

# BOW: First Real-World Demonstration of a Bayesian Optimization System for Wavelength Reconfiguration

Zhizhen Zhong<sup>†</sup>, Manya Ghobadi<sup>†</sup>, Maximilian Balandat<sup>\*</sup>, Sanjeevkumar Katti<sup>\*</sup>  
Abbas Kazerouni<sup>\*</sup>, Jonathan Leach<sup>\*</sup>, Mark McKillop<sup>\*</sup>, Ying Zhang<sup>\*</sup>

<sup>†</sup>Massachusetts Institute of Technology    <sup>\*</sup>Facebook

{zhizhenz, ghobadi}@mit.edu, {balandat, skatti, kazerouni, jleach, mmckillop, zhangying}@fb.com

**Abstract:** We demonstrate a practical Bayesian Optimization system for wavelength reconfiguration at Facebook backbone. Our system uses a firewall for safe deployment. It is open-source, compatible with any vendor, and achieves  $4.76\times$  faster wavelength reconfiguration. © 2021 The Author(s)

## 1. Introduction

Large-scale online service providers (such as Facebook, Google, Amazon, Microsoft) operate their own global long-haul optical backbones. These backbones carry over 10,000 optical wavelengths through millions of miles of fiber. To keep up with the increased traffic from users, these backbones continue to grow by continuously deploying new wavelengths [1]. However, deploying new wavelengths on a fiber that already carries live traffic in production optical backbones is non-trivial. This is because any wavelength reconfiguration system needs to control the complex relationship between amplifiers' gain, newly-added wavelengths, and wavelengths that are already traversing the fiber carrying live traffic [2]. Specifically, a sudden change to the set of wavelengths on a fiber, i.e., deploying or decommissioning wavelengths, could result in power fluctuations on each amplifier which, in turn, leads to packet loss/errors in the user traffic at the IP layer.

To quantitatively demonstrate the challenge of wavelength reconfiguration in real world, we shadow today's process of deploying new wavelengths at Facebook optical backbone using the state-of-the-art devices and algorithms between two major cities in Southeast Asia. Fig. 1 presents per-wavelength OSNR and BER over time on a 360-km production fiber with six cascaded amplifier sites. Initially, the fiber is carrying one wavelength. The figure shows that it takes  $\approx 11$  minutes (671 seconds) to add seven new wavelengths. We observe that each wavelength's power is slowly increased to its desired level, while inducing non-negligible power excursions on both new and existing wavelengths. As a result, it is common practice to add wavelengths in a serial manner to avoid power instabilities, thus increasing the entire time-span of the process. Even worse, this approach requires all amplifiers on the same fiber to be purchased from a single vendor so the amplifiers can work with the proprietary power control algorithm embedded in vendor devices. This requirement is undesirable because it limits each fiber segment into a single vendor and hinders innovation in the optical layer.

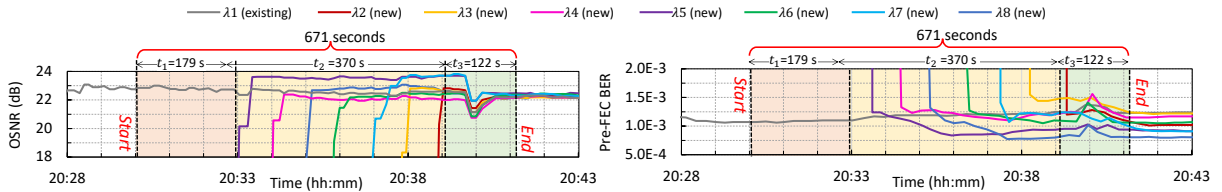


Fig. 1: In real-world production optical backbone, on a fiber with one existing wavelength ( $\lambda_1$ ), deploying seven new wavelengths ( $\lambda_2, \dots, \lambda_8$ ) takes 671 seconds using the state-of-the-art devices. This includes 179s control software's calculation ( $t_1$ ), 370s wavelength turn-up ( $t_2$ ), and 122s post-provisioning adjustments ( $t_3$ ).

## 2. A Firewall-based Bayesian Optimization System to Enable Multi-Vendor Amplifier Control

The core challenge in wavelength reconfiguration is how to control and optimize all cascaded amplifiers *jointly*. In this paper, we propose a firewall-based system for *in-situ* decision making without *a priori* training by leveraging Bayesian Optimization (BO).

**Proprietary amplifier control ossifies optical layer.** Today's systems use a three-step control loop for deploying new wavelengths [3]: (i) observe the status of the network by collecting performance data from devices; (ii) use a *proprietary control algorithm* to decide how to adjust the amplifiers; and (iii) push the changes to devices. This process is repeated until BER and OSNR converge to the desired targets. To move towards open white-box and multi-vendor backbone, prior proposals [4, 5] have focused on using optical-layer SDN and open interfaces for steps (i) and (iii). However, an important missing piece is step (ii) that is the *amplifier control algorithm*. This algorithm is not open-source since it requires proprietary data, such as amplifier gain profiles and noise figures, for its internal analytical model. As a result, this control algorithm is embedded inside ROADMs line cards without external accessibility. This prevents large operators, to scale their backbone by leasing existing optical networks equipped with other vendors devices or building new networks with different vendors. Inspired by recent efforts on open optical networks [6], we aim to fully open-source this process.

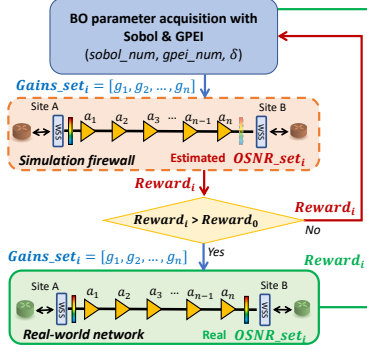


Fig. 2: Our BOW system uses both GNPY simulator and measurements from real network for amplifier gain control.

### Algorithm 1 Our Bayesian Optimization-based algorithm for BOW.

```

1: procedure AUTONOMOUS AMPLIFIER CONTROL WITH BO
  ▷ Input:  $A = \{a_1, a_2, \dots, a_n\}$ : Optical amplifier set on the fiber link;
  ▷ Input:  $sobol\_num$ : number of Sobol trials at the beginning of BO process (e.g., 10);
  ▷ Input:  $gpei\_num$ : number of GPEI trials for adaptive termination (e.g., 5);
  ▷ Output:  $Gains\_set^* = \{g_1^*, g_2^*, \dots, g_n^*\}$ : Selected target gain parameter for amplifiers;
2:  $OSNR\_set_0 = \text{ReadOSNR\_Real}(Dest\_WSS)$  ▷ Get initial per-wavelength OSNR set
3:  $reward_0 = \min(OSNR\_set_0) \times M + \text{sum}(OSNR\_set_0)$  ▷  $M$  is a large number (e.g.,  $10^4$ )
4: while  $i - sobol\_num < gpei\_num$  do ▷  $i$  is the index of BO trials
5:   if  $i < sobol\_num$  then
6:      $Gains\_set_i = \text{Sobol\_Acquisition}(i, reward_{i-1})$ ;
7:   else
8:      $Gains\_set_i = \text{GPEI\_Acquisition}(i, reward_{i-1})$ ;
9:    $OSNR\_set_i = \text{Simulator\_Firewall}(Gains\_set_i, A)$ ; ▷ Get OSNR from simulation
10:   $reward_i = \min(OSNR\_set_i) \times M + \text{sum}(OSNR\_set_i)$ ; ▷  $M$  is a large number (e.g.,  $10^4$ )
11:  if  $reward_i > reward_0$  then ▷ If the parameters do not deteriorate OSNR
12:     $OSNR\_set_i = \text{ReadOSNR\_Real}(Gains\_set_i, A)$ ; ▷ Get OSNR from real network
13:     $reward_i = \min(OSNR\_set_i) \times M + \text{sum}(OSNR\_set_i)$ ; ▷  $M$  is a large number (e.g.,  $10^4$ )
14:  if  $(reward_i - reward_{i-1}) / reward_{i-1} > \delta$  then
15:     $gpei\_num = gpei\_num + i - sobol\_num$  ▷ If reward improvement  $> \delta$ , update  $gpei\_num$ 
16:  return  $Gains\_set^* = \{g_1^*, g_2^*, \dots, g_n^*\}$  with the highest  $reward_i$  value

```

**BOW: Learning to control wavelengths powers with BO.** Finding an optimal control algorithm is challenging, because it requires to have a priori knowledge of the status of the network as well as faithful modeling of all optical components such as fibers and amplifiers. In real-world network environments, the states of these optical components are variable (e.g., fiber cut repairs impact span loss and weather conditions affect wavelength signal quality in aerial fibers), and are usually costly to collect (e.g., operators tend not to profile an amplifier’s gain curve and noise figure once it is carrying live traffic) or even proprietary. To address this challenge, we propose **Bayesian-Optimized Wavelengths (BOW)**, a system that leverage BO to adapt to various optical-layer conditions and learns *in-situ* to make optimal amplifier control decisions on-the-fly without *a priori* knowledge about network devices, hence free of vendor lock. BO is a global optimization method that leverages Bayesian learning to model expensive-to-evaluate *black-box* functions with efficient number of samples. Hence, BO is a perfect fit for the amplifier control problem without the need to pre-model or pre-collect optical-layer characteristics in real-world.

**Augment BO with simulation firewalls.** Fig. 2 shows the high-level design of our BOW system, and Algorithm 1 shows its detailed pseudo-code. We define the BO objective function as:  $Reward = \min(OSNR\_set) \times M + \text{sum}(OSNR\_set)$ , where  $M$  is a large number (i.e.,  $10^4$ ). The intuitions behind this formula is that we aim to optimize worst-case OSNR as the primary goal, and improve overall OSNR as the secondary goal (lines 3, 10, 13 in Algorithm 1). We use OSNR as our primary performance metric because it can be collected directly and quickly from the layer-0 ROADM line systems without the need to query transponders. Inside the BO engine, we use Sobol [7] for the first 10 trials to explore the problem space, then GPEI [8] to fit a Gaussian process and make predictions that optimize expected improvements (lines 5–8). BO trials terminate if the reward value is not improved by  $\delta$  during  $gpei\_num$  consecutive GPEI trials (lines 4, 14, 15). Importantly, a key challenge is that BO is a probabilistic exploration process. As a result, naively using BO cannot provide guarantees on “safe” parameters during BO trials. To avoid network disruptions that would result in service interruptions to billions of users at Facebook during BO’s exploration phase, we augment the conventional BO framework with a firewall. The intuition behind our augmentation is to place a “simulation firewall” module between the BO engine and the real-world network. The “firewall” module is based on GNPY’s open-source simulator [9] and is responsible for pre-evaluating the BO-generated parameters in real-time and estimating per-wavelength OSNR to decide if it is safe to deploy to real network or not (lines 9–13). Finally, BOW selects and returns the parameters with the largest reward value (lines 16).

### 3. Real-World Experimental Evaluation

To demonstrate the feasibility of our proposed BOW system, we use a real-world network at Facebook backbone in Southeast Asia with 3 ROADM sites, 14 amplifier sites, and over 900 km fiber, as shown in Fig. 3a (this is identical as the network we measure in Fig. 1). Our BO module is implemented with Ax [10] and BoTorch [11], and the simulator firewall is implemented based on a GNPY QoT simulator [9]. The BO module reads/writes parameters from/to devices through open-source FCR/SSH sockets [12]. Our code is publicly available at [13].

Using this setup, we deploy 2.1 Tbps IP capacity (seven wavelengths, each of them modulated to 300 Gbps at 56 GBaud with FEC threshold  $2.661 \times 10^{-2}$ ). Then, we disable the vendor-proprietary amplifier control module, and leverage our BOW system to control amplifiers during wavelength reconfiguration and show the results in Fig. 3b. The top figure shows OSNR and bottom one shows Pre-FEC BER. The results show that BOW can cut down the control software calculation time (period  $t_1$  in Fig. 1) to zero because BOW is an *in-situ* learning method that decides amplifier control parameters after wavelengths are added without a priori modeling, hence no need for pre-calculations. Also, BOW cuts the post-provisioning adjustment delay (period  $t_3$  in Fig. 1) by a factor of 1.39 because it is a sample-efficient learning approach that can learn the amplifier configurations with less number

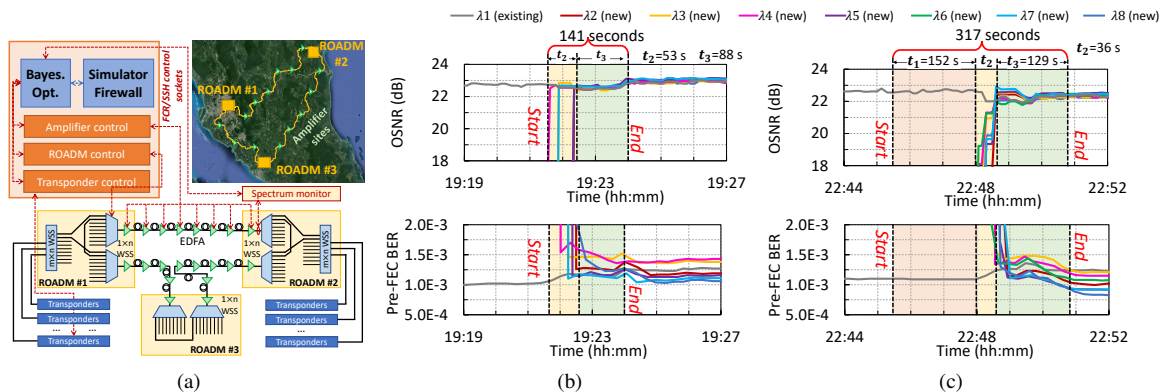


Fig. 3: (a) Real-world optical network setup and geographical information. (b) Wavelength reconfiguration with BOW. (c) Wavelength reconfiguration with “aggressive” state-of-the-art by adding all wavelength all-together.

of iterations, while state-of-the-art methods use analytical models and require more control loops, hence, taking longer time to converge. As a result, BOW reduces the overall wavelength reconfiguration time by  $4.76\times$  (from 671 seconds to 141 seconds).

To benchmark the impact of control loop algorithm, and as an extreme data point, we also test an “aggressive” version of the state-of-the-art method by adding all the wavelengths simultaneously, instead of serially. Note that our production operators will not adopt this aggressive approach since it runs the risk of dropping user-facing traffic. But we use this as a benchmark shown in Fig. 3c. The top figure shows OSNR and bottom one shows Pre-FEC BER. The results show that by “aggressively” adding all wavelength at the same time, only the  $t_2$  wavelength turn-up delay can be directly reduced. However,  $t_1$  and  $t_3$ , which are mostly related to amplifier control algorithm, remain almost the same as in Fig. 1. This experiment confirms the importance of developing new algorithmic systems for open and intelligent amplifier control so as to shorten the  $t_1$  and  $t_3$  delays.

Approach	Avg. OSNR	Min. OSNR	Avg. Pre-FEC BER	Max. Pre-FEC BER
State-of-the-art	22.27 dB	22.07 dB	$1.07\times 10^{-3}$	$1.56\times 10^{-3}$
BOW	22.99 dB	22.77 dB	$1.23\times 10^{-3}$	$1.52\times 10^{-3}$
Difference	+0.72 dB	+0.7 dB	$+1.6\times 10^{-4}$	$-4\times 10^{-5}$

Table 1: Reporting OSNR and pre-FEC BER in each experiment.

Table 1 shows OSNR and pre-FEC BER when new wavelengths are in steady state (after  $t_3$ ) for the both methods in our experiments. It shows that BOW performs wavelength reconfiguration with comparable performance with the state-of-the-art (0.7 dB higher in Min OSNR, and  $4\times 10^{-5}$  less in Max BER).

#### 4. Related Work and Conclusion

**Related work.** Open-source amplifier control is a crucial step towards realizing optical-layer SDNs [4]. Prior works proposed to leverage analytical models [3, 14], case-based reasoning [15], and multi-objective optimizations [16] for adaptive amplifier control. Others proposed to use ASE noise loading to simplify optical-layer operations [17], but they require extra devices such as ASE-compatible line systems to fully load the entire network at all times, with the cost of increased fiber nonlinear impairments and compromised system margin (3 dB) [18].

**Conclusion.** We demonstrate a novel learning-based system, called BOW, for autonomous amplifier control in long-haul open optical backbones. Our method is  $4.76\times$  faster than the state-of-the-art vendor-proprietary method with 0.7 dB higher in minimum OSNR, and  $4\times 10^{-5}$  lower in maximum BER, and is open-source.

**Acknowledgement.** We would like to thank Omar Baldonado, Massimo Tornatore, and Hongzi Mao for helpful discussions. This work was partly supported by ARPA-E ENLITENED PINE, DARPA FastNICs, NSF ASCENT-2023468, and SystemsThatLearn@CSAIL Ignite Grant.

#### References

1. Y. Xia, *et al.*, “A social network under social distancing: risk-driven backbone management during COVID-19 and beyond,” *USENIX NSDI*, 2021.
2. J. Yu, *et al.*, “Machine-learning-based EDFA gain estimation,” *J. Opt. Commun. Netw.*, vol. 13, pp. B83–B91, 2021.
3. K. B. Roberts, *et al.*, “Adjustment of control parameters of section of optical fiber network,” (2020). *US Patent* 10,784,980.
4. Y. Li and D. C. Kilper, “Optical physical layer SDN,” *J. Opt. Commun. Netw.*, vol. 10, pp. A110–A121, 2018.
5. M. Filer, *et al.*, “Multi-vendor experimental validation of an open source QoT estimator for optical networks,” *J. Light. Technol.*, vol. 36, 2018.
6. FSP 3000 open line system: Freedom from vendor lock-in, <https://www.adva.com/en/products/open-optical-transport/fsp-3000-open-line-system>.
7. A. B. Owen, “Quasi-monte carlo sampling,” *Monte Carlo Ray Tracing: Siggraph*, vol. 1, pp. 69–88, 2003.
8. D. R. Jones, *et al.*, “Efficient global optimization of expensive black-box functions,” *J. Glob. optimization*, vol. 13, pp. 455–492 (1998).
9. A. Ferrari, *et al.*, “GNPy: an open source application for physical layer aware open optical networks,” *J. Opt. Commun. Netw.*, 2020.
10. Facebook Open Source Adaptive Experimentation Platform, <https://ax.dev/>.
11. M. Balandat, *et al.*, “BoTorch: A framework for efficient monte-carlo bayesian optimization,” *NeurIPS*, 2020.
12. “FCR: Open source command runner for network devices,” Facebook Engineering, 2017, <https://github.com/facebookincubator/FCR>.
13. <http://bow.csail.mit.edu>.
14. C. A. Al Sayeed, *et al.*, “Spectrum controller systems and methods in optical networks,” (2017). *US Patent* 9,577,763.
15. U. Moura, *et al.*, “SDN-enabled EDFA gain adjustment cognitive methodology for dynamic optical networks,” *ECOC*, 2015.
16. E. de Andrade Barboza, *et al.*, “Adaptive control of optical amplifier operating point using VOA and multi-objective optimization,” *J. Lightw. Technol.*, 2019.
17. M. Filer, *et al.*, “Low-margin optical networking at cloud scale,” *J. Opt. Commun. Netw.*, vol. 11, no. 10, pp. C94–C108, 2019.
18. Y. Pointurier, “Design of low-margin optical networks,” *J. Opt. Commun. Netw.* vol. 9, no. 1, pp. A9–A17, 2017.