Power Excursion Mitigation for Flexgrid Defragmentation with Machine Learning

Yishen Huang, Patricia B. Cho, Payman Samadi, and Keren Bergman

***Abstract*—Flexgrid optical networking relies on spectrum defragmentation to groom channel wavelengths and improve spectral efficiency. Various defragmentation methods—hop, make-before-break, and sweep—interact with the power dynamics of erbium-doped fiber amplifiers (EDFA) differently and result in undesired power excursions that exacerbate the post-EDFA power variance. We present a machine learning engine that characterizes the channel dependence of power excursions from historical data. We further demonstrate that post-EDFA power variance during hop, make-before-break, and sweep defragmentation methods can be greatly mitigated by the trained machine learning engine with automated and expedited power adjustment predictions.**

***Index Terms*—erbium-doped fiber optical amplifiers; flexgrid network; optical power excursion; wavelength division multiplexing.**

I. Introduction

The rising prevalence in dynamic workloads such as content distribution, cloud computing, and mobile connectivity places increasing demands on the functionalities and fault resilience of optical networking. To accommodate the diverse needs in data bandwidth and quality of service (QoS) from varying workloads, flexgrid optical networking [1] has been proposed to improve the agility and efficiency of wavelength-division multiplexing (WDM) networks. By implementing channels with variable allocated bandwidths, flexgrid networks support workloads at different data rates, modulation formats, and QoS requirements. Nevertheless, flexgrid networking faces numerous stringent challenges when designing for dynamic network scenarios and rapidly changing traffic demands. First, the arrival and departure of channels with different bandwidths result in spectrum fragmentation that increases the blocking probability of the network. As a result, defragmentation is performed to reassign channel wavelengths; thus, the groomed spectrum would then be able to accommodate future channel provisioning. Second, broadband optical amplifiers such as the erbium-doped fiber amplifiers (EDFA) express wavelength-dependent power excursions when channels are dynamically added or dropped [2]. Undesired power excursions are defined in [3] as the ones that increase the post-amplifier channel power variance. EDFA power excursions may greatly impact flexgrid networks because of the implementation of super-channels, which occupy multiple contiguous channel bandwidths and induce greater power changes to the EDFAs than individual fixgrid channels. As demonstrated in [4], a change in a super-channel's spectral location can trigger undesired power excursions. Defragmentation, if uncompensated, can increase post-EDFA power variance both during and after its process. In [5], we first proposed and implemented a machine learning (ML) engine that performs power pre-adjustments to reduce the post-EDFA power variance for three main defragmentation methods. In this paper, we extend the work in [5] with (1) additional analysis on each defragmentation method's mechanism that triggers power excursion, (2) further evaluation of the ML engine that provides end-to-end mitigation of undesired power excursions, and (3) scalability study beyond the experimental validation of the ML engine.

The paper is organized as follows: in Section II, we discuss the relevant background for power excursion mechanisms during the three main defragmentation methods as well as the working principles of the ML approach. Section III introduces the methodology of the ML engine through its design philosophy and logical workflow. Section IV presents the experimental implementation of the ML engine and its effective mitigation of post-EDFA power variance. Section V describes the scalability analysis of the approach. We discuss our conclusions in Section VI.

II. Background

A. EDFA Power Excursion

The interaction between the non-flat gain tilt and automatic gain control (AGC) of modern EDFA systems results in the steady-state power excursions of WDM channels [6]. A channel with high gain would increase the mean gain measured by the EDFA system, which triggers AGC to reduce the gain on all channels. This response leads to the high-gain channel effectively stealing power from lower-gain channels [7]. Conversely, adding a low-gain channel feeds power to higher-gain channels [6]. As a result, steady-state power excursions tend to increase the discrepancy of amplified channel power levels. In fixgrid

Manuscript received July 3, 2017; revised October 11, 2017; accepted October 20, 2017; published 0, 0000 (Doc. ID 301511).

The authors are with the Department of Electrical Engineering, Columbia University, New York, New York 10027, USA (e-mail: y.huang@columbia.edu).

https://doi.org/10.1364/JOCN.99.099999

networking, excursions of channel power levels up to 2 dB have been demonstrated experimentally in as few as three cascaded EDFAs through haphazard channel additions [6].

A number of approaches have been proposed to address the channel dependent excursion challenge in fixgrid networking. In [7], channel power levels are pre-emphasized to compensate for the excursion in gains. In [6], a fast tunable laser rapidly switches between two channels of high and low gains to equalize the mean gain measured by the amplifier AGC. In [8], the Raman/EDFA hybrid amplifier's pumping power is adjusted to reduce the power transient variations and steady-state excursions. These approaches rely on deterministic models of the specific systems or specialized hardware and therefore offer limited transferability to other network configurations. Data-analytics-based methods, as demonstrated in [4,9], offer a more flexible solution by learning an EDFA's gain response with historical data and infer optimized channel provisioning to improve OSNR and post-EDFA power discrepancy. In particular, it is demonstrated in [4] that trained machine learning models achieve high accuracy in predicting the optimal location to add or drop a channel and minimizing power discrepancy due to excursion. Nevertheless, these methods do not address the dynamic process of channel defragmentation, in which a flexgrid channel may interact with various parts of the EDFA gain spectrum.

B. Defragmentation Methods

Recent studies [10,11] have shown impressive models in network routing and spectral allocation (RSA), which implement one or more wavelength defragmentation methods. These models coordinate the optimal allocation of routing and assignment of channel wavelengths to enable efficient flexgrid networking, while considering their respective impact on data traffic. Three main defragmentation methods—hop [12], make-before-break (MbB) [13], and sweep [14]—have been demonstrated to effectively re-arrange the spectral usage in a flexgrid system. In hop, a new channel is established while dropping the original channel simultaneously. In MbB, a new channel is created, allowing the traffic to be re-established before dropping the original channel. In sweep, the channel wavelength is shifted gradually at the spectral granularity of the equipment without disruption to the traffic. Evidently, the spectral usage and the total number of channels present at a given instance differ for each method, inducing a unique power impact on the EDFAs. Hop can be treated as a single-step process in which a channel relocates instantly; the new channel location may experience a different EDFA gain and therefore trigger EDFA power excursions under AGC. MbB is a two-step process, in which a new channel co-exists with the original channel for a short time; the change in both the number and locations of channels interact with the EDFA gain spectrum and may result in power excursions in both steps. Sweep is a multistep process as the channel shifts gradually across the spectrum, during which the channel may experience a varying range of EDFA gain and consequently result in the amplifier power excursions throughout the process. We demonstrate that the ML engine is capable of alleviating post-EDFA power discrepancy for all three defragmentation methods, showcasing the wide applicability of our proposed design.

C. Machine Learning Enabled System Optimization

We propose an ML-enabled methodology that allows the flexgrid system to dynamically minimize post-EDFA power discrepancy by training on historical information on channel usage and power levels. ML models and statistical methods are particularly promising to optimize performance of complex systems that are difficult to describe with analytical models and enable these systems to autonomously monitor, optimize, and adapt from their historical operation performance [15]. In contrast with conventional feedback-based controls relying on iterative search and re-measurements [16], the ML-enabled methodology statistically characterizes the system in the training phase in order to make predictions on optimized system settings. While the training phase of the ML approach may be considered as an overhead, this approach provides more expedited operations than feedback-based methods during operation. Tailored ML models, in combination with a generalized and automated workflow, can provide single-step optimization of channel power discrepancy without re-measurements and iteration search, which we demonstrate in the following sections.

III. Methodology

In this section, we describe the design philosophy and logical workflow of the methodology and present the implementation of the methodology as a lightweight control plane application for flexgrid networks with cascaded EDFAs.

A. Machine Learning Models

To accurately predict power pre-adjustments throughout the defragmentation process, it is important to determine two factors: 1) to what extent a channel's spectral location exacerbates the post-EDFA discrepancy, and 2) whether to adjust a channel's power prior to the EDFAs *up* or *down* in order to minimize the post-EDFA discrepancy. We denote the first factor as the *magnitude* of impact and the second factor as the *correlation* of impact in the following sections. Because channels experiencing high and low gain levels can both trigger undesired excursions, we implement two low-complexity ML models to simultaneously learn the magnitude and correlation of a channel's impact on the post-EDFA power level discrepancy.

Ridge regression (RR) is performed to examine to what extent a specific channel contributes to the post-EDFA power discrepancy. RR determines a set of weights, w_{RR}, from which the variance in post-EDFA power levels, y_{var}, is predicted as a weighted linear combination of channel usage:

$$y_{\text{var}} = xw_{\text{RR}}, \tag{1}$$

where x is an $1 \times (n+1)$ array indicating the ON/OFF channel states, as 1 or 0, respectively, of an n-channel system. The one additional dimension is used for the learned bias. w_{RR} has shape $(n+1) \times 1$ and is determined by

$$w_{\text{RR}} = \arg_w^{\min}(\|Y - Xw\|^2 + \lambda\|w\|^2), \tag{2}$$

$$w_{\text{RR}} = (\lambda I + X^T X)^{-1} X^T Y, \tag{3}$$

where X is the set of training data for channel ON/OFF states, consisting of m data points arranged in $m \times (n+1)$, and Y is the set of training data for post-EDFA power level variances arranged in $m \times 1$. I is the identity matrix, and λ is the complexity parameter that encourages a small distribution amongst the dimensions of the learned weight. It is used to avoid heavily attributing the cause of power discrepancy to a few specific channels and thus preventing overfitting of the training data. Through cross-validation λ is set to 2.8 to achieve best prediction accuracy through repeated training across subsets of the training data. Figure 1 shows the learned RR weights corresponding to each of the 24 channels in the experiment system (more details in Section IV). The RR model is trained with 800 data points with randomized channel usage. In [4], the prediction accuracy with respect to the training set size is documented for a similar system, showing marginal improvements of prediction accuracy beyond more than 600 randomized training data points, although this number is expected to increase with increasing system size. Positive and negative weights indicate that a channel would increase or decrease, respectively, the post-EDFA power discrepancy when turned ON. The magnitude of each weight shows relatively each channel's contribution to the increase or decrease of the power discrepancy.

A rise in a channel's pre-EDFA power may increase (correlation $= +1$) or decrease (correlation $= -1$) the post-EDFA power discrepancy, which is conditional upon the other ON channels in the system. Hence we perform logistic regression (LR) to determine whether the specific channel's pre-EDFA power level needs to be increased or decreased to reduce the power variance given the channel usage. During training, LR constructs a distribution $P(s_{\text{ch}}|\{x^-\})$, where s_{ch} is the correlation specific to a channel ch and the set $\{x^-\}$ indicates the ON/OFF states of all other channels in the same light path. We treat s_{ch} in the form of a sigmoid function, whose sign conveniently reflects the correlation factor:

$$s_{\text{ch}} \sim \sigma(xw_{\text{LR}}^{\text{ch}}), \tag{4}$$

where $w_{\text{LR}}^{\text{ch}}$ is a channel-specific set of weights that is learned by LR during training. Because the derivatives of a sigmoid function are well-defined, optimization methods such as gradient descent or Newton method can be used to determine a set of weights that achieve the desired training accuracy. $w_{\text{LR}}^{\text{ch}}$ is channel-specific because it captures how a channel's power level compares with other ON channels in the system. Note that x labels an ON channel as 1 and an OFF channel as 0, which conveniently counts the contribution of ON channels only. Figure 2 illustrates the trained LR weights based on the experiment system (more details in Section IV).

The two ML models learn from the same corpus of data consisting of the ON/OFF states of all channels in the system, and the variance of their post-EDFA power levels—this ensures the two models can operate in parallel, expediting the system training process. RR has a training computation complexity of $O(N^2M)$, where N is the number of predictor values (number of channels + 1) and M is the number of training samples. The complexity of LR training may vary with the approaches to solve the optimization problem. Using gradient descent, each step has complexity $O(N)$, but the overall execution time depends on the number of iterations and stopping conditions. Both models have a prediction computation complexity of $O(N)$. In addition to the efficient training and prediction processes, the ML model complexities have no dependence on the number of EDFAs in the system and therefore are capable of scaling up with increasing the number of amplifiers.

B. Methodology Workflow

Figure 3 illustrates the logical workflow of the ML engine in training and operating on a flexgrid system during

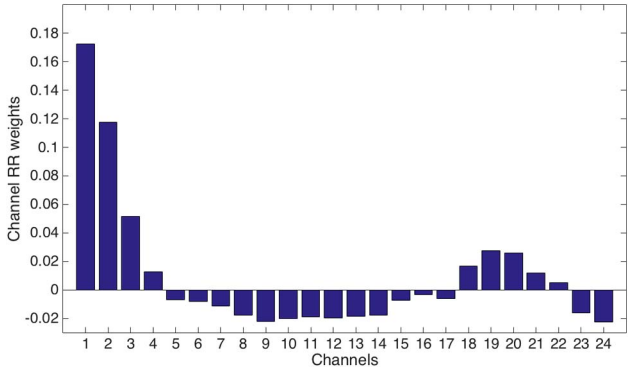


Fig. 1. Values of the learned RR weights corresponding to the 24 channels of the experiment system.

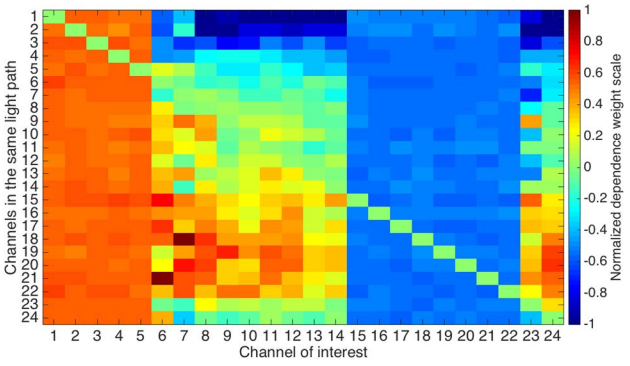
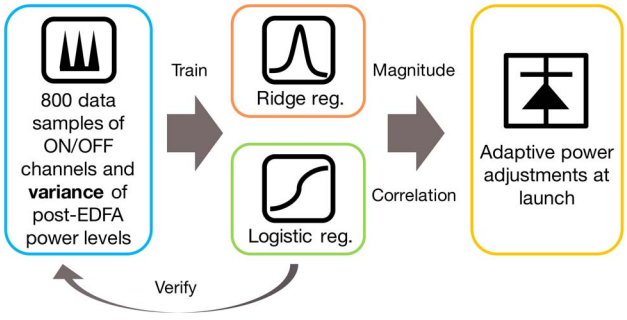


Fig. 2. Values of the learned channel-specific LR weights corresponding to the 24 channels of the experiment system.

the defragmentation process. We present this workflow to be applicable to different defragmentation methods and discuss the detailed implementations in the following section. The ML engine, including the RR and LR models, is trained with historical channel ON/OFF states and post-EDFA power discrepancy. The same corpus of data is used to train both ML models in parallel, whose results are stored as magnitude and correlation metrics of impact, respectively. These two metrics together determine 1) whether a channel will adversely trigger EDFA excursions to increase the power discrepancy, and if so, 2) how to adjust its pre-EDFA power to reduce the adverse effect. Once trained, the ML engine can make a single-step prediction to adjust the power of the relocated channel by monitoring the ON/OFF channel usage alone, thus eliminating the need for iterative power tuning processes that require numerous measurements of post-EDFA power. Because the training process is independent from the defragmentation methods, the ML engine is expected to function on the same system when different defragmentation methods are applied. If the system setup or equipment changes, the ML engine can be conveniently re-trained with operational channel ON/OFF states and post-EDFA power levels.

C. Implementation for Defragmentation Processes

The trained RR and LR models yield two sets of weights, w_{RR} and $w_{LR}^{\{ch\}}$, for the set of channels {ch} in the system directly from channel ON/OFF states and variance of post-EDFA power levels. These two metrics summarize each channel's effect on the power discrepancy, and its power level relative to other ON channels in the system. Both the training data and training process are agnostic to the defragmentation methods implemented and therefore can be transferred flexibly among them. For each defragmentation step, the RR weights are used to predict whether the new channel arrangement would result in a higher post-EDFA power variance than the old arrangement. If the power variance is predicted to increase, the ML engine then uses the LR weights to determine whether the new channel's power is too high or too low, compared with the other ON channels currently in the system, and perform a compensating adjustment to the new channel power prior to EDFAs.



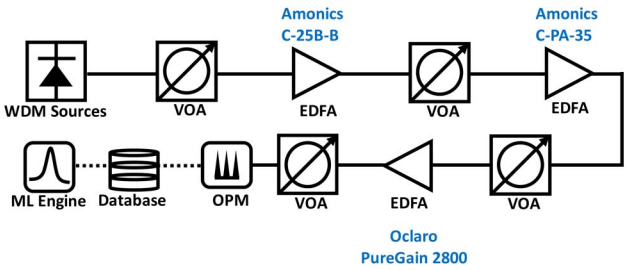
F3:1 F3:2 F3:3 Fig. 3. Workflow schematic of the ML engine showing training of RR and LR models, whose results are used to determine power adjustments.

Hop can be treated as a single-step process in which a new channel is added simultaneously to a channel taken down. The power adjustment is performed by the ML engine once the new channel location is determined. MbB is treated as a two-step process, including an intermediate step when both the new and old channels co-exist. Hence, the ML engine performs the power adjustment process *twice* to the new channel—once when the system contains both the old and the new channel and again after the old channel is taken down. In sweep, a channel's spectral location is continuously changed; therefore, for every step the channel is shifted, the ML engine determines a power adjustment, if necessary, given the channel's new location and the overall spectrum usage at that instance during the defragmentation. For all three implementations, the ML engine only adjusts the power levels when necessary for the newly provisioned channels, thus minimizing the interference on other channels in the system.

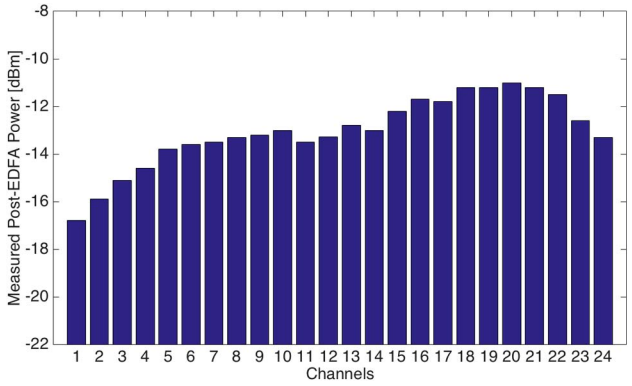
IV. Experiment Implementation

A. Experiment Setup

A total of three C-band EDFAs are cascaded together, in the arrangement shown in Fig. 4, to concatenate their power spectra over 24 WDM channels from ITU-T grid 194.40 THz to 192.10 THz with 100 GHz spacing, launched by 24 Thorlabs Pro8 dense WDM (DWDM) distributed feedback (DFB) laser modules. The power of individual lasers at channel wavelengths are tunable between 7 dBm and 13 dBm at 0.1 dBm steps. Note that, in this work, the channel launch power levels are adjusted by the lasers, but they can also be controlled by other means at channel granularity, such as using a wavelength selective switch. The EDFAs cascaded are of different brands and models to emulate a more complex combination of gain tilts. Wideband variable optical attenuators (VOA) are used to simulate fiber propagation loss and match the per-span average amplifier gain of around 15 dB per channel. A fourth VOA with −25 dB attenuation per channel is used to reduce the optical power due to the input power limitation of the optical performance monitor (OPM), which is used to record channel power levels after the EDFA-VOA spans and communicate with the computer system implementing the database and ML engine. The database records channel usage and power variance data, and the



F4:1 F4:2 Fig. 4. Schematic of experiment setup consists of three cascaded EDFAs.



F5:1 F5:2 Fig. 5. Discrepant post-EDFA power levels from 24 channels launched at uniform power.

ML engine trains on the data collected and controls the WDM sources with computed channel power adjustments. Figure 5 shows the widely discrepant channel power levels measured by the OPM when all channels are launched at a uniform power of 10 dBm.

We can deduce a correlation between Fig. 1 and Fig. 5—channels with post-EDFA power much higher or lower than the mean both contribute positively to the post-EDFA power variance, while channels with post-EDFA power near the mean contribute negatively to the variance. Information about the post-EDFA power spectrum is also captured by the LR model, shown in Fig. 2, whose values range from −1 to 1 and indicate how the power levels of other channels compare with a specific channel. Hence, the dot product between $w_{\mathrm{LR}}^{\mathrm{ch}}$ and the 24-bit ON/OFF vector gives an estimate on whether channel ch power is high or low relative to the other ON channels. A power adjustment, ΔP_{ch}, can then be determined as

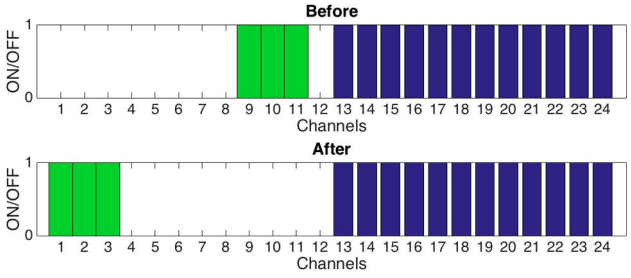
$$\Delta P_{\mathrm{ch}} = \mathrm{sgn}(\sigma(x w_{\mathrm{LR}}^{\mathrm{ch}}) - 0.5) \cdot \frac{w_{\mathrm{RR}}(\mathrm{ch})}{\max(|w_{\mathrm{RR}}|)} \cdot P_{\mathrm{step}}, \qquad (5)$$

where P_{step} is a power tuning unit that we predefine as 3 dBm, and ΔP_{ch} is scaled by the ratio between the channel's RR weight and the maximum RR weight among all channels. It is also evident that the LR model was unable to capture finer details for Ch. 1–5 and 15–22, which are, respectively, at the low and high ends of the gain levels. We expect that finer details can be extracted by the LR model if more system data are available, specifically with cases containing only Ch. 1–5 or Ch. 15–22 as ON. In our demonstration, the lack of details in the extreme low and high ends of the spectrum by the LR model does not significantly impact the ML engine's performance because the current set of LR weights still results in positive adjustments for Ch. 1–5 and negative adjustments for Ch. 15–22.

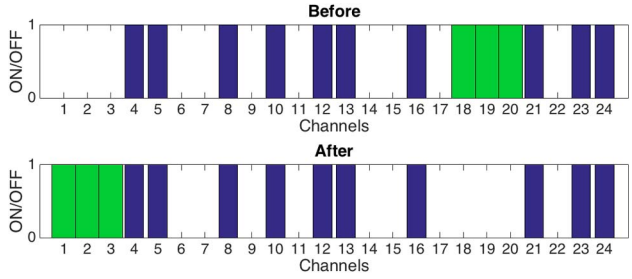
B. Hop Defragmentation

We implement a wavelength grooming for a super-channel in two experiments using the hop defragmentation method. The super-channel is emulated by assigning three spectrally adjacent channels into a single channel entity that is relocated together, with each subchannel's power adjusted individually by the ML engine. In the first experiment, as illustrated in Fig. 6, a super-channel is relocated from Ch. 9–11 to Ch. 1–3, while Ch. 13–24 are in use. This is a representative case because the new location of the super-channel promotes a significant increase in the post-EDFA power discrepancy, according to Fig. 1. The hop method is a single-step procedure in our implementation, and Fig. 8(a) shows immediate change in the variance of post-EDFA channel power levels. We show that, by implementing the ML engine to perform a single-step power adjustment on the relocated super-channel, the post-EDFA power variance is improved by 63%, without re-measuring channel power levels during the defragmentation.

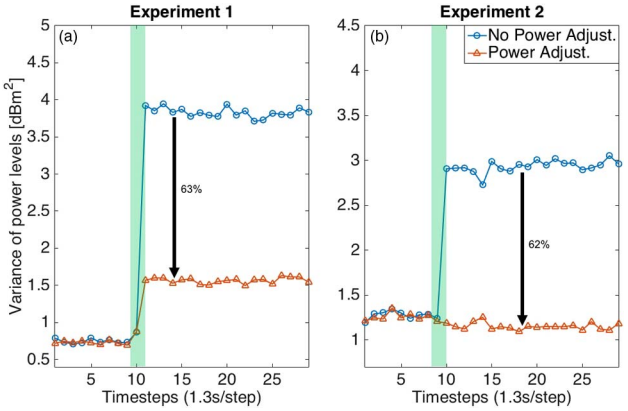
In the second experiment, we evaluate a more complex spectrum usage, as shown in Fig. 7, in which a super-channel is relocated from Ch. 18–20 to Ch. 1–3. This defragmentation procedure opens up an available band at Ch. 17–20, which can accommodate a super-channel of four channel spacing that is previously impossible. In addition, the original location of the super-channel experiences the higher end of the gain tilt, while the new location of the super-channel experiences the lower end of the gain tilt, which examines how the ML engine performs with channels shifting between extreme ends of the gain spectrum. Figure 8(b) illustrates that with power adjustments by the ML engine, the post-EDFA variance is improved by 62%, effectively maintaining the same post-EDFA power discrepancy before and after the defragmentation.



F6:1 F6:2 F6:3 Fig. 6. Illustration of the first defragmentation experiment (relocated super-channel is shown in green; other ON channels are shown in blue).



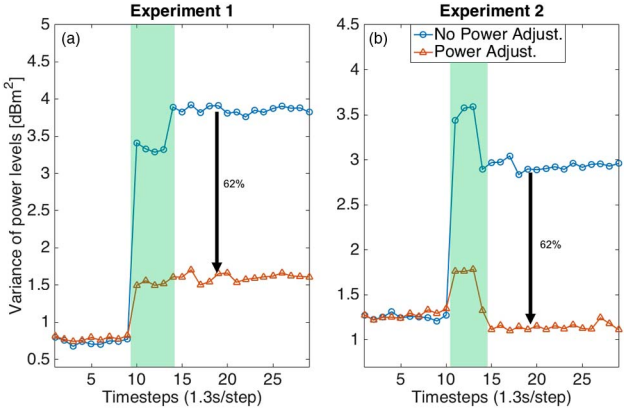
F7:1 F7:2 F7:3 Fig. 7. Illustration of the second defragmentation experiment for Hop and MbB (relocated super-channel is shown in green; other ON channels are shown in blue).



F8:1 F8:2 F8:3 Fig. 8. Comparison of post-EDFA power variance with and without ML-enabled power adjustments in two experiments of hop defragmentation, the duration of which is shaded in green.

C. MbB Defragmentation

We repeat the two experiments with the MbB defragmentation method, which is treated as a two-step process: a new super-channel is turned on, allowing the network traffic to transfer over before turning off the original super-channel. The intermediate stage of MbB captures when both the original and new channels co-exist in the system. The ML engine performs two power adjustments on the new super-channel based on the intermediate and the final channel usage. As shown in Figs. 9(a) and 9(b), we observe improvements in post-EDFA power variance both at the intermediate stage and after the defragmentation for both experiments. In both cases, the final post-EDFA power variances after the MbB defragmentation improve by 62% and match well with the hop defragmentation method, showing consistency of the ML engine across different defragmentation methods.



F9:1 F9:2 F9:3 Fig. 9. Comparison of post-EDFA power variance with and without ML-enabled power adjustments in two experiments of MbB defragmentation, the duration of which is shaded in green.

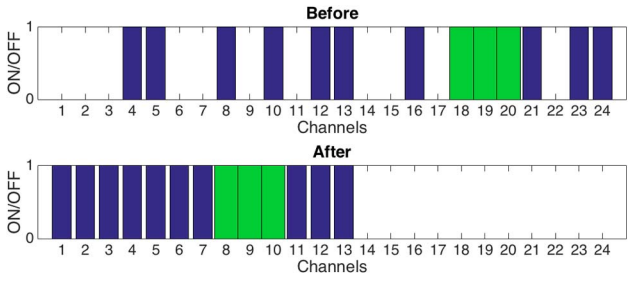
D. Sweep Defragmentation

The two defragmentation experiments are again repeated with the sweep defragmentation method, in which the super-channel is shifted at the spectral granularity of the system until it reaches its new spectral location. In the first case, a super-channel is swept from Ch. 9–11 to Ch. 1–3, while Ch. 13–24 are in use. The super-channel is moved directly because the continuous spectrum between the start and end locations is available. During the experiment, the central frequency of super-channel is shifted every 1.3 s at 100 GHz steps, limited by the measurement sampling frequency of the OPM and the spectral granularity of the system. The ML engine performs a power adjustment on the super-channel at every step based on the spectrum usage at that instance. Figure 11(a) shows that, with ML-enabled power adjustments, the change in post-EDFA power variance is greatly suppressed and optimized. After the defragmentation completes, the ML engine helps to achieve a 75% reduction in post-EDFA power variance. The greater improvement over hop and MbB is due to the multistep process of sweep, which allows the ML engine to perform more adjustments throughout the process.

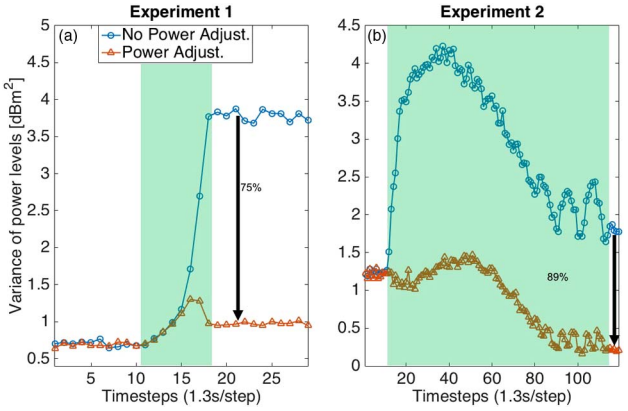
In the second experiment, in which a super-channel is relocated from Ch. 18–20 to Ch. 1–3, existing channels prevent the super-channel from being directly swept across the spectrum. Hence, we perform a sequential sweep of every channel in the system to the lower wavelength, effectively maximizing the available continuous bandwidth of the spectrum, as shown in Fig. 10. We illustrate the evolution of the post-EDFA power variance throughout this process, with and without ML-enabled power adjustments in Fig. 11(b), and show that the ML engine drastically suppresses the change in power variance. At the end of the defragmentation, the spectrum variance is improved by 89%, resulting in a set of much less dispersed post-EDFA channel power levels.

V. Scalability of The Method

The ML engine presented effectively reduces the post-EDFA power discrepancy during and after the defragmentation process and is applicable to all three main defragmentation methods. Here we discuss the potential to scale the ML engine for larger and more complex



F10:1 F10:2 F10:3 Fig. 10. Illustration of the second defragmentation experiment for sweep (relocated super-channel is shown in green; other ON channels are shown in blue).



F11:1 F11:2 F11:3 Fig. 11. Comparison of post-EDFA power variance with and without ML-enabled power adjustments in two experiments of sweep defragmentation, the duration of which is shaded in green.

networks. One limitation of using variance as the optimized metric of the flexgrid optical network is that the discrepancy amongst the channel power levels, instead of their mean, is captured. This may overlook crucial aspects such as light path power penalty, as the ML engine promotes equalizing rather than maximizing channel power levels. A potential solution to this concern can be to set appropriate bounds within which the ML engine can modify the channel power, thus avoiding adjusting a channel's power too low or too high. Another solution can consider a joint metric to achieve both high mean and low variance among post-EDFA power levels.

For optical networks with a variety of channel injection and termination points, such as a mesh network, distributed implementation of the ML engine is possible. Individual ML engines can be trained on each network edge and focus on power discrepancy optimization of the specific edges. Power adjustments can be performed at the network nodes where channels enter the edge. The low-complexity and small-footprint operations of the RR and LR models, in addition to their memory efficient weights, encourage parallel and distributed operations of multiple ML engines in a scaled-up and complex network.

VI. Conclusion

Maintaining channel power stability during fast changing spectrum utilizations is crucial to ensuring the QoS of flexgrid optical networking. We introduce an ML engine to preserve channel power consistency during the defragmentation process of a flexgrid network experiencing EDFA power excursions. The proposed ML engine employees low-complexity ML models in a fully automated workflow, which extracts EDFA power dynamics and performs power adjustments without iterative power measurements. Experimentally, we demonstrate the effectiveness of the ML engine in diverse spectral usage scenarios and show consistent performance and applicability among three main defragmentation methods: hop, MbB, and sweep. In addition, we explain possible improvements and scalable implementations of the ML engine for larger, more complex networks. This work affirms the efficacy of ML-based approaches to extract and predict system dynamics and further drive performance optimizations for optical networking applications.

Acknowledgment

This project is supported by CIAN NSF ERC under grant EEC-0812072 and NSF CNS-1423105 and Department of Energy (DoE) Advanced Scientific Computing Research (ASCR) under Turbo Project (DE-SC0015867). The authors would also like to thank AT&T Labs for its generous donation of the equipment.

References

[1] M. Jinno, H. Takara, B. Kozicki, Y. Tsukishima, Y. Sone, and S. Matsuoka, "Spectrum-efficient and scalable elastic optical path network: architecture, benefits, and enabling technologies," *IEEE Commun. Mag.*, vol. 47, no. 11, pp. 66–73, 2009.

[2] D. C. Kilper, M. Bhopalwala, H. Rastegarfar, and W. Mo, "Optical power dynamics in wavelength layer software defined networking," in *Advanced Photonics*, 2015.

[3] Y. Huang, W. Samoud, C. L. Gutterman, C. Ware, M. Lourdiane, G. Zussman, P. Samadi, and K. Bergman, "A machine learning approach for dynamic optical channel add/drop strategies that minimize EDFA power excursions," in *European Conf. on Optical Communication*, 2016.

[4] Y. Huang, C. L. Gutterman, P. Samadi, P. B. Cho, W. Samoud, C. Ware, M. Lourdiane, G. Zussman, and K. Bergman, "Dynamic mitigation of EDFA power excursions with machine learning," *Opt. Express*, vol. 25, no. 4, pp. 2245–2258, 2017.

[5] Y. Huang, P. B. Cho, P. Samadi, and K. Bergman, "Dynamic power pre-adjustments with machine learning that mitigate EDFA excursions during defragmentation," in *Optical Fiber Communication Conf.*, 2017, paper Th1J.2.

[6] A. S. Ahsan, C. Browning, M. S. Wang, K. Bergman, D. C. Kilper, and L. P. Barry, "Excursion-free dynamic wavelength switching in amplified optical networks," *J. Opt. Commun. Netw.*, vol. 7, no. 9, pp. 898–905, 2015.

[7] P. J. Lin, "Reducing optical power variation in amplified optical network," in *Int. Conf. on Communication Technology*, 2003.

[8] D. A. Mongardien, S. Borne, C. Martinelli, C. Simonneau, and D. Bayart, "Managing channels add/drop in flexible networks based on hybrid raman/Erbium amplified spans," in *European Conf. on Optical Communication*, 2006.

[9] U. Moura, M. Garrich, H. Carvalho, M. Svolenski, A. Andrade, F. Margarido, A. C. Cesar, E. Conforti, and J. Oliveira, "SDN-enabled EDFA gain adjustment cognitive methodology for dynamic optical networks," in *European Conf. on Optical Communication*, 2015.

[10] R. Wang and B. Mukherjee, "Provisioning in elastic optical networks with non-disruptive defragmentation," *J. Lightwave Technol.*, vol. 31, no. 15, pp. 2491–2500, Aug. 2013.

[11] M. Zhang, C. You, H. Jiang, and Z. Zhu, "Dynamic and adaptive bandwidth defragmentation in spectrum-sliced elastic optical networks with time-varying traffic," *J. Lightwave Technol.*, vol. 32, no. 5, pp. 1014–1023, 2014.

[12] M. Zhang, Y. Yin, R. Proietti, Z. Zhu, and S. J. B. Yoo, "Spectrum defragmentation algorithms for elastic optical

networks using hitless spectrum retuning techniques," in *Optical Fiber Communication Conf.*, 2013, paper OW3A.4.

[13] T. Takagi, H. Hasegawa, K. Sato, Y. Sone, A. Hirano, and M. Jinno, "Disruption minimized spectrum defragmentation in elastic optical path networks that adopt distance adaptive modulation," in *European Conf. on Optical Communication*, 2011.

[14] F. Cugini, F. Paolucci, G. Meloni, G. Berrettini, M. Secondini, F. Fresi, N. Sambo, L. Potì, and P. Castoldi, "Push-pull defragmentation without traffic disruption in flexible grid optical networks," *J. Lightwave Technol.*, vol. 31, no. 1, pp. 125–133, 2013.

[15] I. de Miguel, R. J. Durán, T. Jiménez, N. Fernández, J. C. Aguado, R. M. Lorenzo, A. Caballero, I. T. Monroy, Y. Ye, A. Tymecki, I. Tomkos, M. Angelou, D. Klonidis, A. Francescon, D. Siracusa, and E. Salvadori, "Cognitive dynamic optical networks," *J. Opt. Commun. Netw.*, vol. 5, no. 10, pp. A107–A118, 2013.

[16] B. Birand, H. Wang, K. Bergman, D. Kilper, T. Nandagopal, and G. Zussman, "Real-time power control for dynamic optical networks—Algorithms and Experimentation," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 8, pp. 1615–1628, 2014.

Yishen Huang received a B.Sc. degree (honors with distinction) in engineering physics from Queen's University, Kingston, Canada, and an M.S. degree in electrical engineering from Columbia University, New York, USA. He previously worked on metro optical networking at Ciena Networks and performance optimization for distributed systems and applications at IBM. He is currently a Ph. D. candidate in electrical engineering at Columbia University advised by Prof. Keren Bergman. His Ph.D. research focuses on optimized system designs for optical networks and subsystems to enable next-generation optical interconnects.

Patricia B. Cho received a B.A. (thesis with honors) from Williams College in 2010 and is currently working toward her B.A. in astrophysics at Columbia University, New York, with an anticipated graduation in June 2018. In June and July of 2016, she participated in a research internship with Yishen Huang at the Lightwave Research Laboratory, Columbia University, under the advisement of Prof. Keren Bergman. Patricia is currently examining localized heating of companion stars in binary millisecond pulsar systems with Jules Halpern in the Columbia Astrophysics Lab.

Payman Samadi received the Ph.D. degree in photonics from McGill University, Montreal, QC, Canada, in 2012. He is currently a research scientist at the Lightwave Research Laboratory, Columbia University, New York, New York, USA. He was formerly with Optiwave Systems Inc., Canada as a product manager during 2011–2013. He has coauthored more than 40 articles in peer-reviewed conferences and journals. His current research interests include data center networks architecture, software-defined networking, application of optical interconnects for computing platforms, and big data analytics.

Keren Bergman (S'87–M'93–SM'07–F'09) received her B.S. degree from Bucknell University, Lewisburg, Pennsylvania, in 1988 and her M.S. and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, Massachusetts, in 1991 and 1994, respectively, all in electrical engineering. She is currently the Charles Batchelor Professor of Electrical Engineering, Columbia University, New York, where she also directs the Lightwave Research Laboratory. Her research programs involve optical interconnection networks for advanced computing systems, photonic packet switching, and nanophotonic networks on-chip. She is a fellow of the Optical Society of America (OSA) and a fellow of the Institute of Electrical and Electronic Engineers (IEEE).