A hybrid machine learning and evolutionary approach to material selection and design optimization for eco-friendly structures

Luis Yepes Llorente, Joseph Morlier, Saketh Sridhara, Krishnan Suresh

A Hybrid Machine Learning and Evolutionary Approach to Material Selection and Design Optimization for Eco-Friendly Structures

Luis Yepes Llorente[1], Joseph Morlier[2*], Saketh Sridhara[3] and Krishnan Suresh[3]

[1]ISAE-SUPAERO, Université de Toulouse, Toulouse, France.
[2]ICA, Université de Toulouse, ISAE-SUPAERO, MINES ALBI, UPS, INSA, CNRS, Toulouse, France.
[3]Department of Mechanical Engineering, University of Wisconsin-Madison, Madison, USA.

*Corresponding author(s). E-mail(s): joseph.morlier@isae-supaero.fr;
Contributing authors: luisyep@outlook.com; ssridhara@wisc.edu; ksuresh@wisc.edu;

Abstract

In an increasingly competitive and digital industrial environment, the optimization of structures is a key point not only to reduce costs but also to reduce the consumption of natural resources. To this end, different approaches have emerged throughout history based on the tools available at the time. With the current rise of artificial intelligence and the concept of Machine Learning, revolutionary ideas are emerging that allow an optimal dimensioning of structures in record time. This work presents the use of Variational Autoencoders and mixed variable solvers as a proposal for structural optimization and material selection. It has expanded upon previous research by advancing in three directions: (1) Incorporating more material attributes, particularly relevant for environmental considerations. (2) Analyzing in more detail aspects of VAEs such as the dimensionality of the latent space. (3) A two-step hybrid approach to select the optimal candidate: preliminary filtering with VAE and final design via mixed variable model. Various examples demonstrate the applicability of the proposed method.

Keywords: Material selection, Structural design, Optimization, Variational autoencoders (VAE), multi-objective optimization

1 Introduction

The design of structural systems involves selecting the most suitable material and optimizing the geometry [1]. Traditionally, structural designers either choose a material before optimizing the geometry or select the best material for an existing geometry [2], but these approaches do not yield the optimal design of geometry and material selection. Therefore, it is essential to consider both geometry optimization and material selection simultaneously.

The basic problem to understand how both optimization processes are linked is that of, given the configuration of a truss, choosing the area of each of its members while deciding which material to use in order to minimize a function such as the total weight of the structure. This problem can be posed mathematically as [3]:

$$\begin{aligned} \min_{A_i \in \mathbf{A}, m \in M} \quad & \psi(A_1, \ldots, A_N, \zeta_m) \\ \text{subject to} \quad & \mathbf{K}(A_1, \ldots, A_N, E_m)\mathbf{u} = \mathbf{f} \\ & g(A_1, \ldots, A_N, \zeta_m) \leq 0 \\ & A_{min} \leq A_i \leq A_{max} \end{aligned} \tag{1}$$

where ψ is the chosen objective function such as structure mass, cost or CO_2 emissions. The set of constraints (such as yield stress or buckling reserve factor) are denoted by g, $\mathbf{f}$ is the set of forces applied to the truss, $\mathbf{K}$ is the truss stiffness matrix, and $\mathbf{u}$ are the nodal displacements. The design degrees of freedom are each of the truss members areas $\mathbf{A} = \{A_1, A_2 \ldots A_N\}$, and the material of choice m picked from a database M. Each material m would have associated a set of relevant properties denoted as ζ_m such as Young's Modulus E_m. The relevant material properties are listed in Table 1. These properties of the material m are collected inside the variable denoted ζ_m.

It is important to note that the two sets of design variables are closely interconnected, meaning that it is not optimal to just select a material and then optimize cross-sectional areas or vice versa. Simply following either of these approaches does not guarantee the optimal combination of geometry and material [4]. In addition, while cross-sectional areas continuously vary, material selection is a discrete process, which makes it difficult to solve using traditional gradient-based optimization methods. Various non-gradient approaches have been proposed for the purpose of material selection, as

Name	Property	Units
E_m	Young's modulus	Pa
C_m	Cost per unit mass	\$/kg
ρ_m	Mass density	kg/m^3
Y_m	Yield strength	Pa
P_m	CO_2 produced per unit mass	kg/kg
V_m	Energy required per unit mass	J/kg
W_m	Water required per unit mass	L/kg

Table 1: Material properties

evidenced by the literature [5],[6],[7]. However, the incorporation of these methods into gradient-based optimization is precluded.

Mathematically, the problem of material selection and area optimization can be formulated as mixed-discrete nonlinear programming problems (MDNLPs) [8], which are common in engineering. Several methods have been proposed to solve these problems, but they require repeated solutions of a sequence of nonlinear programming problems with relaxations and approximations, making them sensitive to assumptions and underlying models.

A recent study [3] was carried out and can be viewed as a pioneering work in this field. In it, the use of Variational Autoencoders (VAEs) was proposed to solve this type of problems. VAEs are a special form of neural networks that can convert discrete data, such as a material database, into a continuous differentiable space, allowing gradient-based optimization operations to be performed. Other related problems have been already solved:

- with geometry and categorical variables [9, 10]
- with topology and discrete material variables for eco-design for optimal digital manufacturing [11, 12]
- with mixed continuous and discrete variables for HALE design [13].

A very recent paper [14] also tackle the problem of truss topology optimization of timber–steel structures for reduced embodied carbon design.

This work presents the use of Variational Autoencoders and mixed variable models as a proposal for structural optimization and material selection. The hybrid approach is accomplished by means of the open source framework for multi-objective optimization in Python ***pymoo*** [15] to obtain with greater accuracy the optimal solution by means of a mixed use of the VAE and an optimization algorithm using integer variables [16].

The work will be focused mainly in two test cases; mass versus cost and cost versus CO_2, but the available open-source code on GitHub [17] can easily treat any other interesting Pareto front such as mass versus compliance, CO_2

versus energy, etc. See Table B2 for the list of objective functions available by default.

1.1 Contributions of this paper

This paper is an extension of the authors' previous work [3]. It has expanded upon previous research by:

1. Incorporating more material attributes, particularly relevant for environmental considerations.
2. Analyzing the dimensionality of the latent space, taking the three-dimensional case as a sample. Likewise, delving into aspects such as the number of neurons or the loss function.
3. Introducing the new confidence constraint on the optimizer to steer far from vacant spaces in the latent space.
4. Analyzing different applications and putting more emphasis in the comparison of resolution times with a non-VAE method such as the mixed variables model.
5. A two-step hybrid approach to select the optimal candidate: preliminary filtering with VAE and final design via mixed variable model.

2 Theoretical background

This section will present a brief summary of how a variational autoencoder can be used to obtain a continuous and differentiable representation of a discrete material database.

2.1 Variational Autoencoders

A Variational Autoencoder (VAE) is a type of deep neural network that can learn to represent high-dimensional data in a lower-dimensional latent space. The VAE architecture is composed of two components: an encoder network that maps input data to a lower-dimensional latent space, and a decoder network that maps the latent space back to the original data space [18]. The key innovation of VAEs is the use of variational inference to train the network, which allows for the generation of new data samples by sampling from the latent space.

VAEs share similarities with principal component analysis (PCA) as they both extract relevant information from data. However, VAEs' nonlinear characteristics enable them to generalize much better than PCA, as stated in [19]. VAEs have been used as generative models in various scientific applications such as material design [**?**] [**?**], microstructure [**?**] and process-property relationships [**?**]. In this paper, we do not leverage the generative property of the latent space to synthesize new data. Instead, we leverage the two main concepts of nonlinear dimensionality reduction and continuity of the latent

space [19] to represent each material, along with its associated properties, in a continuous and differentiable latent space as first demonstrated in [3].

Specifically, assuming a default two-dimensional latent space (z_0, z_1) and a seven-dimensional feature space ζ_m, we will construct a function $\hat{\zeta} = [\hat{E}, \hat{C}, \hat{\rho}, \hat{Y}, \hat{P}, \hat{J}, \hat{W}] = D(z_0, z_1)$ that maps each point in the latent space to a set of properties. Note that the symbol $\hat{}$ over the properties is used to state that it is a decoded property. The main objective of the VAE is to make that each point in the latent space associated with the materials introduced into the encoder produce properties as close as possible to the input when decoded in the decoder. Figure 1 depicts the VAE architecture proposed for capturing material properties, which comprises the following components:

1. A seven-dimensional input corresponding to the seven properties listed in Table 1.
2. An encoder consisting of a fully-connected 250 neurons network
3. A two-dimensional latent space, denoted by z_0, z_1, where materials are distributed following a Gaussian distribution $z \sim \mathcal{N}(\mu = 0, \sigma = 1)$.
4. A decoder, with a structure similar to the encoder, consisting of a fully-connected network of 250 neurons.
5. A seven-dimensional output corresponding to the original seven properties.

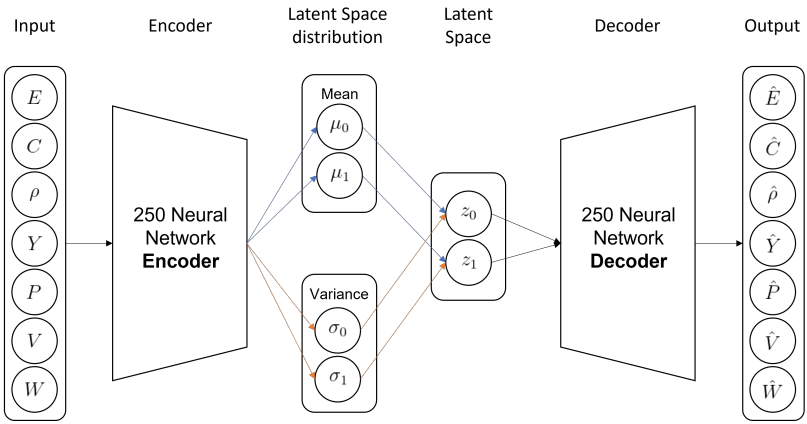


Fig. 1: Architecture of the Variational Autoencoder

It should be noted that the number of neurons has been chosen based on the size and variety of the database. A higher number of neurons would have forced a more complex and input-adjusted composition of the latent space, reducing the generative capacity. A comparison of the composition of the latent space can be seen in Figures A1a and A1b, showing a less smooth gradient but better grouping of materials. In any case, the influence of this

parameter is minimal.

The training of the neural network is done by minimizing the difference between the input and output, denoted as $\|\hat{\zeta} - \zeta_m\|$, and imposing a KL divergence loss to ensure that the latent space follows a standard Gaussian distribution $z \sim \mathcal{N}(\mu = 0, \sigma = 1)$ [20]. The total loss function is given by the following equation:

$$L = \|\hat{\zeta} - \zeta_m\| + \beta \cdot KL(z\|\|\mathcal{N}) \tag{2}$$

where β is set to a recommended value of 5e-5. Therefore, although we would like to keep the original value of the properties as much as possible by minimizing the difference between the input and output, it is necessary to not take a too low value of β in order not to obtain an unbalanced distribution. A comparison of the composition for different values of β is shown in Figures A2a and A2b, and the error is summarized in Table B1.

To ensure that all material properties are weighted equally, the input is scaled between (0, 1), and the output is re-scaled back after training. The structure of the VAE was modeled using the library PyTorch [21] and the minimization of Equation (2) was carried out by the gradient-based Adam optimizer [22].

Figure 2 shows the result for a two-dimensional latent space considering the 7 properties presented previously. The training was set to 50000 iterations based on convergence evolution and took 186 seconds to complete in a Ryzen 2500U CPU.

The latent space contains a coherent distribution of the materials resembling a normal distribution centered at the origin of coordinates. Each of the coordinates, z_0 and z_1, represent a position within the latent space, but have no physical meaning as such. For example, A286 Iron is associated with coordinates (1.4,-0.4), and 2219-T62 with coordinates (0.4,0.3), being this representation unique and unambiguous. In a conceptual sense, it resembles an Ashby chart [2]. The main difference with respect to a material chart is that, firstly, the coordinates do not represent a specific physical quantity but rather dimensionless parameters, and secondly, not only are the materials arranged on a map, but any point on the map is associated with the set of properties $\hat{\zeta} = [\hat{E}, \hat{C}, \hat{\rho}, \hat{Y}, \hat{P}, \hat{J}, \hat{W}] = D(z_0, z_1)$.

To better identify each group of materials, a convex hull was drawn around the material points for each of the associated material types.

It should be noted that there is the possibility of generating a higher dimensional latent space, which can be interesting in certain situations. However, the dimensionality of the latent space will be discussed later.

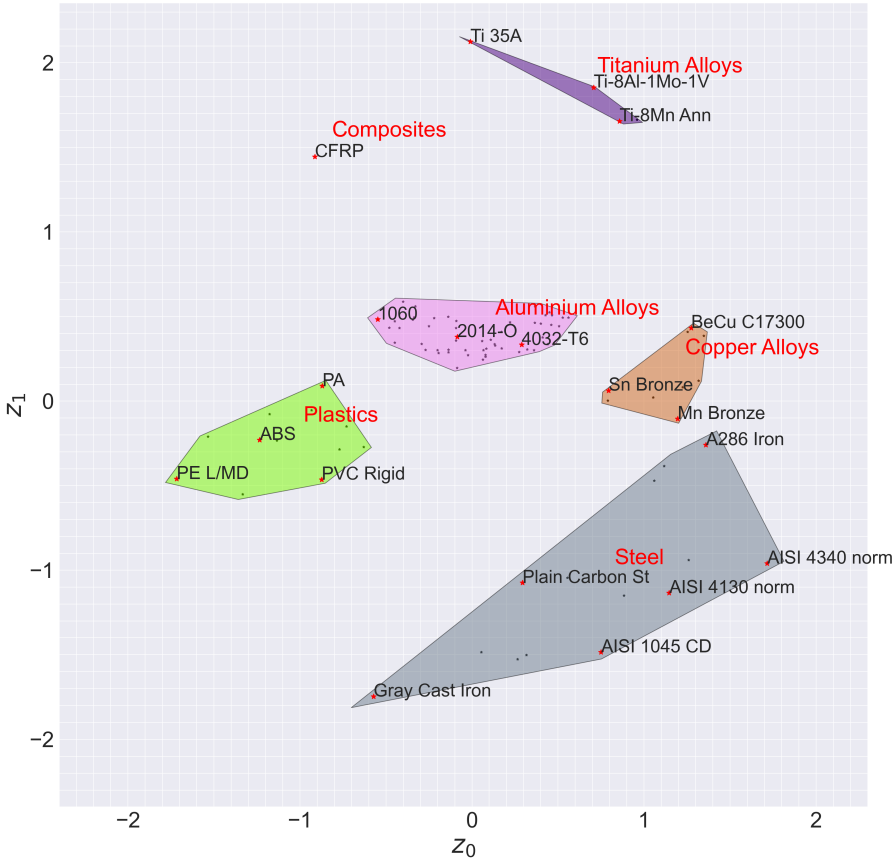


Fig. 2: Material representation in a two-dimensional latent space (7 properties considered)

At intermediate points in the latent space, mixed properties will be obtained that are a blend of nearby materials based on their proximity. By using VAEs to learn this mapping, we can achieve a powerful tool for materials discovery and property prediction. The latent space can be used to explore the relationships between different materials and their properties, and to generate new materials with desired properties by sampling from it. This framework has the potential to greatly accelerate the discovery of new materials with desired properties, particularly in applications where experimentation is expensive or time-consuming.

2.2 Application to material selection

The continuity and differentiability of the latent space are crucial aspects. Through the decoder, analytic activation functions represent each output

material property. For example, the output of the yield strength for a set of coordinates in the latent space is given by $\hat{Y} = D^*_Y(z_0, z_1)$. Hence, one can back propagate through the decoder to compute analytical sensitivities such as $\frac{\partial \hat{Y}}{\partial z_0}$ so that a gradient-based optimizer can be employed.

Figures 3 and 4 show the latent space (figure 2) with a colormap of the values of CO_2 and cost per kilogram at each point in the latent space, illustrating the continuous and coherent gradients along the space.

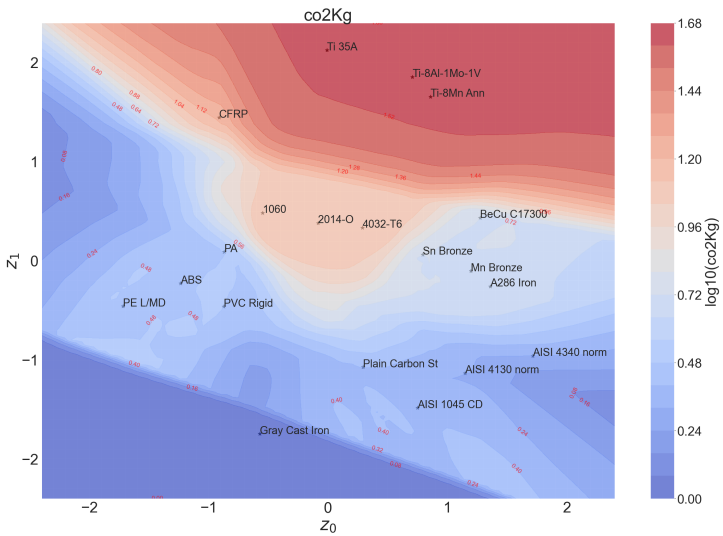


Fig. 3: CO_2 per kilogram colormap over the latent space

Another important feature of this space is that it is possible to measure distance between points. In this way, materials with the most diverse properties will have large distances from each other, and the most similar ones will appear closer together. Note that the coordinates solution in the latent space chosen by the optimizer to minimize a given function will not correspond in general to any particular material. In this case, this distance will be used to determine the real material closest to that point and thus take it as the real optimal solution.

The main disadvantage of passing through the latent space is that the output properties do not exactly match the ones of the input. That is, being Y the original value of Yield strength of a given material, and $\hat{Y}$ the decoded value, we define the Yield strength error as $\Delta Y\% = 100 \cdot \frac{\|Y - \hat{Y}\|}{Y}$. See Table 1 for the complete list of properties.

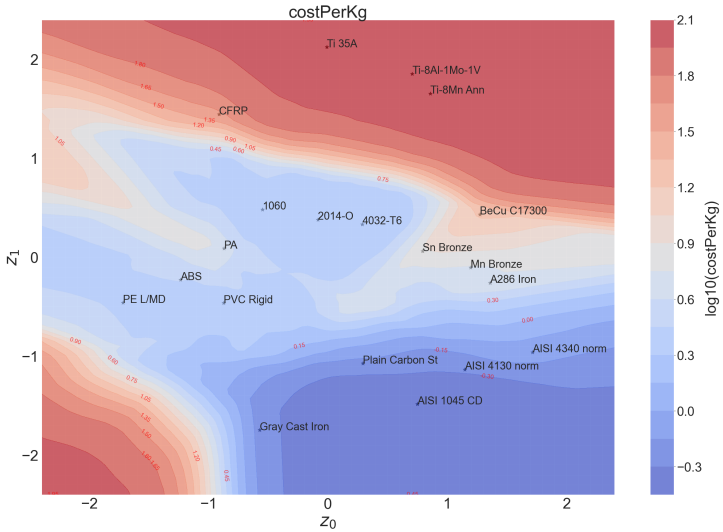


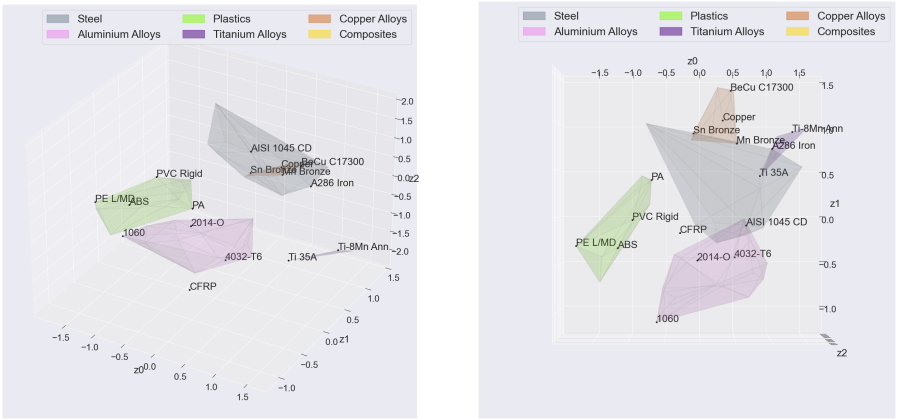
Fig. 4: Cost per kilogram per kilogram colormap over the latent space

Table 2 summarizes the error of all the properties of some materials as well as the maximum and average error present among all materials.

Material	$\Delta E\%$	$\Delta\rho\%$	$\Delta C\%$	$\Delta Y\%$	$\Delta P\%$	$\Delta V\%$	$\Delta W\%$
A286 Iron	7.7	2.3	1.4	2.5	0.1	2.6	4.0
1060	0.4	1.2	0.3	2.2	0.2	0.0	0.3
ABS	1.4	1.0	0.2	0.7	0.6	0.3	0.9
Ti-8Al-1Mo-1V	0.0	0.4	1.2	3.1	0.5	0.5	0.1
Copper	0.6	1.3	2.8	7.8	0.4	0.3	0.9
CFRP	1.2	2.3	0.4	3.6	2.1	2.6	0.2
Max Error	**7.7**	**6.3**	**3.3**	**7.8**	**2.1**	**2.6**	**4.0**
Avg Error	**1.6**	**1.6**	**0.7**	**2.2**	**0.3**	**0.3**	**0.6**

Table 2: Percentage error between actual and decoded data for a two-dimensional latent space with 7 material properties

It should be noted that the discrepancy at decoding is not the only problem with this representation. How the materials are distributed in the space greatly influences the stability and convergence of the solution. Thus, an appropriate latent space will present a continuous and smooth gradient in its domain, while a less appropriate one will present abrupt changes and big gaps among the materials. In fact, increasing the dimension of the latent space, as shown in Figures 5a and 5b, allows a more effective distribution of the materials. This is the reason why, in case of considering many properties and materials, it is interesting to increase the dimensionality of the latent space.



(a) Three-dimensional latent space (isometric view)

(b) Three-dimensional latent space ($z_0 - z_1$)

Fig. 5: Projections of the three-dimensional latent space

Note that, as the disparity between materials increases, the composition of the latent space becomes more difficult as there are larger differences between them. Table 3 shows a summary of the average and maximum error when training the VAE with a database reduced to a subset of the materials, as well as when the encoding is made in a three-dimensional latent space. An important conclusion therefore is that if the material type to be used is known in advance, it is much more efficient to eliminate unsuitable materials from the database.

Subset	Error type	$\Delta E\%$	$\Delta\rho\%$	$\Delta C\%$	$\Delta Y\%$	$\Delta P\%$	$\Delta V\%$	$\Delta W\%$
All (93)	**Max Error**	**7.7**	**6.3**	**3.3**	**7.8**	**2.1**	**2.6**	**4.0**
	Avg Error	**1.6**	**1.6**	**0.7**	**2.2**	**0.3**	**0.3**	**0.6**
All [3D] (93)	**Max Error**	**8.4**	**11.4**	**5.6**	**7.5**	**2.6**	**2.9**	**4.3**
	Avg Error	**1.6**	**2.2**	**0.8**	**1.8**	**0.5**	**0.4**	**0.4**
Steel (14)	Max Error	1.1	2.7	0.1	2.1	0.5	0.1	0.9
	Avg Error	0.2	0.6	0.0	0.6	0.1	0.1	0.2
Aluminium (53)	Max Error	0.1	0.2	0.1	2.8	0.0	0.0	0.0
	Avg Error	0.0	0.1	0.0	1.1	0.0	0.0	0.0
Plastic (12)	Max Error	4.0	0.6	0.3	2.1	0.0	0.0	0.0
	Avg Error	0.8	0.2	0.2	0.5	0.0	0.0	0.0

Table 3: Comparison of decoding error for different dimensions and subsets of materials

In our particular case, increasing the dimensionality of the latent space didn't show a reduction in the decoding discrepancy nor improved the convergence of the solution. In fact, it added one degree of freedom, making the convergence less stable. For this reason, a two-dimensional latent space will be used in this paper with coordinates z_0 and z_1.

2.3 Geometry optimization

Once a continuous and differentiable representation of the physical properties of the materials is available, it becomes possible to reformulate the optimization problem mentioned initially to use gradient-based optimization algorithms. In this problem, the variables involved $\mathbf{A} = \{A_1 \dots A_N\}, z_0, z_1$ are the continuous design variables. Therefore, Equation (1) is turned into Equation (3):

$$\begin{aligned} \min_{A_i \in \mathbf{A}, z_0, z_1} \quad & \psi(A_1, \dots, A_N, z_0, z_1) \\ \text{subject to} \quad & \mathbf{K}(A_1, \dots, A_N, z_0, z_1)\mathbf{u} = \mathbf{f} \\ & g(A_1, \dots, A_N, z_0, z_1) \leq 0 \\ & A_{min} \leq A_i \leq A_{max} \\ & z_{0_{min}} \leq z_0 \leq z_{0_{max}} \\ & z_{1_{min}} \leq z_1 \leq z_{1_{max}} \end{aligned} \tag{3}$$

It is important to establish appropriate limits on the values of the variables in order to limit the solution, avoiding absurd or undesired solutions, as well as to facilitate the work of the solver. The complete list of available objective functions and constraints implemented in the code is summarized in Tables B2 and B3.

2.4 Optimizer

The problem described above is a constrained optimization problem. In the case of multi-objective optimization problem, finding the optimal solution is not straightforward as there may be multiple trade-offs between the objectives considered. The solution is expressed not by a single optimal value but by a set of values known as Pareto front, that represents the set of solutions where no other solution exists that is better in both objectives simultaneously.

As stated before, pymoo will be used to solve the constrained optimization problem (single or multi-objective). Regarding the algorithm, NSGA-II will be used. NSGA-II (Non-dominated Sorting Genetic Algorithm II) is a widely used optimization algorithm for multi-objective problems [23]. The NSGA-II algorithm achieves the solution by using a non-dominated sorting procedure,

where the solutions are sorted into different levels of non-domination, with the first level being the set of non-dominated solutions. The algorithm then assigns a fitness value to each solution, based on its level of non-domination and its distance from the other solutions in the same level. This ensures that the solutions on the Pareto front are well-distributed and cover a wide range of trade-offs between the different objectives. NSGA-II is a good choice for both single and multi-objective optimization problems, and therefore it will be the default algorithm used in both cases.

2.5 Solving process and post-processing

Being the material database the input training data, the procedure begins with training the VAE as described in **Algorithm 1**. The encoder F takes in the set of material data ζ and encodes the seven-dimensional data to the two-dimensional latent space denoted by z_0, z_1 through a probabilistic latent distribution governed by μ, σ. The loss function (2) is then used to drive the training. Once finished, the decoder D is retained for material selection taking the latent space coordinates as input and returning the expected material properties $\hat{\zeta}$.

The next part of the procedure is to optimize the structure, described in **Algorithm 2**. Once the distribution of truss nodes, connectivity and boundary conditions in terms of displacements and forces are loaded, the first task is to compute the global stiffness matrix based on the areas and material properties (previously decoded from a sample from the latent space) provided by the variables involved. Then, the multiple objectives and constraints are computed, and based on them, several iterations provide the optimal set of solutions.

Algorithm 1 Encode Materials

procedure MatEncode(ζ , F, D) ▷ Input: Training data, encoder and decoder
  epoch $= 0$ ▷ iteration counter
  repeat ▷ VAE training
    $F(\zeta) \rightarrow \{\mu, \sigma\}$ ▷ Forward prop. encoder
    $\{\mu, \sigma\} \rightarrow z$ ▷ Reparameterization [20]
    $\{\mu, \sigma\} \rightarrow KL(z\mathcal{N})$ ▷ KL loss
    $D(z) \rightarrow \hat{\zeta}$ ▷ Forward prop. decoder
    $\{\zeta, \hat{\zeta}, KL\} \rightarrow L$ ▷ VAE Loss
    $\boldsymbol{w} + \Delta\boldsymbol{w}(\nabla L) \rightarrow \boldsymbol{w}$ ▷ Update VAE weights
    epoch $++$
  until error is acceptable ▷ Iterate
  return D ▷ Trained decoder
end procedure

Algorithm 2 Truss Optimization

procedure TRUSSOPT(truss, loads, restraints, D^*, ...) ▷ Input: Truss, trained decoder, constraints
  k = 0 ▷ iteration counter
  repeat ▷ Optimization
    $NN(\boldsymbol{w}) \rightarrow \{\boldsymbol{A}, \boldsymbol{z}\}$ ▷ Fwd prop Neural Network NN; compute truss areas and latent coordinates
    $D^*(\boldsymbol{z}) \rightarrow \hat{\boldsymbol{\zeta}}$ ▷ Fwd prop D^*; compute material properties
    $\{\boldsymbol{A}, \hat{E}\} \rightarrow [K]$ ▷ compute stiffness matrix
    $\{[K], \boldsymbol{f}\} \rightarrow \boldsymbol{u}$ ▷ State Equation FEA
    $\{\hat{\boldsymbol{\zeta}}, \boldsymbol{A}, \boldsymbol{u}, \boldsymbol{f}\} \rightarrow \{\psi, \boldsymbol{g}\}$ ▷ Objective function and constraints
    $AD(\psi \leftarrow \boldsymbol{w}\ ,\ \boldsymbol{g} \leftarrow \boldsymbol{w}) \rightarrow \Delta\boldsymbol{w}$ ▷ Auto diff. for sens. w.r.t. NN
    $\boldsymbol{w} + \Delta\boldsymbol{w} \rightarrow \boldsymbol{w}$ ▷ Update weights
    k++
  until $\Delta\boldsymbol{w} < \epsilon^*$ ▷ Check for convergence
  return $\{\boldsymbol{z}, \boldsymbol{A}\}$
  Find nearest material and optimize for area ▷ Post processing
end procedure

Note that this solution will involve a set of coordinates z^* in the latent space not corresponding to any real material z_m in particular. It is then necessary to choose a material and recalculate the areas to get the real optimal solution.

For this purpose, a confidence metric is defined. This value is defined as a scale from 0 to 100 considering a ratio between the Euclidean distance with respect to each of the materials and the largest existing distance (4).

$$\gamma_m(z^*, z_m) = 1 - \frac{\|z^* - z_m\|}{\underset{\forall k \in M}{max}(\|z^* - z_k\|)} \tag{4}$$

However, this method is not entirely appropriate. In most cases, and even more so when there are many properties involved and the gradients take on irregular shapes, the solution coordinates will be quite far away from a real material, with the chosen material having a confidence lower than 80. It is in these cases where there is a very high risk of not obtaining the right optimal material as any nearby material could be chosen as optimal by pure chance.

As a solution to this problem we chose to include an additional confidence constraint, described in Equation (5):

$$\gamma_{min} = 1 - \frac{\left(1 - \frac{\|z^* - z_m\|}{\underset{\forall k \in M}{max}(\|z^* - z_k\|)}\right)}{\gamma^*} \tag{5}$$

With this constraint, those solutions that may be feasible but are too far away from real materials are discarded, thus forcing the solver to choose a coherent point in the latent space.

3 Single objective application

An application with a single objective is now discussed before entering the multi-objective approach. By default, besides the confidence constraint, only two constraints will be imposed: the one due to buckling and the one due to yield stress. For this purpose, a conservative safety factor of 4 will be imposed on both. For more information on how these constraints are defined, see Table B3.

This section contains the results obtained using the proposed method. It should be noted that the same truss will be used in all the simulations. This is illustrated in Figure 6 and consists in a two-dimensional mid-cantilever truss composed by 6 truss members. It is embedded at the left part and a force $f_y = -12.5kN$ is applied at the rightmost end.

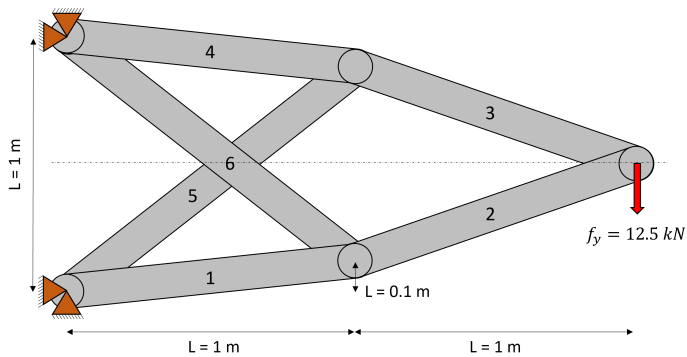


Fig. 6: 6 bar mid-cantilever truss

3.1 Single-objective constrained optimization problem

The first step involves running the solver to obtain a solution that minimizes a specific value while respecting the two default constraints. To this end, the total cost of the structure is decided as the objective to be minimized. The search took 10.56 seconds, and the preliminary results are summarized in Table 4. Hence, the optimal material would be the one placed in the coordinates [0.9, -1.56], which is the point that provides the best material properties for this problem. However, as those coordinates don't correspond to any real material, it is necessary to pick the closest one. Figure 7 illustrates

the optimal point in the latent space, and the closest real materials with their confidence level are summarized in Table 5.

Cost	z_0	z_1	A_1	A_2	A_3	A_4	A_5	A_6
5.92$	0.9	-1.56	1.5e-4 m^2	1.2e-4 m^2	4.5e-4 m^2	4.3e-4 m^2	4.2e-5 m^2	3.3e-4 m^2

Table 4: Preliminary results for minimizing the cost

Material	Confidence (γ_m)
AISI 1045 CD	95.44
AISI 4130	88.99
AISI 4130 norm	86.93

Table 5: Closest material list for minimizing the cost

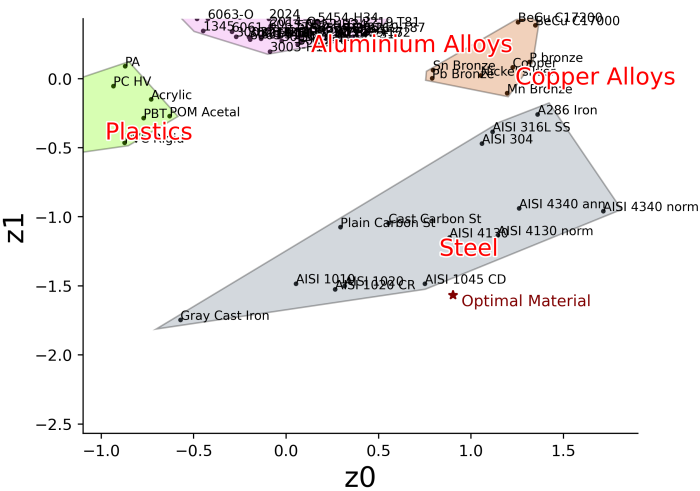


Fig. 7: Solution in the latent space for minimizing the cost of the structure

Hence, AISI 1045 CD is chosen as the optimal material. Its original properties are introduced into the solver, and another calculation is performed, now only varying the areas of the truss members. The calculation took only 2.97 seconds and the results are summarized in Table 6. Therefore, the great

usefulness of this tool and its speed in finding the optimal in the single-objective problem is evident.

Cost	A_1	A_2	A_3	A_4	A_5	A_6
6.37$	1.8e-4 m^2	1.5e-4 m^2	4.4e-4 m^2	4.5e-4 m^2	4.5e-5 m^2	3.7e-4 m^2

Table 6: Final results for minimizing the cost after choosing AISI 1045 CD as material

After performing this operation for all the properties being considered, a summary of the materials that provide the minimum metrics is found in Table 7.

Objective	Material	Value
Cost	AISI 1045 CD	6.371 $
Mass	CFRP	3.45 kg
CO2	AISI 4340 norm	25.84 kg
Water	AISI 1045 CD	727.38 L
Energy	AISI 1045 CD	4.65e08 J
Compliance	Plain Carbon St	0.37 Nm

Table 7: Solutions for the considered objective functions

4 Multi-objective constrained optimization

The general case of multi-objective minimization can be described as:

$$\begin{aligned} \min \quad & f_m(x) && m = 1, .., M \\ \text{s.t.} \quad & g_j(x) \leq 0 && j = 1, .., J \\ & h_k(x) = 0 && k = 1, .., K \\ & x_i^L \leq x_i \leq x_i^U && i = 1, .., N \\ & x \in \Omega \end{aligned} \tag{6}$$

Where $f_m(x)$ can be each $\psi(\mathbf{A}, \zeta_m)$ described in B3, $g_j(x)$ can be any set of constraints (i.e. feasibility constraints such as yield strength and buckling constraints), $h_j(x)$ can be the equilibrium $\mathbf{K}(\mathbf{A}, E_m)\mathbf{u} = \mathbf{f}$ and x_i are bounds

for both continuous and discrete variables $\mathbf{A}$ and ζ_m.

Our framework uses the state of the art in multi-objective optimization pymoo [15] to solve Equation 6. It is worth noting the fact that most of the objectives we are considering (CO_2, cost...) are proportional to the mass by the factor of the associated property (CO_2 per kilogram, $ per kilogram...). Then, for each material, there is only one optimal configuration that minimizes the mass, and this one provides a minimum value of CO2, cost, water... Therefore, the optimal value of each material in these targets translates into a single point in the objective space.

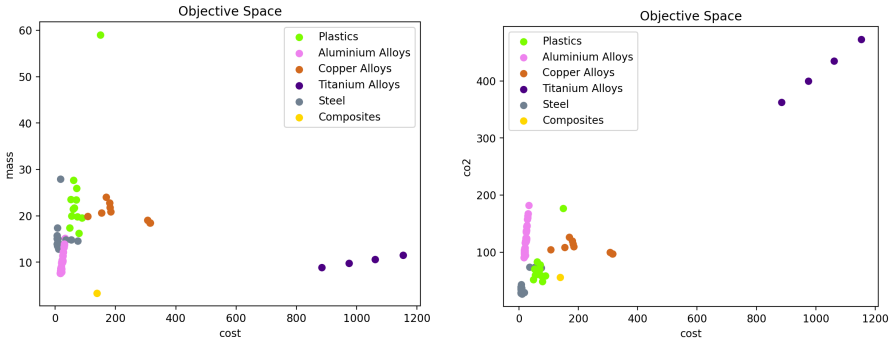
The case of objectives such as compliance is different, since this value decreases as mass increases and vice versa, thus existing a continuous range of optimal values in terms of mass and compliance. There is a minimum area configuration (limited by yield strength and buckling constraints) that guarantees minimum mass but maximum compliance, and a maximum mass configuration limited by the upper limit of the areas that will provide minimum compliance.

In the following sections the solutions for typical bi-objective problems will be analyzed: the problem of minimizing mass and cost (typical in industry), and the problem of minimizing cost and CO_2 (important for the green transition) under the two default structural constraints (yield stress and buckling).

4.1 Multi-objective optimization by brute-force

In order to have a valid and accurate solution with which to compare the solutions obtained by other methods and their computation time, firstly a solution to the problem by brute-force is obtained. This consists of fixing a material and optimizing the structure to obtain the optimal solution. After doing this with all the materials in the database, it is possible to find the optimal solution for each material. This analysis guaranteed an almost exact result for each material. However, it required a really long computational time (on average 320 seconds) which would be higher the greater the number of materials. Figure 8a shows the result of minimizing mass and cost, whereas Figure 8b show the result of minimizing cost and CO_2. Note the above; each material has a single optimal point.

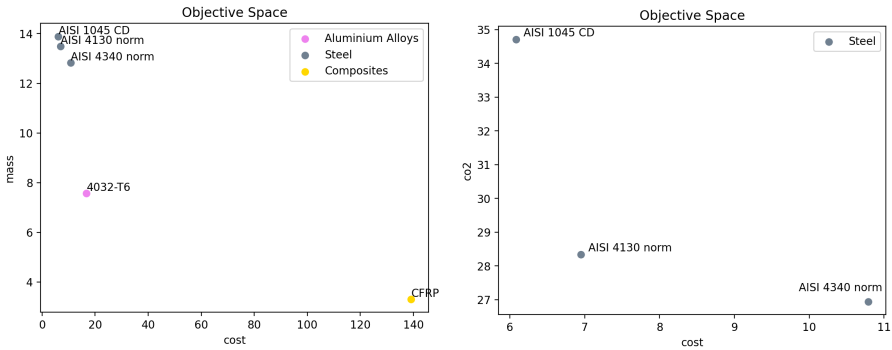
The interesting information about this distribution of solutions is what is known as the Pareto front, that represents the set of solutions where no other solution exists that is better in both objectives simultaneously. The Pareto front of the objectives considered is shown in Figures 9a and 9b.



(a) Mass and cost optimization solution for each material in the database

(b) Cost and CO_2 optimization solution for each material in the database

Fig. 8: Bi-objective solutions obtained by brute-force



(a) Pareto front for mass and cost performed by brute-force

(b) Pareto front for cost and CO_2 performed by brute-force

Fig. 9: Pareto fronts obtained by brute-force

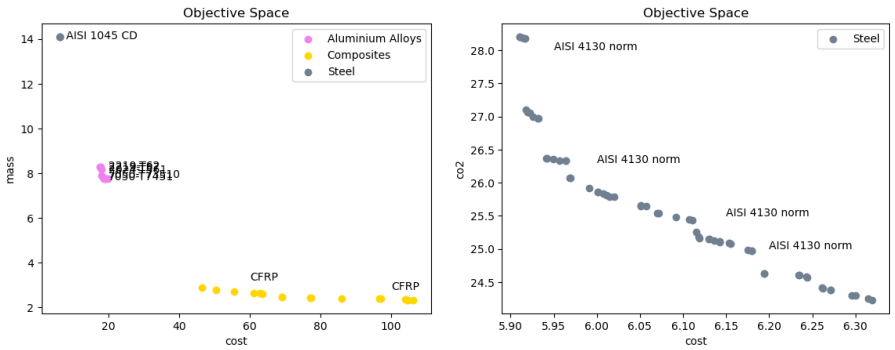
In the following sections we will look for such a solution by using other methods that allow us to obtain it without analyzing all the materials one by one, that is, in much less time.

4.2 Multi-objective optimization with the VAE

This section will present the Pareto front results obtained considering the mass, cost and CO_2 objectives using the VAE. It should be noted that the procedure will be identical, varying the continuous variables z_0, z_1 and the areas, to find the optimum. The material with the greatest confidence value will be taken as the solution and the Pareto front will be generated for the

objectives considered.

Figure 10a shows the result for mass and cost optimization, and Figure 10b shows the result for cost and CO_2 optimization, taking an average of 25 seconds to get the solution.



(a) Pareto front for cost and mass obtained with the VAE

(b) Pareto front for cost and CO_2 obtained with the VAE

Fig. 10: Pareto fronts obtained with the VAE

The most remarkable aspect is that some materials appear multiple times. This is due to the fact that, when exploring the latent space, the properties of the materials in the near environment change so much that multiple versions of the same material can appear and can serve as an optimum in different situations.

Regarding the solution itself, the work carried out by the VAE was exceptional, achieving a really good approximation of the shape of the Pareto front in a short time. However, even though the shape and material type of each region are correct, not all the specific materials are present. Hence, an additional post-processing must be carried out to get the real optimal solution, as the discrepancy in material properties as well as the existence of artificial optimal regions makes impossible for the VAE to find the real optimal solution by itself.

4.3 Multi-objective optimization with the mixed variables model

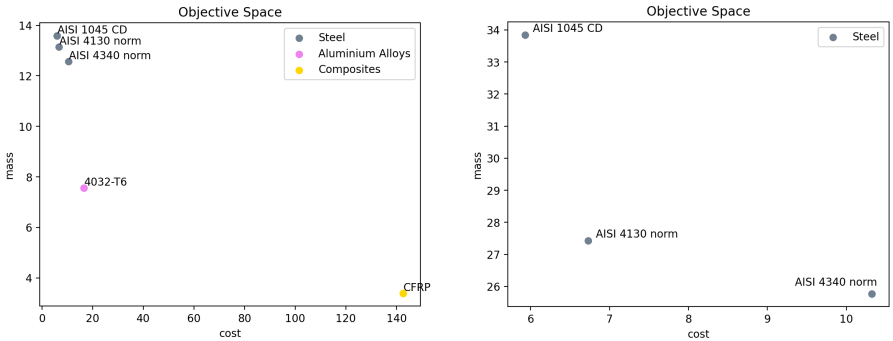
A method similar to brute-force that allows us to obtain the real solution but without having to deeply analyze each material one by one is the Multi-objective Optimization With Mixed Variables model [16]. It involves using a mixed-integer programming (MIP) formulation, where some variables are continuous and others are discrete. In our particular case, the continuous variables will be each of the areas of the truss members, and we will create a new integer variable named material index associated with each of the materials in the database. The NSGA-II algorithm will sweep through all the materials (varying the discrete variable) obtaining the metrics for different solutions of the structure. Thus, those materials that present worse metrics and are farther away from the constraints will be eliminated from the sweep and the solution of the areas of the structure will be refined for the remaining materials, obtaining the Pareto front of the optimal materials in much less time than by brute-force. This procedure is different from the original procedure as the VAE is not involved at any time. The resolution procedure for the mixed variables model is detailed in **Algorithm 3**.

Algorithm 3 Truss Optimization with mixed-variables model

procedure TRUSSOPTMIXED(truss, materials, loads, restraints, ...) ▷ Input: Truss, material database, constraints
  k = 0 ▷ iteration counter
  repeat ▷ Optimization
    $NSGA2(\boldsymbol{w}) \rightarrow \{\boldsymbol{A}, i\}$ ▷ Compute truss areas and material index
    $i \rightarrow \boldsymbol{\zeta}$ ▷ Extract material properties from the database
    $\{\boldsymbol{A}, E\} \rightarrow [K]$ ▷ compute stiffness matrix
    $\{[K], \boldsymbol{f}\} \rightarrow \boldsymbol{u}$ ▷ State Equation FEA
    $\{\hat{\boldsymbol{\zeta}}, \boldsymbol{A}, \boldsymbol{u}, \boldsymbol{f}\} \rightarrow \{\psi, \boldsymbol{g}\}$ ▷ Objective function and constraints
    $AD(\psi \leftarrow \boldsymbol{w}\ ,\ \boldsymbol{g} \leftarrow \boldsymbol{w}) \rightarrow \Delta\boldsymbol{w}$ ▷ Auto diff. for sens.
    $\boldsymbol{w} + \Delta\boldsymbol{w} \rightarrow \boldsymbol{w}$ ▷ Update weights
    k++
  until $\Delta\boldsymbol{w} < \epsilon^*$ ▷ Check for convergence
  return $\{\boldsymbol{A}, i\}$
end procedure

Figure 11a shows the result for mass and cost optimization, and Figure 11b shows the result for cost and CO_2 optimization. The calculation time was 41 and 39.2 seconds respectively.

It is remarkable how the solution is identical to the one obtained by brute-force but in a much shorter time. However, the computation time is almost twice as long as in the case of using the VAE. This is due to the fact that



(a) Pareto front for cost and mass performed with the mixed variables model

(b) Pareto front for cost and CO_2 performed with the mixed variables model

Fig. 11: Pareto fronts obtained with the mixed variables model

since gradient operations cannot be used and the materials are arbitrarily distributed in the database, it is not possible a priori to focus on a series of materials and all of them must be analyzed.

It was also decided to generate the Pareto front of compliance together with the mass to observe how a continuous front is generated due to the multiplicity of solutions. The result is shown in Figure 12.

Obviously, the material that guarantees the minimum in both properties is the CFRP, and therefore it is the one that dominates the plot. However, from a certain value of truss volume, if it is desired to continue reducing the compliance, it is necessary to make a jump in the total mass of the system and use a denser material such as Gray Carbon.

When the structure using the CFRP reaches the maximum value in the section of the areas, the weight of the structure cannot continue to be increased to continue decreasing the compliance. It is in this case that a material with a higher density but also a higher Young's modulus must be used in order to reduce compliance even more. Note that the values of both mass and compliance do not reach infinite values because they are limited by the value of the areas.

4.4 Multi-objective Optimization with the hybrid model

It has been shown that if the VAE is used directly to obtain the Pareto front, due to imprecision in the properties and lack of convergence, it is difficult to obtain an accurate solution. However, by being able to apply gradient optimization, in a short time, the optimizer is able to find a close solution. On the other hand, if the problem is solved directly using an integer variable

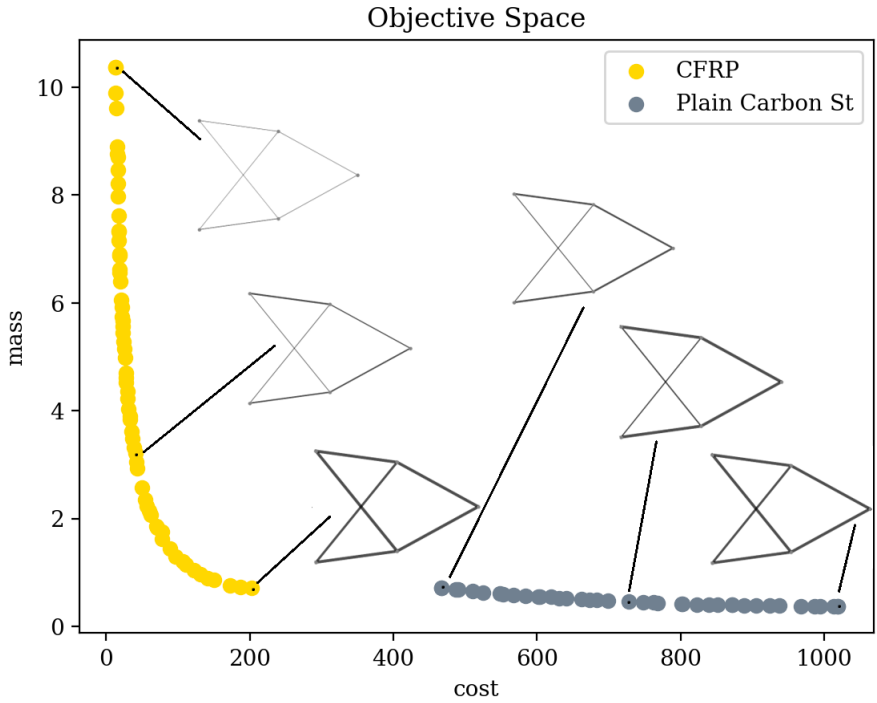


Fig. 12: Pareto front of mass vs compliance, depicting the geometry evolution

that encodes the properties of each material in the database, it is possible to find the exact solution. In addition, the search time will depend very much on the size of the database, which is not the case when using the VAE.

A hybrid method combining the advantages of each approach is proposed. The idea is to use the VAE as a preliminary search for appropriate materials (**Algorithm 1 and 2**), and after applying a filtering based on that solution, perform the optimization with the mixed variables method (**Algorithm 3**) with a proportionally reduced material database. Being conservative, the filter used will be to discard all the materials whose material type is not present in the preliminary solution.

The scheme of the proposed method is shown in Figure 13.

This method will be put into the test for the optimization of mass and cost. First, the preliminary solution using exclusively the VAE was obtained. It took 9.71 seconds and is shown in Figure 14a.

Hence, in this case, only steels, aluminum alloys and composites were kept in the database, reducing it from 93 to 68 materials.

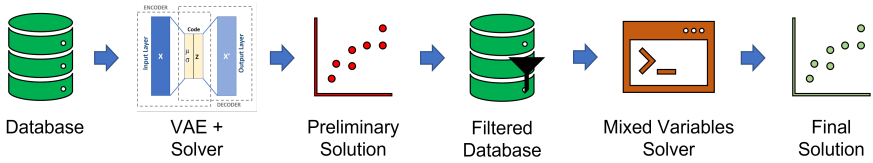
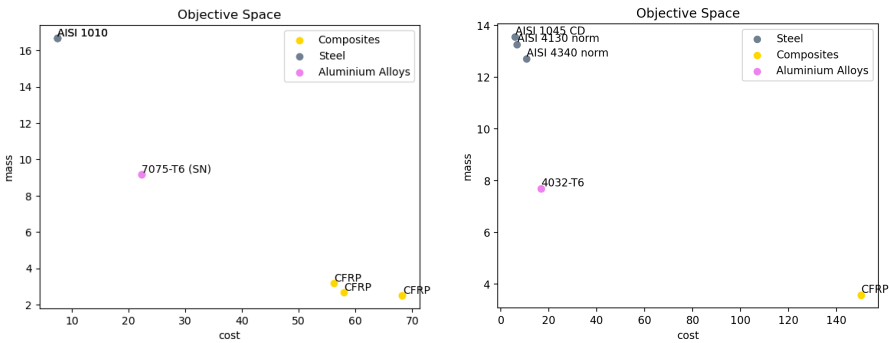


Fig. 13: Scheme of the hybrid model

Once the database has been reduced, the mixed variable method is run, and in this way the final accurate solution is obtained, shown in Figure 14b. It took 19.17 seconds.



(a) Preliminary Pareto front for cost and mass performed with the VAE

(b) Final Pareto front for cost and mass with mixed variables model

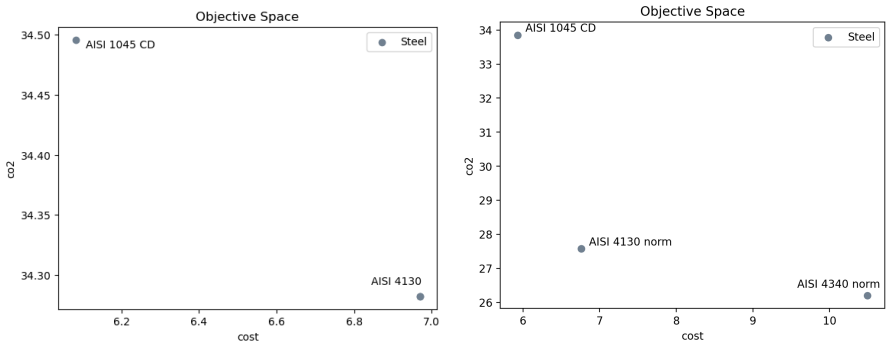
Fig. 14: Cost and mass Pareto front using the hybrid model

This makes the total solution search time (28.88 seconds) equal to or slightly lower than the one of the mixed variables model (41 seconds) with an equally accurate solution.

The same procedure was conducted for CO_2 and cost, obtaining the preliminary solution shown in Figure 15a in 8.76 seconds.

After keeping only the steel in the database (14 materials), the mixed variable method was run generating the solution show in Figure 15b in only 15.72 seconds.

This results in a total solution time of 24.48 seconds. In this last case, as more materials were discarded, the solution time was lower. The calculation times are summarized in Table 8. Note that other results such as the value of the areas or the chosen materials are not compared since they are coincident.



(a) Preliminary Pareto front for cost and CO_2 performed with the VAE

(b) Final Pareto front for cost and CO_2 with mixed variables model

Fig. 15: Cost and CO_2 Pareto front using the hybrid model

The chosen discriminator (material class) is the most conservative. Several preliminary experiments were performed by choosing a different discriminator such as keeping the N materials closest to the preliminary solution or materials with a minimum confidence level. However, this required a more accurate preliminary solution, which increased the time required, and on several occasions some of the optimal materials were not preserved.

Time (s)		Cost vs Mass	Cost vs CO_2
Mixed model		**41**	**39.2**
	VAE	9.71	8.76
Hybrid model	Mixed	19.17	15.72
	Total	**28.88**	**24.48**

Table 8: Total resolution time for the considered cases with hybrid and mixed variables models

4.4.1 Hybrid vs mixed variables model for a complex truss

To further test the model, a similar comparison was performed for a more complex truss, depicted in Figure 16. It is composed by 22 nodes connected by 47 bars, simply supported at the lower ends and subjected to a vertical downward force of 10 kN at each of the higher tips.

The calculation times of each of the models are summarized in Table 9. Figures 17a and 17b show the preliminary and final solution respectively of the Pareto front of cost vs mass, whereas Figures 18a and 18b show the preliminary and final solution respectively of the Pareto front of cost vs CO_2.

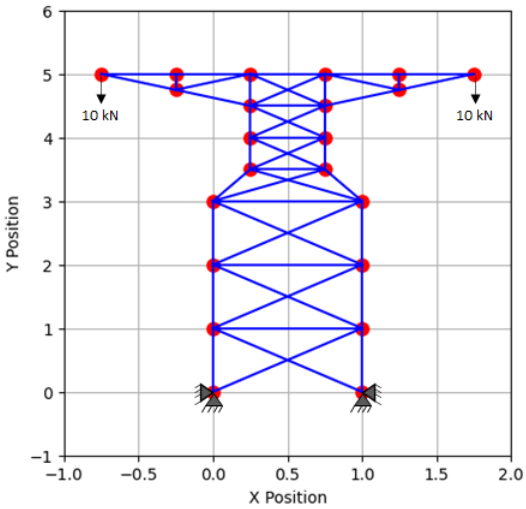


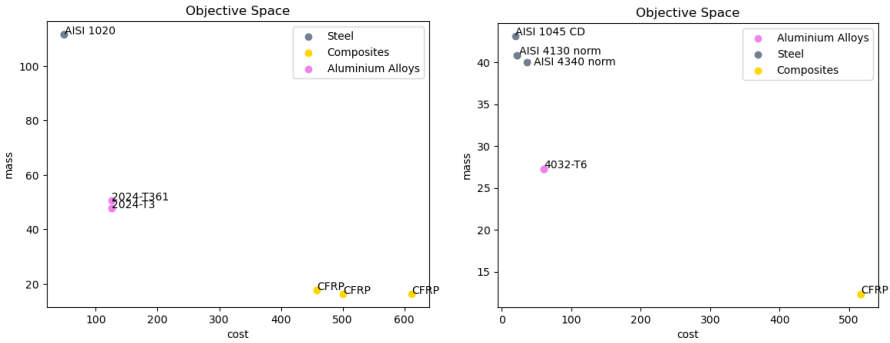
Fig. 16: Antenna truss

Although the structure is different, the resulting materials and their distribution are alike to the previous case. This will be discussed in the following section.

Time (s)		Cost vs Mass	Cost vs CO_2
Mixed model		**112.3**	**117.41**
	VAE	14.32	19
Hybrid model	Mixed	75.7	46.4
	Total	**90.02**	**65.4**

Table 9: Total resolution time for the considered cases with hybrid and mixed variables models (antenna)

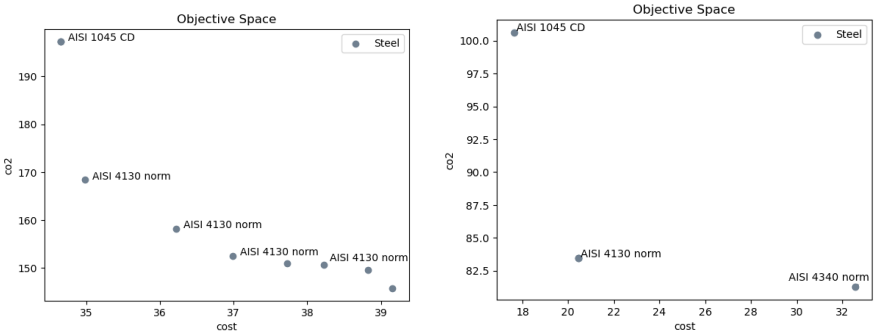
Due to the complexity of the structure, the total calculation time increases. However, it stands out that the preliminary calculation time performed with the VAE remains low despite the complexity of the structure. In addition, the reduction in the total calculation time of the hybrid method for cost vs CO_2 is more evident for this case as the huge reduction of the material database involves a greater impact for complex problems like this one.



(a) Preliminary Pareto front for cost and mass performed with the VAE (antenna)

(b) Final Pareto front for cost and mass with mixed variables model (antenna)

Fig. 17: Cost and mass Pareto front using the hybrid model for the complex antenna truss



(a) Preliminary Pareto front for cost and CO_2 performed with the VAE (antenna)

(b) Final Pareto front for cost and CO_2 with mixed variables model (antenna)

Fig. 18: Cost and CO_2 Pareto front using the hybrid model for the complex antenna truss

5 Practical case: Environmentally friendly three-dimensional truss

In this last section, the hybrid model will be put to the test in a three-dimensional structure with the aim of reducing the impact on the environment. On the one hand it will serve to demonstrate that the method is equally applicable to any structure independently of its complexity, and on the other hand it will demonstrate its usefulness in a real example.

The structure to be analyzed is a three-dimensional 20 bar mid-noded cantilever truss. The geometry is depicted in Figure 19. Regarding the boundary conditions, the truss is simply supported at the left side and a downward force of $20kN$ is applied to the rightmost node. It must be designed in such a way to reduce the impact on the environment. Therefore, the focus will be mainly on the values of CO_2 produced, as well as energy and water consumed. There is no a priori limit in terms of cost or mass. Therefore, we only maintain as usual the buckling and yield stress constraints.

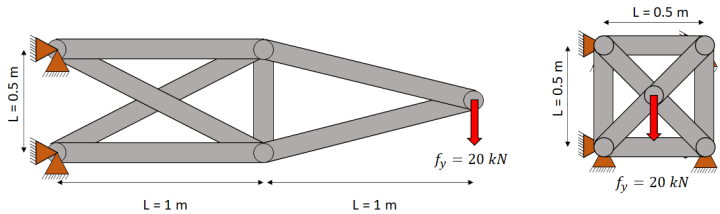


Fig. 19: 20 bar mid-noded cantilever truss

Hence, the first step is to apply the quick search of the solution. It took 26.02 seconds and the results are presented in Figure 20a.

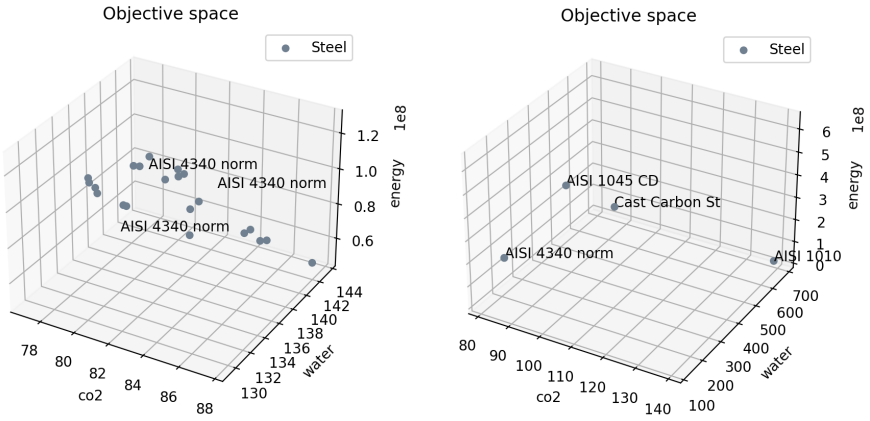
Note that multiple versions of the same material (AISI 4340 norm) appear as solution. This gives some information regarding the final result: this material will be probably one of the best minimizing at the same time all the properties considered, and as a consequence, similar materials (steel alloys) are the only candidates suitable for this problem.

Likewise, the database is filtered to only consider steel alloys and the integer optimization algorithm is run. It took 26.41 seconds and the result is presented in Figure 20b.

As expected, steel AISI 4340 norm is practically the material with less water consumption and CO_2 produced. However, note that AISI 1010 has half of the energy required. In this scenario, it is difficult to decide which material to use to manufacture the structure. To be rigorous, an environmental consumption index will be defined. This index will try to fairly weight the properties so that the most balanced material can be chosen.

Being P_{min}, W_{min} and V_{min} the minimum value of CO_2, water and energy achievable respectively, the index is defined as:

$$H = \frac{1}{3}\frac{P - P_{min}}{P_{min}} + \frac{1}{3}\frac{W - W_{min}}{W_{min}} + \frac{1}{3}\frac{V - V_{min}}{V_{min}} \tag{7}$$



(a) Preliminary Pareto front for minimizing environmental impact

(b) Final Pareto front for minimizing environmental impact

Fig. 20: Pareto fronts of the problem of minimizing environmental impact

In such a way that the perfect material with minimum consumption would yield $H = 0$. To take into account the cost of the structure as well, both values were calculated for each solution and presented in Table 10.

Material	Cost	Env. Consumption
AISI 1045 CD	18.63 $	9.60 [-]
Cast Carbon St	28.68 $	7.85 [-]
AISI 4340 norm	31.91 $	2.28 [-]
AISI 1010	24.04 $	1.73 [-]

Table 10: Cost and environmental consumption values

Hence, the structure with less environmental consumption would be made of AISI 1010. As it is also one of the cheapest solutions, it should be the material chosen. Note that this index wasn't defined by pure chance; by specifying a given weight for each property, one can get a Pareto front balancing this index with another metric (cost, mass) in order to make the selection based on the specific needs.

Finally, one aspect to consider is whether there are major differences between a two-dimensional and a three-dimensional truss when the material database, boundary conditions and shape of the structure are very similar. For this purpose, a mass versus compliance balance analysis was performed as we did for the two-dimensional truss. The results are shown in Figure 21.

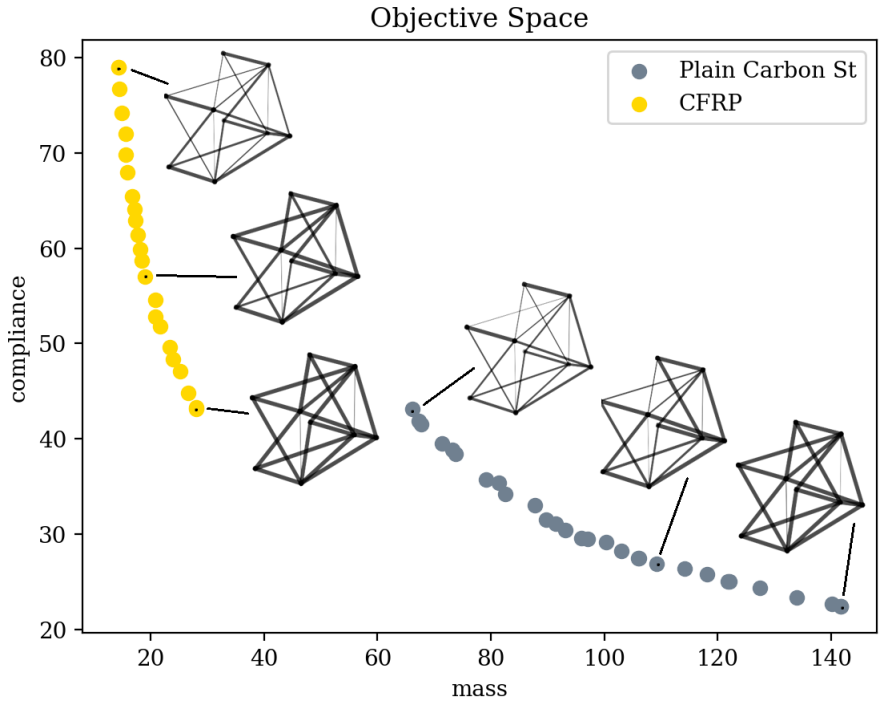


Fig. 21: Pareto front of mass vs compliance, depicting the geometry evolution (three-dimensional case)

Although the result is completely different in terms of values since the problem itself is not the same, it is obvious that the resulting materials and their distribution is very alike to the two-dimensional case. Therefore, we could say that similar problems have similar materials as optimal solution.

6 Conclusions

Throughout this paper, an existing idea of using neural networks to optimize structures has been developed and improved. As introduced at the beginning, the optimization of structures is not at all a new topic, but in recent years new revolutionary ideas such as the one presented in this work related to artificial intelligence are emerging showing promising results.

Although the VAE has proved to be very useful and fast in obtaining the solution, the impossibility of avoiding errors associated with the encoding and the difficulty of having a fully regular map has been a significant enough drawback to require an additional procedure.

At the expense of achieving an accurate solution by means of VAEs, an alternative approach was proposed that combined the velocity of VAEs with the accuracy of the mixed variables model. For our particular case, this was a good trade-off. It was proposed to simply filter by material type, but much more advanced filtering could have been applied to reduce even more the database.

However, this is only a stopgap in order to obtain a solution to this simple problem. Nowadays, with more than 150,000 materials available [2] and increasing in number due to new discoveries it is completely unrealistic to pretend to use a mixed variables model to solve the problem. While the proposal to use VAEs to discriminate materials is viable, efforts should be made to improve the encoding and decoding of materials to directly obtain a fast and accurate solution without relying on methods involving integer variables. The idea of a dynamic encoding, in which only the properties to be used in the problem are encoded in each case, would be a first step to simplify the process and therefore reduce errors.

On the other hand, it would also be interesting to apply a first filter to the encoding of the materials themselves. That is, once the problem is set, discard those materials that a priori are not going to be a solution, and start the encoding process only with the remaining materials. In any case, these are proposals whose objective is to simplify the information to be encoded as much as possible, since inevitably the greater the complexity, the greater the associated error.

Replication of Results

https://github.com/mid2SUPAERO/HybML-EvoMatDesEco

References

[1] Rudy Eggert. *Engineering design* (Pearson/Prentice Hall, 2005).

[2] Michael F. Ashby. *Material selection in mechanical design* (Butterworth-Heinemann, 2011).

[3] Aaditya Chandrasekhar, Saketh Sridhara & Krishnan Suresh. Integrating material selection with design optimization via neural networks. *Engineering with Computers* **38** (5), 4715–4730 (2022) .

[4] Sourav Rakshit & GK Ananthasuresh. Simultaneous material selection and geometry design of statically determinate trusses using continuous optimization. *Structural and Multidisciplinary Optimization* **35**, 55–68 (2008) .

[5] A Jahan, Md Yusof Ismail, SM Sapuan & F Mustapha. Material screening and choosing methods–a review. *Materials & Design* **31** (2), 696–705 (2010) .

[6] R Venkata Rao. A material selection model using graph theory and matrix approach. *Materials Science and Engineering: A* **431** (1-2), 248–255 (2006) .

[7] Chang-Chun Zhou, Guo-Fu Yin & Xiao-Bing Hu. Multi-objective optimization of material selection for sustainable products: artificial neural networks and genetic algorithm approach. *Materials & Design* **30** (4), 1209–1215 (2009) .

[8] Mohit Tawarmalani & Nikolaos V Sahinidis. A polyhedral branch-and-cut approach to global optimization. *Mathematical programming* **103** (2), 225–249 (2005) .

[9] Pierre-Jean Barjhoux, Youssef Diouane, Stéphane Grihon, Dimitri Bettebghor & Joseph Morlier. A bi-level methodology for solving large-scale mixed categorical structural optimization. *Structural and Multidisciplinary Optimization* **62** (1), 337–351 (2020) .

[10] Pierre-Jean Barjhoux, Youssef Diouane, Stéphane Grihon & Joseph Morlier. An outer approximation bi-level framework for mixed categorical structural optimization problems. *Structural and Multidisciplinary Optimization* **65** (8), 214 (2022) .

[11] Edouard Duriez, Joseph Morlier, Catherine Azzaro-Pantel & Miguel Charlotte. Ecodesign with topology optimization. *Procedia CIRP* **109**, 454–459 (2022) .

[12] Edouard Duriez, Catherine Azzaro-Pantel, Joseph Morlier & Miguel Charlotte. A fast method of material, design and process eco-selection via topology optimization, for additive manufactured structures. *Cleaner Environmental Systems* **9**, 100114 (2023) .

[13] Edouard Duriez, Víctor Manuel Guadaño Martín & Morlier Joseph. Co2 footprint minimization of solar-powered hale using mdo and eco-material selection. *Scientific Reports* **13** (1), 11994 (2023) .

[14] Ernest Ching & Josephine V Carstensen. Truss topology optimization of timber–steel structures for reduced embodied carbon design. *Engineering Structures* **252**, 113540 (2022) .

[15] Julian Blank & Kalyanmoy Deb. Pymoo: Multi-objective optimization in python. *IEEE Access* **8**, 89497–89509 (2020) .

[16] Jamie A Manson, Thomas W Chamberlain & Richard A Bourne. Mvmoo: Mixed variable multi-objective optimisation. *Journal of Global Optimization* **80** (4), 865–886 (2021) .

[17] Github code: Structural material selection using deep learning. https://github.com/mid2SUPAERO/HybML-EvoMatDesEco. Accessed: 2023-08-16.

[18] Yuri Burda, Roger Grosse & Ruslan Salakhutdinov. Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519* (2015) .

[19] Ian Goodfellow, Yoshua Bengio & Aaron Courville. *Deep learning* (MIT press, 2016).

[20] Diederik P Kingma, Max Welling *et al.* An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning* **12** (4), 307–392 (2019) .

[21] Adam Paszke *et al.* Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* **32** (2019) .

[22] Diederik P. Kingma & Jimmy Lei Ba. Adam: A method for stochastic optimization. *ICLR 2015 - Conference Track Proceedings. International Conference on Learning Representations* (2015) .

[23] Nsga-ii: Non-dominated sorting genetic algorithm. https://pymoo.org/algorithms/moo/nsga2.html. Accessed: 2023-07-29.

Acknowledgements

The authors would like to acknowledge the assistance from Associate Professor Israel G. García, Departamento de Mecánica de Medios Continuos y Teoría de Estructuras, E.T.S. Ingeniería, Universidad de Sevilla, Camino de los Descubrimientos s/n,41092, Seville, Spain

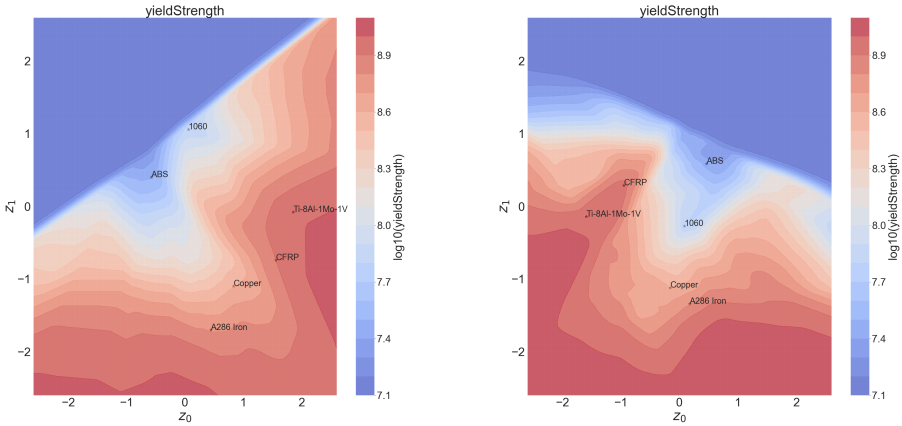
Author contributions statement

J.M. devised the original research project and proposed the global methodology. L.Y. designed the computational framework and analyzed the data. L.Y. wrote the draft manuscript with the inputs from all authors. K.S and S.S. revised the draft manuscript and proposed additional experiments: three dimensions and latent space of environmental attributes. All authors reviewed the results and approved the final version of the manuscript.

Conflict of interest statement

On behalf of all authors, the corresponding author states that there is no conflict of interest.

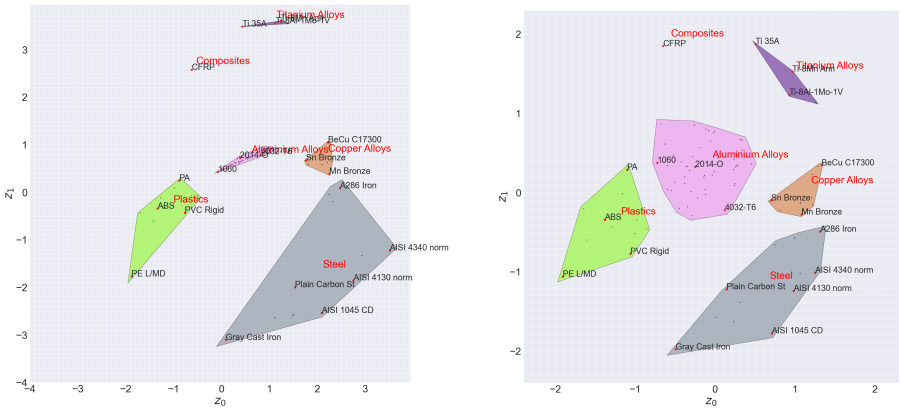
Appendix A Annexed figures



(a) Yield strength colormap for 100 neurons

(b) Yield strength colormap for 1000 neurons

Fig. A1: Comparison in the distribution of Yield Strength values in the latent space for different numbers of neurons



(a) Material representation in a two-dimensional latent space for $\beta = 5e-6$

(b) Material representation in a two-dimensional latent space for $\beta = 5e-4$

Fig. A2: Material representation in a two-dimensional latent space comparison for different values of β parameter. Note that the map enlarges

Appendix B Annexed tables

β value	Error type	$\Delta E\%$	$\Delta\rho\%$	$\Delta C\%$	$\Delta Y\%$	$\Delta P\%$	$\Delta V\%$	$\Delta W\%$
5e-6	**Max Error**	**6.1**	**3.2**	**3.7**	**7.8**	**1.1**	**1.3**	**1.1**
	Avg Error	**1.1**	**0.7**	**0.5**	**1.0**	**0.2**	**0.3**	**0.2**
5e-5	**Max Error**	**7.7**	**6.3**	**3.3**	**7.8**	**2.1**	**2.6**	**4.0**
	Avg Error	**1.6**	**1.6**	**0.7**	**2.2**	**0.3**	**0.3**	**0.6**
5e-4	**Max Error**	**16.8**	**17.7**	**5.8**	**18.4**	**5.6**	**6.8**	**5.8**
	Avg Error	**2.4**	**4.1**	**1.2**	**5.1**	**0.7**	**1.0**	**0.8**

Table B1: Comparison of error for different values of the β parameter

Objective	Equation
Compliance	$\mathbf{f}^\intercal\mathbf{u}(\mathbf{A}, \hat{E})$
Mass	$\hat{\rho}\sum_{k=1}^{N} A_k L_k$
Cost	$\hat{\rho}\hat{C}\sum_{k=1}^{N} A_k L_k$
CO2	$\hat{\rho}\hat{P}\sum_{k=1}^{N} A_k L_k$
Energy	$\hat{\rho}\hat{V}\sum_{k=1}^{N} A_k L_k$
Water	$\hat{\rho}\hat{W}\sum_{k=1}^{N} A_k L_k$

Table B2: Available objective functions

Constraint	Equation
Compliance	$\left(\frac{\mathbf{f}^\intercal \mathbf{u}(\mathbf{A}, \hat{E})}{J^*}\right) - 1$
Mass	$\left(\frac{\hat{\rho}}{M^*} \sum_{k=1}^{N} A_k L_k\right) - 1$
Cost	$\left(\frac{\hat{\rho}\hat{C}}{C*} \sum_{k=1}^{N} A_k L_k\right) - 1$
Yield	$\underset{k}{max}\left(\frac{P_k}{\hat{Y} A_k}\right) - \frac{1}{F_s}$
Buckling	$\underset{k}{max}\left(\frac{-4P_k L_k^2}{\pi^2 \hat{E} A_k^2}\right) - \frac{1}{F_s}$
CO2	$\left(\frac{\hat{\rho}\hat{P}}{C*} \sum_{k=1}^{N} A_k L_k\right) - 1$
Energy	$\left(\frac{\hat{\rho}\hat{V}}{V*} \sum_{k=1}^{N} A_k L_k\right) - 1$
Water	$\left(\frac{\hat{\rho}\hat{W}}{W*} \sum_{k=1}^{N} A_k L_k\right) - 1$
Confidence	$1 - \frac{\left(1 - \frac{\lVert z^* - z_m \rVert}{\underset{\forall k \in M}{max}\left(\lVert z^* - z_k \rVert\right)}\right)}{\gamma^*}$

Table B3: Available constraints