtering ones, and therefore, the opportunity of identifying the failure extends beyond time.

In view of the above, we study the time needed for a right failure classification of gradual and cyclic drift failures. Plots in Fig. 13a for the *Info* mode and Fig. 13b for the *Major* mode represent the evolution of the computed failure probability of a cyclic drift failure as a function of the number of periods since the first bCh event. Note that markers represent only those notifications that actually triggered failure identification phase, that is when the ratio between the last monitored BER and the threshold exceeds δ. In both modes, the most probable failure identified when a triggering notification is received

TABLE III FAILURE IDENTIFICATION ERRORS (FIRST NOTIFICATION)

Failure	**highBER**		**lowBER**	
	Major	***Info***	***Major***	***Info***
Signal overlap	0%	0%	-	0%
Tight filtering	0%	0%	-	0%
Gradual drift	33%	37%	-	30%
Cyclic drift	70%	48%	-	54%

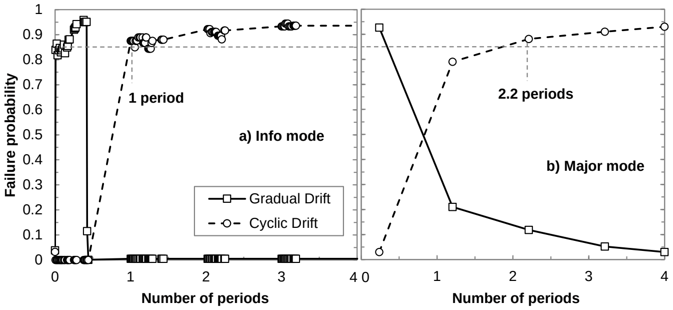

Fig. 13. Cyclic Drift Identification

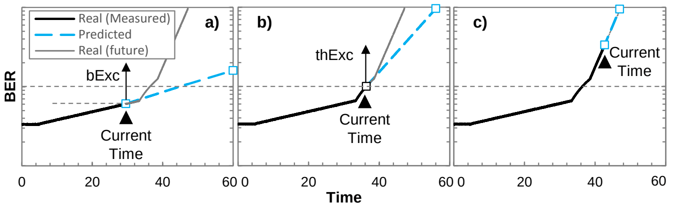

Fig. 14. Max BER anticipation

before the first periodical peak is gradual drift since during the raising front LUCIDA detected a meaningful trend. In contrast, the probability of the cyclic drift failure class is negligible since no periodicity was found. However, when a complete period is observable, BER periodicity feature starts being significant and cyclic drift becomes the most probable failure class from that point on. The difference between both modes is the time for a right failure classification; because under the *Info* mode LUCIDA receives several notifications as a result of different events detected by BANDO, it allows a clearer identification of the non-stationary time series segment, and therefore, it is able to produce right classifications after one single period, i.e., less than half of time compared to the *Major* mode. Although illustrated in Fig. 13 for just an instance, this gain keeps constant for all other cyclic drift instances.

Finally, Fig. 14 illustrates the accuracy of the estimation for the time when max BER (1E-6) will be reached in case of a gradual drift failure. Prediction based on linear extrapolation is shown at three different time instants. Although the failure is perfectly identified as gradual drift upon the reception of a *bExc* at day 30 (Fig. 14a), due to the lack of evidence of the actual future BER trend evolution, no max BER violation in the following 30 days is predicted. Later, upon the reception of a *thExc* at day 36 (Fig. 14b), max BER violation is predicted to happen in the near future. It is not until day 42, i.e., five days before the connection is disrupted, that prediction becomes steady to a constant value, which happens in Fig. 14c; hence, this method provides enough anticipation for an optimal reaction against the failure. Comparable results were obtained for the rest of gradual drift instances.

V. CONCLUDING REMARKS

SLA violations entail money losses for the network operators and hence, minimizing such violations is of paramount importance to them. This paper focused on anticipating BER degradation detection at the optical layer, which typically supports many of the offered services. In addition to a prompt BER degradation detection, the paper targeted at failure identification to help to localize the cause of the failure.

In this regard, two cooperating algorithms have been proposed: *i*) the BER Anomaly Detection (BANDO) algorithm which works inside the optical nodes to take advantage of a fine monitoring granularity, and *ii*) the Failure identification algorithm (LUCIDA) algorithm, working in the centralized network controller. BANDO detects changes in the BER of optical connections and sends notifications to LUCIDA.

To evaluate the performance of the algorithms, different BER degradation failures were considered, including gradual and periodical degradation. Aiming at studying realistic scenarios, experimental measures were carried out on two different setups involving commercial equipment. The results of the experiments were used to generate synthetic data used to simulate the considered BER degradation failures.

Simulation results show that maximum BER violation was anticipated several days before the connection was disrupted, which allows planning a network reconfiguration to be performed on low activity hours. Interestingly, the cooperation of BANDO and LUCIDA algorithms demonstrated its advantage for failure identification compared to a centralized algorithm receiving notifications only after BER threshold violations.

ACKNOWLEDGMENT

This work was partially supported by the EC through the ORCHESTRA (G.A. nº 645360) and METRO-HAUL (G.A. nº 761727) projects, from the Spanish MINECO SYNERGY project (TEC2014-59995-R) and from the Catalan Institution for Research and Advanced Studies (ICREA).

REFERENCES

[1] A. P. Vela, M. Ruiz, F. Fresi, N. Sambo, F. Cugini, L. Velasco, and P. Castoldi, “Early Pre-FEC BER Degradation Detection to Meet Committed QoS,” in Proc. OFC, 2017.

[2] A. Napoli et al., “Next generation elastic optical networks: The vision of the European research project IDEALIST,” IEEE Communications Magazine, vol. 53, pp. 152 162, 2015.

[3] M. Dallaglio, N. Sambo, J. Akhtar, F. Cugini, P. Castoldi, “YANG Model and NETCONF Protocol for Control and Management of Elastic Optical Networks”, in Proc. OFC 2016.

[4] Z. Dong et al., “Optical performance monitoring in DSP-based coherent optical systems,” in Proc. OFC, 2015.

[5] Z. Dong, F. N. Khan, Q. Sui, K. Zhong, C. Lu and A. P. T. Lau, “Optical Performance Monitoring: A Review of Current and Future Technologies,” IEEE/OSA Journal of Lightwave Technology, vol. 34, pp. 525-543, 2016.

[6] Y. Pointurier, “Design of low-margin optical networks,” in Proc. OFC, 2016.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JLT.2017.2747223, Journal of Lightwave Technology

[7] P. Soumplis, K. Christodoulopoulos, M. Quagliotti, A. Pagano, E. Varvarigos, “Actual Margins Algorithm for Multiperiod Planning”, in Proc. OFC 2017.

[8] K. Christodoulopoulos et al., “ORCHESTRA - Optical performance monitoring enabling flexible networking,” in Proc. ICTON, 2015

[9] R. Corsini et al., “Blind Adaptive Chromatic Dispersion Compensation and Estimation for DSP-Based Coherent Optical Systems,” IEEE/OSA Journal of Lightwave Technology, vol. 31, pp. 2131-2139, 2013.

[10] F. Cugini et al., “Push-pull defragmentation without traffic disruption in flexible grid optical networks,” IEEE/OSA Journal of Lightwave Technology, vol. 31, pp. 125-133, 2013.

[11] WaveLogic Ai, Ciena, http://www.ciena.com/products/wavelogic/wavelogic-Ai/

[12] Z. Pan et al., “Volterra filtering for nonlinearity impairment mitigation in DP-16QAM and DP-QPSK fiber optic communication systems,” in Proc. OFC, 2011.

[13] D. Marsella et al., “Maximum Likelihood sequence detection for mitigating nonlinear effects,” IEEE/OSA Journal of Lightwave Technology, vol. 32, pp. 908-916, 2014.

[14] Y. Sakamaki, T. Kawai, T. Komukai, M. Fukutoku, and T. Kataoka, “Evaluation of optical filtering penalty in digital coherent detection system,” IEICE Communications Express, vol. 1, pp. 54-59, 2012.

[15] A. Ghazisaeidi et al., “Impact of tight optical filtering on the performance of 28 Gbaud Nyquist-WDM PDM-8QAM over 37.5 GHz grid,” in Proc. OFC, 2013.

[16] N. Sambo et al., “Modeling and Distributed Provisioning in 10-40-100 Gb/s Multirate Wavelength Switched Optical Networks,” IEEE/OSA Journal of Lightwave Technology, vol. 29, pp. 1248-1257, 2011.

[17] N. Sambo et al, “Lightpath establishment assisted by off-line QoT estimation in transparent optical networks,” IEEE Journal of Optical Communications and Networking, vol. 2, pp. 928-937, 2010.

[18] D. Zibar, “Machine Learning Techniques in Optical Communication,” in Proc. ECOC, 2015.

[19] J. Tapolcai, P.-H. Ho, L. Rónyai, and B. Wu, “Network-wide local unambiguous failure localization (NWL-UFL) via monitoring trails,” IEEE/ACM Transactions on Networking, vol. 20, pp. 1762-1773, 2012.

[20] H. Zeng, C. Huang, and A. Vukovic, “A novel fault detection and localization scheme for mesh all-optical networks based on monitoring-cycles,” Photonic Network Communications, vol. 11, pp. 277-286, 2006.

[21] J. Tapolcai et al., “Neighborhood Failure Localization in All-Optical Networks via Monitoring Trails,” in IEEE/ACM Transactions on Networking, vol. 23, pp. 1719-1728, 2015.

[22] M. Ruiz et al, “Service-triggered failure identification/ localization through monitoring of multiple parameters,” in Proc. ECOC, 2016.

[23] Ch. Bishop, *Pattern Recognition and Machine Learning*, Springer-Verlag, 2006.

[24] A. P. Vela, M. Ruiz, L. Velasco, “Distributing Data Analytics for Efficient Multiple Traffic Anomalies Detection,” Elsevier Computer Communications, vol. 107, pp. 1-12, 2017.

[25] M. Priestley, *Spectral Analysis and Time Series*, Academic Press, 1991.