Designing Ultra-High Bandwidth Optical Networks Using Machine Learning Techniques - Extended Abstract

Alexandre Paulo da Costa Freitas, 90005

***Abstract*—This thesis presents a machine learning solution, specifically one based on Deep Neural Network (DNN) models, with the objective of quickly and reliably determining the total network capacity and average channel capacity of transparent optical backbone networks based on the networks' physical topology parameters. Furthermore, to overcome C-band transmission bandwidth limitations, this thesis also proposes adding extra optical fibers to the network, eliminating traffic blocking and increasing network capacities. A second DNN model that predicts the total cost in kilometers of fiber deployed and the network capacity was developed as a result. To generate training data for the DNN models a generative graph model that creates networks similar to real-world optical backbone networks and a heuristic routing algorithm were implemented. The DNN models were shown to provide accurate predictions in just a few milliseconds, proving to be useful tools that can assist in optical backbone network design.**

***Index Terms*—transparent optical backbone network, optical channel, capacity, optical reach, deep neural network.**

I. Introduction

OPTICAL networks are telecommunications networks where data is transmitted using light through optical fiber links. These networks offer very high bandwidths and allow for transmission over great distances. For that reason, these networks have become increasingly more prevalent over the last decades due to the rapid pace at which telecommunications traffic has been growing [1]. Predictions indicate that this trend in traffic growth is expected to continue for the next years. This makes optical networks fundamental in the evolving telecommunications field, making the development of solutions related to this type of networks particularly relevant.

Optical networks can be divided into different types depending on their geographic extension and capacity [2]: (a) backbone networks, which span hundreds to thousands of kilometers, and carry traffic from millions of users, offering very high capacities (in the order of dozens of Tbit/s); (b) metro networks, which cover cities and metropolitan areas, spanning tens to hundreds of kilometers, and offer capacities in the order of the hundreds of Gbit/s; and (c) access networks, which span smaller areas (few kilometers), connecting end-users to the network providers and delivering data rates on the order of a few Gbit/s. Optical networks can also be classified as opaque, transparent or translucent. In opaque networks, the node functions (such as multiplexing, switching, routing, etc.) take place in the electrical domain, while in transparent networks this is done in the optical domain. In translucent networks there are nodes operating in the electrical domain and others operating in the optical domain.

The work developed in this dissertation project focuses on transparent optical backbone networks. In this type of networks the nodes are typically implemented using Reconfigurable Optical Add-Drop Multiplexers (ROADMs), which are optical devices that allow for the switching of traffic at the wavelength level, and the links are implemented with optical fibers.

Wavelength Division Multiplexing (WDM) is an essential technology in optical networks that allows for the concurrent transmission of multiple optical signals (also called optical channels) on the same optical fiber, each channel using a different wavelength. The number of optical channels that can co-exist in a fiber depends on the spacing between the channels as well as the bandwidth of the WDM signal. Typically, optical backbone networks function within the C-band, which has a bandwidth of approximately 4800 GHz. This bandwidth corresponds to that of the Erbium-Doped Fiber Amplifier (EDFA), the most prevalent optical amplifier used to compensate fiber losses [3].

A way to considerably increase the number of optical channels in a network is to utilize Band Division Multiplexing (BDM) or Space Division Multiplexing (SDM). BDM explores the use of bands other than the C-band which includes, for example, multi-band transmission on the C+L+S-bands. SDM relies, for example, on adding more optical fibers to the network's links, while still operating in the C-band. The complexity of the physical modeling in the case of multi-band transmission (BDM solution) motivated the author to explore the use of an SDM solution as a way of implementing networks with higher bandwidths.

An important way of assessing the performance of optical networks is the network capacity. The network capacity is the maximum amount of data that the entire network can handle per unit of time. It is related to the channel capacity, which is the maximum data rate at which the information can be transmitted through a noisy medium without errors [4].

However, the determination of the optical channel capacity and network capacity can be complex tasks due to non-linear effects present in optical fiber transmission as well as the need to consider both physical layer characteristics and network layer aspects such as topology, traffic demands, and routing. Mathematical and analytical models can estimate channel capacity, but they often involve long computations. Some studies have addressed this problem using Integer Linear

Programming (ILP) or heuristic algorithms (see [5] and references therein), but the computation time for large networks remains very high.

To address these computational limitations, Machine Learning (ML) techniques, notably Artificial Neural Network (ANN) models (which includes DNNs), have emerged as effective solutions. These types of models have seen a wide and successful use in the field of optical networks, providing precise results and low computation times [6], [7].

For this reason, the main objective of this dissertation is to develop a DNN model capable of determining the network capacity and the average channel capacity of transparent optical backbone networks operating in the C-band, accounting for the aspects related to both the physical and the network layer. This model should be able to predict these outputs based on input parameters related to the network's physical topology in a fast and accurate manner. Furthermore, given the SDM solution proposed to overcome the bandwidth limitations of C-band transmission, a second DNN model that predicts the networks' capacity and the total cost in kilometers of fiber was also developed.

The training of neural networks requires the use of a large amount of data. For that reason, the use of a generative graph model is essential, as it provides a way of obtaining thousands of artificial networks from which that data can be obtained. This generative model should be able to produce graphs with topologies similar to that of real-world optical backbone networks. From these networks, the set of features (the inputs of the DNN model, which are parameters related to the physical topology of the networks) and labels (the outputs of the DNN model, that is, the total network capacity, the average channel capacity and fiber cost) are then determined. This process involves the consideration of the physical layer aspects (which were accounted for through the optical reach), and the network layer aspects, for which a routing algorithm needs to be used. Given the consideration of the two scenarios, with and without SDM, two types of routing algorithm were developed: (a) a constrained routing algorithm, that takes into account the limitations in the number of optical channels in the optical fibers; and (b) an unconstrained routing algorithm with fiber assignment, a routing solution where the limitation in the number of optical channels was overcome through the introduction of additional optical fibers to the network's links. For the SDM solution two fiber assignment algorithms were developed, one focused on computational performance (which is essential when determining the parameters of thousands of networks), and the other where a more comprehensive analysis on the placement of fibers is made, leading to less fibers being introduced (at the expense of longer computation times).

The work developed in this thesis, specifically the generative graph model and the routing algorithms, led to the publication of an article focused on the topic of capacity in optical backbone networks, analyzing how the different network and physical layer parameters influence this value. This article was published on the journal "Photonics" [5]: *A. Freitas and J. Pires, "Investigating the Impact of Topology and Physical Impairments on the Capacity of an Optical Backbone Network," Photonics, vol. 11, no. 4, 2024.*

The remainder of this extended abstract is organized as follows: Section II lays out the fundamental concepts that serve as the basis for the developed work; Section III provides an overview of the generative graph model; Section IV details the heuristic algorithms for the routing implementation and network parameter determination; Section V discusses the implementation and testing of the DNN models; and Section VI summarizes the main conclusions of the work.

II. Fundamental Concepts

A. Network Characterization and Routing

Telecommunication networks can be defined by their physical and logical topologies.

The physical topology of a network describes the physical layout of the various elements of the network. It can be represented as a graph, $G = (V, E)$, with $V = \{v_1, ..., v_N\}$ being the set of vertices (or nodes) and $E = \{e_1, ..., e_K\}$ the set of edges (or links), where N is the number of nodes and K is the number of links. The links can have attributes associated to them, such as length, capacity, cost and others. When a graph has links with such attributes (also called weights) it is called a weighted graph, otherwise it is called an unweighted graph. In the context of this work, weighted graphs with the weights representing the distance between the nodes (link lengths) are considered. Furthermore, the physical topology of a network can also be represented by an adjacency matrix. The adjacency matrix of a network with N nodes, A, is a square ($N \times N$) matrix that represents a graph. For weighted graphs, the element a_{ij} of the matrix will have the value of the link's weight (link length in this case).

The logical topology of a network describes how the data is transmitted. It is represented by a demand matrix, D, which is a square ($N \times N$) matrix where the element d_{ij} is equal to 1 if there is a traffic demand between nodes v_i and v_j, or zero if that demand doesn't exist. In cases where static traffic is assumed (when the traffic demands are known *a priori* and remain constant in time), as is the case in this work, the traffic demands can be described through a traffic matrix. A traffic matrix, T, is a square ($N \times N$) matrix where the element t_{sd} represents the volume of traffic per unit time (usually in bit/s) from node s (source) to node d (destination). In this work a full-mesh logical topology with unitary traffic is considered, meaning that there is a single traffic demand between every pair of nodes in the network. In this case there are $N(N-1)$ unidirectional traffic demands and the traffic matrix can be given by:

$$t_{s,d} = \begin{cases} 1, & \text{if } s \neq d \\ 0, & \text{if } s = d \end{cases} \tag{1}$$

Related to both the physical and logical topology, there are various network parameters that are relevant in the context of this work: (a) path ($\pi_{s,d}$), represents the physical route of an optical channel, defining the sequence of links/nodes from s to d; (b) path length $l(\pi_{s,d})$, is the sum of the weights of all the links in that path (sum of all link lengths); (c) distance, is the length of the shortest path between two nodes; (d) diameter of a network, is the maximum number of links between any pair of nodes considering shortest-path routing, i.e., the number of links in the longest shortest path; (e) node degree, δ_i, is the number of links incident on node v_i; (f) average node degree of a network, $\langle\delta\rangle$, can be obtained by computing the average of the node degree across all the nodes in the network; (g) number of hops per demand, is defined as the number of links that are traversed by a traffic demand; (h) link load, is the sum of all the traffic in a link; (i) average link load, is the average of the traffic loads in every link of the network; (j) capacity of a link, is the maximum amount of traffic that link can accommodate; (k) residual capacity of a link, is the difference between its capacity and load; (l) node connectivity (κ-connectivity), $\kappa(G)$, is the minimum number of nodes that can be removed from a graph to disconnect it; (m) edge connectivity (λ-connectivity), $\lambda(G)$, is the minimum number of edges that can be removed from a graph to disconnect it; and (n) algebraic connectivity, a parameter related to the robustness of a network (larger algebraic connectivity being associated to better connected networks), given by the second smallest eigenvalue of the graphs's Laplacian matrix L (with $L = D_\delta - A$, where A is the graph's adjacency matrix and D_δ is the degree matrix, a diagonal matrix that contains the node degree for each node) [8], [9].

Also associated to the physical and logical topologies is the concept of routing. Routing is the process responsible for mapping the logical topology on the physical topology; that is, it involves selecting the path ($\pi_{s,d}$) to be followed by a traffic flow associated with a specific traffic demand. There are two types of routing processes: (a) the constrained routing process, where the objective is to maximize the allocated traffic for a given physical topology, link capacity and traffic matrix, with the requirement that the traffic in each link cannot exceed its capacity (during this process, if there are no links with enough residual capacity for a given traffic demand, then that traffic demand is blocked); and (b) the unconstrained routing process, where the goal is to find, for a given physical topology and traffic matrix, the set of paths that support all the traffic demands and to determine the capacities needed for each of the links in the network.

In either case, there is a general strategy for routing that can be followed for static traffic problems. Given as input parameters the physical topology described as a graph $G(V,E)$ and a traffic matrix T, one should perform the following steps to route the traffic through the network:

- Find the shortest paths for all traffic demands using a heuristic algorithm (like *Dijkstra*'s algorithm).
- Order the demands according to a sorting strategy.
- Route the demands according to the previous ordering. To break a tie, choose the path that minimizes the load in the most loaded link.

To order the traffic demands there are the following strategies:

- Shortest-First: The demands with the shortest path length come first;
- Longest-First: The demands with the longest path length come first;
- Largest-First: The demands with the highest number of traffic units come first.

B. Channel Capacity, Network Capacity and Optical Reach

As previously mentioned, in WDM transmission multiple optical channels can co-exist on the same optical fiber by using different wavelengths. An optical channel can be defined as the communication pathway through which data is transmitted, in the optical domain, from a sender (s) to a destination (d). It is characterized by its carrier wavelength λ_c (or carrier frequency ν_c), bandwidth B_{ch} (in Hz), and bit rate R_b (in bit/s). The bit rate can be related to the symbol rate R_s (in baud) through $R_b = R_s \cdot \log_2(M)$, where M is the number of symbols of the modulation scheme used. The capacity of an optical channel (C_{ch}), in bit/s, is the maximum data rate at which information can be transmitted through the physical medium (optical fiber). It can be determined through the Shannon capacity theorem [4], under the assumption that the noise sources are modeled as additive, white and gaussian noise sources and considering the channel bandwidth (B_{ch}) equal to the symbol rate (R_s), being given by [10]:

$$C_{ch} = 2 \cdot R_s \cdot \log_2(1 + \mathit{SNR}) \tag{2}$$

where we have the Signal-to-Noise Ratio (SNR) at the receiver end, given by [10]:

$$\text{SNR} = \frac{P_{ch}}{N_0 \cdot R_s} \tag{3}$$

with P_{ch} being the received average optical power per channel (in watts), and N_0 the noise power spectral density (in watt/Hertz). In Eq. (2) the factor 2 results from the fact that in optical transmission it is possible for two optical channels to be transmitted through the same fiber, at same the frequency, by using two orthogonal polarization states, being referred to as Polarization Multiplexed (PM) optical channels.

The network-wide average channel capacity is defined in [11] by the following expression:

$$\bar{C}_{ch} = \frac{\sum_k C_{ch,k}}{\sum_k \gamma_k} \tag{4}$$

where γ_k is the expected utilization ratio of a channel k. Usually, for simplicity, γ_k is considered to be 1, and so, the sum in the numerator of Eq. (4) is equal to the total capacity of the network, and the sum in the denominator is the total number of channels in the network.

The total network capacity can then be determined with:

$$C_{net} = \bar{C}_{ch} \times (N_{ch} - N_{\text{blocked}}) \tag{5}$$

where N_{ch} is the total number of traffic demands and N_{blocked} the number of blocked traffic demands (in the cases of constrained routing).

The determination of the network capacity (C_{net}) and the average channel capacity ($\bar{C}_{ch}$), the two output parameters of the DNN model, are both dependant on the channel capacity (C_{ch}), which in turn depends on the SNR. It is through the SNR that the various physical aspects of the transmission are taken into account. The determination of the SNR involves the consideration of the Amplified Spontaneous Emission (ASE) noise (N_{ASE}), which is a result of the optical amplifiers, as well as the noise resultant from Non-linear Interference (NLI) (N_{nli}). In this thesis, as a way to simplify the determination of the channel capacity, avoiding the more complex formulations that describe the determination of the SNR, the computation of the optical channel capacity is done through the optical reach.

The optical reach (or transmission reach) can be defined as the maximum transmission distance over which the channel can maintain a data rate close to its Shannon capacity. In the context of this work it is used to account for the impact of the physical layer on the performance of the optical channels. Table I, from [5], shows the optical reach for different values of Shannon channel capacities (referred to as such because they are obtained using the Shannon theory) for transmission at 64 Gbaud.

TABLE I: Optical Reach for transmission at 64 Gbaud (From [5])

Reach (km)	Capacity (Gbit/s)
23120	200
11120	300
5840	400
3280	500
1760	600
1040	700
560	800
320	900
160	1000
80	1100

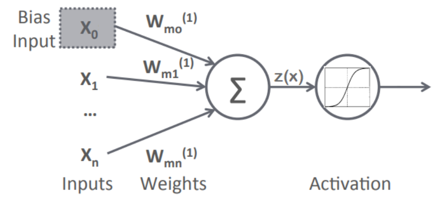
Knowing the length of each optical channel (determined in the routing process) it is possible to approximate their capacity through Table I. For example, if the length of a given optical channel, $l(\pi_k)$, is 400 km, then the largest capacity value in Table I that is able to accommodate the 400 km requirement corresponds to a capacity of 800 Gbit/s (that allows for transmission up to 560 km). Having the capacity of each channel and the number of blocked traffic demands (also determined in the routing process), Eq. (4) and Eq. (5) can then be used to determine, respectively, the average channel capacity and the total network capacity.

C. Theoretical Principles of Artificial Neural Networks

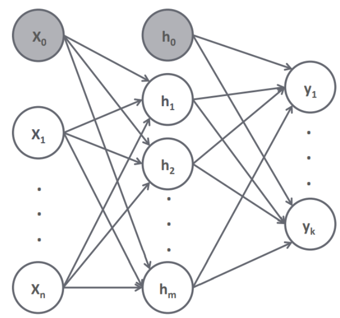
Artificial Neural Networks (ANNs), are supervised ML models where functional transformations are applied to the inputs with the goal of predicting outputs. These networks are constituted of neurons (or units), that apply a non-linear function to the combination of the inputs, and are separated into different types of layers: input, hidden, and output layers. Deep Neural Networks (DNNs) are neural networks with multiple hidden layers. These additional layers enable DNNs to model more complex, non-linear relationships in the data, enhancing their predictive capabilities. It was proved that DNNs with at least one hidden layer can be used to approximate any continuous function arbitrarily well, as long as there is a sufficient number of neurons [12].

ANNs can be used to solve two types of machine learning problems: regression problems (when the output variable is continuous) and classification problems (when the output variable is discreet). In this work, since the values being predicted are continuous, the developed DNN models are used to solve regression problems.

Figure 1 represents a neural network, with Fig. 1(a) detailing the operations of a generic neuron m, and Fig. 1(b) illustrating the overall structure of a neural network with one hidden layer.



(a) Detail of a generic neuron m.



(b) Neural Network.

Fig. 1: Neural network with one hidden layer and detail of neuron operations. (Adapted from [6])

Within each neuron, two key processes take place: the input activation (also known as pre-activation), and then the generation of the neuron's output through an activation function. The input activation, represented by $z(x)$, is described by:

$$z(x) = \sum_{i=1}^{n} w_{mi} x_i + x_0 \tag{6}$$

where x_i are the input variables, w_{mi} are each of the connection weights and x_0 is the bias term. The activation of the neuron is given by:

$$h_m(x) = g(z(x)) \tag{7}$$

where g is the activation function. Typical activation functions include: linear, sigmoid, hyperbolic tangent, rectified linear unit or softmax. In regression problems the Rectified Linear Unit (ReLU) ($g(z) = max(0, z)$) is commonly chosen as the activation function [13].

The expression that describes the pre-activation for a generic layer ℓ is:

$$z^{(\ell)}(x) = W^{(\ell)} h^{(\ell-1)}(x) + h_0^{(\ell)} \tag{8}$$

with $W^{(\ell)}$ being a matrix (of size $m_\ell \times m_{\ell-1}$, m_ℓ being the number of hidden units of hidden layer ℓ) containing the values of the connection weights, $h^{(\ell-1)}(x)$ the output of the previous layer, and $h_0^{(\ell)} \in \mathbb{R}^{m_\ell}$ the bias vector.

The activation of hidden layer ℓ is given by:

$$h^{(\ell)}(x) = g(z^{(\ell)}(x)) \tag{9}$$

In the case of the output activation, this function is typically the linear function ($o(z) = z$) in regression problems [6].

The training of neural networks consists of determining the value for all of the neuron's weights (the values of the W matrices) and bias values that minimize a given loss (or error) function with a given iterative method (optimizer algorithm). This process makes use of a large set of features (inputs) and labels (outputs), which correspond, in this work, to the physical topology parameters (features) and the total network capacity and average channel capacity, in the first DNN model, and the total network capacity and total fiber cost in kilometers, for the second DNN model (labels). To have this set of features and labels in a large enough quantity to train the DNN models it is necessary to generate a large set of random networks (as reference networks, real-world networks often used in the context of scientific investigation, are not sufficient in number) and then apply, to each network, the routing algorithms from which the data can be obtained.

III. Generating Random Networks

Random networks are networks whose physical topology is generated according to a given probability distribution, meaning that the edges, and in some models the node spatial positions as well, are random variables. These networks are represented by random graphs, which is a type of graph that was first studied by Paul Erdós and Alfred Rényi, who proposed a model for generating random networks, the Erdós-Rényi model [14]. The Erdós-Rényi model is the simplest way of generating truly random networks, but, if the goal is to generate networks similar to real-world optical backbone networks, total randomness is not necessarily the best option, as real topologies have certain characteristics that are not taken into account in this model.

Another widely used generative graph model is the Waxman model. This model considers the physical distance between nodes in the probability of node attachment, being thus designated as a geometric model [15]. This contrasts with the Erdós-Rényi model, which is a non-geometric model. In the Waxman model the nodes are placed uniformly at random in a two dimensional plane and the links are formed with a higher probability between nodes that are closer together. The probability that a node i establishes a link to node j is described by:

$$P(i,j) = \beta \exp \frac{-d(i,j)}{L_w \alpha} \tag{10}$$

where $d(i,j)$ is the Euclidean distance between nodes i and j, L_w is the maximum distance between any two nodes, and α and β are parameters in the range of 0 to 1. Increasing β leads to an overall larger probability of links between any two nodes, while increasing α leads to a larger ratio of larger links to shorter links.

Despite being better suited than the Erdós-Rényi model, the Waxman model still has a few drawbacks that make the generated topologies differ from their real-world counterparts [15]: (a) there is no guarantee that the generated topologies are connected; (b) there is no guarantee that the generated topologies are resilient to single link failures (equivalent to having $\lambda(G) \geq 2$, meaning that between any pair of nodes there is always an alternative path in case there is a link failure); (c) connections between distant nodes are too common (even with smaller values of α and β); and (d) the network's average node degree tends to excessive values when the number of nodes grows larger (which leads to networks with too many connections).

The model described in [15], which makes use of various principles of the Waxman model (including the disposition of the nodes in a 2D plane and the Waxman link probability, described in Eq. (10)) was designed specifically to obtain networks with topologies that reproduce real-world optical backbone networks, making alterations to the Waxman model to account for its failings. In the implementation made in this thesis, based on the model of [15], the author made a few additional alterations with the goal of assuring that the generated topologies more closely resemble their real-world counterparts, improving the overall functioning of the model given the specificities of this work, and also when the described implementation was lacking in detail. This implementation made use of the *Python* library *NetworkX* [16], with the networks being *NetworkX* graph objects. This allows for a simpler manipulation of the networks (easily adding edges and giving them weights) as well as determining relevant network parameters (like the average node degree) directly through this library's functions. Nevertheless, despite the alterations made by the author, the main functioning of the implemented model can still be described by Fig. 2, which is a flow diagram from [15].

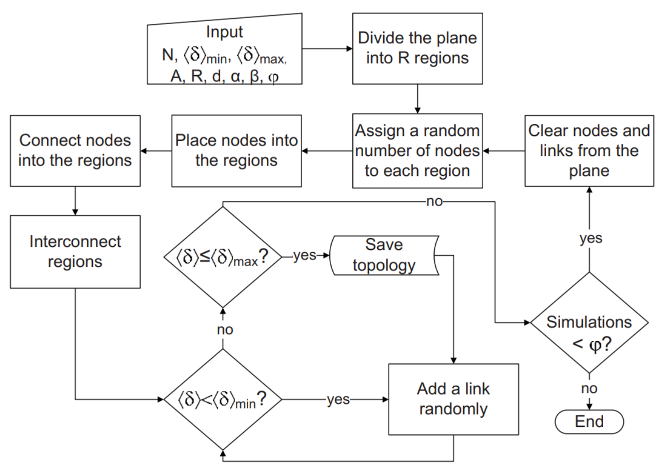


Fig. 2: Flow diagram of the generative model. (From [15])

As described in Fig. 2, this model has the following inputs: the number of nodes, N; minimum and maximum average node degrees ($\langle\delta_{min}\rangle$ and $\langle\delta_{max}\rangle$); the area of the plane A, specified through the side length L (the plane is a square, so $A = L^2$); the number of regions, R; the minimum distance between nodes, d; the α and β parameters of the Waxman link probability (in Eq. (10)); and the number of simulations (φ).

The model has the following general functioning: (1) dividing the plane into R regions; (2) placing a random number of nodes inside each region; (3) adding links between the nodes inside each region; (4) adding links between nodes of different regions; (5) adding links with Waxman probability until $\langle \delta_{min} \rangle$ is reached; and (6) adding links with Waxman probability until $\langle \delta_{max} \rangle$ is reached, saving new networks as the links are added. Every time a link is established a weight is assigned to it, corresponding to the distance between the nodes in the plane.

After step 4, the network (defined as a graph *G*) should have a connected topology that is survivable to single link failures. For that reason, the connections established between the nodes, either nodes inside the same region or nodes of different regions, needs to be done according to specific rules.

For the intra-region connections (step 3): (a) when there are two nodes in a region, they are directly connected; and (b) if there are three nodes or more, they are connected as a cycle according to their relative angular positions with respect to the centroid of their locations (the nodes are sorted based on the angles they make with the centroid, and each node is then connected to its next and previous node in this sorted list, forming a cycle).

The inter-region connections (step 4) are established by creating an auxiliary graph G_2 where each node represents a region in the original graph *G*, and the weight of an edge between two nodes is the distance between the corresponding regions. A Minimum Spanning Tree (MST) algorithm is applied to G_2 to connect all the nodes (regions) with the minimum total edge weight. An MST is a subset of the edges of a graph that connects all the nodes together, without any cycles and with the minimum possible total edge weight. Using an MST in this context ensures that all regions are connected while minimizing the total distance between them, ensuring that the closest regions are the ones where the connections are established. The edges of the MST are then used to connect the regions in *G*, with each edge in the MST corresponding to two links between each of the considered regions in *G*. For each pair of regions connected by an edge in the MST, the closest pair of nodes, one from each region, are connected. If a region has more than one node, a different node from the region is used for the second link. This ensures that there are at least two link-disjoint paths between any two regions, enhancing the resilience of the network.

The way the intra and inter-region connections are made is a key difference as to what was proposed in [15]. In [15], the connections in regions with more than three nodes, as well as the inter-region connections, are made following the Waxman link probability. But, during the implementation, it was seen that making connections in this manner had the tendency of establishing links between very distant nodes (even after adjusting the α and β parameters) and adding too many links inside the regions (given the random nature of the Waxman connections and the fact that enough connections need to be added until the minimum node degree of 2 is reached). Making the connections as described above proved to be a better solution that guarantees the minimum node degree of 2 to all nodes, while also providing a good structure over which more connections can be added, resulting in realistic topologies. The remaining links (added in steps 5 and 6), are established with the Waxman link probability (the same way as described in [15]).

It was shown that the networks generated with this model have topologies akin to real-world optical backbone networks, sharing with the real networks the following essential characteristics [15]: (a) the networks are connected; (b) the networks are resilient to single link failures ($\lambda(G) \geq 2$); (c) their node degree distribution follows the Poisson distribution; and (d) their link length distribution follows the Waxman link probability. This makes the implemented generative graph model adequate in the context of this thesis.

IV. Routing, Wavelength Assignment and Fiber Assignment Algorithms

In order to determine, from each individual network, the data needed to train the DNN models, two routing algorithms were implemented: a constrained routing algorithm, where the limitation in the number of optical channels per fiber in WDM transmission is considered and there is the possibility of traffic demands being blocked; and an unconstrained routing algorithm with fiber assignment, where the limitation in the number of optical channels per fiber is overcome through the addition of optical fibers to the network's links, which corresponds to an SDM solution.

During the routing process, each path $\pi_{s,d}$ is physically established with an optical channel and so it is also during this process that the wavelength for each optical channel is attributed. This is called wavelength assignment. For that reason, the implemented routing algorithm is a heuristic algorithm used to solve the Routing and Wavelength Assignment (RWA) problem, with the main objective being, for each optical channel k (corresponding to each traffic demand), the definition of a given path π_k and wavelength λ_k. Besides the RWA solution (through which the network capacity, average channel capacity and fiber cost are determined) this algorithm is also used to determine the parameters used as the DNN's inputs.

A. Constrained Routing Algorithm

The constrained routing algorithm has the following inputs: a file containing a list of *NetworkX* Graphs (the networks to be analyzed), the value of the maximum number of optical channels per link (link capacity), and the sorting strategy used to define the order the traffic demands will be routed (shortest-first, longest-first or largest-first). Optionally, the user can input a traffic matrix. If it is not specified, a matrix for a full-mesh logical topology (in Eq. (1)) is assumed by default.

In this algorithm each loaded network, *G*, is considered at a time in a loop and the following steps occur: (1) the determination of the shortest-paths between all pairs of nodes with Dijkstra's algorithm; (2) the ordering of the traffic demands according to the specified sorting strategy; (3) the routing of the traffic demands following the specified order, including the determination of the paths π_k and the wavelengths λ_k; (4) the calculation of the average channel capacity and total network capacity according to Eq. (4) and Eq. (5), with the capacity of each optical channel being determined through the optical reach, as explained in Section II-B; and (5) the determination of the network parameters related to the physical topology.

In the routing process (step 3) each traffic demand is considered individually in a loop and the following occurs: (i) if, for that traffic demand, there are various shortest paths with the same length, the one that minimizes the load in the most loaded link is chosen; (ii) a wavelength is assigned to that optical channel with the first-fit strategy [17] (being represented as an integer between 1 and the link capacity), subject to wavelength continuity (all the links in the path of an optical channel must use the same wavelength) and, if multiple optical channels pass through the same link, each must have a different wavelength (to avoid interference); (iii) if it is not possible to assign a wavelength given these restrictions, then that traffic demand is blocked. If it is possible then the link loads are updated and a π_k and λ_k are associated to that traffic demand; (iv) if a given link has reached residual capacity zero, that link is removed from the network and new shortest paths are calculated so that, from that point on, the routing of the following traffic demands will be done through the new paths; (v) if it is not possible to find a path for a given traffic demand (as the links become congested), that traffic demand is blocked.

At the end of this process the RWA solution is found, being represented in: (a) the *load_matrix*, a matrix of size $N \times N$ that contains in position (i, j) the total load in the link between nodes i and j; (b) the *path_matrix* ($N \times N$) which has, in position (i, j), the path chosen to route the traffic demand between node i and node j (represented as a sequence of nodes); (c) the *distance_matrix* ($N \times N$) that contains, in position (i, j), the length of that given path between nodes i and j; (d) the variable *blocked_traffic* that represents the amount of blocked traffic demands; (e) the list *blocked_paths* that contains the paths corresponding to the traffic demands which were blocked; (f) the matrix *path_wavelength_matrix* ($N \times N$) that has, in position (i, j), the wavelength attributed to the optical channel between nodes i and j; and (g) the matrix *link_wavelength_matrix* that has, in position (i, j), a list of all the wavelengths present in the link between nodes i and j. Furthermore, besides the average channel capacity and total network capacity, the following parameters, to be used as inputs of the DNN models, are also determined: number of nodes, number of links, minimum, maximum and average link length, variance of link length, minimum, maximum and average node degree, variance of node degree, network diameter, and algebraic connectivity.

B. Unconstrained Routing with Fiber Assignment Algorithm

The second routing algorithm, meant to implement the SDM solution, has the goal of completely eliminating the traffic demand blockages through the deployment of additional optical fibers to the network's links. The addition of enough fibers to the network allows for the reduction and eventual elimination of the blocking, as each fiber added to a link increases that link's capacity. The process of determining which links should receive additional fibers, and how many fibers to add, is referred to as fiber assignment. The determination of a fiber assignment solution that results in the elimination of blocking while also minimizing the amount of deployed fiber can be, however, a computationally complex procedure [18]. For that reason, the implemented solution has the goal of eliminating the traffic demand blockages while also prioritizing reduced computation times, which is essential when analyzing thousands of networks.

This routing algorithm is identical to the previous algorithm (described in Section IV-A) in almost all aspects, but now it is assumed that there is no limit to the number of optical channels per link (unconstrained routing). Another difference to the previous algorithm are the outputs. Since now there is no blocking, the outputs *blocked_traffic* and *blocked_paths* are no longer present. Instead there are now new outputs: (a) *fiber_link_matrix*, a matrix of size $N \times N$, where the element in position (i, j) has the number of fibers used in the link between nodes i and j; and (b) *total_cost*, the total number of kilometers of fiber in the network. This last output is given by:

$$\Lambda_{\text{net}} = \sum_{i,j} l_{i,j} \times nf_{i,j} \tag{11}$$

where $l_{i,j}$ is the length of the link between the nodes i and j (in km), and $nf_{i,j}$ is the number of optical fibers in the link between nodes i and j, as represented in the *fiber_link_matrix*. In this matrix the number of fibers is presented by direction, that is, the number of fibers in the link from i to j separately from the number of fibers in the link from j to i (each in the respective positions of the matrix). Given the way the routing is done, this matrix (as well as the *link_wavelength_matrix*) is symmetric when the traffic matrix is symmetric.

The fiber assignment process occurs after the RWA solution is determined (in step 3 of the previous algorithm). This process is done knowing the wavelengths in each link (based on the *link_wavelength_matrix*) and the maximum number of optical channels than can exist in a fiber, *max_wavelengths* (defined by the channel spacing and the bandwidth of the WDM signal, similarly to the link capacity in the previous algorithm, as mentioned in Section I). This algorithm works as follows:

- if there is no traffic in a given link but the link does exist in the network's physical topology, then $nf_{i,j} = 1$;
- if there is traffic in a link, $nf_{i,j}$ will be the maximum number of repeated wavelengths in the link between nodes i and j.

This last step is what guarantees the placement of the correct number of fibers in each link. When the routing is made in an unconstrained manner the number of wavelengths in a link will not have a limit. Since there is not, however, an unlimited number of wavelengths to assign to the optical channels (that limit being *max_wavelengths*), the list of wavelengths in a link (represented in the respective position of the *link_wavelength_matrix*), which is a list of integers without repetitions ranging from 1 to $+\infty$ (resulting from the first-fit strategy done in the RWA process in step 3), has in fact repeated wavelengths, which exist in the same link, but in different fibers (as two same wavelength signals cannot exist in the same fiber).

To determine the number of repeated wavelengths, the list of wavelengths in the link needs to be normalized to the range between 1 and *max_wavelengths*. This is achieved through the modulo operation (the remainder of a division) between that wavelength's number and *max_wavelengths*. For example, if the list of wavelengths in a given link is [1,2,3,51,52,101] and *max_wavelengths* is 50, the normalized list would then be [1,2,3,1,2,1], meaning that the wavelengths 1, 51 and 101 are in fact the same wavelength but in different fibers (as well as wavelengths 2 and 52). The maximum number of repeated wavelengths is then determined based on the normalized wavelengths list. In this case, wavelength 1 occurs 3 times, wavelength 2 occurs twice, and wavelength 3 only once. The maximum number of repeated wavelengths is thus 3, and so the number of fibers assigned to that link will be 3, as that is the number of fibers needed to accommodate all the wavelengths. This strategy guarantees that each "repeated" wavelength has its own fiber, meaning that there are no optical channels with the same wavelength on the same fiber.

As previously mentioned, this solution is one that prioritizes fast computation times over the minimization of the number of added fibers. In the development of this thesis, a solution that attempts to improve the efficiency of the fiber assignment process was also developed. While this alternative method did show improvements over the previously described method in terms of reducing the number of fibers introduced, the computation times ended up being very high, taking, for example, almost 28 hours to compute the routing solution in a randomly generated network with 60 nodes considering a full-mesh logical topology. Given these high computation times, the method described in this section was chosen instead, as it ends up providing the best balance between efficiency in number of fibers added and fast computation.

V. Results of the DNN models

The two regression DNN models consider the same 12 input parameters (features): number of nodes, number of links, minimum, maximum and average link length, variance of link length, minimum, maximum and average node degree, variance of node degree, network diameter, and algebraic connectivity. The outputs of the models (labels) are: in the first DNN model, the total network capacity and average channel capacity, subject to the restriction imposed by the limited number of optical channels in the network's links; and in the second DNN model, the total network capacity and the total cost in kilometers of fiber for the case where that limitation is overcome through the addition of optical fibers to the network (SDM solution). The data used to train and test these models was obtained from the application of the respective routing strategies described in Section IV, to sets of random networks generated with the model described in Section III.

A. First DNN Model

The first DNN model (with the outputs total network capacity and average channel capacity determined in the context of constrained routing) was trained with the data of 15245 random networks. These networks were generated considering 2D planes with side lengths varying from 1000 km to 5000 km in increments of 1000 km, number of regions in the plane set to 4, number of nodes varying from 5 to 55, and an average node degree varying from 2 to around 5. The Waxman parameters chosen were $\alpha = \beta = 0.4$. The constrained routing solution was done considering a full-mesh logical topology (in Eq. (1)), the shortest-first sorting strategy, and transmission at 64 Gbaud in the C-band, resulting in a limit of 75 optical channels per link (this link capacity results from fixed bandwidth of 4800 GHz in C-band transmission and a channel spacing of 64 GHz, equal to the symbol rate, as assumed in Section II-B). The computation of this routing solution took around 3 hours 42 minutes for the entire set of networks. As a result of the hyperparameter tuning process (through a Grid Search method [19]) a structure with 2 hidden layers with 10 hidden units each was chosen, as it was found to lead to the best performance on the validation set, while also being a structure with few trainable parameters (262 weight and bias terms).

To evaluate the performance of this DNN model, an additional test set with 750 random networks was generated under the same conditions as the training set. For this entire set, the routing process took approximately 7 minutes and 10 seconds. In contrast, the DNN model was able to make the predictions for the entire set of 750 networks in just 22 milliseconds.

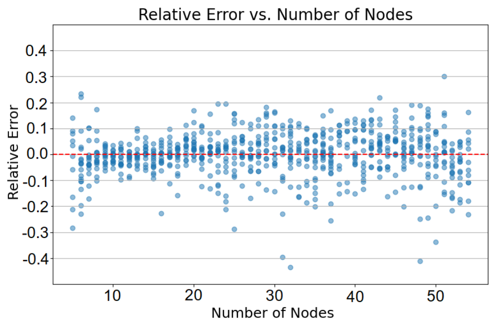
Figure 3 shows the scatter plot of the relative errors against the number of nodes for both outputs. Each dot represents the relative error between the value determined from the routing solution (y_i) and the prediction made with the DNN model ($\hat{y}_i$) for an individual network. The relative error for each data point is calculated as:

$$\text{Relative Error} = \frac{y_i - \hat{y}_i}{y_i} \tag{12}$$

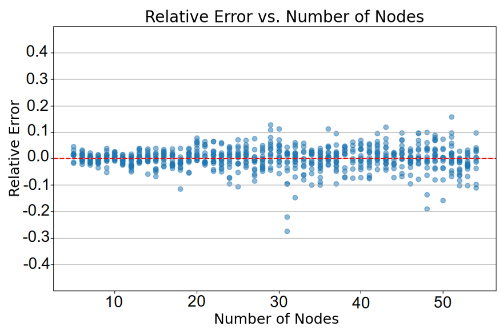
As depicted in Fig. 3(a), for the total network capacity output, the majority of relative errors are under 10%. Specifically, 79.23% of the examples exhibit a relative error below 10%, and 90.81% of examples have a relative error below 15%. The average channel capacity output, as illustrated in Fig. 3(b), generally presents even smaller relative errors: 83.36% of examples have a relative error below 5%, and a significant 97.60% below 10%.

These results show that, despite providing a good performance in both outputs, this is not uniform across the two outputs, as the model exhibits a superior performance for the average channel capacity, having smaller relative errors. This disparity in performance between the two outputs could be indicative of the model's ability to learn and generalize from the training data, with the superior performance on the average channel capacity suggesting that the features present in the training data could be more predictive for this output, allowing the DNN model to learn more accurately.

When making predictions on reference networks, it was shown that the DNN model is capable of predicting both outputs with low relative errors: under 5% in almost all predictions of both outputs, for the group of 5 reference networks considered (CESNET, COST239, DTAG, NSFNET and UBN). This indicates that the model is able to make accurate predictions in real-world optical transport networks,



(a) Total Network Capacity



(b) Average Channel Capacity

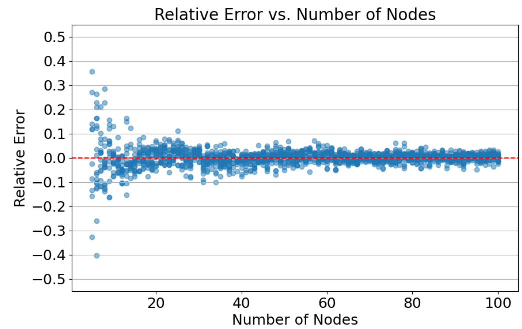
Fig. 3: Relative errors against the number of nodes for both outputs of the first DNN model

which further confirms that the topologies of the generated networks are close to their real-world counterparts.

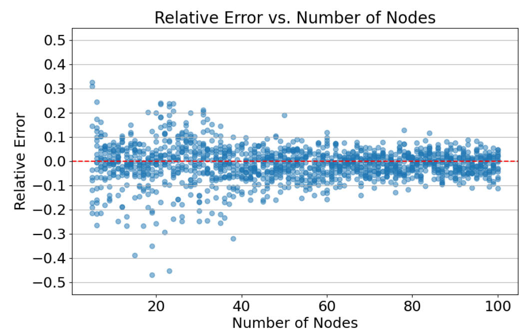
B. *Second DNN Model*

To train the second DNN model (with the outputs total network capacity and the total cost in kilometers of fiber in the case where the SDM solution is applied) a set with 8480 networks was generated under the same conditions as the set used for the first DNN model, but now with the number of nodes varying from 5 to 100. As in the previous scenario, a logical full-mesh topology was considered, the shortest-first sorting strategy was used, and transmission was done at 64 Gbaud on the C-band (and so a limit of 75 wavelengths per fiber was set). The larger range in number of nodes allows for this model to be used in a wider range of networks. However, the trade-off is that the number of examples used to train the model must be reduced due to the increased computational time associated to the routing process in larger networks. The computation time for the unconstrained routing with fiber assignment was 7 hours 51 minutes for the entire set of networks. As a result from the optimization of the hyperparameters, a DNN with 1 hidden layer with 50 hidden units was chosen.

To test the performance of this DNN model, a data set relative to 1440 networks (generated under the same conditions as the training set) was used. The unconstrained routing with fiber assignment took around 1 hour and 16 minutes for the entire set of networks, while the prediction time with the DNN model was just 11 milliseconds. Figure 4 plots the relative errors against the number of nodes for this test set.



(a) Total Network Capacity



(b) Total Fiber Cost

Fig. 4: Relative errors against the number of nodes for both outputs of the second DNN model (N in [5,100])

From Figure 4 it was concluded that, for the total network capacity (Fig. 4(a)), 89.45% of the examples have a relative error below 5%, and 96.67% of the examples have a relative error below 10%. In the case of the total fiber cost (Fig. 4(b)) 87.02% of the examples have a relative error below 10%, and 94.24% of the examples have a relative error below 15%.

It can also be seen that the model tends to have better performance for networks with a higher number of nodes, having a more irregular performance on networks with very few nodes. A possible explanation for this behavior is that smaller networks might exhibit more variability in their features and in the relationships between features and labels. This variability could make them more challenging for the model to predict accurately. On the other hand, larger networks could be more homogeneous, exhibiting more consistent patterns that the model can learn and predict more effectively.

The evaluation on the same 5 reference networks reveals that the model delivers accurate predictions in the majority of cases. However, there are instances where higher relative errors have been observed, with three instances exceeding a 10% error rate. These discrepancies can likely be attributed to specific characteristics of those networks, which the model may not have effectively learned during its training phase. It's worth noting that, despite these occasional prediction inaccuracies, the model significantly reduces computation times when compared to the heuristic routing method.

VI. Conclusions

The work developed in this thesis resulted in the successful development of two DNN models. The first model was designed to predict the total network capacity and average channel capacity of transparent optical backbone networks, considering a constrained number of optical channels per link due to the bandwidth limitations of fiber transmission. The second model was developed to estimate the unconstrained total network capacity and the total cost in kilometers of fiber deployed, under a scenario where an SDM solution is implemented to overcome said limitations, effectively increasing the network's bandwidth.

The training of the DNN models was done through the data obtained from sets of randomly generated networks, given the need to use a large amount of data in the training process. The implemented generative graph model was shown to produce random networks with topologies similar to real-world optical backbone networks, having connected topologies with a node degree distribution that follows the Poisson distribution, a link length distribution that follows the Waxman link probability, and an edge connectivity of at least two (being resilient to single link failures).

The routing algorithms, which allow for the specification of a traffic matrix, a sorting order and the link capacity, determine the RWA solution as well as the parameters to be used as the inputs of the DNN models. The optical channel capacity (which is used to determine the average channel capacity and the total network capacity) was determined through the optical reach, which accounts for the physical aspects of optical transmission. By knowing the length of the optical channels (a result of the routing process) their capacity is approximated through the optical reach and Shannon capacity relation presented in Table I. The SDM solution was implemented through the addition of optical fibers to the links "as needed", effectively eliminating the blocking of traffic. This was achieved through the execution of the unconstrained routing followed by an algorithm that attributes additional fibers to the links based on the wavelengths present. This fiber assignment solution was built with the requirement of having short computation times, given the need to use it in thousands of graphs. An alternative algorithm, that results in less fibers being added to the network, was also developed, but due to its long computation times it was not applied in this work.

The DNN models were shown to provide overall accurate results on both random networks as well as reference networks, being able to predict in just a few milliseconds, even when considering sets with thousands of networks. This is a very significant advantage over the heuristic routing algorithm's computation times, which can take hours in larger sets of networks. Given the DNN models' overall good performance and quick prediction times, these models could prove to be useful tools in the design of optical backbone networks.

References

[1] P. J. Winzer and D. T. Neilson, "From scaling disparities to integrated parallelism: A decathlon for a decade," *Journal of Lightwave Technology*, vol. 35, no. 5, pp. 1099–1115, 2017.

[2] J. J. Pires, "Chapter 1 - Introduction," Optical Networks, Instituto Superior Técnico, 2023.

[3] N. Deng, L. Zong, H. Jiang, Y. Duan, and K. Zhang, "Challenges and Enabling Technologies for Multi-Band WDM Optical Networks," *Journal of Ligthwave Technology*, vol. 40, no. 11, pp. 3385–3394, 2022.

[4] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[5] A. Freitas and J. Pires, "Investigating the Impact of Topology and Physical Impairments on the Capacity of an Optical Backbone Network," *Photonics*, vol. 11, no. 4, 2024. [Online]. Available: https://www.mdpi.com/2304-6732/11/4/342

[6] 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 Communications Surveys and Tutorials*, vol. 21, no. 2, pp. 1383–1408, 2019.

[7] R. Gu, Z. Yang, and Y. Ji, "Machine learning for intelligent optical networks: A comprehensive survey," *Journal of Network and Computer Applications*, vol. 157, no. 102576, 2020.

[8] K. Higashimori, T. Inoue, T. Tanaka, F. Inuzuka, and T. Ohara, "Impact of physical topology features on performance of optical backbone networks," in *2022 International Conference on Optical Network Design and Modeling (ONDM)*, 2022, pp. 1–6.

[9] "NetworkX - algebaric_connectivity," Available at: https://networkx.org/documentation/stable/reference/generated/networkx.linalg.algebraicconnectivity.algebraic_connectivity.html, 2023, accessed on 10.05.2024.

[10] R.-J. Essiambre, G. Kramer, P. J. Winzer, G. J. Foschini, and B. Goebel, "Capacity limits of optical fiber networks," *Journal of Lightwave Technology*, vol. 28, no. 4, pp. 662–701, 2010.

[11] A. Souza, B. Correia, N. Costa, J. Pedro, and J. Pires, "Accurate and scalable quality of transmission estimation for wideband optical systems," in *2021 IEEE 26th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 2021, pp. 1–6.

[12] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0893608089900208

[13] S. Chugh, S. Ghosh, A. Gulistan, and B. Rahman, "Machine Learning Regression Approach to the Nanophotonic Waveguide Analyses," *Journal of Lightwave Technology*, vol. 37, no. 24, pp. 6080–6089, 2019.

[14] J. Bisht. (2022, 7) Erdos-Renyi Model for generating random graphs. Accessed on 08.05.2024. [Online]. Available: https://www.geeksforgeeks.org/erdos-renyl-model-generating-random-graphs/

[15] C. Pavan, R. M. Morais, J. R. F. da Rocha, and A. N. Pinto, "Generating Realistic Optical Transport Network Topologies," *Journal of Optical Communication Networks*, vol. 2, no. 1, pp. 80–90, 2010.

[16] A. A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring Network Structure, Dynamics, and Function using NetworkX," Proceedings of the 7th Python in Science Conference (SciPy 2008), 2008.

[17] H. Zang, J. P. Jue, B. Mukherjee *et al.*, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks," *Optical networks magazine*, vol. 1, no. 1, pp. 47–60, 2000.

[18] D. A. Moniz, "Design of spectral- and cost-efficient high-capacity optical transport networks," PhD thesis, Instituto Superior Técnico, 2022.

[19] J. Brownlee, "Hyperparameter optimization with random search and grid search," 2020, accessed on 15-05-2024. [Online]. Available: https://machinelearningmastery.com/hyperparameter-optimization-with-random-search-and-grid-search/