Highlights

Deep learning classifier for life cycle optimization of steel-concrete composite bridges

D. Martínez-Muñoz, J. García, J.V. Martí, V. Yepes

- This research proposes a methodology to build a deep learning model to assess bridge compliance and optimize design calculations.
- The model is integrated into metaheuristic optimization algorithms to evaluate their performance in terms of time and the quality of the solutions obtained.
- An environmental and social life cycle analysis is carried out, which involves more complex objective functions.
- An increase in steel yield strength for optimal solutions is observed for both environmental and social objective functions in the life cycle assessment.

Deep learning classifier for life cycle optimization of steel-concrete composite bridges

D. Martínez-Muñoz[a,*], J. García[b,**], J.V. Martí[a], V. Yepes[a]

[a]Institute of Concrete Science and Technology (ICITECH). Universitat Politècnica de València. València, 46022, Spain. damarmu1@cam.upv.es, jvmartia@cst.upv.es, vyepesp@cst.upv.es
[b]Escuela de Ingeniería de Construcción y Transporte. Pontificia Universidad Católica de Valparaíso. Valparaíso, 2362807, Chile. jose.garcia@pucv.cl

---

Abstract

The ability to conduct life cycle analyses of complex structures is vitally important for environmental and social considerations. Incorporating the life cycle into structural design optimization results in extended computational durations, underscoring the need for an innovative solution. This paper introduces a methodology leveraging deep learning to hasten structural constraint computations in an optimization context, considering the structure's life cycle. Using a composite bridge composed of concrete and steel as a case study, the research delves into hyperparameter fine-tuning to craft a robust model that accelerates calculations. The optimal deep learning model is then integrated with three metaheuristics: the Old Bachelor Acceptance with a Mutation Operator (OBAMO), the Cuckoo Search (CS), and the Sine Cosine Algorithms (SCA). Results indicate a potential 50-fold increase in computational speed using the deep learning model in certain scenarios. A comprehensive comparison reveals economic feasibility, environmental ramifications, and social life cycle assessments, with an augmented steel yield strength observed in optimal design solutions for both environmental and social objective functions, highlighting the benefits of meshing deep learning with civil engineering design optimization.

Keywords: deep learning, sustainability, optimization, bridges, machine learning, composite structures

*Corresponding author: David Martínez-Muñoz, e-mail: damarmu1@cam.upv.es
**Corresponding author: Jose García, e-mail: jose.garcia@pucv.cl

1. Introduction

The economic viability and social growth of most countries are found to be closely tied to the development, reliability, and durability of their infrastructure [1]. Infrastructure is seen as critical due to its profound influence on economic activity, growth, and employment. However, the activities related to it can exert substantial environmental and social impacts, potentially resulting in irreversible consequences that may jeopardize the present and future of society. Being a carbon-intensive industry [2], construction has been the focus of much research aiming at minimizing emissions, with the reduction of the environmental impact of construction projects becoming increasingly important. In the pursuit of the state-of-the-art, studies have been conducted on sustainable building [3, 4], optimization of energy consumption [5], and the analysis of the life cycle of CO2 emissions from concrete structures [6, 7, 8]

However, it should be noted that regardless of the criteria that researchers consider to represent the sustainability of structures, there is widespread agreement that a comprehensive evaluation of sustainability must encompass the entire life cycle of the structure [9, 10, 11, 12]. This necessitates, on one hand, the consideration of the three pillars of sustainability: economic, environmental, and social. Besides, when defining the objective function guiding this optimization, the full life cycle analysis must be taken into account, with the life cycle divided into four stages: Manufacturing, Construction, Use and Maintenance, and End of Life [13]. Furthermore, all structural designs involve variability and uncertainty [14, 15]. This implies that the optimization process becomes more complex due to the increase in the complexity of the objective functions, making the acceleration of calculations a crucial point.

One method to accelerate these calculations is through the application of machine learning techniques. For instance, dimensionality reduction techniques can be employed to simplify the dimensionality of the search space or the objective function. Alternatively, the objective function or the constraints can be replaced with a model that emulates them. For example, in the study reported in [16], the kriging technique was utilized to decrease the computation times for a concrete box-girder bridge. In [17], neural networks were used to model viscosity and conductivity values, which were then integrated into the NSGA-II (Nondominated Sorting Genetic Algorithm II) for optimization purposes.

Studies in the field of structural engineering have utilized neural networks to predict the transfer length in prestressed concrete [18]. Similarly, neural networks have been applied to forecast the energy consumption of heating, ventilation, and air conditioning systems in buildings. Subsequently, a multi-objective genetic algorithm was employed to determine the optimal consumption conditions [19]. As a result, the multi-objective optimization demonstrated improved outcomes in terms of thermal comfort and energy consumption when compared to the base case design.

In light of the remarks outlined in prior sections, a model rooted in deep learning techniques has been proposed within this work. Its primary intention is to supplant the constraints delineated in the steel-concrete composite bridge (SCCB) design. This approach not only streamlines optimization calculations but also paves the way for modeling intricacies with heightened complexity. Notably, the essence of this methodology aims at accelerating computation tasks, thereby facilitating the exploration of more intricate scenarios. Although this work focuses on a specific case, the methodology should inherently be adaptable to a range of other structural configurations.

Specifically, the contribution of this article includes:

- A methodology has been introduced to construct a deep learning model tailored for assessing bridge compliance and optimizing design calculations.
- Integration of this model into metaheuristic optimization algorithms has been realized, and its performance concerning solution quality and time efficiency has been assessed.
- A comprehensive environmental and social life cycle analysis, which involves more complex objective functions, has been conducted.

The results indicate that the deep learning model is capable of accelerating calculations by a factor of 50 when utilizing swarm-type algorithms and by a factor of 37 when using trajectory algorithms. Additionally, the outcomes from the life cycle assessment reveal an increase in steel yield stress for optimal solutions for both environmental and social objective functions. This occurs because an increase in yield strength does not result in a corresponding increase in impact. Conversely, for the cost optimization results, an increase in steel resistance directly translates into a cost increase, and optimal solutions yield lower stress values.

The structure of the content is outlined briefly as follows: Section 2 details the deep learning techniques used, the optimization techniques applied, the objective functions considered, as well as the definition of the optimization problem. The results obtained are described in Section 3. Initially, the different experiments carried out to achieve the suitable model for accelerating the calculations are outlined, followed by a detailed report on the results obtained from the structure's life cycle analysis. Finally, in Section 4, the primary conclusions and the suggested next steps are presented.

2. Deep Learning metamodel assisted optimization

Structural problems are often characterized by their high complexity, which results in substantial computational costs. The complexity of the model often entails such high computational costs that it necessitates the elimination of some constraints from the initial model or the simplification of the associated objective functions. Moreover, multiple runs of these complex structural models are required during optimization processes to obtain the optimal result. To reduce computation time, this research proposes a Deep Neural Network (DNN) metamodel, explained in Section 2.1, to predict the feasibility of structural solutions for a steel-concrete composite bridge (SCCB) deck. This metamodel has been applied to various metaheuristics, as described in Section 2.2, to compare the results obtained. Furthermore, this study considers three objective functions, defined in Section 2.3, to compare results concerning the three pillars of sustainability, treated as single objective optimizations.

2.1. Deep neural networks model

This section elaborates on the proposed methodology for training the deep neural network model designed to accelerate optimization calculations. It should be noted that the constructed model resolves the issue of whether or not the bridge to be optimized adheres to the imposed constraints. In this sense, the model addresses a binary classification problem. The primary components of the developed methodology for constructing the classification model involve deep learning-based methods. Essentially, there are three aspects to be developed. The first aspect relates to the construction of the training dataset; the second involves the definition of the network topology and the hyperparameters used. Lastly, the third aspect entails defining the metrics and evaluating the best configuration. These points will be discussed in this section.

2.1.1. Methodology used for the construction of the training data set

This section details how the dataset used to train various deep neural network models was constructed. Multiple datasets were assembled to ensure the networks were calibrated, with the aim of identifying the most effective training approach. Different optimization techniques were explored, and full runs were executed for both OBAMO and SCA. During each optimization, data was collected and checked against predefined structural standards. Owing to an imbalance between the conditions that were met and those that were not, a decision was made to compare cases of unbalanced data with cases where the training datasets were balanced using the Synthetic Minority Oversampling Technique (SMOTE). Independent training sessions for OBAMO, SCA, and a hybrid scenario where both datasets were merged were also compared. Data integration for both unbalanced datasets and those balanced with SMOTE was evaluated. In the case of SMOTE, the sampling strategy parameter was set to one.

2.1.2. Topology network definition, hyper-parameters explored and metrics used

For the network topology's definition, multilayer perceptron neural networks were used within the TensorFlow framework. In the initial topology definition, a single-layer network with different node quantities was examined. Configurations with 64, 128, and 256 nodes were specifically tested. After the first layer was finalized, the addition of a second layer, having half the number of nodes as the first layer, was considered. If improvements in the defined metrics were observed with the introduction of this second layer compared to the single-layer network, the potential inclusion of a third layer was assessed. In this third layer, the number of nodes was set to be $\frac{n}{4}$ of the first layer's node count. The explored hyperparameters were the optimization algorithm, the batch size, and the number of epochs. Three techniques were evaluated for the optimization algorithm: SGD, RMSprop, and Adam. Configurations of 32, 64, and 128 were tested for the batch size. A maximum value of 100 was set for epochs, and early stopping was implemented. According to this rule, if no improvement was seen in the test set after 10 iterations, the training process was halted. Due to the importance of minimizing both false positives and false negatives in the used metrics, the

F1-score metric was chosen, which calculates the harmonic average between precision and recall.

2.2. Hybrid metaheuristics

This section presents the metaheuristics utilized in this study, which can be categorized into two primary groups: trajectory-based and swarm intelligence techniques. All the algorithms in this research have undergone a process of hybridization. The trajectory-based techniques introduce minor modifications to the variable vector to adjust the solution and seek the optimum. Mutation operators have been incorporated into these algorithms as part of the hybridization process to enhance the optimization process's exploration capacity. On the other hand, swarm intelligence techniques vary the solution by adjusting the variables to search for a particular characteristic of the best individual in the population. In this instance, hybridization has been achieved through the implementation of a k-means clustering technique. It's worth noting that all algorithms have been modified to accommodate the discrete nature of the optimization problem.

Furthermore, all methods of structural optimization necessitate a structural check module to ascertain the solution's feasibility, which typically accounts for approximately 80% of the computation time for each iteration of the optimization problem. To curtail computation time, a DNN model has been trained to predict the solution's feasibility. Detailed information about the DNN model can be found in Section 2.1. It should be noted that while it is possible for the model to encounter failures, once the optimization process is complete, the constraints of the structural problem are verified using Python-developed software [20].

2.2.1. Trajectory-based: Old Bachelor Acceptance with a Mutation Operator (OBAMO)

The search strategy employed by such algorithms involves making minor alterations to the variable vector and evaluating the consequent changes in the objective function. These metaheuristics accept inferior solutions at certain stages of the optimization process to avert local optima confinement and encourage exploration. A threshold must be defined to restrict the acceptance of solutions that exceed acceptable boundaries. In this study, the threshold was dynamically adjusted during optimization, being increased or decreased based on the solution acceptance rate. The Old Bachelor Acceptance with a Mutation Operator (OBAMO2) is an adaptive threshold algorithm utilized in other structural optimization problems [21]. In this study, the OBAMO2 method was hybridized with a characteristic of Genetic Algorithms, specifically, the mutation operator, which allows for certain mutations during optimization to stimulate exploration.

The Old Bachelor Acceptance (OBA) algorithm is an iterative heuristic optimization method proposed by Hu et al. [22]. This procedure begins with an initial solution and modifies it through movement. If the new solution falls within the defined threshold, it is accepted, even if its objective function value is inferior. Contrary to Simulated Annealing (SA) [23], which utilizes a monotonically decreasing acceptance scheme with decreasing temperature, the acceptance criterion used by OBA is based on a dynamically changing threshold that adheres to the principle of 'decreasing expectations'. After each unsuccessful attempt to improve the solution, the threshold is increased to permit the transition to somewhat inferior solutions. Conversely, with successive enhancements in the solutions, the threshold is reduced. Hu et al. [22] highlight several advantages of OBA over SA, such as the non-monotonic acceptance scheme, the self-adjusting growth and decay of the thresholds, and the ability to adapt to a preset calculation time.

The OBA algorithm was selected for this study because it has been successfully applied to other structural optimization problems in the past [24]. In an effort to enhance exploration during the optimization process, a mutation operator was incorporated, drawing on recent research [21]. OBAMO is a hybrid algorithm that combines the algorithm presented in Algorithm 1 with a mutation operator. The algorithm depends on five parameters: the number of iterations (N), the threshold updating parameter (Δ), the limit of movements without improvement (δ), the standard deviation (SD), and the number of variables (VN) permitted to change between iterations. The most effective combination of these parameters was determined using a Design of Experiments method [25], yielding values of 20,000, 0.3, 1, 100, and 9 for N, SD, VN, Δ, and δ, respectively.

2.2.2. Swarm intelligence: SCA and CS

Swarm intelligence methods mimic the behavior of natural systems in the pursuit of optimal solutions. These methods generate populations of individuals that interact with one another, emulating the behavior of specific species. Two such algorithms that have been proposed include the Sine Cosine Algorithm (SCA), which employs sine and cosine functions to simulate individual movements, and Cuckoo Search (CS), which models the behav-

Algorithm 1 Old Bachelor Acceptance 2 [22]

M = Maximum iteration number
Δ = Threshold updating parameter
δ = Limit of movements without improvement
count = Counter of consecutive movements accepted
$T_0 = 0$; *prev_age* = M
Choose of random solution s_0
for i=0 to M-1 **do**
    Choose a random neighboring solution s'
    if $f(s') < f(s_i) + T_i$ **then**
        $s_{i+1} = s'$
        $age = 0$
        if $prev\_age < \delta$ **then**
            $count = count + 1$
        else
            $count = 1$
        end if
        $T_{i+1} = T_i - count \cdot \Delta \cdot (1 - i/M)$
    else
        $s_{i+1} = s_i$
        $age = age + 1$
        $T_{i+1} = T_i + \Delta/\delta \cdot (1 - i/M)$
    end if
    $prev\_age = age$
end for
$s_i = s_i$ corresponding with minimum $f(s_i)$ with $0 \leq i \leq M$

ior of natural cuckoo populations. Furthermore, recent studies in structural optimization have suggested that the introduction of a hybridization technique, such as K-means clustering, can enhance the performance of these metaheuristics [26, 27].

Sine Cosine Algorithm (SCA). The SCA is a swarm intelligence method devised by Mirjalili [28], utilizing sine and cosine functions to explore the solution space. The movement of individuals is governed by P_j^t, typically drawn from the best solution found at the location of the optimal solution for iteration t and dimension j. Additionally, the algorithm employs three random numbers: r_1, r_2, and r_3. The values of these numbers determine whether the movement of the solutions is orchestrated by a sine or a cosine function, as illustrated in Equations 1 and 2, respectively.

$$x_{i,j}^{t+1} = x_{i,j}^{t} + r_1 \times sin(r_2) \times \mid r_3 P_j^t - x_{i,j}^t \mid \tag{1}$$

$$x_{i,j}^{t+1} = x_{i,j}^{t} + r_1 \times cos(r_2) \times \mid r_3 P_j^t - x_{i,j}^t \mid \tag{2}$$

Cuckoo Search (CS). The CS algorithm is inspired by the cuckoo bird species, which lays its eggs in the nests of other bird species and sometimes mimics the hues and patterns of the host species' eggs. In this algorithm, an egg represents a solution, and the basic idea is to replace inadequate solutions with better ones, analogous to cuckoos replacing the host bird's eggs. The CS algorithm is based on three essential principles:

1. Each cuckoo lays one egg at a time, which is randomly placed in a nest.
2. Only the best nests, which produce high-quality eggs, are considered for the next generation.
3. The number of available nests is fixed, and the host bird has a probability $p_a \in (0, 1)$ of discovering the cuckoo's egg.

$$x_{i,j}^{t+1} = x_{i,j}^{t} + \alpha \bigoplus \text{Lévy}(\lambda) \tag{3}$$

The step size $\alpha > 0$ should be chosen proportionally to the scales of the problem. The operator $\bigoplus$ denotes element-wise multiplication. To simulate a random walk, the Lévy flight draws the step length from a Lévy distribution Lévy $\sim t^{-\lambda}$, where $1 < \lambda \leq 3$.

Hybridization technique: K-means clustering. The hybrid method is employed for swarm intelligence metaheuristics, given that both methods are naturally suited for continuous domains. The hybrid method takes as input the metaheuristic MH, the list of discrete solutions obtained in the previous iteration $lSol$, and a list of transition probabilities $transitionProbs$, and returns a new list of discrete solutions $lSol$. In the initial stage, the discretization method computes the velocity of MH. For CSA and CS, this velocity corresponds to the component obtained from the difference $|x_{i,j}^{t+1} - x_{i,j}^{t}|$ in Equations 1 through 3.

Following that, a transfer function is applied to convert the velocity values, which range over $\mathbb{R}$, into values between [0, 1). A v-shape transfer function, specifically $|\tanh(v)|$, is used in this instance. Then, for each solution and dimension, the value of $lSolProbability$, obtained by applying the transfer function, is compared with a randomly generated number r_1 between [0,1). If the value of $lSolProbability$ is greater than the random number, an update is triggered in that dimension; otherwise, it remains unaltered.

The updating process presents two possibilities: firstly, a parameter β is considered, and a random number r_2 is generated. If r_2 is less than β, the value is replaced with the best solution value obtained for that dimension. Otherwise, a random update is executed to enhance the exploration of the search space.

Subsequently, a k-means clustering technique is employed to convert the velocity values, which range over $\mathbb{R}$, into transition probability values that fall within the range of [0,1). The k-means technique forms clusters, in this instance, five clusters, and orders them from the smallest to the largest centroids. The smallest transition probability is assigned to all velocities within the cluster with the smallest centroid, while the largest transition probability is assigned to all points within the cluster with the largest centroid. Figure 1 graphically demonstrates the k-means procedure. The transition probability values utilized in this study were [0.1, 0.2, 0.4, 0.8, 0.9].

In each dimension of every solution, the transition probability, denoted by $DimSolProb_{i,j}$, is calculated. If this probability is greater than a random number r_1 and if β is greater than another random number r_2, the dimension value of the solution is updated with the best solution identified thus far. If the condition related to β is not met, the dimension is updated with a random permissible value. However, if neither the transition probability condition nor the β condition are satisfied, the dimension of the solution is not updated. This final option serves to broaden the exploration of the search space.

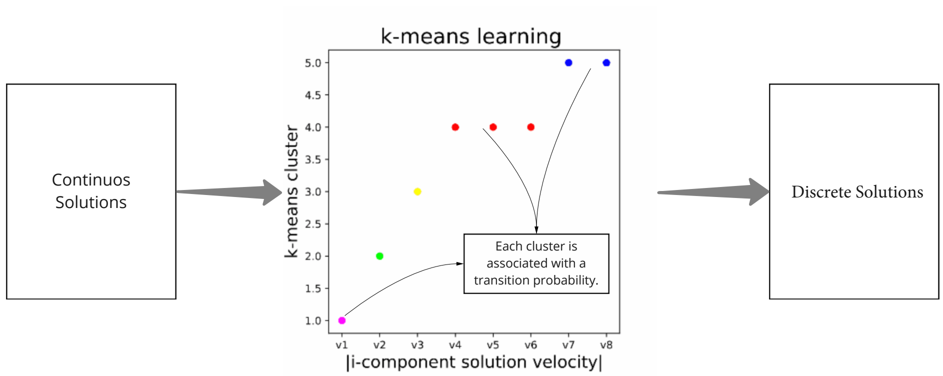

Figure 1: K-means discretization techniques diagram.

Algorithm 2 Hybrid algorithm

Function Discretization($lSol$, MH,$transitionProbs$)
Input $lSol$, MH, $transitionProbs$
Output $lSol$
$vlSol \leftarrow$ getVelocities($lSol$, MH)
$lSolClustered \leftarrow$ appliedKmeansClustering($vlSol$, K)
for (each Sol_i in $lSolClustered$) **do**
  for (each $dimSol_{i,j}l$ in Sol_i) **do**
    $dimSolProb_{i,j}$ = getClusterProbability($dimSol$, $transitionProbs$)
    if $dimSolProb_{i,j} > r_1$ **then**
      if $beta > r_2$ **then**
        Update $lSol_{i,j}$ considering the best.
      else
        Update $lSol_{i,j}$ with a random value allowed.
      end if
    else
      Don't update the element in $lSol_{i,j}$
    end if
  end for
end for
return $lSol$

Table 1: Cost values of every construction unit for SCCB [29]

Construction unit	**Unit**	**Cost (€)**
Ccncrete C25/30	m^3	88.86
Concrete C30/37	m^3	97.80
Concrete C35/45	m^3	101.03
Concrete C40/50	m^3	104.08
Precast pre-slab	m^3	27.10
Reinforcement steel B400S	kg	1.40
Reinforcement steel B500S	kg	1.42
Rolled steel S275	kg	1.72
Rolled steel S355	kg	1.85
Rolled steel S460	kg	2.01
Shear-connector steel	kg	1.70

2.3. Objective functions

The optimization problem in this study involves determining the best design for an SCCB while also upholding sustainability. This is achieved through the incorporation of objective functions that reflect the pillars of sustainability. Specifically, we evaluate the economic cost, environmental, and social life cycle assessments of the SCCB deck. These are represented by equations 4, 5, and 6 respectively.

$$C(\vec{x}) = \sum_{i=1}^{n} p_i \cdot m_i(\vec{x}) \tag{4}$$

The total cost of bridge construction is calculated by the objective cost function, which multiplies the unit cost of each required activity with the corresponding measurement. A comprehensive list of all construction units and their respective costs, sourced from the BEDEC database [29], is presented in Table 1. In equation 4, p_i is indicative of the price of each construction unit, while its measurement is represented by m_i.

$$ELCA(\vec{x}) = \sum_{i=1}^{n} \sum_{j=1}^{p} elca_j \cdot m_j(\vec{x}) \tag{5}$$

$$SLCA(\vec{x}) = \sum_{i=1}^{n} \sum_{j=1}^{p} slca_j \cdot m_j(\vec{x}) \tag{6}$$

The evaluation of the environmental (ELCA) and social impact (SLCA) of the structure, accounting for all involved processes from raw material extraction to demolition and transportation to a landfill site, is the primary objective of the life cycle assessment (LCA). In equations 5 and 6, each life cycle stage is represented by i, with $elca_j$ and $slca_j$ corresponding to the environmental and social impact of each process within a given stage, respectively. The corresponding measurement of each process is indicated by m_j. The environmental and social impact of each process along with their corresponding measurement are detailed in Table 2. The LCA methodology is described in section 2.3.1.

Table 2: Ecoinvent processes LCA environmental and social impact values

Process	**Unit**	$elca_i$ **(points)**	$slca_i$ **(mrh)**
concrete production 25MPa	m^3	2.037E+01	1.254E+05
concrete production 30MPa	m^3	2.631E+01	1.668E+05
concrete production 35MPa	m^3	2.478E+01	1.554E+05
concrete production 40MPa	m^3	2.585E+01	1.623E+05
steel production 71% of recycling rate	kg	1.523E-01	1.941E+03
steel production 98% of recycling rate	kg	1.036E-01	2.067E+03
transport, freight, lorry 16-32 metric ton, EURO6	t·km	2.502E-02	4.116E+01
transport, freight, lorry 3.5-7.5 metric ton, EURO6	t·km	7.755E-02	1.655E+02
welding, arc, steel	m	2.350E-02	2.535E+02
welding, gas, steel	m	2.303E-02	2.429E+02
diesel, burned in building machine	MJ	1.361E-02	8.764E+00
carbon dioxide	kg	4.369E-02	0.000E+00
rock crushing	kg	7.223E-05	8.305E-01

2.3.1. Life cycle assessment method

The processes involved in an activity or product, including all the necessary stages to complete it, are evaluated for environmental and social impact by the life cycle assessment (LCA). The ISO 14040:2006 [13] regulation is adhered to for carrying out the environmental LCA for bridges, while the assessment of the social impact is guided by the Guidelines for Social Life Cycle Assessment of Products [30]. Impact information from databases and a chosen life cycle impact assessment (LCIA) method are required to model the life cycle of a structure. The ReCiPe 2008 method [31] for environmental LCA and the social impacts weighting method (SIWM) for social impact are utilized in this research. The ecoinvent v3.7.1 [32] and soca v2 [33] databases are used for environmental and social LCA, respectively, due to their reliability and frequent updates [34]. Additionally, the ability of the soca database to associate ecoinvent processes with the PSILCA [35] database social impacts renders it valuable for scientists [36].

Four stages are identified in this research to assess the impact of an SCCB: manufacturing, construction, use, and end-of-life, which align with the stages defined in previous LCA studies on bridges [36]. The manufacturing phase is comprised of the transformation of raw materials into products needed for construction and the transportation of these products to the construction site, taking into consideration the waste generated during these activities. A significant influence on the global environmental impact of the SCCB is noted from the impact of recycled steel, particularly in the production of steel products [36]. The distinction between structural and reinforcement steel is deemed critical, given the differing recycling percentages. For instance, a 71% recycling rate is reported for reinforcement steel, while a rate of 98% is reported for structural steel in developed countries such as the US [37].

The construction phase includes actions required to build the bridge, such as equipment and building style, and location. Formwork, scaffolding, vibrators, and concrete pouring must be considered, and procedures for welding the steel sections should be established for steel and steel-concrete composite bridges. The diesel consumption of machinery during construction, based on manufacturer information, literature, or other sources, is included in the LCA model for modeling construction activities.

All activities required throughout the structure's lifetime are encompassed within the use and maintenance stage. The potential for concrete carbonation to sequester CO2 has been explored in recent research [38, 39]. An expression for concrete carbonation was developed by García-Segura et al. [40], represented by equation 7. The service life t, the carbonation coefficient k, the exposed area A, and the amount of cement C in one cubic meter of concrete are considered in this equation. Additionally, k represents the amount of clinker in the cement.

$$CO_2 fixed\,(kg) = 0.383 \cdot \frac{k\left(\frac{mm}{\sqrt{year}}\right) \cdot \sqrt{t(year)}}{1000} \cdot A(m^2) \cdot C\left(\frac{kg}{m^3}\right) \cdot k(\%) \quad (7)$$

The end-of-life stage includes the procedures that occur after the structure's lifetime, specifically the dismantling of the structure. This stage involves using machinery to demolish the structure and transporting and treating the waste generated during this process. The distances between the building site and the landfill or waste treatment facilities must be specified as part of the analysis. Depending on the properties of the waste materials, there are three primary options for their disposal: reuse, recycling, or landfilling. Concrete and steel are the most commonly used materials in bridge construction, and waste treatment options depend on the region and population's needs.

The inventory analysis involves collecting data on all the materials and energy consumed during the bridge life cycle. Considering these processes' outputs allows for determining the product's environmental impact. Figure 2 shows the processes involved in each stage.

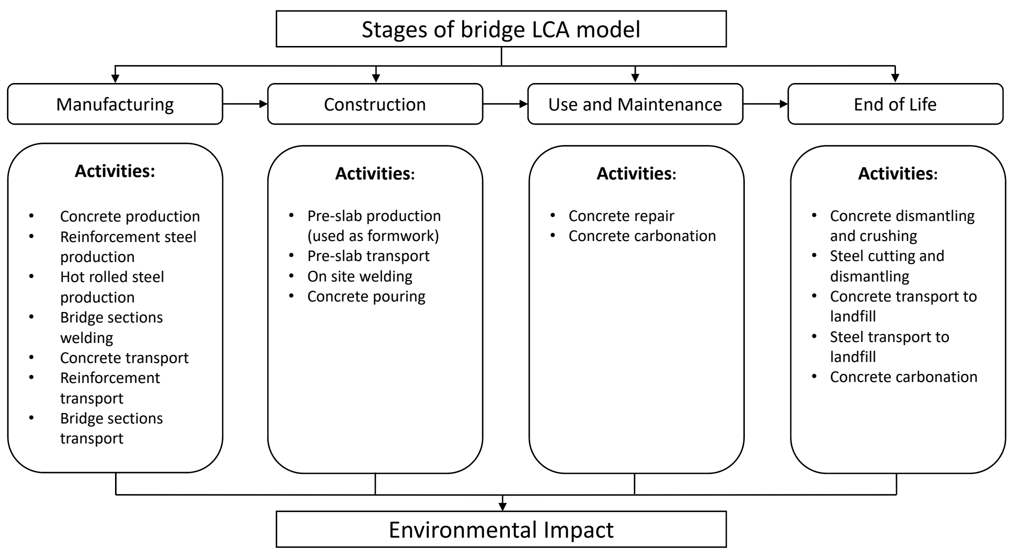

Figure 2: Bridge life cycle model stages and activities

The LCA impact was assessed using a Python script incorporating data from Ecoinvent version 3.7.1 [32] and soca version 2 [33]. One unit of each product was modeled using GreenDelta's OpenLCA software, an open-source tool widely used in the scientific community for LCA [41].

2.4. Problem definition

The optimization of a 60-100-60 meter SCCB deck structure with a box-girder geometry is the aim of this study. The optimization problem has been defined in previous research, which used single-objective metaheuristic optimization methods to evaluate cost, CO2 emissions, and embodied energy

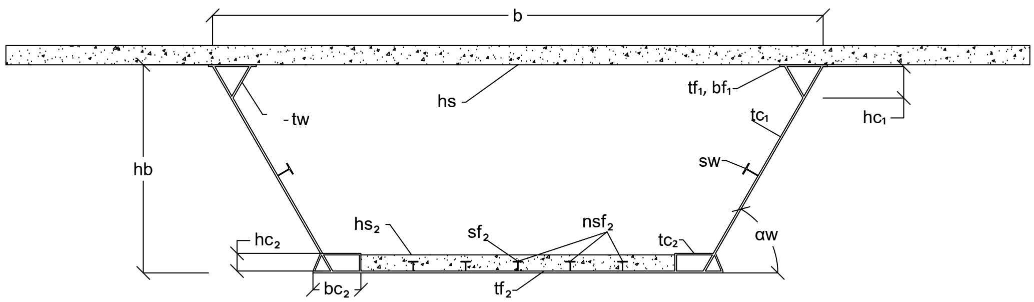

Figure 3: SCCB structural optimization problem cross-section variables

[26, 27, 5]. A metamodel-assisted strategy utilizing deep neural networks (DNN) for environmental and social life cycle assessment (LCA) optimization is introduced in this paper. The strategy enables a comparison of the computational costs and design changes associated with considering a comprehensive social and environmental impact profile.

2.4.1. Variables and parameters

The structural problem for this research involves a 60-100-60 meter SCCB deck with a box-girder geometry. The problem includes 34 design variables considering the bridge's cross-section, stiffeners geometry, slab reinforcement, and material strength. The variables are grouped into four categories: cross-section geometry variables (b, α_w, h_s, h_b, h_{fb}, t_{f_1}, b_{f_1}, h_{c_1}, t_{c_1}, t_w, h_{c_2}, t_{c_2}, b_{c_2}, t_{f_2}, h_{s_2}); stiffener and floor beam variables ($n_{s_{f_2}}$, d_{st}, d_{sd}, s_{f_2}, s_w, s_t, h_{fb}, b_{fb}, $t_{f_{fb}}$, $t_{w_{fb}}$), which define the stiffeners' and transverse elements' position and geometry; reinforcement and shear connector variables (n_{r_1}, n_{r_2}, ϕ_{base}, ϕ_{r_1}, ϕ_{r_2}, h_{sc}, ϕ_{sc}); and material strength variables (f_{ck}, f_{yk}, f_{sk}). The geometric variables' position in the cross-section is shown in Figure 3, while the floor beams and stiffeners variables are presented in Figure 5. The optimization problem is discrete, as previously reported in related research on this optimization problem [27]. Lower and upper bounds and step sizes have been defined for all SCCB variables, and the discretization of the variables is summarized in Table 3. Considering all possible combinations, the number of designs is equal to 1.38×10^{46}.

Additionally, there are parameters in the optimization problem that are kept constant throughout the optimization process, referred to as fixed parameters. These parameters remain consistent with those defined in the original problem [36]. The first fixed parameters consist of the bridge's length

Table 3: Optimization problem variables and boundaries [26, 5, 27]

Variables	**Unit**	**Lower Limit**	**Upper Limit**	**Step Size**	**Possibilities**
Geometrical variables					
b	m	7	10	0.01	301
α_w	deg	45	90	1	46
h_s	mm	200	400	10	21
h_b	cm	250 ($L/40$)	400 ($L/25$)	1	151
t_{f_1}	mm	25	80	1	56
b_{f_1}	mm	300	1000	10	71
h_{c_1}	mm	0	1000	1	101
t_{c_1}	mm	16	25	1	10
t_w	mm	16	25	1	10
h_{c_2}	mm	0	1000	10	101
t_{c_2}	mm	16	25	1	10
b_{c_2}	mm	300	1000	10	71
t_{f_2}	mm	25	80	1	56
h_{s_2}	mm	150	400	10	26
Stiffeners and floor beams					
$n_{s_{f_2}}$	u	0	10	1	11
d_{st}	m	1	5	0.1	41
d_{sd}	m	4	10	0.1	61
s_{f_2}	mm	IPE 200 – IPE 600 *			12
s_w	mm	IPE 200 – IPE 600 *			12
s_t	mm	IPE 200 – IPE 600 *			12
h_{fb}	mm	400	700	100	31
b_{fb}	mm	200	1000	100	9
$t_{f_{fb}}$	mm	25	35	1	11
$t_{w_{fb}}$	mm	25	35	1	11
Reinforcement and shear connectors					
n_{r_1}	u	200	500	1	301
n_{r_2}	u	200	500	1	301
ϕ_{base}	mm	6, 8, 10, 12, 16, 20, 25, 32			8
ϕ_{r_1}	mm	6, 8, 10, 12, 16, 20, 25, 32			8
ϕ_{r_2}	mm	6, 8, 10, 12, 16, 20, 25, 32			8
h_{sc}	mm	100, 150, 175, 200			4
ϕ_{sc}	mm	16, 19, 22			3
Material strength					
f_{ck}	MPa	25, 30, 35, 40			4
f_{yk}	MPa	275, 355, 460			3
f_{sk}	MPa	400, 500			2

* Following the series of IPE profiles defined in [42].

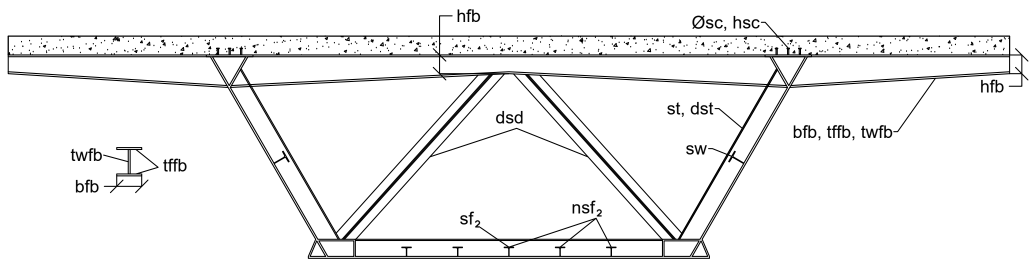

Figure 4: SCCB structural optimization stiffeners and floor beam variables

and width. A total length of 200 m is spanned by the bridge, with two lateral spans of 60 m and a central span of 100 m, and the width (W) is set at 16 m. The bounds of the variables defined in Table 3 are also treated as fixed parameters. Furthermore, the position and minimum values for certain elements, such as the reinforcement areas, lower flange, web thicknesses, and lower slab distributions, are defined by other parameters, as shown in Figure 5. Specific design guidelines [43, 44] stipulate that the minimum values of the web and bottom flange thicknesses ($t_{w_{min}}$, t_{f2min}) should be 15 mm and 25 mm, respectively. The last geometrical parameter, the reinforcement coating, is set to 45 mm by Eurocode 2 [45] for an XD2 environment.

In addition, the following parameters define the characteristics of the concrete according to Eurocode 2 [45] regulations. These parameters include the maximum aggregate size, fixed at 20 mm, and the steel and concrete Young's longitudinal and transverse moduli. The parameter values for steel are fixed at 210,000 MPa and 80,769 MPa, respectively, while for concrete, they depend on the strength, with the expressions $22 \cdot ((f_{ck} + 8)/10)^3$ and $E_{cm}/(2 \cdot (1 + 0.2))$.

Finally, the last set of parameters defines the bridge service life, structural class, and loading parameters. The service life for this type of structure is set at 100 years, while the structural class is determined to be S5 following Eurocodes [46]. The loads considered in the bridge include self-weight, dead loads, traffic, temperature variation, and wind, with all loads defined per Eurocode 1 [46].

2.4.2. Constraints

The optimization problem is subject to constraints that ensure structural safety (ULS) and serviceability (SLS), as prescribed by Eurocodes [47, 48, 45].

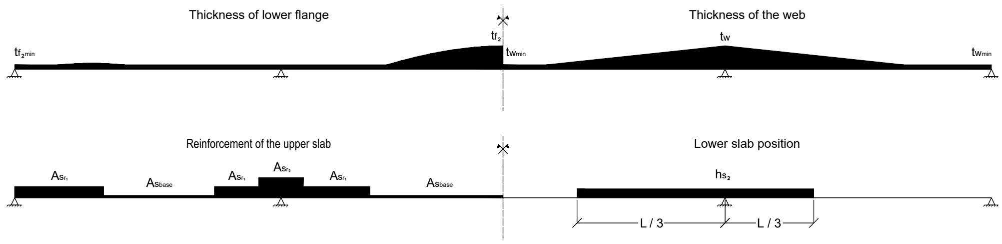

Figure 5: Reinforcement, thicknesses and lower slabs distribution in bridge spans

Specific design guidelines [43, 44] were also considered to establish additional constraints.

Structural resistance of the bridge sections falls under ULS constraints, while SLS constraints relate to prescribed stresses and deflection limitations of materials and the structure. Load and combination prescriptions were taken from Eurocode 1.

Both local and global structural models were utilized to perform ULS checking. The global analysis evaluated shear, flexure, torsion, and flexure-shear interaction, checking for solution feasibility. Shear lag [47] and Class 4 section slenderness [45] were taken into account when determining section resistance. A 10^{-6} accuracy was specified for the iterative Class 4 reduction method. Homogenization of sections was done by considering the coefficient (n) between the longitudinal deflection modulus of steel (E_s) and concrete (E_cm) according to Equation 8. Concrete creep and shrinkage was determined according to Eurocodes [47, 48, 45]. Local modeling was employed to assess floor beam and diaphragm response to ULS.

$$n = \frac{Es}{Ecm} \tag{8}$$

Regarding SLS constraints, the deflection limit was determined according to Spanish regulation IAP-11 [49], which stipulates a maximum deflection value of L/1000 for frequent combinations of live loads, where L denotes the length of each span. Structural and geometrical constraints were also specified. All structural tests were performed using a Python-programmed numerical model [20].

The ULS and SLS checking coefficients were determined based on the difference between the design values of the effects of actions (E_d) and their corresponding resistance values (R_d), as illustrated by Equation 9. The section satisfies the constraints if these coefficients are greater than or equal to one.

$$\frac{R_d}{E_d} \geq 1 \tag{9}$$

3. Results and discussion

This section provides details of the primary experiments conducted in the integration of the deep learning model with the described optimization algorithms. For ease of understanding, the results section is divided into two sub-sections. In the first sub-section, 3.1, the central experiments that facilitate the construction of the deep learning model are detailed. Subsequently, the results concerning the times and minimums obtained by applying the deep learning model to the different optimization algorithms are described, using the best model obtained. Once the best configurations are identified, the algorithms are applied to environmental and social life cycle analysis in the second sub-section. The comparison and discussion of these results are detailed in sub-section 3.2.

3.1. Algorithm Analysis

This section is dedicated to detailing the methodology employed to develop the deep learning model. The primary hyperparameters and techniques utilized in achieving the model are outlined. Subsequently, a comparison is made between different metaheuristics that solve the optimization problem, with and without the integration of the deep learning model

3.1.1. Neural Network models comparison

The construction of the classification model considered multilayer perceptron networks, [50]. The values of the 34 variables that define the design of a bridge were used as input variables (Table 3). A series of parameters that require exploration for proper tuning are encompassed within multilayer perceptron networks. Prominent among these parameters are the number of layers and the optimization method employed for network learning. In addition, due to an imbalance between the classes, SMOTE, [51], was employed as an oversampling method. Moreover, the data set used for the training, a critical aspect of the model construction, was carefully selected. Building a good data set for this type of problem presents several difficulties, such as class imbalance and fewer points usually associated with values close to the minimum of the objective functions. Therefore, various experiments were conducted to build the training set. Two types of heuristic techniques, one based on trajectory, OBAMO, and another of the swarm class, SCA, were employed to generate the data set. Three scenarios were tested: a dataset generated by OBAMO, one generated by SCA, and one that integrates both datasets.

The data set hybrid used has approximately 20,000 bridges that satisfy the constraints of the structural problem and 7,000 points that do not meet the conditions. Table 4 shows the results of the 5-fold cross-validation considering 1, 2, and 3 hidden layers and using oversampling with SMOTE. The test set was generated prior to performing the oversampling process. It is also important to consider that the Batch Size parameter, the optimization method, and the type of dataset used (hybrid) remained fixed in the experiment. When looking at the F1-score, it is clear from the table that using three hidden layers performs better when using the original data set or the oversampled dataset. We also observe that the oversampling case is higher than the standard model in the four indicators analyzed.

Models	Data							
	Training				Testing			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
1 hidden layer (128)	0.62	0.61	0.75	0.67	0.61	0.60	0.74	0.67
2 hidden layer (128-64)	0.79	0.73	0.93	0.82	0.78	0.84	0.72	0.78
3 hidden layer (128-64-32)	0.85	0.94	0.76	0.84	0.85	0.94	0.76	0.85
1 hidden layer-SMOTE	0.84	0.94	0.75	0.83	0.84	0.94	0.75	0.83
2 hidden layer-SMOTE	0.83	0.79	0.93	0.85	0.83	0.79	0.93	0.85
3 hidden layer-SMOTE	0.93	0.93	0.94	0.93	0.92	0.92	0.93	0.92

Table 4: Neural network configurations explored. The parameters used in the structure of the networks were ADAM as optimization algorithm, 128 as batch size, and hybrid data set.

Another relevant experiment aims to quantify whether the hybrid dataset obtains better metrics than the other datasets. Table 5 summarizes the results using a batch size of 128, ADAM, and a three-layer network topology. The table shows that the hybrid case is more robust than each of the datasets separately in the four indicators. Finally, in Table 6, three techniques are evaluated to carry out the learning process, keeping the rest of the parameters constant. From the table, it can be seen that the ADAM method works better than Rmsprop and SGD. From the above, it is observed that the training set, the number of layers, and the oversampling are essential to obtain a model with good metrics. From now on, the model with three layers, Adam, batch size 128, will continue to be used.

Models	Data							
	Training				Testing			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
OBAMO dataset	0.87	0.90	0.85	0.87	0.87	0.90	0.85	0.87
SCA dataset	0.86	0.80	0.97	0.88	0.86	0.80	0.97	0.88
Hybrid dataset	0.93	0.93	0.94	0.93	0.92	0.92	0.93	0.92

Table 5: Exploration of different data sets. The network configuration was ADAM, with three hidden layers and a batch size of 128 and SMOTE oversampling.

Models	Data							
	Training				Testing			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
SGD	0.88	0.82	0.93	0.87	0.87	0.81	0.92	0.86
RmsProp	0.90	0.90	0.91	0.90	0.90	0.89	0.90	0.89
ADAM	0.93	0.93	0.94	0.93	0.92	0.92	0.93	0.92

Table 6: Exploration of different optimization algorithms. The network configuration was three hidden layers and a batch size of 128, SMOTE oversampling, and hybrid data set.

3.1.2. Time and optimization values analysis

With the classification model defining whether the bridge complies with the constraints, the integration of the model into the different algorithms described in section 2.2 is undertaken. The primary aim of the classification model is to accelerate calculations. The purpose of this section is to assess this acceleration efficiency through the execution times of the optimization. A correction factor must be incorporated for a fair evaluation, especially in the case of the algorithm using the classification model. This is due to the potential for errors in the model, which could invalidate the final result. Each algorithm should generate 30 valid executions; for those incorporating the DNN model, the total execution times will be added and divided by the times of the valid executions. This process yields a factor greater than one, which will be applied to the time of each valid execution conducted by the algorithm. The results, upon applying the correction factor, are displayed in table 7, with the cost functioning as the objective function in this case. The table shows a significant reduction in execution times. The algorithm with

DNN is 38 times faster in the case of OBAMO, and 50 times faster for CS and SCA. In absolute terms, CS was the fastest, followed by SCA. Another notable point is the improved optimization values; on average, all models with DNN obtain better values, and the dispersion of the values decreases as well. The next step is to utilize the algorithms with DNN for more complex objective functions.

3.2. Comparison of Objective Function Results

The primary objective of this research is to achieve a sustainable and optimal design for an SCCB. To fulfill this purpose, the impact of various variables and material quantities has been examined. To ensure a consistent comparison of solutions across all objectives, 100 iterations were conducted, and the top 30 results were selected from each of three distinct single-objective optimization sets, considering cost, ELCA, and SLCA. This approach was chosen due to the varying number of feasible solutions associated with each optimization objective. This section also includes a comparison with recent SCCB optimization studies.

The primary parameters of the cross-section and transverse stiffeners were examined initially. As depicted in Figure 6, the results exhibited similarity in terms of the distance of stiffeners and diaphragms (d_{st}, d_{sd}), with values oscillating between 2 to 3.5 m for the three objectives for transverse stiffeners and 5.5 to 8 m for diaphragms. The most pronounced disparity was discerned in the web angle α_w, where values ranged from 60 to 75 degrees for ELCA, while for both cost and SLCA, the range was higher, spanning from 60 to 85 degrees. For the ELCA and SLCA objective functions, the height of the steel beam tended towards lower values. The value distribution analysis revealed that, for SLCA and ELCA, higher groupings correlated with lower heights. This is due to the fact that the cost objective's design sought solutions with lower yield strength, thereby necessitating an increase in the cross-section height to avoid surpassing the tension limit.

The results underscore the delicate balance between sustainability considerations (as represented by ELCA and SLCA) and cost, a challenge frequently encountered in real-world design scenarios. Given the increasing emphasis on sustainability in contemporary construction practices, the distinctions in parameters observed in this study offer crucial insights for stakeholders.

For instance, the variations in web angle α_w and the height of the steel beam are not merely numerical distinctions; they represent tangible trade-offs in design choices. Engineers, designers, and policymakers can utilize these

	OBAMO		OBAMO_DNN		CS_HYBRID		CS_HYBRID_DNN		SCA_HYBRID		SCA_HYBRID_DNN	
Execution	Cost	Time (s)	Cost	Time (s)	Cost	Time (s)	Cost	Time (s)	Cost	Time (s)	Cost	Time (s)
1	3829827.6	9776.8	3839893.5	260.3	3974520.4	7988.3	3824822.2	158.7	3830092.8	7954.6	3824135.4	161.0
2	3837246.4	9763.1	3828573.8	251.4	3825115.3	7976.0	3826485.0	158.6	3864886.9	7945.0	3824413.7	161.1
3	3834063.5	9590.9	3837194.4	246.9	3825644.8	7971.3	3825419.7	158.6	3826395.0	7939.5	3824950.9	161.1
4	3837598.6	9835.6	3827829.6	259.7	3830529.3	7959.3	3826348.2	158.7	3825919.0	7929.9	3832390.1	161.1
5	3844258.2	9702.6	3825336.3	256.1	3822875.9	7957.0	3822847.9	158.5	3823801.1	7929.2	3823679.6	161.0
6	3832969.7	9984.8	3824605.0	253.4	3827681.4	8010.3	3822723.1	158.4	3835442.1	7937.7	3829824.0	161.2
7	3834233.0	9732.5	3836782.7	251.3	3824141.4	8008.0	3830633.3	158.8	3826324.6	7933.6	3824035.8	160.9
8	3834992.2	9877.2	3830826.8	255.9	3827522.6	7998.4	3823971.2	158.7	3826206.4	7948.7	3825117.2	161.0
9	3829559.1	9644.4	3837172.2	262.1	3827541.5	8057.1	3824922.4	158.8	3830234.3	7937.5	3832786.9	161.1
10	3845712.4	9588.6	3838059.9	252.6	3825756.9	8055.9	3824833.5	158.9	3825188.7	7931.7	3827681.4	161.0
11	3829112.4	9958.9	3833956.0	248.1	3824519.6	8280.7	3824445.8	158.7	3828878.5	7933.7	3822723.1	160.7
12	3836563.2	9711.7	3832930.0	243.6	3831847.4	8292.6	3823369.8	158.7	3831864.4	7931.1	3826397.4	161.0
13	3841417.9	9556.4	3831357.0	256.6	3827980.2	8305.2	3822723.1	158.5	3823462.5	7941.6	3823574.1	159.9
14	3845663.1	9551.4	3832462.6	255.3	3823891.8	8295.2	3822723.1	158.8	3828178.8	7938.0	3822723.1	160.2
15	3840202.2	9862.5	3836540.1	269.1	3825444.4	8315.8	3828432.2	158.8	3826847.5	7933.9	3828301.7	160.4
16	4701903.1	9879.8	3831624.3	251.4	3823063.7	8293.0	3826550.6	158.8	3824311.6	7951.8	3827148.2	160.4
17	3834439.0	9720.5	3830846.2	256.9	3832782.3	8321.9	3822972.7	158.8	3822723.1	7938.9	3823375.5	158.1
18	3838868.5	9536.8	3835900.0	263.0	3828246.8	8370.1	3825259.0	158.9	3824024.1	7925.6	3823551.4	158.2
19	4004603.5	9960.6	3836922.4	253.3	3831724.5	8305.0	3823381.2	158.9	3824115.0	7944.5	3832864.8	158.6
20	3826259.7	9414.0	3835033.2	251.0	3824459.1	8305.9	3827552.8	158.9	3829979.1	7930.7	3832761.0	158.4
21	3838964.1	9662.6	3836282.6	251.3	3830466.9	7911.0	3822723.1	158.6	3823245.0	7949.5	3831155.3	158.3
22	3833027.5	9618.8	3839704.5	254.8	3825593.7	7922.4	3824300.2	158.8	3828654.6	7945.4	3826851.3	158.4
23	3838077.1	9712.4	3828455.5	260.1	3826446.6	7925.5	3825898.2	158.7	3827333.5	7948.6	3826665.9	158.3
24	3836306.3	9329.1	3838283.9	252.5	3827796.8	7914.9	3822723.1	158.7	3907488.2	7929.0	3828490.4	158.4
25	3829965.4	9916.6	3830012.7	253.6	3822766.6	7936.4	3826811.5	158.8	3830913.2	7962.0	3825716.6	158.2
26	3837030.1	9618.3	3831387.9	248.3	3822723.1	7964.9	3824686.0	158.8	3829366.5	7933.5	3831455.9	158.5
27	3832832.9	9649.1	3834392.2	251.6	3822723.1	7964.8	3824663.3	159.0	3833463.0	7942.8	3825966.2	158.4
28	3840493.0	9438.0	3836860.7	246.7	3825907.6	7969.6	3824260.5	158.8	3824394.8	7924.6	3825427.8	158.4
29	3826142.7	9868.1	3831073.3	249.7	3823593.0	7963.3	3822723.1	158.8	3823562.7	7934.1	3823494.7	158.3
30	3836720.7	9566.1	3837142.9	253.9	3830083.2	7967.4	3824680.3	158.8	3830124.4	7947.6	3829812.6	158.6
Average	3870301.8	9700.9	3833581.4	254.0	3831446.3	8083.6	3824796.2	158.7	3831247.4	7939.1	3826915.7	159.7
Max	4701903.1	9984.8	3839893.5	269.1	3974520.4	8370.1	3830633.3	159.0	3907488.2	7962.0	3832864.8	161.2
Min	3826142.7	9329.1	3824605.0	243.6	3822723.1	7911.0	3822723.1	158.4	3822723.1	7924.6	3822723.1	158.1
std	160135.9	169.3	4139.7	5.4	27182.9	165.7	1924.9	0.1	16274.2	9.1	3327.5	1.3

Table 7: Comparison of results with and without deep learning model for cost optimization. The comparison was made with the results obtained in [52].

insights to make informed decisions that harmoniously blend sustainability with cost-efficiency. Furthermore, the results suggest that when transitioning to a more sustainable infrastructure paradigm, certain traditional design practices might need revisiting.

Building on this, considering the global drive towards sustainable infrastructure, it's imperative to understand how these SCCB optimization insights can be adapted to various geographic or climatic contexts.

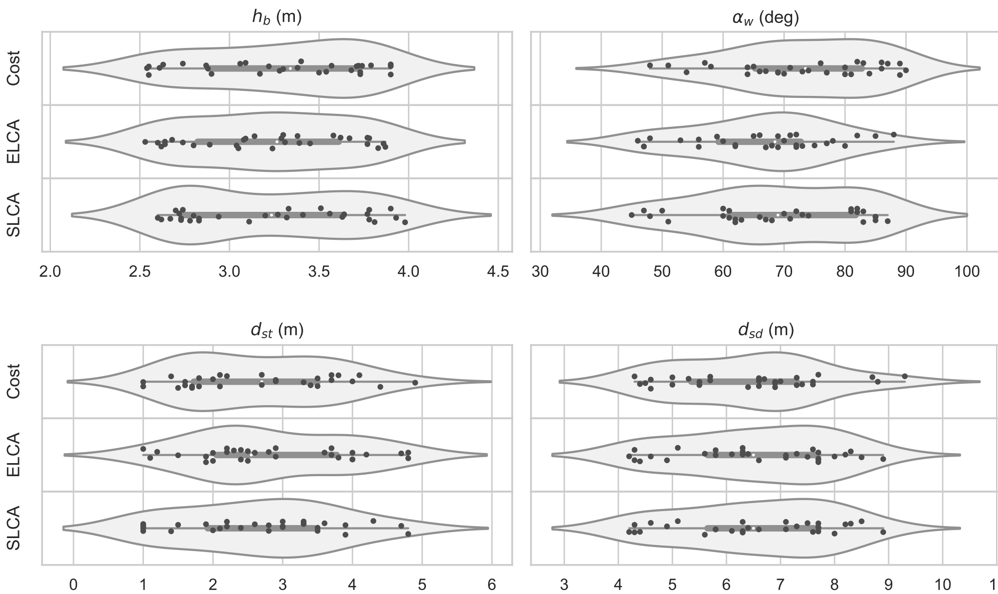

Figure 6: Cross-section main variables results for Cost, ELCA and SLCA objective functions

The variables subsequently analyzed in this study are related to the suggested cell values for the design. As depicted in Figure 7, positive values were exhibited by the height variables (h_{c_1}, h_{c_2}) for both upper and lower cells, confirming the efficacy of these elements in reducing the distance between steel plate webs without stiffening. In contrast, the thickness of these elements was minimal for the upper cell t_{c_1}, while for the lower one, values oscillated between 17 to 22. A contribution to improving the flexural behavior of the cross-section, reducing the section reduction that is often classified as class 4, was made by these elements [48].

The quantities of primary materials and the values of the objective func-

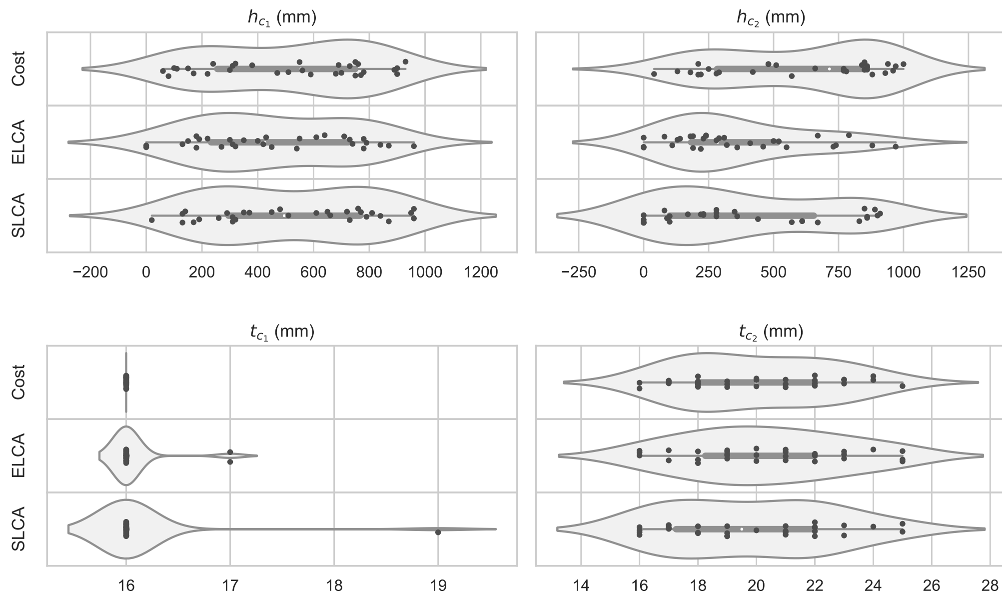

Figure 7: Cross-section cells geometry and thicknesses results for Cost, ELCA and SLCA objective functions

tion achieved by each optimization method are examined in this study. These results are summarized in Figures 8 and 9. It was observed that an identical amount of structural steel was produced by all optimization methods. However, the quantity of reinforcing steel was marginally higher for SLCA and ELCA. This increase was not substantial enough to highlight a distinct difference between the methods. Focusing on the rate of material reduction, the structural steel's quantity decreased more slowly with ELCA and Cost optimizations than with SLCA. This reduction was influenced by the inclusion of recycled steel (steel scrap) in the production process. Recent research [36] indicates a growing trend in steel production to maximize the utilization of steel scrap, aiming for optimal material reuse. Nevertheless, this tends to amplify the impact on the social aspect of sustainability, resulting in an elevated overall effect. Given that structural steel significantly influences objective functions, its quantity is curtailed in social optimization to mitigate this impact.

A further implication of the steel scrap's quantity used in the steel production process is depicted in Figure 9. Recent research addressing this optimization challenge, focusing on CO_2 emissions and embodied energy as sustainability criteria [27, 5, 26], as well as the LCA of SCCB [36], suggest that the environmental and social impacts of steel are not linked to yield stress. Instead, they primarily depend on the volume of steel scrap utilized during manufacturing. In contrast, the cost is closely related to yield strength. This is attributed to the prevalent yield stress of commercial profiles being 275 MPa. The demand for steels with a higher yield strength is less, leading to reduced production and an increased cost. This notable distinction is illustrated in Figure 9, where the correlation between cost reduction and a decrease in ELCA and SLCA is evident, though the reverse isn't necessarily true. These findings align with the outcomes presented by Martínez-Muñoz et al. [5, 27], reinforcing the notion that CO_2 emissions and embodied energy can serve as accurate indicators of environmental sustainability. A comparison of the top individual outcomes revealed that ELCA and SLCA result in solutions with superior yield stress compared to cost. To derive a balanced solution, it would be pertinent to employ a multi-objective optimization approach, a direction worth exploring in subsequent studies.

The results of the best individuals obtained through metamodel-assisted optimizations are displayed in Table 8. These are the best feasible individuals selected from 100 algorithm runs. The primary difference lies in the yield stress values, which can be observed in the table. Higher values are exhibited by the best individuals for ELCA and SLCA since there is no penalty for increasing resistance in the objective function. Although the steel distribution across the cross-section may differ, the total material amount remains unchanged. These results can be compared to those obtained in previous studies by Martínez-Muñoz et al. [5, 27] that consider CO_2 and embodied energy as environmental impact indicators. Furthermore, a comparison with recent SCCB optimization studies indicates that the number of stiffeners in the lower flange is reduced to zero in this optimization problem. However, this outcome is heavily dependent on the chosen construction method.

Table 8: Best solutions obtained for cost, ELCA, and SLCA objective functions

Variables	**Unit**	**Cost**	**ELCA**	**SLCA**
b	m	7	7	7
α_w	deg	64	71	73
h_s	mm	200	200	200
h_b	cm	255	262	363
h_{fb}	mm	400	590	530
t_{f_1}	mm	25	25	25
b_{f_1}	mm	300	300	300
h_{c_1}	mm	690	430	370
t_{c_1}	mm	16	16	16
t_w	mm	16	16	16
h_{c_2}	mm	840	0	0
t_{c_2}	mm	18	22	19
b_{c_2}	mm	300	300	300
t_{f_2}	mm	25	25	25
h_{s_2}	mm	150	150	150
$n_{s_{f_2}}$	u	0	0	0
d_{st}	m	3.7	2.6	1
d_{sd}	m	5.7	6.3	4
b_{fb}	mm	500	900	500
$t_{f_{fb}}$	mm	29	26	30
$t_{w_{fb}}$	mm	27	31	25
n_{r_1}	u	200	200	200
n_{r_2}	u	204	200	200
ϕ_{base}	mm	6	6	6
ϕ_{r_1}	mm	6	6	6
ϕ_{r_2}	mm	6	6	6
$s_{f_2}{}^*$	mm	300	500	450
$s_w{}^*$	mm	300	360	240
$s_t{}^*$	mm	360	600	400
h_{sc}	mm	100	100	100
ϕ_{sc}	mm	19	22	16
f_{ck}	MPa	25	25	25
f_{yk}	MPa	275	460	355
f_{sk}	MPa	500	500	500
Structural steel	kg	2,060,892	2,060,892	2,060,892
Reinforcement steel	kg	56,271	56,239	56,239
Concrete	m^3	528	528	528

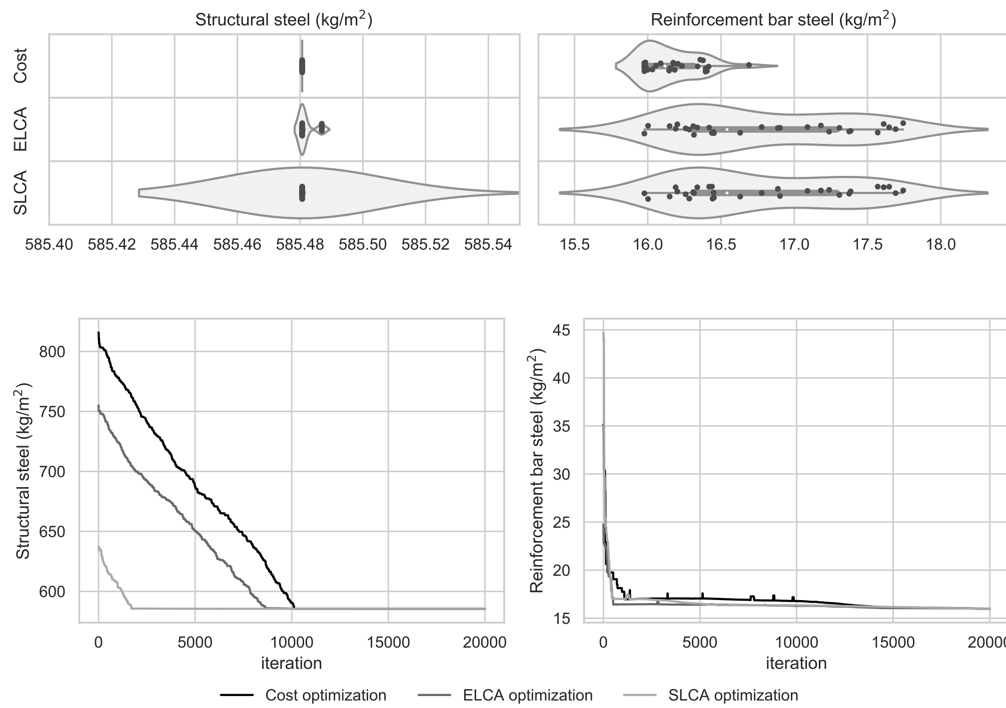

Figure 8: Steel amounts results in trajectories for Cost, ELCA and SLCA objective functions

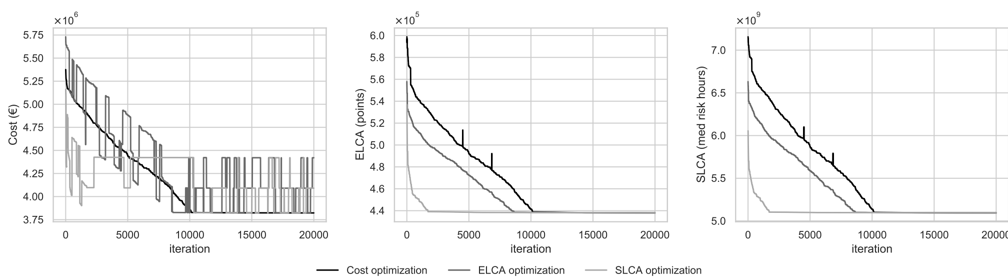

Figure 9: Cost, ELCA, and SLCA variation for every objective function

4. Conclusions

Incorporating the deep learning model to identify compliance with the hybrid bridge's regulations led to a substantial acceleration of calculations across the evaluated metaheuristics. Specifically for OBAMO, the acceleration factor was 38.18 times. For CS and SCA, the impact was even more pronounced, with rates of 50.93 and 49.71, respectively. Moreover, regarding solution quality, it was observed that for OBAMO, the results were enhanced on average; the solution with deep learning improved by 0.94%. For CS and CSA, the improvements were 0.17% and 0.11%, respectively.

In the context of this research, a deep neural network metamodel was integrated to expedite the optimization of an SCCB. The optimization and performance assessments were carried out utilizing the SCA, CS, and OBAMO algorithms. The neural network model adopted in this investigation manifested significant elevations in optimization velocity, spanning between 37 to 50 times swifter than conventional approaches. Notably, while the neural network model occasionally yielded non-feasible solutions, the heightened calculation speed rendered such discrepancies tolerable.

Additionally, when using the validation model in the optimization process, more feasible results were obtained for ELCA and SLCA due to the higher steel yield stress. However, since the environmental and social impact of the design is independent of the yield stress, solutions considering these as objective functions resulted in higher yield stress.

In general, the solutions obtained using different objective functions consistently involved the use of cells in the bridge's cross-section. This study suggests that deep learning models have immense potential in optimizing complex engineering designs, particularly in reducing the computational time required for optimization. However, the trade-off between speed and accuracy needs to be carefully considered in practical applications. Future work will apply this DL acceleration to multi-objective and robust optimization techniques to derive more comprehensive design solutions. Additionally, there is an interest in exploring other machine learning techniques, such as Support Vector Machine and the Gaussian process. Notably, these techniques have been applied to structural problems as highlighted in [53, 54]. Furthermore, probing the methodology's applicability to varied types of structural design problems becomes essential to assess its universality.

Acknowledgments

The authors gratefully acknowledge the funding received from the following research projects:

- Grant PID2020-117056RB-I00 funded by MCIN/AEI/10.13039/501100011033 and by "ERDF A way of making Europe".

- Grant FPU-18/01592 funded by MCIN/AEI/10.13039/501100011033 and by "ESF invests in your future"

- Grant PROYECTO DI REGULAR: 039.300/2023.

References

[1] D. M. Frangopol, Life-cycle performance, management, and optimisation of structural systems under uncertainty: accomplishments and challenges 1, Structure and infrastructure Engineering 7 (6) (2011) 389–413.

[2] T. Ramesh, R. Prakash, K. Shukla, Life cycle energy analysis of buildings: An overview, Energy and buildings 42 (10) (2010) 1592–1600.

[3] A. Serpell, J. Kort, S. Vera, Awareness, actions, drivers and barriers of sustainable construction in chile, Technological and Economic Development of Economy 19 (2) (2013) 272–288.

[4] N. Yusof, N. Z. Abidin, S. H. M. Zailani, K. Govindan, M. Iranmanesh, Linking the environmental practice of construction firms and the environmental behaviour of practitioners in construction projects, Journal of Cleaner Production 121 (2016) 64–71.

[5] D. Martínez-Muñoz, J. García, J. V. Martí, V. Yepes, Hybrid swarm intelligence optimization methods for low-embodied energy steel-concrete composite bridges, Mathematics 11 (1) (2023) 140.

[6] J. M. Barandica, G. Fernández-Sánchez, Á. Berzosa, J. A. Delgado, F. J. Acosta, Applying life cycle thinking to reduce greenhouse gas emissions from road projects, Journal of cleaner production 57 (2013) 79–91.

[7] T. Wang, I.-S. Lee, A. Kendall, J. Harvey, E.-B. Lee, C. Kim, Life cycle energy consumption and ghg emission from pavement rehabilitation with different rolling resistance, Journal of Cleaner Production 33 (2012) 86–96.

[8] E. Wang, Z. Shen, A hybrid data quality indicator and statistical method for improving uncertainty analysis in lca of complex system–application to the whole-building embodied energy analysis, Journal of cleaner production 43 (2013) 166–173.

[9] I. J. Navarro, V. Yepes, J. V. Martí, F. González-Vidosa, Life cycle impact assessment of corrosion preventive designs applied to prestressed concrete bridge decks, Journal of Cleaner Production 196 (2018) 698–713.

[10] J. J. Pons, V. Penadés-Plà, V. Yepes, J. V. Martí, Life cycle assessment of earth-retaining walls: An environmental comparison, Journal of Cleaner Production 192 (2018) 411–420.

[11] Y. Yılmaz, S. Seyis, Mapping the scientific research of the life cycle assessment in the construction industry: A scientometric analysis, Building and Environment 204 (2021) 108086.

[12] Y. Zhang, Y. Mao, L. Jiao, C. Shuai, H. Zhang, Eco-efficiency, eco-technology innovation and eco-well-being performance to improve global sustainable development, Environmental Impact Assessment Review 89 (2021) 106580.

[13] ISO, Environmental Management, Life Cycle Assessment Principles and Framework (ISO 14040:2006), International Organization for Standardization, 2006.

[14] M. S. Phadke, Quality engineering using robust design, Prentice Hall PTR, 1995.

[15] G. Taguchi, Introduction to quality engineering, asian productivity organization, Dearborn, Michigan: American Supplier Institute Inc (1986).

[16] V. Penadés-Plà, T. García-Segura, V. Yepes, Accelerated optimization method for low-embodied energy concrete box-girder bridge design, Engineering Structures 179 (2019) 556–565.

[17] M. H. Esfe, P. Razi, M. H. Hajmohammad, S. H. Rostamian, W. S. Sarsam, A. A. A. Arani, M. Dahari, Optimization, modeling and accurate prediction of thermal conductivity and dynamic viscosity of stabilized ethylene glycol and water mixture al2o3 nanofluids by nsga-ii using ann, International Communications in Heat and Mass Transfer 82 (2017) 154–160.

[18] J. R. Marti-Vargas, F. J. Ferri, V. Yepes, Prediction of the transfer length of prestressing strands with neural networks, Computers and Concrete 12 (2) (2013) 187–209.

[19] P. Satrio, T. M. I. Mahlia, N. Giannetti, K. Saito, et al., Optimization of hvac system energy consumption in a building using artificial neural network and multi-objective genetic algorithm, Sustainable Energy Technologies and Assessments 35 (2019) 48–57.

[20] G. Van Rossum, F. L. Drake, Python 3 Reference Manual, CreateSpace, Scotts Valley, CA, 2009.

[21] F. J. Martínez-Martín, V. Yepes, F. González-Vidosa, A. Hospitaler, J. Alcalá, Optimization design of RC elevated water tanks under seismic loads, Applied Sciences 12 (11) (2022) 5635.

[22] T. C. Hu, A. B. Kahng, C.-W. A. Tsao, Old bachelor acceptance: A new class of non-monotone threshold accepting methods, ORSA Journal on Computing 7 (4) (1995) 417–425.

[23] S. Kirkpatrick, C. D. J. Gelatt, M. P. Vecchi, Optimization by simulated annealing, Science 220 (4598) (1983) 671–680.

[24] A. Ruiz-Vélez, J. Alcalá, V. Yepes, Optimal design of sustainable reinforced concrete precast hinged frames, Materials 16 (1) (2023) 204.

[25] D. C. Montgomery, Design and analysis of experiments, John Wiley & Sons, Hoboken, 2013.

[26] D. Martínez-Muñoz, J. García, J. V. Martí, V. Yepes, Discrete swarm intelligence optimization algorithms applied to steel–concrete composite bridges, Engineering Structures 266 (2022) 114607.

[27] D. Martínez-Muñoz, J. García, J. V. Martí, V. Yepes, Optimal design of steel–concrete composite bridge based on a transfer function discrete swarm intelligence algorithm, Structural and Multidisciplinary Optimization 65 (11) (2022) 312.

[28] S. Mirjalili, Sca: a sine cosine algorithm for solving optimization problems, Knowledge-based systems 96 (2016) 120–133.

[29] Catalonia Institute of Construction Technology. BEDEC ITEC Materials Database https://metabase.itec.cat/vide/es/bedec, accessed on January 2021.

[30] C. Benoît, B. Mazijn, Guidelines for Social Life Cycle Assessment of Products, Vol. 15, UNEP/SETAC Life Cycle Initiative,Sustainable Product and Consumption Branch, Paris, France, 2011.

[31] M. Goedkoop, R. Heijungs, M. Huijbregts, A. De Schryver, J. Struijs, R. Van Zelm, ReCiPe 2008. Report I: Characterisation, Ministry of Housing, Spatial planning and Environment (VROM), 2009.

[32] R. Frischknecht, G. Rebitzer, The ecoinvent database system: a comprehensive web-based lca database, Journal of Cleaner Production 13(13) (2005) 1337–1343.

[33] GreenDelta GmbH, Soca v. 2 add-on: adding social impact information to ecoinvent, Description of methodology to map social impact information from PSILCA v3 to ecoinvent v. 3.7.1 (2021).

[34] J. Pascual-González, G. Guillén-Gosálbez, J. M. Mateo-Sanz, L. Jiménez-Esteller, Statistical analysis of the ecoinvent database to uncover relationships between life cycle impact assessment metrics, Journal of Cleaner Production 112 (2016) 359–368.

[35] A. Ciroth, F. Eisfeldt, PSILCA – A product social impact life cycle assessment database, Database version 1 (2016) 1–99.

[36] D. Martínez-Muñoz, J. V. Martí, V. Yepes, Social impact assessment comparison of composite and concrete bridge alternatives, Sustainability 14 (9) (2022) 5186.

[37] SRI, Construction | SRI - Steel Recycling Institute https://www.steelsustainability.org/construction, accessed on 30 January 2021.

[38] F. Collins, Inclusion of carbonation during the life cycle of built and recycled concrete: influence on their carbon footprint, The International Journal of Life Cycle Assessment 15(6) (2010) 549–556.

[39] A. Dodoo, L. Gustavsson, R. Sathre, Carbon implications of end-of-life management of building materials, Resources, Conservation and Recycling 53(5) (2009) 276–286.

[40] T. García-Segura, V. Yepes, J. Alcalá, Life cycle greenhouse gas emissions of blended cement concrete including carbonation and durability, The International Journal of Life Cycle Assessment 19 (2014) 3–12.

[41] A. Ciroth, ICT for environment in life cycle applications openlca - a new open source software for life cycle assessment, The International Journal of Life Cycle Assessment 12 (2007) 209.

[42] CEN, EN 10365:2017: Hot rolled steel channels, I and H sections. Dimensions and masses, European Committee for Standardization, Brussels, Belgium, 2017.

[43] I. Vayas, A. Iliopoulos, Design of steel-concrete composite bridges to Eurocodes, CRC Press, Boca Raton, 2017.

[44] S. Monleón, Diseño estructural de puentes (in Spanish), Universitat Politècnica de València, València, 2017.

[45] CEN, Eurocode 2: Design of concrete structures, European Committee for Standardization, Brussels, Belgium, 2013.

[46] CEN, Eurocode 1: Actions on structures, European Committee for Standardization, Brussels, Belgium, 2019.

[47] CEN, Eurocode 4: Design of composite steel and concrete structures, European Committee for Standardization, Brussels, Belgium, 2013.

[48] CEN, Eurocode 3: Design of steel structures, European Committee for Standardization, Brussels, Belgium, 2013.

[49] MFOM, IAP-11: Code on the actions for the design of road bridges, Ministerio de Fomento, Madrid, 2011.

[50] F. Murtagh, Multilayer perceptrons for classification and regression, Neurocomputing 2 (5-6) (1991) 183–197.

[51] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, Smote: synthetic minority over-sampling technique, Journal of artificial intelligence research 16 (2002) 321–357.

[52] D. Martínez-Muñoz, J. García, J. Martí, V. Yepes, Discrete swarm intelligence optimization algorithms applied to steel–concrete composite bridges, Engineering Structures 266 (2022) 114607.

[53] A. Garg, M.-O. Belarbi, A. Tounsi, L. Li, A. Singh, T. Mukhopadhyay, Predicting elemental stiffness matrix of fg nanoplates using gaussian process regression based surrogate model in framework of layerwise model, Engineering Analysis with Boundary Elements 143 (2022) 779–795.

[54] A. Garg, P. Aggarwal, Y. Aggarwal, M. Belarbi, H. Chalak, A. Tounsi, R. Gulia, Machine learning models for predicting the compressive strength of concrete containing nano silica, Computers and Concrete 30 (1) (2022) 33.