ARCHIVIO ISTITUZIONALE
DELLA RICERCA

Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

2D Frequency-based topological optimization: efficient dataset creation for neural networks to aid in tuning simulation parameters

This is the final peer-reviewed author's accepted manuscript (postprint) of the following publication:

Published Version:
Bacciaglia, A., Ciccone, F., Ceruti, A., Peruzzini, M. (2026). 2D Frequency-based topological optimization: efficient dataset creation for neural networks to aid in tuning simulation parameters. INTERNATIONAL JOURNAL ON INTERACTIVE DESIGN AND MANUFACTURING, 20, 2017-2035 [10.1007/s12008-025-02349-9].

Availability:
This version is available at: https://hdl.handle.net/11585/1019532 since: 2026-04-07

Published:

DOI: http://doi.org/10.1007/s12008-025-02349-9

Terms of use:

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (https://cris.unibo.it/).
When citing, please refer to the published version.

(Article begins on next page)

10 October 2026

ACCEPTED MANUSCRIPT

2D Frequency-based topological optimization: efficient dataset creation for neural networks to aid in tuning simulation parameters

This Accepted Manuscript (AM) is a PDF file of the manuscript accepted for publication after peer review, when applicable, but does not reflect post-acceptance improvements, or any corrections. Use of this AM is subject to the publisher's embargo period and AM terms of use. Under no circumstances may this AM be shared or distributed under a Creative Commons or other form of open access license, nor may it be reformatted or enhanced, whether by the Author or third parties. By using this AM (for example, by accessing or downloading) you agree to abide by Springer Nature's terms of use for AM versions of subscription articles: https://www.springernature.com/gp/open-research/policies/accepted-manuscript-terms

The Version of Record (VOR) of this article, as published and maintained by the publisher, is available online at: https://doi.org/10.1007/s12008-025-02349-9. The VOR is the version of the article after copy-editing and typesetting, and connected to open research data, open protocols, and open code where available. Any supplementary information can be found on the journal website, connected to the VOR.

For research integrity purposes it is best practice to cite the published Version of Record (VOR), where available (for example, see ICMJE's guidelines on overlapping publications). Where users do not have access to the VOR, any citation must clearly indicate that the reference is to an Accepted Manuscript (AM) version.

Accepted manuscript

ACCEPTED MANUSCRIPT

2D Frequency-based topological optimization: Efficient dataset creation for Neural Networks to aid in tuning simulation parameters

1. Antonio Bacciaglia*, PhD. Department of Industrial Engineering (DIN), University of Bologna, Bologna, Italy; Email: antonio.bacciaglia2@unibo.it; Orcid: 0000-0002-4384-6300;

2. Francesco Ciccone. Department of Industrial Engineering (DIN), University of Bologna, Bologna, Italy; Email: francesco.ciccone2@unibo.it; Orcid: 0000-0002-0983-3266;

3. Alessandro Ceruti, Prof. Department of Industrial Engineering (DIN), University of Bologna, Bologna, Italy; Email: alessandro.ceruti@unibo.it; Orcid: 0000-0001-7947-3932.

4. Margherita Peruzzini, Prof. Department of Industrial Engineering (DIN), University of Bologna, Bologna, Italy; Email: margherita.peruzzini@unibo.it; Orcid: 0000-0003-2260-0392;

*: corresponding author: Antonio Bacciaglia

2D Frequency-based topological optimization: Efficient dataset creation for Neural Networks to aid in tuning simulation parameters

Abstract

Structural topology optimization is a key method for designing lightweight and efficient components, particularly in additive manufacturing. In aerospace and automotive engineering applications, where components bear dynamic loads, it is crucial to account for natural frequencies and weight reduction. Traditional optimization methods often require labour-intensive manual setup, especially for tuning input parameters that influence the algorithm's performance and convergence. Machine learning offers an alternative, streamlining this process and reducing reliance on trial-and-error adjustments.

This study introduces a machine learning-based framework to optimize input parameters—such as evolutionary rates and mesh dimensions—while ensuring the creation of lightweight, optimized structures with consistent natural frequencies. The framework uses a neural network trained on a dataset built to describe several configurations of a cantilever beam case study. The primary focus is determining the optimal dataset size for effective neural network training. The analysis aims to minimize dataset size, reducing the time and effort required for data preparation while avoiding overfitting in the neural network. A dissimilarity metric based on problem metadata is used to guide parameter tuning. After training, the artificial neural network is incorporated into the Bayesian Optimization framework as a surrogate model, enabling the new model to rapidly estimate outcomes without running full simulations. This approach substitutes traditional topology optimization algorithms, accelerates time-to-market, and is well-suited for applications with only minor variations in design conditions.

Keywords: Topology Optimization, Frequency Optimization, Bidirectional Evolutionary Structural Optimization (BESO), Artificial Neural Network, Dataset Analysis, Design for Additive Manufacturing.

Accepted manuscript

ACCEPTED MANUSCRIPT

Paper type: Research paper – Special Issue

List of acronyms

TO	Topology Optimization	NN	Neural Network
CAD	Computer-Aided Design	BO	Bayesian Optimizer
NURBS	Non-Uniform Rational Basis Spline	ReLU	Rectified Linear Unit
FE	Finite Element	MSE	Mean Squared Error
SIMP	Solid Isotropic Material Penalization	NRMSE	Normalized Root Mean Squared Error
ESO	Evolutionary Structural Optimization	MRE	Mean Root Error
BESO	Bi-directional ESO	GPR	Gaussian Process Regression
AM	Additive Manufacturing	APE	Absolute Percentage Error
ML	Machine Learning	PCA	Principal Component Analysis

1. Introduction

Topology Optimization (TO) methods are widely utilized in industrial engineering to design lightweight and efficient components [1]. These methodologies enable optimized structural designs by considering parameters such as applied loads, available design space, and material properties and avoid following pre-designed shapes. Often, the resulting designs aim to achieve minimal compliance for a given design volume fraction. However, accounting for natural frequencies is crucial for components subjected to dynamic loads, particularly in aerospace [2] and automotive applications [3]. Consistency between the optimized configuration and the actual reassembled geometry in a Computer-Aided Design (CAD) environment is a key issue of TO. The challenge has been handled in recent works by merging the Non-Uniform Rational Basis Spline (NURBS) curves and surfaces with TO algorithms [4, 5]. This approach contrasts with the Finite Element (FE) model, which is employed in the optimization task and provides an "element-wise" description of the resultant topology. However, the latter enables the handling of complicated problems that are characteristic of the industrial environment; thus, the prospect of approaching TO studies using the traditional FE method remains appealing.

Although stiffness optimization has been extensively studied in the literature [6, 7], research on frequency optimization remains relatively limited [8, 9]. This topic is critical for industrial applications involving cyclic load environments, such as aerospace and automotive [10]. In these fields, factors like resonance, power spectral density, and fatigue are so significant that components often operate under loads far below the yielding stress, which would typically be the design constraint. Resonance, caused by external excitation frequencies matching or nearing a structure's natural frequency, can lead to significant vibrations, deformations or fractures [11]. It is essential to design structures with appropriate natural frequencies that maintain a safe margin from operating frequencies to avoid such failures.

Previous research has shown that the density-based TO algorithms [12] employing the Solid Isotropic Material with Penalization (SIMP) scheme [13] for minimizing compliance are inadequate for frequency optimization. This limitation arises from developing artificial localized vibration modes in areas of low mass [14] and the mode-switching phenomenon [15, 16]. Mode switching happens when, during the optimization process, the most critical mode (i.e., that related to the first eigenvalue) can change during the optimization. Conversely, studies on Evolutionary Structural Optimization (ESO) and Bi-directional ESO (BESO) methods [17] have demonstrated their ability to address localized vibration issues in frequency optimization. However, challenges persist with these approaches, particularly in achieving robust solutions for frequency-based design.

To address these shortcomings, Huang et al. proposed a modified SIMP-BESO approach that maintains a consistent mass-to-stiffness ratio for solid and void elements [18]. This modification reduces the

Accepted manuscript

ACCEPTED MANUSCRIPT

occurrence of artificial localized modes during frequency optimization. Their work also incorporates a sensitivity analysis specific to the target frequency, identifying the relative importance of elemental contributions. These sensitivity numbers are then integrated into the BESO method. By employing a straightforward BESO update scheme alongside rigorous optimality criteria, their method identifies optimal designs that satisfy prescribed volume or weight constraints, enhancing the effectiveness of frequency-based topology optimization.

The complex and intricate geometries produced by TO analyses often present significant challenges for traditional manufacturing methods, such as chip removal, when creating 3D models [19]. In contrast, these geometrically complex designs are well-suited to Additive Manufacturing (AM), which builds structures layer by layer through material deposition [20]. AM is celebrated for its design flexibility, enabling the production of complex biomimetic shapes [21] with exceptional strength-to-weight ratios [22]. Additionally, it offers a shorter transition from design to production and allows for integrating multiple components into a single part, reducing the need for bolted connections or weldings and simplifying assembly processes [23].

TO is highly effective for creating optimized designs, particularly when paired with AM [24]. Despite its proven benefits, TO methods frequently demand considerable manual intervention to configure and fine-tune input parameters critical to algorithm performance and convergence [25]. If these parameters are not optimally set, TO can occasionally produce unrealistic or impractical results [26].

Recent advances in Machine Learning (ML) have shown great promise across various fields, including applications in AM [27, 28]. ML methods can potentially replace the costly trial-and-error process of manually tuning simulation parameters [29]. While previous studies have used ML for structural optimization with a limited set of parameters [30], no research has addressed ML for input parameter tuning in BESO frequency optimization.

Several recent studies have started to use machine- learning surrogates to automate the tuning of numerical parameters in density or level set- based topology optimization [31–34]. None of these contributions, however, couples the tuning task with an explicit, quantitative study of dataset orthogonality (e.g., correlation heat- maps, PCA spectra, condition number) and shows how those quality metrics propagate to surrogate accuracy and, in turn, to the efficiency of a Bayesian optimizer.

To the authors' best knowledge, there is a technological gap in using Neural Networks (NNs) for tuning input parameters in the topology optimization of 2D structures. This gap is particularly evident in approaches emphasizing dataset quality analysis while aiming to reduce simulation times effectively.

This study introduces a framework that leverages ML to automate the tuning of input parameters such as evolutionary rate and mesh dimension for frequency-based topology optimization. The aim is to achieve lightweight, optimized designs with consistent natural frequencies. A NN is trained on a repository of previously solved problems using a cantilever 2D beam, where input parameters are randomly varied. Huang's algorithm [35] has been selected as a reference for building the aforementioned dataset for its consistency, accuracy, and ease of implementation. The dataset has been thoroughly analyzed for quality, including correlation, dataset asset quality, and condition metrics [36]. This analysis ensures the dataset's suitability for training while minimizing overfitting [37]. Additionally, the dataset size has been reduced to strike an efficient balance between computational cost and NN performance.

The NN models the relationship between input parameters and a simulation quality metric (the frequency ratio between optimized and initial designs). Once trained, it is integrated into a Bayesian optimization framework [38], replacing the traditional BESO method to accelerate the design process. This approach reduces the manual effort required to fine-tune simulation parameters while maintaining the natural frequency of the target structure. The methodology is particularly suited for industrial applications where

Accepted manuscript

ACCEPTED MANUSCRIPT

structural configurations remain fixed but minor boundary condition variations, such as changes in volume fraction, are expected. By enabling the generation of optimal designs with reduced computational time, this framework facilitates faster time-to-market for products.

The paper is organized as follows. Section 2 reviews the foundational optimization problem for frequency, introducing the frequency optimization model available in the literature proposed by Huang et al. This tool will be used to produce the dataset for the NN. The same section provides an overview of integrating the NN and Bayesian Optimizer (BO) developed in this study. Section 3 presents numerical results demonstrating the performance of the proposed machine learning method as a substitute for the BESO approach, along with an evaluation of the dataset quality used to train the NN. Section 4 offers a discussion of the findings detailed in the previous section. Finally, Section 5 concludes the paper with key insights and recommendations for future enhancements.

2. Methodology

The proposed methodology is applied to a generalized 2D cantilever beam measuring 8x1 units in length and composed of a hypothetical material with a unit density (1 kg/m^3), a Poisson's ratio of 0.3, and an elastic modulus of 10^7 Pa. Users can customize the material properties as needed without compromising the accuracy of the approach. The beam is further modified by adding a concentrated mass m (in kilograms) on the lower edge of the beam at a specific point, defined as a percentage of the beam's total length (x_m). A schematic representation of the beam is provided in Fig. 1.

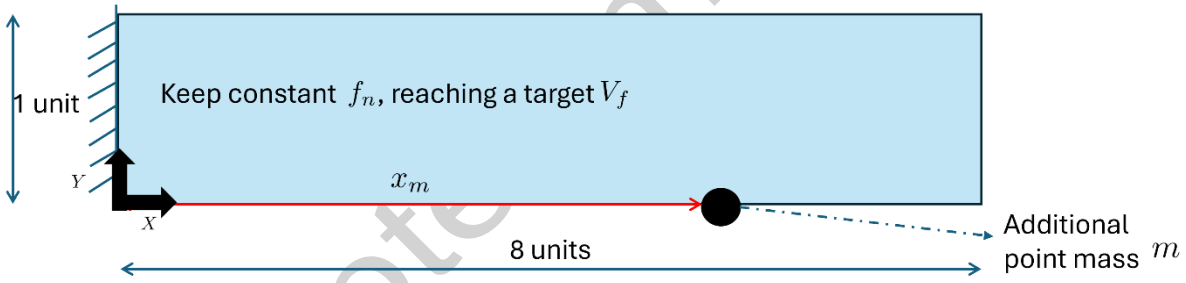

Fig. 1 Cantilever beam topology optimization problem with a concentrated point mass

The constraint function g for the problem is expressed in Eq. 1, where V represents the volume of the solid material, V_0 denotes the total volume of the design space and V_f specifies the imposed volume fraction limit. Even if this study deals with 2D geometries, the wording' volume fraction' is used to be consistent with some reference studies which deal with 2D problems available in the literature [35] (in 2D studies, the volume fraction should be intended as volume fraction per length).

$$g = \left(\frac{V}{V_0}\right) - V_f \tag{1}$$

The objective of the optimization is to achieve a lighter structure by meeting the target volume fraction V_f while maintaining the first natural frequency f_n of the overall structure as constant as possible.

To achieve this goal, a skilled user may apply the modified SIMP-BESO method by Huang, described in Section 2.1.

2.1 Evolutionary topological optimization for natural frequencies

ACCEPTED MANUSCRIPT

The dynamic behaviour of a continuous structure is mathematically represented through the principal eigenvalue problem, as defined in Eq. 2:

$$\left(\boldsymbol{K} - {f_i}^2 \boldsymbol{M}\right)\boldsymbol{u}_i = 0 \tag{2}$$

In this formulation:

- $\boldsymbol{K}$ is the global stiffness matrix,
- $\boldsymbol{M}$ is the global mass matrix,
- f_i represents the i-th natural frequency,
- $\boldsymbol{u}_i$ is the corresponding eigenvector to f_i.

The relationship between the i-th natural frequency and its associated eigenvector is expressed by the Rayleigh quotient (Eq. 3).

$$f_i^2 = \frac{\boldsymbol{u}_i^T \boldsymbol{K} \boldsymbol{u}_i}{\boldsymbol{u}_i^T \boldsymbol{M} \boldsymbol{u}_i} \tag{3}$$

For the solid-void design, the problem formulation follows in Eq. 4:

$$\begin{gathered} Keep\ constant\ f_n \\ Subject\ to: V_f = \sum_{i=1}^{N} V_i x_i \\ x_i = x_{min}\ or\ 1 \end{gathered} \tag{4}$$

A binary value is assigned to the design variable x_i, indicating the density of the i-th element, with V_i the volume of the single element. A void element is assigned a small value ($x_{min} = 10^{-6}$) to avoid matrix singularities, while a bulk element can be recognized by a unitary value of x_i.

It is crucial to interpolate the material properties between x_{min} and 1 using a suitable material interpolation scheme to obtain gradient information for the design variable. The widely adopted power-law penalization approach in SIMP models is not directly applicable in frequency optimization due to the occurrence of artificial localized modes in regions with low material density. Thus, other alternative interpolation schemes are proposed in the literature [39]; among them, Huang proposes an alternative interpolation method defined in Eq. 5, which ensures that the present model asymptotically converges to the original SIMP model as x_{min} approaches zero. This methodology mitigates the challenges associated with low-density regions while preserving the effectiveness of material interpolation in TO.

$$\begin{cases} \rho(x_i) = x_i \rho^1 \\ E(x_i) = \left[\dfrac{x_{min} - x_{min}^p}{1 - x_{min}^p}\left(1 - x_i^p\right) + x_i^p\right] E^1 \quad for\ 0 < x_{min} \le x_i \le 1 \end{cases} \tag{5}$$

In Eq. 5, ρ^1 and E^1 express the density and Young's modulus of the solid material.

The elemental stiffness $\boldsymbol{K}_i^1$ and mass $\boldsymbol{M}_i^1$ matrices for solid elements are computed as derivatives from the global stiffness and mass matrices with respect to the design variable x_i. Using Eq. 2 and the definitions of

Accepted manuscript

ACCEPTED MANUSCRIPT

$\boldsymbol{K}_i^1$ and $\boldsymbol{M}_i^1$ and assuming the eigenvector $\boldsymbol{u}_i$ is normalized by $\boldsymbol{M}$, the sensitivity of the i-th frequency f_i can be formulated as shown in Eq. 6.

$$\frac{df_i}{dx_i} = \frac{1}{2f_i}\boldsymbol{u}_i^T\left(\frac{1-x_{min}}{1-x_{min}^p}px_i^{p-1}\boldsymbol{K}_i^1 - f_i^2\boldsymbol{M}_i^1\right)\boldsymbol{u}_i \tag{6}$$

In the original method by Huang et al., the sensitivity numbers (α_i) are used in evolutionary structural optimization to rank elements based on their sensitivity and explicitly defined, as illustrated in Eq. 7, if x_{min} tends to 0. Huang et al. suggest using p=3 in the material interpolation scheme, a choice widely supported in the literature [40]. This paper adopts the same value for consistency with established practices.

$$\alpha_i = \frac{1}{p}\frac{df_i}{dx_i} = \begin{cases} \frac{1}{2f_i}\boldsymbol{u}_i^T\left(\boldsymbol{K}_i^1 - \frac{f_i^2}{p}\boldsymbol{M}_i^1\right)\boldsymbol{u}_i & for\ x_i = 1 \\ -\frac{f_i}{2p}\boldsymbol{u}_i^T\boldsymbol{M}_i^1\boldsymbol{u}_i & for\ x_i = x_{min} \end{cases} \tag{7}$$

Lastly, a mesh-independence filter is designed to address issues such as checkerboard patterns and mesh dependency in TO problems by averaging the sensitivity numbers, also called nodal sensitivity numbers, of an element with those of its neighbouring elements, employing image-processing techniques to smooth out the results [41]. The sensitivity numbers are denoted as α_j^n, and they are calculated by averaging the sensitivity values of the connected elements. As shown in Eq. 8, these node-based sensitivity values are then remapped into the elements using a filtering scheme, using a radius of action defined by r_{ij}, which represents the distance between the center of the i-th element and node j, being M is the total number of nodes in the structure, and $w(r_{ij})$ a weight factor defined as in Eq. 8.

$$\widehat{\alpha_i} = \frac{\sum_{j=1}^{M} w(r_{ij})\alpha_j^n}{\sum_{j=1}^{M} w(r_{ij})}, where\ w(r_{ij}) = \begin{cases} r_{min} - r_{ij}\ for\ r_{ij} < r_{min} \\ 0 \qquad\quad for\ r_{ij} \geq r_{min} \end{cases} \tag{8}$$

The update strategy for the design variables involves transitioning elements with the lowest sensitivity values from 1 to x_{min}, while elements with the highest sensitivity values transition from x_{min} to 1. The modified BESO method initially applies to a bulk design domain and progressively reduces the total structural volume. An evolutionary rate controls this reduction (E_r), which specifies the proportion of volume removed relative to the current design volume. Concurrently, the added volume is restricted by a maximum addition ratio (A_r), indicating the maximum allowable addition as a fraction of the current volume. Once the target volume is reached, the design volume is held constant. A good pre-set value is to choose $E_r = A_r$, according to the benchmark study [35], as will be set in this research. The evolutionary process terminates upon achieving convergence, defined as a change in the evaluated natural frequency below a specified tolerance (set to 10^{-3} in this study).

Huang's method, like BESO, SIMP, and other TO approaches [42, 43], requires substantial manual tuning of numerical parameters to ensure effective convergence. Inadequate parameter settings can result in suboptimal designs or even nonsensical outcomes. Table 1 presents the tuning parameters used for the cantilever beam problem analyzed in this study, along with meta-features describing the design problem

Accepted manuscript

ACCEPTED MANUSCRIPT

and metrics for quantitatively evaluating solution quality. These metrics include the frequency ratio between the optimized and initial designs and the simulation runtime.

Input parameter	Description	Symbol
Mesh size X	N. of discrete elements along X	$mesh_x$
Mesh size Y	N. of discrete elements along Y	$mesh_y$
Evolution rate	Huang's volume decrease rate	E_r
Filter radius	The radius of the mesh independence filter of BESO	r_{ij}
Meta-features		
Final volume fraction	Target % of the volume occupied by material	V_f
Point mass position	Position of the point mass as % of beam length	x_m
Point mass weight	Weight of the point in kg	m
Outputs		
Simulation metric	Optimized vs initial design frequency ratio (as close as possible to 1)	$\Phi = \frac{f_n}{f_{n_0}}$
Simulation running time	Time required to achieve convergence of the TO algorithm	t

Table 1 Tuning parameters for the cantilever beam TO problem, solved with Huang's algorithm

In this research, an ML-based framework is proposed to optimize the input parameters of the frequency-based TO algorithm of Huang et al.. This optimization should guarantee the production of lightweight, optimized 2D cantilever beam structures. Huang's algorithm has been considered the benchmark for this study due to its simplicity of implementation and consistency, even if the switching mode issue is not addressed; however, an alternative TO algorithm can be easily implemented to build the dataset similarly and replicate the proposed framework.

To assist designers in selecting the appropriate set of input parameters, this research proposes a robust tool that integrates an NN with a BO. This tool suggests suitable parameters and provides an initial estimate of the simulation metrics. The Neural Network and Bayesian Optimizer are detailed in the following section. The data- generation workflow (Fig. 2, steps S1–S3) starts from a 2- D beam definition and produces a raw pool of 3000 simulations.

2.2 Neural Networks and Bayesian Optimizer

Fine-tuning the input parameters for simulations in TO is a complex and challenging task that has long hindered progress in this field. This study focused on developing a methodology to achieve optimal quality metrics for TO, a critical goal.

The overall procedure is depicted in Fig.2, where two distinctive phases are highlighted. The offline phase regards all the procedures to create the final dataset (S1-S5), while the online phase is assumed to be the phase where the NN surrogate is trained and then used with the BO to get the optimal parameters (S6-S8).

An NN was trained to approximate the relationships between input parameters and output metrics to address the scope of the research. NNs are well-suited for modeling complex, nonlinear functions [44, 45], making them ideal for capturing the interactions between input parameters such as V_f, E_r, r_{ij}, $mesh_x$,

Accepted manuscript

ACCEPTED MANUSCRIPT

$mesh_y$, x_m, and point mass m, and the resulting quality metric Φ. However, for completeness, it is important to highlight that NNs are not the only approaches capable of emulating complex non-linear models. Indeed, there are different types of surrogate models (also known as metamodels) are to approximate the behaviour of complex systems by fitting data from the original model, thereby reducing computational cost while aiming to maintain acceptable accuracy. Various metamodelling techniques co-exist with the NNs, including Kriging based on Gaussian process modelling [46], methods based on variable separation, and methods based on basis functions, such as NURBS [47]. Specifically, NURBS hyper-surfaces approximate high-dimensional functions and fit given data points. NURBS entities are well-known for their flexibility and ability to represent complex geometries, making them versatile tools for various applications beyond geometric modeling [48]. Nonetheless, ANNs are used in this research due to their wide adoption in similar studies [30].

A key innovation of this approach was integrating a Bayesian Optimizer [49] to identify optimal parameter sets. Typically, using BO would require running a simulation for every input configuration, a computationally expensive process. Training the NN on input-output data made it possible to predict outputs directly from inputs without running full simulations. This method significantly enhanced the efficiency of the BO, reducing both time and computational cost.

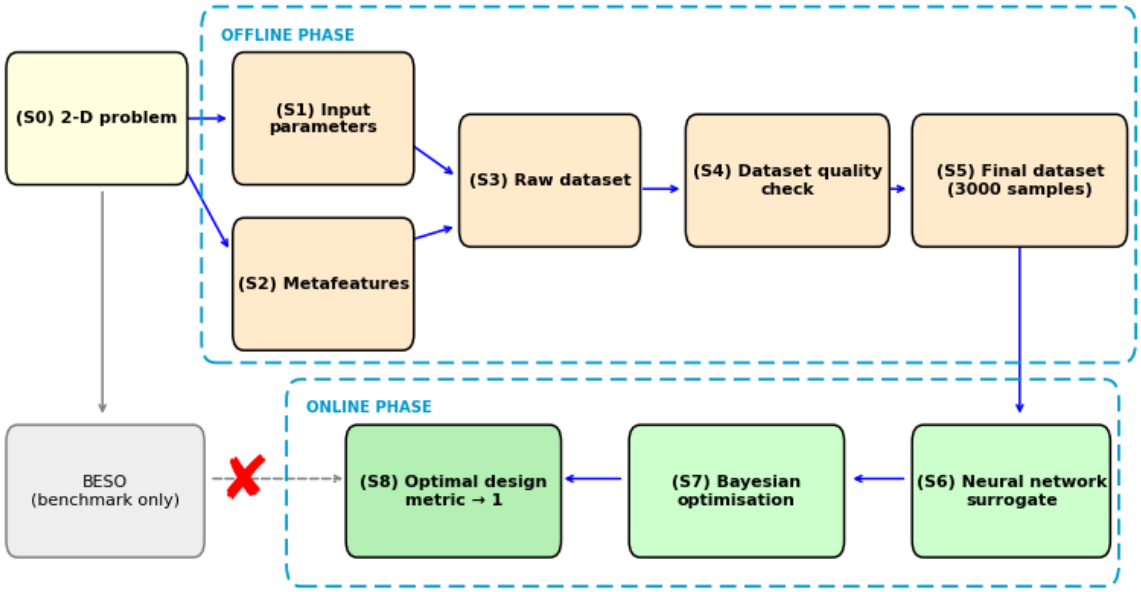

Fig. 2 Flowchart of the overall methodology: the cumbersome trial-and-error process for achieving optimal design with BESO is avoided through the use of a neural network and Bayesian optimization

In essence, training an NN to model the input-output dynamics enabled the effective use of BO in scenarios where computational costs would otherwise be prohibitive. This strategy opens new possibilities for TO, allowing for the exploration of more extensive and complex parameter spaces with far greater efficiency.

The proposed methodology leverages deep neural networks to model the complex relationship between input parameters and output metrics, ultimately improving TO quality metrics. NNs, known for their ability to learn and generalize from complex datasets, are increasingly used for such tasks. The study focused on a specific architecture and training strategy to address the challenges unique to TO.

ACCEPTED MANUSCRIPT

The reference database used for hyperparameter tuning comprised 3000 simulations of Huang's algorithm with randomly perturbed cantilever-beam configurations. The dataset is then split 70-20-10 % into training, validation, and test subsets.

The NN employed is a feedforward model comprising three layers with 16 neurons each. It takes seven input parameters and outputs values of time and metric, with the Rectified Linear Unit (ReLU) activation function [50] enabling nonlinear transformations. Because the dataset is low- dimensional and tabular, a simple feedforward multilayer perceptron was the natural starting point. We performed a coarse- to- fine manual grid search: first varying the number of hidden layers from 1 to 5 and the neurons per layer from 8 to 64, then refining around the most promising ranges. Validation- set MSE rapidly decreased to three hidden layers of roughly sixteen neurons each and levelled off thereafter. This architecture balances problem complexity with computational efficiency. Figure 3 represents the NN adopted and its structure.

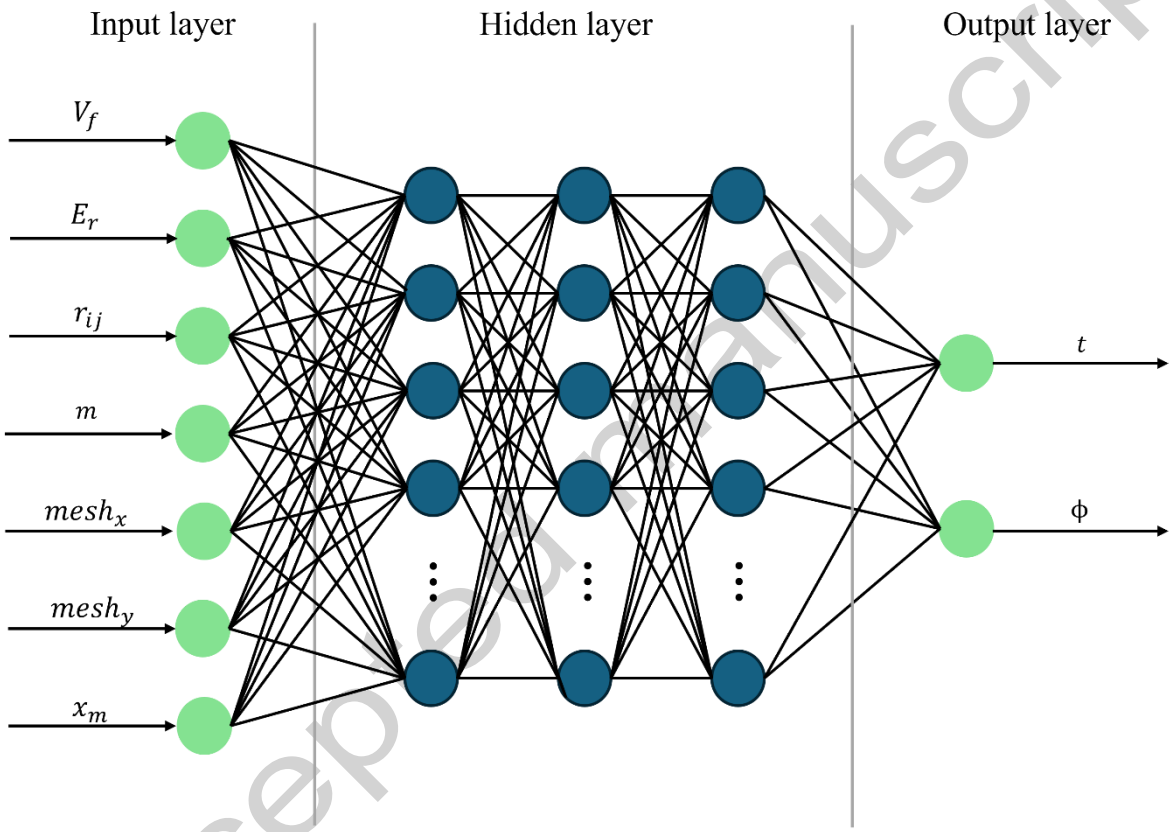

Fig. 3 ANN model with seven inputs, three hidden layers, and final outputs

Training was critical to ensure the network captured the data patterns effectively. The model was trained over 300 epochs, each representing a full pass through the dataset, allowing iterative refinement of weights and biases. The learning rate was tuned across $1 \times 10^{-4} - 1 \times 10^{-2}$. A value of 0.005 converged about 25 % faster than the default 0.001, yet remained stable, whereas 0.01 occasionally diverged. The Adam optimizer [51], known for adaptive learning rate adjustment, was employed to streamline gradient-based optimization. The training was conducted in batches of 32 samples, balancing computational efficiency and convergence.

Table 2 summarizes the hyperparameters used to train the NN, which includes architectural and optimization parameters carefully tuned for TO applications. Before using data to train the NN, the "*MinMaxScaler*" function was used to scale data in the range [0, 1].

ACCEPTED MANUSCRIPT

Hyperparameters	
Type of network	Feedforward
N° of layers	5 (3 hidden layers)
N° of inputs	7
N° of hidden neurons	16
N° of outputs	2
Activation function	ReLu
N° of epochs	300
Learning rate	0.005
Optimizer	Adam
Batch size	32

Table 2 Hyperparameters and Neural Network characteristics

The formula is shown in Eq. 9 and 10, where:

- X is the input data;
- X_{min} and X_{max} represent the minimum and maximum values for each feature (column) in X;
- max and min define the desired range for scaling (in our case, [0, 1]).

$$X_{std} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{9}$$

$$X_{scaled} = X_{std} \cdot (max - min) + min \tag{10}$$

During optimization, the network minimizes the Mean Squared Error (MSE) loss defined in Eq. 11, which is chosen for its smooth differentiability and strong penalty on outliers. MSE is tracked on the training (70 %) and validation (20 %) folds to monitor convergence and guard against over-fitting.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{11}$$

In Eq. 11 n is the total number of data used, y_i is the i^{th} real (or ground truth) value of the data, while $\hat{y}_i$ is the i^{th} value predicted by NN.

After training, predictive accuracy is assessed on an independent test subset (10 % of the data, unseen during optimization). In addition to MSE, we compute Normalised Root Mean Squared Error (NRMSE), Mean Relative Error (MRE), coefficient of determination (R^2), and maximum local error ($max|e_i|$) as shown in Eq.12.

$$NRMSE = \frac{\sqrt{MSE}}{y_{max} - y_{min}}, MRE = \frac{1}{n}\sum_{i}\frac{y_i - \hat{y}_i}{y_i}, R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}, max|e_i| \tag{12}$$

Accepted manuscript

ACCEPTED MANUSCRIPT

Bayesian optimization is a probabilistic framework designed for optimizing objective functions that are computationally expensive to evaluate. It is particularly well-suited for scenarios where the function lacks a closed-form expression, is treated as a black box, and may exhibit noisy evaluations.

The optimization process in BO begins with constructing a surrogate model that approximates the behaviour of the objective function. This surrogate model is typically built using Gaussian Process Regression (GPR), a Bayesian machine learning approach capable of providing predictions and associated uncertainties. By leveraging the probabilistic nature of GPR, BO dynamically balances exploration (sampling areas with high uncertainty) and exploitation (sampling areas predicted to yield optimal values) to efficiently navigate the search space.

An acquisition function is then defined based on the surrogate model, guiding the selection of candidate points for evaluation. These points represent locations in the input space where the subsequent function evaluations should be performed to improve the optimization process.

The optimization task can be mathematically formulated as shown in Eq. 13, where $f(x)$ represents the objective function to minimize, subject to the following constraints:

1. f is treated as a black-box function with no explicit analytical form.
2. Evaluations of f are computationally expensive.
3. Evaluations of $y = f(x)$ may be noisy.

$$x^* = argmin_x f(x) \tag{13}$$

In this study, the objective function, f is explicitly defined to assess the fitness of optimized parameters. The function is represented in Eq. 14, where ϕ is a fitness metric that quantifies the model's performance under evaluation:

$$f_{obj}(\Phi) = (|1 - \Phi|)^2 \tag{14}$$

Within BO, the NN surrogate acts as a black-box predictor x ↦ ϕ; the predicted ratio is converted into the explicit fitness (Eq.14) that the acquisition function minimizes.

This optimization identifies parameters that maximize the fitness metric, ensuring its value approaches 1. The quadratic formulation of the objective function imposes a strict penalty on deviations from this target value, effectively driving the optimization process toward solutions that achieve high model accuracy and reliability.

BO iteratively refines the search for optimal parameters by sequentially updating the surrogate model based on observed data, ensuring efficient parameter space exploration even in the presence of noise and computational constraints. This methodology is particularly advantageous in cases where function evaluation costs prohibit exhaustive search strategies.

The comparison employed the Absolute Percentage Error (APE) between the surrogate-predicted metric $\widehat{\Phi}$ and the BESO-computed metric Φ_{BESO}, as shown in Eq. 15.

$$APE = 100 \left| \frac{\Phi_{BESO} - \widehat{\Phi}}{\Phi_{BESO}} \right| \tag{15}$$

From previous studies developed by the authors [52], it was possible to select the most extensive set of data valid to train the proposed NN, which was fixed to 3000 runs of Huang's algorithm with randomly

ACCEPTED MANUSCRIPT

chosen 2D cantilever beam problems. This research has also used the same maximum size as a starting point. However, one of the aims of this paper is to find the optimal size of the dataset useful to train an NN capable of adequately tuning input parameters for the TO algorithm. This methodology demonstrates how NNs can benefit TO, making it feasible to explore larger parameter spaces with unprecedented precision and efficiency.

2.3 Dataset quality analysis metrics

Evaluating the quality and orthogonality of a dataset used to train an NN involves assessing its ability to represent the problem effectively and support learning. Different metrics are used to evaluate the orthogonality and thus the dataset quality [53], such as:

- Pairwise correlation to check correlations among input features to avoid redundancy;
- Principal Component Analysis (PCA) to transform the dataset into a set of linearly independent (orthogonal) principal components to check if it is dominated by a few principal components, indicating that some features are redundant or highly correlated;
- Visual asset dataset input distribution analysis to ensure that the input space covers a wide range of possible scenarios and that the data distribution is balanced;
- Objective value distribution analysis to ensure that the dataset includes examples where the algorithm successfully found reasonable solutions, as well as challenging cases where it failed or converged poorly to help the neural network generalize;
- Condition number κ (the ratio of the largest to smallest singular values) helps to highlight multicollinearity (lack of independence) in the case of large values [54].

Checking the orthogonality of a dataset used to train an NN refers to evaluating whether the input features are linearly independent or exhibit high redundancy or correlation. In the context of ML, orthogonality often relates to feature independence and feature space quality.

The screening applied to each candidate dataset—pairwise-correlation matrix, PCA variance spectrum, distribution plots, and the condition number κ—has a dual purpose.

First, it guarantees that the seven input features are as orthogonal as possible, limiting redundancy and keeping the numerical problem well-conditioned. Although every candidate set yields $\kappa < 10$ (the usual multicollinearity alarm level), we monitor how κ evolves as the sample size is reduced. A modest rise ($\approx 4.0 \rightarrow 4.1$ when moving from 3000 to 500 samples) is acceptable; a sharper uptick signals that the network capacity is no longer fully supported by the data, and over-fitting is likely. In practice, in this study, we adopt the smallest dataset whose κ still sits on the "flat" portion of the curve, 500 samples.

Second, a well-conditioned dataset feeds directly into model accuracy: it lowers the variance of the NN predictions and produces tighter uncertainty bands. Because Bayesian Optimisation relies on both the predicted mean and its uncertainty to steer the search, a sharp surrogate accelerates convergence and improves reliability, whereas an over-fitted surrogate inflates uncertainty and can misguide the optimizer.

3. Results

The largest dataset for training the Neural Network was generated using 3,000 distinct runs of Huang's algorithm implemented in MATLAB. The dataset was created on a workstation equipped with an i9-14900F CPU @2.0 GHz and 96 GB of RAM, requiring a total elapsed time of 40.8 hours. Input parameters were varied randomly to simulate different cantilever beam configurations, with meta-feature values sampled within the ranges specified in Table 3. These intervals were determined based on reference values from the literature tailored to Huang's algorithm [35] for this study.

Accepted manuscript

ACCEPTED MANUSCRIPT

Symbol	Interval for dataset creation
$mesh_x$	[112, 320]
$mesh_y$	$\frac{1}{8} mesh_x$
E_r	[1, 4]/100
r_{ij}	$5 \cdot \frac{dimension\ beam\ in\ X}{mesh_x}$
V_f	[0.4, 0.9]
x_m	[0, 1]
m	[0, 1000]
Φ	[0, 1]
t	$]0, +\infty[$

Table 3 Interval of values used for the tuning parameters for the cantilever beam TO problem, solved with Huang's algorithm to build the dataset

The 3000-point dataset, saved as a .csv file, has been analyzed using the tools mentioned before and iteratively reduced in length to find the optimal trade-off between good dataset quality and minimal computational time to build it. The initial dataset is described as a 3000x9 matrix, where each column represents the values of the seven inputs of the NN and both the simulation time and metric Φ.

The dataset has been iteratively reduced in length by neglecting the last n rows, and the analysis repeated. The tested lengths are 3000, 2000, 1000, 500, and 250 data. Only the graphs related to the 3000 dataset are reported in the following for the sake of brevity, but a similar analysis has been applied for each tested size.

Before applying the analysis tools, the dataset was normalized because, without normalization, features with larger magnitudes can dominate measures like covariance, correlation, and PCA, leading to biased results. Thus, a min-max scaling (Eq. 9) has been applied to scale the features to a range of [0, 1], useful for datasets where relative magnitudes are important.

Fig. 4 shows the pairwise correlation matrix to check for correlations and dependencies between the inputs. Due to the dependencies of r_{ij} and $mesh_y$ to $mesh_x$, the correlation matrix shows values close to 1 and -1, as expected, while the other values reflect an absence of dependency between inputs. Similar results are obtained for all the smaller datasets.

Accepted manuscript

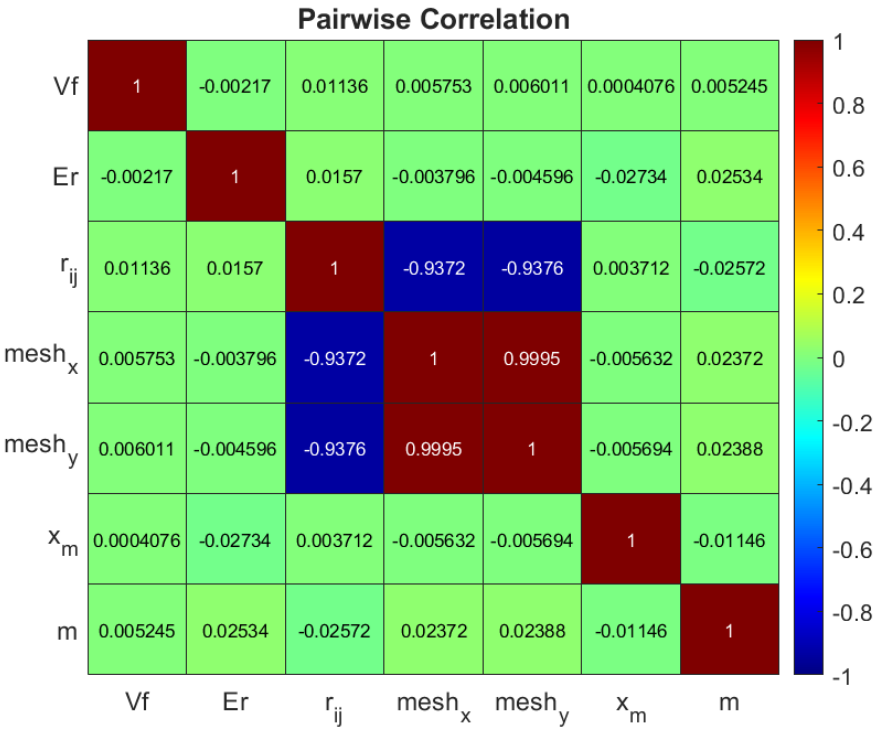

Fig. 4 Pairwise heat map of the correlation matrix between the inputs for the 3000-data: values close to 0 indicate independence (orthogonality). Values close to 1 or -1 indicate strong linear dependencies

The PCA analysis in Fig. 5 shows the variance distribution of the principal components. The dependent input variables r_{ij} and $mesh_y$ are neglected because they are known dependent variables. This analysis indicates that none of the other inputs are redundant or highly correlated for all the dataset dimensions, as the important information in the dataset can be captured by just the first four main components. Considered together, the four parameters $V_f, E_r, mesh_x$ and x_m account for more than 98% of all the differences found in the data, which means the dataset is highly structured and can be described using fewer dimensions without losing much detail.

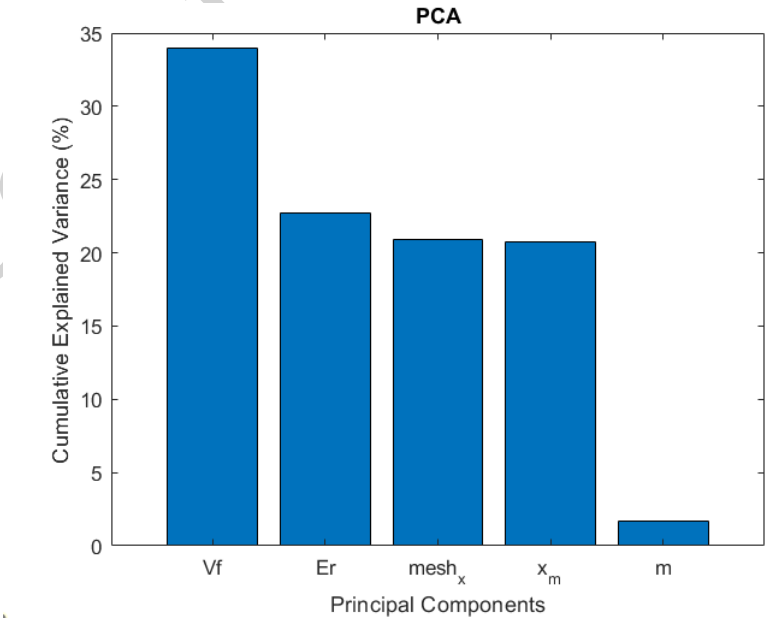

Fig. 5 PCA analysis on the 3000-data: all the principal components explain most of the variance with the same magnitude of principal component variances; such a result demonstrates that none of the inputs is redundant or highly correlated.

Moreover, the distribution inside the dataset of each input has been visually checked (Fig. 6) to look for skewness, imbalance, or anomalies in feature distributions, but the plots show a balanced spreading of the inputs for all five tested datasets.

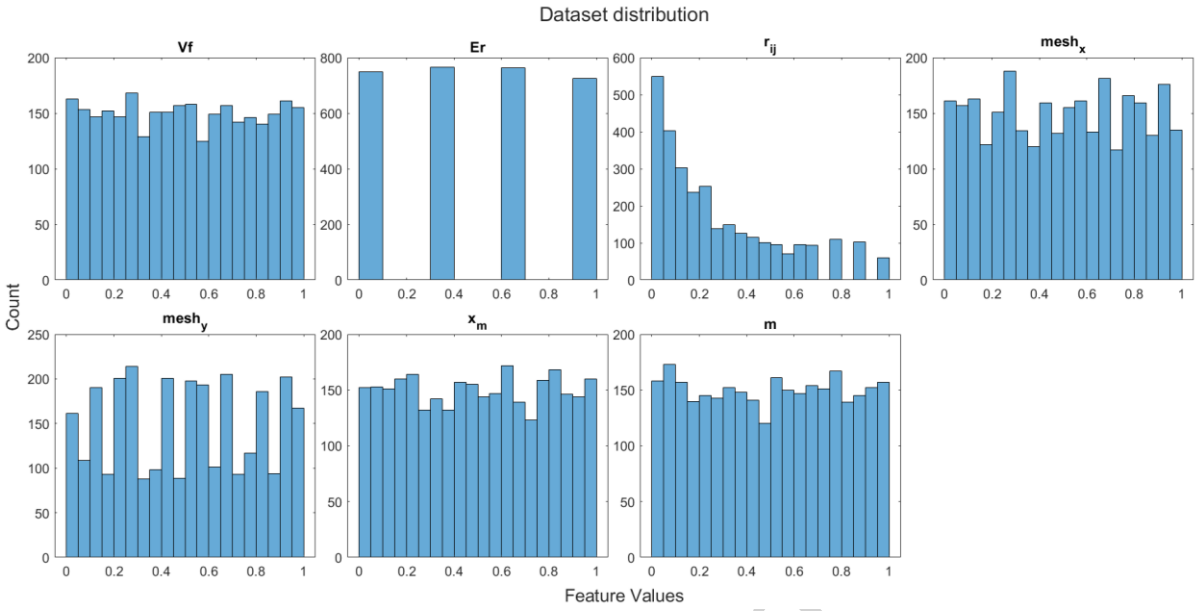

Fig. 6 Input distribution inside the intervals reported in Table 3 for the 3000-data: no imbalance or anomaly is visible in the built dataset

As a last visual check, the analysis of how the input features influence the simulation metric has been plotted for each dataset. For example, Fig. 7 reports the output behaviour of the 3000-length dataset: no clear trend can be depicted to indicate meaningful relationships between inputs and outputs, denoting that no bias or strong dependency is present inside the dataset for all the input variables, excluded x_m.

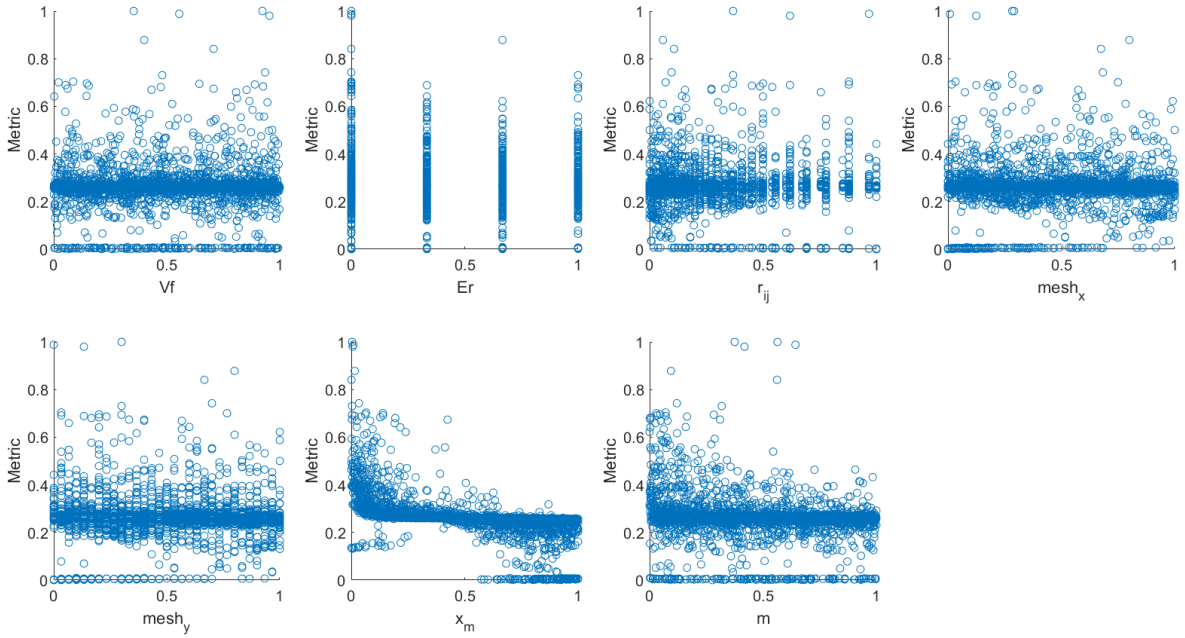

Fig. 7 Output distribution when the input parameters are varied inside their intervals for the 3000-data: no imbalance or anomaly is visible in the built dataset. In this figure, the input and metric values are not normalized to improve the readability of the plots.

ACCEPTED MANUSCRIPT

Further analysis demonstrates a negative correlation for the mass position variable, as visible in Fig. 8, while it confirms that no strong correlation exists between the remaining set of inputs and the output metric. Regarding the mass position, as x_m. increases from 0 to 1, the Metric tends to decrease. Moreover, the spread of the points is larger for smaller x_m, and becomes tighter and lower as x_m approaches 1.

A similar result is also applied to the other dataset dimensions.

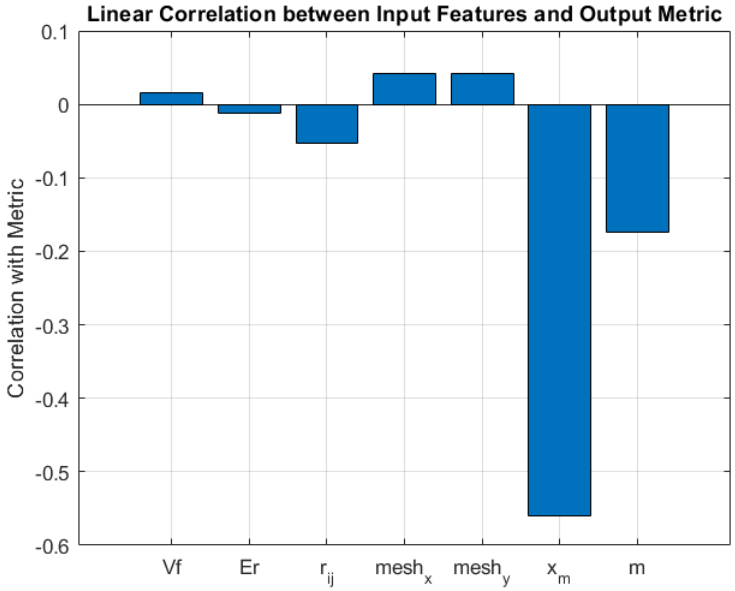

Fig. 8 Pearson correlation between each input feature and the output metric shown in a bar plot

The last parameter checked for the dataset analysis is the condition number κ to highlight a possible lack of independence between inputs and output. Empirical rules suggest that κ<10 means no significant multicollinearity, 10≤κ<30 suggests moderate multicollinearity, and κ>1000 means severe multicollinearity [55]. Table 4 collects all the computed condition numbers for the different dataset sizes and the computational time required to build them. From the last two rows of Table 4, it is possible to highlight a sudden increase in the condition number. Indeed, initial % difference fluctuations remain within the ±1.1% range, while for the 250-size database, a 4.58% difference represents an 8× larger value than the previous highest increase (0.57%) and 4.2× larger than the average absolute change (~1.1%) of other steps. The trend is confirmed with an additional halving by testing a 125-size database. The conditioning number deviation shows an accelerating growth, meaning that the dataset length with 250 data becomes the lowest bound of the dataset length, and no further shortening is necessary.

Dataset size	Condition number κ	κ % difference w.r. previous value	Dataset running time [hours]
3000	4.0351	/	40.87
2000	4.0788	1.08%	28.39
1000	4.0391	-0.97%	14.09
500	4.0623	0.57%	7.02
250	4.2483	4.58%	3.65
125	4.4685	5.18%	1.88

ACCEPTED MANUSCRIPT

Table 4 The condition number and the required time to build the different sizes of tested datasets

Once the lower bound of the dataset length has been found, these data are used to train and test the NN & BO to substitute Huang's algorithm and speed up the frequency optimization of the cantilever beams.

The NN is trained by minimizing MSE; MSE is monitored on the 70 % training folds and on the 20 % validation folds to watch for over-fitting.

Using four complementary statistics, predictive skill is quantified on an independent test subset (10 % of the data, never seen during training or validation). For the sake of brevity, the quantitative results reported in this section refer to the surrogate trained on the complete 3000-sample database:

- NRMSE = 0.2446;
- MRE = 3.6636
- R^2 = 0.6145;
- $max|e_i|$ = 0.9040.

However, in case of insufficient training data, overfitting may occur. This phenomenon is common in NN training, where a model performs very well on the training data but fails to generalize to unseen or test data. This issue happens when the network learns not only the underlying patterns in the training data but also the noise, outliers, and random fluctuations, leading to overly complex models that do not represent the actual data distribution [56]. Due to a conspicuous shortening of the dataset, the overfitting check has been included on the tested lengths to avoid this undesirable effect.

In particular, the comparison of the MSE in the train and test phase of the NN design has been used to detect the overfitting of the NN during the shortening of the dataset. For each length of the tested dataset, five runs of the training and testing phase of the NN have been executed to mitigate eventual outliers. The MSE % deviation was computed for each run, and the mean % error was calculated for each dataset size. Table 5 collects all the data.

		3000	2000	1000	500	250
Run #1	Train	0,0026	0,0054	0,0039	0,0021	0,0019
	Test	0,0024	0,0046	0,0038	0,0019	0,0028
	% Error	7,6923	14,8148	2,5641	9,5238	47,3684
Run #2	Train	0,0037	0,0027	0,0031	0,0026	0,0021
	Test	0,0028	0,0025	0,0018	0,0039	0,0017
	% Error	24,3243	7,4074	41,9355	50,0000	19,0476
Run #3	Train	0,0026	0,0035	0,0030	0,0031	0,0027
	Test	0,0027	0,0036	0,0022	0,0030	0,0091
	% Error	3,8462	2,8571	26,6667	3,2258	237,0370
Run #4	Train	0,0035	0,0027	0,0030	0,0037	0,0050
	Test	0,0033	0,0035	0,0026	0,0020	0,0010
	% Error	5,7143	29,6296	13,3333	45,9459	80,0000
Run #5	Train	0,0024	0,0047	0,0034	0,0066	0,0021
	Test	0,0023	0,0060	0,0030	0,0073	0,0218
	% Error	4,1667	27,6596	11,7647	10,6061	938,0952
Mean % error		9,1487	16,4737	19,2529	23,8603	264,3097
STD % error		8,6191	11,9223	15,3223	22,2376	385,9786

Table 5 For each length of the tested dataset, five runs have been used to access the MSE deviation between the train and test data; the mean % error is used to check the overfitting of the Neural Network training

Accepted manuscript

ACCEPTED MANUSCRIPT

A closer look at the last two rows of Table 5—*mean % error* and *std % error* calculated over five independent random seeds—confirms the expected error-growth pattern as the training set shrinks. Both indicators rise monotonically from 3000, 2000, 1000, and 500 samples and then explode when the size is halved again to 250 samples (mean = 264 %, std = 386 %), a clear symptom of overfitting. The apparently large deviations observed in a few individual runs (e.g., Runs 2 & 4 for the 500-sample model and Runs 2 & 3 for the 1000-sample model) stem from the stochastic nature of training: different initial weights and batch orders can occasionally steer the optimizer into poorer local minima. These outliers are nevertheless captured in the standard deviation column. Hence, while isolated spikes are inevitable, the aggregate statistics show a smooth, interpretable trend that justifies selecting the 500-sample database as the practical lower bound and highlights the overfitting risk below this threshold.
To better understand the results of Table 5, Fig. 9 highlights the sharp increase in MSE for the smallest dataset.

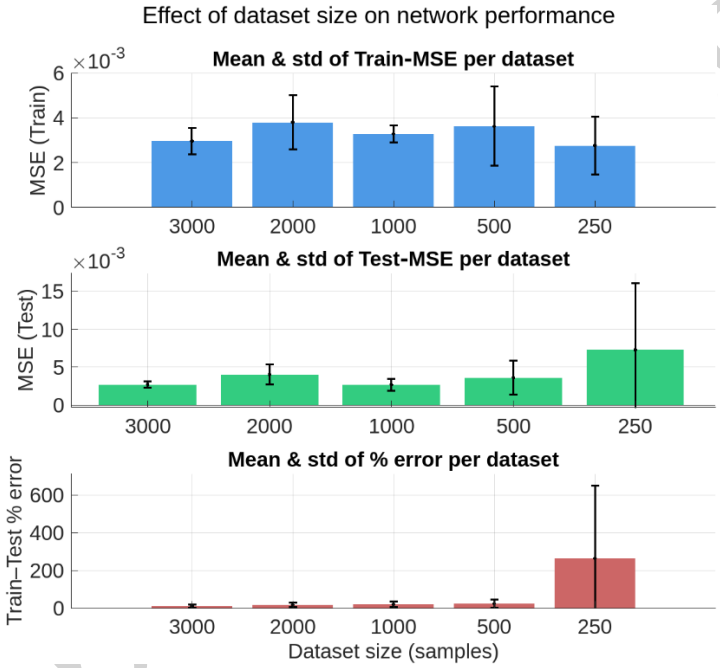

Fig. 9 *Comparison of MSE results for the different datasets trained on five runs. The figure depicts the results reported in Table 5. There is an apparent "explosion" in terms of MSE for the dataset composed of 250 samples, highlighting a clear symptom of overfitting.*

3.2 Topology Optimization of Cantilever Beam

After training the NN with the four remaining datasets (3000, 2000, 1000, and 500 data), the model was coupled with the BO for the final tests to assess the capability of replacing Huang's method to accelerate the frequency optimization of cantilever beams. Eleven randomly selected meta-feature configurations were tested to evaluate the BO's effectiveness. Each design problem has been solved with the NN&BO method, trained on different dataset sizes. The BO consistently provided a set of input parameters and the metric function within a fixed runtime of 11 seconds. The results obtained from the BO were then compared to those of Huang's approach: the suggested input parameters were applied to the eleven design problems, and both the metric and runtime were assessed using Huang's algorithm.

Table 6 presents all the data of the eleven randomly chosen cantilever beam configurations. In particular, the V_f, x_m and m metafeatures are randomly chosen inside the investigated intervals. For each simulation, the values of the other input parameters ($E_r, r_{ij}, mesh_x, mesh_y$) are suggested by the BO&NN approach for further checks with the benchmark BESO model and an estimation of the metric value. Table 6 includes the percentage error with sign in the BO's metric compared to the traditional TO approach and the

ACCEPTED MANUSCRIPT

differences in simulation time. This way, the reader can understand the under and over-estimation directions of the proposed approach. In particular, the BO-NN approach with 3000 and 2000-length datasets mainly tends to overestimate the fitness function with limited error. Once the dataset is reduced to 1000 and 500 data, this tendency is lacking, and under-estimation is visible in several runs, still with limited magnitude. In this research, it is assumed that for the initial conceptual design phase, a 15% fitness deviation is acceptable; indeed, at this stage, the designer explores all the possible structural configurations to find the optimal one. Once the user selects the optimal design using the NN&BO data-driven deterministic model, more precise numerical tools (i.e., Finite Element Analyses) should be used to assess the performance of the structural component with high accuracy.

Accepted manuscript

ACCEPTED MANUSCRIPT

Dataset dim	#	V_f	E_r	r_{ij}	$mesh_x$	$mesh_y$	x_m	m	Φ_{NN-BO}	Φ_{BESO}	Err [%]	t_{BESO}	ΔT[%]
3000	1	0,5	0,010	0,153	261	33	0,5	400	1,101	1,013	+8,687	300,81	-96,343
2000		0,5	0,032	0,125	319	40	0,5	400	1,154	1,010	+14,257	178,5	-93,838
1000		0,5	0,010	0,187	214	27	0,5	400	1,027	1,014	+1,282	88,52	-87,573
500		0,5	0,010	0,125	320	40	0,5	400	0,990	1,013	-2,270	712,05	-98,455
3000	2	0,9	0,010	0,125	320	40	0,8	500	1,083	1,001	+8,192	147,77	-92,556
2000		0,9	0,031	0,148	271	34	0,8	500	1,119	1,001	+11,788	51,19	-78,511
1000		0,9	0,010	0,385	104	13	0,8	500	0,928	1,001	-7,293	14,53	-24,295
500		0,9	0,040	0,159	251	31	0,8	500	1,040	1,001	+3,896	31,08	-64,607
3000	3	0,65	0,010	0,131	307	38	0,5	900	1,098	1,006	+9,145	334,91	-96,716
2000		0,65	0,024	0,200	200	25	0,5	900	1,114	1,005	+10,846	59,2	-81,419
1000		0,65	0,010	0,381	105	13	0,5	900	0,922	1,006	-8,350	16,12	-31,762
500		0,65	0,040	0,165	243	30	0,5	900	1,012	1,005	+0,697	27,31	-59,722
3000	4	0,75	0,010	0,125	320	40	0,7	200	1,121	1,088	+3,033	355,2	-96,903
2000		0,75	0,010	0,126	318	40	0,7	200	1,168	1,009	+15,758	228,97	-95,196
1000		0,75	0,010	0,357	112	14	0,7	200	0,939	1,009	-6,938	32,81	-66,474
500		0,75	0,040	0,187	214	27	0,7	200	1,001	1,009	-0,793	18,38	-40,152
3000	5	0,82	0,010	0,125	320	40	0,9	327	1,059	0,996	+6,325	253,05	-95,653
2000		0,82	0,030	0,144	278	35	0,9	327	1,094	0,996	+9,839	57,47	-80,860
1000		0,82	0,010	0,381	105	13	0,9	327	0,900	0,990	-9,091	24,84	-55,717
500		0,82	0,040	0,180	222	28	0,9	327	0,998	0,996	+0,201	20,16	-45,437
3000	6	0,62	0,010	0,385	104	13	0,1	180	1,783	1,661	+7,345	14,23	-22,699
2000		0,62	0,010	0,125	320	40	0,1	180	1,327	1,562	-15,045	582,25	-98,111
1000		0,62	0,010	0,320	125	16	0,1	180	1,503	1,595	-5,768	64,78	-83,019
500		0,62	0,010	0,264	153	19	0,1	180	2,230	1,624	+37,315	18,86	-41,676
3000	7	0,53	0,034	0,127	314	39	0,6	675	1,043	0,985	+5,856	168,64	-93,477
2000		0,53	0,032	0,125	320	40	0,6	675	1,076	0,986	+9,150	256,46	-95,711
1000		0,53	0,040	0,126	317	40	0,6	675	0,921	0,988	-6,781	133,27	-91,746
500		0,53	0,010	0,125	320	40	0,6	675	0,965	0,985	-2,030	623,72	-98,236
3000	8	0,71	0,010	0,125	320	40	0,1	999	1,211	1,045	+15,885	414,13	-97,344
2000		0,71	0,011	0,364	110	14	0,1	999	1,170	1,056	+10,795	16,32	-32,598
1000		0,71	0,010	0,125	320	40	0,1	999	1,213	1,045	+16,077	414,13	-97,344
500		0,71	0,011	0,248	161	20	0,1	999	1,116	1,051	+6,185	14,97	-26,520
3000	9	0,58	0,020	0,137	293	37	0,7	429	1,037	0,973	+6,578	179,98	-93,888
2000		0,58	0,030	0,125	320	40	0,7	429	1,072	0,980	+9,388	198,17	-94,449
1000		0,58	0,040	0,125	320	40	0,7	429	0,907	0,971	-6,591	175,18	-93,721
500		0,58	0,040	0,211	190	24	0,7	429	0,947	0,958	-1,148	14	-21,429
3000	10	0,86	0,010	0,125	320	40	0,2	29	1,447	1,280	+13,047	280,41	-96,077
2000		0,86	0,015	0,126	317	40	0,2	29	1,261	1,281	-1,561	157,97	-93,037
1000		0,86	0,010	0,125	320	40	0,2	29	1,184	1,280	-7,500	280,41	-96,077
500		0,86	0,040	0,328	122	15	0,2	29	1,619	1,268	+27,681	21,75	-49,425
3000	11	0,64	0,019	0,125	320	40	0,5	862	1,115	1,006	+10,835	305,46	-96,399
2000		0,64	0,026	0,198	202	25	0,5	862	1,112	1,006	+10,537	17,19	-36,009
1000		0,64	0,029	0,125	319	40	0,5	862	0,927	1,006	-7,853	109,48	-89,953
500		0,64	0,040	0,164	244	31	0,5	862	1,006	1,005	+0,100	29,56	-62,788

Table 6 Eleven designs were used to test the NN & BO method; the % error on the metric function between Huang's algorithm and BO is evaluated, and the running time of each simulation is compared.

Accepted manuscript

ACCEPTED MANUSCRIPT

Thus, assuming a 15% deviation as the maximum threshold in consistency with the previous study [52], Fig. 10 shows that the mean absolute % error for the NN&BO metric, compared to the actual metric evaluated through Huang's algorithm, shows satisfactory values. For three of the four tested sizes of the dataset, the mean % error is below the threshold value. However, the 500-length dataset has a higher dispersion of error values, as visible by the highest standard deviation rate, and from runs # 6 and 10, with a metric error higher than 25%.

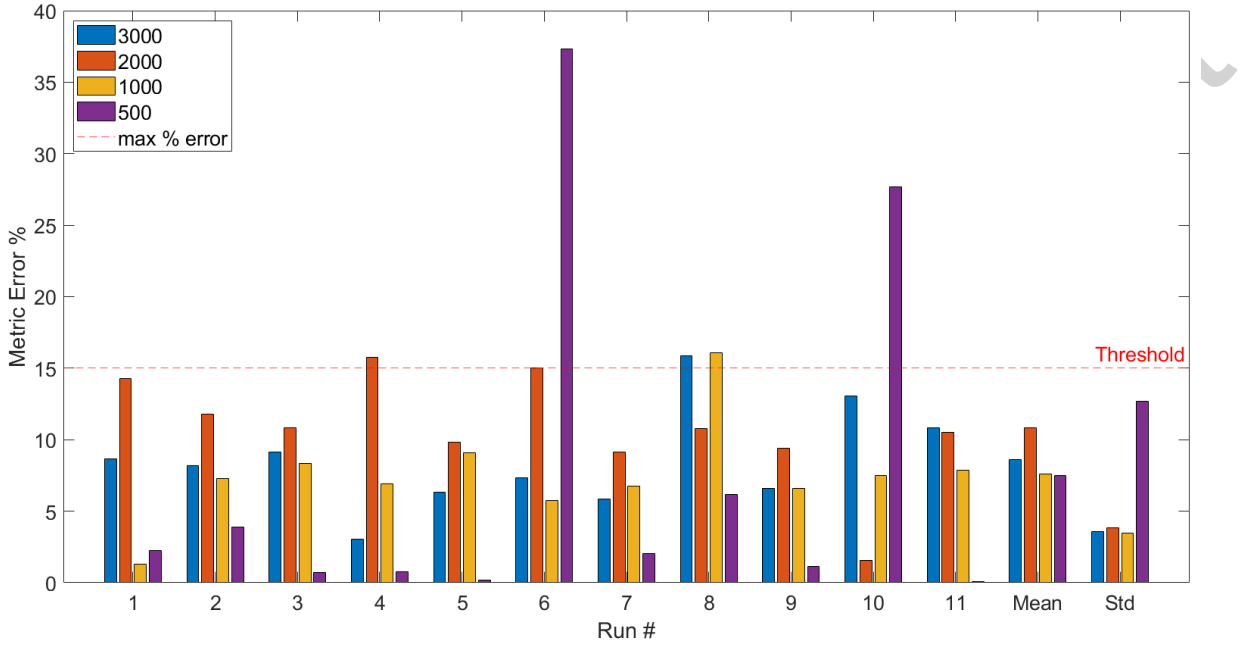

Fig. 10 *Metric error % when comparing the NN & BO approach with Huang's algorithm computed for each run for each dataset size; the last two sets of data represent the mean and standard deviation among the different runs.*

Lastly, Fig. 11 shows three randomly chosen optimized cantilever beams after the TO using the input parameters tuned by the NN&BO framework.

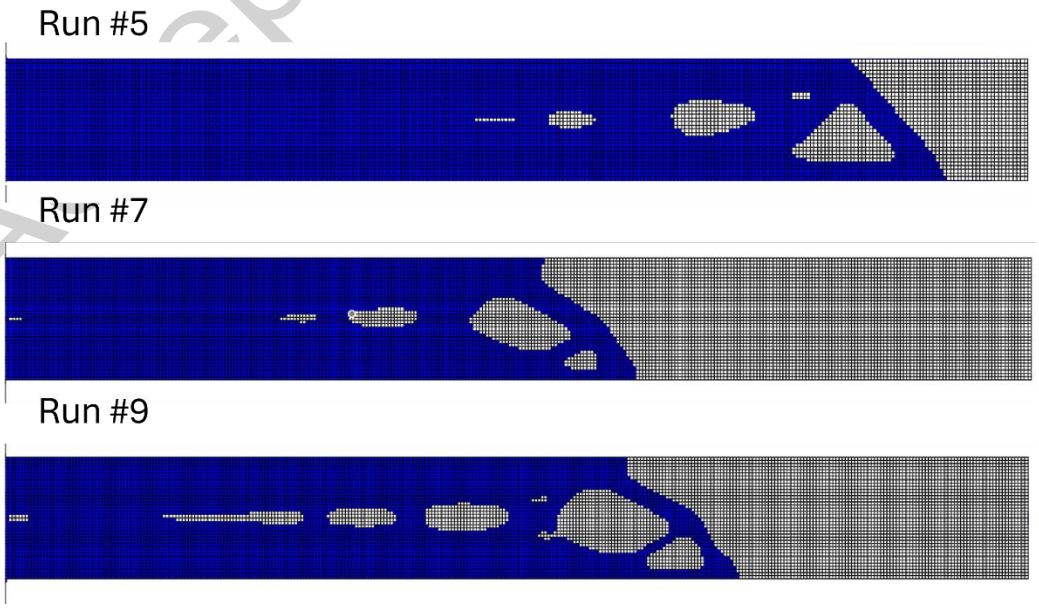

Fig. 11 *Comparison of the optimized cantilever beam structure of three different configurations*

ACCEPTED MANUSCRIPT

One of the main limitations of the proposed approach is its application to 2D benchmark problems; however, the user can easily modify the boundary conditions applied to the 2D beam and apply to similar case studies the proposed methodology to tune and create a surrogate model for structure optimization. The extension of the methodology to 3D topology optimization will be investigated in future research based on the approach developed in this study.

4. Discussion of the results

The results in the previous section demonstrate that when many configurations should be tested and structurally optimized in the preliminary design stages, the NN&BO framework can provide consistent results if compared to Huang's algorithm but with a speedup of the numerical simulations. This result is obtained while keeping the same dynamic response of components. The proposed method can fine-tune the input parameters for Huang's algorithm and provide a satisfactory estimation of the metric function.

The estimation process is possible thanks to an NN trained on a large dataset of similar problems. However, accessing the correct dimension of the required data is not straightforward. This study demonstrates that it is possible to find the optimal dataset size by analyzing its condition number, which is the best factor that expresses the dataset's quality compared to the other tested tools in this manuscript. Among the tested sizes of 3000, 2000, 1000, 500, and 250 data, the results demonstrate that 500 data are sufficient to describe the design space entirely for the case study (Table 4). Indeed, the 250-size and 125-size datasets show a κ % difference increase; thus, an additional halving may lead to potential numerical instability during training, as it suggests that some features are linearly dependent or nearly so.

The analysis of the dataset demonstrates that the input and output variables for the TO in frequency have been correctly chosen and do not present strong correlations apart from r_{ij} and $mesh_y$ which has a known dependency to $mesh_x$ (Fig. 6). Moreover, the simulation metric output does not show biases due to input parameters. There is just a faint dependency of the metric with r_{ij}: generally, small values of the filter radius positively influence Φ with a higher tendency of being close to the optimal unitary value (Fig. 8). Furthermore, analyzing the depth the input-output correlation, as x_m (the relative position of an added mass along the beam) increases—from the fixed end (0) to the free end (1)—the metric tends to decrease and stabilize to the optimal value of 1. When the added mass is closer to the free end, it has a larger impact on the natural frequency, and the BESO algorithm must retain more material in critical regions to keep constant the natural frequency. However, when the added mass is far from the fixed end, this constraint becomes harder to satisfy with a low volume fraction, leading to a compromise: the optimizer allows a non-optimal fitness value ($metric \neq 1$) in order to reduce mass. The tighter spread near $x_m = 1$ suggests more predictable performance in those configurations. Conversely, the larger variance near the fixed end may reflect more design flexibility when the mass affects the system less significantly.

Another important result of this research comes from the overfitting study. In particular, the overfitting phenomenon during the training and test phases of the NN is a critical and limiting factor in the correct quantity of data required by the NN to estimate the output accurately, and the dataset cannot be reduced tremendously. From this research, it has been found that the lower bound of the dataset size is again 500 data, which coincides with the condition number analysis (Table 4).

Regarding the results that come from the TO of cantilever beam analysis, it is possible to say that the proposed NN&BO framework is a valid substitute for Huang's algorithm with a reasonable estimation of the objective metric (the frequency ratio). Indeed, the mean absolute deviation is about 10% for all the tested sizes of the dataset, with a substantial temporal saving from 55% up to 88%, respectively, from 500 to 3000 data used. The signed error deviation tends to show a slight overestimation when using larger datasets, while for smaller datasets, some runs also exhibit underestimation, still generally within the 15% range.

Accepted manuscript

ACCEPTED MANUSCRIPT

Moreover, across 11 runs, only two design configurations exhibited deviations exceeding 15% (Fig. 8), which is a promising outcome. These two designs are characterized by an added mass position close to the fixed end, and input-output correlation analysis showed that for values of x_m close to 0, a high metric value spread is noted. One configuration showed a 37% deviation (500-size dataset); however, Huang's method produced a metric value far from the optimal design ($\Phi \sim 1$), yielding a value of 1.6 among the different tested datasets. In contrast, the BO approach closely approximated the high metric value achievable with the recommended input parameters. These results support the aim of this research and significantly streamline the iterative tuning process, eliminating the need for labour-intensive manual adjustments of input parameters while optimizing the target structure, thereby accelerating the time-to-market for specific products.

The last important result emerges from the estimation of the mean and standard deviation of the metric % error (Fig. 10). It is possible to say that the best trade-off between accurate metric estimation, high dataset quality and minimal simulation time to build the dataset can be achieved with 500 data, among the sizes tested in this research. This way, from the starting size of 3000 runs of Huang's algorithm, an 82.9% time saving can be achieved, lowering the total time needed to build the dataset from more than 40 hours to just 7 hours without compromising the method's accuracy. The methodology developed and described in this paper can be applied to any structure, and its extension to 3D problems is also possible. The extension of the methodology to 3D topology optimization will be investigated in future research.

The following possible limitations, however, affect the suggested approach:

- only one particular mechanical problem (cantilever beam) has been tested as part of the preliminary research;
- NNs are probabilistic; therefore, the estimated simulation metrics do not correspond to the actual ones.

Nonetheless, the findings show that a favourable forecast can be made during the initial design phases in the frequency TO industrial contexts.

5. Conclusions and future developments

Even for structures subjected to dynamic loads, topology optimization has significant potential for developing efficient mechanical components and systems. However, these methods frequently need plenty of human adjustments, particularly when changing parameters that affect the convergence and behaviour of the algorithm. In order to overcome this challenge, the study suggests a Bayesian optimizer based on a Neural Network framework as a surrogate model that advises users to adjust parameters in a well-consolidated approach available in the literature, widely adopted in the TO frequency context, avoiding the costly trial-and-error procedure involved in manual adjustments.

The dataset used to train the NN was thoroughly investigated for quality and dimension using metrics such as PCA and conditioning number. Results showed that the condition number, aligned with the study of the overfitting phenomenon, determined the minimum dataset size for reliable model performance. For this study, the dataset length was set to 500 data points. Furthermore, constructing this dataset resulted in an 83% time saving compared to larger datasets used in the previous study, highlighting the efficiency of the proposed approach.

The proposed approach has two primary phases that incorporate NN and BO. An NN is first trained using the dataset that contains input parameters, meta-features, and the simulation metric from related structural problems as part of the learning process. The second phase involves structural optimization, which returns an estimated simulation metric by fine-tuning the input parameters using a Bayesian optimizer. The findings show that the NN-BO framework may be regarded as a legitimate alternative to the approach available in the literature in the early phases of design when several configurations need to be

Accepted manuscript

ACCEPTED MANUSCRIPT

tried. The new method allows for a time reduction of over 55% with a 15% error limit, regardless of the examined meta-features.

Nevertheless, the approach has some limitations: NNs are probabilistic and do not always provide the same outcomes. Furthermore, a more thorough investigation employing more generative NNs that directly deal with b/w matrices that describe the mass distribution might enhance this study. Instead of only suggesting tuning parameters and estimating the optimized frequency, this method would use the same input data as the present study to provide outputs that reflect the new structure's outcomes.

Data availability. The authors declare that the data supporting the findings of this study are available within the paper.

Compending interests. The authors declare that they have no competing interests.

Replication of the results. All the information required to replicate the results found in Section 3 is fully disclosed in this paper. The same results will be obtained by implementing the methodology presented here. The algorithms discussed in this paper are implemented in Matlab by the authors and can be shared by contacting the corresponding author.

References

1. Ibhadode, O., Zhang, Z., Sixt, J., Nsiempba, K.M., Orakwe, J., Martinez-Marchese, A., Ero, O., Shahabad, S.I., Bonakdar, A., Toyserkani, E.: Topology optimization for metal additive manufacturing: current trends, challenges, and future outlook. Virtual and Physical Prototyping. 18, e2181192 (2023). https://doi.org/10.1080/17452759.2023.2181192
2. Oktay, E., Akay, H.U., Sehitoglu, O.T.: Three-dimensional structural topology optimization of aerial vehicles under aerodynamic loads. Computers & Fluids. 92, 225–232 (2014). https://doi.org/10.1016/j.compfluid.2013.11.018
3. Torigaki, T., Fujitani, K.: Power of a voxel approach to structural analysis and topology-shape optimization in automobile industries. Japan J. Indust. Appl. Math. 17, 129–147 (2000). https://doi.org/10.1007/BF03167341
4. Costa, G., Montemurro, M.: Eigen-frequencies and harmonic responses in topology optimization: A CAD-compatible algorithm. Engineering Structures. 214, 110602 (2020). https://doi.org/10.1016/j.engstruct.2020.110602
5. Cottrell, J.A., Hughes, T.J.R., Bazilevs, Y.: Isogeometric Analysis: Toward Integration of CAD and FEA. Wiley (2009)
6. López-Castro, J.D., Marchal, A., González, L., Botana, J.: Topological optimization and manufacturing by Direct Metal Laser Sintering of an aeronautical part in 15-5PH stainless steel. Procedia Manufacturing. 13, 818–824 (2017). https://doi.org/10.1016/j.promfg.2017.09.121
7. Sigmund, O., Maute, K.: Topology optimization approaches: A comparative review. Struct Multidisc Optim. 48, 1031–1055 (2013). https://doi.org/10.1007/s00158-013-0978-6

Accepted manuscript

ACCEPTED MANUSCRIPT

8. Langley, R.S., Bardell, N.S.: A review of current analysis capabilities applicable to the high frequency vibration prediction of aerospace structures. Aeronaut. j. 102, 287–297 (1998). https://doi.org/10.1017/S0001924000065325
9. Zargham, S., Ward, T.A., Ramli, R., Badruddin, I.A.: Topology optimization: a review for structural designs under vibration problems. Struct Multidisc Optim. 53, 1157–1177 (2016). https://doi.org/10.1007/s00158-015-1370-5
10. Qatu, M.S., Abdelhamid, M.K., Pang, J., Sheng, G.: Overview of automotive noise and vibration. IJVNV. 5, 1 (2009). https://doi.org/10.1504/IJVNV.2009.029187
11. Wu, Z., Fan, F., Xiao, R., Yu, L.: The substructuring-based topology optimization for maximizing the first eigenvalue of hierarchical lattice structure. Numerical Meth Engineering. 121, 2964–2978 (2020). https://doi.org/10.1002/nme.6342
12. Bendsøe, M.P.: Optimal shape design as a material distribution problem. Structural Optimization. 1, 193–202 (1989). https://doi.org/10.1007/BF01650949
13. Zuo, Z.H., Xie, Y.M.: A simple and compact Python code for complex 3D topology optimization. Advances in Engineering Software. 85, 1–11 (2015). https://doi.org/10.1016/j.advengsoft.2015.02.006
14. Pedersen, N.L.: Maximization of eigenvalues using topology optimization. Struct Multidisc Optim. 20, 2–11 (2000). https://doi.org/10.1007/s001580050130
15. BendsØe, M.P., Olhoff, N., Taylor, J.E.: A Variational Formulation for Multicriteria Structural Optimization*. Journal of Structural Mechanics. 11, 523–544 (1983). https://doi.org/10.1080/03601218308907456
16. Li, Q., Wu, Q., Liu, J., He, J., Liu, S.: Topology optimization of vibrating structures with frequency band constraints. Struct Multidisc Optim. 63, 1203–1218 (2021). https://doi.org/10.1007/s00158-020-02753-7
17. Huang, X., Xie, Y.M.: Evolutionary Topology Optimization of Continuum Structures: Methods and Applications. Wiley (2010)
18. Huang, X., Zuo, Z.H., Xie, Y.M.: Evolutionary topological optimization of vibrating continuum structures for natural frequencies. Computers & Structures. 88, 357–364 (2010). https://doi.org/10.1016/j.compstruc.2009.11.011
19. Zegard, T., Paulino, G.H.: Bridging topology optimization and additive manufacturing. Struct Multidisc Optim. 53, 175–192 (2016). https://doi.org/10.1007/s00158-015-1274-4
20. Montalti, A., Ferretti, P., Santi, G.M.: A Cost-effective approach for quality control in PLA-based material extrusion 3D printing using 3D scanning. Journal of Industrial Information Integration. 41, 100660 (2024). https://doi.org/10.1016/j.jii.2024.100660
21. Donnici, G., Freddi, M., Liverani, A.: RSM applied to lattice patterns for stiffness optimization. RPJ. 30, 345–356 (2024). https://doi.org/10.1108/RPJ-03-2024-0134
22. Murr, L.E.: Frontiers of 3D Printing/Additive Manufacturing: from Human Organs to Aircraft Fabrication†. Journal of Materials Science & Technology. 32, 987–995 (2016). https://doi.org/10.1016/j.jmst.2016.08.011
23. Montalti, A., Galiè, G., Pignatelli, E., Liverani, A.: Enhancing surface roughness of material extrusion additive manufacturing components via an innovative ironing process. Virtual and Physical Prototyping. 19, e2401929 (2024). https://doi.org/10.1080/17452759.2024.2401929

Accepted manuscript

ACCEPTED MANUSCRIPT

24. Primo, T., Calabrese, M., Del Prete, A., Anglani, A.: Additive manufacturing integration with topology optimization methodology for innovative product design. Int J Adv Manuf Technol. 93, 467–479 (2017). https://doi.org/10.1007/s00170-017-0112-9
25. Sigmund, O.: A 99 line topology optimization code written in Matlab. Struct Multidisc Optim. 21, 120–127 (2001). https://doi.org/10.1007/s001580050176
26. Han, Y., Xu, B., Liu, Y.: An efficient 137-line MATLAB code for geometrically nonlinear topology optimization using bi-directional evolutionary structural optimization method. Struct Multidisc Optim. 63, 2571–2588 (2021). https://doi.org/10.1007/s00158-020-02816-9
27. Oehlmann, P., Osswald, P., Blanco, J.C., Friedrich, M., Rietzel, D., Witt, G.: Modeling Fused Filament Fabrication using Artificial Neural Networks. Prod. Eng. Res. Devel. 15, 467–478 (2021). https://doi.org/10.1007/s11740-021-01020-y
28. Deneault, J.R., Chang, J., Myung, J., Hooper, D., Armstrong, A., Pitt, M., Maruyama, B.: Toward autonomous additive manufacturing: Bayesian optimization on a 3D printer. MRS Bulletin. 46, 566–575 (2021). https://doi.org/10.1557/s43577-021-00051-1
29. Jiang, X., Wang, H., Li, Y., Mo, K.: Machine Learning based parameter tuning strategy for MMC based topology optimization. Advances in Engineering Software. 149, 102841 (2020). https://doi.org/10.1016/j.advengsoft.2020.102841
30. Lynch, M.E., Sarkar, S., Maute, K.: Machine Learning to Aid Tuning of Numerical Parameters in Topology Optimization. Journal of Mechanical Design. 141, 114502 (2019). https://doi.org/10.1115/1.4044228
31. Shin, S., Shin, D., Kang, N.: Topology optimization via machine learning and deep learning: a review. Journal of Computational Design and Engineering. 10, 1736–1766 (2023). https://doi.org/10.1093/jcde/qwad072
32. Sanu, S.M., Aragon, A.M., Bessa, M.A.: Neural topology optimization: the good, the bad, and the ugly, http://arxiv.org/abs/2407.13954, (2024)
33. Abueidda, D.W., Koric, S., Sobh, N.A.: Topology optimization of 2D structures with nonlinearities using deep learning. (2020). https://doi.org/10.48550/ARXIV.2002.01896
34. Ha, D., Carstensen, J.: Automatic hyperparameter tuning of topology optimization algorithms using surrogate optimization. Structural and Multidisciplinary Optimization. 67, 157 (2024). https://doi.org/10.1007/s00158-024-03850-7
35. Huang, X., Zuo, Z.H., Xie, Y.M.: Evolutionary topological optimization of vibrating continuum structures for natural frequencies. Computers & Structures. 88, 357–364 (2010). https://doi.org/10.1016/j.compstruc.2009.11.011
36. Lin, H.Y., Xing, H.H.: Sensitivity Analysis in Artificial Neural Network and it's Applications on the Research of Attributes Correlation. AMR. 785–786, 1441–1446 (2013). https://doi.org/10.4028/www.scientific.net/AMR.785-786.1441
37. Li, T., Zhang, H., Fan, L., Wang, H., Liu, Q.: Research on deep neural network model construction and overfitting. In: Tiwari, R. (ed.) International Conference on Neural Networks, Information, and Communication Engineering (NNICE 2022). p. 13. SPIE, Qingdao, China (2022)
38. Feurer, M., Springenberg, J., Hutter, F.: Initializing Bayesian Hyperparameter Optimization via Meta-Learning. AAAI. 29, (2015). https://doi.org/10.1609/aaai.v29i1.9354

Accepted manuscript

ACCEPTED MANUSCRIPT

39. Zhang, X., Duan, G., Fang, Y., Yin, X., Li, W.: Topology optimization based on the improved high-order RAMP interpolation model and bending properties research for curved square honeycomb sandwich structures. Composite Structures. 351, 118643 (2025). https://doi.org/10.1016/j.compstruct.2024.118643
40. Liu, K., Tovar, A.: An efficient 3D topology optimization code written in Matlab. Struct Multidisc Optim. 50, 1175–1196 (2014). https://doi.org/10.1007/s00158-014-1107-x
41. Huang, X., Xie, Y.M.: Bi-directional evolutionary topology optimization of continuum structures with one or multiple materials. Comput Mech. 43, 393–401 (2009). https://doi.org/10.1007/s00466-008-0312-0
42. Challis, V.J.: A discrete level-set topology optimization code written in Matlab. Struct Multidisc Optim. 41, 453–464 (2010). https://doi.org/10.1007/s00158-009-0430-0
43. Rozvany, G.I.N.: A critical review of established methods of structural topology optimization. Struct Multidisc Optim. 37, 217–237 (2009). https://doi.org/10.1007/s00158-007-0217-0
44. DeVore, R., Hanin, B., Petrova, G.: Neural network approximation. Acta Numerica. 30, 327–444 (2021). https://doi.org/10.1017/S0962492921000052
45. Liang, S., Srikant, R.: Why Deep Neural Networks for Function Approximation?, https://arxiv.org/abs/1610.04161, (2016)
46. Vuillod, B., Zani, M., Hallo, L., Montemurro, M.: Handling noise and overfitting in surrogate models based on non-uniform rational basis spline entities. Computer Methods in Applied Mechanics and Engineering. 425, 116913 (2024). https://doi.org/10.1016/j.cma.2024.116913
47. Audoux, Y., Montemurro, M., Pailhès, J.: Non-Uniform Rational Basis Spline hyper-surfaces for metamodelling. Computer Methods in Applied Mechanics and Engineering. 364, 112918 (2020). https://doi.org/10.1016/j.cma.2020.112918
48. Zani, M., Panettieri, E., Montemurro, M.: On adaptive sampling techniques for metamodels based on NURBS entities from unstructured data. Computer Methods in Applied Mechanics and Engineering. 437, 117781 (2025). https://doi.org/10.1016/j.cma.2025.117781
49. Frazier, P.I.: A Tutorial on Bayesian Optimization, https://arxiv.org/abs/1807.02811, (2018)
50. Agarap, A.F.: Deep Learning using Rectified Linear Units (ReLU), https://arxiv.org/abs/1803.08375, (2018)
51. Kingma, D.P., Ba, J.: Adam: A Method for Stochastic Optimization, https://arxiv.org/abs/1412.6980, (2014)
52. Bacciaglia, A., Ciccone, F., Ceruti, A., Peruzzini, M.: 2D Frequency-Based Topological Optimization: Machine Learning to Aid Tuning Simulation Parameters. In: Di Stefano, P., Gherardini, F., Nigrelli, V., Rizzi, C., Sequenzia, G., and Tumino, D. (eds.) Design Tools and Methods in Industrial Engineering IV. pp. 20–29. Springer Nature Switzerland, Cham (2025)
53. Shen, A., Lai, Z., Zhang, L.: Systematic Analysis of Low-Precision Training in Deep Neural Networks: Factors Influencing Matrix Computations. Applied Sciences. 14, 10025 (2024). https://doi.org/10.3390/app142110025

Accepted manuscript

ACCEPTED MANUSCRIPT

54. Dai, W., Yoshigoe, K., Parsley, W.: Improving Data Quality through Deep Learning and Statistical Models. (2018). https://doi.org/10.48550/ARXIV.1810.07132
55. Muir, W.W., Belsley, D.A., Kuh, E., Welsch, R.E.: Regression Diagnostics: Identifying Influential Data and Sources of Collinearity. Journal of the Royal Statistical Society. Series A (General). 144, 367 (1981). https://doi.org/10.2307/2981802
56. Ying, X.: An Overview of Overfitting and its Solutions. J. Phys.: Conf. Ser. 1168, 022022 (2019). https://doi.org/10.1088/1742-6596/1168/2/022022

Accepted manuscript