Performance studies of evolutionary transfer learning for end-to-end QoT estimation in multi-domain optical networks [Invited]

*To be published in Journal of Optical Communications and Networking:*

**Title:** Performance Studies of Evolutional Transfer Learning for End-to-End QoT Estimation in Multi-domain Optical Networks

**Authors:** Xiaoliang Chen,Che-Yu Liu,Roberto Proietti,S. J. Ben Yoo

**Accepted:** 23 November 20

**Posted** 24 November 20

**DOI:** https://doi.org/10.1364/JOCN.409817

# Performance Studies of Evolutionary Transfer Learning for End-to-End QoT Estimation in Multi-domain Optical Networks [Invited]

CHE-YU LIU[1], XIAOLIANG CHEN[2,*], ROBERTO PROIETTI[2], AND S. J. BEN YOO[2,*]

[1] Department of Computer Science, University of California, Davis, Davis, CA 95616, USA
[2] Department of Electrical and Computer Engineering, University of California, Davis, Davis, CA 95616, USA
[*] Corresponding author: xlichen@ucdavis.edu, sbyoo@ucdavis.edu

Compiled November 18, 2020

**This paper proposes an evolutionary transfer learning approach (Evol-TL) for scalable QoT estimation in multi-domain elastic optical networks (MD-EONs). Evol-TL exploits a broker-based MD-EON architecture that enables cooperative learning between the broker plane (end-to-end) and domain-level (local) machine learning functions while securing the autonomy of each domain. We designed a genetic algorithm to optimize the neural network architectures and the sets of weights to be transferred between the source and destination tasks. We evaluated the performance of Evol-TL with three case studies considering the QoT estimation task for lightpaths with *i*) different path lengths (in terms of the numbers of fiber links traversed), *ii*) different modulation formats, and *iii*) different device conditions (emulated by the introducing of different levels of wavelength-specific attenuation to the amplifiers). The results show that the proposed approach can reduce the average amount of required training data by up to 13× while achieving an estimation accuracy above 95%.** 

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

## 1. INTRODUCTION

Elastic optical networking (EON) has been recognized as one of the most promising solutions for next-generation transport networks [1], thanks to its unprecedented flexibility in the control and management of the optical fiber spectrum. While there has been a large number of comprehensive studies on single-domain EONs in the recent years [2–6], optimizing the inter-networking of multi-domain elastic optical networks (MD-EONs) [7] remains a challenging task due to the limited amount of intra-domain information that can be exchanged between domains (the domains autonomy constraints). In particular, accurate quality-of-transmission (QoT) modeling is essential for low-margin (thus, resource-efficient) connection provisioning [8] in MD-EONs. However, the domain autonomy constraints make the conventional analytical models relying on full network configuration states [9–11] hard to be applied. Moreover, these models may suffer from poor adaptability as they typically fail to capture the evolving network conditions (e.g., due to hardware components degradation).

Recently, the research community has made a series of breakthroughs in machine learning (ML) and demonstrated its successful applications in optical communications and networking [12–15]. Specifically, ML makes it possible to develop data analytics functions that can autonomically learn complex network rules from data and hereby realize knowledge-based cognitive networking [16]. Previous works have investigated various ML-aided designs for optical networks, including QoT estimation [17–22], failure/attack detection and identification [23–29], and resource allocation [30–34]. While these existing works have shown promising benefits, only a few of them are tailored for MD-EONs. In particular, existing ML-aided QoT estimation models require end-to-end data and network configuration information such as signal power, modulation format, fiber length, and link channel occupancy [17]. Obtaining this information in a multi-domain scenario is not trivial for the reasons discussed above. In this context, we recently proposed a hierarchical learning approach, where the QoT of inter-domain lightpaths can be inferred through cooperative learning between a broker-plane (end-to-end) and domain-level (local) ML functions [35]. Experimental results have shown that the proposed approach can achieve an estimation accuracy close to that of an omniscient model relying on full multi-domain state data. However, the hierarchical learning approach still requires the domain managers (DMs) to collect a significant amount of optical performance monitoring (OPM) data for every inter-domain lightpath, which can add non-negligible operational costs to MD-EONs and cause scalability issues.

Transfer learning (TL) can ease the training of ML models by

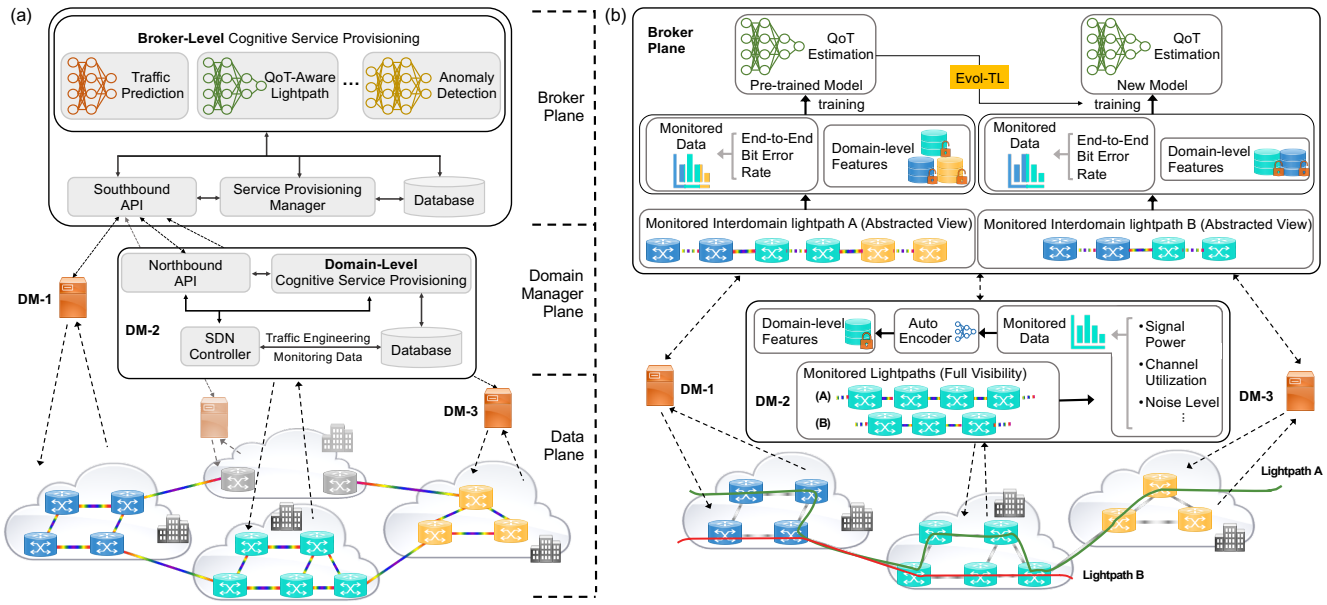



**Fig. 1.** (a) Broker-based MD-EON architecture with hierarchical learning, and (b) principle of inter-domain QoT estimation.

allowing the knowledge sharing between relevant tasks [36] and therefore has been successfully applied to real-word applications (e.g., natural language processing [37, 38] and computer vision [39, 40]). In particular, the merit of TL for optical networks has been confirmed by several recent studies, revealing that TL can effectively reduce the amount of training data or time required for QoT estimation [41–44] or resource allocation tasks [45, 46]. Nevertheless, these works only consider single-domain scenarios. In [47], we proposed an evolutionary TL approach (Evol-TL) for scalable QoT estimation in MD-EONs with hierarchical learning. Evol-TL makes use of a genetic algorithm (GA) to determine the optimal structures of ML functions to use (i.e., the neural network architectures) and the most effective sets of weights to transfer between tasks.

In this paper, we extend our previous work in [47] by presenting a more elaborate description of Evol-TL and by performing case studies incorporating more comprehensive network scenarios. Specifically, in Section 2, we describe the principle of inter-domain QoT estimation in a broker-based MD-EON architecture enabling cognitive inter-domain networking. The problem of TL for QoT estimation is mathematically formulated in Section 3 together with the principle of the proposed GA-based knowledge transferring scheme. Section 4 reports the performance of Evol-TL in three scenarios, where we transfer knowledge between QoT estimators for inter-domain lightpaths with *i*) different path lengths (in terms of the numbers of fiber links traversed), *ii*) different modulation formats, and *iii*) different device conditions. In Section 5, we perform a brief review of the recent works on ML applications in optical networks. Finally, Section 6 summarizes the main findings in the paper.

## 2. QOT ESTIMATION IN MD-EONS

### A. Network Architecture

Fig. 1(a) shows the block diagram of the proposed broker-based MD-EON architecture supporting cognitive inter-domain networking. Each DM manages its EON data plane with a software-defined networking (SDN) paradigm to provide flex-grid optical connection services among metro networks, data centers, and scientific facilities. OPM functions are deployed at various data plane locations to enable DMs to sense the network state in real time. For instance, an optical spectrum analyzer (OSA) can be used to monitor the signal power and channel utilization on a specific fiber link. Taking advantage of the OPM capability and the centralized control mechanism by SDN, DMs can develop ML functions to realize cognitive intra-domain networking (e.g., QoT-aware lightpath provisioning, fault detection and recovery) following an observe-analyze-act cycle [16]. Specifically, at each operation point (e.g., upon receiving a service request), a DM first collects the instant OPM data using a telemetry service [48] (*observe*). Then, the DM performs data analytics by executing the related ML functions (*analyze*). Finally, an intelligent operation decision can be made based on the perception of the network states and rules (*act*).

To facilitate effective inter-domain networking, we introduce a broker plane as a new and higher network control and management hierarchy. The broker plane offers cognitive inter-domain services (also following an observe-analyze-act cycle) to the DMs according to specific mutual service level agreements (SLAs). For instance, an SLA can require a DM to abstract its domain as a virtual topology consisting of domain ingress and egress nodes connected by virtual links (i.e., intra-domain path segments) on which certain amounts of spectrum resources can be reserved. The broker plane will notify which of the virtual links can be used by the end-to-end service schemes afterward. Note that, the abstracted domain information is often inadequate for the broker-plane ML functions to correctly learn the rules of MD-EONs. In this case, a hierarchical learning mechanism can be applied, where the broker-plane and domain-level ML functions learn cooperatively to accomplish inter-domain service provisioning [35]. The domain-level ML functions learn local features or operation policies from full network state data while the broker-plane ML functions aggregate the local learning results and produce end-to-end learning targets. Such a scheme allows the broker plane to implicitly exploit the intra-domain

information while at the same time securing the privacy of each domain.

### B. Operation Principle

The principle of QoT estimation in MD-EONs with cooperative learning is illustrated by the example in Fig. 1(b), where two inter-domain lightpaths, i.e., lightpaths $A$ and $B$, are established successively. Each DM collects the network state and OPM data (e.g., link load, signal power, noise level) related to the corresponding intra-domain path segments and employs an ML model to extract domain-level features from the data. For instance, autoencoder, which is known as to be able to encode data while preserving the information contained in the data, can be implemented to perform domain-level ML functions. As such, reporting only the domain-level features to the broker plane makes the original OPM data hard to be derived and protects the confidentiality of domains. The broker plane combines the domain-level features to train an end-to-end QoT estimator for each lightpath. Here, although lightpaths $A$ and $B$ traverse different numbers of links and probably use devices of different vendors/conditions, the two QoT estimation tasks still share high similarities as the physical layer principles of the two systems are essentially the same. Therefore, the broker plane applies TL and reuses the model parameters learned with the data from lightpath $A$ to significantly reduce the amount of OPM data required to train the QoT estimator for lightpath $B$.

## 3. EVOL-TL DESIGN

In this section, we formulate the problem of TL for QoT estimation in MD-EONs and detail the algorithm design for Evol-TL.

### A. Problem Formulation

Let $\mathcal{M} = \{\mathcal{X}, P(\mathcal{X}), \mathcal{Y}, \mathcal{F}\}$ denote an inter-domain QoT estimation task characterized by a feature space $\mathcal{X} = \{\mathcal{X}^1, ..., \mathcal{X}^N\}$ (a conjunction of $N$ domain-wise feature spaces) with a probability distribution $P(\mathcal{X})$, a label (i.e., QoT value) space $\mathcal{Y}$, and a mapping function $\mathcal{F} : \mathcal{X} \rightarrow \mathcal{Y}$. According to the discussions in Section 2, we can decompose $\mathcal{M}$ into $N$ domain-level subtasks and a broker-plane subtask. Let $\mathcal{M}^n = \{\mathcal{X}^n, P(\mathcal{X}^n), \mathcal{F}^n\}$ denote a subtask for domain $n$, then, a broker-plane subtask can be represented by $\hat{\mathcal{M}} = \{\hat{\mathcal{X}}, P(\hat{\mathcal{X}}), \mathcal{Y}, \hat{\mathcal{F}}\}$, where $\hat{\mathcal{X}} = \{\mathcal{F}^1(\mathcal{X}^1), ..., \mathcal{F}^N(\mathcal{X}^N)\}$ is the space of domain-level features. Normally, $\mathcal{F} = \{\mathcal{F}^1, ..., \mathcal{F}^N, \hat{\mathcal{F}}\}$ can be learned from a data set $\{(x_i, y_i) | x_i \in \mathcal{X}, y_i \in \mathcal{Y}\}$ using the hierarchical learning approach [35], by minimizing the average error between $\mathcal{F}(x_i)$ and $y_i$. Note that, $\mathcal{Y}^n$ is not required for each $\mathcal{M}^n$ because the training of $\mathcal{M}^n$ can be done either using unsupervised learning (for instance, when an autoencoder is used) or by using the supervision signals (i.e., gradients) distributed by the broker plane through a collaborative learning approach presented in [49]. Now, let us consider a source QoT estimation task $\mathcal{M}_S = \{\mathcal{X}_S, P(\mathcal{X}_S), \mathcal{Y}_S, \mathcal{F}_S\}$ that has been trained, and a target task $\mathcal{M}_T = \{\mathcal{X}_T, P(\mathcal{X}_T), \mathcal{Y}_T, \mathcal{F}_T\}$, whose feature and label spaces may both differ, i.e., $\mathcal{X}_S \neq \mathcal{X}_T$, $\mathcal{Y}_S \neq \mathcal{Y}_T$. The goal of Evol-TL is to minimize the number of data instances required to learn an $\mathcal{F}_T$ with target accuracy performance. To achieve the goal, Evol-TL needs to explore and transfer the most effective knowledge from $\mathcal{F}_S$. Note that, knowledge transferring can happen within both domains and the broker plane, i.e., from $\mathcal{F}_S^n$ to $\mathcal{F}_T^n$, and from $\hat{\mathcal{F}}_S$ to $\hat{\mathcal{F}}_T$.

**Algorithm 1.** Optimization procedures of Evol-TL.

**Input:** $\eta$, $\lambda$, $\gamma$, $\hat{\mathcal{M}}_S$, $\hat{\mathcal{M}}_T$
**Output:** $C^*$

1: $P \leftarrow$ GenerateRandomPopulation($J$)
2: **for** $r = 1$ to $R$ **do**
3: ComputeFitnessFunc($P$)
4: $P_0 \leftarrow$ ParentSelection($P$, $\eta$)
5: $P' \leftarrow$ CrossoverOperation($P_0$, $\lambda$)
6: $P' \leftarrow$ MutationOperation($P'$, $\gamma$)
7: $P^+ \leftarrow$ GenerateRandomPopulation($(1 - \eta\%) \cdot J$)
8: $P = P' \cup P^+$
9: **end for**
10: **return** $C^* = \arg\max_{C_j} \mathcal{F}(C_j)$
11:
12: **procedure** ComputeFitnessFunc($P$)
13: **for** $C_j$ in $P$ **do**
14: Train each $\hat{\mathcal{F}}_T$ based on the solution encoded in $C_j$
15: Calculate $\mathcal{F}(C_j)$
16: **end for**
17: Sort $P$ in the descending order of $\mathcal{F}(C_j)$
18: **return** $P$
19: **end procedure**

### B. Optimization Procedures

We consider a neural network-based QoT estimator design. Fig. 2 depicts the principle of knowledge transferring between broker-plane tasks (i.e., $\hat{\mathcal{M}}_S$ and $\hat{\mathcal{M}}_T$) in Evol-TL. Given a pre-trained model $\hat{\mathcal{F}}_S$ parameterized by a neural network of a few hidden layers, we build $\hat{\mathcal{F}}_T$ with the first one or multiple hidden layers copied from $\hat{\mathcal{F}}_S$ and a few more randomly initialized layers inserted thereafter (including an output layer). We lock the copied layers and fine-tune $\hat{\mathcal{F}}_T$ with a small amount of newly collected OPM data. Here, by adding randomly initialized layers, we can potentially mitigate the overfitting issue as the copied layers are typically overfitted to $\hat{\mathcal{M}}_S$. Still, it is necessary to determine the right sets of weights to transfer (i.e., the number of layers that should be copied) because the higher layers of $\hat{\mathcal{F}}_S$ may be severely overfitted and cause negative transferring effects [50], especially when the difference between $\hat{\mathcal{M}}_S$ and $\hat{\mathcal{M}}_T$ is significant. Meanwhile, the number of newly added layers and other hyperparameters, such as the number of neurons in each layer, can largely impact the performance of $\hat{\mathcal{F}}_T$ and also should be optimized.

Unlike existing works that usually decide the hyperparameters of ML models based on random or brute-force searches [51], Evol-TL makes use of a GA approach that performs bio-inspired evolutionary optimizations. Specifically, we encode each feasible solution, i.e., a possible setting of the hyperparameters, as an individual denoted by $C_j = [L_c, L_i, \mathbf{K}, \mathcal{H}, \mathcal{G}]$, where $L_c$ and $L_i$ represent respectively the number of hidden layers to copy from $\hat{\mathcal{F}}_S$ and the number of new layers to insert, $\mathbf{K}$ is a vector containing the number of neurons in each of the new layers, $\mathcal{H}$ indicates the activation function to adopt, and $\mathcal{G}$ tells the optimizer to use in training. A set of individuals then form a population $P = \{C_j, j \in [1, J]\}$. For evaluating the performance of each individual, we also define a fitness function $f(C_j)$ that returns the average prediction accuracy from $\hat{\mathcal{F}}_T$ trained based on the solution encoded in $C_j$. Algorithm 1 summarizes the optimization procedures of Evol-TL. In *Line 1*, we first initialize a population of size $J$ by randomly generating the individuals.

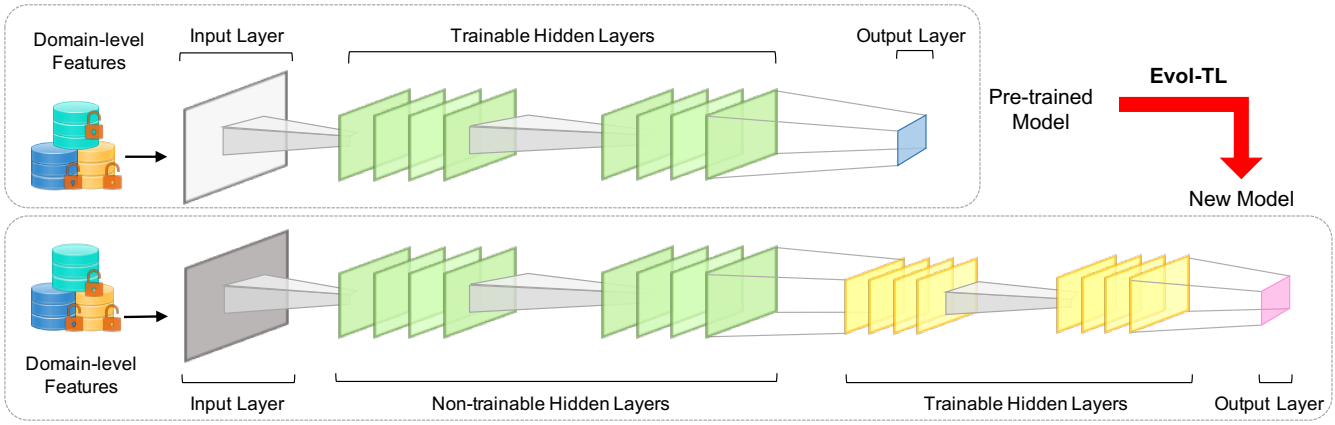



**Fig. 2.** Principle of knowledge transferring between broker-plane tasks in Evol-TL.

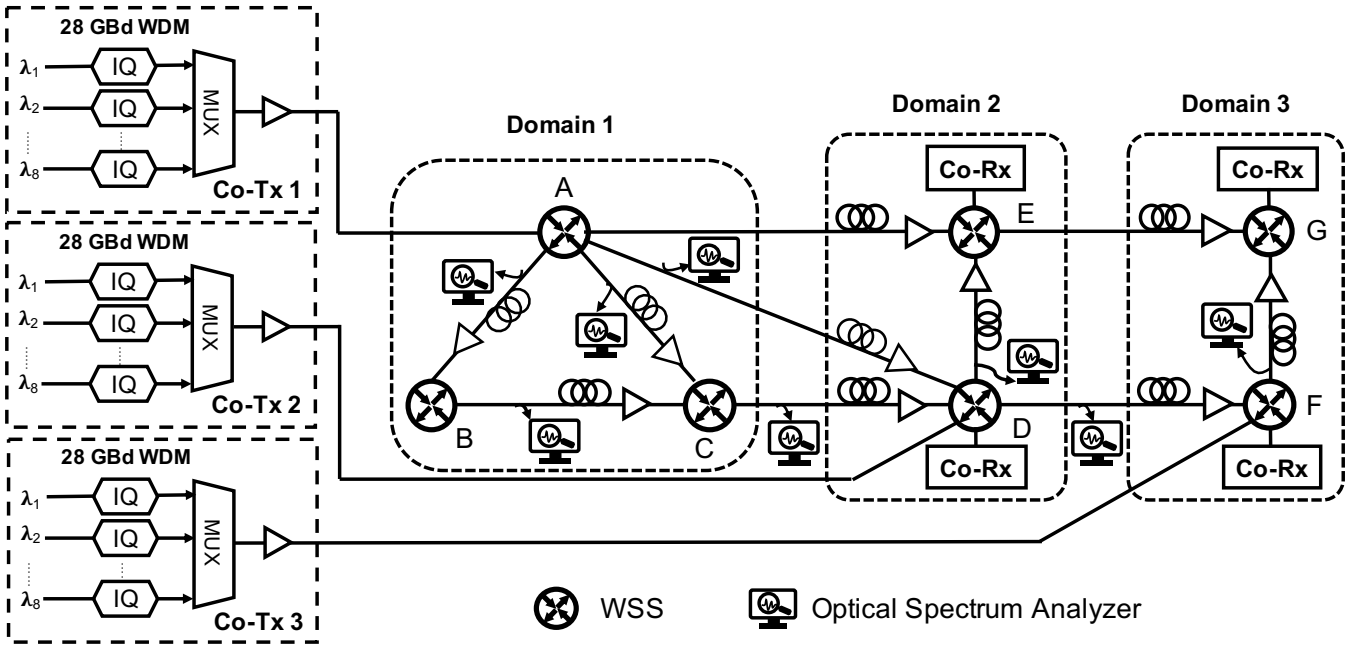



**Fig. 3.** Three-domain EON setup implemented in VPItransmissionMaker™ Optical Systems simulator.



*Lines 2-9* performs evolutionary optimizations for $R$ iterations. In each iteration, *Line 2* (expanded by *Lines 12-19*) calculates the fitness of each individual in $P$ and sort the individuals in the descending order of their fitnesses. In *Line 4*, the best $\eta\%$ of the individuals are selected as parents, which afterward perform crossover (*Line 5*) and mutation (*Line 6*) operations to produce the offsprings. Specifically, in each crossover operation, the selected parents are randomly paired, with each pair randomly exchanging $\lambda$ of their genes (i.e., the elements composing the individuals). The crossover operations thus allow exploiting the advantageous knowledge during optimization. Whereas, by committing random changes on genes with a relatively low probability of $\gamma$, the mutation operations encourage exploration of the optimization space to prevent falling into local optima. Further, to improve the diversity of the new population as well as to maintain the population size, we add another $(100\% - \eta\%) \cdot J$ random individuals (*Line 7-8*). Finally, after optimization of $R$ iterations, *Line 10* returns the best individual $C^*$in $P$ as the solution for $\hat{\mathcal{M}}_T$.

Let $|P|$ and $|C|$ denote respectively the number of individuals in a population and the number of hyperparameters in an individual. Since the complexity of crossover and mutation operations in each optimization iteration are both $\mathcal{O}(|P| * |C|)$, the complexity of Algorithm 1 is $\mathcal{O}(R * (|P| * \mathcal{O}_{fitness} + 2 * |P| * |C|))$, where $\mathcal{O}_{fitness}$ represents the complexity of computing the fitness of each individual. Because computing the fitness function involves a training process, $\mathcal{O}_{fitness}$ is typically much larger than $|P| * |C|$. Therefore, we can further obtain the complexity of Algorithm 1 as $\mathcal{O}(R * |P| * \mathcal{O}_{fitness})$. In contrast, the time complexity of the brute-force approach is $\mathcal{O}(L_c * L_i * \mathbf{K} * \mathcal{H} * \mathcal{G} * \mathcal{O}_{fitness})$. Since the possible combinations of hyperparameters in an individual (i.e., $L_c * L_i * \mathbf{K} * \mathcal{H} * \mathcal{G}$) are typically larger than $R * |P|$, the complexity of brute-force approach is greater than Algorithm 1.

## 4. PERFORMANCE EVALUATION

We evaluated the performance of the proposed Evol-TL design with data sets collected using VPItransmissionMaker™ Optical

**Table 1.** Description of simulation setup in the VPItransmissionMaker™ Optical Systems simulator.

| Path | # of Nodes in Domain 1 | # of Nodes in Domain 2 | # of Nodes in Domain 3 | Type of Fiber | Type of Amplifier | Launch Power | | QoT indicator |
|---|---|---|---|---|---|---|---|---|
| | | | | | | Co-Tx ID | Range of the power (Watt) | |
| A-C-D | 2 | 1 | 0 | Single Mode Fiber | Gain-controlled Amplifier | 1 | $7e^{-4}$ - $5e^{-3}$ | Measured BER Value at Co-Rx |
| | | | | | | 2 | $1e^{-3}$ - $3e^{-3}$ | |
| A-B-C-D | 2 | 2 | 0 | | | 1 | $9e^{-4}$ - $4e^{-3}$ | |
| | | | | | | 2 | $1e^{-3}$ - $3e^{-3}$ | |
| A-B-C-D-E | 3 | 2 | 0 | | | 1 | $1e^{-3}$ - $2e^{-3}$ | |
| | | | | | | 2 | $1e^{-3}$ - $2e^{-3}$ | |
| A-D-F-G | 1 | 1 | 2 | | | 1 | $1.5e^{-3}$ - $7e^{-3}$ | |
| | | | | | | 2 | $1e^{-3}$ - $2e^{-3}$ | |
| | | | | | | 3 | $1e^{-3}$ - $2e^{-3}$ | |
| A-B-C-D-F | 3 | 1 | 1 | | | 1 | $1.5e^{-3}$ – $3.6e^{-3}$ | |
| | | | | | | 2 | $1.3e^{-3}$ – $2.3e^{-3}$ | |
| | | | | | | 3 | $1.3e^{-3}$ – $2.3e^{-3}$ | |

Systems simulator implementing a three-domain EON topology (see Fig. 3). The three I/Q modulator blocks, each fed by eight WDM lasers, are operated at 28 Gbaud (224 Gbaud) adopting 4-QAM or 16-QAM modulations. Each node is connected to other nodes using a standard single-mode fiber module in VPI (FiberNLS_PMD). The length of each fiber link was set as 100 kilometers. The fiber losses between two nodes are compensated using gain-controlled amplifiers with a noise figure of 4 dB. We set up five inter-domain lightpaths, three of which traverse two domains (i.e., domain one and domain two) and are consisted of three (A-C-D), four (A-B-C-D), and five (A-B-C-D-E) nodes, respectively. The rest two lightpaths, i.e., lightpaths A-D-F-G and A-B-C-D-F, traverse all the three domains. The optical spectrum analyzers (OSAs) along the lightpaths monitored the power of the eight background channels, one of which was picked as the testing channel. For each run, we measured the BER of each testing channel using a standard coherent receiver (Co-Rx) module that estimates the BER from the constellation plot obtained from the sampled data. We recorded these BER values (in logarithmic scale) as the labels of the data instances. The BER values range from $10^{-14}$ to $10^{-2}$. For each lightpath, we randomly changed the number of background channels (by configuring the three I/O blocks), the power of the background channels, and the position of the testing channel to emulate various network conditions. Note that, we measured the BER with respect to different values of the launch power and chose a range that shows the impact of nonlinear effects. We processed the OPM data to generate 2400 data instances for each of the inter-domain paths. Each data instance has three input features, including *i)* the average power of the background channels, *ii)* the position of the testing channel, and *iii)* the number of active background channels. Table. 1 summarizes the description of the simulation setup in the VPI simulator.

To cover various network scenarios, we performed three different case studies, i.e., knowledge transferring between QoT estimation tasks for lightpaths *i*) traversing different numbers of nodes, *ii*) adopting different modulation formats, and *iii*) confronting different device conditions [emulated by introducing a total of 2 dB attenuation or different gain tilts for the amplifiers in the domains]. Here, we denote gain tilts applied to the amplifiers in domain one and domain two as $g1$ and $g2$, respectively. Specifically, for $g1$, we varied the gains on the eight channel frequencies in each amplifier from 15.5 to 22.5 dB with a step of 0.5 dB. Whereas for $g2$, we considered gains ranging from 17.75 to 21.25 dB with a step of 1 dB introduced to each amplifier.

We pre-trained a QoT estimator built with a neural network of three fully-connected layers (20 neurons in each layer) for the three-node lightpath with 4-QAM modulation under the ideal amplifier condition (0 dB attenuation and no gain tilt). We denote the source tasks across two and three domains as $\mathcal{M}_S^1$ and $\mathcal{M}_S^2$, respectively. Table. 2 summarizes the target tasks evaluated in different use cases. Specifically, for the lightpaths traversing two domains, we considered two target tasks $\mathcal{M}_T^{i,1}$ and $\mathcal{M}_T^{i,2}$ for each use case $i$: $\mathcal{M}_T^{1,1}$ and $\mathcal{M}_T^{1,2}$ differ from $\mathcal{M}_S^1$ in only the path length, i.e., using the four-node and five-node paths, respectively; $\mathcal{M}_T^{2,1}$ and $\mathcal{M}_T^{2,2}$ use the three-node and four-node paths, respectively, with 16-QAM modulation; $\mathcal{M}_T^{3,1}$ and $\mathcal{M}_T^{3,2}$ use the four-node and five-node paths, respectively, with 4-QAM modulation and a total of 2 dB attenuation introduced to the amplifiers; $\mathcal{M}_T^{3,3}$ uses the four-node path with 4-QAM modulation and gain tilts (i.e., $g1$ and $g2$) introduced to the amplifiers. Similarly, for lightpaths traversing three domains, $\mathcal{M}_S^2$ differs from $\mathcal{M}_T^{1,3}$, $\mathcal{M}_T^{2,3}$, and $\mathcal{M}_T^{3,4}$ in the path length, the modulation format adopted, and the level of wavelength-specific attenuation introduced, respectively.

### A. Results for Use Case 1

We first evaluated the performance of Evol-TL with $\mathcal{M}_S^1$ and $\mathcal{M}_T^{1,1}$. Fig. 4(a) shows the performance of Evol-TL as a function

**Table 2.** Setup of the use cases.

| Case ID | # Domains | Source Task | Target task |
|---|---|---|---|
| 1 | 2 | $\mathcal{M}_S^1$ = [3-node, 4-QAM, 0 dB Atten.] | $\mathcal{M}_T^{1,1}$= [4-node, 4-QAM, 0 dB Atten.] |
| | | | $\mathcal{M}_T^{1,2}$= [5-node, 4-QAM, 0 dB Atten.] |
| | 3 | $\mathcal{M}_S^2$ = [4-node, 4-QAM, 0 dB Atten.] | $\mathcal{M}_T^{1,3}$= [5-node, 4-QAM, 0 dB Atten.] |
| 2 | 2 | $\mathcal{M}_S^1$ = [3-node, 4-QAM, 0 dB Atten.] | $\mathcal{M}_T^{2,1}$= [3-node, 16-QAM, 0 dB Atten.] |
| | | | $\mathcal{M}_T^{2,2}$= [4-node, 16-QAM, 0 dB Atten.] |
| | 3 | $\mathcal{M}_S^2$ = [4-node, 4-QAM, 0 dB Atten.] | $\mathcal{M}_T^{2,3}$= [4-node, 16-QAM, 0 dB Atten.] |
| 3 | 2 | $\mathcal{M}_S^1$ = [3-node, 4-QAM, 0 dB Atten.] | $\mathcal{M}_T^{3,1}$= [4-node, 4-QAM, 2 dB Atten.] |
| | | | $\mathcal{M}_T^{3,2}$= [5-node, 4-QAM, 2 dB Atten.] |
| | | | $\mathcal{M}_T^{3,3}$= [4-node, 4-QAM, g1,g2] |
| | 3 | $\mathcal{M}_S^2$ = [4-node, 4-QAM, 0 dB Atten.] | $\mathcal{M}_T^{3,4}$= [5-node, 4-QAM, 2 dB Atten.] |

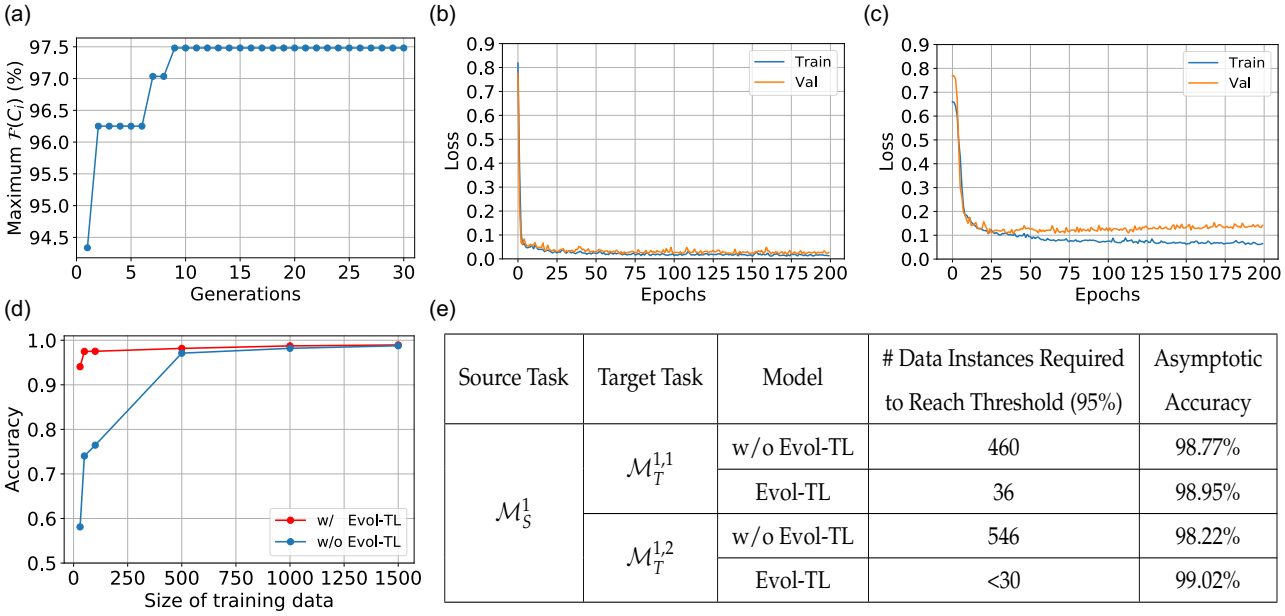
of the optimization iteration, where 150 data instances were used (50 for training, 50 for validation, and another 50 for testing). We can see that the optimization procedure converged at the ninth iteration, achieving the maximum estimation accuracy of 97.5%, which is defined as $1 - |Q_{est} - Q_{meas}|/Q_{meas}$ ($Q_{est}$ and $Q_{meas}$ represent the estimated and measured QoT, respectively). The solution of Evol-TL suggests that the optimal configuration for the target task is transferring one hidden layer from the pre-trained model while newly inserting three randomly initialized layers of 26 neurons. Figs. 4(b) and (c) show the evolutions of the training and validation losses from Evol-TL and a baseline that performs training from scratch without using Evol-TL. The results show that Evol-TL facilitates much lower training and validation losses and combats overfitting even with only a small amount of data. Whereas, a clear divergence of the validation loss from the training loss is observed when Evol-TL is not involved. Fig. 4(d) shows the comparison of the accuracy performance from Evol-TL and the baseline with different amounts of training data used. It can be seen that Evol-TL can achieve an accuracy of $> 95\%$ with only 36 new data instances. However, without Evol-TL, the accuracy at this point is only 62.88%, and to achieve the same accuracy performance, another 424 instances are needed. The results indicate a 12.8$\times$ training data reduction by Evol-TL. In Fig. 4(e), we summarize the results of the number of required training data instances to reach the 95% accuracy threshold and the asymptotic accuracy for the two target tasks, where asymptotic accuracy is defined as the accuracy that can be achieved using the full training datasets. The results show the similar performance of Evol-TL with the two tasks, with the asymptotic accuracies being $\sim 99\%$.

### B. Results for Use Case 2

Next, we evaluated the performance of Evol-TL when transferring knowledge between tasks adopting different modulation formats. Fig. 5 (a) shows the convergence process of Evol-TL with $\mathcal{M}_T^{2,1}$, where 100 data instances were used. Similar to the observations that can be drawn from the evaluations for use case *1*, Evol-TL converges at the tenth iteration and achieves an accuracy of 95.9%. Based on the solution conveyed by the best individual, a fully-connected neural network of five hidden layers should be used, among which three are newly inserted. Different from the optimal configuration for $\mathcal{M}_T^{1,1}$, where only the first layer is copied from the pre-trained model, the best configuration for $\mathcal{M}_T^{2,1}$ takes two pre-trained layers. We presume that this is because $\mathcal{M}_T^{2,1}$ is more similar to $\mathcal{M}_S$ compared with $\mathcal{M}_T^{1,1}$, as typically, the more similar two tasks are, the more knowledge is useful for transferring [50]. Fig. 5 (b) shows that the model is well fitted when Evol-TL is applied while Fig. 5 (c) indicates overfitting due to the short of training data. From Fig. 5 (d), we can see that Evol-TL can achieve $\sim 11\times$ reduction in the amount of training data needed to reach the 95% accuracy threshold. Fig. 5 (e) summarizes the results of asymptotic accuracy and the amount of training data required to achieve 95% accuracy. The results show a smaller (i.e., 7$\times$) data reduction from Evol-TL with $\mathcal{M}_T^{2,2}$ compared to that with $\mathcal{M}_T^{2,1}$. This is because the difference between $\mathcal{M}_T^{2,2}$ and $\mathcal{M}_S^1$ is more significant, which degrades the effectiveness of knowledge transfer. For both tasks, asymptotic accuracies of $> 98\%$ can be achieved.

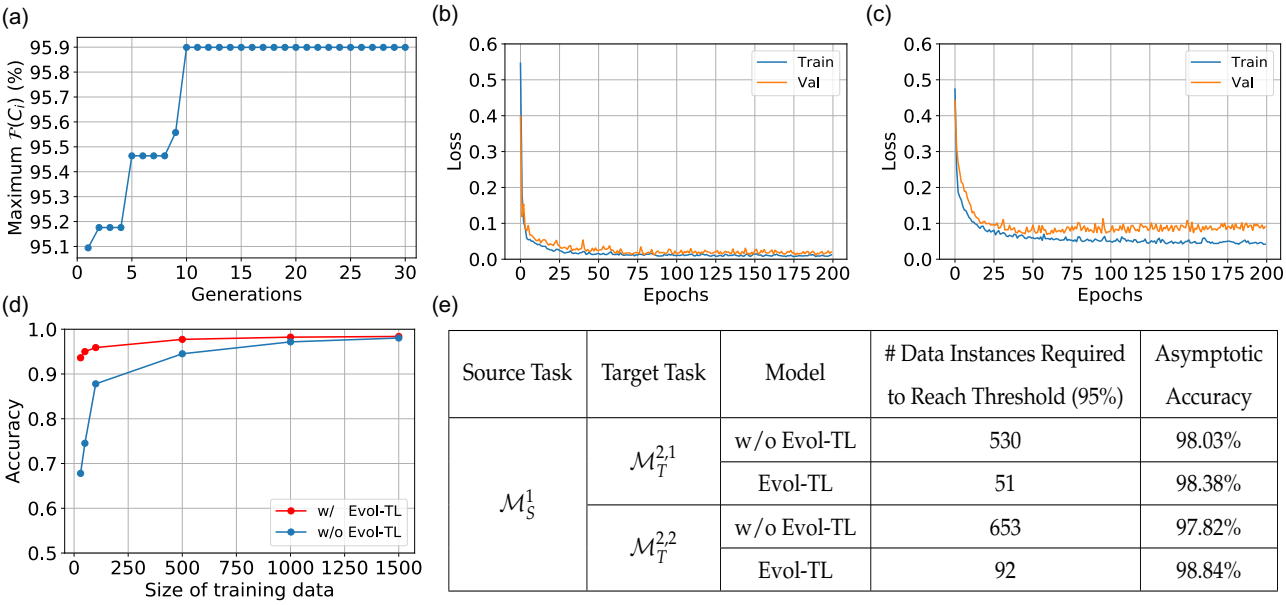
### C. Results for Use Case 3

We evaluated the performance of Evol-TL in the case where we transfer knowledge between the QoT models for lightpaths confronting different device conditions. Fig. 6 (a) shows the convergence process of Evol-TL with $\mathcal{M}_T^{3,1}$, where 50 training data instances were used. The results show that Evol-TL derived the best individual with an accuracy of 97.15% at the fifteenth iteration. The solution encoded by the individual suggests that the QoT estimator for $\mathcal{M}_T^{3,1}$ should be built with one hidden layer from the pre-trained model and three new layers of 19 neurons. Again, this observation can be explained by the fact that $\mathcal{M}_T^{3,1}$ differs from $\mathcal{M}_S$ in both path length and device condition, i.e., exhibiting more significant differences. Figs. 6(b) and (c) give the comparison of training and validation losses between Evol-TL and the baseline that learns from scratch, which show




| Source Task | Target Task | Model | # Data Instances Required to Reach Threshold (95%) | Asymptotic Accuracy |
|---|---|---|---|---|
| $\mathcal{M}_S^1$ | $\mathcal{M}_T^{1,1}$ | w/o Evol-TL | 460 | 98.77% |
| | | Evol-TL | 36 | 98.95% |
| | $\mathcal{M}_T^{1,2}$ | w/o Evol-TL | 546 | 98.22% |
| | | Evol-TL | <30 | 99.02% |

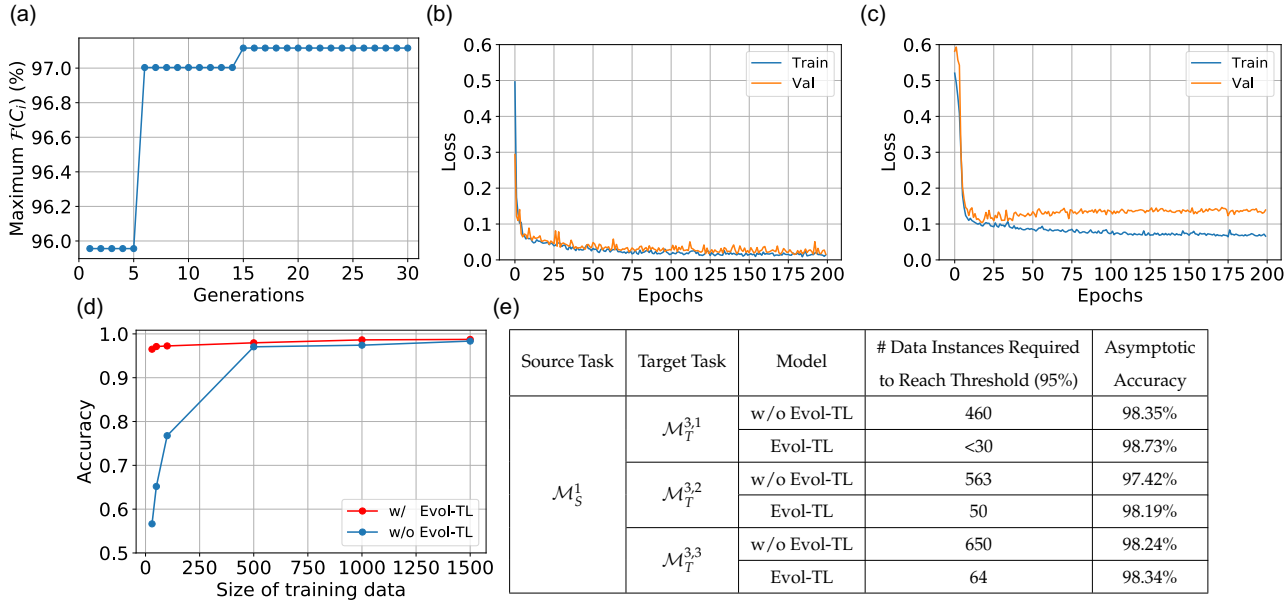
**Fig. 4.** Results for use case 1: (a) convergence process of Evol-TL with $\mathcal{M}_T^{1,1}$; (b-c) evolutions of the training and validation losses vs training epochs (b) with Evol-TL and (c) without Evol-TL. (number of training data instances = 50); (d) accuracy performance with $\mathcal{M}_T^{1,1}$ as a function of the size of training data set used; (e) results of required training data instances to reach above 95% accuracy and asymptotic accuracy for $\mathcal{M}_T^{1,1}$ and $\mathcal{M}_T^{1,2}$.




| Source Task | Target Task | Model | # Data Instances Required to Reach Threshold (95%) | Asymptotic Accuracy |
|---|---|---|---|---|
| $\mathcal{M}_S^1$ | $\mathcal{M}_T^{2,1}$ | w/o Evol-TL | 530 | 98.03% |
| | | Evol-TL | 51 | 98.38% |
| | $\mathcal{M}_T^{2,2}$ | w/o Evol-TL | 653 | 97.82% |
| | | Evol-TL | 92 | 98.84% |

**Fig. 5.** Results for use case 2: (a) convergence process of Evol-TL with $\mathcal{M}_T^{2,1}$; (b-c) evolutions of the training and validation losses vs training epochs (b) with Evol-TL and (c) without Evol-TL. (number of training data instances = 100); (d) accuracy performance with $\mathcal{M}_T^{2,1}$ as a function of the size of training data set used; (e) results of required training data instances to reach above 95% accuracy and asymptotic accuracy for $\mathcal{M}_T^{2,1}$ and $\mathcal{M}_T^{2,2}$.

trends similar to those we can observe from Figs. 4 and 5. We plotted the accuracy performance with $\mathcal{M}_T^{3,1}$ as a function of the size of the training data set in Fig. 6(d). It can be seen that Evol-TL can achieve a 16× reduction in the amount of training data needed for reaching the 95% accuracy threshold. We summarize the results with $\mathcal{M}_T^{3,1}$, $\mathcal{M}_T^{3,2}$, and $\mathcal{M}_T^{3,3}$ in Fig. 6(e). The results show that Evol-TL achieves around 11× reductions in the amount of data required for reaching the 95% threshold with both $\mathcal{M}_T^{3,2}$ and $\mathcal{M}_T^{3,3}$, confirming the effectiveness of the proposed approach in confronting diversified device conditions.




| Source Task | Target Task | Model | # Data Instances Required to Reach Threshold (95%) | Asymptotic Accuracy |
|---|---|---|---|---|
| $\mathcal{M}_S^1$ | $\mathcal{M}_T^{3,1}$ | w/o Evol-TL | 460 | 98.35% |
| | | Evol-TL | <30 | 98.73% |
| | $\mathcal{M}_T^{3,2}$ | w/o Evol-TL | 563 | 97.42% |
| | | Evol-TL | 50 | 98.19% |
| | $\mathcal{M}_T^{3,3}$ | w/o Evol-TL | 650 | 98.24% |
| | | Evol-TL | 64 | 98.34% |

**Fig. 6.** Results for use case 3: (a) convergence process of Evol-TL with $\mathcal{M}_T^{3,1}$; (b-c) evolutions of the training and validation losses vs training epochs (b) with Evol-TL and (c) without Evol-TL. (number of training data instances = 50); (d) accuracy performance with $\mathcal{M}_T^{3,1}$ as a function of the size of training data set used; (e) results of required training data instances to reach above 95% accuracy and asymptotic accuracy for $\mathcal{M}_T^{3,1}$, $\mathcal{M}_T^{3,2}$ and $\mathcal{M}_T^{3,3}$.

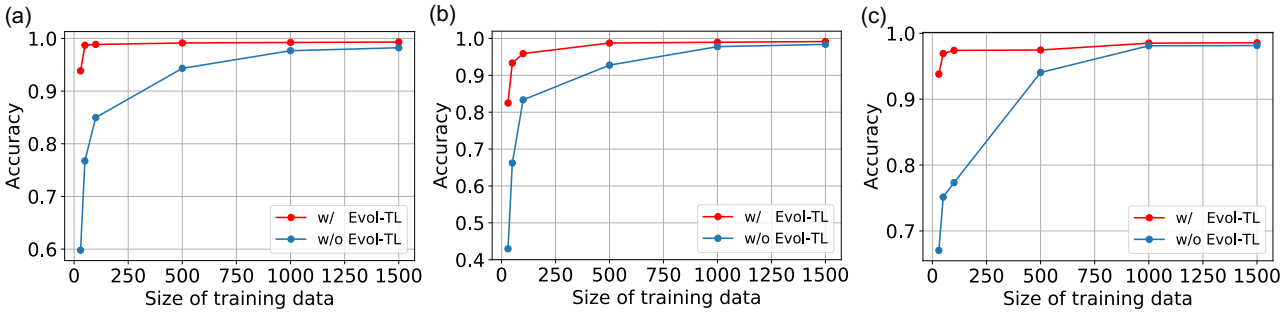
### D. Scalability Studies

Finally, we performed scalability studies of Evol-TL with evaluations using lightpaths traversing three domains, i.e., considering the source task being $\mathcal{M}_S^2$ and tasks $\mathcal{M}_T^{1,3}$, $\mathcal{M}_T^{2,3}$, and $\mathcal{M}_T^{3,4}$ being the target tasks. Fig. 7 shows the results of the amount of training data needed to achieve 95% accuracy and the asymptotic accuracies. We can see that Evol-TL can achieve training data reductions of around 17×, 9× and 16× for $\mathcal{M}_T^{1,3}$, $\mathcal{M}_T^{2,3}$, and $\mathcal{M}_T^{3,4}$, respectively. For all three tasks, asymptotic accuracies of higher than 98% can be achieved. The results show solid performance of Evol-TL under more complex lightpath configurations, which, to a certain extent, verifies the scalability of the proposed approach. Meanwhile, we want to emphasize that the input size of a multi-domain QoT estimator increases linearly with the length of a lightpath. In an extreme case where a lightpath traverses ten domains, the input size for the QoT estimator is 17*10 = 170 (8 bits for conveying the position of the testing channel in the one-hot form, 8 bits for representing whether each of the eight background channels is active, and 1 bit for carrying the average power of the background channels). On the other hand, recent studies have demonstrated successful knowledge transfer with similar approaches (i.e., transferring knowledge by reusing trained weights) applied to deep neural networks with input sizes of more than three hundred for different applications [52, 53]. Therefore, we expect that our approach can also scale to larger-scale multi-domain networks.

## 5. RELATED WORK

Recent research activities in ML-aided cognitive optical networking mainly address QoT estimation, fault management, and resource allocation tasks [12–15].

*QoT estimation*: In [8], Pointurier reviewed various margins used in optical networks, i.e., unallocated margins (the differences between user demands and system capacities), system margins (time-varying operating conditions owing to evolving physical-layer impairments and traffic profiles) and design margins (the differences between the planned and the real QoT values, typically, caused by inaccurate QoT estimations), and discussed the techniques for handling these margins. In [17], the authors developed a random forest-based classifier to predict whether the QoT of an unestablished lightpath meets the minimum system requirements. The work in [18] evaluated the effectiveness of different ML-aided QoT models [e.g. K-nearest neighbors, logistic regression, support vector machine (SVM), artificial neural networks (ANNs)] for an unestablished lightpath. The results indicate that ANNs achieve better generalization with prediction accuracy of nearly 99.9% when compared with the other models (accuraries > 90%). The authors of [19] compared the QoT estimation models built with different ML schemes, i.e., random forest (RF), K-nearest neighbors, and SVM, showing that the SVM-based model achieves superior performance. The authors of [20] studied two QoT estimator designs: ML physical layer model (ML-PLM) and ML model (ML-M). ML-PLM makes use of ML to learn the input parameters of the PLM, whereas ML-M learns the QoT models directly from the OPM data. The ML-PLM can achieve higher estimation accuracies but requires longer training time, while ML-M offers better flexibility as it does not rely on prior physical layer knowledge. In [21], the authors demonstrated a novel deep graph convolutional neural network-based QoT estimator design that can capture unseen network states (e.g., inter-core crosstalk) in optical networks using multi-core fibers. Our previous work in [22] experimentally demonstrated an alien wavelength monitoring scheme in MD-EONs. Based on the proposed monitoring scheme, an ML-aided QoT estimator design was also studied, showing an optical-signal-to-noise ratio (OSNR) estimation error of $< 6\%$.




(d)

| Source Task | Target Task | Model | # Data Instances Required to Reach Threshold (95%) | Asymptotic Accuracy |
|---|---|---|---|---|
| $\mathcal{M}_S^2$ | $\mathcal{M}_T^{1,3}$ | w/o Evol-TL | 603 | 98.22% |
| | | Evol-TL | 35 | 99.31% |
| | $\mathcal{M}_T^{2,3}$ | w/o Evol-TL | 724 | 98.38% |
| | | Evol-TL | 83 | 99.15% |
| | $\mathcal{M}_T^{3,4}$ | w/o Evol-TL | 618 | 98.14% |
| | | Evol-TL | 38 | 98.56% |

**Fig. 7.** Results of the three-domain EON topology. (a)-(c) Accuracy performance with (a) $\mathcal{M}_T^{1,3}$ (b) $\mathcal{M}_T^{2,3}$ (c) $\mathcal{M}_T^{3,4}$ as a function of the size of training data set used. (d) results of required training data to achieve 95% accuracy and the asymptotic accuracies for $\mathcal{M}_T^{1,3}$, $\mathcal{M}_T^{2,3}$(c) and $\mathcal{M}_T^{3,4}$.

*Fault management*: In [23], Vela et al. proposed two algorithms to detect bit-error-rate (BER) degradation and to identify the root causes of the failures, including signal overlap, tight filtering, gradual drift, and cyclic drift. The authors in [24] proposed an approach, based on SVM and double exponential smoothing, to detect and forecast optical equipment failures in an optical network. The approach achieves a failure prediction accuracy of $>$ 95% according to the experimental results. In [25], Natalino et al. developed a SVM and ANN based framework for detecting and identifying optical jamming signal attacks. In [26], the authors devised a Gaussian process classifier complemented by a graph-based correlation heuristic to localize single-link failures in optical networks. The results show localization accuracies of $>$ 91% and remarkable reductions in monitoring cost (compared with the traditional schemes) from the proposed scheme. Our previous work in [27] proposed a hybrid unsupervised and supervised ML approach that can detect unknown-type soft failures from a large set of unlabeled OPM data. In [28], the authors performed an experimental study on different ML classifiers [e.g., RF, SVM, ANN] for detection and identification of physical-layer attacks. The results reveal that the ANN-based design can achieve the best performance, i.e., a classification accuracy of $>$ 99.9%. The authors of [29] proposed a dual-stage soft-failure detection and identification framework, where only the BER and optical power monitored by the coherent receivers are used for failure detection at the first stage. Once an anomalous sample is detected, extra digital spectrum features will be extracted and analyzed at the second state.

*Resource allocation*: In [30], Suárez-Varela et al. devised a deep reinforcement learning (DRL) scheme to manage the routing in optical transport networks with a refined representation of network states. In [31], we proposed a cognitive routing, modulation, and spectrum assignment (RMSA) scheme with a modified actor-critic algorithm. The results demonstrate that the proposed mechanism can achieve more blocking reductions than heuristic approaches. The authors of [32] developed an integer linear programming model assisted by an ML-based QoT estimator for optimizing the QoT-aware RMSA problem in EONs. In [33], the authors devised a reinforcement learning (RL) formulation to learn effective policies for determining the number of frequency slots to allocate in EONs. The results indicate that the proposed RL approach facilitates more efficient spectrum utilization while meeting the differentiated QoS requirements. In [34], the authors proposed a DRL-based service function chaining framework to manage the setup duration of virtual network functions in inter-datacenter optical networks.

The aforementioned existing works mostly focus on ML designs for enhanced network performance (i.e. higher QoT estimation accuracy or resource efficiency), while rarely taking into account important aspects such as data efficiency and training overhead. These practical aspects have been attracting increasing research interests lately as they impact the scalability and feasibility of various ML approaches and applications. In [41], Yu et al. applied an ANN-based QoT estimator design and showed that by transferring knowledge between systems using different modulation formats, only a small number of additional data instances are needed to fine-tune a new QoT estimator. Similarly, in [42], the authors demonstrated that TL can help to significantly reduce the amount of training data and training time required in OSNR estimation tasks. The authors of [43] investigated the accuracies achieved by active learning and transfer learning with small numbers of training samples and showed that the two approaches facilitate similar performance gains. The results show comparable improvements of accuracy can be obtained either with active learning or transfer learning. In [44], the authors compared two types of domain adaption approaches (i.e., feature augmentation and correlation alignment) applied to estimate the QoT of an unestablished lightpath. The results indicate the domain adaption approaches can effectively enhance the model accuracy in the cases when training data

instances are rare in the target domains. The authors of [45] made use of TL to facilitate the training of an ML model used for predicting the spectrum defragmentation time. More recently, we investigated the TL design for DRL applications in EONs and proposed to train a multi-task learning agent for transferring better-generalized knowledge across tasks [46]. Case studies for the proposed approach with RMSA tasks under different topologies demonstrated remarkable reductions in the number of training steps required.

## 6. CONCLUSION

In this paper, we proposed an evolutionary transfer learning design for scalable QoT estimation in MD-EONs. We investigated three use cases, i.e., knowledge transferring between QoT estimation tasks for lightpaths with *i*) different numbers of nodes, *ii*) different modulation formats, and *iii*) different device conditions. The results from the performance simulation studies reveal that Evol-TL can reduce significantly the amount of training data needed for the destination tasks by optimizing the neural network design and transfer learning process with a genetic algorithm. Future research directions include: building an MD-EON testbed and evaluating the performance of Evol-TL with experimental data covering more diversified network scenarios, for instance, lightpaths using different baud rates, channel spacing, and vendor devices; adapting Evol-TL to the service provisioning (e.g., routing and spectrum assignment) and fault management (e.g., soft-failure detection and localization) tasks in MD-EONs.

## ACKNOWLEDGMENT

This work was supported in part by NSF ICE-T:RC award # 1836921.

## REFERENCES

1. O. Gerstel, M. Jinno, A. Lord, and S. J. B. Yoo, "Elastic optical networking: a new dawn for the optical layer?" IEEE Commun. Mag. **50**, s12–s20 (2012).
2. L. Gong, X. Zhou, X. Liu, W. Zhao, W. Lu, and Z. Zhu, "Efficient resource allocation for all-optical multicasting over spectrum-sliced elastic optical networks," J. Opt. Commun. Netw. **5**, 836–847 (2013).
3. L. Liu, Y. Yin, M. Xia, M. Shirazipour, Z. Zhu, R. Proietti, Q. Xu, S. Dahlfort, and S. J. B. Yoo, "Software-defined fragmentation-aware elastic optical networks enabled by openflow," in *Proc. Eur. Conf. Opt. Commun.*, (2013). Paper We.3.E.2.
4. Z. Zhu, W. Lu, L. Zhang, and N. Ansari, "Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing," J. Light. Technol. **31**, 15–22 (2013).
5. Y. Yin, H. Zhang, M. Zhang, M. Xia, Z. Zhu, S. Dahlfort, and S. J. B. Yoo, "Spectral and spatial 2D fragmentation-aware routing and spectrum assignment algorithms in elastic optical networks," J. Opt. Commun. Netw. **5**, A100–A106 (2013).
6. L. Gong and Z. Zhu, "Virtual optical network embedding (VONE) over elastic optical networks," J. Light. Technol. **32**, 450–460 (2014).
7. P. Lu, L. Zhang, X. Liu, J. Yao, and Z. Zhu, "Highly-efficient data migration and backup for big data applications in elastic optical inter-datacenter networks," IEEE Netw. **29**, 36–42 (2015).
8. Y. Pointurier, "Design of low-margin optical networks," J. Opt. Commun. Netw. **9**, A9–A17 (2017).
9. H. Beyranvand and J. A. Salehi, "A quality-of-transmission aware dynamic routing and spectrum assignment scheme for future elastic optical networks," J. Light. Technol. **31**, 3043–3054 (2013).
10. I. Sartzetakis, K. Christodoulopoulos, C. P. Tsekrekos, D. Syvridis, and E. Varvarigos, "Quality of transmission estimation in wdm and elastic optical networks accounting for space–spectrum dependencies," IEEE/OSA J. Opt. Commun. Netw. **8**, 676–688 (2016).
11. P. Layec, A. Dupas, A. Bisson, and S. Bigo, "Qos-aware protection in flexgrid optical networks," IEEE/OSA J. Opt. Commun. Netw. **10**, A43–A50 (2018).
12. D. Rafique and L. Velasco, "Machine learning for network automation: Overview, architecture, and applications [invited tutorial]," J. Opt. Commun. Netw. **10**, D126–D143 (2018).
13. F. Khan, Q. Fan, C. Lu, and A. Lau, "An optical communication's perspective on machine learning and its applications," J. Light. Technol. **37**, 493–516 (2019).
14. F. Musumeci, C. Rottondi, G. Corani, S. Shahkarami, F. Cugini, and M. Tornatore, "A tutorial on machine learning for failure management in optical networks," J. Light. Technol. **37**, 4125–4139 (2019).
15. F. Musumeci, C. Rottondi, A. Nag, T. Macaluso, D. Zibar, M. Ruffini, and M. Tornatore, "An overview on application of machine learning techniques in optical networks," IEEE Commun. Surv. Tutor. **21**, 1383–1408 (2019).
16. X. Chen, R. Proietti, H. Lu, A. Castro, and S. J. B. Yoo, "Knowledge-based autonomous service provisioning in multi-domain elastic optical networks," IEEE Commun. Mag. **56**, 152–158 (2018).
17. L. Barletta, A. Giusti, C. Rottondi, and M. Tornatore, "Qot estimation for unestablished lighpaths using machine learning," In Proc. Conf. Opt. Fiber Commun (OFC)., 2017, Paper Th1J.1.
18. R. M. Morais and J. ao Pedro, "Machine learning models for estimating quality of transmission in dwdm networks," J. Opt. Commun. Netw. **10**, D84–D99 (2018).
19. S. Aladin and C. Tremblay, "Cognitive tool for estimating the qot of new lightpaths," In Proc. Conf. Opt. Fiber Commun (OFC)., 2018, Paper M3A.3.
20. 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).
21. T. Panayiotou, G. Savva, B. Shariati, I. Tomkos, and G. Ellinas, "Machine learning for qot estimation of unseen optical network states," In Proc. Conf. Opt. Fiber Commun (OFC)., 2019, Paper Tu2E.2.
22. R. Proietti, X. Chen, K. Zhang, G. Liu, M. Shamsabardeh, A. Castro, L. Velasco, Z. Zhu, and S. J. B. Yoo, "Experimental demonstration of machine-learning-aided qot estimation in multi-domain elastic optical networks with alien wavelengths," J. Opt. Commun. Netw. **11**, A1–A10 (2019).
23. A. P. Vela, M. Ruiz, F. Fresi, N. Sambo, F. Cugini, G. Meloni, L. Potì, L. Velasco, and P. Castoldi, "Ber degradation detection and failure identification in elastic optical networks," J. Light. Technol. **35**, 4595–4604 (2017).
24. Z. Wang, M. Zhang, D. Wang, C. Song, M. Liu, J. Li, L. Lou, and Z. Liu, "Failure prediction using machine learning and time series in optical network," Opt. Express **25**, 18553–18565 (2017).
25. C. Natalino, M. Schiano, A. Di Giglio, L. Wosinska, and M. Furdek, "Field demonstration of machine-learning-aided detection and identification of jamming attacks in optical networks," in *Proc. Eur. Conf. Opt. Commun.*, (2018), pp. 1–3.
26. T. Panayiotou, S. P. Chatzis, and G. Ellinas, "Leveraging statistical machine learning to address failure localization in optical networks," IEEE/OSA J. Opt. Commun. Netw. **10**, 162–173 (2018).
27. X. Chen, B. Li, R. Proietti, Z. Zhu, and S. J. B. Yoo, "Self-taught anomaly detection with hybrid unsupervised/supervised machine learning in optical networks," J. Light. Technol. **37**, 1742–1749 (2019).
28. C. Natalino, M. Schiano, A. Giglio, L. Wosinska, , and M. Furdek, "Experimental study of machine-learning-based detection and identification of physical-layer attacks in optical networks," J. Light. Techn. **37**, 4173–4182 (2019).
29. L. Shu, Z. Yu, Z. Wan, J. Zhang, S. Hu, and K. Xu, "Dual-stage soft failure detection and identification for low-margin elastic optical network by exploiting digital spectrum information," J. Light. Technol. **38**, 2669–2679 (2020).
30. J. Suarez-Varela, A. Mestres, J. Yu, L. Kuang, H. Feng, A. Cabellos-Aparicio, and P. Barlet-Ros, "Routing in optical transport networks with

deep reinforcement learning," J. Opt. Commun. Netw. **11**, 547–558 (2019).
31. X. Chen, B. Li, R. Proietti, H. Lu, Z. Zhu, and S. J. B. Yoo, "DeepRMSA: A deep reinforcement learning framework for routing, modulation and spectrum assignment in elastic optical networks," J. Light. Technol. **37**, 4155–4163 (2019).
32. M. Salani, C. Rottondi, and M. Tornatore, "Routing and spectrum assignment integrating machine-learning-based qot estimation in elastic optical networks," In Proc. IEEE Conf. Comput. Commun. (INFOCOM), pp. 1738–1746 (2019).
33. T. Panayiotou, K. Manousakis, S. P. Chatzis, and G. Ellinas, "A data-driven bandwidth allocation framework with qos considerations for eons," J. Light. Technol. **37**, 1853–1864 (2019).
34. B. Li, W. Lu, and Z. Zhu, "Deep-nfvorch: leveraging deep reinforcement learning to achieve adaptive vnf service chaining in dci-eons," IEEE/OSA J. Opt. Commun. Netw. **12**, A18–A27 (2020).
35. G. Liu, K. Zhang, X. Chen, H. Lu, J. Guo, J. Yin, R. Proietti, Z. Zhu, and S. J. B. Yoo, "Hierarchical learning for cognitive end-to-end service provisioning in multi-domain autonomous optical networks," J. Light. Technol. **37**, 218–225 (2019).
36. S. J. Pan and Q. Yang, "A survey on transfer learning," IEEE Transactions on Knowl. Data Eng. **22**, 1345–1359 (2010).
37. P. Swietojanski, A. Ghoshal, and S. Renals, "Unsupervised cross-lingual knowledge transfer in dnn-based lvcsr," in *2012 IEEE Spoken Language Technology Workshop (SLT),* (2012), pp. 246–251.
38. J. Huang, J. Li, D. Yu, L. Deng, and Y. Gong, "Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing,* (2013), pp. 7304–7308.
39. X. Cao, D. Wipf, F. Wen, G. Duan, and J. Sun, "A practical transfer learning algorithm for face verification," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV),* (2013).
40. A. R. Zamir, A. Sax, W. Shen, L. J. Guibas, J. Malik, and S. Savarese, "Taskonomy: Disentangling task transfer learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* (2018).
41. J. Yu, W. Mo, Y. 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).
42. 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).
43. D. Azzimonti, C. Rottondi, A. Giusti, M. Tornatore, and A. Bianco, "Active vs transfer learning approaches for qot estimation with small training datasets," In Proc. Conf. Opt. Fiber Commun (OFC)., 2020, Paper M4E.1.
44. C. Rottondi, R. di Marino, M. Nava, A. Giusti, and A. Bianco, "On the benefits of domain adaptation techniques for quality of transmission estimation in optical networks," IEEE/OSA J. Opt. Commun. Netw. **13**, A34–A43 (2021).
45. Q. Yao, H. Yang, A. Yu, and J. Zhang, "Transductive transfer learning-based spectrum optimization for resource reservation in seven-core elastic optical networks," J. Light. Technol. **37**, 4164–4172 (2019).
46. X. Chen, R. Proietti, C. Liu, Z. Zhu, and S. J. Ben Yoo, "Exploiting multi-task learning to achieve effective transfer deep reinforcement learning in elastic optical networks," In Proc. Conf. Opt. Fiber Commun (OFC)., 2020, Paper M1B.3.
47. C. Liu, X. Chen, R. Proietti, and S. J. Ben Yoo, "Evol-tl: Evolutionary transfer learning for qot estimation in multi-domain networks," In Proc. Conf. Opt. Fiber Commun (OFC)., 2020, Paper Th3D.1.
48. F. Paolucci, A. Sgambelluri, F. Cugini, and P. Castoldi, "Network telemetry streaming services in SDN-based disaggregated optical networks," J. Light. Technol. **36**, 3142–3149 (2018).
49. X. Chen, B. Li, R. Proietti, C.-Y. Liu, Z. Zhu, and S. J. B. Yoo, "Demonstration of distributed collaborative learning with end-to-end qot estimation in multi-domain elastic optical networks," Opt. Express **27**, 35700–35709 (2019).
50. J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" Adv. Neural Inf. Process. Syst. pp. 3320–3328 (2014).
51. J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," J. Mach. Learn. Res. **13**, 281–305 (2012).
52. Y. Cui, Y. Song, C. Sun, A. Howard, and S. Belongie, "Large scale fine-grained categorization and domain-specific transfer learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* (2018).
53. T. Wen and R. Keyes, "Time series anomaly detection using convolutional neural networks and transfer learning," in *AI for Internet of Things Workshop in IJCAI,* (2019).