POLITECNICO DI TORINO
Repository ISTITUZIONALE

Fiber-Agnostic Machine Learning-Based Raman Amplifier Models

Original
Fiber-Agnostic Machine Learning-Based Raman Amplifier Models / de Moura, U.C., Zibar, D., ROSA BRUSIN, A.M., Carena, A., Da Ros, F.. - In: JOURNAL OF LIGHTWAVE TECHNOLOGY. - ISSN 0733-8724. - STAMPA. - 41:1(2023), pp. 83-95. [10.1109/JLT.2022.3210769]

Availability:
This version is available at: 11583/2974549 since: 2023-01-12T15:26:12Z

Publisher:
Institute of Electrical and Electronics Engineers Inc.

Published
DOI:10.1109/JLT.2022.3210769

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©2023 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

07 November 2026

Fiber-agnostic machine learning-based Raman amplifier models

Uiara C. de Moura, Darko Zibar, A. Margareth Rosa Brusin, Andrea Carena, and Francesco Da Ros

***Abstract*—Machine learning techniques have been applied to solve many open and highly complex problems in optical communications. In particular, neural networks (NN) have proved to be effective in learning the complex mapping between pump powers and gain profiles in Raman amplifiers. Since the Raman scattering process is highly dependent on the optical fiber, these NN-based Raman amplifier (RA) models are specific for a single optical fiber type. Therefore, countless NN models are required, one for each optical fiber type. In this work, we first show that by combining experimental data from different optical fiber sources, we can build a fiber-agnostic (or general) NN-based RA model. This single NN model can predict the gain profile of a new fiber type (unseen by the model during training) with a maximum absolute error (averaged over 1500 cases) as low as 0.22 dB. However, this generalization is only possible when the unseen fiber is similar to the fibers used to build the model. Therefore, a training dataset with a wide range of optical fibers is needed to enhance the chance of accurately predicting the gain of a new fiber. The first contribution of this work aims at avoiding the time-consuming experimental measurements of countless fibers. For that, here we extend and improve our general model by numerically generating the dataset. By doing so, it is possible to generate uniformly distributed data that covers a wide range of optical fiber types. To guarantee that the numerical simulation is able to reproduce the experimental results, we also propose a fitting procedure to extract the Raman gain coefficient from a few experimental measurements. The proposed *numerical data-based* general model is used to predict the gain of real fibers considering their corresponding Raman gain coefficient (extracted by the fitting procedure). The results show that the averaged maximum prediction error is reduced when compared to the limited *experimental data-based* general models. However, even after the fitting procedure, this fiber-agnostic model purely based on numerical data is still limited by the discrepancies between numerical and experimental measurements. Therefore, as the second and final contribution of this work, we propose the use of transfer learning (TL) to re-train the numerical data-based general model using just a few experimental measurements. Compared with the fiber-specific models, this TL-upgraded general model reaches very similar accuracy, with just 3.6% of the experimental data. These results demonstrate that the already fast and accurate NN-based RA models can be upgraded to have strong fiber generalization capabilities. They can, therefore, significantly reduce the total number of RA models that would be needed in multi-span optical networks.**

***Index Terms*—optical communications, optical amplifiers, machine learning, neural networks.**

Manuscript received xx xx, 20xx; revised xx xx, 20xx. This project has received funding from the European Research Council through the ERC-CoG FRECOM project (grant agreement no. 771878), the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 754462, the Villum Foundations (VYI OPTIC-AI grant no. 29344), and Ministero dell'Università e della Ricerca (PRIN 2017, project FIRST).

U.C. de Moura was with DTU Electro, Department of Electrical and Photonics Engineering, Technical University of Denmark, Kgs. Lyngby, Denmark, she is now with NKT Photonics, Denmark.

D. Zibar and F. Da Ros are with DTU Electro, Department of Electrical and Photonics Engineering, Technical University of Denmark, Kgs. Lyngby, Denmark. (e-mail: fdro@fotonik.dtu.dk).

A. M. Rosa Brusin and A. Carena are with Dipartimento di Elettronica e Telecomunicazioni (DET), Politecnico di Torino, Torino, Italy.

I. Introduction

THE concepts of open and disaggregated optical networks and software-defined networking (SDN) can improve network flexibility and manageability [1], [2]. Flexible networks can adapt to changes that might degrade the signal quality of transmission (QoT). Therefore, accurate and fast QoT estimator tools play an essential role. On the one hand, they trigger actions in response to varying network conditions, allowing the network to quickly adapt to changes. On the other hand, inaccurate QoT models require additional design margins and consequently network over-dimensioning [3].

Optical amplifiers are the main source of noise in optical communication systems, having a high impact on the signal QoT [4]. Rare-earth doped fiber amplifiers (xDFA) and Raman amplifiers (RAs) are the most promising technologies for multi-band optical transmission [5]. Unlike xDFAs, whose amplified frequency range is defined by the fiber dopant, RAs have the advantage of providing lumped or distributed gain at any frequency and over a variety of optical fibers. Moreover, broadband Raman amplification is possible by using multiple pump lasers at appropriate frequencies [6], [7]. Another attractive feature that is particular to RAs is the control of the gain spectral shape by adjusting the Raman pump powers. This concept of programmable

gain profiles [8] brings an additional degree of freedom to SDN-based controlled networks.

Very accurate optical amplifier models numerically solve a set of nonlinear ordinary differential equations (ODEs) [9], [10]. However, for real-time QoT estimation purposes, these models can be time-consuming, especially for multi-band RA with a high number of pumps. Moreover, such numerical models are not differentiable for the backward pumping scheme (the preferred one due to its robustness to pump power fluctuations). This prevents their applications in system optimizations based on gradient descent routines [11]. Therefore, machine learning (ML) has been recently applied to provide fast and highly-accurate gain and noise predictions for erbium-DFAs (EDFAs) [12]–[15] and RAs [16]–[21]. These models have a wide range of applications, such as effective signal power optimization [22], [23], signal-to-noise ratio optimization [11], and QoT estimation [24], [25].

For the RA, these ML-based models are specific to the given optical fiber used for gathering the training data. This is because the Raman gain strength and spectral shape depend on the material composition, the length, the attenuation, and the effective area of the optical fiber [26]. Relying on these fiber-specific models requires countless models to be available, one for each considered fiber span [27], and a massive amount of data to train each of them. Therefore, generalizable RA models that can predict the RA performance for fiber types unseen during training are critically needed.

We recently proposed an artificial neural network (NN) model for the backward pumping RA that includes the fiber parameters as additional inputs [28]. By considering an experimental training dataset with samples from multiple fiber types, such a fiber-agnostic (or general) model shows good generalization properties when the fiber-under-test (FUT) is not part of the training dataset. However, this is only true when the FUT has similar properties as the fibers used during training and has sufficient cases around it. Other fiber cases, whose parameters are outside the model training range, have a very poor prediction performance. This could be solved by generating comprehensive experimental data with a wide variety of fiber types and span lengths. However, this brings the same problem as for the fiber-specific models: the need for massive experimental data.

In this work, we extend [28] by enhancing the training dataset with numerically (synthetically) generated data. In this case, the fiber parameters can be varied uniformly, making it possible to cover a wider range of optical fibers. This synthetic dataset is generated after a fitting procedure to extract the Raman gain coefficient from the experimental data. This procedure is needed to match synthetic and experimental results and, therefore, make it possible to apply the numerical NN models to predict experimental data by using the estimated Raman gain coefficient. The proposed numerical data-based model is tested over an experimental dataset considering different commercial optical fibers. Obtained results show improvements when compared to the experimental data-based general models for most of the fiber cases. However, this approach is still limited by discrepancies between numerical and experimental measurements, even after extracting the experimental Raman gain coefficient.

Finally, to overcome these experimental-versus-numerical discrepancies, we apply transfer learning (TL) [29] by re-training the numerical data-based model with a few experimental cases. This new TL general model is able to provide gain predictions with very similar accuracy as for the fiber-specific models, with the great advantage of requiring just 3.6% of the training data. We, therefore, propose a TL-upgrade process for the general model that takes only $\sim$1 hour, going against the 8.5 hours required just to get the experimental data to train a single fiber-specific model. We experimentally demonstrate that such a general model based on numerical data and enhanced by TL has great generalization performance for different optical fibers. Moreover, it also avoids the need to embed a full numerical ODE solver into the QoT estimator as in [30], [31].

The paper is structured as follows. Section II reviews numerical and ML-based RA models, highlighting their main advantages and drawbacks. Section III describes the experimental setup used to gather the experimental data as detailed in Section IV. Part of this data will be used to train the experimental data-based NN models and the other part will be used to test all the NN models generated in this work. Section V reviews the fiber-specific and general models based on experimental data, presenting and discussing the main results from [28]. Section VI introduces the new general model based on numerical data. In this section, we describe in detail how the data is generated and the models are trained. Since we have some discrepancies between experimental and numerical data, we also show how we update the fiber parameters w.r.t the ones used in Section V and how we perform a fitting procedure to extract the Raman gain coefficient. Finally, to make the numerical data-based general model more robust to experimental versus numerical errors, Section VII uses the concepts of TL to enhance the numerical model generalization accuracy considering just a few experimental cases. Section VIII then summarizes and compares all the NN models results, presenting an overall discussion. Section IX

concludes this work.

II. Raman amplifier models

Signal and pump power evolution in RA is described by a set of ODEs [32]:

$$\frac{dP_{s,i}}{dz} = -\alpha_s P_{s,i} + g_R P_{p,j} P_{s,i}, \tag{1}$$

$$\frac{-dP_{p,j}}{dz} = -\alpha_p P_{p,j} - \frac{f_{p,j}}{f_{s,i}} g_R P_{s,i} P_{p,j}, \tag{2}$$

where $P_{s,i}$ is the signal power at frequency $f_{s,i}$, $P_{p,j}$ is the pump power at frequency $f_{p,j}$, $\alpha_{s,i}$ and $\alpha_{p,j}$ are the signal and pump attenuation coefficients (also at frequencies $f_{s,i}$ and $f_{p,j}$, respectively), and g_R is the Raman gain efficiency in $(\mathrm{W}\cdot\ \mathrm{km})^{-1}$ normalized with respect to the fiber effective area A_{eff}. g_R is a function of the frequency shift between pump and signal $(f_{p,j} - f_{s,i})$. These equations are for the counterpumping RA configuration considered in this work. A thorough equation that accounts for N_p pumps and N_s signal channels can also be found in [32]. This thorough model also includes pump–pump/signal-signal interactions and spontaneous Raman scattering.

When evaluating RAs, we are interested in their response in terms of gain and/or noise, given changes in the parameters that can affect this response. These parameters can be represented by a vector $\mathbf{X}$. Throughout this paper, $\mathbf{X}$ will assume different dimensions depending on what parameters are considered for a given RA. As for $\mathbf{Y}$, we will focus on predicting the Raman gain spectral shape $\mathbf{G}$ and, therefore, $\mathbf{Y} = \mathbf{G}$. The function $f(\mathbf{X}) = \mathbf{Y}$ can be obtained by solving equations (1-2) applying numerical methods like the one within GNPy library [9]. Notice that by using such a numerical approach, the effect of different parameters can be investigated such as the above mentioned $P_{s,i}$, $P_{p,j}$, α_s, α_p, g_R, and A_{eff}. $\mathbf{X}$ can incorporate one or more of these parameters and any other physical layer parameter such as fiber length and lumped attenuations (splices, couplers, etc). Although highly accurate, these numerical methods are very complex to be embedded into real-time QoT applications. They also require a detailed physical layer parameters extraction, as in [31], to accurately predict the gain profile. Finally, their non-differentiable characteristics limit their application to gradient-free system optimization approaches.

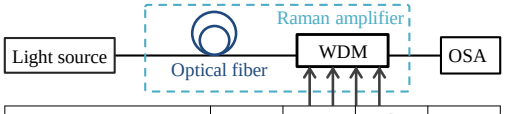
NNs, on the other hand, are universal approximators applied to extract the underlying function from noisy data [33]. For the RA case, they can be applied to extract the function $f(\mathbf{X}) = \mathbf{Y}$ from experimental (or numerical) generated data. Relying on matrix multiplications, these models are faster than applying numerical methods to solve equations (1-2). One of the main advantages of using machine learning to model the behavior of the RA is that these models are built exclusively and directly from experimental data. Therefore, previous knowledge of the experimental system is not critical. Although not as accurate as numerical models, NN-based RA models have proven to be very efficient in predicting Raman gain profiles [16], [17], [19], [27], amplified spontaneous emission (ASE) noise [18], [34] and noise figure (NF) [20]. In these models, $\mathbf{X}$ incorporates just the Raman pump powers and has, therefore, dimension N_p (number of pumps). They will be referred to henceforth as the *fiber-specific RA models* because they predict the gain of a specific optical fiber. It means that the fiber parameters such as α_s, α_p, g_R, A_{eff}, and length are fixed and do not need to be considered in $\mathbf{X}$.

Moreover, NN models are also applied to learn the inverse mapping $f(\mathbf{Y})^{-1} = \mathbf{X}$ to solve the inverse system design problem for the RA [8], [35]. These *inverse NNs* are also fiber-specific models and they are used for the RA inverse design. In other words, they provide a direct retrieval of the pump power configuration given the desired gain profile. In this work, NN will be applied to learn only the direct mapping $f(\mathbf{X}) = \mathbf{Y}$.

III. Experimental setup

Fig. 1 shows the RA experimental setup used to generate the datasets to train and/or test the RA NN models. The light source is an ASE source covering the C-band (192-196 THz). The total power of the fully lit bandwidth is +2.6 dBm at the input of the optical fiber, however, good generalizability to a quite large range of input power levels has already been shown, e.g. in [35] , as long as you remain in the unsaturated regime. The RA consists of an optical fiber and a commercial Raman pump module combining N_p = 4 lasers with fixed frequency and adjustable powers. Pump wavelength and maximum power values are shown in Fig. 1. These fixed pump frequencies are able to amplify the full C-band. Pumps and signals are combined through a wavelength division multiplexer (WDM) in a counter-propagating scheme. At the RA output, an optical spectrum analyzer (OSA) captures the power spectra to calculate the on-off gain profiles, i.e., the difference between the output power spectra with the lasers turned on and off. The gain profile is down-sampled to N_s = 40 signal channels at 100-GHz (ITU-T grid).

Five different optical fiber types are considered in this setup. These fibers cover a wide range of fiber lengths (L), signal (α_s) and pump (α_p) attenuation coefficients, and effective areas (A_{eff}), as shown in



Raman pumps, p	1	2	3	4
Frequency, f [THz]	206.1	207.5	209.0	210.6
Maximum power [mW]	145.0	158.5	180.0	152.5

Fig. 1. Raman amplifier setup for experimental and numerical dataset generation. WDM: wavelength division multiplexer, OSA: optical spectrum analyser.

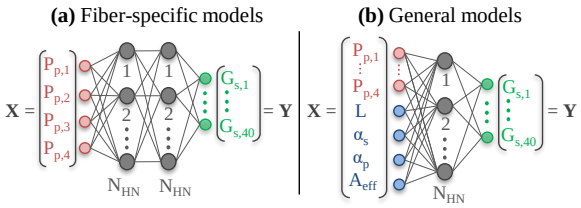


Fig. 2. Neural network architecture for (a) fiber-specific and (b) general models based on experimental data. N_{HN} is the number of nodes in the hidden layer.

Table I. Standard single-mode fiber (SMF) is well known and widely deployed in terrestrial systems. Two lengths are considered for the SMF: 50 and 100 km, referred to as SMF1 and SMF2, respectively. Highly nonlinear fiber (HNLF) and dispersion compensating fiber (DCF) are special fibers commonly used for lumped (discrete) RAs [36], with the latter presenting the advantage of also compensating for chromatic dispersion. Optimizations in the DCF design for submarine applications resulted in the inverse dispersion fiber (IDF) [37]. IDF has also been used for lumped RAs as an alternative to DCFs due to its lower attenuation [36]. Ultra low loss fiber (ULLF) is used for submarine and unrepeated links due to its lower loss and wider effective area for nonlinear impairments reduction. Both SMF and ULLF can be used as the gain medium for distributed RAs.

IV. Experimental Data Generation

For each optical fiber case in Table I, we collect 1,500 measurements, giving a full dataset with a total of 9,000 samples. Each sample k consists of a pump power configuration $\mathbf{P}^{(k)} = [P_{p,1}^{(k)}, P_{p,2}^{(k)}, P_{p,3}^{(k)}, P_{p,4}^{(k)}]^T$, the fiber parameter information $\mathbf{F}^{(k)} = [L^{(k)}, \alpha_s^{(k)}, \alpha_p^{(k)}, A_{eff}^{(k)}]^T$, and the corresponding measured RA on-off gain $\mathbf{G}^{(k)} = [G_{s,1}^{(k)}, G_{s,2}^{(k)}, ..., G_{s,40}^{(k)}]^T$, where T denotes the transpose. The pump power values are drawn from uniform distributions, whose minimum value is 0 mW and maximum values are the ones in Fig. 1. The fiber parameters in $\mathbf{F}^{(k)}$ are fixed and according to Table I. In our experimental setup, each sample takes around 20 seconds to be measured. Thus, the total time required to measure the dataset is around 8.5 hours per fiber case.

TABLE I
Optical fibers parameters

Parameter	L	α_s	α_p	A_{eff}
Units	km	dB/km	dB/km	μm^2
HNLF	7.5	**1.00**	**1.20**	**10**
DCF	**4.8**	0.50	0.80	15
IDF	15	0.23	0.31	31
SMF1	50	0.20	0.25	80
SMF2	**100**	0.20	0.25	80
ULLF	50	**0.16**	**0.2**	**153**

L: fiber length, α: attenuation coefficient, A_{eff}: effective area. Maximum/minimum values are in bold.

Another dataset is generated in the same way, with an identical number of samples, for testing the models. The full and fiber-specific test datasets are labeled $\mathcal{D}_{test}$ and $\mathcal{D}_{OF,test}$, respectively.

V. Experimental Data-Based RA Models

A. Fiber-specific NN models

A simplified version of the fiber-specific datasets are used to train the fiber-specific models. This version removes the fiber parameter information $\mathbf{F}_{OF}^{(k)}$ since it is not relevant for these models. The simplified ($simp$) specific dataset is then given by $\mathcal{D}_{OF,simp} = \{\mathbf{P}^{(k)}, \mathbf{G}^{(k)}\}_{k=1}^{1500}$.

Six fiber-specific NN models for each fiber case in Table I are independently trained over its corresponding training dataset $\mathcal{D}_{OF,simp}$. These models are referred to as NN_{OF}. They consider the fully connected two-hidden-layer NN architecture illustrated in Fig. 2(a), with pump power configuration $\mathbf{P}$ as input $\mathbf{X}$ and Raman gain $\mathbf{G}$ as output $\mathbf{Y}$ (consistent with $\mathcal{D}_{OF,simp}$). The training updates the NN weights using backpropagation in Tensorflow (Keras Library) through 500 epochs, a batch size of 32, and considering mean squared error as loss function. $\mathcal{D}_{OF,simp}$ is divided in the following way: 100 samples are reserved for hyperparameter optimization (detailed later in this section), and the remaining 1400 are used for training. The choice of 500 epochs is kept for all the NN models, as it is sufficient to ensure training convergence for all cases here considered. Under such conditions, the training of these models takes of the order of tens of minutes on a low-tier laptop, with none of the models exceeding 30 min.

First we analyze the models convergence w.r.t the training size, going from 50 to 1400 samples in $\mathcal{D}_{OF,simp}$. The obtained results are shown in Fig. 3, where MaxAE is the maximum absolute error along

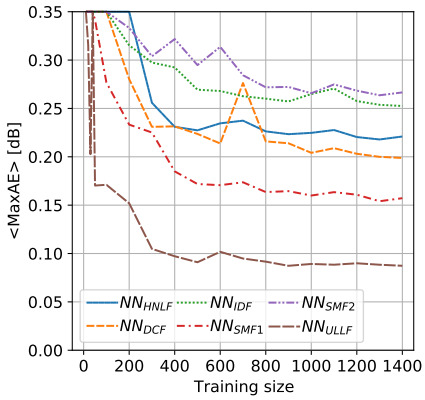


Fig. 3. Convergence analysis for the fiber-specific models NN_{OF} in terms of <MaxAE>, i.e., the maximum absolute error MaxAE averaged over the 100 samples reserved for hyperparameter optimization.

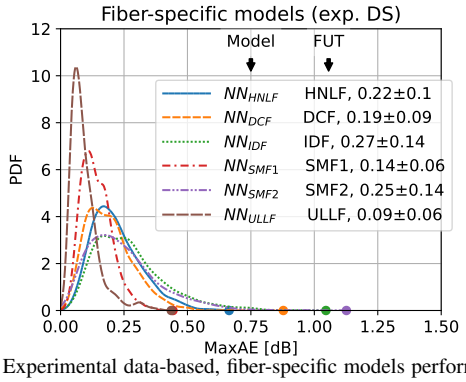


Fig. 4. Experimental data-based, fiber-specific models performance in terms of probability density function (PDF) of the maximum absolute error MaxAE indicating mean (<MaxAE>) and standard deviation (σ) values as <MaxAE>$\pm\sigma$. In this case, the fiber-under-test (FUT) is the same fiber (OF) used to generate the dataset to train the models NN_{OF}.

the 40 signal channels. It is defined as MaxAE = $max(|\mathbf{G}-\tilde{\mathbf{G}}|)$, with $\tilde{\mathbf{G}}$ being the NN gain prediction, i.e. $\tilde{\mathbf{G}} = NN_{OF}(\mathbf{P})$. In this case, <MaxAE> is calculated as:

$$<\text{MaxAE}> = \frac{1}{K}\sum_{k=1}^{K} max(|\mathbf{G}^{(k)} - \tilde{\mathbf{G}}^{(\mathbf{k})}|), \quad (3)$$

with K = 100 samples reserved for hyperparameter optimization. The results shown in Fig. 3 confirm that a good convergence is achieved with 1400 samples for all models. Therefore, the final fiber-specific models are trained considering 1400 samples.

The hyperparameter optimization consists in a grid search over the number of hidden nodes ($N_{HN} = \{10, 20, 30\}$), the activation function ($f_{act} = \{sin, tanh, sigmoid\}$), the optimization algorithm (opt), and the learning rate ($LR = \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$). More information about the optimization algorithms available within Tensorflow can be found in [38]. It means that several NN models are trained considering different hyperparameters. Each model is evaluated over the 100 samples are reserved for hyperparameter optimization. The hyperparameters with the best performance are shown in Table II. The activation function of the output nodes is linear and it is not optimized in this process. The training reserves 30% of the training data to verify if the model is over-fitting. It means that during the training, the model is tested over a different set, not used for training, to verify if the error is decreasing. When this validation error starts to increase, the training stops (early-stop criteria).

The performance of the final fiber-specific models are presented in Fig. 4 in terms of probability density function (PDF) of MaxAE. In this case, MaxAE is calculated over the samples in a simplified version of the test dataset given by $\mathcal{D}_{OF,test,simp} = \{\mathbf{P}^{(k)}, \mathbf{G}^{(k)}\}_{k=1}^{1500}$, i.e., removing the fiber parameter information $\mathbf{F}_{OF}^{(k)}$. Mean (<MaxAE>) and standard deviation (σ) for MaxAE are also shown as <MaxAE>$\pm\sigma$ in the legend. In this case, NN_{OF} is trained and tested considering the same fiber and, therefore, FUT in Fig. 4 corresponds to OF in NN_{OF}. These models are highly accurate for all fiber cases, with MaxAE ranging from 0.09±0.06 dB (ULLF) to 0.27±0.14 dB (IDF).

However, these fiber-specific models have very poor generalization properties, i.e., they cannot be applied to predict the gain profile for an optical fiber different from the one used during the training. Fig. 5 shows the performance of these models when $FUT \neq OF$. In this case, FUT are all the fibers, except OF. We refer to it as $FUT = OF_{out}$ in Fig. 5. This poor accuracy is

TABLE II
Neural network parameters for the fiber-specific models (experimental data-based)

Model	N_{HN}	f_{act}	opt	LR
NN_{HNLF}	20	sigmoid	adamax[1]	10^{-2}
NN_{DCF}	30	tanh	adam[2]	10^{-3}
NN_{IDF}	30	sigmoid	ftrl[3]	10^{-1}
NN_{SMF1}	30	sigmoid	adam[2]	10^{-2}
NN_{SMF2}	10	sigmoid	nadam[4]	10^{-2}
NN_{ULLF}	20	sigmoid	adam[2]	10^{-3}

1. Adamax = adaptive moment estimation (Adam) variant based on the infinity norm,
2. Adam: adaptive moment estimation,
3. ftrl: follow the regularized leader,
4. Nadam: Nesterov accelerated adaptive moment estimation (Adam).

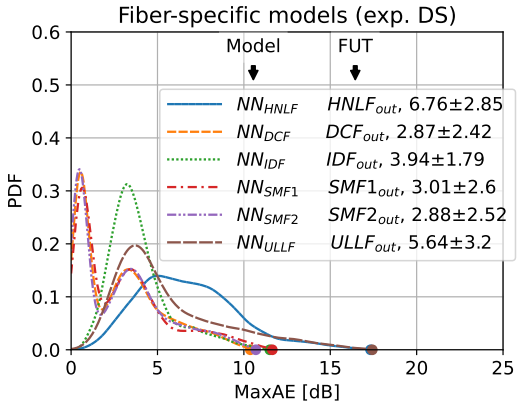


Fig. 5. Experimental data-based, fiber-specific models performance in terms of probability density function (PDF) of the maximum absolute error MaxAE indicating mean (<MaxAE>) and standard deviation (σ) values as <MaxAE>$\pm\sigma$. In this case, the fiber-under-test (FUT) is referred to as OF_{out}, and corresponds to all the other fibers different from the fiber (OF) used to generate the dataset to train the models NN_{OF}.

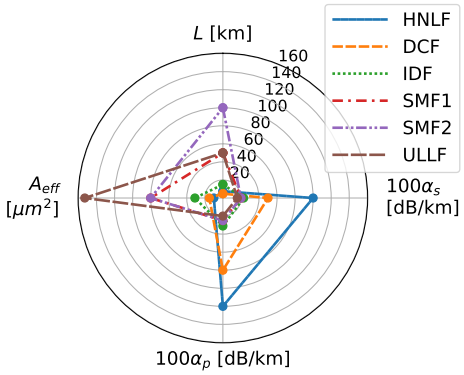


Fig. 6. Optical fiber parameters distribution showing the extreme cases.

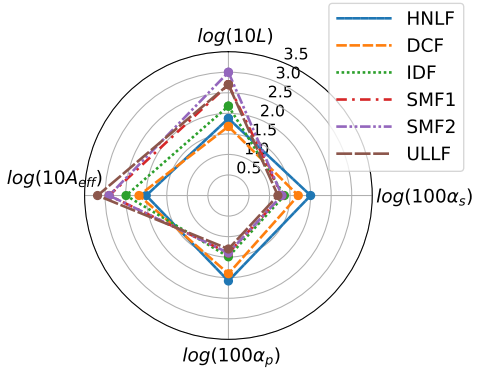


Fig. 7. Optical fiber parameters distribution after applying the logarithmic transformation function indicated in each axis.

expected since the Raman gain shape and level are highly dependent on the properties of the optical fiber [26]. In other words, a model without knowledge of the fiber characteristics cannot generalize to a different fiber type. This means that a dedicated NN model is needed for each fiber type/length. Considering the high diversity of fiber types and span lengths in optical communication systems, this would require a huge number of fiber-specific models and also a massive amount of measurements to gather the data for training. Moreover, QoT estimator tools would need to incorporate countless RA NN models and update them if a new fiber is deployed in the system. Therefore, a general model able to predict the RA performance for fiber types unseen during training is highly desired. This would require QoT estimator tools to incorporate just a single RA model that would work for any optical fiber.

B. General NN models (experimental data-based)

The general model is applied to learn not only the mapping between pump configuration and gain profiles but also how the fiber characteristics affect these gain profiles. Therefore, the fiber parameters shown in Table I are incorporated into the NN input $\mathbf{X}$, resulting in $\mathbf{X} = [P_{p,1}, P_{p,2}, P_{p,3}, P_{p,4}, L, \alpha_s, \alpha_p, A_{eff}]$, as shown in Fig. 2(b). To keep the approach practical, these parameters were chosen due to their importance in determining the gain profiles and also because they are commonly provided by the fiber's suppliers on the fibers' datasheets [28]. The Raman gain coefficient (g_R) requires additional characterization. However, we verified that general NN models trained considering g_R do not show higher accuracy w.r.t the models not considering it. These results are not shown here because they are not relevant to the general discussion.

In Fig. 6 we can visualize how the fiber parameters are distributed in a non-uniform way. Therefore, we need to perform an input transformation to the fiber parameters to make them more uniformly distributed among the optical fibers. This transformation is needed to improve the accuracy of the models. The resulted transformation is shown in Fig. 7, where a log function is applied to all fiber parameters after being multiplied by 10 (L and A_{eff}) or 100 (α_s and α_p).

To be able to evaluate the generalization properties of the proposed general model, i.e. its ability in predicting the gain profile for an unseen FUT, we build six general NN models. Each model is trained over the dataset $\mathcal{D} \setminus \mathcal{D}_{FUT}$, where $\setminus$ is the set difference operator. This means that the data corresponding to the FUT is removed from

the training dataset $\mathcal{D}$. These models are referred to as $NN_{FUT_{out}}$ and they are tested over the test dataset of the removed fiber $\mathcal{D}_{FUT,test}$. $NN_{FUT_{out}}$ training and testing datasets are shown in Table III. These datasets were defined in Section IV.

These general models consider the single-hidden layer architecture illustrated in Fig. 2(b). We apply 10-fold cross-validation for model selection. During the training, the weights are updated using an extreme learning machine (ELM) [39] method implemented in MATLAB. In this single-step and fast training approach, referred to as either ELM or random-projection method, only the last layer's weights are updated using the training dataset by regularized least squares (regularization parameter λ). The hidden layer's weights are randomly assigned according to a normal distribution (zero mean and standard deviation σ_w) and are kept constant during the training. The hyperparameter optimization resulted in f_{act} = sine, $\sigma_w = 10^{-2.5}$, and $\lambda = 10^{-8}$ for all general models and N_{HN} according to Table III. A linear activation function is considered for the output nodes. Model averaging using 20 parallel NNs is applied to compensate for the hidden layer's weights random assignment. This means that the predicted gain is the averaged output of 20 independently trained NN models. We also trained the general models using backpropagation, applying the same procedure as for the fiber-specific models described in Section V-A. However, such models trained using backpropagation are not as accurate as the ones considering ELM. Therefore, their results are not considered here.

The performance of the models is evaluated similarly to the fiber-specific models. Fig. 8 shows MaxAE's PDF for each general model $NN_{FUT_{out}}$, where a degradation in performance is observed especially for the fiber cases whose parameters have maximum or minimum values in Fig. 7. It is known that the generalization of a NN

TABLE III
Training and testing datasets (DS) and the number of hidden layers (N_{HN}) for the general models (experimental data-based)

Model	Train DS	Testing DS	N_{HN}
$NN_{HNLF_{out}}$	$\mathcal{D} \setminus \mathcal{D}_{HNLF}$	$\mathcal{D}_{HNLF,test}$	500
$NN_{DCF_{out}}$	$\mathcal{D} \setminus \mathcal{D}_{DCF}$	$\mathcal{D}_{DCF,test}$	400
$NN_{IDF_{out}}$	$\mathcal{D} \setminus \mathcal{D}_{IDF}$	$\mathcal{D}_{IDF,test}$	700
$NN_{SMF1_{out}}$	$\mathcal{D} \setminus \mathcal{D}_{SMF1}$	$\mathcal{D}_{SMF1,test}$	600
$NN_{SMF2_{out}}$	$\mathcal{D} \setminus \mathcal{D}_{SMF2}$	$\mathcal{D}_{SMF2,test}$	800
$NN_{ULLF_{out}}$	$\mathcal{D} \setminus \mathcal{D}_{ULLF}$	$\mathcal{D}_{ULLF,test}$	700

f_{act} = sine, $\sigma_w = 10^{-2.5}$, and $\lambda = 10^{-8}$ for all models. $\setminus$ is the set difference operator.

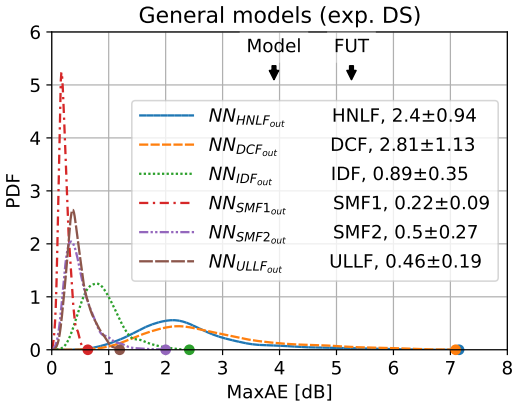


Fig. 8. Experimental data-based, general models performance in terms of probability density function (PDF) of the maximum absolute error MaxAE indicating mean (<MaxAE>) and standard deviation (σ) values as <MaxAE>$\pm\sigma$. In this case, the data of the fiber-under-test (FUT) is removed from the dataset used to generate the model, providing a total of 6 models corresponding to each removed FUT.

model is influenced by the size of the training data and how representative the training data is over the space of interest [40]. To verify if this low accuracy is due to insufficient data in the training dataset, we perform a convergence analysis w.r.t to training dataset size. For that, each fiber case in the training dataset $\mathcal{D} \setminus \mathcal{D}_{FUT}$ contributes with the same number of samples (50 to 1500) to the overall training dataset, providing a total of 250 to 7500 samples. The results are presented in Fig. 9, showing that after around 500 samples, the performance is already very similar to the ones obtained in Fig 8. This hints that the problem is more likely related to the fiber parameter distribution in the training dataset, rather than the need for more samples. In Fig 9, <MaxAE> is calculated for the fiber not considered for training, i.e., $\mathbf{G}^{(k)} \in \mathcal{D}_{FUT,test}$ and K = 1500 in equation (3). Model averaging is not considered for the convergence analysis.

In fact, returning to Fig. 8, HNLF has the highest [α_s,α_p] and the lowest A_{eff} values. Thus, when predicting the HNLF gain profiles using $NN_{HNLF_{out}}$, the inputs are outside the training range, and NNs are known for their poor ability to extrapolate information [33]. IDF parameters, on the other hand, have no minimum/maximum values in Table I. However, $NN_{IDF_{out}}$ has a relatively low accuracy with <MaxAE>$\approx$ 1 dB. Even though the input transformation provides a better distribution (or representativeness) of the fiber parameters, recall that $NN_{IDF_{out}}$ is trained over $\mathcal{D} \setminus \mathcal{D}_{IDF}$. When removing the IDF data in Fig 7, a gap appears, affecting the uniform distribution. Therefore, this suggests that $NN_{IDF_{out}}$ is not trained with sufficient samples around its fiber parameters. This is not the

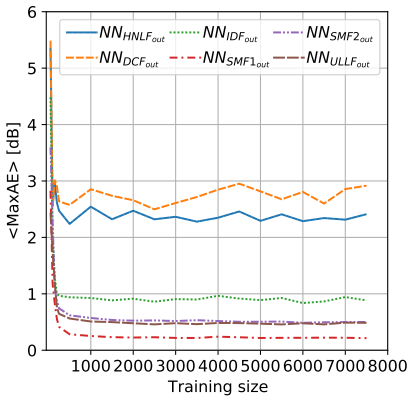


Fig. 9. Convergence analysis for the general models $NN_{FUT_{out}}$ in terms of <MaxAE>, i.e., the maximum absolute error MaxAE averaged over all the cases in the corresponding test dataset $\mathcal{D}_{FUT,test}$.

case for the $NN_{SMF1_{out}}$. $NN_{SMF1_{out}}$ shares all its parameters with other fiber cases (SMF2 and ULLF). This leads to the best generalization performance, with a <MaxAE> of 0.22 dB, i.e. close to the fiber-specific model performance shown in Fig. 4. Therefore, a good generalization is most likely to be achieved when the new FUT is inside the model training range and has sufficient samples around it.

These poor generalization properties could be improved by considering a large number of optical fibers with different parameters to diversify the training dataset. However, this would require time and effort for massive experimental measurements over a wide variety of fiber types and span lengths. Therefore, as the main contribution of this work, we propose to generate the dataset numerically, with better control of the granularity of the fiber parameters and, therefore, a more uniform distribution of the cases within the dataset. Moreover, synthetic data would allow for parallelized generation, with high contrast with the experimental data.

VI. Numerical data-based RA models

To overcome the limitations and drawbacks of generating a comprehensive experimental dataset, we generate a numerical dataset considering uniform distributions of the optical fiber parameters. The simulations considered the numerical solver available within the open-source library GNPy [9] and referred to as the Raman solver (RS) from now on. The new general models, trained over the numerical data, will be tested over the experimental data, i.e., for each experimental $\mathcal{D}_{FUT,test}$.

The setup considered for the numerical simulations matches the experimental setup of Fig. 1. In this case, the light source is 40 continuous-wave lasers in the C-band (ITU-T grid, 100 GHz spaced) with a total power of +2.6 dBm, as in the experimental dataset collection. Also different from the experiment, all pump lasers have a maximum power of 200 mW.

A. Fiber parameters update

To make the approach simple and to avoid additional laboratory measurements, Table I in Section III considers typical values retrieved from the literature [36], [37] and from fiber suppliers specifications [41]–[43]. However, the attenuation coefficients (α_s and α_p) are not accurate enough for the RS to reproduce the experimental results. Therefore, we update the attenuation coefficients after simple measurements using a frequency-tunable optical time-domain reflectometer (OTDR). These updated ($updt$) values are presented in Table IV and they were measured at 1550 nm (signal) and 1450 mm (pump).

Moreover, RS considers the Raman gain coefficient (g_R) instead of the A_{eff} to solve the ODEs describing the Raman effect. Therefore, to build a model with numerically generated data from the RS and apply it to predict the gain of an experimental fiber, we need to extract the fiber's g_R and use it as one of the inputs of the NN model. Fig.10 shows the normalized g_R curves from the literature [36] for the fibers considered in this work (except for ULLF because it is very similar to the SMF). As we mentioned in Section V-B, obtaining these curves and their respective peaks would require additional and more complex characterization for each optical fiber. Therefore, here we propose a simple procedure to extract g_R peak (g_R^{peak}) from a few experimental measurements (a small part of the same experimental data used to train the models in Section V-B). Furthermore, in this work, we consider the same attenuation coefficient (α_p) and the same Raman gain coefficient for all pumps. Therefore, to compensate for these simplifications, the proposed fitting procedure also extracts pump power calibration values l_{P_i} for each pump i at the fiber input. These include, e.g. the frequency dependence of g_R^{peak} and α_p, and connection losses.

The proposed fitting procedure consists in solving an optimization problem to extract g_R^{peak} and $l_{P_{1-4}}$ from a few experimental measurements for a given fiber case. The solution of this problem has the form $\mathbf{s} = [g_R^{peak}, l_{P_1}, ..., l_{P_4}]^T$. To find the optimum solution $\mathbf{s}_{opt}$, we apply the RS to numerically solve equation (1-2) and provide the RA response in terms of gain for a trial solution $\mathbf{s}$. The idea is to find a way to adjust this trial solution until the error between numerical

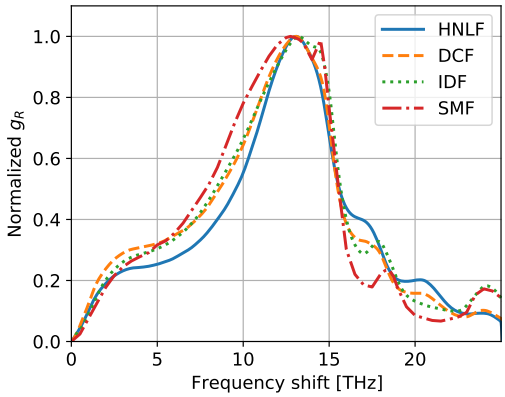


Fig. 10. The normalized Raman gain coefficient versus the frequency shift between pump and signal for the optical fibers considered in this work (ULLF curve is very similar to the SSMF) [36]

and experimental gain measurements is small enough. It is very challenging to obtain an expression for the gradient of these equations to apply a gradient descent optimization technique to find $\mathbf{s}_{opt}$. Therefore, we solve this problem by using the differential evolution (DE) algorithm, a derivative-free optimization method based on genetic algorithm concepts [44].

The DE algorithm starts with a population of initial solutions $\mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N\}$. These solutions are evaluated and updated in an iterative process until they converge to a (sub-)optimum solution. As a stochastic global optimization method, DE tries to escape from local optima, but an optimum solution is not guaranteed [44]. In general words, at each epoch, each candidate solution is evaluated over n experimental measurements from $\mathcal{D}_{OF}$. This evaluation considers a cost function $f_{cost}(\mathbf{s})$ relating experimental $\mathbf{G}_{exp}^{(k)} \in \mathcal{D}_{OF}$ and numerical $\mathbf{G}_{num}^{(k)}$ gains for each solution $\mathbf{s}$:

$$f_{cost}(\mathbf{s}) = max\{max(|\mathbf{G}_{exp}^{(k)} - \mathbf{G}_{num}^{(k)}|)\}_{k=1}^{n} \quad (4)$$

where $\mathbf{G}_{num}^{(k)}$ is a function of $\mathbf{s}$. It is obtained by running the RS considering the pump configuration from $\mathcal{D}_{OF}$ attenuated by the pump lumped losses in $\mathbf{s}$, and g_R^{peak} according to the solution $\mathbf{s}$. It also considers the updated values for α_s and α_p from Table IV. These values are kept constant during the optimization process.

A more detailed explanation of DE is out of the scope of this work and can be found in [44]. The population size is N = 60 and the optimization set has n = 10 experimental measurements from $\mathcal{D}_{OF}$. The maximum number of epochs is set to 100. A validation set with 10 new experimental measurements (also from $\mathcal{D}_{OF}$) is used to monitor the best solution of the iteration and avoid over-fitting. It means that the final solution $\mathbf{s}_{opt}$ is the one that provides the best performance in the validation set. Additionally, we also consider increasing n. However, it just increases the optimization time without bringing improvements to the results.

The complete optimization takes less than 30 minutes considering running 12 parallel instances of the RS, a very similar time processing achieved by [31]. In our case, this time could be significantly reduced by at least a factor of 10 if we could evaluate all N = 60 candidate solutions over the optimization set (n = 10), which gives 600 RS instances, totally in parallel for each iteration.

The optimized values of g_R^{peak} are also shown in Table IV. The optimized pump power calibration values are shown in Table V and they do not have a clear trend w.r.t the wavelength. This is because they account for the impact of several physical effects and measurement uncertainties based on lumped effective values. Such calibration values compensate for the choice of a simplified model which does not account for the wavelength dependence of attenuation, gain peak and effective area, as well as measurement uncertainty that can generate connector loss and pump diode driving current errors.

Finally, we reproduce the full experimental data $\mathcal{D} \cup \mathcal{D}_{test}$ numerically, considering the updated fiber parameters from Table IV, the g_R curves from Fig 10, and the pump power calibration values from Table V. The maximum absolute error between $\mathbf{G}_{exp}$ and $\mathbf{G}_{num}$ along the channels $max(|\mathbf{G_{exp}} - \mathbf{G_{num}}|)$ is summarized in Fig. 11. Mean ($<max(|\mathbf{G_{exp}} - \mathbf{G_{num}}|)>$) and standard

TABLE IV
Updated optical fibers parameters

Parameter Units	L km	$\alpha_{s,updt}$ dB/km	$\alpha_{p,updt}$ dB/km	g_R^{peak} $(W \cdot km)^{-1}$
HNLF	7.5	**0.89**	**1.12**	**5.15**
DCF	**4.8**	0.48	0.68	3.04
IDF	15	0.24	0.33	1.59
SMF1	50	0.19	0.24	0.39
SMF2	**100**	0.19	0.24	0.34
ULLF	50	**0.15**	**0.2**	**0.21**

L: fiber length α: attenuation coefficient, g_R^{peak}: Raman gain coefficient peak. Maximum/minimum values are in bold

TABLE V
Optimized pump power calibration values

Parameter	l_{P_1} (dB)	l_{P_2} (dB)	l_{P_3} (dB)	l_{P_4} (dB)
HNLF	1.31	1.53	0.98	0.50
DCF	2.50	2.56	1.69	0.94
IDF	2.77	2.45	1.39	1.16
SMF1	0.91	0.71	0.30	0.47
SMF2	0.59	-0.23	-0.48	0.01
ULLF	2.36	2.09	0.95	0.97

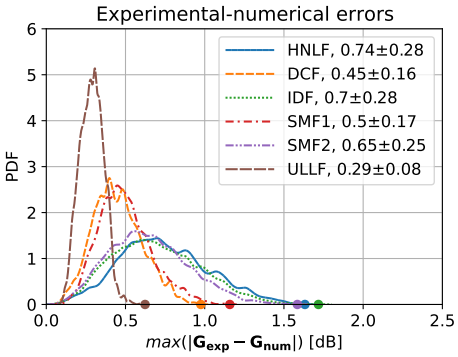


Fig. 11. Probability density function for the fitting error $max(|\mathbf{G_{exp}} - \mathbf{G_{num}}|)$ calculated over the experimental data $\mathcal{D} \cup \mathcal{D}_{test}$ (removing 20 cases used for the DE optimization). Numerical simulations consider the updated fiber parameters from Table IV. Mean ($<max(|\mathbf{G_{exp}} - \mathbf{G_{num}}|)>$) and standard deviation (σ) values for $max(|\mathbf{G_{exp}} - \mathbf{G_{num}}|)$ are indicated as $<max(|\mathbf{G_{exp}} - \mathbf{G_{num}}|)>\pm\sigma$ in the legend.

deviation (σ) (shown as $<max(|\mathbf{G_{exp}} - \mathbf{G_{num}}|)>\pm\sigma$ on the right side of each legend) are calculated over all 2980 cases (removing the cases used for the DE optimization). These errors are related to measurement inaccuracies and can be considered as the lower limit for the numerical data-based general models performance.

The new fiber parameters from Table IV and the lumped pump losses from Table V are used to update the experimental datasets from Section IV. This is because we test all the numerical data-based models built in this section over the experimental data. We also use the experimental data to re-train the numerical model by means of transfer learning, as it will be detailed in Section VII. These updated datasets will be referred to as $\mathcal{D}_{OF,updt}$ and are given by $\mathcal{D}_{OF,updt} = \{\mathbf{P}^{(k)}_{updt}, \mathbf{F}^{(k)}_{OF,updt}, \mathbf{G}^{(k)}\}_{k=1}^{1500}$, with $\mathbf{P}_{updt} = [P_{p,1} - l_{P_1}, ..., P_{p,4} - l_{P_4}]$ ($P_{p,i}$ is in dBm), $\mathbf{F}_{OF,updt} = [L, \alpha_{s,updt}, \alpha_{p,updt}, g_R^{peak}]$, and $\mathbf{G}^{(k)}$ as in $\mathcal{D}_{OF}$ (Section IV). The same update is performed to obtain the test datasets $\mathcal{D}_{OF,test,updt}$ from $\mathcal{D}_{OF,test}$ and the values in Tables IV and V. Notice that A_{eff} is replaced by g_R^{peak} in $\mathbf{F}_{OF,updt}$. We opt to use A_{eff} in the first part of this work because it is promptly available in the fiber's suppliers data-sheet. However, as already mentioned, RS considers g_R^{peak} instead of A_{eff} to generate the numerical data. Therefore, the numerical models need the information of the fiber g_R^{peak} to predict its the gain profile. Moreover, having both g_R^{peak} and A_{eff} is redundant since g_R^{peak} is normalized by A_{eff}.

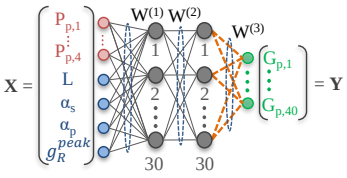


Fig. 12. Neural network architecture for general models based on numerical data.

B. Numerical data generation

The numerical dataset could be generated by uniformly varying from minimum to maximum the values of the fiber parameters in Table IV. However, this would lead to unrealistic cases, such as high attenuation fibers with long transmission distances. Therefore, we generate two numerical dataset groups corresponding to lumped (lump) RAs with high g_R and distributed (dist) RAs with low g_R. They are referred to as $\mathcal{D}_{lump}$ and $\mathcal{D}_{dist}$, respectively, and their intervals are according to Table VI. $\mathcal{D}_{lump}$ has high attenuation coefficients over up to 20 km of fiber lengths. Whereas $\mathcal{D}_{dist}$ is the opposite (low attenuation coefficients and fiber lengths higher than 20 km). We generate the numerical data considering the literature g_R curves from Fig. 10. $\mathcal{D}_{dist}$ uses the SMF g_R curve and $\mathcal{D}_{lump}$ uses an average between HNLF, DCF, and IDF. More than 100,000 samples were generated for each dataset group.

Each sample k in the numerical dataset consists of a pump power configuration $\mathbf{P}^{(k)}_{num} = [P^{(k)}_{p,1}, P^{(k)}_{p,2}, P^{(k)}_{p,3}, P^{(k)}_{p,4}]^T$, a fiber parameter information $\mathbf{F}^{(k)}_{num} = [L^{(k)}, \alpha^{(k)}_s, \alpha^{(k)}_p, g_R^{peak(k)}]^T$, and the corresponding RA response in terms of on-off gain $\mathbf{G}^{(k)}_{num} = [G^{(k)}_{s,1}, G^{(k)}_{s,2}, ..., G^{(k)}_{s,40}]^T$. The pump power values in $\mathbf{P}_{num}$ are drawn from $\mathcal{U}[0; 200]$ mW. Different from the experimental dataset in Section IV, here the fiber parameters in $\mathbf{F}_{num}$ are drawn from uniform distributions considering the intervals in Table VI.

The final numerical datasets are given by $\mathcal{D}_{lump/dist} = \{\mathbf{P}^{(k)}_{num}, \mathbf{F}^{(k)}_{num}, \mathbf{G}^{(k)}_{num}\}_{k=1}^{K}$, with K = 105,521 for $\mathcal{D}_{lump}$ and K = 106,561 for $\mathcal{D}_{dist}$.

TABLE VI
Fiber parameter ranges for the numerical datasets

Parameter	$\mathcal{D}_{lump}$	$\mathcal{D}_{dist}$
L (km)	U[1:20]	U[20:120]
α_s (dB/km)	U[0.20:1.20]	U[0.15:0.22]
α_p (dB/km)	U[0.28:1.28]	U[0.18:0.32]
g_R^{peak} $((W \cdot km)^{-1})$	U[1.00:5.50]	U[0.20:0.99]
g_R curve (Fig. 10)	$< [HNLF, DCF, IDF] >$	SMF

C. General NN models (numerical data-based)

We consider three numerical models: a general model trained over the full $\mathcal{D}_{num} = \mathcal{D}_{lump} \cup \mathcal{D}_{dist}$ dataset, and two independently models trained over $\mathcal{D}_{lump}$ and $\mathcal{D}_{dist}$. These models are referred to as NN_{num}, NN_{dist}, and NN_{lump}, respectively. $\mathcal{D}_{lump}$ generalizes for HNLF, DCF and IDF and $\mathcal{D}_{dist}$ generalizes for SMF1, SMF2 and ULLF. $\mathcal{D}_{num}$, $\mathcal{D}_{lump}$ and $\mathcal{D}_{dist}$ are divided into training, hyperparameter optimization (*hyperparam. opt.*), and testing, according to the sizes shown in Table VII. Just part of all samples generated in Section VI-B is used. The training sizes were obtained after a convergence analysis checking the NNs accuracy in the *hyperparam. opt.* set for different training sizes. Notice that the training data size for NN_{num} is larger than the sum of the training data for NN_{lump} and NN_{dist}. It is because NN_{num} should generalize for lumped and distributed RAs.

These models consider the NN architecture presented in Fig. 12, with $\mathbf{X} = [P_{p,1}, ..., P_{p,4}, L, \alpha_s, \alpha_p, g_R^{peak}]^T$ and $\mathbf{Y} = \mathbf{G}$ (consistent with $\mathcal{D}_{num}$ and $\mathcal{D}_{OF,test,updt}$). They are trained in a similar way as for the fiber-specific models in Section V-A, i.e applying backpropagation in TensorFlow to update the NN weights. We consider 500 epochs and a batch size of 32. A portion of 30% of the training set is reserved to avoid over-fitting. An hyperparameter optimization phase defines the number of hidden nodes (N_{HN}), the optimization algorithm (opt), the learning rate (LR), and the activation function (f_{act}) for the hidden nodes. This is done by evaluating the candidate models over the *hyperparam. opt.* set and selecting the best model, whose parameters are also shown in Table VII. The output layer nodes consider a liner activation function.

Fig. 13 shows the performance in terms of MaxAE ($max(|\mathbf{G} - \tilde{\mathbf{G}}|)$) of the proposed numerical models tested over the numerical testing set (Table VII). Recall that $\tilde{\mathbf{G}}$ is the NN gain prediction. All three models

TABLE VII
Dasaset sizes and neural network parameters for the numerical data-based general models

NN models	NN_{lump}	NN_{dist}	NN_{num}
Training	90,000	30,000	170,000
Hyperparam. opt.	10,000	10,000	20,000
Testing	10,000	10,000	20,000
LR	10^{-3}	10^{-2}	10^{-3}
opt	Nadam[1]	Adamax[2]	Nadam[1]

f_{act} = sigmoid, and N_{HN} = 30 for all models.
1. Nadam: Nesterov accelerated adaptive moment estimation (Adam),
2. Adamax = adaptive moment estimation (Adam) variant based on the infinity norm.

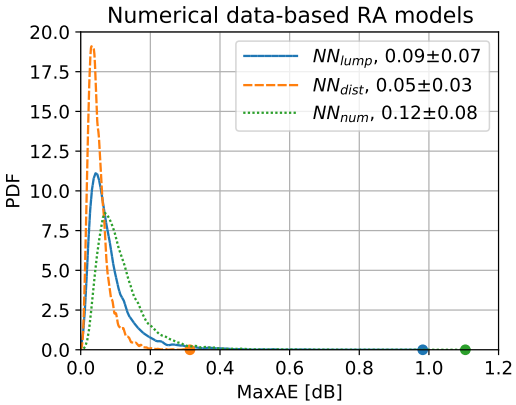


Fig. 13. Numerical data-based, general models performance in terms of probability density function (PDF) of the maximum absolute error MaxAE indicating mean (<MaxAE>) and standard deviation (σ) values as <MaxAE>$\pm\sigma$. Training and test datasets are numerical.

are highly accurate, with very low <MaxAE> values. NN_{dist} has the best performance, very similar to the one obtained for a fiber-specific numerical model (only SMF) in [18]. NN_{lump} and NN_{dist} present a smaller <MaxAE> degradation. Regarding maximum MaxAE, it goes from 0.4 dB for NN_{dist} to around 1 and 1.6 dB for NN_{lump} and NN_{num}, respectively. However, for both NN_{lump} and NN_{num}, just ≈1% of the cases have maximum MaxAE > 0.4 dB.

Fig. 14 shows the performance over the experimental data ($\mathcal{D}_{FUT,test,updt}$) for NN_{num} (Fig. 14(a)) and for NN_{lump} and NN_{dist} (Fig. 14(b)). That is testing the model traning using the numerical dataset (with updated parameters) on the measured test dataset. These results are very similar to each other, showing that dedicated models for each group is not necessary. When compared to the general experimental model results in Fig. 8, the numerical models are able to bring down the errors for the highly inaccurate cases, i.e. HNLF and DCF. <MaxAE> drops from 2.4 to 1.01 dB (HNLF) and from 2.81 to 0.41 dB (DCF). While maximum MaxAE goes from ≈7 dB to ≈2 dB (HNLF) and to ≈1.4 dB (DCF). Improvements are also observed for IDF and ULLF, especially for maximum MaxAE, which reduces from ≈2.5 to ≈1.8 dB (IDF) and from ≈1.2 to ≈0.6 dB (ULLF). Except for the HNLF, the performance of the numerical models are very close to the fit errors in Fig. 11. These errors limit the performance of the numerical models in Fig. 14 and are the reason why it is not possible to improve the performances for the SMF cases (SMF1 and SMF2). They also prevent to achieve performance close to the fiber-specific experimental models in Fig. 4.

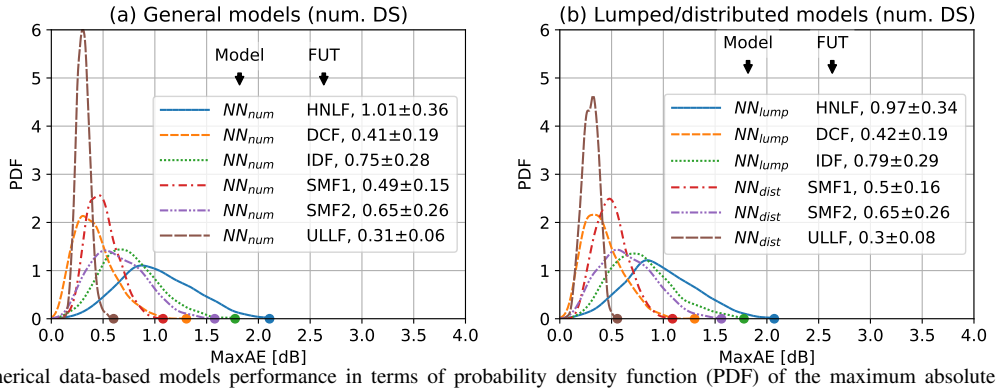


Fig. 14. Numerical data-based models performance in terms of probability density function (PDF) of the maximum absolute error MaxAE indicating mean (<MaxAE>) and standard deviation (σ) values as <MaxAE>$\pm\sigma$. The models are tested over the experimental data corresponding to the fiber-under-test (FUT) $\mathcal{D}_{FUT,test,updt}$.

VII. Transfer Learning Data-Based Models

It is well known in the ML community that a NN model previously trained on a large and general enough dataset can be customized via transfer learning (TL) for a more specific task [29]. This is done by re-training a base (pre-trained) model considering the dataset of a new but related, task. In the ML context, this concept is applied when the training data for the new specific task is limited. Therefore, the re-training step requires fewer cases, when compared to the number needed to train the model from scratch.

Here, we apply TL to improve the performance of NN_{num}. In other words, it means that NN_{num}, already trained over a very general and numerically generated RA data, can be fine-tuned for specific and experimental fibers by means of TL. In this case, we consider the experimental data of each fiber case (updated in Section VI-A), i.e., $\mathcal{D}_{OF,updt}$. The re-training consists in using the random experimental data to update just the output layer weights of NN_{num} (highlighted in orange in Fig. 12), using its current values as an initial condition. The hidden layers' weights are kept constant and are responsible for keeping the knowledge already learned from the general numerical data.

As done for the experimental fiber-specific cases, we build six models, each one corresponding to an optical fiber case OF used for the re-training. These models are referred to as $NN_{TL,OF}$ and are also trained by backpropagation in Tensorflow (epochs = 500, batch size = 32). Recall that each $\mathcal{D}_{OF,updt}$ has 1500 samples, of which we use a random subset for re-training. Here, we also reserve 100 samples for hyperparameter optimization.

Since the idea of TL is to use as few training samples as possible, we start by evaluating how many samples we need to re-train NN_{num}. Fig. 15 shows how <MaxAE> changes with the training size for the $NN_{TL,OF}$ models. In this case, <MaxAE> is calculated using equation (3) over K = 100 cases reserved for hyperparameter optimization. These results show that all models converge for 200 random samples in the training dataset. This value corresponds to only 14.3% of the samples used to train the fiber-specific models in Section V-A.

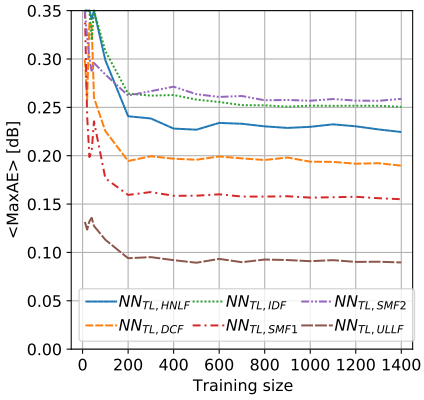


Fig. 15. Transfer learning convergence analysis when re-training the NN_{num} with different TL training sizes in terms of <MaxAE>, i.e., the maximum absolute error MaxAE averaged over all the cases in the corresponding $\mathcal{D}_{OF,test,updt}$ dataset for each OF.

An hyperparameter optimization is applied to find the best optimization algorithm (opt) and learning rate (LR). It results in the values presented in Table VIII for TL training sizes of 50 and 200. The activation function (f_{act}) and number of hidden layers (N_{HN}) are the same as for the NN_{num} (Table VII), as the network architecture is kept fixed in the TL.

Fig. 16 shows the performance the $NN_{TL,OF}$ models over $\mathcal{D}_{OF,test,updt}$ when considering just 50 samples in the re-training dataset. This corresponds to less than

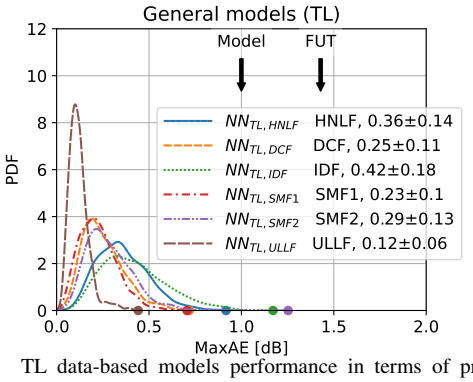


Fig. 16. TL data-based models performance in terms of probability density function (PDF) of the maximum absolute error MaxAE indicating mean (<MaxAE>) and standard deviation (σ) values as <MaxAE>$\pm\sigma$ over the test dataset $\mathcal{D}_{FUT,test,updt}$. In this case, FUT the fiber case used to re-train NN_{num} with just 50 samples.

3.6% of the samples used to train the fiber-specific models and already outperforms NN_{num} accuracy shown in Fig. 14. These results show how just a few cases of the new fiber can be useful to fine-tune NN_{num}. Moreover, it also shows how NN_{num} can provide valuable background knowledge to have an accurate general model for a specific fiber.

Therefore, the proposed general model considering TL can be applied to a totally new optical fiber. For that, we have to perform the following steps, given that the base model NN_{num} is already trained. **1.** measure 50 random experimental data as described in Section IV (∼17 minutes); **2.** use 20 of these data for the simple procedure described in Section VI-A to extract g_R^{peak} and lumped pump losses $l_{P_{1-4}}$ (<30 minutes); **3.** perform a quick measure for the fiber attenuation coefficients (α_s and α_p) using an OTDR, as also described in Section VI-A (<20 minutes); **4.** use these 50 experimental data to re-train the base model NN_{num} (∼2 minutes considering hyperparameter optimization). This entire process takes less ∼1 hour per new optical fiber. In contrast with the 8.5 hours to measure the experimental training data to build the fiber-specific models.

TABLE VIII
Neural network parameters for the TL data-based models

TL training size ⟶	50		200	
Hyperparameter ⟶	*opt*	*LR*	*opt*	*LR*
$NN_{TL,HNLF}$	adagrad[1]	10^{-1}	adamax[2]	10^{-2}
$NN_{TL,DCF}$	adagrad[1]	10^{-1}	adam[4]	10^{-2}
$NN_{TL,IDF}$	adamax[2]	10^{-1}	adadelta[5]	10^{-2}
$NN_{TL,SMF1}$	ftrl[3]	10^{-2}	nadam[6]	10^{-2}
$NN_{TL,SMF2}$	adagrad[1]	10^{-1}	ftrl[3]	10^{-2}
$NN_{TL,ULLF}$	adamax[2]	10^{-2}	adamax[2]	10^{-2}

1. Adagrad: adaptive gradient algorithm,
2. Adamax: adaptive moment estimation (Adam) variant based on the infinity norm,
3. ftrl: follow the regularized leader,
4. Adam: adaptive moment estimation,
5. Adadelta: adaptive gradient algorithm (Adagrad) extension, LR adaptation based on a moving window of gradient updates,
6. Nadam: Nesterov accelerated adaptive moment estimation (Adam).

VIII. Summarized Results and Discussions

Fig. 17 summarizes the performance of all NN models evaluated in this work. Fitting errors from Fig. 11 are also plotted in Fig. 17 as a grey dashed line for the mean a grey shadow for the standard deviation. This is done to recall that the performance of the numerical data-based models is limited by them. The specific experimental data-based models have the best performance. The general models considering the experimental data have the worst performance for most cases. The accurate result for the SMF cases is due to the fact that these fiber cases share most of their parameters with other fibers used for training the general model.

Even limited by the numerical-experimental fitting errors, most cases have improvements in mean and maximum MaxAE for the numerical data-based models. General NN models for the lumped (NN_{lump}) and distributed (NN_{dist}) RAs have the same performance as a single general NN model NN_{num}. Comparing experimental and numerical data-based general models, the latter is able to improve the <MaxAE> of the fibers with the poorest generalization properties by Δ<MaxAE>=-2.3 dB (HNLF) and Δ<MaxAE>=-2.4 dB (DCF) (see Fig. 17(a) and (b), respectively). This comes with the cost of degrading the performance of the SMF1 and SMF2 cases by Δ<MaxAE>=+0.3 dB (SMF1, Fig. 17(d)) and Δ<MaxAE>=+0.15 dB (SMF2, Fig. 17(e)) due to numerical-experimental fitting errors limitations. This is a small degradation when compared to the HNLF and DCF improvements.

Finally, applying TL to the numerical data-based model has significantly improved its performance, achieving the same gain prediction accuracy as for the fiber-specific models if we consider 200 experimental cases for the re-training. In Fig. 17 we compare both, 50 and 200 experimental training sizes, corresponding to 3.6% and 14.3% of the data considered for the fiber-specific models training, respectively. This shows that with just 3.6% of the data needed for an accurate fiber-specific model, the re-trained general model outperforms its previous accuracy. Moreover, only applying TL (or the undesirable fiber–specific models) it is possible to overcome the fitting errors between experimental and numerical measurements.

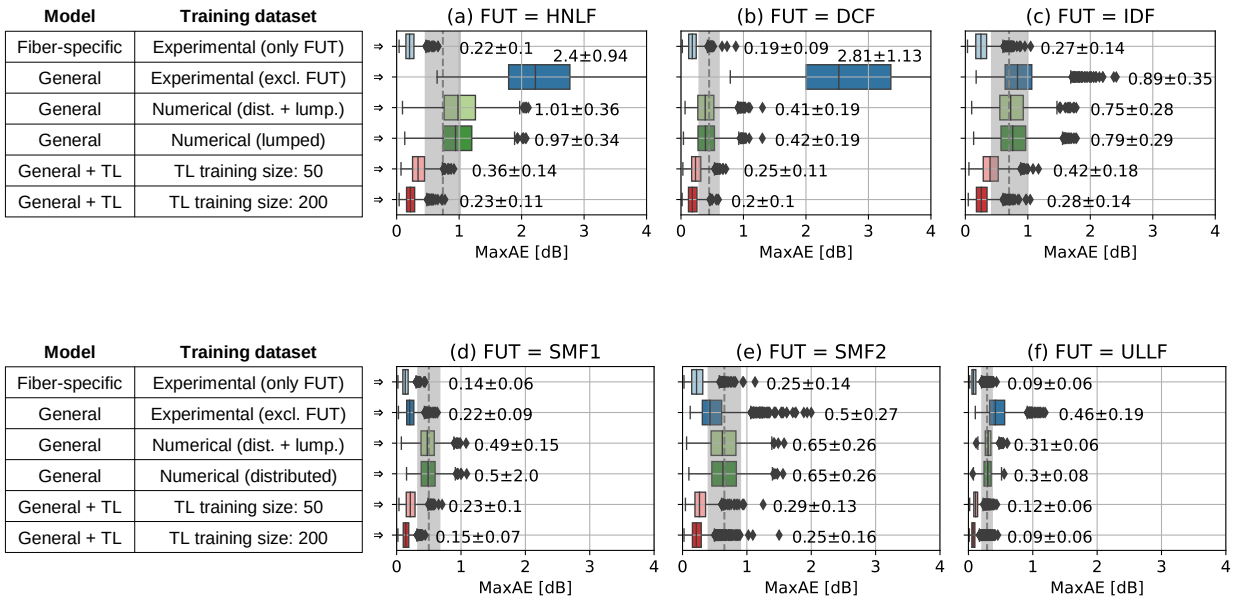


Fig. 17. Maximum absolute errors (MaxAE) bar plots for all models tested over different fiber cases: (a) HNLF, (b) DCF, (c) IDF, (d) SMF1, (e) SMF2, and (f) ULLF. Mean (<MaxAE>) and standard deviation (σ) values are shown as <MaxAE>$\pm\sigma$. Fitting errors between experimental and numerical measurements (Fig. 11) are plotted as a grey dashed lines for the mean and grey shadow for the standard deviation.

In this work, the proposed framework is applied to the direct mapping $f(\mathbf{X}) = \mathbf{Y}$. But it could be extended for the inverse mapping $f^{-1}(\mathbf{Y}) = \mathbf{X}$ as well.Additionally, as inter-channel and inter-pump stimulated Raman scattering effects are both modeled and experimentally measured, the framework could directly extend to ultra-wideband amplifiers as shown for fiber-specific models in [8] and [35].

IX. Conclusion

In the first part of this paper, we summarized our previous works on the neural network-based RA models. We showed that very accurate Raman gain predictions are achieved when applying fiber-specific models based on experimental data. We also showed that neural network models can generalize to different optical fibers. This is done by incorporating the fiber parameters as additional inputs to the neural network. These so-called *general models* are able to predict the gain profile of a new fiber not considered in the training stage with a maximum prediction error as low as 0.22 dB (averaged over 1500 testing samples). However, this is only true when the parameters of the new fiber have values within the parameters range considered during training and have enough samples around them. Fiber cases that do not fulfill this requirement have a maximum prediction error of up to 2.81 dB (also averaged over 1500 testing samples). This could certainly be improved by considering a wide range of optical fibers parameters in the experimental data, but it would require several experimental measurements to account for the broad variety of optical fiber types commercially available.

In the second part, we proposed an upgrade of the general models by generating the data numerically. This numerical data covered a wide range of optical fiber parameters uniformly distributed. General models based purely on numerical data have better generalization capabilities when compared to the general models based on experimental data, with improvements of up to 2.4 dB in the averaged maximum prediction error (also over 1500 testing samples). We saw that the performance of the numerical general models was limited by the errors between experimental and numerical data. To reduce these errors, we propose the use of transfer learning to re-train the numerical general models considering just a few experimental samples. After transfer learning, the general models were able to provide very similar gain prediction accuracy as to the fiber-specific models with just 3.6% of the training data. Finally, we demonstrate a complete procedure to fine-tune a general model for a specific fiber case based on a few measurements that take around 1 hour in total.

ACKNOWLEDGMENT

The authors would like to thank Dr. Md. Asif Iqbal for providing the fiber profile data, Dr. Lars Grüner-Nielsen for fruitful discussions on the Raman gain coefficient, and Lasse Glente and Christian Koefoed Schou for measuring the fiber attenuation coefficient.

REFERENCES

[1] A. Giorgetti, A. Sgambelluri, R. Casellas, R. Morro, A. Campanella, and P. Castoldi, "Control of open and disaggregated transport networks using the Open Network Operating System (ONOS) [Invited]," *J. Opt. Commun. Netw.*, vol. 12, no. 2, pp. A171–A181, 2020.

[2] A. D'Amico, S. Straullu, G. Borraccini, E. London, S. Bottacchi, S. Piciaccia, A. Tanzi, A. Nespola, G. Galimberti, S. Swail, and V. Curri, "Enhancing Lightpath QoT Computation With Machine Learning in Partially Disaggregated Optical Networks," *IEEE open j. Commun. Soc.*, vol. 2, pp. 564–574, 2021.

[3] Y. Pointurier, "Design of Low-Margin Optical Networks," *J. Opt. Commun. Netw.*, vol. 9, no. 1, pp. A9–A17, 2017.

[4] J. Lu, Q. Fan, G. Zhou, L. Lu, C. Yu, A. P. T. Lau, and C. Lu, "Automated training dataset collection system design for machine learning application in optical networks: an example of quality of transmission estimation," *J. Opt. Commun. Netw.*, vol. 13, no. 11, pp. 289–300, Nov 2021.

[5] L. Rapp and M. Eiselt, "Optical amplifiers for multi–band optical transmission systems," *J. Lightw. Technol.*, vol. 40, no. 6, pp. 1579–1589, 2022.

[6] M. Islam, "Raman amplifiers for telecommunications," *IEEE J. Sel. Top. Quantum Electron.*, vol. 8, no. 3, pp. 548–559, 2002.

[7] J. Bromage, "Raman amplification for fiber communications systems," *J. Lightw. Technol.*, vol. 22, no. 1, pp. 79–93, 2004.

[8] U. C. de Moura, M. A. Iqbal, M. Kamalian, L. Krzczanowicz, F. Da Ros, A. M. R. Brusin, A. Carena, W. Forysiak, S. Turitsyn, and D. Zibar, "Multi–Band Programmable Gain Raman Amplifier," *J. Lightw. Technol.*, vol. 39, no. 2, pp. 429–438, 2021.

[9] GNPy. [Online]. Available: DOI: 10.5281/zenodo.3458320, https://github.com/Telecominfraproject/oopt-gnpy

[10] A. Ferrari, M. Filer, K. Balasubramanian, Y. Yin, E. L. Rouzic, J. Kundrát, G. Grammel, G. Galimberti, and V. Curri, "GNPy: an open source application for physical layer aware open optical networks," *J. Opt. Commun. Netw.*, vol. 12, no. 6, pp. C31–C40, Jun 2020.

[11] M. P. Yankov, P. M. Kaminski, H. E. Hansen, and F. Da Ros, "SNR Optimization of Multi-Span Fiber Optic Communication Systems Employing EDFAs With Non-Flat Gain And Noise Figure," *J. Lightw. Technol.*, vol. 39, no. 21, pp. 6824–6832, 2021.

[12] Y. You, Z. Jiang, and C. Janz, "Machine Learning-Based EDFA Gain Model," in *Proc. Eur. Conf. Opt. Commun.*, 2018, paper Mo3E-5.

[13] F. Da Ros, U. C. de Moura, and M. P. Yankov, "Machine learning-based EDFA Gain Model Generalizable to Multiple Physical Devices," in *Proc. Eur. Conf. Opt. Commun.*, 2020, paper Tu1A-4.

[14] J. Yu, S. Zhu, C. L. Gutterman, G. Zussman, and D. C. Kilper, "Machine-learning-based EDFA gain estimation [Invited]," *J. Opt. Commun. Netw.*, vol. 13, no. 4, pp. B83–B91, 2021.

[15] E. de Andrade Barboza, A. A. B. da Silva, J. C. P. Filho, M. J. da Silva, C. J. A. Bastos-Filho, and J. F. Martins-Filho, "Optical amplifier response estimation considering non-flat input signals characterization based on artificial neural networks," *J. Lightw. Technol.*, vol. 39, no. 1, pp. 208–215, 2021.

[16] J. Zhou *et al.*, "Robust, Compact, and Flexible Neural Model for a Fiber Raman Amplifier," *J. Lightw. Technol.*, vol. 24, no. 6, p. 2362, 2006.

[17] M. Ionescu, "Machine Learning for Ultrawide Bandwidth Amplifier Configuration," in *Proc. 21st Int. Conf. Transparent Opt. Netw.*, 2019, paper We.B7.3.

[18] A. M. Rosa Brusin, U. C. de Moura, V. Curri, D. Zibar, and A. Carena, "Introducing Load Aware Neural Networks for Accurate Predictions of Raman Amplifiers," *J. Lightw. Technol.*, vol. 38, no. 23, pp. 6481–6491, 2020.

[19] X. Ye, A. Arnould, A. Ghazisaeidi, D. L. Gac, and J. Renaudier, "Experimental Prediction and Design of Ultra-Wideband Raman Amplifiers Using Neural Networks," in *Proc. Opt. Fiber Commun. Conf.*, 2020, paper W1K.3.

[20] U. C. de Moura, A. M. Rosa Brusin, A. Carena, D. Zibar, and F. Da Ros, "Simultaneous gain profile design and noise figure prediction for Raman amplifiers using machine learning," *Opt. Lett.*, vol. 46, no. 5, pp. 1157–1160, 2021.

[21] C. Mineto, L. S. Schanner, L. G. M. Riveros, J. H. da Cruz Jr., F. D. Simões, T. Sutili, F. B. F. Lopes, R. C. Figueiredo, and E. Conforti, "Performance of artificial-intelligence-based modelling for distributed raman amplification," in *Proc. SBMO/IEEE MTT-S Int. Microw. Optoelectron. Conf.*, 2021, paper R0107-1.

[22] M. Ionescu, A. Ghazisaeidi, J. Renaudier, P. Pecci, and O. Courtois, "Design Optimisation of Power-Efficient Submarine Line through Machine Learning," in *Proc. Conf. Lasers Electro-Opt.* Optical Society of America, 2020, paper STh4M.5.

[23] M. P. Yankov, U. C. de Moura, and F. Da Ros, "Power evolution modeling and optimization of fiber optic communication systems with edfa repeaters," *J. Lightw. Technol.*, vol. 39, no. 10, pp. 3154–3161, 2021.

[24] A. Mahajan, K. Christodoulopoulos, R. Martínez, S. Spadaro, and R. Muñoz, "Modeling EDFA Gain Ripple and Filter Penalties With Machine Learning for Accurate QoT Estimation," *J. Lightw. Technol.*, vol. 38, no. 9, pp. 2616–2629, 2020.

[25] Y. Pointurier, "Machine learning techniques for quality of transmission estimation in optical networks," *J. Opt. Commun. Netw.*, vol. 13, no. 4, pp. B60–B71, 2021.

[26] J. Bromage, K. Rottwitt, and M. Lines, "A method to predict the raman gain spectra of germanosilicate fibers with arbitrary index profiles," *IEEE Phot. Tech. Lett.*, vol. 14, no. 1, pp. 24–26, 2002.

[27] U. C. de Moura, F. Da Ros, A. M. R. Brusin, A. Carena, and D. Zibar, "Experimental Characterization of Raman Amplifier Optimization Through Inverse System Design," *J. Lightw. Technol.*, vol. 39, no. 4, pp. 1162–1170, 2021.

[28] U. C. de Moura, D. Zibar, A. M. Rosa Brusin, A. Carena, and F. Da Ros, "Generalization Properties of Machine Learning-based Raman Models," in *Proc. Opt. Fiber Commun. Conf.*, 2021, paper Th1A.28.

[29] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *J. Big Data*, vol. 3, no. 9, pp. 2196–1115, 2016.

[30] B. Correia, R. Sadeghi, E. Virgillito, A. Napoli, N. Costa, J. ao Pedro, and V. Curri, "Power control strategies and network performance assessment for C+L+S multiband optical transport," *J. Opt. Commun. Netw.*, vol. 13, no. 7, pp. 147–157, Jul 2021.

[31] G. Borraccini, S. Staullu, S. Piciaccia, A. Tanzi, G. Galimberti, and V. Curri, "Cognitive Raman Amplifier Control Using an Evolutionary Optimization Strategy," *IEEE Phot. Tech. Lett.*, vol. 34, no. 4, pp. 223–226, 2022.

[32] C. Headley and G. Agrawal, *Raman amplification in fiber optical communication systems*. Academic press, 2005.

[33] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[34] A. M. R. Brusin, V. Curri, D. Zibar, and A. Carena, "An ultra-fast method for gain and noise prediction of Raman amplifiers," in *Proc. Eur. Conf. Opt. Commun.*, 2019, paper Th.1.C.3.

[35] D. Zibar *et al.*, "Inverse System Design Using Machine Learning: The Raman Amplifier Case," *J. Lightwave Technol.*, vol. 38, no. 4, pp. 736–753, 2020.

[36] M. A. Iqbal, M. A. Z. Al-Khateeb, L. Krzczanowicz, I. D. Phillips, P. Harper, and W. Forysiak, "Linear and Nonlinear

Noise Characterisation of Dual Stage Broadband Discrete Raman Amplifiers," *J. Lightw. Technol.*, vol. 37, no. 14, pp. 3679–3688, 2019.
[37] L. Grüner-Nielsen, M. Wandel, P. Kristensen, C. Jorgensen, L. V. Jorgensen, B. Edvold, B. Pálsdóttir, and D. Jakobsen, "Dispersion-compensating fibers," *J. Lightw. Technol.*, vol. 23, no. 11, p. 3566, 2005.
[38] Public API for TensorFlow (Keras) Optimizers. [Online]. Available: https://www.tensorflow.org/api_docs/python/tf/keras
[39] G.-B. Huang, D. H. Wang, and Y. Lan, "Extreme learning machines: a survey," *Int. J. Mach. Learn. Cyb.*, vol. 2, no. 2, pp. 107–122, 2011.
[40] S. Haykin, *Neural Networks and Learning Machines*. Pearson Education, 2009.
[41] TeraWave SCUBA 150 Ocean Optical Fiber. [Online]. Available: OFS Fitel, LLC, https://stage.fiber-optic-catalog.ofsoptics.com/documents/pdf/TeraWave-Scuba-150-Ocean-Fibers-fiber-168-web.pdf
[42] AllWave Optical Fiber - Zero Water Peak. [Online]. Available: OFS Fitel, LLC, https://fiber-optic-catalog.ofsoptics.com/documents/pdf/AllWave-117-web.pdf
[43] Standard Highly Non-Linear Optical Fiber. [Online]. Available: OFS Fitel, LLC, https://fiber-optic-catalog.ofsoptics.com/documents/pdf/HNLF-Standard-Highly-Non-Linear-Optical-Fiber-web.pdf
[44] S. Koziel and X.-S. Yang, *Computational optimization, methods and algorithms*. Springer, 2011, vol. 356.