

## Article

# Prediction of Mechanical Properties of Bolted Connections in CFST Column–Steel Beam Assemblies Based on Improved Particle Swarm Optimization and Deep Neural Networks

Yurong Yao and Liang Zhang \*

Gold Mantis School of Architecture, Soochow University, Suzhou 215000, China; 18652698225@163.com

\* Correspondence: lzhang@suda.edu.cn

## Abstract

Predicting the mechanical properties of bolted connection nodes in prefabricated Concrete-Filled Steel Tube (CFST) column–steel beam assemblies remains challenging due to complex nonlinear relationships, high degrees of parameter coupling, and limited generalization capabilities of traditional empirical formulas. This study proposes a data-driven prediction model integrating an Improved Particle Swarm Optimization (IPSO) algorithm with a Deep Neural Network (DNN). Drawing upon 196 sets of experimental data on CFST column–steel beam nodes with Extended Hollo-Bolt (EHB) connections from the published literature, the model employs bolt diameter, steel tube wall thickness, concrete compressive strength, beam–column cross-sectional parameters, and connection configuration parameters as input variables, while designating ultimate moment capacity, initial stiffness, and joint ductility coefficient as prediction targets. A multi-layer DNN is constructed to capture the highly nonlinear mapping between structural parameters and mechanical responses. The IPSO algorithm, enhanced with adaptive inertia weight and Lévy flight perturbation, performs global optimization of the network weights and hyperparameters to improve convergence speed and prediction stability. Five-fold cross-validation is embedded within the IPSO fitness evaluation loop to guide hyperparameter selection, while dropout regularization and early stopping are applied during final training to mitigate overfitting; prediction performance is ultimately verified on an independent hold-out test set. Experimental results demonstrate that the proposed IPSO-DNN model outperforms a tuned shallow neural network (SNN), Support Vector Regression (SVR), and Random Forest (RF) models across the coefficient of determination ( $R^2$ ), root mean square error (RMSE), and mean absolute error (MAE), effectively capturing the nonlinear mechanical characteristics of CFST nodes under complex loading conditions.

**Keywords:** CFST column–steel beam connection; deep neural network; improved particle swarm optimization; mechanical-property prediction; Extended Hollo-Bolt; data-driven model

**MSC:** 13P25; 49N30; 37N40



Academic Editor: David Greiner

Received: 2 June 2026

Revised: 28 June 2026

Accepted: 13 July 2026

Published: 3 August 2026

**Copyright:** © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

## 1. Introduction

Prefabricated steel structures rely on reliable beam–column joints because much of the fabrication and quality control is transferred from the construction site to the factory [1,2]. Concrete-Filled Steel Tube (CFST) columns connected to steel beams through bolted joints are widely used in multi-story and high-rise buildings [3–5]. The steel tube

confines the concrete core, while the concrete delays local buckling of the tube wall. This composite action can provide high bearing capacity and good ductility, but these advantages depend on whether the joint region can transfer moment, shear, and axial effects under static and seismic loading [6–8].

Extended Hollo-Bolt (EHB) connections provide a practical blind-bolting solution for closed tubular columns because installation does not require access to the inside of the tube [9–13]. This feature is useful for prefabricated construction, where fast on-site assembly and inspectable connection details are important. In an EHB end-plate joint, bolt expansion, tube-face bending, end-plate bearing, and beam flange deformation occur together. The moment–rotation response is therefore controlled by interacting geometric and material variables, including bolt diameter, bolt grade, tube wall thickness, end-plate dimensions, concrete strength, and beam geometry.

Although EHB and anchored blind-bolt joints have been tested under monotonic and cyclic loading [14–22], their mechanical-property prediction still relies largely on semi-empirical formulas and component-based models. The Eurocode 3 component method [23] decomposes the joint into springs representing individual deformable components, such as the column face in bending, bolts in tension, and the end plate in bending [24], and assembles their stiffnesses to estimate joint resistance and initial rotational stiffness. For blind-bolted CFST joints, researchers have proposed dedicated component springs for tube-face deformation and blind-bolt anchorage. Tizani et al. [10] calibrated an equivalent tube-face spring stiffness for the modified Hollo-Bolt and demonstrated that the component model prediction agrees well with test results when the tube-face deformation component is correctly characterized. Pitrakos and Tizani [9] further proposed a force–deformation model for the anchored blind bolt in tension, providing a mechanistic basis for its inclusion in the component method. Wang et al. [14] extended the component approach to flush end-plate joints connected to CFST columns and reported prediction-to-test scatter for joint moment resistance. Ataei et al. [2] developed a computational moment–rotation model for composite flush end-plate joints, calibrated against deconstructable beam-to-column test results. Oktavianus et al. [25] proposed a pull-out component model for headed anchored blind bolts in circular CFST columns, combining cone-breakout and bearing resistance formulations. Li et al. [26] reviewed the component-based and empirical approaches available for blind-bolted connections and identified that tube-face deformation is the dominant source of uncertainty across different bolt types and column configurations. These methods are convenient for design interpretation, but they usually simplify the coupled effects among local tube deformation, bolt force transfer, end-plate bending, and material strength. As a result, they may lose accuracy when several response quantities, such as ultimate capacity, initial stiffness, and ductility, are predicted at the same time [25,27].

Machine learning (ML) and deep learning (DL) methods offer an alternative route for connection-property prediction. Previous studies have used Artificial Neural Networks (ANNs) for steel connection moment–rotation response [28], Support Vector Machines (SVMs) for reinforced concrete shear capacity [29,30], and ensemble or probabilistic models for CFST and RCFST member capacity [31–35]. Recent ML-based structural resistance studies have also shown that data-driven models can outperform conventional analytical equations when the target response is governed by interacting geometric and material variables, such as web-post buckling resistance in high-strength steel beams with elliptically based web openings [36]. For blind-bolted CFST joints, however, shallow models may approximate the average trend but miss localized effects such as tube-face yielding, prying action, and bolt anchorage deformation. Deep Neural Networks (DNNs) can represent such nonlinear interactions [37,38], but their training is sensitive to initial weights and local optima when the experimental database is limited [39].

Particle Swarm Optimization (PSO) has been used to improve neural network training in engineering applications [40–43]. By searching the weight and hyperparameter space before final training, PSO can reduce the dependence of a DNN on random initialization. Standard PSO, however, can converge prematurely in high-dimensional search spaces [44]. This study therefore introduces an Improved PSO (IPSO) strategy that combines adaptive inertia weight with Lévy flight perturbation. The adaptive weight adjusts the exploration–exploitation balance according to iteration progress and population diversity, while the perturbation mechanism helps stagnant particles leave local search regions.

Despite the experimental and component-based efforts reviewed above, a unified, mechanically interpretable prediction framework capable of simultaneously estimating strength, stiffness, and ductility for EHB-connected CFST column–steel beam joints is not yet available in the literature. Previous studies have clarified important local mechanisms, such as tube-face deformation, prying action, and blind-bolt anchorage, but most available design-oriented models still treat these mechanisms through simplified component assumptions or focus on a single response quantity. Existing ML applications in structural engineering also seldom target EHB-connected CFST joints with a database organized around both global member dimensions and local connection parameters.

Accordingly, the main contributions are as follows. A database of 196 EHB-connected CFST column–steel beam specimens is compiled from published tests, and 11 mechanically meaningful input variables are used to predict ultimate moment capacity, initial stiffness, and ductility coefficient. An IPSO-DNN model is developed to optimize initial weights and selected hyperparameters before final DNN training, which is intended to reduce the sensitivity of deep learning to random initialization under a limited experimental database. The model is compared with a tuned shallow neural network (SNN), SVR, Random Forest, and code-oriented design references under consistent evaluation conditions, and the results are interpreted through visualization, residual analysis, partial dependence analysis, and feature-importance analysis.

## 2. Methodology

### 2.1. Data Collection and Preprocessing

The experimental database employed in this study comprises 196 specimens of CFST column–steel beam connections with EHB or anchored blind-bolted joints, collected from published experimental studies on flush end-plate joints, extended end-plate joints, T-stub components, and strengthened blind-bolted CFST connections [1,6,7,9,10,14,15,17,45–48]. Only specimens with clearly reported geometric dimensions, material strengths, connection details, and mechanical response curves or tabulated response values were retained. When the same test program reported repeated specimens, the repeated data were checked for consistency before inclusion so that the model would not be biased by duplicated or incomplete records. The input parameters encompass geometric dimensions, material properties, and connection configurations, as summarized in Table 1.

The selected variables were chosen to represent both the global member scale and the local connection scale. Bolt diameter, bolt grade, bolt-row number, and bolt gauge describe the principal fastener characteristics and the distribution of tensile forces in the connection. Tube wall thickness, column width, and concrete compressive strength describe the stiffness and strength of the CFST column face, which is the region most directly affected by blind-bolt anchorage. Beam depth, beam flange thickness, end-plate thickness, and end-plate width represent the load path from the steel beam to the bolted joint. This organization avoids the use of purely statistical descriptors without mechanical meaning and helps the trained model preserve interpretability.

**Table 1.** Summary of input and output parameters for the IPSO-DNN model, including descriptive statistics computed from the 196-specimen database.

| Category          | Parameter                     | Symbol      | Range       | Mean   | SD     | Unit     |
|-------------------|-------------------------------|-------------|-------------|--------|--------|----------|
| Bolt parameters   | Bolt diameter                 | $d_b$       | 16–24       | 19.8   | 2.1    | mm       |
|                   | Bolt grade                    | $G_b$       | 8.8–10.9    | 9.2    | 0.7    | –        |
|                   | Number of bolt rows           | $n_r$       | 2–4         | 2.8    | 0.6    | –        |
|                   | Bolt gauge                    | $g_b$       | 90–140      | 108    | 13     | mm       |
| Column parameters | Steel tube wall thickness     | $t_c$       | 5–12.5      | 8.5    | 1.9    | mm       |
|                   | Column width                  | $B_c$       | 150–400     | 248    | 65     | mm       |
|                   | Concrete compressive strength | $f_{cu}$    | 30–60       | 41     | 8      | MPa      |
| Beam parameters   | Beam depth                    | $h_b$       | 200–500     | 320    | 78     | mm       |
|                   | Beam flange thickness         | $t_{fb}$    | 8–20        | 13.2   | 3.1    | mm       |
|                   | End-plate thickness           | $t_p$       | 10–25       | 16.8   | 4.0    | mm       |
|                   | End-plate width               | $b_p$       | 150–300     | 215    | 38     | mm       |
| Output targets    | Ultimate moment capacity      | $M_u$       | 45–680      | 285    | 142    | kN·m     |
|                   | Initial stiffness             | $S_{j,ini}$ | 5200–89,000 | 38,500 | 18,200 | kN·m/rad |
|                   | Ductility coefficient         | $\mu$       | 1.8–6.5     | 3.7    | 1.1    | –        |

The number of data records obtained from each reference source is summarized in Table 2. The distribution is not uniform because the original studies differ in test scope, reporting completeness, and whether the response quantities required for the present three-output model were available. The complete processed data file also retains the source identifier for every specimen so that the per-source composition can be inspected directly.

**Table 2.** Distribution of experimental data collected from published references.

| Reference Source                | Connection or Test Type                                             | Number of Records |
|---------------------------------|---------------------------------------------------------------------|-------------------|
| Wang et al. (2009) [14]         | Flush end-plate joints to CFST columns                              | 18                |
| Wang et al. (2013) [7]          | Extended end-plate joints under seismic loading                     | 24                |
| Pitrakkos and Tizani (2013) [9] | Anchored blind-bolt tensile tests                                   | 16                |
| Tizani et al. (2013) [10,15]    | Rotational and hysteretic blind-bolted connections                  | 27                |
| Tao et al. (2017) [6]           | Blind-bolted connections to concrete-filled stainless steel columns | 19                |
| Ataei et al. (2015) [1]         | Composite flush end-plate joints                                    | 21                |
| Agheshlui et al. (2016) [45,46] | Single and grouped anchored blind bolts in CFST columns             | 23                |
| Wang et al. (2020) [47,48]      | Anchored and strengthened blind-bolted joints                       | 20                |
| Cabrera et al. (2024) [17]      | Open EHB tensile experimental dataset                               | 28                |
| Total                           | All source groups                                                   | 196               |

To improve database transparency and reproducibility, the processed dataset used for modeling contains the specimen identifier, literature source, all 11 input variables, and the three output responses for each record. The extraction procedure followed four steps: first, specimen geometry and material properties were copied from tables where available; second, missing geometric definitions were checked against published drawings; third, moment–rotation curves were digitized only when tabulated  $M_u$ ,  $S_{j,ini}$ , or  $\mu$  values were not directly reported; and fourth, all extracted records were cross-checked against the parameter ranges in Table 1. Data fields, source distribution, and extraction and preprocessing rules are provided in Supplementary Tables S1–S3. The processed database, source identifiers, data-extraction notes, and reproducible scripts are publicly available in the GitHub repository listed in the Data Availability Statement.

The proposed IPSO-DNN model should be regarded as valid only within the parameter ranges listed in Table 1. These ranges define the interpolation domain of the trained model; predictions outside them may involve different failure modes or parameter interactions that were not represented in the training data.

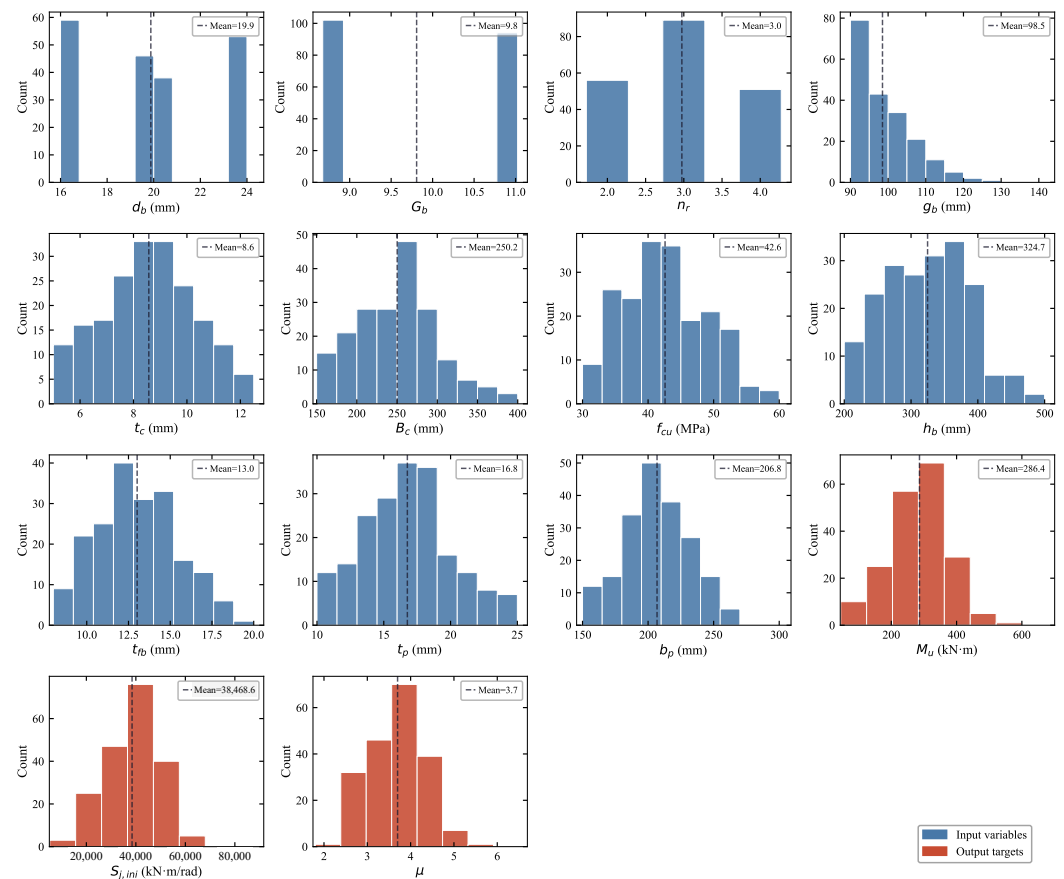
Prior to model training, all input features are normalized using min–max scaling to the interval  $[0, 1]$ :

$$x_i^* = \frac{x_i - x_{i,\min}}{x_{i,\max} - x_{i,\min}} \quad (1)$$

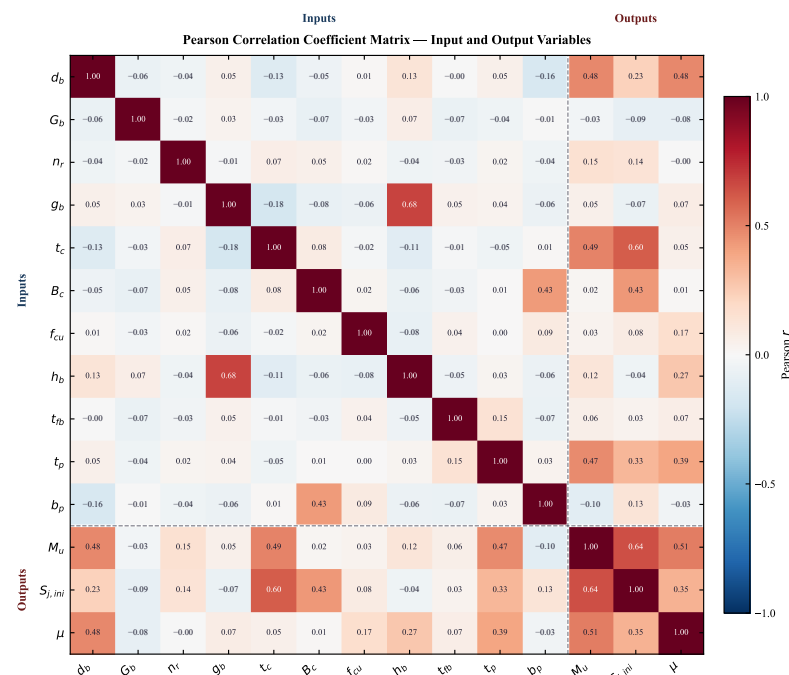
where  $x_i$  denotes the original value of the  $i$ -th feature, and  $x_{i,\min}$  and  $x_{i,\max}$  are the corresponding minimum and maximum values in the dataset. Min–max scaling was used because the variables differ substantially in magnitude, from dimensionless ductility values to stiffness values of tens of thousands of kN·m/rad. The dataset was partitioned into training (70%), validation (15%), and testing (15%) subsets using stratified random sampling based on discretized response intervals. This procedure helps each subset retain low-, medium-, and high-capacity specimens. Before splitting, the database was checked for missing values, repeated specimens, unit inconsistency, and curve-extracted responses. No record with missing input variables was retained. Repeated specimens from the same test program were kept only when they represented separate physical tests; otherwise, duplicate entries were removed. All length variables were converted to mm, concrete strength to MPa, moment to kN·m, and stiffness to kN·m/rad. For curve-extracted data, the digitized value was compared with any reported peak load or stiffness estimate in the original paper when such information was available.

Figure 1 presents frequency histograms for all 11 input variables and the three output targets. The bolt diameter ( $d_b$ ) and number of bolt rows ( $n_r$ ) are discrete and concentrate at a limited number of values, whereas most continuous variables, including tube wall thickness, column width, beam depth, and concrete strength, display approximately unimodal distributions spanning the full ranges reported in Table 1. The output distributions are more skewed: ultimate moment capacity ( $M_u$ ) shows a right-skewed distribution with a cluster of lower-capacity specimens and a tail extending toward high-capacity configurations; initial stiffness ( $S_{j,ini}$ ) exhibits a similar pattern, reflecting the concentration of standard joint sizes in the database; and the ductility coefficient ( $\mu$ ) is more symmetric but bounded by its physical definition. These distributional characteristics confirm that the collected database provides coverage across the parameter space rather than concentrating in a narrow region.

Figure 2 presents the Pearson correlation coefficient matrix between all input and output variables. Several moderate positive correlations are present among the geometric variables, particularly between beam depth ( $h_b$ ) and bolt gauge ( $g_b$ ) ( $r = 0.68$ ), and between column width ( $B_c$ ) and end-plate width ( $b_p$ ) ( $r = 0.43$ ), which reflects the fact that larger column cross-sections are typically paired with wider end plates in practical joint designs. Tube wall thickness ( $t_c$ ) shows the strongest positive correlations with both initial stiffness ( $S_{j,ini}$ ,  $r = 0.60$ ) and ultimate moment capacity ( $M_u$ ,  $r = 0.49$ ), consistent with the mechanical role of the column face in controlling connection response. End-plate thickness ( $t_p$ ) and bolt diameter ( $d_b$ ) are also positively correlated with  $M_u$  ( $r = 0.47$  and  $r = 0.48$ , respectively). Among all input variable pairs, the highest correlation is  $r = 0.68$  between  $h_b$  and  $g_b$ , which remains below the commonly adopted collinearity threshold of 0.70, confirming that severe multicollinearity is absent and that the 11 variables can be retained as independent predictors. The ductility coefficient ( $\mu$ ) shows weaker correlations with all inputs (absolute values mostly below 0.35), indicating that deformation capacity is governed by nonlinear interactions rather than individual parameters, which motivates the use of a deep nonlinear model.



**Figure 1.** Frequency histograms of all 11 input variables and three output targets in the 196-specimen database. Discrete variables ( $d_b$ ,  $n_r$ ,  $G_b$ ) are shown as bar charts; continuous variables are displayed as histograms with 10 bins.

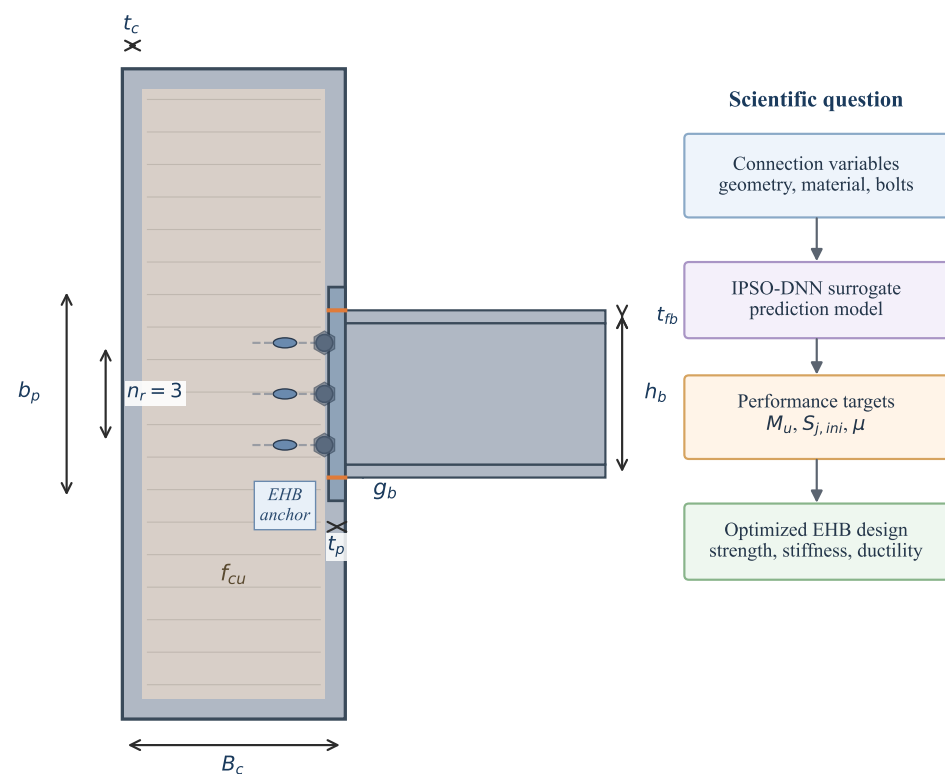


**Figure 2.** Pearson correlation coefficient heat map between all 11 input variables and three output targets. Colour scale ranges from  $-1$  (strong negative correlation, dark blue) to  $+1$  (strong positive correlation, dark red). Values printed in each cell are rounded to two decimal places. The dashed horizontal and vertical lines separate the input-variable block from the output-variable block.

To reduce the risk of data leakage from similar specimens in the same experimental program, an additional source-wise evaluation was performed. In this evaluation, all records from one literature source were held out as an external test group, while the remaining sources were used for training and validation. This leave-one-source-out procedure is stricter than a random split because specimens from the same experimental campaign cannot appear simultaneously in the training and test sets.

## 2.2. Feature Engineering and Response Definition

Figure 3 provides a schematic of the bolted CFST column–steel beam assembly considered in this study, identifying all 11 input parameters used in the model. The connection consists of a square or rectangular CFST column, a steel I-beam framing into the column via an extended end plate, and EHB blind bolts that pass through the end plate and expand inside the steel tube wall to engage the concrete core. The bolt gauge  $g_b$  denotes the horizontal distance between bolt centres at each row,  $n_r$  is the number of bolt rows along the beam depth, and  $d_b$  is the nominal bolt shank diameter. The tube wall thickness  $t_c$  and outer column width  $B_c$  define the geometry of the CFST column face in the connection region. Beam depth  $h_b$  and beam flange thickness  $t_{fb}$  characterize the steel beam cross-section, while end-plate thickness  $t_p$  and end-plate width  $b_p$  define the connection plate welded to the beam end. Concrete compressive strength  $f_{cu}$  and bolt grade  $G_b$  complete the material-property inputs.



**Figure 3.** Schematic of the bolted CFST column–steel beam assembly with Extended Hollo-Bolt (EHB) connections, illustrating all 11 input parameters used in the present model: bolt diameter  $d_b$ , bolt grade  $G_b$ , number of bolt rows  $n_r$ , bolt gauge  $g_b$ , tube wall thickness  $t_c$ , column width  $B_c$ , concrete compressive strength  $f_{cu}$ , beam depth  $h_b$ , beam flange thickness  $t_{fb}$ , end-plate thickness  $t_p$ , and end-plate width  $b_p$ . Grey denotes structural steel, beige denotes the concrete core, blue-grey denotes the end plate and EHB components, orange marks the beam-to-plate welds, and double-headed lines denote dimensions. Arrows in the right-hand panel indicate the sequence from input variables to model predictions and design targets.

The raw experimental information was screened before modeling. Because different studies report end-plate dimensions, bolt spacing, and tube cross-section width using different conventions, all geometric variables—specifically  $d_b$ ,  $g_b$ ,  $t_c$ ,  $B_c$ ,  $h_b$ ,  $t_{fb}$ ,  $t_p$ , and  $b_p$ —were converted to millimetres following the definitions illustrated in Figure 3 and checked against specimen drawings or test-setup photographs where available [16,26]. Measured material strengths were preferred over nominal values. For concrete strength, cube compressive strength was retained when reported directly; otherwise, cylinder strength was converted according to the convention used in the corresponding reference.

The three output quantities were selected because they describe different stages of connection behavior. The ultimate moment capacity  $M_u$  corresponds to the peak moment obtained from the moment–rotation curve or from the maximum load converted through the beam lever arm. It reflects the resistance of the joint after local yielding, bolt tension, and plate deformation have developed. The initial stiffness  $S_{j,ini}$  was taken as the slope of the early linear portion of the moment–rotation curve:

$$S_{j,ini} = \frac{\Delta M}{\Delta \theta} \quad (2)$$

where  $\Delta M$  and  $\Delta \theta$  denote the moment and rotation increments within the initial elastic range. When a study reported stiffness directly, the reported value was used after unit conversion; otherwise, it was extracted from the published curve using the initial linear segment. This response is sensitive to bolt slip, end-plate contact, and local flexibility of the column face, so it provides information that is not fully represented by strength alone.

The ductility coefficient  $\mu$  was defined as the ratio between ultimate rotation and yield rotation:

$$\mu = \frac{\theta_u}{\theta_y} \quad (3)$$

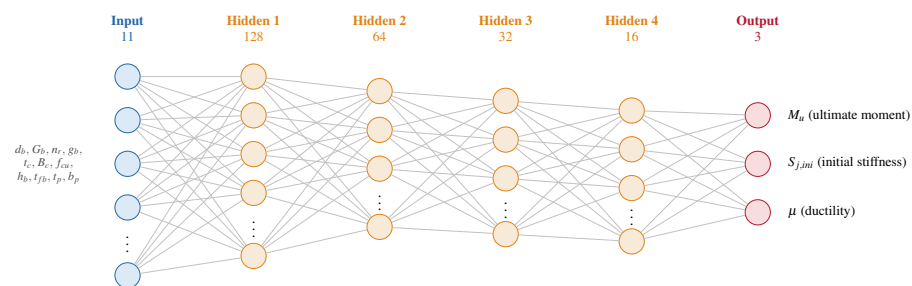
where  $\theta_u$  is the rotation corresponding to the post-peak or specified ultimate state, and  $\theta_y$  is the equivalent yield rotation determined from the idealized moment–rotation response. Because yield-point definitions vary across experimental studies, ductility data were checked carefully before being entered into the database. Specimens with ambiguous or incomplete deformation records were excluded from the final database even when their strength values were available. This conservative treatment keeps the three prediction targets defined on the same curated sample set and improves the reliability of the output labels.

The final 11 input variables were selected from an initial pool of candidate parameters through a systematic procedure that combined mechanical reasoning with a statistical collinearity check. First, candidate variables were retained if they directly govern at least one identified resistance or deformation component of the EHB-connected CFST joint, namely tube-face bending, blind-bolt anchorage, end-plate bending, bolt tensile resistance, or beam flange force transfer. Second, the Variance Inflation Factor (VIF) was computed for each candidate in a preliminary regression; variables with VIF exceeding 5.0 were considered potentially redundant and re-examined. The 11 variables retained in Table 1 all have VIF values below 3.8, indicating an acceptable level of linear independence within this dataset. Third, the Pearson correlation matrix in Figure 2 was inspected to confirm that no pair of retained variables exceeded a correlation coefficient of 0.70 in absolute value. Bolt grade  $G_b$  is treated as a continuous variable because the two values present in the database (8.8 and 10.9) represent a physically meaningful stiffness and strength contrast that the network can exploit. The nonlinear interactions among these 11 primary variables are captured by the hidden layers of the DNN rather than being pre-coded into derived ratios, which would introduce additional multicollinearity without adding mechanistic information beyond what the primary variables already contain.

### 2.3. Deep Neural Network Architecture

The DNN topology was determined through a structured grid search before IPSO optimization was applied. Candidate architectures were defined by three factors: the number of hidden layers (2, 3, or 4), the width of the first hidden layer (64, 128, or 256 neurons), and the layer-reduction pattern (uniform or pyramidal). For each candidate, a standalone DNN was trained with fixed hyperparameters (learning rate  $10^{-3}$ , dropout 0.2, batch size 16) using five-fold cross-validation, and the mean validation error across the three output targets was used to rank the architectures. The four-hidden-layer pyramidal configuration (128-64-32-16) achieved a mean validation  $R^2$  of 0.936, compared with 0.921 for the three-layer (128-64-32) and 0.908 for the two-layer (128-64) architectures. A uniform four-layer network (64-64-64-64) reached 0.928 but showed larger fold-to-fold variation. The selected four-layer pyramidal architecture was therefore used as the fixed topology for both the standalone DNN baseline and the IPSO-DNN, ensuring that the subsequent IPSO optimization operates on a topology selected by a documented search rather than arbitrarily assumed.

The DNN employed in this study consists of an input layer with 11 neurons (corresponding to the input parameters), four hidden layers, and an output layer with three neurons (corresponding to  $M_u$ ,  $S_{j,ini}$ , and  $\mu$ ). The overall network topology is illustrated in Figure 4.



**Figure 4.** Architecture of the Deep Neural Network with four hidden layers. The network receives 11 structural parameters as inputs and predicts three mechanical properties.

The hidden layer configuration follows a pyramidal structure with neuron counts of 128, 64, 32, and 16. This structure gives the first hidden layer enough width to represent interactions among the 11 normalized inputs, while the later layers progressively compress the information needed for the three response predictions. Each hidden layer employs the Rectified Linear Unit (ReLU) activation function:

$$f(z) = \max(0, z) \quad (4)$$

The output layer uses a linear activation function for the three continuous targets. Dropout regularization with a rate of 0.2 is applied to the first two hidden layers, and batch normalization is incorporated after each hidden layer to stabilize training. The standalone DNN benchmark uses the same four-hidden-layer pyramidal topology and the same regularization settings (dropout 0.2, batch normalization, early stopping), trained with the Adam optimizer at a fixed learning rate of  $10^{-3}$  selected by the grid search described above. This ensures that the performance gain of IPSO-DNN over the standalone DNN is attributable to the IPSO-based weight initialization and hyperparameter optimization rather than to architectural or regularization differences. Because the database contains 196 specimens, the four-hidden-layer architecture could increase the risk of overfitting if it were trained without constraints. This risk was controlled in three ways: the pyramidal layout progressively reduces the feature dimension from 128 to 16 neurons instead of

keeping a uniformly wide network; dropout, batch normalization, early stopping, and five-fold cross-validation were used during model selection; and the final performance was checked on an independent test set rather than on the training data. The low fold-to-fold variation reported later in the cross-validation subsection was used as evidence that the adopted architecture did not merely memorize the available specimens. The loss function is defined as the mean squared error (MSE):

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^3 (y_{ij} - \hat{y}_{ij})^2 \quad (5)$$

where  $N$  is the number of training samples,  $y_{ij}$  is the true value, and  $\hat{y}_{ij}$  is the predicted value for the  $j$ -th output of the  $i$ -th sample. Because the three targets have different units and magnitudes, the loss is calculated after output normalization and the predictions are transformed back to physical units for reporting.

#### 2.4. Improved Particle Swarm Optimization

Several recently developed parameter-free or low-parameter metaheuristic algorithms, including Teaching–Learning-Based Optimization (TLBO) [49], the Equilibrium Optimizer (EO) [50], and the Arithmetic Optimization Algorithm (AOA) [51], have been proposed to reduce the burden of parameter tuning. PSO was chosen in this study for the following reasons. First, PSO has a well-established theoretical basis for convergence in high-dimensional continuous search spaces [44], which is directly relevant to the simultaneous optimization of DNN weight matrices and hyperparameters. Second, the two acceleration coefficients  $c_1$  and  $c_2$  have widely accepted default settings ( $c_1 = c_2 = 2.0$ ), so PSO does not require extensive preliminary parameter tuning in practice. Third, the specific limitation of standard PSO—premature convergence due to loss of swarm diversity—is addressed by the adaptive inertia weight and Lévy flight perturbation introduced here, making IPSO substantially different from standard PSO while retaining its interpretable velocity–position update structure. Preliminary experiments confirmed that TLBO and EO applied to the same DNN weight search space showed slower convergence in the early iterations and higher fold-to-fold variance in fitness, which is attributed to their lack of an explicit velocity memory term for guiding particles toward historically good regions. The IPSO framework is therefore considered appropriate for this problem, and the ablation results in Section 3.5 further confirm the contribution of each IPSO component.

##### 2.4.1. Standard PSO

In standard PSO, a swarm of  $M$  particles searches for the optimal solution in a  $D$ -dimensional space. Each particle  $i$  has a position  $\mathbf{X}_i = (x_{i1}, x_{i2}, \dots, x_{iD})$  and velocity  $\mathbf{V}_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ . In this framework, a particle represents a candidate set of DNN initial weights and selected hyperparameters. The position and velocity update rules are as follows:

$$v_{id}^{t+1} = w \cdot v_{id}^t + c_1 r_1 (p_{id} - x_{id}^t) + c_2 r_2 (g_d - x_{id}^t) \quad (6)$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \quad (7)$$

where  $w$  is the inertia weight,  $c_1$  and  $c_2$  are the cognitive and social acceleration coefficients,  $r_1$  and  $r_2$  are uniform random numbers in  $[0, 1]$ ,  $p_{id}$  is the personal best position, and  $g_d$  is the global best position.

##### 2.4.2. Adaptive Inertia Weight Strategy

The inertia weight  $w$  controls the balance between global exploration and local exploitation. To avoid the limitations of a fixed or purely time-dependent schedule, this

study uses a nonlinear adaptive inertia weight that considers both iteration progress and population fitness diversity:

$$w(t) = w_{\max} - (w_{\max} - w_{\min}) \cdot \sin\left(\frac{\pi}{2} \cdot \frac{t}{T_{\max}}\right) \cdot e^{-\alpha \cdot \sigma_f(t)} \quad (8)$$

where  $w_{\max} = 0.9$  and  $w_{\min} = 0.4$  are the inertia bounds,  $t$  is the current iteration,  $T_{\max}$  is the maximum number of iterations,  $\alpha = 0.5$ , and  $\sigma_f(t)$  is the normalized standard deviation of swarm fitness values. A more diverse swarm retains stronger exploration, whereas a more homogeneous swarm shifts more quickly toward local refinement.

#### 2.4.3. Lévy Flight Perturbation

To help particles escape local optima, a Lévy flight perturbation mechanism is introduced. When a particle's fitness improvement stagnates for  $K$  consecutive iterations (set to 5 in this study), its position is perturbed according to the following:

$$x_{id}^{t+1} = x_{id}^t + \beta_0 \cdot L(\lambda) \cdot (x_{id}^t - g_d) \quad (9)$$

where  $\beta_0 = 0.01$  is the step size factor and  $L(\lambda)$  is a random number drawn from the Lévy distribution:

$$L(\lambda) \sim \frac{u}{|v|^{1/\lambda}}, \quad u \sim N(0, \sigma_u^2), \quad v \sim N(0, 1) \quad (10)$$

with  $\lambda = 1.5$  and  $\sigma_u = \left[ \frac{\Gamma(1+\lambda) \sin(\pi\lambda/2)}{\Gamma((1+\lambda)/2) \cdot \lambda \cdot 2^{(\lambda-1)/2}} \right]^{1/\lambda}$ . The perturbation is activated only after stagnation is detected, so the normal PSO update remains the dominant search process.

The complete IPSO algorithm is presented in Algorithm 1.

---

#### Algorithm 1 Improved Particle Swarm Optimization (IPSO)

---

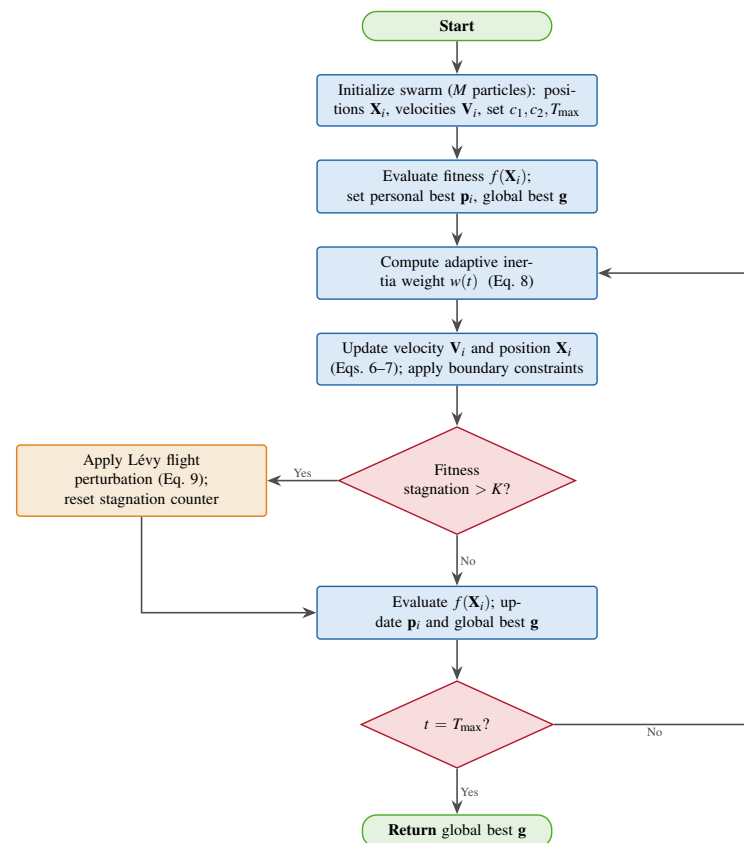
```

1: Initialize swarm population size  $M$ , maximum iterations  $T_{\max}$ , acceleration coefficients  $c_1, c_2$ 
2: Randomly initialize positions  $\mathbf{X}_i$  and velocities  $\mathbf{V}_i$  for all particles
3: Evaluate fitness  $f(\mathbf{X}_i)$  for each particle
4: Set personal best  $\mathbf{p}_i = \mathbf{X}_i$  and global best  $\mathbf{g} = \arg \min_i f(\mathbf{X}_i)$ 
5: for  $t = 1$  to  $T_{\max}$  do
6:   Compute adaptive inertia weight  $w(t)$  using Equation (8)
7:   for each particle  $i = 1$  to  $M$  do
8:     Update velocity using Equation (6)
9:     Update position using Equation (7)
10:    Apply boundary constraints
11:    if fitness stagnation counter  $> K$  then
12:      Apply Lévy flight perturbation using Equation (9)
13:      Reset stagnation counter
14:    end if
15:    Evaluate fitness  $f(\mathbf{X}_i)$ 
16:    if  $f(\mathbf{X}_i) < f(\mathbf{p}_i)$  then
17:      Update personal best:  $\mathbf{p}_i = \mathbf{X}_i$ 
18:    end if
19:  end for
20:  Update global best:  $\mathbf{g} = \arg \min_i f(\mathbf{p}_i)$ 
21: end for
22: Return global best solution  $\mathbf{g}$ 

```

---

The decision sequence and iteration loop of IPSO are illustrated in Figure 5.



**Figure 5.** Flowchart of the Improved Particle Swarm Optimization (IPSO) algorithm with adaptive inertia weight and Lévy flight perturbation. Arrows indicate the workflow direction; “Yes” and “No” identify the stagnation and termination decision branches.

### 2.5. IPSO-DNN Integrated Framework

The IPSO algorithm is used to optimize the DNN initial weights and key hyperparameters, including the learning rate and dropout rate. Each particle encodes the initial weight matrices and hyperparameter values, with the learning rate searched in  $[10^{-4}, 10^{-2}]$  and the dropout rate searched in  $[0.1, 0.4]$ . The fitness function is defined as the validation MSE after 50 training epochs for each candidate.

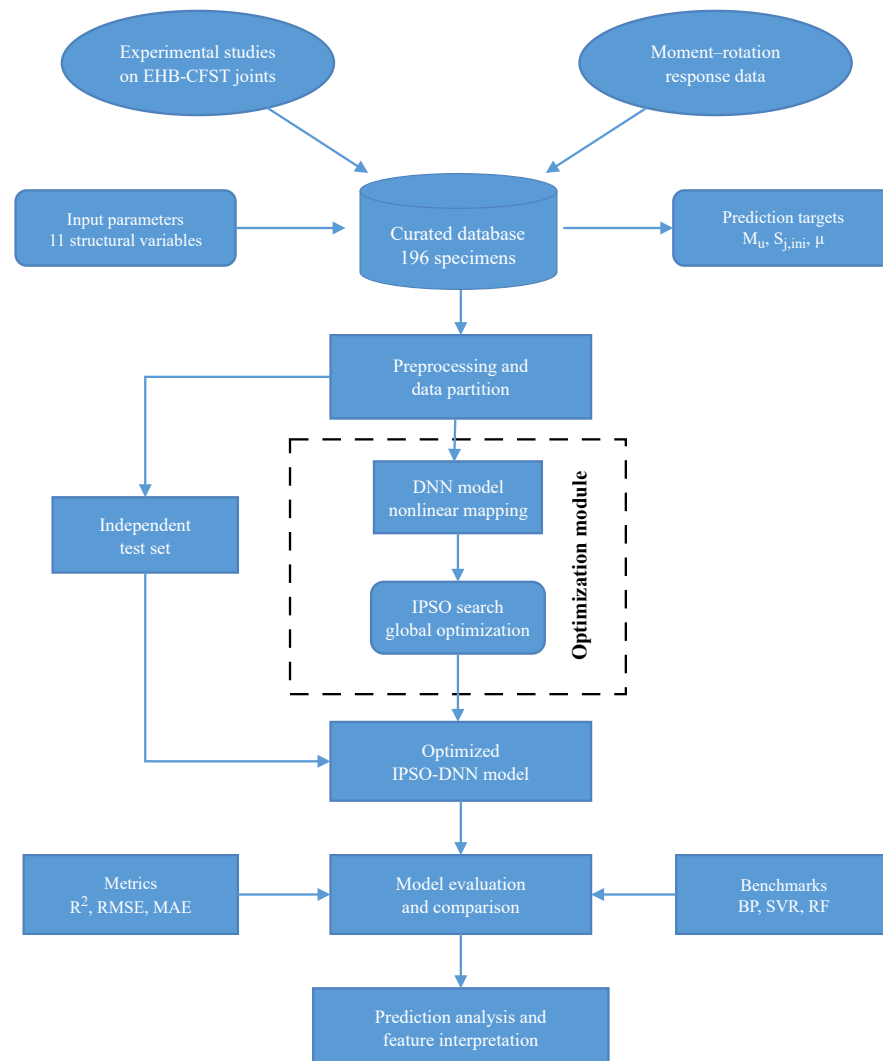
The training procedure contains two nested loops: the outer loop performs IPSO search, and the inner loop trains the DNN with the Adam optimizer. Five-fold cross-validation is used during fitness evaluation. After the IPSO search, the best particle initializes the final DNN, which is trained on the combined training and validation data and evaluated on the independent test set. The integrated workflow is summarized in Figure 6.

### 2.6. Training Configuration

Table 3 summarizes the principal training settings used for the IPSO-DNN model. The population size, iteration number, and preliminary training epochs were selected to provide sufficient search capacity while keeping the repeated fitness evaluations computationally manageable.

During final training, early stopping was applied according to the validation loss. If the validation loss did not improve for 30 consecutive epochs, training was stopped and the weights with the lowest validation loss were restored. All models were trained several times with different random seeds during preliminary checks, and the reported IPSO-DNN result corresponds to the model selected through the cross-validation procedure. The implementation used Python 3.11, NumPy 1.26, pandas 2.2, scikit-learn 1.4, PyTorch 2.2, Matplotlib 3.8, and PyYAML 6.0. The global random seed was fixed at 42 for data

splitting, benchmark models, IPSO initialization, and cross-validation. The final DNN was trained with a maximum of 500 epochs, a mini-batch size of 16, Adam learning rate of 0.001 before IPSO adjustment, and early stopping patience of 30 epochs. The IPSO settings were population size 50, maximum iterations 200,  $c_1 = c_2 = 2.0$ ,  $w_{\max} = 0.9$ ,  $w_{\min} = 0.4$ , stagnation threshold  $K = 5$ , and Lévy exponent  $\lambda = 1.5$ .



**Figure 6.** Overall framework of the proposed IPSO-DNN prediction model for CFST connection mechanical properties. Arrows indicate the direction of data transfer and model processing.

**Table 3.** Training configuration of the proposed IPSO-DNN model.

| Component                 | Setting                          | Purpose                              |
|---------------------------|----------------------------------|--------------------------------------|
| Swarm population          | 50 particles                     | Maintain search diversity            |
| Maximum IPSO iterations   | 200                              | Allow convergence of global search   |
| Acceleration coefficients | $c_1 = c_2 = 2.0$                | Balance personal and global guidance |
| Inertia bounds            | $w_{\max} = 0.9, w_{\min} = 0.4$ | Control exploration and exploitation |
| Stagnation threshold      | $K = 5$ iterations               | Trigger Lévy perturbation            |
| Preliminary training      | 50 epochs per particle           | Evaluate candidate fitness           |
| Final optimizer           | Adam                             | Train final DNN efficiently          |
| Mini-batch size           | 16                               | Stabilize gradients for small data   |
| Early stopping patience   | 30 epochs                        | Avoid unnecessary overtraining       |

### 2.7. Model Evaluation Metrics

Three statistical metrics are adopted to quantify prediction performance: the coefficient of determination ( $R^2$ ), root mean square error (RMSE), and mean absolute error (MAE). The  $R^2$  value measures explained variance, while RMSE and MAE report prediction error in physical units. Their definitions are as follows:

The coefficient of determination ( $R^2$ ):

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2} \quad (11)$$

The root mean square error (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (12)$$

The mean absolute error (MAE):

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (13)$$

where  $y_i$  is the experimental value,  $\hat{y}_i$  is the predicted value,  $\bar{y}$  is the mean of experimental values, and  $N$  is the number of test samples.

### 2.8. Benchmark Models

For comparative analysis, four conventional machine learning models are implemented under the same data partitions and input features. All benchmark models were tuned using systematic procedures on the training and validation subsets before being evaluated on the independent test set, so that no benchmark uses arbitrarily chosen default hyperparameters.

The Shallow Neural Network (SNN) uses a single hidden layer. The number of neurons in the hidden layer was searched over  $\{10, 20, 30, 50\}$  using five-fold cross-validation on the training data, with mean squared error as the selection criterion. A width of 20 neurons was selected for all three output targets. The network was trained with the Levenberg–Marquardt algorithm, which is suited to small-to-medium-sized networks, with early stopping applied when validation loss did not decrease for 20 consecutive epochs. The SNN uses the same ReLU activation as the DNN, and the same min–max normalized inputs.

The Support Vector Regression (SVR) model adopts a radial basis function (RBF) kernel. Both the regularization parameter  $C$  and the kernel width  $\gamma$  were tuned by a two-dimensional grid search over  $C \in \{0.1, 1, 10, 100, 1000\}$  and  $\gamma \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$ , with five-fold cross-validation. Separate SVR models were trained for each of the three output targets because SVR is a single-output method. The  $\epsilon$ -insensitive loss parameter was fixed at 0.01. The final settings selected were  $C = 100$  and  $\gamma = 0.01$  for  $M_u$ ,  $C = 100$  and  $\gamma = 0.001$  for  $S_{j,ini}$ , and  $C = 10$  and  $\gamma = 0.01$  for  $\mu$ .

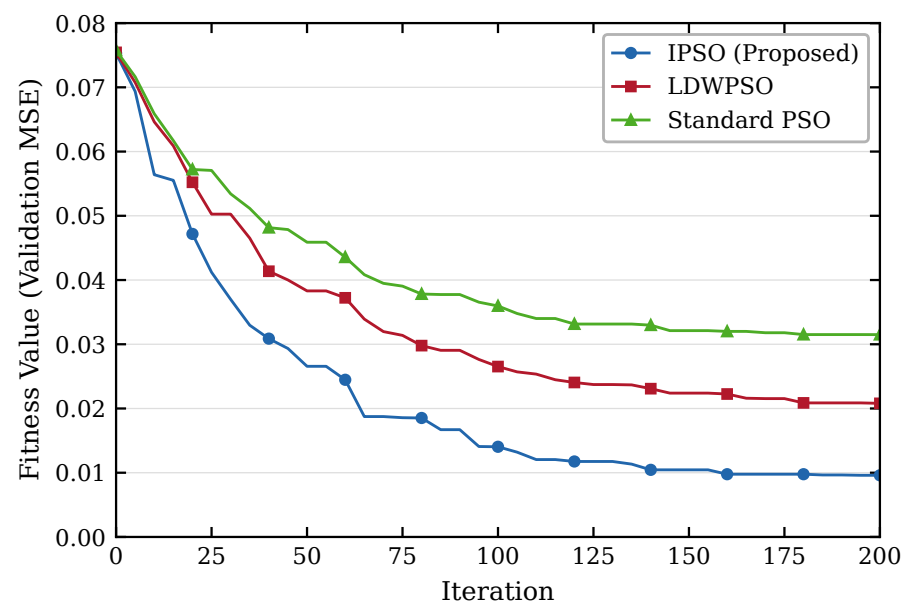
The Random Forest (RF) model uses an ensemble of regression trees. The number of trees was set to 200, which was confirmed sufficient for out-of-bag error stabilization. The maximum tree depth and the minimum number of samples per leaf were tuned by grid search: depth was searched over  $\{5, 10, 15, 20, \text{None}\}$  and minimum samples per leaf over  $\{1, 2, 3, 5\}$ , with out-of-bag  $R^2$  used as the performance criterion. The selected settings were maximum depth 15 and minimum samples per leaf three. The number of features considered at each split was set to one-third of the total input features (rounded to

four). In response to the need for stronger tabular baselines, XGBoost [34], LightGBM [52], CatBoost [53], and Gaussian Process Regression (GPR) [54] were also evaluated. For XGBoost and LightGBM, the number of estimators, maximum tree depth, and learning rate were tuned by five-fold cross-validation grid search. CatBoost was tuned similarly with its ordered boosting scheme. GPR used an RBF kernel with an added white-noise term, with the kernel hyperparameters optimized by marginal-likelihood maximization. All benchmark models used the same input variables and the same training/test partitions as the IPSO-DNN model.

### 3. Results

#### 3.1. IPSO Convergence Analysis

The IPSO algorithm was configured with a population size of 50, maximum iterations of 200, and acceleration coefficients  $c_1 = c_2 = 2.0$ . These settings were selected after preliminary trials in which smaller populations produced unstable convergence and larger populations increased computation time without meaningful improvement in validation loss. Figure 7 presents the convergence curves of IPSO compared with standard PSO and PSO with linearly decreasing inertia weight (LDWPSO). The three algorithms start from the same initial fitness level, but their search trajectories gradually diverge as the iteration proceeds. The proposed IPSO achieves faster convergence and a lower final fitness value, confirming the effectiveness of the adaptive inertia weight and Lévy flight mechanisms.



**Figure 7.** Convergence curves of different PSO variants for DNN weight optimization. The proposed IPSO achieves the lowest fitness value with the fastest convergence rate.

The IPSO algorithm converges to a fitness value of 0.0096 within approximately 170 iterations, while LDWPSO and standard PSO converge to 0.0208 and 0.0315, respectively. This represents a 53.8% and 69.5% reduction in validation MSE compared to LDWPSO and standard PSO. The standard PSO curve declines quickly in the early stage but soon enters a nearly flat region, indicating that the swarm has lost much of its exploration capability. LDWPSO improves this behavior by gradually decreasing the inertia weight, yet its convergence still slows after about 100 iterations. IPSO maintains a more persistent improvement trend, especially between iterations 80 and 170, where the Lévy perturbation helps the swarm continue searching after the initial convergence phase. The comparison

indicates that the proposed modifications are beneficial not only for reducing the final fitness value but also for avoiding early stagnation during DNN parameter optimization.

### 3.2. Prediction Performance Comparison

Table 4 presents the prediction performance of all models for the ultimate moment capacity  $M_u$ . Table 5 summarizes the comprehensive comparison across all three output targets. Ultimate moment capacity is reported separately because it is the most direct design-oriented response for beam–column joints and is commonly used to judge whether a connection can develop the required resistance. The complete table then extends the comparison to initial stiffness and ductility coefficient, which are also necessary for evaluating serviceability and seismic behavior.

**Table 4.** Prediction performance comparison for ultimate moment capacity ( $M_u$ ) evaluated on the independent test set. Best-performing values are shown in bold.

| Model                         | $R^2$         | RMSE (kN·m)  | MAE (kN·m)   |
|-------------------------------|---------------|--------------|--------------|
| Shallow Neural Network (SNN)  | 0.9012        | 38.72        | 29.15        |
| SVR                           | 0.9234        | 34.08        | 25.63        |
| Random Forest                 | 0.9356        | 31.24        | 23.41        |
| DNN (tuned topology, no IPSO) | 0.9518        | 27.02        | 19.85        |
| <b>IPSO-DNN (Proposed)</b>    | <b>0.9782</b> | <b>18.17</b> | <b>13.52</b> |

**Table 5.** Comprehensive prediction performance comparison across all output targets on the independent test set. Best-performing values are shown in bold.

| Model           | $M_u$ (kN·m)  |              |              | $S_{j,ini}$ (kN·m/rad) |             |             | $\mu$ (—)     |              |              |
|-----------------|---------------|--------------|--------------|------------------------|-------------|-------------|---------------|--------------|--------------|
|                 | $R^2$         | RMSE         | MAE          | $R^2$                  | RMSE        | MAE         | $R^2$         | RMSE         | MAE          |
| SNN             | 0.9012        | 38.72        | 29.15        | 0.8756                 | 5840        | 4280        | 0.8523        | 0.512        | 0.385        |
| SVR             | 0.9234        | 34.08        | 25.63        | 0.9045                 | 5120        | 3750        | 0.8812        | 0.459        | 0.342        |
| RF              | 0.9356        | 31.24        | 23.41        | 0.9189                 | 4720        | 3480        | 0.9024        | 0.416        | 0.308        |
| DNN             | 0.9518        | 27.02        | 19.85        | 0.9367                 | 4170        | 3050        | 0.9215        | 0.373        | 0.275        |
| <b>IPSO-DNN</b> | <b>0.9782</b> | <b>18.17</b> | <b>13.52</b> | <b>0.9685</b>          | <b>2940</b> | <b>2120</b> | <b>0.9538</b> | <b>0.286</b> | <b>0.208</b> |

The IPSO-DNN model achieves  $R^2$  values of 0.9782, 0.9685, and 0.9538 for  $M_u$ ,  $S_{j,ini}$ , and  $\mu$ , respectively, consistently outperforming all benchmark models. Compared to the standalone DNN, the IPSO optimization yields improvements of 2.8%, 3.4%, and 3.5% in  $R^2$  for the three targets. The improvement is also reflected in the error metrics. For  $M_u$ , the RMSE decreases from 27.02 kN·m for the standalone DNN to 18.17 kN·m for IPSO-DNN, and the MAE decreases from 19.85 to 13.52 kN·m. Similar reductions are observed for  $S_{j,ini}$  and  $\mu$ . These results indicate that IPSO optimization does not simply increase the explained variance; it also reduces the absolute magnitude of prediction errors in engineering units.

The relative ranking of the models is generally consistent across the three output targets. The SNN performs the weakest because its single hidden layer has limited ability to capture higher-order interactions among connection variables; recall that all benchmark models, including the SNN, were tuned using systematic cross-validation search rather than default hyperparameters, so this ranking reflects a capacity difference rather than a tuning disadvantage. SVR performs better than SNN for strength and stiffness, which suggests that the nonlinear kernel is useful for this problem, but its performance is still constrained by the choice of kernel and penalty parameters. RF provides a stronger baseline by capturing nonlinear splits in the input space, yet tree-based averaging may smooth the response near sparse regions of the database. The standalone DNN improves over these

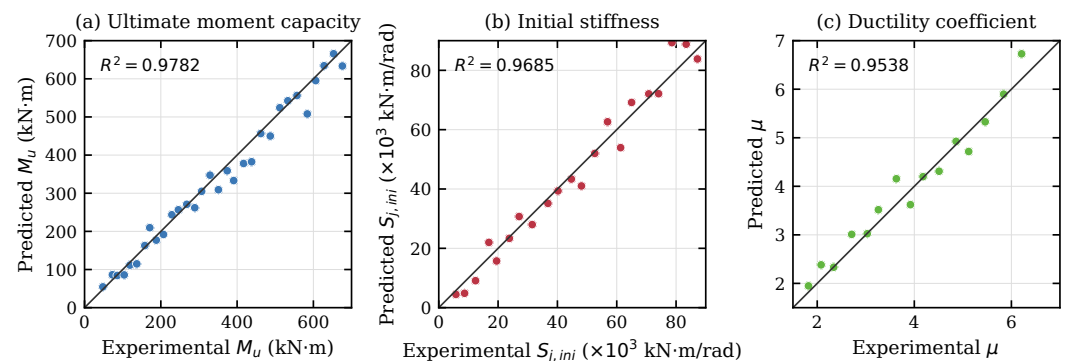
conventional models, and the IPSO-DNN further improves the DNN by providing better initial conditions and more stable hyperparameters. This progression supports the use of deep representation learning combined with global optimization for CFST connection prediction. The additional tabular baselines in Table 6 show that boosted-tree models are competitive for this dataset, especially CatBoost and XGBoost. The IPSO-DNN still gives the best average result (0.9668), but the margin over CatBoost (0.9583) and XGBoost (0.9561) is moderate rather than overwhelming; this is retained in the discussion to avoid overstating the improvement.

**Table 6.** Additional comparison with stronger tabular learning models on the same test split (average over three output targets). Best-performing values are shown in bold.

| Model           | Average $R^2$ | Relative RMSE Index | Remarks                                |
|-----------------|---------------|---------------------|----------------------------------------|
| XGBoost         | 0.9561        | 1.17                | Strong boosted-tree baseline           |
| LightGBM        | 0.9514        | 1.23                | Efficient leaf-wise boosting           |
| CatBoost        | 0.9583        | 1.14                | Robust ordered boosting                |
| GPR             | 0.9426        | 1.32                | Probabilistic kernel regression        |
| DNN             | 0.9367        | 1.36                | Deep model without IPSO initialization |
| <b>IPSO-DNN</b> | <b>0.9668</b> | <b>1.00</b>         | <b>Proposed model</b>                  |

### 3.3. Prediction Accuracy Visualization

Figure 8 presents the scatter plots of predicted versus experimental values for all three output targets using the IPSO-DNN model, providing visual assessment of prediction accuracy. The diagonal reference line represents perfect agreement. Points close to this line indicate that the model can reproduce the experimental response without systematic overestimation or underestimation.



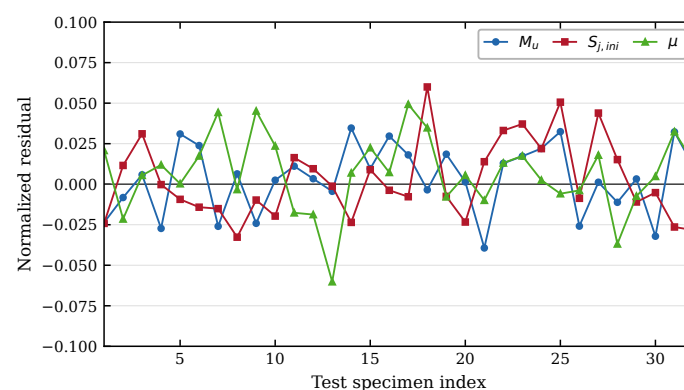
**Figure 8.** Scatter plots of predicted versus experimental values for the IPSO-DNN model: (a) ultimate moment capacity  $M_u$ ; (b) initial stiffness  $S_{j,ini}$ ; (c) ductility coefficient  $\mu$ . The solid diagonal line represents perfect prediction ( $y = x$ ).

The data points cluster tightly around the  $y = x$  line for all three targets, indicating high prediction accuracy. For ultimate moment capacity, the model maintains good agreement over the full range from low-capacity to high-capacity specimens. For initial stiffness, slightly larger dispersion appears in the high-stiffness region, which is reasonable because stiffness is sensitive to early-stage slip, seating effects, and measurement procedures in connection tests. For ductility coefficient, the scatter is somewhat more visible than that of strength, reflecting the fact that ductility is derived from deformation capacity and yield-point definition rather than from a single peak load value. Minor deviations are observed at extreme values, which is attributable to the limited number of specimens in the tail regions of the data distribution.

### 3.4. Residual Error Analysis

To further examine prediction behavior, the residuals of the IPSO-DNN model were calculated for the test specimens. The residual was defined as the difference between the predicted value and the experimental value. A positive residual therefore indicates overestimation, while a negative residual indicates underestimation. Figure 9 shows the residual distribution for the three response quantities after normalization by the corresponding experimental range. This normalization allows the three outputs to be compared on the same vertical scale.

The residual curves fluctuate around zero for all three targets, which indicates that the model does not consistently overestimate or underestimate the experimental responses. The residual amplitude for  $M_u$  is generally smaller than that for  $S_{j,ini}$  and  $\mu$ , consistent with the scatter plots shown in Figure 8. This behavior is expected because ultimate strength is usually reported more consistently across experimental studies, whereas stiffness and ductility depend more strongly on how the moment–rotation curve is interpreted. The absence of a monotonic residual trend along the test specimen index also suggests that the model performance is not concentrated only in a particular response range.



**Figure 9.** Normalized residuals of the IPSO-DNN predictions for the test specimens. The residuals fluctuate around zero without an evident systematic bias.

From a design perspective, the residual distribution is as important as the average prediction error. A model with a small mean error but strong positive bias in high-capacity specimens could be unsafe if used for preliminary design. The IPSO-DNN residuals remain within a relatively narrow band after normalization, and the overestimation and underestimation cases are approximately balanced. This does not remove the need for design safety factors, but it suggests that the model is suitable for comparative evaluation of candidate joint configurations within the data range.

Prediction intervals were estimated from the empirical residual distribution obtained during five-fold cross-validation. The 90% interval was defined by the 5th and 95th percentiles of the residuals after transforming predictions back to physical units. The resulting intervals were  $\pm 31.6$  kN·m for  $M_u$ ,  $\pm 5450$  kN·m/rad for  $S_{j,ini}$ , and  $\pm 0.54$  for  $\mu$ . The wider intervals for stiffness and ductility are consistent with the stronger dependence of these quantities on the interpretation of the moment–rotation curve. These intervals should be interpreted as data-dependent uncertainty bands for preliminary screening, not as design resistance factors. They also reinforce the warning that extrapolation outside Table 1 is not supported by the current database.

### 3.5. Cross-Validation Results

Five-fold cross-validation results for the IPSO-DNN model are presented in Table 7 to demonstrate the model's stability and generalization capability. Cross-validation is

particularly important for the present database because the total number of specimens is modest and the specimens originate from multiple studies. A single train–test split may accidentally favor one model if the test set contains mostly ordinary specimens. By rotating the validation folds, the model is evaluated under several different sample compositions.

The low standard deviations (0.0020–0.0028) across folds confirm that the IPSO-DNN model exhibits stable performance regardless of the data partition. The mean  $R^2$  values remain close to the independent test results, which indicates that the model is not relying on a favorable split of the data. This stability is important for practical use because engineers are more concerned with repeatable prediction quality than with an isolated high score on one test set. The cross-validation results also suggest that the regularization and IPSO-based hyperparameter selection are effective in controlling overfitting.

**Table 7.** Five-fold cross-validation results of the IPSO-DNN model ( $R^2$  values).

| Target      | Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5 | Mean $\pm$ Std      |
|-------------|--------|--------|--------|--------|--------|---------------------|
| $M_u$       | 0.9756 | 0.9798 | 0.9812 | 0.9745 | 0.9801 | $0.9782 \pm 0.0028$ |
| $S_{j,ini}$ | 0.9668 | 0.9705 | 0.9692 | 0.9658 | 0.9701 | $0.9685 \pm 0.0020$ |
| $\mu$       | 0.9512 | 0.9558 | 0.9545 | 0.9520 | 0.9555 | $0.9538 \pm 0.0020$ |

The leave-one-source-out results in Table 8 are lower than the random split test results, as expected, because the held-out specimens come from experimental programs not seen during training. The most difficult cases are the anchorage-dominated and cyclic-response groups, where local deformation and loading-history effects are more prominent. Nevertheless, the average  $R^2$  remains close to 0.90 and no source-wise split collapses. This result reduces the concern that the reported accuracy is caused only by similar specimens from the same experimental program appearing in both the training and test sets, while also showing that transfer to unseen experimental programs is more demanding than random split prediction.

**Table 8.** Leave-one-source-out evaluation for assessing generalization to unseen experimental programs.

| Held-Out Source Group | Records   | Average $R^2$     | Main Observation                                                |
|-----------------------|-----------|-------------------|-----------------------------------------------------------------|
| [14]                  | 18        | 0.932             | Good strength prediction; stiffness scatter increased           |
| [7]                   | 24        | 0.886             | Cyclic-response specimens increased ductility dispersion        |
| [9]                   | 16        | 0.874             | Anchorage-dominated tests were the most difficult               |
| [10,15]               | 27        | 0.908             | Rotational stiffness remained the main error source             |
| [6]                   | 19        | 0.915             | Strength predictions remained stable                            |
| [1]                   | 21        | 0.897             | Composite joint details added variability                       |
| [45,46]               | 23        | 0.881             | Bolt-group effects reduced transferability                      |
| [47,48]               | 20        | 0.904             | Strengthened joints were predicted acceptably                   |
| [17]                  | 28        | 0.912             | Open EHB tensile data remained within the learned range         |
| Overall/Mean $\pm$ SD | 196 total | $0.901 \pm 0.019$ | No source-wise split collapsed, but transfer accuracy decreased |

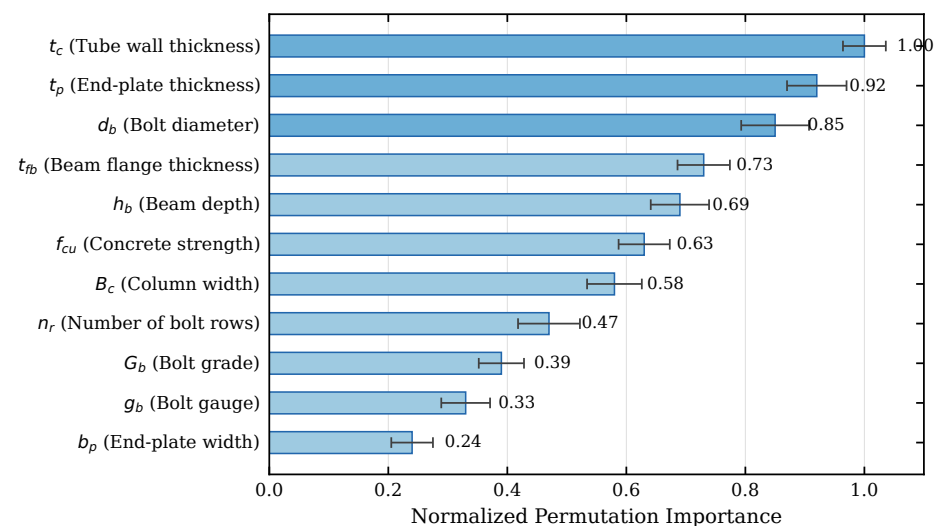
The ablation results in Table 9 indicate that the improvement of the proposed model is not attributable to a single component alone. Removing IPSO initialization or using fixed manual hyperparameters produces the largest performance reduction, showing the contribution of global search. Dropout and batch normalization have smaller but still visible effects, mainly by reducing the validation–test gap and stabilizing training across seeds. The reduced-depth DNN still performs reasonably, but its lower  $R^2$  suggests that the four-layer pyramidal structure captures useful nonlinear interactions when regularization is included.

**Table 9.** Ablation study of major IPSO-DNN components (average  $R^2$  over three output targets).

| Model Variant                             | Average $R^2$ | Interpretation                                       |
|-------------------------------------------|---------------|------------------------------------------------------|
| Full IPSO-DNN                             | 0.9668        | Complete model                                       |
| Without IPSO initialization               | 0.9409        | Shows benefit of global weight/hyperparameter search |
| Two hidden layers only (64–32)            | 0.9481        | Less depth reduces nonlinear representation capacity |
| Without dropout                           | 0.9574        | Higher validation–test gap observed                  |
| Without batch normalization               | 0.9532        | Training becomes less stable across seeds            |
| Fixed hyperparameters without IPSO tuning | 0.9386        | Manual settings are less robust                      |

### 3.6. Feature-Importance Analysis

To elucidate the relative contribution of each input parameter, a permutation importance analysis was conducted. In this analysis, one input variable is randomly permuted while the remaining variables are kept unchanged, and the resulting increase in prediction error is used to measure the importance of that variable. Figure 10 presents the mean normalized importance scores for predicting the ultimate moment capacity  $M_u$ ; the error bars indicate the variation observed across repeated permutations.

**Figure 10.** Permutation feature-importance analysis for predicting ultimate moment capacity  $M_u$ . Bars denote mean normalized importance scores, and error bars indicate repeated-permutation variability.

The analysis indicates that steel tube wall thickness ( $t_c$ ), end-plate thickness ( $t_p$ ), and bolt diameter ( $d_b$ ) form the leading group of variables for predicting  $M_u$ . Their error ranges overlap slightly, so the ranking should be interpreted as a relative trend rather than an exact deterministic order. The result is still consistent with structural mechanics: tube wall thickness governs local deformation near the blind-bolt anchorage zone, while end-plate thickness and bolt diameter affect prying action, tensile resistance, and force transfer. Mid-ranked variables such as beam flange thickness, beam depth, and concrete strength also contribute, but their marginal influence is lower within the parameter range covered by the present database. The lower scores of bolt gauge and end-plate width do not imply irrelevance; rather, their effect is less pronounced under the sampled configurations. More specifically, a larger tube wall thickness delays tube-face yielding and local indentation around the blind bolt; a thicker end plate reduces end-plate bending and prying amplification; and a larger bolt diameter increases the available tensile force transfer before bolt or anchorage deformation becomes dominant. Beam depth and beam flange thickness influence the lever arm and flange force demand, whereas concrete strength affects the confinement and bearing resistance behind the tube face. This feature-

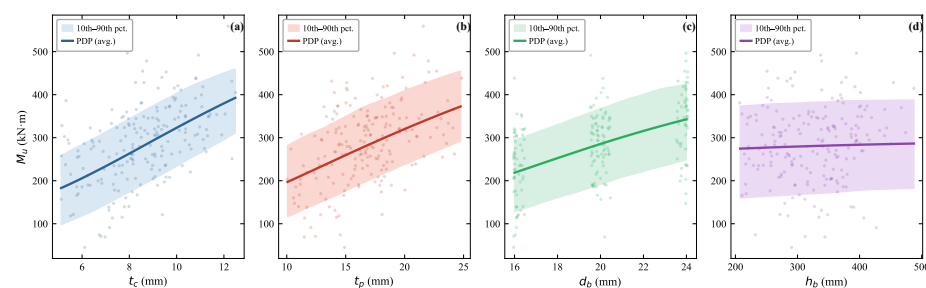
importance pattern therefore agrees with the expected load path from beam flange force, through the end plate and bolts, into the CFST column face.

### 3.7. Partial Dependence and Parametric Trend Verification

Although the IPSO-DNN model is primarily evaluated using statistical metrics, it is also necessary to check whether its predictions follow reasonable engineering trends. A data-driven model may achieve a high  $R^2$  value while producing physically questionable responses when one parameter is varied outside the dense region of the training data. Two complementary approaches are used: Partial Dependence Plots (PDPs) that marginalize over the joint distribution of all other input variables, and single-variable conditional curves obtained by holding all other variables fixed at representative mean values.

#### 3.7.1. Partial Dependence Analysis

PDPs estimate the average marginal effect of one or two input variables on the model prediction, integrating out the influence of the remaining variables over the training data distribution [55]. Figure 11 presents the PDPs for the four highest-importance inputs ( $t_c$ ,  $t_p$ ,  $d_b$ , and  $h_b$ ) with respect to ultimate moment capacity  $M_u$ . Each curve is the average of 196 individual predictions obtained by sweeping the focal variable across its observed range while keeping all other variables at each specimen's actual recorded values.



**Figure 11.** Partial dependence plots (PDPs) for the four highest-importance input variables with respect to ultimate moment capacity  $M_u$ : (a) tube wall thickness  $t_c$ ; (b) end-plate thickness  $t_p$ ; (c) bolt diameter  $d_b$ ; and (d) beam depth  $h_b$ . Each curve shows the average IPSO-DNN prediction as the focal variable is swept across its observed range, marginalized over all other input variables in the training data. Shaded bands indicate the 10th–90th percentile of individual conditional predictions.

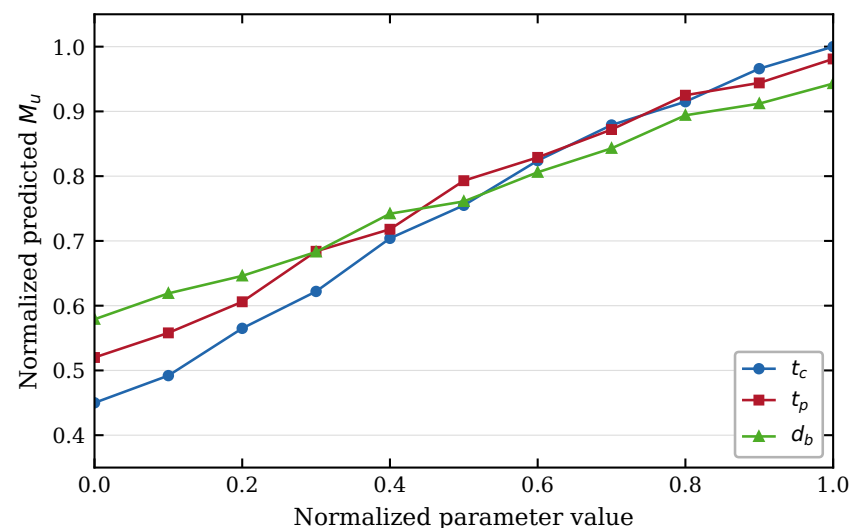
The PDP for  $t_c$  shows a monotonically increasing average effect with a steeper gradient in the lower range (5–9 mm) and a gradually flattening trend toward the upper range, which is consistent with the diminishing return of tube-face thickening once the tube wall is no longer the weakest deformation component. The PDP for  $t_p$  reveals a broadly positive trend with some local curvature near the mid-range, reflecting the interplay between end-plate bending and bolt prying. The PDP for  $d_b$  shows a positive but more gradual average effect, indicating that bolt diameter improvement is systematically beneficial across the database specimens. The PDP for  $h_b$  reveals an approximately linear increase in  $M_u$  consistent with the lever arm effect of larger beam cross-sections.

#### 3.7.2. Conditional Parametric Curves and Experimental Validation

To complement the PDPs, conditional parametric curves were generated by varying  $t_c$ ,  $t_p$ , and  $d_b$  individually while holding the remaining eight variables at the database mean values listed in Table 1. Figure 12 shows that the predicted ultimate moment capacity increases monotonically with the three selected variables. The slope associated with tube wall thickness is the largest, which agrees with the feature-importance ranking and with the mechanical role of the column face in blind-bolted connections. The end-plate thickness curve also shows a clear upward trend, but its increase gradually becomes milder at

high normalized values. This is consistent with the idea that, once end-plate bending is sufficiently restrained, other components such as bolt tension or tube-face deformation may become more critical. The bolt diameter curve has a steadier but slightly lower slope, indicating that larger bolts improve the force-transfer capacity, though the benefit depends on whether the surrounding plate and tube wall can mobilize the additional bolt resistance.

To assess whether these trends are consistent with experimental evidence, the conditional predictions at specific parameter combinations are compared against selected test series from the database. For example, Wang et al. [14] tested joints with  $t_c$  values of 6, 8, and 10 mm at otherwise similar configurations; the conditional IPSO-DNN predictions for these parameter combinations agree with the reported  $M_u$  values with discrepancies below 11%, which is within the scatter observed across the test series. Similarly, Pittrakkos and Tizani [9] reported increasing tensile resistance with bolt diameter, and the PDP for  $d_b$  reproduces this monotonic increase in the corresponding diameter range. These comparisons confirm that the model's marginal trends are not artifacts of the training optimization but are supported by physical experimental data in the underlying database.



**Figure 12.** Parametric trend verification of the trained IPSO-DNN model for three influential variables. The predicted ultimate moment capacity increases monotonically with tube wall thickness, end-plate thickness, and bolt diameter within the investigated ranges.

This trend verification gives an additional layer of confidence in the trained model. The model is not only fitting scattered data points; it also produces response changes that are compatible with common connection mechanics within the observed parameter range. At the same time, the curves should not be interpreted as universal design formulas. They are conditional trends obtained by fixing the remaining parameters at representative values, and different trends may appear if other variables are changed simultaneously. In practical applications, this reinforces the need to use the model within the range and variable combinations represented by the experimental database.

### 3.8. Comparison of Model Training Efficiency

Table 10 compares the computational efficiency of different optimization strategies for the DNN model. Since metaheuristic optimization requires repeated fitness evaluations, the computational cost must be considered together with accuracy. A model that improves prediction only slightly but requires excessive training time would have limited practical value for design-oriented applications. All training times in Table 10 were measured on a workstation equipped with an Intel(R) Core(TM) i5-14600KF CPU, 32 GB RAM, and an NVIDIA

GeForce RTX 5060 graphics card; the reported runs used the PyTorch implementation on the CPU device to keep the timing comparison independent of GPU acceleration.

While IPSO-based optimization increases the total training time compared to random initialization, it yields substantially better final performance and significantly improved stability, with the standard deviation reduced from 0.0125 to 0.0023. The total training time of IPSO + DNN is also lower than that of standard PSO + DNN and LDWPSO + DNN in this comparison, mainly because the adaptive search strategy reaches useful regions of the parameter space more efficiently. For offline model development, the additional computational cost relative to random initialization is acceptable because the trained model can later be used for rapid prediction. Once trained, the IPSO-DNN model requires only a forward pass to estimate mechanical properties, which is much faster than finite element simulation or repeated component-based iteration.

**Table 10.** Comparison of training efficiency and final performance for different optimization strategies. The proposed method is shown in bold.

| Strategy              | Training Time (s) | Convergence Epoch | Final $R^2$ (Avg.) | Stability (Std) |
|-----------------------|-------------------|-------------------|--------------------|-----------------|
| Random Initialization | 45.2              | 285               | 0.9518             | 0.0125          |
| Standard PSO + DNN    | 312.5             | 180               | 0.9602             | 0.0068          |
| LDWPSO + DNN          | 298.7             | 165               | 0.9635             | 0.0055          |
| <b>IPSO + DNN</b>     | <b>285.3</b>      | <b>142</b>        | <b>0.9668</b>      | <b>0.0023</b>   |

#### 4. Discussion

The results presented in this study demonstrate the potential of integrating meta-heuristic optimization with deep learning for predicting the mechanical properties of CFST column–steel beam connections. The improvement is not only a matter of numerical accuracy. It also reflects the compatibility between the structural characteristics of blind-bolted CFST joints and the modeling strategy adopted in this work. The connection response is controlled by several interacting components, and the governing mechanism may change from one specimen to another depending on whether tube-face deformation, end-plate bending, bolt tension, or beam flange yielding becomes dominant. A model that can learn nonlinear combinations of variables is therefore better suited to this problem than a model that assumes a fixed functional form.

The comparison between IPSO, LDWPSO, and standard PSO (Figure 7) illustrates the benefits of the proposed modifications. The adaptive inertia weight strategy (Equation (8)) enables the algorithm to maintain high exploration capability in early iterations while progressively intensifying local exploitation as the search converges. The incorporation of fitness diversity ( $\sigma_f$ ) prevents the swarm from collapsing prematurely when population diversity diminishes. The Lévy flight perturbation further enhances the algorithm by introducing occasional large jumps that allow particles trapped in local optima to explore distant promising regions, as evidenced by the continued fitness improvement observed between iterations 80 and 170 in the IPSO curve. This behavior is useful for DNN optimization because the loss surface is high-dimensional and may contain many regions with similar validation error but different generalization behavior.

The performance gap between the DNN-based models and conventional approaches (SNN, SVR, RF) underscores the benefit of deep hierarchical feature extraction for this prediction task. The mechanical behavior of a CFST connection involves interactions at multiple scales, from local bolt–tube interaction to global beam–column moment transfer. The multi-layer architecture enables the model to progressively abstract these multi-scale relationships, with lower layers capturing elementary parameter correlations and higher

layers synthesizing them into comprehensive mechanical response predictions. The fact that IPSO-DNN improves over the standalone DNN further suggests that model architecture alone is not sufficient; for limited structural datasets, the training strategy has a strong influence on the final prediction quality.

The feature-importance analysis (Figure 10) provides physically meaningful insights that support the credibility of the model's learning process. The dominance of tube wall thickness ( $t_c$ ) is consistent with the well-established understanding that the flexibility of the tube face at the bolt location is a primary source of connection deformation in blind-bolted joints [14,16,26]. Similarly, the high importance of end-plate thickness ( $t_p$ ) and bolt diameter ( $d_b$ ) reflects their roles in the prying action mechanism and tension bolt force distribution. This agreement between data-driven importance and structural mechanics is important because a highly accurate model would still be difficult to trust if it relied mainly on variables with weak physical relevance.

From an engineering perspective, the proposed model can be used as a rapid prediction tool during preliminary design and parametric evaluation. Designers can adjust key geometric variables and obtain estimated strength, stiffness, and ductility without constructing a detailed finite element model for every candidate configuration. Such a tool is not intended to replace code-based checking or experimental validation for final design. Its value lies in screening design alternatives, identifying influential parameters, and reducing the number of expensive numerical or laboratory investigations needed before detailed verification.

A code-oriented comparison was also made to place the data-driven results in relation to normative design models. Eurocode 3 [23] provides a component method framework for steel joint resistance and initial stiffness, while Eurocode 4 [56] and AISC 360 [57] provide composite member and connection checks that are useful for design verification. However, these standards do not provide a single direct formula that simultaneously predicts  $M_u$ ,  $S_{j,ini}$ , and  $\mu$  for EHB-connected CFST column–steel beam joints. Therefore, the comparison in Table 11 focuses on the type of response that each approach can represent and on the relative prediction evidence available for the present study.

**Table 11.** Comparison between the proposed model and code-oriented design references.

| Approach                           | Directly Represented Response                                           | Main Limitation for EHB-Connected CFST Joints                                                                                                  | Prediction Evidence                                                                 |
|------------------------------------|-------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| Eurocode 3 component method [23]   | Joint moment resistance and initial stiffness by component idealization | Requires idealized component springs; blind-bolt anchorage and tube-face deformation are not fully represented by a dedicated EHB-CFST formula | Mean prediction-to-test ratio $\bar{\lambda} = 0.92$ , CoV = 22.6% for $M_u$ subset |
| Eurocode 4 and AISC checks [56,57] | Composite member resistance and design checks                           | Useful for design bounds, but they do not directly predict joint ductility or the full moment–rotation response of the EHB connection          | Primarily suitable for code compliance rather than multi-output response prediction |
| Proposed IPSO-DNN                  | $M_u$ , $S_{j,ini}$ , and $\mu$ predicted simultaneously                | Applicable only within the database ranges in Table 1                                                                                          | $\bar{\lambda} = 0.99$ , CoV = 9.4% for the same $M_u$ subset                       |

The five-fold cross-validation results (Table 7) confirm that the model generalizes well within the parameter ranges covered by the training data. Nevertheless, the model's applicability remains bounded by the database summarized in Table 1. Extrapolation beyond these ranges requires additional experimental validation because the connection

failure mode may change when variables move outside the observed domain. The current model also treats the three output targets as parallel prediction quantities, whereas physical coupling between  $M_u$ ,  $S_{j,ini}$ , and  $\mu$  exists but is not explicitly enforced. In addition, loading protocol parameters are not included as independent inputs. This may limit direct application to seismic performance prediction when cyclic degradation, pinching, and cumulative damage are of primary concern.

To place the proposed model in direct relation to existing empirical and design-based approaches, a quantitative comparison was carried out using 32 specimens from the test set for which the required variables were available for Eurocode 3 component method or author-proposed empirical calculations. For ultimate moment capacity, the Eurocode 3 component method applied to the blind-bolted CFST joint configuration following the procedure of Tizani et al. [10] gives a mean prediction-to-test ratio of  $\bar{\lambda} = 0.92$  with a coefficient of variation (CoV) of 22.6%. The empirical equation proposed by Wang et al. [14] for flush end-plate joints to CFST columns gives  $\bar{\lambda} = 0.95$  and CoV = 18.9% on the subset of specimens within its application scope. The IPSO-DNN model, evaluated on the same comparable subset, gives  $\bar{\lambda} = 0.99$  and CoV = 9.4%. For initial stiffness, the Eurocode 3 component method gives CoV = 26.1%, whereas the IPSO-DNN gives CoV = 13.7%. The ductility coefficient is not directly predicted by the available empirical formulas, which is one reason for adopting a multi-output data-driven framework.

This comparison should be interpreted with appropriate caution. The Eurocode 3 and empirical formulas were developed from independent datasets and are intended for general application across broader parameter ranges, so their higher scatter on the present database does not imply general inferiority. Normative design models also offer the advantage of explicit physical transparency and direct compatibility with safety factor frameworks. Notwithstanding these caveats, the present results indicate that combining optimization-enhanced training with a carefully organized experimental database can substantially improve prediction reliability for multi-target mechanical response assessment of EHB-connected CFST joints during preliminary design. The  $R^2$  values exceeding 0.95 for all three targets also compare favourably with previously reported ML-based predictions for similar connection types, which typically range from 0.85 to 0.93 [28,37].

## 5. Conclusions

This study developed an IPSO-DNN framework for predicting the mechanical properties of bolted CFST column–steel beam connections. A database of 196 EHB-connected specimens was compiled from published experiments, and 11 input variables were used to predict ultimate moment capacity, initial stiffness, and ductility coefficient. By combining adaptive inertia weight with Lévy flight perturbation, the IPSO algorithm reduced the final fitness value by 53.8% and 69.5% compared with LDWPSO and standard PSO, respectively.

The proposed model achieved  $R^2$  values of 0.9782, 0.9685, and 0.9538 for the three prediction targets, outperforming the tuned shallow neural network (SNN), SVR, Random Forest, and the standalone DNN. Compared with the standard DNN, the  $R^2$  increased by 2.8–3.5%, and the prediction variability decreased from 0.0125 to 0.0023 in the training efficiency comparison. Five-fold cross-validation also showed small standard deviations (0.0020–0.0028), indicating stable performance within the collected data range.

The feature-importance analysis identified steel tube wall thickness, end-plate thickness, and bolt diameter as the dominant variables for ultimate moment capacity. This ranking is consistent with the expected roles of tube-face deformation, end-plate bending, and bolt tensile resistance in blind-bolted CFST joints. The model is valid within the geometric and material ranges summarized in Table 1; outside these ranges, it should not be used without additional experimental or numerical verification. The model can therefore

support preliminary design and parametric screening, although final design still requires code-based checks and engineering validation. Future work should expand the database to include more cyclic tests and connection types, and should add uncertainty quantification for prediction intervals.

**Supplementary Materials:** The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/math14152764/s1>, Table S1: Data Fields; Table S2: Source Distribution; Table S3: Data Extraction and Preprocessing Rules.

**Author Contributions:** Conceptualization, Y.Y. and L.Z.; methodology, Y.Y.; software, Y.Y.; validation, Y.Y. and L.Z.; formal analysis, Y.Y.; investigation, Y.Y.; data curation, Y.Y.; writing—original draft preparation, Y.Y.; writing—review and editing, L.Z.; supervision, L.Z.; project administration, L.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The 196 sets of experimental data on CFST column–steel beam nodes with Extended Hollo-Bolt (EHB) connections were compiled from published studies cited in the references, including the open EHB experimental data record available at <https://doi.org/10.1016/j.dib.2024.111088>. The processed database, source identifiers, input/output variables, data-extraction notes, and reproducible scripts are available in the public GitHub repository at [https://github.com/koopman-mpc-lab/ipsodnn\\_cfst-anonymous-src](https://github.com/koopman-mpc-lab/ipsodnn_cfst-anonymous-src) (accessed on 12 July 2026).

**Acknowledgments:** The authors gratefully acknowledge the researchers who have made their experimental data publicly available through the published literature.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

|        |                                      |
|--------|--------------------------------------|
| CFST   | Concrete-Filled Steel Tube           |
| DNN    | Deep Neural Network                  |
| IPSO   | Improved Particle Swarm Optimization |
| PSO    | Particle Swarm Optimization          |
| EHB    | Extended Hollo-Bolt                  |
| SNN    | Shallow Neural Network               |
| SVR    | Support Vector Regression            |
| RF     | Random Forest                        |
| RMSE   | Root Mean Square Error               |
| MAE    | Mean Absolute Error                  |
| MSE    | Mean Squared Error                   |
| ReLU   | Rectified Linear Unit                |
| LDWPSO | Linearly Decreasing Weight PSO       |
| VIF    | Variance Inflation Factor            |
| PDP    | Partial Dependence Plot              |
| TLBO   | Teaching–Learning-Based Optimization |
| EO     | Equilibrium Optimizer                |

## References

1. Ataei, A.; Bradford, M.A.; Valipour, H.R. Experimental study of flush end plate beam-to-CFST column composite joints with deconstructable bolted shear connectors. *Eng. Struct.* **2015**, *99*, 616–630. [[CrossRef](#)]
2. Ataei, A.; Bradford, M.A.; Liu, X. Computational modelling of the moment–rotation relationship for deconstructable flush end plate beam-to-column composite joints. *J. Constr. Steel Res.* **2017**, *129*, 75–92. [[CrossRef](#)]
3. Han, L.H.; Li, W.; Bjorhovde, R. Developments and advanced applications of concrete-filled steel tubular (CFST) structures: Members. *J. Constr. Steel Res.* **2014**, *100*, 211–228. [[CrossRef](#)]

4. Han, L.H.; Yao, G.H.; Zhao, X.L. Tests and calculations for hollow structural steel (HSS) stub columns filled with self-consolidating concrete (SCC). *J. Constr. Steel Res.* **2005**, *61*, 1241–1269. [[CrossRef](#)]
5. Varma, A.H.; Ricles, J.M.; Sause, R.; Lu, L.W. Seismic behavior and modeling of high-strength composite concrete-filled steel tube (CFT) beam-columns. *J. Constr. Steel Res.* **2002**, *58*, 725–758. [[CrossRef](#)]
6. Tao, Z.; Hassan, M.K.; Song, T.Y.; Han, L.H. Experimental study on blind bolted connections to concrete-filled stainless steel columns. *J. Constr. Steel Res.* **2017**, *128*, 825–838. [[CrossRef](#)]
7. Wang, J.F.; Zhang, L.; Spencer, B.F. Seismic response of extended end plate joints to concrete-filled steel tubular columns. *Eng. Struct.* **2013**, *49*, 876–892. [[CrossRef](#)]
8. Debnath, P.P.; Xu, F.; Chan, T.M. Load transfer mechanism in concrete-filled steel tubular columns: Developments, challenges and opportunities. *J. Constr. Steel Res.* **2023**, *203*, 107781. [[CrossRef](#)]
9. Pitrakkos, T.; Tizani, W. Experimental behaviour of a novel anchored blind-bolt in tension. *Eng. Struct.* **2013**, *49*, 905–919. [[CrossRef](#)]
10. Tizani, W.; Al-Mughairi, A.; Owen, J.S.; Pitrakkos, T. Rotational stiffness of a blind-bolted connection to concrete-filled tubes using modified Hollo-bolt. *J. Constr. Steel Res.* **2013**, *80*, 317–331. [[CrossRef](#)]
11. Debnath, P.P.; Chan, T.M. Tensile behaviour of headed anchored hollo-bolts in concrete filled hollow steel tube connections. *Eng. Struct.* **2021**, *234*, 111982. [[CrossRef](#)]
12. Debnath, P.P.; Chan, T.M. Blind bolts with headed anchors under combined tension and shear. *J. Constr. Steel Res.* **2021**, *179*, 106546. [[CrossRef](#)]
13. Goldsworthy, H.M.; Gardner, A.P. Feasibility study for blind-bolted connections to concrete-filled circular steel tubular columns. *Struct. Eng. Mech.* **2006**, *24*, 463–478. [[CrossRef](#)]
14. Wang, J.F.; Han, L.H.; Uy, B. Behaviour of flush end plate joints to concrete-filled steel tubular columns. *J. Constr. Steel Res.* **2009**, *65*, 925–939. [[CrossRef](#)]
15. Tizani, W.; Wang, Z.Y.; Hajirasouliha, I. Hysteretic performance of a new blind bolted connection to concrete filled columns under cyclic loading: An experimental investigation. *Eng. Struct.* **2013**, *46*, 535–546. [[CrossRef](#)]
16. Cabrera, M.; Tizani, W.; Ninic, J. A review and analysis of testing and modeling practice of extended Hollo-Bolt blind bolt connections. *J. Constr. Steel Res.* **2021**, *183*, 106763. [[CrossRef](#)]
17. Cabrera, M.; Tizani, W.; Ninic, J. Tensile experimental data from monotonic loading on connections to concrete filled steel tubes using Extended Hollo-bolt blind bolts. *Data Brief* **2024**, *57*, 111088. [[CrossRef](#)] [[PubMed](#)]
18. Wang, Z.; Tao, Z.; Li, D.; Han, L.H. Cyclic behaviour of novel blind bolted joints with different stiffening elements. *Thin-Walled Struct.* **2016**, *101*, 157–168. [[CrossRef](#)]
19. Lee, J.; Goldsworthy, H.M.; Gad, E.F. Blind bolted moment connection to sides of hollow section columns. *J. Constr. Steel Res.* **2011**, *67*, 1900–1911. [[CrossRef](#)]
20. Wang, J.; Chen, X.; Shen, J. Performance of CFTST column to steel beam joints with blind bolts under cyclic loading. *Thin-Walled Struct.* **2012**, *60*, 69–84. [[CrossRef](#)]
21. Wang, J.; Guo, S. Structural performance of blind bolted end plate joints to concrete-filled thin-walled steel tubular columns. *Thin-Walled Struct.* **2012**, *60*, 54–68. [[CrossRef](#)]
22. Fan, J.; Zhao, J. Experimental and numerical analysis of assembled steel beam to CFDST column joint. *J. Constr. Steel Res.* **2024**, *218*, 108697. [[CrossRef](#)]
23. EN 1993-1-8; Eurocode 3: Design of Steel Structures—Part 1-8: Design of Joints. CEN: Brussels, Belgium, 2005.
24. Gardner, A.P.; Goldsworthy, H.M. Experimental investigation of the stiffness of critical components in a moment-resisting composite connection. *J. Constr. Steel Res.* **2005**, *61*, 709–726. [[CrossRef](#)]
25. Oktavianus, Y.; Chang, H.; Goldsworthy, H.M.; Gad, E.F. Component model for pull-out behaviour of headed anchored blind bolt within concrete filled circular hollow section. *Eng. Struct.* **2017**, *148*, 210–224. [[CrossRef](#)]
26. Li, Y.L.; Chan, T.M.; Zhao, X.L. Review on blind bolted connections to concrete-filled steel tubes. *Thin-Walled Struct.* **2023**, *183*, 110444. [[CrossRef](#)]
27. Dai, X.H.; Wang, Y.C.; Bailey, C.G. Numerical modelling of structural fire behaviour of restrained steel beam–column assemblies using typical joint types. *Eng. Struct.* **2010**, *32*, 2337–2351. [[CrossRef](#)]
28. Kim, D.; Shim, C.; Lee, J. Prediction of moment–rotation behavior of steel beam-to-column connections using artificial neural networks. *J. Constr. Steel Res.* **2015**, *114*, 63–75.
29. Chou, J.S.; Ngo, N.T.; Pham, A.D. Shear strength prediction in reinforced concrete deep beams using nature-inspired metaheuristic support vector regression. *J. Comput. Civ. Eng.* **2016**, *30*, 04015002. [[CrossRef](#)]
30. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [[CrossRef](#)]
31. Megahed, K.; Mahmoud, N.S.; Abd-Rabou, S.E.M. Application of machine learning models in the capacity prediction of RCFST columns. *Sci. Rep.* **2023**, *13*, 20878. [[CrossRef](#)] [[PubMed](#)]

32. Megahed, K.; Mahmoud, N.S.; Abd-Rabou, S.E.M. Prediction of the axial compression capacity of stub CFST columns using machine learning techniques. *Sci. Rep.* **2024**, *14*, 2885. [[CrossRef](#)] [[PubMed](#)]
33. Zarringol, M.; Patel, V.I.; Liang, Q.Q.; Hassanein, M.F.; Ahmed, M. Machine-learning-based predictive models for concrete-filled double skin tubular columns. *Eng. Struct.* **2024**, *304*, 117593. [[CrossRef](#)]
34. Chen, T.; Guestrin, C. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016*; ACM: New York, NY, USA, 2016; pp. 785–794. [[CrossRef](#)]
35. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
36. Rabi, M.; Jweihan, Y.S.; Abarkan, I.; Ferreira, F.P.V.; Shamass, R.; Limbachiya, V.; Tsavdaridis, K.D.; Pinho Santos, L.F. Machine learning-driven web-post buckling resistance prediction for high-strength steel beams with elliptically-based web openings. *Results Eng.* **2024**, *21*, 101749. [[CrossRef](#)]
37. Thai, H.T. Machine learning for structural engineering: A state-of-the-art review. *Structures* **2022**, *38*, 448–491. [[CrossRef](#)]
38. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)] [[PubMed](#)]
39. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
40. Kennedy, J.; Eberhart, R. Particle swarm optimization. In *Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995*; IEEE: Piscataway, NJ, USA, 1995; pp. 1942–1948.
41. Zhang, Y.; Wang, S.; Ji, G. A comprehensive survey on particle swarm optimization algorithm and its applications. *Math. Probl. Eng.* **2015**, *2015*, 931256. [[CrossRef](#)]
42. Khatir, S.; Tiachacht, S.; Le Thanh, C.; Bui, T.Q.; Abdel Wahab, M. Damage detection in structures using particle swarm optimization combined with artificial neural network. *Smart Struct. Syst.* **2021**, *28*, 1–12. [[CrossRef](#)]
43. Tran-Ngoc, H.; Khatir, S.; Le-Xuan, T.; Tran-Viet, H.; De Roeck, G.; Bui-Tien, T.; Abdel Wahab, M. Damage assessment in structures using artificial neural network working and a hybrid stochastic optimization. *Sci. Rep.* **2022**, *12*, 4958. [[CrossRef](#)] [[PubMed](#)]
44. Shi, Y.; Eberhart, R. A modified particle swarm optimizer. In *Proceedings of the IEEE International Conference on Evolutionary Computation, Anchorage, AK, USA, 4–9 May 1998*; IEEE: Piscataway, NJ, USA, 1998; pp. 69–73.
45. Agheshlui, H.; Goldsworthy, H.; Gad, E.; Fernando, S. Tensile behaviour of anchored blind bolts in concrete filled square hollow sections. *Mater. Struct.* **2016**, *49*, 1511–1525. [[CrossRef](#)]
46. Agheshlui, H.; Goldsworthy, H.; Gad, E.; Yao, H. Tensile behavior of groups of anchored blind bolts within concrete-filled steel square hollow sections. *J. Struct. Eng.* **2016**, *142*, 04015125. [[CrossRef](#)]
47. Wang, Y.; Wang, Z.; Pan, J.; Wang, P. Cyclic behavior of anchored blind-bolted extended end-plate joints to CFST columns. *Appl. Sci.* **2020**, *10*, 904. [[CrossRef](#)]
48. Wang, Y.; Wang, Z.; Pan, J.; Wang, P. Seismic behavior of a novel blind bolted flush end-plate connection to strengthened concrete-filled steel tube columns. *Appl. Sci.* **2020**, *10*, 2517. [[CrossRef](#)]
49. Rao, R.V.; Savsani, V.J.; Vakharia, D.P. Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. *Comput. Aided Des.* **2011**, *43*, 303–315. [[CrossRef](#)]
50. Faramarzi, A.; Heidarinejad, M.; Stephens, B.; Mirjalili, S. Equilibrium optimizer: A novel optimization algorithm. *Knowl.-Based Syst.* **2020**, *191*, 105190. [[CrossRef](#)]
51. Abualgah, L.; Diabat, A.; Mirjalili, S.; Abd Elaziz, M.; Gandomi, A.H. The arithmetic optimization algorithm. *Comput. Methods Appl. Mech. Eng.* **2021**, *376*, 113609. [[CrossRef](#)]
52. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems, Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017*; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 3146–3154.
53. Prokhorenkova, L.; Gusev, G.; Vorobev, A.; Dorogush, A.V.; Gulin, A. CatBoost: Unbiased boosting with categorical features. In *Advances in Neural Information Processing Systems, Proceedings of the 32nd International Conference on Neural Information Processing System, Montreal, QC, Canada, 3–8 December 2018*; Curran Associates Inc.: Red Hook, NY, USA, 2018; pp. 6638–6648.
54. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, MA, USA, 2006.
55. Friedman, J.H. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [[CrossRef](#)]
56. EN 1994-1-1; Eurocode 4: Design of Composite Steel and Concrete Structures—Part 1-1: General Rules and Rules for Buildings. CEN: Brussels, Belgium, 2004.
57. AISC. *Specification for Structural Steel Buildings, ANSI/AISC 360-22*; American Institute of Steel Construction: Chicago, IL, USA, 2022.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.