Machine learning regression for QoT estimation of unestablished lightpaths

Comparison of Domain Adaptation and Active Learning Techniques for Quality of Transmission Estimation with Small Sized Training Datasets [Invited]

DARIO AZZIMONTI[1], CRISTINA ROTTONDI[2,*], ALESSANDRO GIUSTI[1], MASSIMO TORNATORE[3], AND ANDREA BIANCO[2]

[1] *Dalle Molle Institute for Artificial Intelligence, Lugano, Switzerland*
[2] *Department of Electronics and Telecommunications, Politecnico di Torino, Turin, Italy*
[3] *Department of Electronics, Information and Bioengineering, Politecnico di Milano, Milan, Italy*
[*] *Corresponding author: cristina.rottondi@polito.com*

Compiled November 11, 2021

Machine Learning (ML) is currently being investigated as an emerging technique to automate Quality of Transmission (QoT) estimation during lightpath deployment procedures in optical networks. Even though the potential network-resource savings enabled by ML-based QoT estimation has been confirmed in several studies, some practical limitations hinder its adoption in operational network deployments. Among these, the lack of a comprehensive training dataset is recognized as a main limiting factor, especially in the early network deployment phase. In this study we compare the performance of two ML methodologies explicitly designed to augment small-sized training datasets, namely, active learning (AL) and domain adaptation (DA), for the estimation of the signal to noise ratio (SNR) of an unestablished lightpath. This comparison also allows us to provide some guidelines for the adoption of these two techniques at different life stages of a newly-deployed optical-network infrastructure. Results show that both AL and DA permit, starting from limited data sets, to reach a QoT-estimation capability similar to that achieved by standard supervised-learning approaches working on much larger datasets. More specifically, we observe that few dozens of additional samples acquired from selected probe lightpaths already provide significant performance improvement for AL, whereas few hundreds of samples gathered from a external network topology are needed in the case of DA.

http://dx.doi.org/10.1364/ao.XX.XXXXXX

1. INTRODUCTION

Several supervised Machine Learning (ML) algorithms for Quality of Transmission (QoT) estimation of unestablished lightpaths have recently been investigated [1] to offer alternatives to traditional QoT-estimation methods. In fact, traditional methods, typically based on approximated mathematical models (such as the Gaussian Noise model in [2]) or on direct simulation of the optical signal propagation, present some limitations. Approximated mathematical models usually introduce conservative design margins to compensate for uncertainties on input parameters and/or to account for simplifying modeling assumptions, whereas directly simulating optical signal propagation along the fiber core is typically unaffordable if applied in real-time operational scenarios due to its computational complexity. Supervised learning algorithms for QoT estimation leverage a set of historical training samples, which are constituted by a vector of *features* characterizing the lightpath (e.g. path length, the amount of served traffic, the adopted modulation format, etc.) associated with a *target variable* that measures a lightpath's QoT metric, such as the signal to noise ratio (SNR) or the bit error rate (BER). Unfortunately, supervised algorithms require a very large set of training samples, at least in the order of hundreds or even of a few thousands of lightpaths' observations [3], to learn an accurate prediction model. SNR and/or BER are assumed to be measured at the receiver and then collected via telemetry equipment [4]. However, in real installations, the collection of training samples is often hindered by practical limitations such as absence of optical monitors in some network nodes or scarcity of monitorable lightpaths in the case of networks in their early deployment phase.

When the number of available training samples is limited, two solutions can be applied to enlarge the training dataset and improve the prediction capability of a ML-based QoT estimator.

The first approach consists in acquiring a limited number of additional samples via dedicated probe lightpaths, i.e., light-

paths that do not transport user traffic and are deployed with the only objective of monitoring their QoT. In this case, one can leverage the so called *Active Learning* (AL), a special ML technique in which the learning algorithm can explicitly request the labeling of new data points to optimize performance. Hence, in our context, AL algorithms can be used to cleverly select the training instances (route and transmission configurations of the probe lightpaths) to be acquired to improve the model prediction capability, as done in our previous work [5]. To the best of our knowledge, the only other study [6] adopting AL for optical network management compares three pool-based AL selection processes for collecting training data points of an Erbium Doped Fiber Amplifier (EDFA) model aimed at estimating the gain of a single channel, given a set of binary features indicating whether each input channel is in use. In this paper, we work with both continuous and discrete features in small to moderate training-set dimensions, and we use Gaussian processes as predictor. Such predictor is a more powerful non-parametric method than the linear models used for example in [6]. Indeed, the new training instances are selected by minimizing an acquisition function tailored on the Gaussian process model, which is not analytical as in [6], but that can be easily optimized.

The second viable solution consists in exploiting additional "external" training data collected from a different network, and use *Trasfer Learning* (TL) [7] to extract new knowledge from such data. Also TL is a special case of ML that permits to improve the performance of a ML model by training it with samples gathered from a *source domain* that is different from the *target domain* where the model is actually tested and used. In the context of QoT estimation, a few recent papers applied TL methodologies to shorten the training time of (deep) neural networks, by initializing their weights with values acquired from a model previously trained with data extracted from a different network [8–12]. In this paper we focus on a specific subset of TL named *Domain Adaptation* (DA) [13, 14], in which the source and target domains share the same feature space. In particular, we use a DA technique [15] which operates a domain-dependent transformation of the features in a pre-processing phase.

In both AL and DA, acquiring additional samples comes at a price, because *i*) AL needs lightpath probes which require dedicated transmission equipment and occupy spectral resources, and *ii*) DA implies acquiring external samples from another network, possibly managed by a different operator. Hence, the choice among the two approaches is driven both by performance criteria such as QoT-prediction accuracy, and by the cost of collecting data.

In this study, we provide a comparison of the QoT-estimation accuracy achieved by some AL and DA methods on two different network topologies, as a function of the number of additional training instances that can be acquired. A preliminary performance comparison for AL and DA approaches appeared in our study in [16]. In that work, we consider QoT estimation achieved through *classification*, i.e., predicting whether the BER of a candidate lightpath exceeds a given system threshold. Differently from that study, we now consider QoT estimation achieved through *regression* on the SNR values (similarly to what already done in [17–19]), which consists in directly estimating the value of the SNR (and not only classifying if the SNR is above a given threshold). We were motivated to move from classification to regression because a direct estimation of SNR provides more complete information regarding the QoT of a lightpath, as it makes possible to evaluate how close that SNR is to the system threshold. Moreover, in our study, by comparing AL and DA on a common set of assumptions (fixing, e.g., the same network topology and physical-layer modeling), we can also provide guidelines for the adoption of DA and AL techniques (or even of a combination of the two techniques) at different life stages of a newly-deployed optical-network infrastructure. Finally, we provide a cost-performance tradeoff analysis of the AL and DA approaches, depending on the costs that the network operator would incur when deploying a probe lightpath or acquiring external samples from a different network.

The remainder of this paper is organized as follows: we present the adopted AL and DA approaches in Sec.2. In Sec.3 we describe the considered network scenario, highlighting the specific stages of the network life cycle when each technique finds more useful application. An illustrative numerical evaluation is provided in Sec.4. Sec.5 concludes the paper.

2. ACTIVE LEARNING AND DOMAIN ADAPTATION APPROACHES FOR QOT ESTIMATION

In this Section we first provide some background notions on Gaussian Processes (the ML model used throughout the entire study) and then we discuss the approaches used to incorporate AL and DA in a Gaussian Process model. Note that, for both AL and DA approaches, we assume that the feature vector that describes a lightpath in the datasets contains: total lightpath length, maximum link length (i.e. the sum of the lengths of the consecutive fiber spans constituting the longest link among the ones belonging to the considered lightpath), number of traversed links, amount of transmitted traffic and modulation format adopted for transmission (such end-to-end features have been widely adopted in studies on ML approaches for QoT estimation, e.g. in [3, 20, 21]). Optionally, the following additional features can be included to characterize the lightpaths' neighbor channels: traffic volume, modulation format and guardband size of the spectrally-nearest right and left adjacent channels co-propagating along at least one of the links traversed by the considered lightpath. Each lightpath sample is associated with the SNR value measured at the receiver node.

A. Regression Model based on Gaussian Processes

We consider a Gaussian-Processes (GPs) model [22], a probabilistic non-parametric learning algorithm which provides both a prediction and a quantification of its uncertainty. This learning model is preferred to other models (e.g. neural networks) because of its reliable and fast-to-compute uncertainty quantification, which is necessary to implement AL approaches, as it allows to explore the input space more efficiently. A GP model outputs a prediction of the associated SNR value when receiving at the input the features of a candidate lightpath $\mathbf{x}^* \in \mathbb{X} \subset \mathbb{R}^5$. In particular, we observe a training set of ℓ points $\mathbf{X}_\ell = \{\mathbf{x}_1, \ldots, \mathbf{x}_\ell\} \subset \mathbb{X}$, coupled with ℓ response values $\mathbf{y} = (y_1, \ldots, y_\ell)^T \in \mathbb{R}^\ell$ where

$$y_i = f(\mathbf{x}_i) + \varepsilon \tag{1}$$

for $\mathbf{x}_i \in \mathbb{X}$, $i = 1, \ldots, \ell$ with a measurement error $\varepsilon \sim N(0, \sigma^2_{\text{noise}})$. Eq. (1) means that we assume that SNR observations are generated by a latent function f and corrupted by a small measurement noise ε. This observation model permits to handle data points where two observations with the same input features produce slightly different SNR values, to increase data set variability. We denote by $\mathbf{f} = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_\ell)) \in \mathbb{R}^\ell$ the latent function values. The observation model described in Eq. (1) can be summarized as $p(\mathbf{y} \mid \mathbf{f}) = N(\mathbf{f}, \sigma^2_{\text{noise}} I_\ell)$, where $I_\ell \in \mathbb{R}^{\ell \times \ell}$

is the identity matrix. Note that for the problem at hand, $\mathbf{x}_i$ would be a vector describing the ith lightpath features in the training set, coupled with y_i, the (noisy) SNR value observed for that lightpath. The vector $\mathbf{x}_i$ is considered here as a vector of real values in $[0,1]^5$. The j-th ordinal feature is thus transformed with a linear transformation that maps each $x^j \in [min_j, max_j]$ into $\tilde{x}^j \in [0,1]$ as $\tilde{x}^j = (x^j - min_j)/(max_j - min_j)$, where min_j, max_j are the minimum and the maximum values possible for the feature j. In particular for the modulation format the number of constellation points are first transformed with a base two logarithm.

GP regression assumes that the latent function f is a realization of a Gaussian process. Thus, in a Bayesian sense, we assume that the latent vector $\mathbf{f}$ has a prior distribution given by $p(\mathbf{f}) = N(m(\mathbf{X}_\ell), K)$, with $m(\mathbf{X}_\ell) = (m(\mathbf{x}_1), \ldots, m(\mathbf{x}_\ell))^T \in \mathbb{R}^\ell$ and K a positive definite matrix with elements $K_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$. The function k is a positive definite kernel which, along with m, the mean function, defines the Gaussian process.

The GP mean function m and kernel k are chosen before observing the data, and therefore encode our prior knowledge. The prior mean is an arbitrary function that encodes trends of the latent function f known before observing any data. In our scenario, since no known trends are available, we use a zero mean function. The kernel determines the smoothness of the GP regression fit and can be used to encode prior knowledge on f. For example, if the latent function is known to be periodic, we can choose a periodic kernel and all prior realizations of the GP will be periodic functions. However, the SNR function does not have, a priori, specific properties; thus, we choose a kernel from a stationary family, so as to have a dependency on a few hyper-parameters tuned from data. In particular, we choose a kernel from the Matérn family, with smoothness parameter equal to $\nu = 3/2$ (see [5] for more details on this choice and a discussion on alternative options). The other hyper-parameters of this kernel, such as characteristic length-scales for each input and kernel variance, are chosen by maximizing the marginal likelihood of the model (see [22], chapter 2).

The prior and the observation model in Eq. (1) can be combined with Bayes theorem to obtain the posterior distribution.

$$p(\mathbf{f} \mid \mathbf{y}) = \frac{p(\mathbf{f})p(\mathbf{y} \mid \mathbf{f})}{p(\mathbf{y})}.$$

In GP regression, the posterior distribution $p(\mathbf{f} \mid \mathbf{y})$ has the remarkable property of being normally distributed, with analytical expressions for its mean and covariance (see [22], chapter 2). This means that no sampling is needed to compute the posterior distribution. However, because the analytical formulae have a complexity of $O(\ell^3)$, they become practically infeasible for large datasets. This issue is often not problematic in AL applications because the size of the training set is intrinsically limited by the cost of acquiring new samples.

In the following subsections, we describe the applied AL and DA methods, and define baselines to be used as benchmarks.

B. Active Learning Technique

We assume that only a small training dataset $\mathbf{T}$ containing lightpath samples from the network under consideration (referred to in the following as *target domain*) is available. AL aims at adaptively increasing the size of the training dataset by adding new samples that minimize a problem-specific acquisition function. We consider the Integrated Mean Squared Error (IMSE) acquisition function introduced in [23] which associates to any new untried input $\mathbf{x}$ the (integrated) posterior variance of the GP, assuming this new point is added to the training set. Given a training set $\mathbf{X}_\ell$ and a new point $\mathbf{x}_{\ell+1}$, we can compute the posterior variance $s^2_{\ell+1}(\mathbf{x})$ at any input location $\mathbf{x}$ with the formula

$$s^2_{\ell+1}(\mathbf{x}) = s^2_\ell(\mathbf{x}) - \frac{k_\ell(\mathbf{x}, \mathbf{x}_{\ell+1})^2}{k_\ell(\mathbf{x}_{\ell+1}, \mathbf{x}_{\ell+1}) + \sigma^2_{\text{noise}}}, \tag{2}$$

where σ^2_{noise} is the variance of the measurement noise, $s^2_\ell(\cdot)$ and $k_\ell(\cdot,\cdot)$ are the posterior variance and the posterior covariance kernel given $(\mathbf{X}_\ell, \mathbf{y}_\ell)$. The value of the latent function f at the new point $\mathbf{x}_{\ell+1}$ is not needed to compute $s^2_{\ell+1}$; thus, this quantity can be computed before measuring any new SNR value. Furthermore, once k_ℓ and s^2_ℓ are given, the function in Eq. (2) is very fast to evaluate at any $\mathbf{x} \in \mathbb{X}$. The IMSE acquisition function is then defined as the integral over the input domain of the quantity in Eq. (2), i.e.

$$\mathcal{I}_\ell(\mathbf{x}_{\ell+1}) = \int_{\mathbb{X}} s^2_{\ell+1}(\mathbf{x}) d\mathbf{x}. \tag{3}$$

The integral in Eq. (3) is computed with a Monte Carlo method using a discretization of the integral over n_{int} integration points ($n_{int} = 1500$, in our simulations) selected with an importance sampling procedure (see [24] for details).

We can then select the next input location $\mathbf{x}^{AL}_{\ell+1}$ by minimizing the function $\mathcal{I}_\ell$. We can proceed with this minimization in two ways: *i*) we constrain the acquisition function to obtain only feasible samples as in [5] and use a continuous optimizer; *ii*) we pre-select a pool of possible input locations, evaluate $\mathcal{I}_\ell$ at such locations and select the minimizer. Here we select $\mathbf{x}^{AL}_{\ell+1}$ as the minimizer from a pre-selected pool of input locations to have results comparable with those of the DA techniques. In our experiments the pool of input location corresponds to the same training dataset from which samples are selected to estimate the quantities required by DA algorithms. We increase the size of the training set $\mathbf{T}_0 = \mathbf{T}$ with an iterative procedure where at each step ℓ:

1. the GP-based model is trained with $\mathbf{T}_\ell = \mathbf{T}_0 \cup \{(\mathbf{x}^{AL}_1, y_1), \ldots, (\mathbf{x}^{AL}_\ell, y_\ell)\}$, where y_ℓ is the SNR value obtained from a probe with features $\mathbf{x}^{AL}_\ell$;
2. the trained model and, in particular, the posterior variance s_ℓ is used to build the acquisition function $\mathcal{I}_\ell$ as described in Eq. (3);
3. we minimize $\mathcal{I}_\ell$ to select the next inputs $\mathbf{x}^{AL}_{\ell+1}$;
4. we estimate the SNR value $y_{\ell+1}$ with a probe that has features $\mathbf{x}^{AL}_{\ell+1}$ and we update the training set with $\mathbf{T}_{\ell+1} = \mathbf{T}_\ell \cup \{(\mathbf{x}^{AL}_{\ell+1}, y_{\ell+1})\}$;

The procedure is repeated until the number of acquirable samples is exhausted.

C. Domain Adaptation Techniques

In the case of DA techniques, in addition to the training set $\mathbf{T}$ of the target domain, we leverage a second training set $\mathbf{S}$ obtained from a different network (named as *source domain*). We consider two supervised DA techniques (which rely on the availability of labels for $\mathbf{T}$), and one unsupervised DA technique (which does not need labels for $\mathbf{T}$). We briefly describe below the three considered DA techniques.

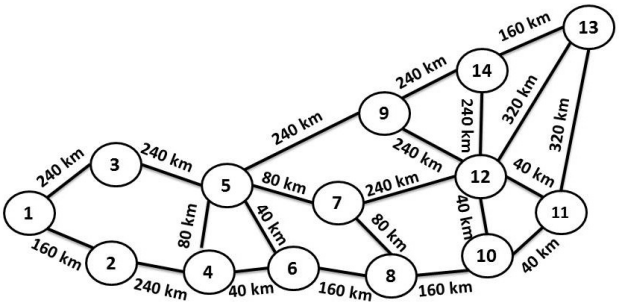

Fig. 1. Japan network topology. Nodes are labeled using integer numbers.

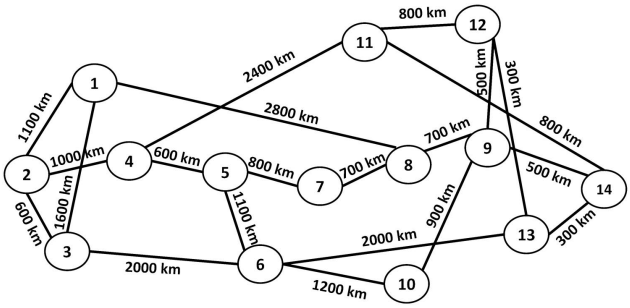

Fig. 2. NSF network topology. Nodes are labeled using integer numbers.

C.1. Bayesian updating

Bayesian Updating (BU) consists in using a model trained on the source dataset $\mathbf{S}$ as prior model and updating it with the data from the target domain $\mathbf{T}$. This is a basic supervised DA technique which is computationally reasonable with GP models because they allow for analytical updates of the posterior distribution (see, e.g., [25]), so the distribution can be updated exactly at reasonable costs.

C.2. Feature Augmentation

Feature Augmentation (FA) [15] is a supervised DA technique that encodes the domain of a sample by augmenting its feature vector. More in detail, the length of each feature vector $\mathbf{x}$ is tripled with a rule that depends on the domain, with the aim of capturing both the commonalities between the two domains and the unique characteristics of each domain. If the sample comes from $\mathbf{S}$, the tripled feature vector is defined as $\mathbf{x}' = \langle \mathbf{x}, \mathbf{x}, \mathbf{0} \rangle$; whereas if the sample comes from $\mathbf{T}$, $\mathbf{x}' = \langle \mathbf{x}, \mathbf{0}, \mathbf{x} \rangle$. This augmentation is applied to all samples, both for training and inference.

C.3. Correlation Alignment

Correlation Alignment (CORAL) [26] is an unsupervised DA technique that transforms the features in $\mathbf{S}$ to match the second-order statistics of the features in $\mathbf{T}$. Because of the difference in the domains, the instances in $\mathbf{S}$ are contained in a different manifold of the space of features than the ones in $\mathbf{T}$: a model learned on $\mathbf{S}$ will therefore underperform on the target domain. CORAL applies a transformation ϕ that re-colors the whitened features of the samples in $\mathbf{S}$ with the covariance matrix estimated from the feature distribution of the samples in $\mathbf{T}$; the model is learned on the transformed data. Because estimating such covariance matrix does not require information about the labels of samples in $\mathbf{T}$, the approach is unsupervised and we use the notation $\mathbf{T}_{\text{unlabeled}}$.

Note that, in our application scenarios, generating $\mathbf{T}_{\text{unlabeled}}$ simply requires to select the routes and transmission configurations of a large set of potential lightpaths, without measuring their SNR. Since generating feature vectors associated to perspective lightpath configurations comes at no cost, the cardinality of set $\mathbf{T}_{\text{unlabeled}}$ is assumed to be large. The method estimates the transformation ϕ from $\mathbf{S}$ to $\mathbf{T}_{\text{unlabeled}}$, then trains the prediction model on $\phi(\mathbf{S})$.

D. Benchmarks

In the result section, we will also consider the following three baselines:

- *Source Domain Baseline* (SDB) trains the regressor only on $\mathbf{S}$;
- *Reduced Target Domain Baseline* (RTDB) trains the regressor only on $\mathbf{T}$;
- *Large Target Domain Baseline* (LTDB) trains the regressor on $\mathbf{T}'$ containing a larger number of samples from the target domain (i.e., $|\mathbf{T}'| \gg |\mathbf{T}|$).

3. WHEN TO APPLY AL/DA DURING NETWORK LIFECYCLE

We now discuss how the DA and AL techniques described in the previous section can be applied during the early life-stages of an optical network. To this aim, we consider a newly deployed optical network and we assume that at time t_0 the network is completely empty. When the first lightpath request has to be provisioned, $\mathbf{T} = \oslash$. Hence, in this situation, the viable ML options for QoT estimation are:

- acquire a dataset $\mathbf{S}$ from a different network domain and use it to train the SNR estimator (SDB);
- acquire a dataset $\mathbf{S}$ from a different domain and a dataset $\mathbf{T}_{\text{unlabeled}}$ of unlabeled data from the current network domain and apply Correlation Alignment (CORAL) to train the SNR predictor (note that, as already mentioned, the acquisition of $\mathbf{T}_{\text{unlabeled}}$ is straightforward and comes at no cost, because it consists of a collection of lightpath configurations, without need of assessing their SNR).

Furthermore, if the operator is willing to install probe lightpaths (i.e., lightpaths that do not carry user traffic), AL can be applied to improve the prediction model obtained by both SDB and CORAL. This implies enlarging set $\mathbf{S}$ with samples indicated by the AL algorithm and acquired by dedicated probes. The combination of the above mentioned DA and AL approaches will be indicated in the following as CORAL+AL or SDB+AL.

Alternatively, if an external dataset cannot be acquired (i.e., $\mathbf{S} = \oslash$), the first deployments must rely on traditional SNR prediction methods (e.g., the Gaussian Noise model [27]).

Once a few lightpaths have been deployed within the network, their SNR can be measured by monitors and set $\mathbf{T}$ starts being populated ($0 < |\mathbf{T}| < |\mathbf{T}_0|$). At this stage $t_1 > t_0$, two additional DA options become applicable to improve the performance of the SNR predictor:

- apply the Bayesian Update (BU) on the model previously trained on dataset $\mathbf{S}$ leveraging the new samples from dataset $\mathbf{T}$;
- apply the Feature Augmentation (FA) technique on the datasets $\mathbf{S}$ and $\mathbf{T}$ and use both of them to train the SNR predictor.

Table 1. Taxonomy of the use of DA/AL techniques, depending on the size of the source and target domain datasets

	$\|\mathbf{S}\| = 0$		$\|\mathbf{S}\| > 0$	
Probes	No	Yes	No	Yes
$\|\mathbf{T}\| = 0$	-	-	SDB CORAL	SDB+AL CORAL+AL
$0 < \|\mathbf{T}\| < \|\mathbf{T}_0\|$	-	-	BU, FA	BU, FA
$\|\mathbf{T}\| \geq \|\mathbf{T}_0\|$	RTDB	RTDB+AL	BU, FA	BU, FA
$\|\mathbf{T}\| \gg \|\mathbf{T}_0\|$	LTDB	LTDB	LTDB	LTDB

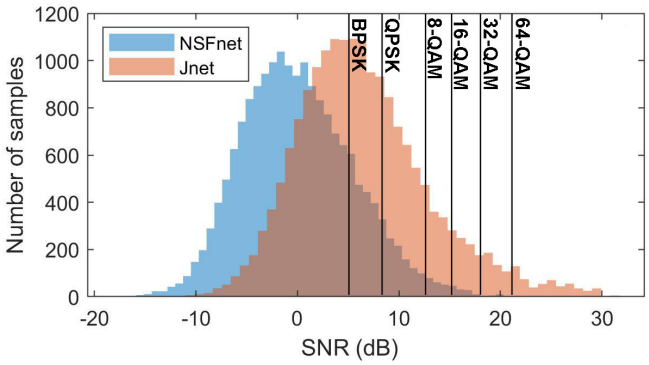

Fig. 3. SNR distribution for 18000 randomly selected lightpaths in Jnet and NSFnet, with SNR thresholds for the considered modulation formats

If no external dataset has been acquired (i.e., $\mathbf{S} = \oslash$), the adoption of AL becomes nevertheless viable when set $\mathbf{T}$ contains a sufficient number of samples ($|\mathbf{T}| \geq |\mathbf{T}_0|$) to boost an initial model training. This case will be indicated as RTDB+AL. Of course, nothing prevents from training the SNR predictor exclusively on $\mathbf{T}$ (RTDB). This latter option can be applied if no external dataset is available and the installation of probes is undesired.

Finally, at time $t_2 > t_1$, when the size of $\mathbf{T}$ becomes large ($|\mathbf{T}| \gg |\mathbf{T}_0|$), it can be expected that training the SNR predictor with dataset $\mathbf{T}$ yields to a sufficiently good estimation performance (LTDB) and AL/DA techniques are no longer necessary.

Tab.1 summarizes the above mentioned scenarios.

4. RESULTS

A. Simulation Settings

We consider a 4 THz wide flexible optical grid with 12.5 GHz slice width and adaptive transceivers operating at 28 Gbaud, each one occupying a 37.5 GHz optical bandwidth. Transceivers adopt a modulation format chosen among dual polarization DP-BPSK, DP-QPSK and DP-n-QAM, with $n = 8, 16, 32, 64$, resulting in supported bit rates of $50, 100, 150, 200, 250$ and 300 Gbps, respectively. Traffic demands exceeding the capacity of a single transceiver are served by optical superchannels containing multiple adjacent transceivers, switched and filtered as a single entity. The minimum guardband size required to separate two spectrally adjacent (super)channels is 12.5 GHz. It follows that the maxiumum number of channels per fiber is 80.

Dataset	BPSK	QPSK	8-QAM	16-QAM	32-QAM	64-QAM
Jnet	50.27	9.90	0.63	0	0	0
NSFnet	92.87	51.03	10.50	7.23	3.27	0.03

Table 2. Percentage (%) of samples exhibiting above-threshold SNR values for each considered modulation format, computed for 18000 randomly selected lightpaths in Jnet and NSFnet

We consider the Japan (Jnet) and NSF (NSFnet) networks depicted in Figs.1 and 2 respectively, and evaluate the performance of the DA and AL approaches presented in Sec.2 in terms of the R2 metric, defined as:

$$R2 = 1 - \frac{\sum_{i=1}^{\ell_{test}} (y_i^{pred} - y_i)^2}{\sum_{i=1}^{\ell_{test}} (\bar{y} - y_i)^2} \quad \textbf{(4)}$$

where $\bar{y} = \frac{1}{\ell_{test}} \sum_{i=1}^{\ell_{test}} y_i$ and ℓ_{test} is the test data size, and of the root mean square error (RMSE), defined as:

$$RMSE = \sqrt{\frac{1}{\ell_{test}} \sum_{i=1}^{\ell_{test}} (y_i^{pred} - y_i)^2} \quad \textbf{(5)}$$

The R2 metric describes the fraction of variance explained by the model with respect to the total variance of the test data. It is always below or equal to one: a perfect prediction achieves an R2 equal to one, whereas a baseline model that always returns $\bar{y}$ yields R2$= 0$; a model with R2$= 0.9$ explains 90% of the variance of the data, and is unable to capture the remaining 10%. The RMSE is also a measure of accuracy of a model, but unlike R2 it is dependent on the scale of the target variable, and thus more difficult to directly relate to the quality of the model; the RMSE is non-negative, and a perfect model achieves RMSE$= 0$.

For each topology, we build a dataset $\mathcal{R}$ including several thousands of samples generated according to the following procedure:

1. randomly choose a source-destination node pair, a modulation format and a traffic demand uniformly selected in the range $[50 - 500]$ Gbps, with 50 Gbps granularity

2. randomly choose one out of the 3 shortest paths from source to destination, perform random spectrum allocation and compute the ligthpath SNR using the E-tool, then repeat from point 1. If routing and spectrum assignment fails, halt and restart from point 1.

3. After 10 consecutive failures in the routing and spectrum assignment procedure, compute the feature vector associated to every deployed lightpath, clear the network from all ligthpath allocations and restart from point 1.

Note that the above described procedure is unlikely to lead to the most efficient lightpath allocation in terms of overall served traffic, average spectral efficiency and spectrum fragmentation and may lead to lightpath configurations exhibiting too high SNR to ensure a sufficiently low BER, but allows for obtaining more diversified feature vectors with respect to other approaches such as choosing the modulation format based on an a-priori estimation of its reach and on the lightpath length, or using first-fit spectrum allocation. This ensures that the dataset generation procedure explores the whole feature space, which is a necessary requirement to train an unbiased prediction model. As discussed in [3, 28], in real scenarios dedicated probe lightpaths should be

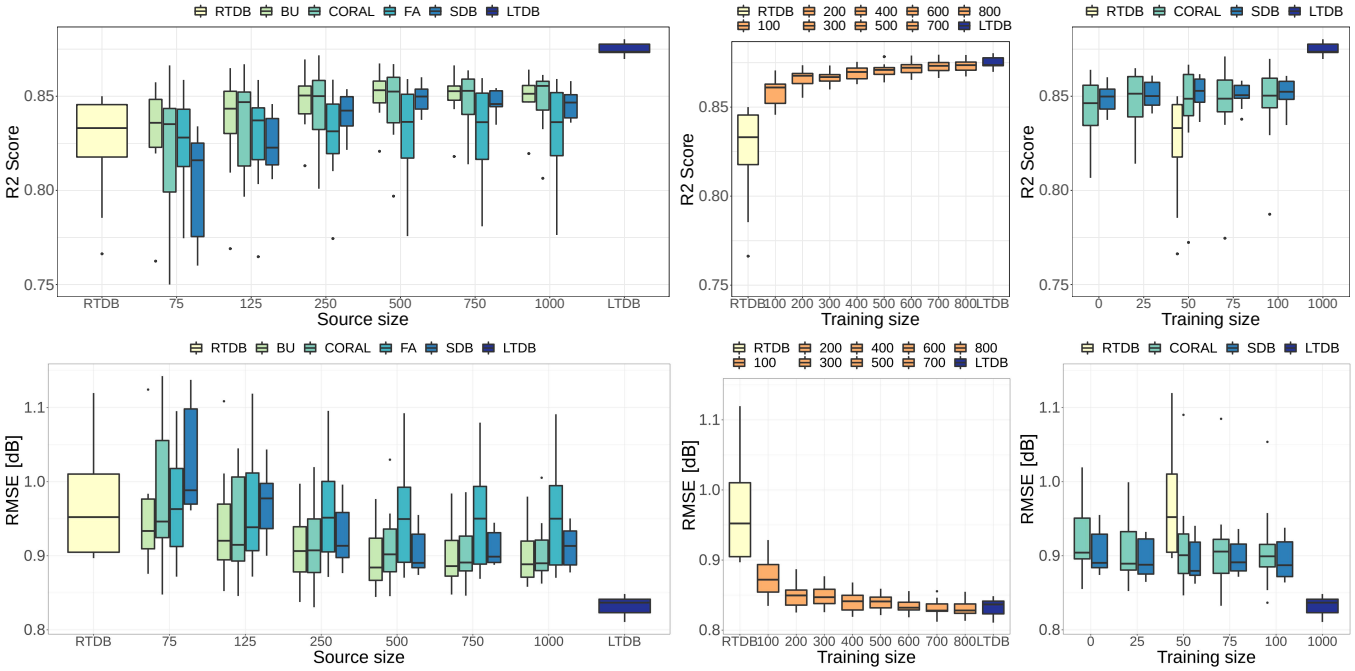

Fig. 4. R2 (top) and RMSE (bottom) obtained by DA (left), AL (center) and DA+AL (right) approaches with target domain NSFnet and source domain Jnet.

deployed to acquire SNR measurements configurations leading to unacceptable BER values.

To generate synthetic SNR data we use the E-tool presented in [3]. For a given candidate lightpath, traffic volume and modulation format (which constitute the feature vector $\mathbf{x}_i$), the E-tool calculates the Signal-to-Noise ratio measured at the channel decoder via the approximated AWGN model of dispersion uncompensated transmission over single mode fibers. In our Signal-to-Noise calculations, we take into account the effect of Amplified Spontaneous Emission (ASE) noise and also the Nonlinear Interference (NLI) noise generated by the Kerr effect due to the propagation of the signal on optical fiber (as introduced in Eqn. (5) in [29]). The SNR is defined as the ratio of the power of the signal (P_{sig}) and the power of the noise (P_{noise}). P_{noise} is composed of the noise power due to the ASE (P_{ase}) and the noise power due to the NLI (P_{nli}), Therefore, SNR can be expressed as:

$$SNR = \frac{P_{sig}}{P_{ase} + P_{nli}} \tag{6}$$

Note also that the nonlinear contribution considered in our E-tool is calculated according to [30].

Therefore, the E-tool implements the function $f(\mathbf{x}_i)$. The E-tool also adds random penalties to account for model uncertainties, according to an exponential distribution with average of 1 dB. Such uncertainties are captured the GP model by the term ε. We assume transparent links of dispersion uncompensated standard single-mode fibers. Fiber spans are assumed to be 100 km long in both topologies (if the length of a given link is not an integer multiple of the span length, the last span constituting the link is assumed to be shorter). Transmission adopts optimal launch power levels per channel (in dBmW) computed according to [2] and the signal power is restored by identical optical amplifiers with 20 dB gain, located at the end of each span. For the Jnet topology, we adopt a fiber attenuation coefficient of 0.25 dB/km and a 7 dB amplifier noise figure, whereas for the NSFnet topology the two values are set to 0.2 dB/km and 5 dB, respectively. By changing the values of fiber attenuation and amplifier noise parameters in the two topologies, we aim at capturing the different characteristics of two networks that leverage different transmission equipment and/or have been differently impacted by ageing. The distribution of the obtained SNR values is reported in Fig.3 for both topologies, as well as the SNR thresholds for each of the considered modulation formats, required to guarantee a pre-FEC BER lower than $4 \cdot 10^{-3}$. In the NSFnet, the SNR ranges between -15 and 20 dB, with a peak around -5 dB, whereas in the Jnet the SNR range is $[-10, 30]$ dB, with a peak around 7 dB. In Table 2, we partition the samples based on the adopted modulation format and, for each of the considered modulation formats, we report the percentage of lightpaths that exhibit above-threshold SNR values in both networks. It emerges that the vast majority of samples leveraging QPSK and 8-QAM and all the samples leveraging 16,34 and 64-QAM need to be acquired by means of probe lightpaths, since such transmission parameter configuration would never be observed in regular lightpaths carrying user traffic. Some initial analysis on the effect of the number of probes on the final accuracy of ML-based QoT estimation can be found in [3] and a more comprehensive study on this topic is left as future work.

Note that in the learning procedures, the SNR values of each topology are standardized to obtain a distribution with zero mean and unitary variance. The mean and standard deviation are computed here from a dataset that does not include test data but includes all data used in the different procedures compared in the benchmark. This choice allows for a comparison that does not depend on the standardization procedure.

Each method is evaluated by computing the R2 and RMSE metrics on 6000 test data which are chosen at random from the original full dataset $\mathcal{R}_{\text{target}}$ before any training. The remaining part of $\mathcal{R}_{\text{target}}$, denoted $\mathcal{R}^{\text{train}}_{\text{target}}$ (12000 points), is used to select points for the training sets and as the pool of pre-selected points

for the acquisition function optimization. The AL and DA methods are validated against the RTDB and LTDB baselines, where LTDB assumes $|\mathbf{T}'| = 1000$ and RTDB assumes $|\mathbf{T}| = 50$ (where $\mathbf{T}, \mathbf{T}' \subset \mathcal{R}_{\text{target}}^{\text{train}}$). We assume $|\mathbf{S}| = 75, 125, 250, 500, 750, 1000$ (where $\mathbf{S} \subset \mathcal{R}_{\text{source}}$). For CORAL only, we assume $\mathbf{T}_{\text{unlabeled}} = 1000$ (where $\mathbf{T}_{\text{unlabeled}}$ contains the feature vectors of 1000 elements in $\mathcal{R}_{\text{target}}^{\text{train}}$). For the AL algorithm, we start from RTDB and add up to 750 new samples and we evaluate R2 and RMSE every 100 iterations. Experiments are repeated 10 times, with random extraction of elements of $\mathbf{S}$ (resp. $\mathbf{T}$) from set $\mathcal{R}_{\text{source}}$ (resp. $\mathcal{R}_{\text{target}}^{\text{train}}$). If not differently stated, the feature vector of each sample includes the five features associated to the lightpath (i.e., total lighpath length, longest link length, number of traversed link, traffic volume and modulation format).

B. Comparison of Prediction Performance

Fig. 4 shows on the top left the R2 values obtained by applying DA techniques versus the cardinality of $\mathbf{S}$, when the target domain is the NSFnet and the source domain is the Jnet. Recall that the average lightpath length in the Jnet is much smaller than in the NSFnet. To correctly interpret Fig.4, keep in mind that: *i*) RTDB (on the extreme left) represents a low performance bound, as it is obtained with limited data from target domain; *ii*) LTDB (on the extreme right) represents a high performance bound, as it is obtained with a large training dataset from target domain; *iii*) all the approaches in the middle represent the possible increase in performance obtained thanks to DA (where DA includes BU, FA and CORAL approaches) w.r.t. the benchmark constituted by SDB. We start by noticing that the R2 median achieved by SDB ranges from 0.816 (when $|\mathbf{S}|$=75) to 0.847 (when $|\mathbf{S}|$=1000). Using the same amount of samples, CORAL outperforms SDB, with a R2 median in the range [0.835-0.856], showing an increase of R2 as the cardinality of $\mathbf{S}$ raises. When $|\mathbf{S}|$=75, the performance of CORAL is comparable to that of RTDB (median 0.833). Thus, with only 75 samples gathered from the Jnet domain (and no samples from the NSFnet domain), CORAL permits to achieve the same prediction capabilities that would be obtained training the GP with 50 samples collected from the NSFnet. Moreover, when $|\mathbf{S}|$ is low, BU shows comparable performance w.r.t. RTDB, showing that, when a simple Bayesian model update is applied, the benefit derived from dataset $|\mathbf{S}|$ is marginal once $\mathbf{T}$ is at disposal, but improvements are visible for higher cardinalities of $|\mathbf{S}|$, with an ascending trend comparable to that of CORAL. Indeed, samples coming from Jnet help the NSFnet dataset to be complemented of useful information on the SNR of short paths. Conversely, FA appears not to bring significant benefits in this scenario: it is always below CORAL and it slightly outperforms RTDB only when $|\mathbf{S}| \geq 500$.

On the top center, Fig 4 shows the results obtained with a pure AL procedure. Here the x-axis denotes the size of $\mathbf{T}$. The reference in this case is RTDB, i.e. $|\mathbf{T}_0| = 50$, and we show the AL results every 100 newly added points. After 100 AL iterations, i.e. for $|\mathbf{T}| = 100$, AL obtains a median R2 score of 0.859, which is comparable to the best performance obtained by DA techniques (i.e., by CORAL with 1000 source domain samples). With $|\mathbf{T}| = 200$, AL achieves a median R2 score of 0.866, higher than any DA procedure. This shows that AL is generally much more data efficient than DA procedures. However, note that generating the initial training set could be more costly than in a DA method, so a pure AL procedure might not be always feasible from an economical perspective.

On the top right, Fig. 4 reports the results obtained combining AL with either SDB or CORAL, assuming that set $|\mathbf{S}|$ consists of 500 source domain data points. For $|\mathbf{T}| = 50$, the performance of RTDB is reported as a reference. SDB+AL has a slightly higher median than CORAL+AL, but with marginal differences from a practical perspective. The AL method on top of both DA techniques results in a marginally increased R2 score: from 0.844 ($|\mathbf{T}| = 0$) to 0.850 ($|\mathbf{T}| = 100$) for CORAL and from 0.849 ($|\mathbf{T}| = 0$) to 0.852 ($|\mathbf{T}| = 100$) for SDB. Note that a pure AL approach outperforms any of the cases tested for DA+AL when the number of iterations is 50 or higher. The initial training set plays a crucial role when selecting the next few samples in the AL procedure. In CORAL+AL or SDB+AL the initial model of the AL procedure is built on many (500) source domain samples and achieves better scores than RTDB, however the increase in R2 obtained there after adding 25 new samples is much smaller (median increase 0.001) than the improvement obtained in the pure AL case (median increase 0.02). This is due to the fact that the initial model of the DA+AL methods does not build a very reliable uncertainty quantification, as opposed to RTDB.

At the bottom left, Fig. 4 reports the RMSE calculated comparing SNR predictions to the ground truth values. CORAL and BU exhibit a median RMSE in the range $[0.889 - 0.946]$ dB and show a noticeable decrease w.r.t. SDB which ranges from 0.913 to 0.988 dB, thus both techniques reduce the gap from the reference value reached by LTDB (0.837 dB). RTDB shows a median RMSE of 0.952 dB, close to the top of the range spanned by CORAL and BU. The most consistent RMSE reduction is obtained by AL, which exhibits an RMSE as low as 0.872 using 100 probes (bottom center). The combination of AL with SDB or CORAL yields an RMSE in the range $[0.887 - 0.910]$ dB (bottom right), thus reducing the values obtainable by the two techniques without AL.

In order to give an intuition on the practical relevance of these results, we report in Table 3 the percentage (averaged over 10 repetitions) of test data that recorded an error below 0.5 dB, 1 dB and 2 dB in the best scenario for each method. Note that for AL with 800 iterations more that 50% of the errors are below 0.5 dB, and more than 97% are below 2 dB.

Method	RMSE	$\epsilon < 0.5$	$0.5 \leq \epsilon < 1$	$1 \leq \epsilon < 2$	$\epsilon \geq 2$
RTDB	0.9522	0.4598	0.2910	0.2087	0.0405
BU (1000)	0.8885	0.4164	0.3343	0.2239	0.0254
CORAL (1000)	0.8899	0.4632	0.3059	0.1939	0.0370
FA (1000)	0.9501	0.4618	0.2932	0.1989	0.0461
SDB (500)	0.8905	0.4018	0.3352	0.2449	0.0181
LTDB	0.8367	0.5152	0.3184	0.1448	0.0216
AL (800 its)	0.8260	0.5090	0.3143	0.1492	0.0275

Table 3. Target NSFnet. Percentage of absolute errors, ϵ, smaller than thresholds in dB.

We now focus on a different scenario, where the NSFnet is considered as the source domain and the Jnet as the target domain. As reported at the the top left of Fig. 5, SDB performs much worse than RTDB, showing that knowledge about short lightpaths carried by the samples from NSFnet is not sufficient to achieve good prediction capabilities. Indeed, even when $|\mathbf{S}| = 1000$, the R2 values obtained by SDB is below those obtained by RTDB with only 50 samples gathered from Jnet. CORAL outperforms SDB when the cardinality of $\mathbf{S}$ is low and RTDB when the number of samples from the source domain

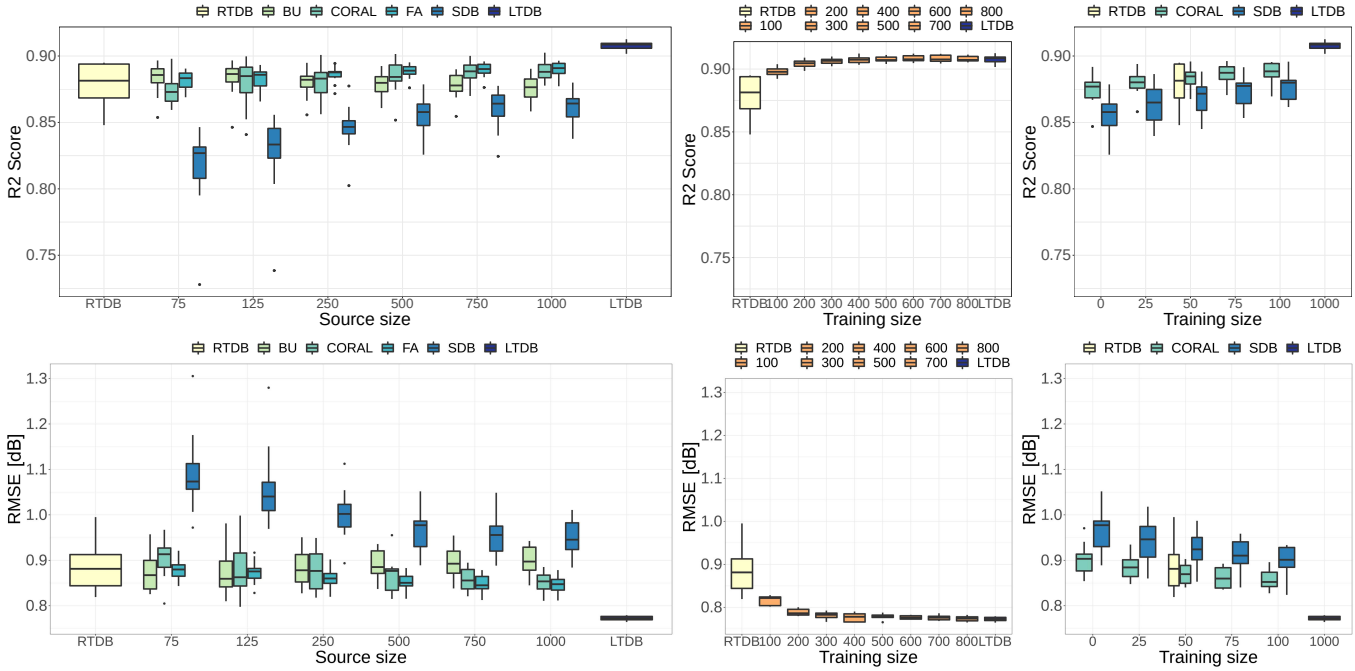

Fig. 5. R2 (top) and RMSE (bottom) obtained by DA (left), AL (center) and DA+AL (right) approaches with target domain Jnet and source domain NSFnet.

increases, reaching 0.888 for $|\mathbf{S}| = 1000$. Using BU appears not to bring additional benefits in this scenario. Indeed, BU does not outperform RTDB, hinting that the samples gathered from the NSFnet (which exhibits a much larger range of lightpath lengths w.r.t. Jnet) do not bring significant improvements on the predictive capabilities of a GP model, when a few samples gathered from Jnet are available. Conversely, FA outperforms RTDB when $|\mathbf{S}|$ is large, showing the advantage of leveraging a more advanced domain adaptation mechanism w.r.t. BU. FA outperforms the other techniques when $|\mathbf{S}| \geq 250$.

At the top center,Fig. 5 shows the performance of an AL procedure trained with starting training set RTDB. After 50 AL iterations, i.e. with a training set of size $|\mathbf{T}| = 100$, we obtain a median R2 value of 0.895 which is higher than any DA technique. Especially in this scenario, where the source domain NSFnet is not too informative w.r.t. the target domain Jnet, an AL procedure could achieve better results than DA methods with few samples.

The top right graph in Fig. 5 reports the performance of combining AL with either SDB or CORAL. As already observed in the DA methods comparison, SDB+AL performs much worse than CORAL+AL. Furthermore, this difference is preserved even after adding samples from the target domain with AL. SDB does not reach the performance of RTDB even when 100 samples from $\mathbf{T}$ are added by means of the AL procedure, showing that the initial training set used for the AL procedure is important. On the other hand, CORAL with $|\mathbf{S}| = 500$ and $|\mathbf{T}| \geq 50$ outperforms RTDB and with $|\mathbf{T}| = 100$ has performance comparable with any DA method with $|\mathbf{S}| = 1000$. However, after 100 iterations, none of the combined techniques outperforms the pure AL approach.

The bottom part of Fig. 5 reports the RMSE achieved by DA and AL techniques. In this scenario results are on average lower than in the previous one and confirm that the pure AL approach yields to the lowest RMSE (0.821 dB when $|\mathbf{T}| = 100$).

Table 4 shows the distribution of absolute errors for Jnet. All methods achieve here better results compared to the NSFnet, in particular in the case of AL, after 750 iterations, 53% of the errors are below 0.5 dB and 97% are below 2 dB.

Method	RMSE	$\epsilon < 0.5$	$0.5 \leq \epsilon < 1$	$1 \leq \epsilon < 2$	$\epsilon \geq 2$
RTDB	0.8813	0.4646	0.3144	0.1908	0.0302
BU (125)	0.8594	0.4939	0.2929	0.1773	0.0359
CORAL (1000)	0.8536	0.4649	0.3213	0.1924	0.0214
FA (1000)	0.8474	0.4808	0.3157	0.1736	0.0299
SDB (1000)	0.9454	0.4499	0.2990	0.2112	0.0399
LTDB	0.7724	0.5412	0.3213	0.1228	0.0147
AL (800)	0.7680	0.5268	0.3138	0.1335	0.0259

Table 4. Target Jnet. Percentage of absolute errors, ϵ, smaller than thresholds in dB.

The methods have all reasonable computational costs, however the GP algorithm scales cubically in the training size, so it is not recommended for large datasets. In the examples above, training a GP model with 800 data points takes about 190 seconds on a regular laptop. This cost is mainly driven by the hyper-parameter optimization and it strongly depends on the implementation of such optimization. The AL part requires another optimization (minimization of $\mathcal{I}_\ell$) which costs on average 76 seconds. Here we use the R package DiceKriging [31] for fitting the GP and KrigInv [32] for the AL part.

Based on the results discussed above, it emerges that comparable R2 improvements over RTDB can be obtained either with a few additional samples acquired from the target domain by means of an AL approach, or with some hundreds of samples gathered from another network topology. The decision on which approach to adopt depends both on the availability and

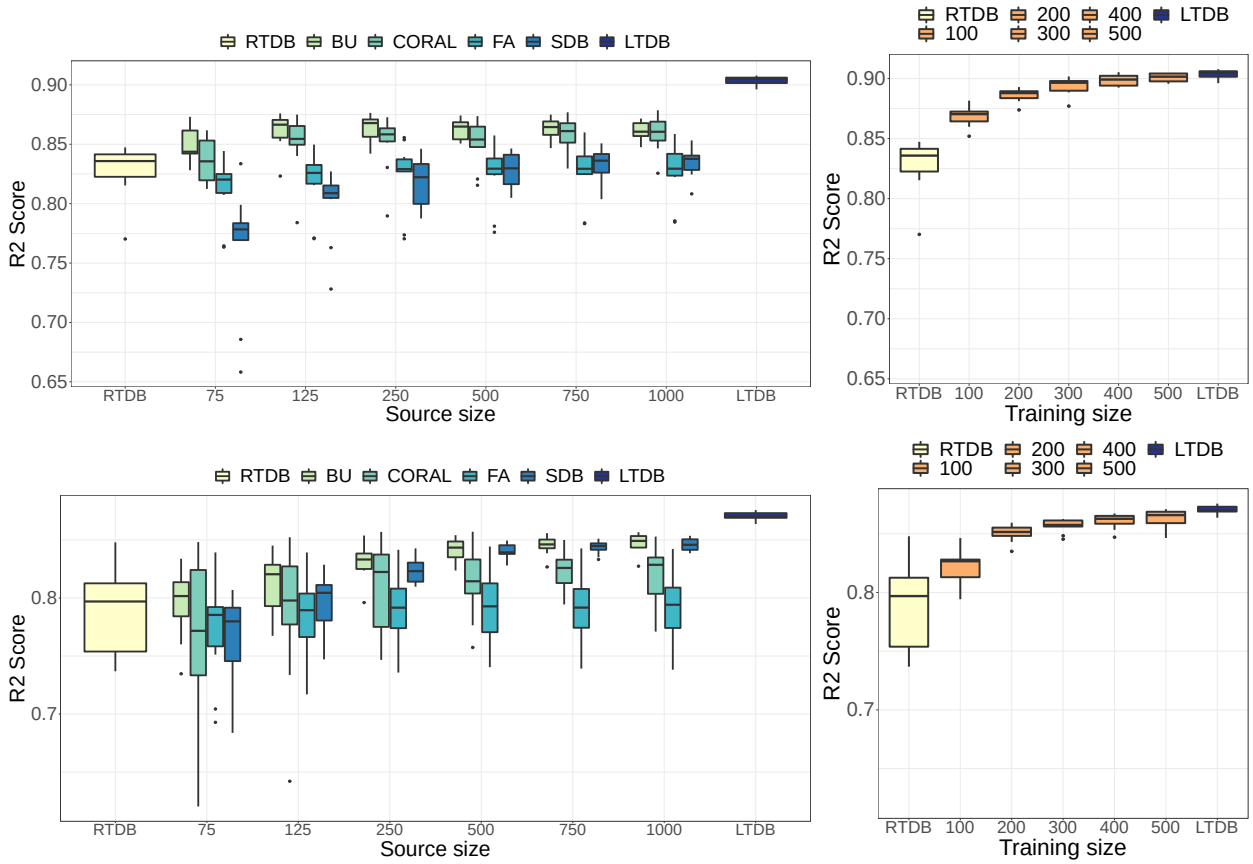

Fig. 6. R2 obtained by DA (left) and AL (right) approaches with target domain Jnet (top) and target domain NSFnet (bottom).

acquisition costs of external samples or probe. Therefore, in the following subsection we explore this aspect by conducting a performance-cost trade-off analysis.

C. Impact of the neighbor channel features

We now partially replicate the experiments of the previous subsection with samples that includes 6 features in addition to the already available features, i.e. guardband, modulation format and traffic volume of the nearest left and right neighbor ligthpaths. The setup is identical to the one used in the previous section, except for AL where we add only 450 new samples here.

Figure 6 shows on the top the R2 values obtained by DA and AL with target domain Jnet and source domain NSFnet. On the bottom we report the R2 values with target domain NSFnet and source domain Jnet. The performances in this case are overall slightly worse: for example, when considering Jnet as target target domain, AL achieves a R2 higher than 0.9 only after 450 iterations, training size of 500, while in the 5 feature cases the value was achieved already 150 iterations, i.e. training size 200 achieves R2 above 0.9. This is due to the increase in the number of dimensions of the feature space. The conclusion of the comparison developed in the previous subsection, however, still hold. In particular, note how AL achieves better results than all DA techniques tested after 150 iterations (i.e. training size 200) in both cases. Moreover, CORAL generally outperforms both FA and SDB as in the 5 features case. Interestingly, as opposed to the 5 features case, BU shows an improvement in performance with target domain NSFnet as we increase the source domain size.

D. Evaluation of Costs/Benefits Trade-off

We now discuss the trade-off between prediction performance achieved by the considered DA/AL approaches and the costs incurred by each of them. We assume that acquiring one sample from an external domain has unitary cost and that the ratio between the cost required to deploy a probe in the considered domain and the cost to acquire a sample from an external domain is $a = 1, 10, 100$ (the computation of such amounts may also consider the costs incurred for the installation of the necessary telemetry equipment). Fig.7 plots the median R2 versus the total cost required to obtain the samples in sets $\mathbf{S}$ and $\mathbf{T}$, for all techniques reported in Tab.1 and different cardinalities of the two sets (where $|\mathbf{S}|$ ranges from 0 to 1000 and $|\mathbf{T}|$ ranges from 0 to 150), under the same scenarios explored in Figs.4 and 5. In both cases, AL achieves the highest R2 median values. Therefore, if $a = 1$ (i.e., acquiring a sample from the source domain costs as much as deploying a probe in the target domain) AL turns out to be the preferable approach. Conversely, if $a = 10$, the cost of acquiring hundreds of samples from the source domain becomes comparable to that of installing some tens of probes in the target domain. Therefore, AL and DA techniques span the same cost region. In this case, though AL still provides the highest R2, a performance close to that of AL could be obtained by DA approaches such as CORAL (when the target domain is NSFnet, see Fig.7, left) and FA (when the target domain is Jnet, see Fig.7, right), with approximately the same cost. The added benefits would be to avoid reserving network resources to probe lightpaths. Finally, if $a = 100$, AL techniques become far more expensive than DA techniques, therefore their slight increase

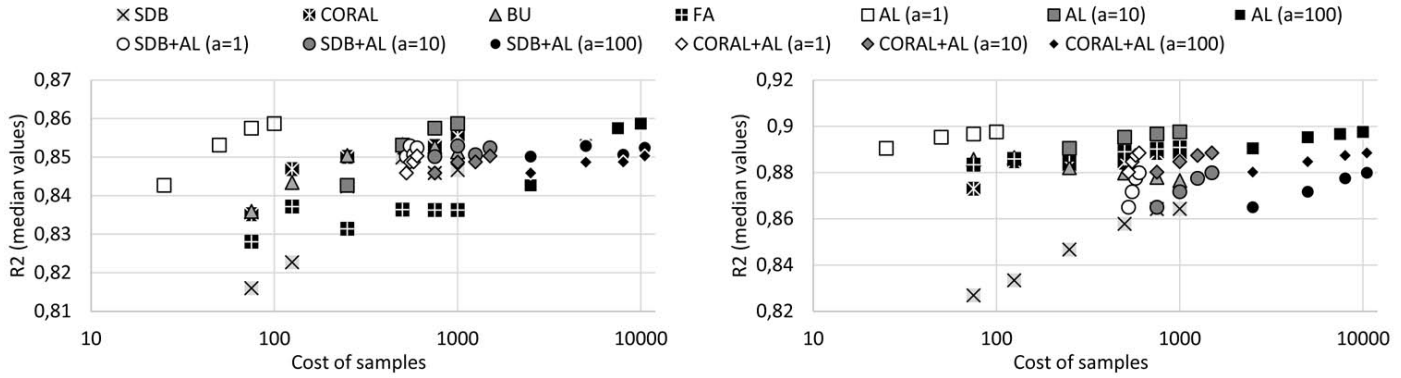

Fig. 7. R2 vs. cost trade-off obtained with target domain NSFnet (left) or Jnet (right), for various DA/AL techniques and sample cost ratios.

in R2 w.r.t. FA and CORAL may not be worth the sheer cost increase.

5. CONCLUSION

We studied the performance of AL (Active Learning) and DA (Domain Adaptation) techniques for SNR estimation in flexible grid networks. We showed that both AL and DA provide estimation capabilities similar to those of classical supervised approaches, without requiring a large data set that may be difficult and costly to obtain, particularly for recently deployed optical networks. Depending on the relative costs of acquiring available data from an existing source network or obtaining instances via lightpath probes in the target network, AL or DA may be preferred. We also discuss the opportunity to jointly exploit AL and DA approaches to enhance the ONSR estimation capability.

ACKNOWLEDGMENTS

We thank prof. Luca Barletta for his precious assistance with the E-tool data generation sofware.

FUNDING

Dario Azzimonti acknowledges funding from the SNSF, grant number 167199. Massimo Tornatore acknowledges funding from the National Science Foundation (NSF) grant number 1716945.

REFERENCES

1. F. Musumeci, C. Rottondi, A. Nag, I. Macaluso, D. Zibar, M. Ruffini, and M. Tornatore, "An overview on application of machine learning techniques in optical networks," IEEE Commun. Surv. & Tutorials **21**, 1383–1408 (2018).
2. G. Bosco, V. Curri, A. Carena, P. Poggiolini, and F. Forghieri, "On the performance of nyquist-WDM terabit superchannels based on PM-BPSK, PM-QPSK, PM-8QAM or PM-16QAM subcarriers," J. Light. Technol. **29**, 53–61 (2011).
3. C. Rottondi, L. Barletta, A. Giusti, and M. Tornatore, "Machine-learning method for quality of transmission prediction of unestablished lightpaths," J. Opt. Commun. Netw. **10**, A286–A297 (2018).
4. K. Christodoulopoulos, P. Kokkinos, A. Di Giglio, A. Pagano, N. Argyris, C. Spatharakis, S. Dris, H. Avramopoulos, J. Antona, C. Delezoide *et al.*, "Orchestra-optical performance monitoring enabling flexible networking," in *2015 17th International Conference on Transparent Optical Networks (ICTON),* (IEEE, 2015), pp. 1–4.
5. D. Azzimonti, C. Rottondi, and M. Tornatore, "Reducing probes for quality of transmission estimation in optical networks with active learning," J. Opt. Commun. Netw. **12**, A38–A48 (2020).
6. A. Vahdat, M. Belbahri, and V. P. Nia, "Active learning for high-dimensional binary features," arXiv preprint arXiv:1902.01923 (2019).
7. K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," J. Big data **3**, 9 (2016).
8. J. Yu, W. Mo, Y.-K. Huang, E. Ip, and D. C. Kilper, "Model transfer of QoT prediction in optical networks based on artificial neural networks," IEEE/OSA J. Opt. Commun. Netw. **11**, C48–C57 (2019).
9. L. Xia, J. Zhang, S. Hu, M. Zhu, Y. Song, and K. Qiu, "Transfer learning assisted deep neural network for OSNR estimation," Opt. Express **27**, 19398–19406 (2019).
10. Y. Cheng, W. Zhang, S. Fu, M. Tang, and D. Liu, "Transfer learning simplified multi-task deep neural network for PDM-64QAM optical performance monitoring," Opt. Express **28**, 7607–7617 (2020).
11. C.-Y. Liu, X. Chen, R. Proietti, and S. B. Yoo, "Evol-tl: evolutionary transfer learning for QoT estimation in multi-domain networks," in *2020 Optical Fiber Communications Conference and Exhibition (OFC),* (IEEE, 2020), pp. 1–3.
12. Z. Xu, C. Sun, T. Ji, H. Ji, and W. Shieh, "Transfer learning aided neural networks for nonlinear equalization in short-reach direct detection systems," in *2020 Optical Fiber Communications Conference and Exhibition (OFC),* pp. 1–3.
13. S. Sun, H. Shi, and Y. Wu, "A survey of multi-source domain adaptation," Inf. Fusion **24**, 84 – 92 (2015).
14. W. M. Kouw and M. Loog, "An introduction to domain adaptation and transfer learning," (2018).
15. H. Daumé III, "Frustratingly easy domain adaptation," arXiv preprint arXiv:0907.1815 (2009).
16. D. Azzimonti, C. Rottondi, A. Giusti, M. Tornatore, and A. Bianco, "Active vs transfer learning approaches for QoT estimation with small training datasets," in *Optical Fiber Communication Conference,* (Optical Society of America, 2020), pp. M4E–1.
17. I. Sartzetakis, K. K. Christodoulopoulos, and E. M. Varvarigos, "Accurate quality of transmission estimation with machine learning," J. Opt. Commun. Netw. **11**, 140–150 (2019).
18. E. Seve, J. Pesic, C. Delezoide, S. Bigo, and Y. Pointurier, "Learning process for reducing uncertainties on network parameters and design margins," IEEE/OSA J. Opt. Commun. Netw. **10**, A298–A306 (2018).
19. P. Samadi, D. Amar, C. Lepers, M. Lourdiane, and K. Bergman, "Quality of transmission prediction with machine learning for dynamic operation of optical WDM networks," in *2017 European Conference on Optical Communication (ECOC),* (2017), pp. 1–3.
20. T. Panayiotou, S. P. Chatzis, and G. Ellinas, "Performance analysis of a data-driven quality-of-transmission decision approach on a dynamic multicast-capable metro optical network," J. Opt. Commun. Netw. **9**, 98–108 (2017).
21. T. Jimenez, J. C. Aguado, I. de Miguel, R. J. Duran, M. Angelou,

N. Merayo, P. Fernandez, R. M. Lorenzo, I. Tomkos, and E. J. Abril, "A cognitive quality of transmission estimator for core optical networks," J. Light. Technol. **31**, 942–951 (2013).
22. C. K. Williams and C. E. Rasmussen, "Gaussian processes for machine learning," MIT Press. **2**, 4 (2006).
23. J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, "Design and analysis of computer experiments," Stat. Sci. **4**, 409–423 (1989).
24. C. Chevalier, V. Picheny, and D. Ginsbourger, "KrigInv: An efficient and user-friendly implementation of batch-sequential inversion strategies based on kriging," Comput. Stat. Data Analysis **71**, 1021–1034 (2014).
25. C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*, vol. 2 (MIT press Cambridge, MA, 2006).
26. B. Sun, J. Feng, and K. Saenko, "Correlation alignment for unsupervised domain adaptation," in *Domain Adaptation in Computer Vision Applications,* (Springer, 2017), pp. 153–171.
27. P. Poggiolini, "The GN model of non-linear propagation in uncompensated coherent optical systems," J. Light. Technol. **30**, 3857–3879 (2012).
28. Y. Pointurier, M. Coates, and M. Rabbat, "Cross-layer monitoring in transparent optical networks," J. Opt. Commun. Netw. **3**, 189–198 (2011).
29. A. Ferrari, M. Filer, K. Balasubramanian, Y. Yin, E. Le 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. **12**, C31–C40 (2020).
30. R. Dar, M. Feder, A. Mecozzi, and M. Shtaif, "Accumulation of nonlinear interference noise in fiber-optic systems," Opt. express **22**, 14199–14211 (2014).
31. O. Roustant, D. Ginsbourger, and Y. Deville, "DiceKriging, DiceOptim: Two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization," J. Stat. Softw. **51**, 1–55 (2012).
32. C. Chevalier, V. Picheny, and D. Ginsbourger, "The KrigInv package: An efficient and user-friendly R implementation of kriging-based inversion algorithms," Comput. Stat. & Data Analysis **71**, 1021–1034 (2014).