scientific reports

OPEN

Structural optimization of framed automotive body-in-white using back propagation neural network optimized based on the sparrow search algorithm

Hong Fu[1✉], Yutan Li[1], Duanjun Zhang[1], Pengcheng Lu[1] & Chao Ma[1,2]

This study develops an enhanced surrogate modeling method integrating back propagation neural network (BPNN) with an improved sparrow search algorithm (SSA) reinforced by reinforcement learning (ISSA-RL). The SSA algorithm is substantially modified through Tent chaotic mapping for population initialization to improve distribution uniformity, combined with a nonlinear adaptive weighting strategy to better balance global and local search capabilities. A multi-agent reinforcement learning framework based on Q-learning is incorporated to dynamically adjust search strategies according to prediction error and population diversity metrics. The proposed method demonstrates high predictive accuracy, which is rigorously validated through benchmark functions and engineering applications. The BPNN-based surrogate model effectively replaces computationally expensive finite element analyses, while uncertainty quantification techniques enhance model robustness against material property fluctuations and loading variations. To determine the optimal configuration of a framed body-in-white (BIW), a structural optimization considering four types of discrete design variables is formulated and optimized by the proposed method. The results show that a 29.2% mass reduction and higher computational efficiency are achieved. The integration of multi-variable optimization with enhanced neural network training and intelligent search algorithms significantly improves both design quality and computational efficiency for complex BIW structures.

Framed structures refer to the space frame structure formed mainly by shaped pipes at the joint[1]. Recently, more and more body-in-white (BIW) of new energy vehicles are framed structures. This type of BIW is easily altered and updated, because the main structure of the BIW is welded by easy-to-manufacture, shaped pipes, and the number of stamping parts is greatly reduced compared to the common BIW. Thus, the investment in the stamping dies is greatly reduced, too. In addition, the BIW is easy to update, which is conducive to improving the product update iteration speed and making it adapt to the increasingly competitive market.

However, it is still a challenge to determine the layout, cross-sectional shapes and sizes and material types of members in the framed BIW. To determine the layout of the members, most researchers use the layout/topology optimization method to optimize the framed structure. This involves adding multiple pipes to the existing structure, and subsequently removing some of these pipes during the optimization process[2,3]. Finally, the optimal layout can be found. In the optimization of the cross-sectional shape of the member, the coordinates of the key nodes in the cross-section are typically used as the design variables to obtain the optimal cross-sectional shape[4,5]. However, the cross-sectional shape obtained by this method is mostly irregular and sometimes even violates the requirements of manufacturing. The sizing optimization of the cross-section of the member can be divided into the continuous sizing optimization and the discrete sizing optimization[3,6–8]. The design variable used in continuous sizing optimization is a continuous value, which needs to be rounded after optimization. Thus, the performance of the optimal design may be affected[9]. The discrete sizing optimization uses the sizes of

[1]School of Intelligent Manufacturing, Jiangsu Vocational Institute of Architectural Technology, Xuzhou 221116, Jiangsu, China. [2]School of Automotive Studies, Tongji University, Shanghai 201804, China. ✉email: amg797@163.com

the manufacturer's existing pipes as the design variable. Thus, the optimal design can be put into use directly. Therefore, the discrete sizing optimization is usually more in line with the engineering requirements. In addition, a BIW consists of various types of materials, and a reasonable selection of material types of components to pursue the best body performance has gradually become a research hotspot[10–15].

The aforementioned researches focus on single layout optimization, shape optimization, or sizing optimization. However, researches[1,16] show that single structural optimization may not fully exploit the optimal performance of the structure. A better design can be obtained as more types of design variables are used. For example, when Ma et al.[16] took the cross-sectional shape, size and material type of the members of a framed BIW as design variables, simultaneously, the weight reduction of the BIW reached 12.1%. As the presence and absence of the members are further considered, the weight loss is 17.6%[1]. Li et al.[17] emphasized the need to consider connection reliability alongside structural parameters, avoiding performance degradation caused by mismatched material-connection combinations. Li et al.[18] further demonstrated that simultaneous optimization of geometric parameters (wall thickness) and material attributes can achieve a more balanced trade-off between weight and stiffness.

However, considering multiple types of design variables simultaneously results in a greatly increased degree of variable coupling and optimization complexity. Traditional optimization algorithms, such as the gradient method, are difficult to deal with such problems[12,19]. Although the heuristic algorithm does not rely on gradient information, it has defects such as poor global convergence and high computational cost[20–22]. In recent years, with the development of artificial neural network (ANN) technology, the advantages of ANN in processing speed and simulation of nonlinear systems have attracted more and more researchers to use ANN to establish implicit parametric models[23]. This method reduces the dependence on complex numerical calculations and improves the computational efficiency and global convergence of structural optimization.

Hong et al.[24] use a genetic algorithm-based back propagation neural network (BPNN) to optimize the superstructure of a bus, and a weight reduction of 7.7% is achieved. Li et al.[25] proposed an ANN-based surrogate model to predict the sound insulation characteristics of the floor of a high-speed train, which has been subsequently used for the structural optimization of the floor. The results show that a weight reduction of 10.9 kg is achieved and the weighted sound reduction index is increased by 6.3 dB. Beyond static performance optimization, ANNs have also been validated in dynamic crashworthiness scenarios. Homsnit et al.[26] proposed an ANN-based optimization method for automotive S-rails, balancing energy absorption, initial peak force, and mass to enhance crash safety—this study confirmed that ANNs can effectively capture complex mechanical responses in dynamic load cases, providing a reference for BIW optimization under multi-condition constraints. However, traditional ANN training often relies on gradient-based methods or basic heuristic algorithms (e.g., genetic algorithm (GA)), which suffer from limitations such as premature convergence to local optima and insufficient adaptability to high-dimensional design spaces, especially when dealing with discrete variables (e.g., material type, cross-sectional shape) that are common in BIW design.

From the perspective of optimization algorithms, the sparrow search algorithm (SSA), as an emerging swarm intelligence algorithm inspired by sparrow foraging and anti-predation behavior[27], has shown advantages in solving complex optimization problems due to its strong global search ability. A recent survey on SSA[28] summarized its applications in engineering optimization, noting that while standard SSA outperforms traditional algorithms (e.g., particle swarm optimization (PSO), GA) in some scenarios, it still faces challenges such as uneven initial population distribution and poor balance between global exploration and local exploitation. To address these limitations, multi-objective variants of SSA (e.g., MOSSA[29] have been developed, which use non-dominated sorting and elite retention strategies to handle multi-objective optimization problems and multi-performance constraints coexist—existing SSA-based methods lack adaptive adjustment mechanisms for search strategies, and their integration with surrogate models (especially BPNN) has not been fully explored.

Against this backdrop, the limitations of current research are evident: (1) Surrogate models such as traditional BPNN, Kriging, and response surface methodology often have insufficient predictive accuracy for high-dimensional discrete variables, and their training efficiency needs improvement; (2) Most BIW lightweight studies focus on static load cases (e.g., bending, twisting)[1,16], while dynamic crash performance has not been fully incorporated into the optimization framework; (3) Optimization algorithms for surrogate model training lack adaptability, making it difficult to balance computational efficiency and solution quality in complex design spaces.

Aiming at the lightweight design of the BIW of a certain electric vehicle, this study takes various performances as the objective function, and takes the cross-sectional shape, size, material type and the presence and absence of members as the design variables to optimize the structure. However, the concurrent optimization of four distinct types of design variables—topology, shape, size, and material—poses significant computational and methodological challenges. The primary difficulties include: (1) the high-dimensional, discrete, and strongly coupled nature of the design space, which easily traps conventional algorithms in local optima; (2) the prohibitive computational cost of performing numerous high-fidelity finite element analyses during optimization; and (3) the need for the optimization framework to be robust against inherent uncertainties in material properties and loading conditions.

To remedy the aforementioned issues, a surrogate model based on BPNN optimized by an improved SSA is proposed to solve the structural optimization problems with multiple types of design variables. The remaining parts of the study are organized as follows: In "Methodology", the proposed method is introduced. In "Discrete design variable structural optimization of the framed BIW", the optimization problem is solved based on the proposed method. Finally, the conclusion is presented in "Conclusion".

Methodology

Back propagation neural network

BPNN is a multi-layer inverse feedback neural network model trained by errors. The given input is transmitted from the input layer to the hidden layer, and then transmitted to the output layer after processing by the activation function of the neurons in the hidden layer. When there is an error between the output and the desired output, the error is transmitted from the output layer to the input layer, apportioned to each neuron, and the connection weight is modified. The process is iterated until the optimal weight is obtained and the model is established.

Sparrow search algorithm

SSA is a swarm intelligence algorithm inspired by sparrow foraging and anti-predation behavior[27]. The basic principle of the algorithm is to simulate the interaction between producer and scrounger during sparrow foraging and mimic the characteristics of sparrows in nature that face predators to carry out global searches. SSA performs well in solving complex optimization problems. The mathematical model of SSA can be summarized as follows.

Firstly, a sparrow population, population size n, is randomly generated in the search space:

$$x_{i,j} = lb_j + \eta_{ij} \times (ub_j - lb_j) \quad i = 1, \ldots, n, j = 1, \ldots, d, \eta_{ij} \in (0,1) \tag{1}$$

where $x_{i,j}$ denotes the value of the j-th dimension of the i-th individual, x_i, ub_j and lb_j denote the upper and lower bounds of the $x_{i,j}$, respectively. η_{ij} indicates a random value. d denotes the number of design variables.

Then, the fitness values of individuals are calculated based on the formulation. In SSA, the individual with the higher fitness value serves as the producer who is responsible for searching the better solutions and guiding the direction of the population. Besides the producer, there are scroungers whose role is to follow the producer in searching the design space and trying to achieve a higher fitness value.

The producer updates the location through a global search to further search for possible global optimal solutions. The position of the producer is updated based on the following rule:

$$\mathbf{x}_i^{t+1} = \begin{cases} \mathbf{x}_i^t \exp\left(\frac{-i}{\theta I_{max}}\right) & R_2 < S_T \\ \mathbf{x}_i^t + \mathbf{V} & R_2 < S_T \end{cases} \tag{2}$$

where t denotes the current iteration number. $\mathbf{x}t+1\, i$ is the position of the i-th individual at iteration $t+1$. I_{max} is the maximum iteration number. θ is a random value. R_2 and S_T are alarm value and safety threshold, respectively. $R_2 \in [0, 1]$, $S_T \in [0.5, 1]$. $\mathbf{V}$ is a d-dimensional vector in which its elements obey the normal distribution.

The scrounger adjusts the position by local search to optimize the current solution. The scrounger is updated by Eq. (3):

$$\mathbf{x}_i^{t+1} = \begin{cases} \mathbf{V} \exp\left(\frac{\mathbf{x}_w^t - \mathbf{x}_i^t}{i^2}\right) & i > \frac{n}{2} \\ \mathbf{x}_p^{t+1} + \left|\mathbf{x}_i^t - \mathbf{x}_p^{t+1}\right| \mathbf{A}^+ \mathbf{L} & else \end{cases} \tag{3}$$

where $\mathbf{x}_{t\,p}$ is the optimal position and $\mathbf{x}_{t\,w}$ is the worst position of the producer in the current iteration. $\mathbf{A}$ is a row vector in which each element is 1 or -1, and $\mathbf{A}^+ = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}$.

To avoid the local optima, the population imitates the characteristics of sparrows facing the predators:

$$\mathbf{x}_i^{t+1} = \begin{cases} \mathbf{x}_b^t + \varphi\left|\mathbf{x}_i^t - \mathbf{x}_b^t\right| & f_i > f_g \\ \mathbf{x}_i^t + K\left(\frac{\mathbf{x}_i^t - \mathbf{x}_w^t}{(f_i - f_w) + \varepsilon}\right) & f_i = f_g \end{cases} \tag{4}$$

where $\mathbf{x}_{t\,b}$ is the optimal position of the current iteration. φ is a parameter to control the step size. f_i is the fitness value of the i-th individual. f_g and f_w are the optimal and worst fitness values till to current iteration. ε is a constant value to avoid the zero-division error. K is a random value and $K \in [-1, 1]$.

The optimal solution is updated based on the fitness values for the next iteration. The above steps are repeated until the convergence condition is met.

Improved sparrow search algorithm (ISSA)

To improve the global search ability of SSA, Tent mapping is used to initialize the sparrow population. To accelerate convergence, a nonlinear weight scheme is employed to update the producer position, thereby balancing the local search and global search capabilities.

Tent mapping
The original SSA initializes the population randomly. However, a more uniform distribution of the initial solution can improve the optimization efficiency and accuracy. The chaotic sequence generated by the Tent mapping has good distribution and randomness. Besides the optimization efficiency and accuracy, it is also conducive to the optimization algorithm to escape the local optimal trap. Therefore, Tent mapping is used to initialize the population in this study.

The chaotic sequences generated by the Tent mapping can be mathematically represented by Eq. (5):

$$y_{t+1,j} = \begin{cases} \mu y_{t,j} & y_{t,j} \leqslant \frac{1}{2} \\ \mu(1 - y_{t,j}) & \frac{1}{2} < y_{t,j} < 1 \end{cases} \tag{5}$$

where $y_{t,j}$ is the j-th value in the chaotic sequence of the t-th iteration, $\mathbf{y}_t$, and $\mu \in (0, 2)$ is the chaotic parameter, which is proportional to the chaos.

Therefore, population initialization based on Tent mapping can be expressed by Eq. (6).

$$x_{ij} = lb_j + y_{t,j} \times (ub_j - lb_j) \tag{6}$$

This initialization strategy enhances the population diversity at the start of the optimization, which is crucial for preventing premature convergence and improving the global exploration capability of the algorithm.

Nonlinear weight scheme
To balance the global and local search ability of the algorithm, an adaptive nonlinear weight scheme (Eq. (6)) is proposed to update the position of the producer, which adjusts based on the change of individual fitness value.

$$w_{\mathrm{p}} = \begin{cases} w_{\min} - (w_{\max} - w_{\min})\frac{f_p - f_w}{f_{avg} - f_w}, & f_p \leqslant f_{avg} \\ w_{\max}, & f_p > f_{avg} \end{cases} \tag{7}$$

where f_p, f_{avg} and f_w are producer, average and minimum fitness values, respectively. w_p, $w_{\min}$ and $w_{\max}$ are current, minimum and maximum weights, respectively. In this study, $w_{\min}$ and $w_{\max}$ are 0.4 and 0.9, respectively.

The update rule of the producer can be expressed by Eq. (8).

$$\mathbf{x}_i^{t+1} = \begin{cases} \mathbf{x}_i^t w_p & R_2 < S_T \\ \mathbf{x}_i^t + \mathbf{V} & R_2 < S_T \end{cases} \tag{8}$$

ISSA enhanced by reinforcement learning (ISSA-RL)

To enhance the adaptive search capability of the algorithm in complex design spaces, the ISSA is integrated with the Q-learning reinforcement learning framework to construct a multi-agent collaborative optimization model, e.g., ISSA-RL. This model achieves a dynamic balance between global exploration and local exploitation through information exchange and strategy learning among agents.

Multi-agent system architecture
The sparrow population is divided into several cooperative subgroups (agents), and each subgroup is responsible for the search of a specific subset of design variables. The decision-making process of the agent is implemented based on Q-learning:

State space definition: the state space is defined as $\mathbf{s} = \{e, H, t, I_d\}$.

where e, H, t and I_d denote the prediction error of the agent model, the entropy value of population diversity, the current iteration number and the collaboration density of subgroups, respectively. The prediction error of the agent model can be expressed as:

$$e = \frac{|\hat{y} - y|}{\hat{y}} \tag{9}$$

where $\hat{y}$ is the predicted value of BPNN and y is the true value obtained by the finite element model. e is used to determine the timing of the search strategy switch.

Action space design: Each agent can perform three basic actions: global exploration (a_1, based on Tent chaotic perturbation), local development (a_2, based on nonlinear weight iteration), and information sharing (a_3, passing the optimal solution to adjacent subgroups). Action selection is based on an epsilon-greedy strategy, balancing exploration and exploitation.

Reward Function Construction:

$$r = \omega_1 \times \Delta f + \omega_2 \times (1 - e) + \omega_3 \times I_d \tag{10}$$

where Δf represents the fitness improvement amount, and ω_1, ω_2 and ω_3 are the weight coefficients. When $e > 0.15$, increase the global exploration reward weight ω_1; When $e < 0.05$, increase the local development reward weight ω_2. The initial weights are set as: $\omega_1 = 0.4$, $\omega_2 = 0.4$, $\omega_3 = 0.2$, respectively.

The design of the state space, particularly the inclusion of population entropy H, allows the agent to monitor and actively promote diversity. The action space provides explicit mechanisms for balancing exploration (action a_1) and exploitation (action a_2). The reward function then reinforces this balance by rewarding fitness improvements achieved through either strategy.

Dynamic collaboration strategy
The agents achieve strategy collaboration through multiple mechanisms. In terms of strategy sharing, the Q-table parameters are exchanged once every 10 generations, and the local strategy is updated by using the weighted average method. Specifically, follow the formula:

$$Q_{local} \leftarrow \tau Q_{local} + (1 - \tau)Q_{global}, \ \tau = 0.7 \tag{11}$$

where τ is the trust coefficient.

The adaptive switching mechanism is adjusted according to the change of the surrogate model error e. When the error e shows an upward trend for three consecutive generations, the global search mode will be triggered,

and the execution probability of τ is increased to 0.8. When the error e below the threshold, the system will switch to the local development mode, and τ is set to 0.7.

In conflict resolution, if the optimal solution of a subgroup conflicts and the fitness difference exceeds 5%, a compromise solution is selected through a voting mechanism to prevent the collaboration process from falling into a local optimum.

Validation of the proposed algorithm

Comparative analysis with well-known heuristic algorithms

To further validate the optimization performance of the proposed ISSA-RL algorithm, comprehensive comparative experiments are conducted against several well-established heuristic optimization algorithms. The GA and PSO are selected as benchmarks due to their proven effectiveness in structural optimization problems. GA demonstrates robust global search capability, particularly suited for discrete and mixed-variable optimization[30], while PSO exhibits efficient local search characteristics and rapid convergence in continuous design spaces[31]. All algorithms are configured with consistent parameters ($n = 100$, $I_{max}=500$) to ensure fair comparison. Specifically, GA employs tournament selection with a crossover rate of 0.8 and mutation rate of 0.05; PSO uses inertia weight of 0.5 with cognitive and social coefficients are of 1; standard and improved SSA variants maintain safety threshold $S_T=0.8$, and alert value $R_2=0.7$; while ISSA-RL incorporated the reinforcement learning framework with a learning rate $\alpha = 0.8$, discount factor $\gamma = 0.95$, and exploration rate $\xi = 0.95$.

The evaluation utilized three benchmark functions from the CEC 2017 test suite[32] (F1-F3, as can be seen in Eq. (12)) with $d = 10$. As can be seen from Fig. 1, ISSA-RL demonstrated superior performance across all test functions, achieving the lowest fitness values (F1: 1.94e-61, F2: 1.39e-31, F3: 3.09e-59). The convergence curves in Fig. 1 reveal that ISSA-RL not only exhibited the fastest initial convergence rate but also maintained the most stable convergence behavior in later stages. This performance superiority can be attributed to the integrated optimization mechanisms: Tent chaotic mapping ensures diverse population initialization, the nonlinear adaptive weight strategy effectively balances exploration and exploitation, and the reinforcement learning framework dynamically adjusts search strategies based on real-time feedback. In comparison, the SSA and the ISSA showed susceptibility to local optima, while GA and PSO demonstrated slower convergence speeds in handling these high-dimensional problems.

$$F1 : f_1(\mathbf{x}) = x_1^2 + 10^6 \sum_{i=2}^{d} x_i^2$$

$$F2 : f_2(\mathbf{x}) = \sum_{i=1}^{d} x_i^2 + \left(\sum_{i=1}^{d} 0.5x_i \right)^2 + \left(\sum_{i=1}^{d} 0.5x_i \right)^4 \qquad (12)$$

$$F3 : f_3(\mathbf{x}) = \sum_{i=1}^{d-1} \left(100\left(x_i^2 - x_{i+1} \right)^2 + (x_i - 1)^2 \right)$$

Sensitivity analysis of parameters of ISSA-RL

To evaluate the influence of key parameters on the performance of the ISSA-RL algorithm, a systematic sensitivity analysis is conducted using the control variate method. Four critical parameters are examined: the learning rate (α), which controls the update step size of the Q-table; the discount factor (γ), which balances immediate and future rewards; the exploration rate (ξ), governing the trade-off between exploration and exploitation; and the stagnation threshold (T_R), which triggers forced exploration after a period of no improvement. The tested value ranges are as follows: $\alpha = [0.2, 0.5, 0.8, 0.9]$, $\gamma = [0.7, 0.8, 0.9, 0.95]$, $\xi = [0.6, 0.8, 0.9, 0.98]$, and $T_R = [10, 20, 30, 40]$. For each parameter configuration, the algorithm is run 10 times independently on benchmark functions F1–F3, and the average best fitness (lower is better) and average convergence iteration (fewer iterations are better) are recorded as performance metrics.
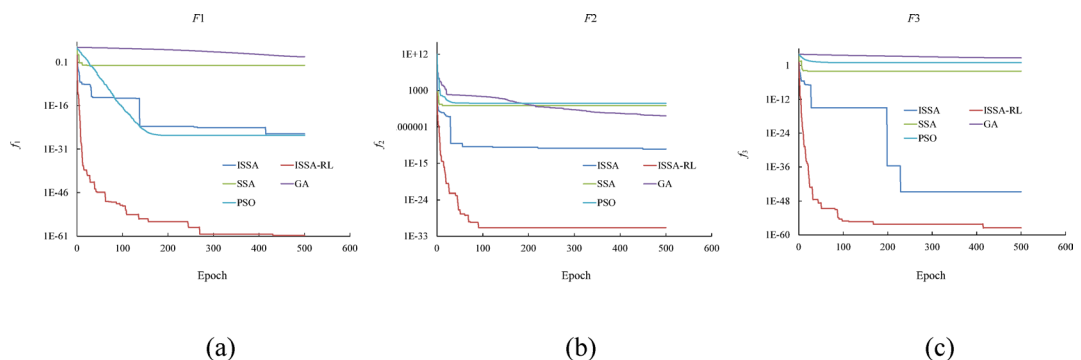

Fig. 1. Convergence diagram of each algorithm. (**a**) *F1*. (**b**) *F2*. (**c**) *F3*.

As illustrated in Fig. 2, which plots parameter values against the average best fitness on a logarithmic scale across different test functions, the following patterns are observed:

The performance of ISSA-RL is found to be largely insensitive to variations in the learning rate. As shown in Fig. 2, the algorithm maintains stable performance across a wide range of α values (from 0.2 to 0.9). Consequently, the learning rate is set to 0.8 for all subsequent experiments, as this value lies within the stable performance region and aligns with common practices in the reinforcement learning literature.

As the discount factor (γ): The best performance is observed at $\gamma = 0.95$. A lower γ (e.g., 0.7) leads to myopic behavior that overlooks long-term rewards, whereas $\gamma = 0.95$ effectively balances immediate and future gains.

As the exploration rate (ξ): The highest convergence accuracy is attained at $\xi = 0.98$. An insufficient exploration rate ($\xi = 0.8$) increases the risk of becoming trapped in local optima, while an excessively high rate ($\xi = 0.9$) slows convergence due to over-exploration.

As the stagnation threshold (T_R): The best performance is obtained at $T_R = 20$. An overly small threshold ($T_R = 10$) disrupts exploitation by triggering forced exploration too frequently, whereas a large threshold ($T_R = 40$) delays the algorithm's response to genuine stagnation.

Overall, the ISSA-RL algorithm demonstrates higher sensitivity to ξ and T_R, while exhibiting stronger robustness to variations in α and γ. Based on the experimental results, the following parameter set is recommended: $\alpha = 0.8$, $\gamma = 0.95$, $\xi = 0.98$, and $T_R = 20$.

Explain ability analysis of ISSA-RL using Shapley additive explanations
To enhance the interpretability of the proposed ISSA-RL model and validate the rationality of its optimization mechanism, the Shapley Additive exPlanations (SHAP) framework[33] is employed. SHAP quantifies the contribution of each input feature to the model output based on Shapley values from game theory, enabling a rigorous explanation of complex black-box models like ISSA-RL. The analysis focused on the impact of input features (decision variables and hyperparameters) on the optimization performance.

ISSA-RL is an optimization algorithm rather than a traditional supervised learning model, so direct application of SHAP is infeasible. Instead, an approach that integrates a surrogate model and the SHAP framework is adopted:

The input-output samples are generated by running ISSA-RL under diverse conditions. Input features included 10-dimensional decision variables (within the range of [-100, 100]) and key hyperparameters of ISSA-RL (learning rate α, exploration rate ξ, and number of subpopulations n_a). Outputs are the optimized objective function values (CEC2017 F1). Then, a random forest regressor is trained on the generated samples to approximate the input-output mapping of ISSA-RL. As shown in Fig. 3, feature x_9 exhibits the highest mean
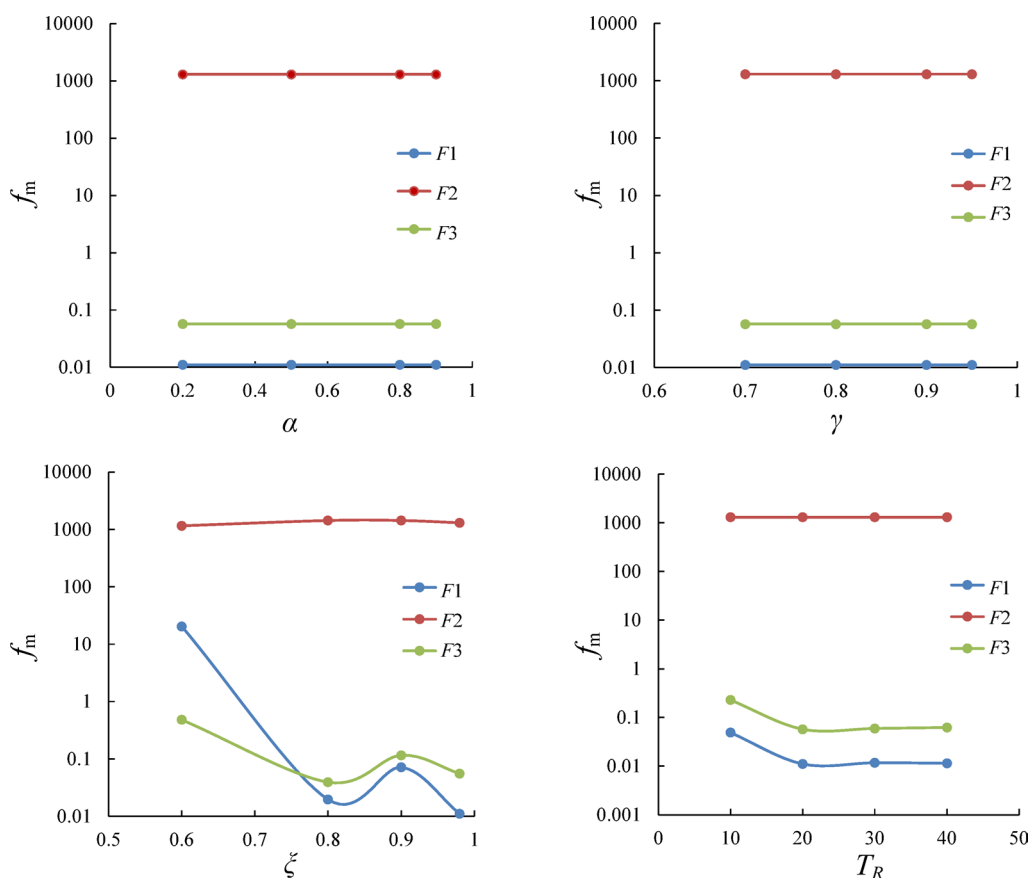

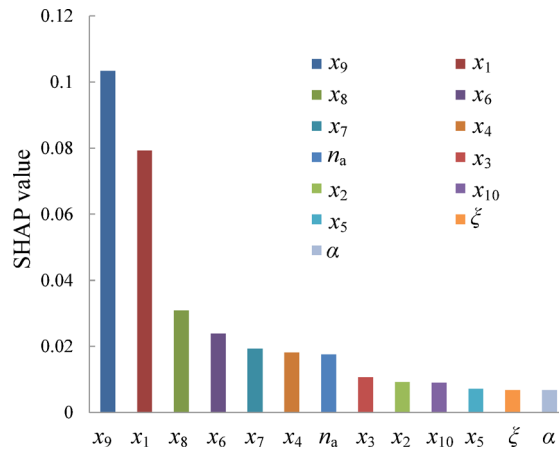**Fig. 2**. Sensitivity analysis of parameters of the ISSA-RL.

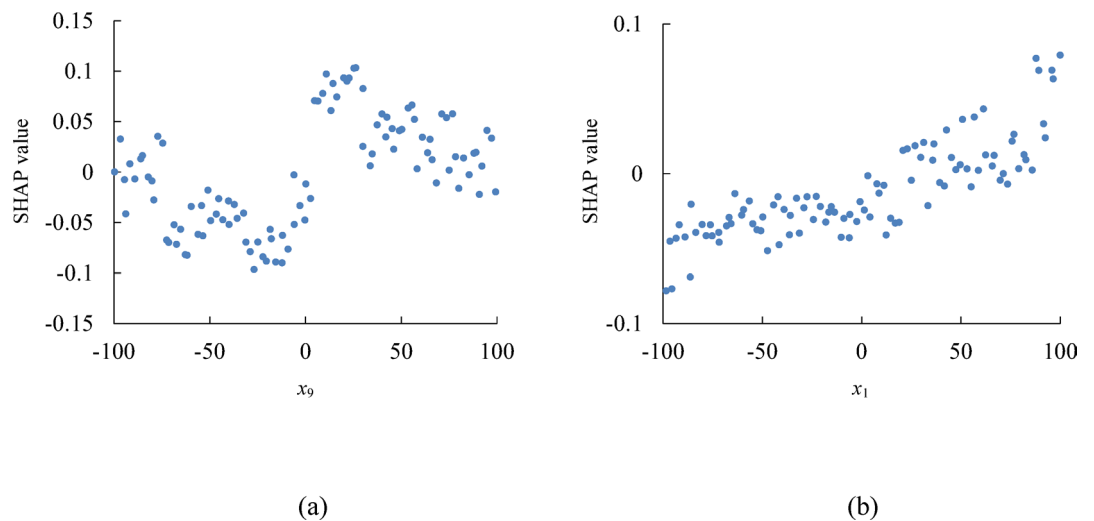**Fig. 3**. Feature importance for optimization performance.

(a)   (b)

Fig. 4. Effects of key features on SHAP value, (**a**) Effect of x_9, (**b**) Effect of x_1.

SHAP value (0.1034), indicating its greatest contribution to the model output. It is followed by x_1 (0.0793) and x_8 (0.0309), ranking as the second and third most influential features, respectively. In contrast, x_5 (0.0072), ξ (0.0068), and α (0.0068) demonstrate relatively minor impacts, with SHAP values approximately only 15% of that of the most significant feature, x_9.

When x_9 values reside within the [-70, -50] range, SHAP values are significantly negative (reaching a minimum of -0.0967), indicating a substantial negative impact on optimization performance, as can be seen in Fig. 4. Conversely, for x_9 values in the[20,30] interval, SHAP values become markedly positive (peaking at 0.1034), reflecting a strong positive contribution. Feature x_1 generates relatively high positive SHAP values at extreme positive (> 90) regions (e.g., 0.0793 at 99.94).

The SHAP-based analysis demonstrates that the optimization performance of ISSA-RL is primarily driven by decision variables x_9, x_1, and x_8, which align with the mathematical characteristics of the benchmark functions used.

Validation on the multi-objective optimization
The global search ability and population diversity of the proposed algorithm are evaluated using the ZDT3 benchmark (Eq. (13))[34]. As the proposed algorithm is a single-objective optimizer, the weighted sum method is employed to convert the ZDT3 problem into a series of single-objective subproblems. The Pareto front is approximated by uniformly varying the weight combinations and aggregating results from multiple independent runs, allowing the assessment of solution diversity. All optimization processes are completed on a computer configured with an Intel Core i5 2.5 GHz processor and 8GB of memory. The algorithm parameters are set as follows: population size 200, maximum number of iterations 2000, the number of producers and the number of sparrows who perceive the danger account for 20% and 10%, respectively, and $S_T = 0.8$.

$$\min : f(\mathbf{x}) = w_1 \times f_1(\mathbf{x}) + w_2 \times f_2(\mathbf{x})$$

$$f_1(\mathbf{x}) = x_1$$

$$f_2(\mathbf{x}) = g(\mathbf{x}) \cdot h(f_1(\mathbf{x}), g(\mathbf{x}))$$

$$g(\mathbf{x}) = 1 + \frac{9}{n-1} \cdot \sum_{i=2}^{n} x_i \tag{13}$$

$$h(f_1(\mathbf{x}), g(\mathbf{x})) = 1 - \sqrt{\frac{f_1(\mathbf{x})}{g(\mathbf{x})}} - \frac{f_1(\mathbf{x})}{g(\mathbf{x})} \cdot \sin(10\pi f_1(\mathbf{x}))$$

$$0 \le x_i \le 1, i = 1, \cdots, n$$

where w_1 and w_2 are weights uniformly sampled from [0, 1] with $w_1 + w_2 = 1$.

Figure 5 shows the distribution of the optimal solutions obtained by the algorithm, where the gray circles represent the true Pareto optimal front (Ground truth), and the green, blue, and red circles represent the Pareto fronts obtained by SSA, ISSA, and ISSA-RL, respectively. It can be seen that the optimal solutions obtained by the algorithm proposed in this paper (ISSA and ISSA-RL) highly overlap with the true Pareto optimal front, among which the distribution of the Pareto front obtained by the ISSA-RL is more uniform.

The inverted generational distance (IGD) metric[34] is further adopted to quantitatively evaluate the convergence and population diversity of the algorithm. IGD reflects the distribution quality of solutions by calculating the geometric distance between the solution set of the algorithm and the ideal Pareto optimal set. IGD can be defined as:

$$\mathrm{IGD}(\mathbf{Pa}_R, \mathbf{Pa}) = \frac{\sum_{\boldsymbol{p}_j \in \mathbf{Pa}_R} d(\boldsymbol{p}_j, \mathbf{Pa})}{|\mathbf{Pa}_R|} \tag{14}$$

where $\mathbf{Pa}_R$ represents the ideal Pareto solution set, $\mathbf{Pa}$ is the solution set generated by the algorithm, $d(\boldsymbol{p}_j, \mathbf{Pa})$ indicates the minimum Euclidean distance from individual $\boldsymbol{p}_j$ to the solution set $\mathbf{Pa}$, and $|\mathbf{Pa}_R|$ is the number of individuals in the ideal solution set. The smaller the dimensionless index value, the better the convergence accuracy and distribution uniformity of the algorithm. To reduce computational errors, each algorithm is independently run 10 times, and the average IGD value is taken as the final performance metric. The experimental results show that the average IGD values of SSA, ISSA, and ISSA-RL are 0.000624, 0.000448, and 0.000336, respectively. The superior IGD value and the more uniform distribution of the Pareto front obtained by ISSA-RL, as shown in Fig. 5, provide quantitative evidence of its excellent balance between exploration and exploitation, and its ability to maintain high population diversity throughout the search process.

Construction of a robust surrogate model based on uncertainty quantification

To enhance the resistance of the surrogate model to various uncertain factors in actual engineering scenarios, the uncertainty quantification (UQ) method is specially introduced to construct a probabilistic BPNN model that can comprehensively consider material performance fluctuations and load errors, enabling the model to maintain stable and reliable prediction performance in complex and changeable engineering environments.
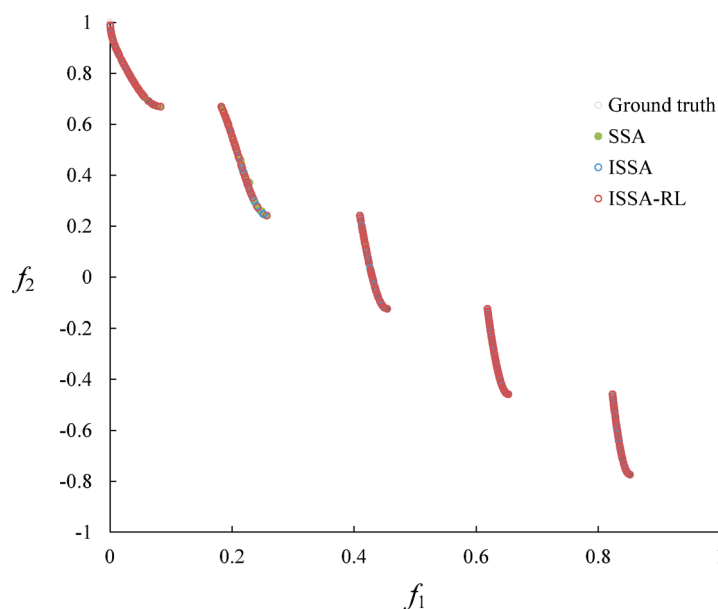

Fig. 5. The Pareto fronts obtained by SSA, ISSA and ISSA-RL on ZDT3.

Modeling of uncertainty sources

In terms of material performance fluctuations, based on the statistical analysis of a large amount of experimental data, the key performance parameters of the material, such as elastic modulus (E) and density (ρ), are modeled as a truncated normal distribution, that is:

$$E \sim \mathcal{N}\left(\mu_E, \sigma_E^2\right)$$
$$\rho \sim \mathcal{N}\left(\mu_\rho, \sigma_\rho^2\right)$$

(15)

where μ and σ_2 denote the mean and the variance, respectively. For the three commonly used engineering materials—steel, aluminum, and magnesium alloy—the coefficients of variation of their performance parameters are selected as 3%, 5%, and 8%, respectively. The coefficients of variation for material properties are determined based on a comprehensive analysis of material certification data from multiple automotive-grade material suppliers and our prior experimental measurements. This statistical data reflects the typical batch-to-batch variability observed in industrial manufacturing processes.

To mimic the uncertainty of the load in practical applications. The load in actual working conditions is regarded as an interval variable. Considering the possible errors in the load measurement and application process in engineering applications, a random disturbance of ± 10% is introduced, which is mathematically expressed as:

$$F = F_0 \times (1 + \delta), \delta \sim \mathcal{U}(-0.1, 0.1)$$

(16)

where F and F_0 denote the actual and the nominal load, respectively. δ is the load uncertainty, a uniform distribution within the interval [-0.1, 0.1]. To integrate these uncertainty sources into the optimization, a sampling strategy is essential. For material properties (E, ρ), the Latin Hypercube Sampling (LHS) method was employed to generate 10,000 samples for each material type. LHS ensures a stratified and space-filling sampling, providing a more accurate estimation of the statistical moments with fewer samples compared to simple random sampling. δ is sampled via LHS, too. These sampled uncertainty parameters are then combined with the deterministic design variables to form the enhanced input vector for the probabilistic BPNN.

Construction of probabilistic BPNN

To enhance the model's ability to handle uncertain factors, the uncertainty quantification of the surrogate model is achieved through the following steps:

1) Input enhancement. Based on the original training samples, uncertainty parameters related to material properties and loads (such as E, ρ, δ) are added to construct the extended input vector $\mathbf{x}_{ext}= [\mathbf{x}, E, \rho, \delta]$, which enables the model to simultaneously learn the comprehensive impact of design variables and uncertain parameters on the target performance.

2) probability prediction layer. Introducing the normal distribution assumption in the output layer of BPNN

$$y \sim \mathcal{N}(\mu_y(\mathbf{x}_{ext}), \sigma_y^2(\mathbf{x}_{ext}))$$

(17)

which not only predicts the mean of the target performance, μ_y, but also simultaneously predicts its variance $\sigma 2$ y and quantifies the uncertainty of prediction results.

The loss function adopts a negative log-likelihood function:

$$\mathcal{L} = \sum \left(\log \sigma_y + \frac{(y - \mu_y)^2}{2 \times \sigma_y^2} \right)$$

(18)

By training the model with this loss function, the prediction accuracy of both the mean and variance can be optimized simultaneously.

3) Model training and verification. The construction and validation of the probabilistic BPNN proceeded through the following steps:

To efficiently train the probabilistic model without performing a prohibitive number of FEA calculations, a data augmentation strategy is employed. A standard deterministic BPNN is first trained on the 700 deterministic FEA samples. Then, for each of these 700 designs, 50 sets of uncertainty parameters (E, ρ, δ) are generated via LHS. The corresponding performance outputs for these uncertain scenarios are synthesized using the pre-trained deterministic BPNN. This process created an augmented training dataset of 35,000 samples ([$\mathbf{x}, E, \rho, \delta$], $\hat{y}$), which teaches the probabilistic BPNN the relationship between input uncertainties and output distributions at minimal computational cost.

The augmented dataset of 35,000 samples is used to train the probabilistic BPNN (with the loss function defined in Eq. (18)), enabling it to predict both the mean μ_y and variance $\sigma 2$ y of any performance response.

The predictive reliability of the trained probabilistic BPNN is rigorously validated using a hold-out test set completely independent of the training process. This verification set consisted of 1,000 samples generated via Monte Carlo Simulation. Each sample is a tuple ($\mathbf{x}, E, \rho, \delta$).

where $\mathbf{x}$ is a design vector from the hold-out set of 300 designs (not used in training either the deterministic or probabilistic BPNN), with its true performance y obtained from FE analysis. (E, ρ, δ) are newly sampled uncertainty parameters drawn from their respective distributions via LHS.

The coverage probability of the 95% confidence interval is calculated by checking whether the true FEA value y falls within the predicted interval $[\mu_y \pm 1.96\sigma_y]$ for all 1,000 samples. This probability is required to be no less than 90% to ensure the model's reliability under uncertainty.

Robust optimization objective correction

To ensure that the optimization results can still maintain good performance in the real engineering environment with uncertainties, the original deterministic optimization objective is extended to a robust objective function:

$$\mathbb{E}\left[f\left(\mathbf{x}\right)\right] + \beta \times Var\left[f\left(\mathbf{x}\right)\right] \tag{19}$$

where E(f) represents the mean of the target performance which reflects the overall expected level of performance; $Var[f]$ represents the variance of the target performance, reflecting the degree of performance fluctuation; β is the robustness coefficient, which is used to balance the relationship between performance expectations and stability, ensuring that the optimization results have strong anti-fluctuation ability while maintaining a high average performance. $\beta = 0.1$ in this study.

BPNN-ISSA-RL surrogate model

In previous studies[1], when heuristic algorithms are used to solve structural optimization problems, the computational efficiency is low because of the large number of finite element analyses. Therefore, this study proposes to use BPNN to construct a surrogate model for the optimization of the BIW to reduce the computational cost.

The flowchart of the BPNN-ISSA-RL surrogate model is shown in Fig. 6, and the main steps can be summarized as follows:

1) data Preparation and preprocessing

Input and normalization: The process begins with the input of sample data (design variables and performance responses) obtained from finite element analysis. All data is normalized to ensure the stability and convergence of the training process.

2) neural network topology definition

The number of nodes in the input layer, hidden layer, and output layer of the neural network is determined. Based on numerical experiments, the final structure is set as follows: 3 nodes in the input layer, 8 nodes in the hidden layer, and 1 node in the output layer.

The activation function for the hidden layer neurons is determined to be the Sigmoid function.

3) ISSA-RL optimization of neural network parameters

This core module is responsible for optimizing and identifying the optimal weights and thresholds for the BPNN. Its internal workflow is:

Population initialization: The Tent chaotic mapping is used to generate the initial sparrow population. Each sparrow individual represents a set of weight and threshold parameters for the BPNN.

Fitness calculation: The fitness value of each individual in the population is calculated.

Role allocation: Individuals are assigned roles (producer, scrounger, or alerter) based on their fitness values.
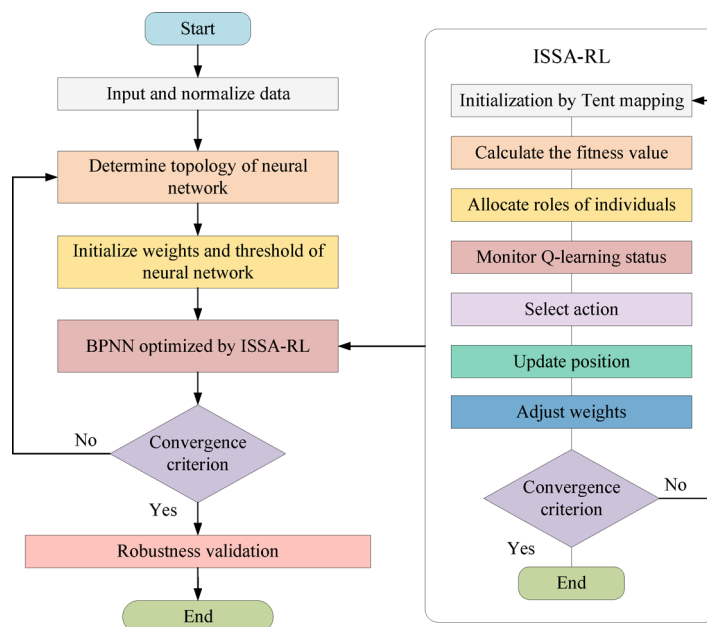

Fig. 6. The flowchart of the back-propagation neural network optimized by the improved sparrow search algorithm.

Q-learning state monitoring: A Q-learning framework is integrated to monitor the state of the agents in real-time (e.g., prediction error, population diversity).

Action selection: Based on mechanisms like the ε-greedy strategy, actions (global exploration, local development, or information sharing) are selected for each agent.

Position update: The position of each individual (i.e., the neural network parameters) is updated according to its role and the selected action, following corresponding rules.

Weight adjustment: A nonlinear adaptive weighting strategy is employed to dynamically adjust the update step size, balancing global and local search capabilities.

Convergence check: The above process iterates until the convergence criteria are met.

4) surrogate model training and validation

The optimal weight and threshold parameters obtained from the ISSA-RL optimization are assigned to the BPNN, finalizing the training of the surrogate model.

The robustness of the trained surrogate model is validated using a test dataset to ensure its predictive accuracy and reliability.

5) efficient optimization application

The validated, trained surrogate model is embedded into the main optimization loop to replace computationally expensive FE analyses. It rapidly predicts performance responses for different design variable combinations, thereby significantly enhancing the computational efficiency of the entire BIW structural optimization process.

This framework successfully constructs a high-precision surrogate model by optimizing BPNN parameters using the intelligent ISSA-RL algorithm, providing an efficient solution for handling complex optimization problems with multiple types of design variables.

Discrete design variable structural optimization of the framed BIW
Introduction to the BIW

The studied BIW of an electric vehicle modeled by shell finite elements (FE) is presented in Fig. 7a. The BIW consists of shaped pipes, such as box-shaped pipes, C-shaped pipes, and tubes. By considering the position of the members and the symmetry, all members are categorized into 36 groups, as indicated in Fig. 7b. The twisting and bending stiffnesses of the BIW are inadequate due to the force feedback from the road cannot be directly transmitted to the A-pillar. To enhance the structural performance of the BIW, this study proposes adding 10 groups of members to the body, as shown in Fig. 7c. During the optimization process, the material type, topology
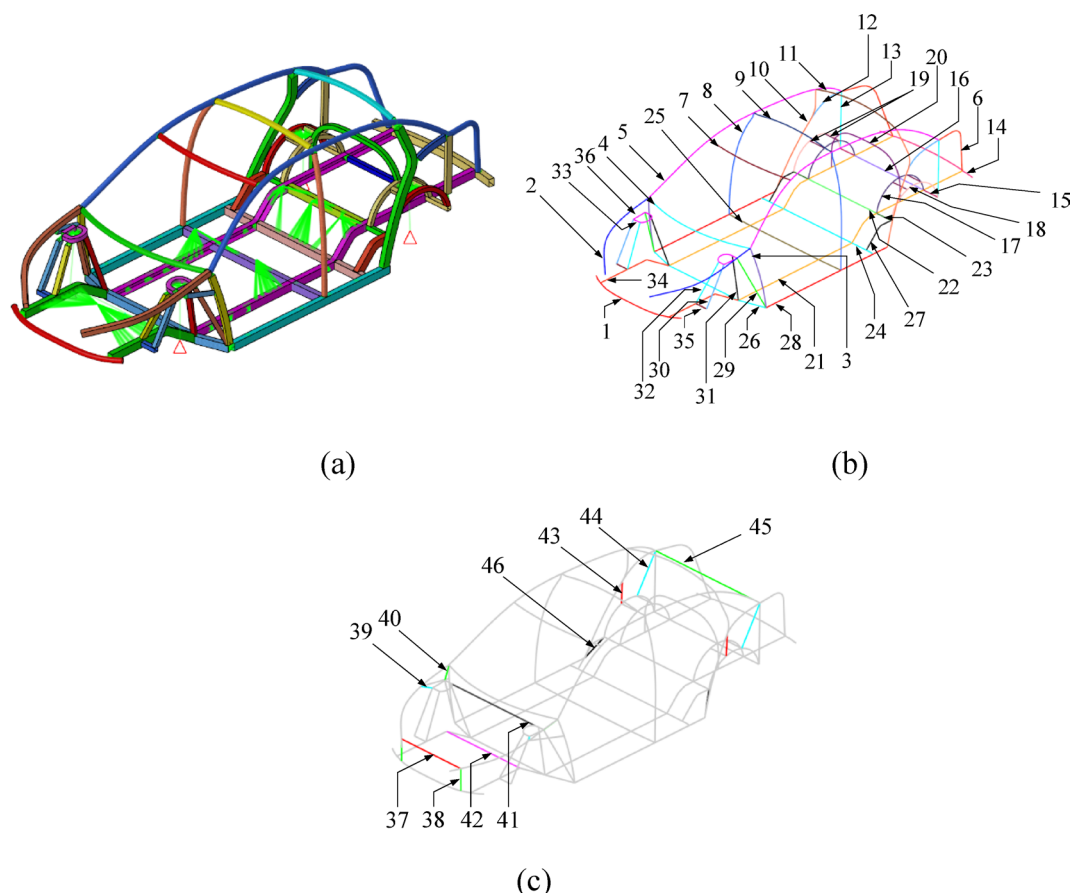

(a)

(b)

(c)

Fig. 7. The BIW and its configuration: (**a**) the shell FE model of the BIW, (**b**) the initial configuration of the BIW and (**c**) members to be added to the BIW.

and cross-sectional shape and size of these 10 groups of members will be optimized. To preserve spatial and layout requirements, the initial 36 groups of members are only optimized the material type, and cross-sectional shape and size. The material types, cross-sectional shapes and sizes of all members, along with the available cross-sectional shapes during optimization, are listed in Table 1. Fe, Al and Mg denote steel, aluminum and magnesium alloy, respectively. The properties of these materials are listed in Table 2. B, T, H and C denote box-shaped pipes, tube, hat-shaped pipes and C-shaped pipes, respectively. The illustration of cross-sectional sizes of pipes is shown in Fig. 8, where w_s represents the width of the cross-section, h represents the cross-sectional

No.	Before optimization				After optimization		
	Material	Shape	Size	Alternative shapes	Material	Shape	Size
1	Fe	T	25×22.5	B, T,H, C	Al	B	50×30×2
2	Fe	B	50×30×2	B, T,H, C	Fe	B	20×20×2
3	Fe	B	60×30×2	B, T	Al	B	70×50×3
4	Fe	T	25×22.5	B, T,H, C	Fe	T	17×15
5	Fe	T	25×22.5	B, T	Fe	B	70×50×2
6	Fe	T	25×22.5	B, T	Al	B	60×30×3
7	Fe	T	25×22.5	B, T,H	Al	B	70×50×3
8	Fe	T	25×22.5	B, T,H	Fe	B	70×50×2
9	Fe	T	25×22.5	B, T	Fe	B	50×50×2.5
10	Fe	B	70×50×2	B, T,C	Fe	B	40×20×1.5
11	Fe	T	25×22.5	B, T,H, C	Fe	B	60×30×2
12	Fe	B	60×30×2	B, T	Fe	B	30×20×2
13	Fe	B	60×30×2	B, T,C	Fe	T	30×28
14	Fe	B	70×50×2	B, T,C	Al	T	25×21
15	Fe	B	40×40×2	B, T	Al	T	30×25
16	Fe	B	40×40×2	B, T	Al	B	30×30×3
17	Fe	B	40×40×2	B, T,C	Mg	B	80×60×4
18	Fe	B	40×40×2	B, T	Fe	B	40×20×2
19	Fe	B	40×40×2	B, T	Fe	B	40×40×2
20	Fe	B	40×40×2	B, T,C	Al	B	80×60×4
21	Fe	C	70×50×2.5	B, T,C	Fe	B	40×30×2
22	Fe	B	70×50×2	B, T,C	Fe	B	20×20×2
23	Fe	B	40×40×2	B, T,C	Fe	T	10×8
24	Fe	B	70×50×2	B, T,C	Al	B	50×30×2
25	Fe	B	70×50×2	B, T,C	Fe	B	70×50×2
26	Fe	B	70×50×2	B, T,C	Fe	B	40×20×1.5
27	Fe	B	70×50×2	B, T,C	Fe	T	30×28
28	Fe	B	70×50×2	B, T,C	Al	B	70×50×3
29	Fe	B	40×40×2	B, T,C	Fe	B	40×40×2
30	Fe	B	40×40×2	B, T,C	Fe	B	20×20×2
31	Fe	B	40×40×2	B, T,C	Al	T	25×21
32	Fe	B	40×40×2	B, T,C	Fe	T	20×18
33	Fe	B	40×40×2	B, T,C	Fe	B	40×30×2
34	Fe	B	70×50×2	B, T,C	Mg	T	30×26
35	Fe	B	40×40×2	B, T,C,	Mg	B	80×60×4
36	Fe	B	40×40×2	B, T,C	Al	B	40×30×2
37	Fe	T	25×22.5	B, T,C,**topo**	-	-	-
38	Fe	T	25×22.5	B, T,C,**topo**	Al	B	50×30×2
39	Fe	T	25×22.5	B, T,C,**topo**	Al	T	25×21
40	Fe	T	25×22.5	B, T,C,**topo**	Mg	B	70×50×3
41	Fe	T	25×22.5	B, T,C,**topo**	-	-	-
42	Fe	T	25×22.5	B, T,C,**topo**	-	-	-
43	Fe	T	25×22.5	B, T,C,**topo**	Fe	B	70×50×2
44	Fe	T	25×22.5	B, T,C,**topo**	-	-	-
45	Fe	T	25×22.5	B, T,C,**topo**	-	-	-
46	Fe	T	25×22.5	B, T,C,**topo**	Fe	B	40×30×2

Table 1. Materials, cross-sectional shapes and sizes of members of the BIW before and after the optimization.

Material	Modulus (GPa)	Poisson's ratio	Density (kg/m³)	Price per unit (¥/kg)
Fe	210	0.3	7900	4.62
Al	70	0.33	2700	15.4
Mg	45	0.35	1840	20

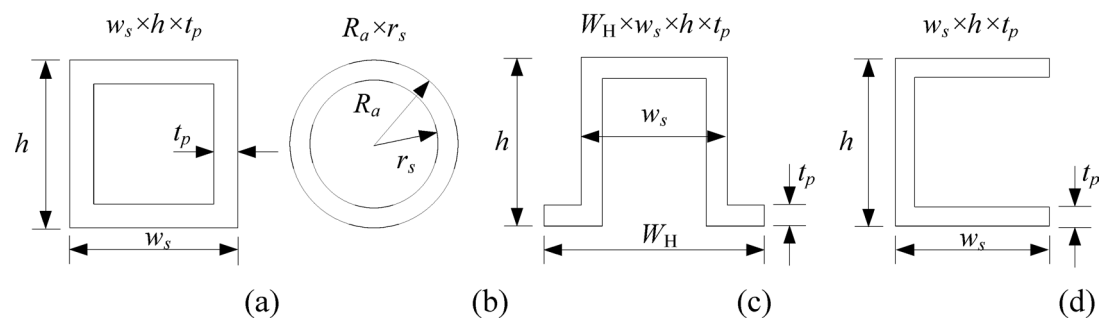**Table 2**. Properties and unit prices of steel, aluminum and magnesium alloy.

Fig. 8. Schematic of cross-sectional shapes and sizes of pipes: (**a**) the square pipe, (**b**) the tube, (**c**) the hat-shaped pipes, and (**d**) the C-shaped pipe.

Load Cases	Constraints at the suspension points				Load/direction
	SS1	SS2	SS3	SS4	
Bending (I)	2,3	3	1,2,3	1,3	2 g/-z
Twisting (II)	-	-	1,2,3	1,3	2000 N·m
Bending and Twisting (III)	2,3	-	1,2,3	1,3	1.3 g/-z
Cornering (IV)	2,3	3	1,2,3	1,3	0.4 g/y, 1.3 g/-z
Braking (V)	2,3	3	1,2,3	1,3	0.8 g/-x, 2 g/-z

Table 3. Boundary conditions of the framed BIW in different load cases.
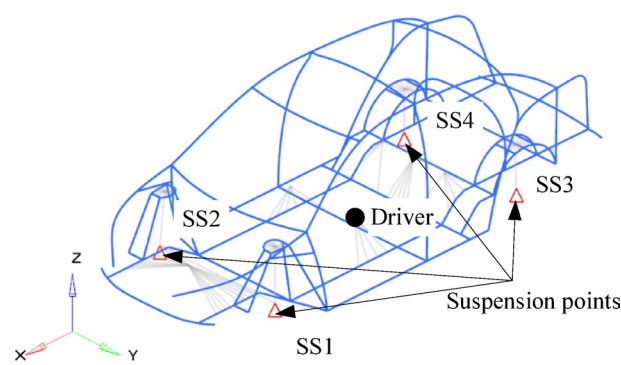
Fig. 9. Beam FE model of the framed BIW.

height, t_p represents the thickness of the pipes, W_H represents the total width of the hat-shaped beam with flange edge, r_s and R_a represent the inner and outer radii of the round pipe, respectively.

The total number of beam finite elements in the studied BIW is 1,082. The loads the BIW undertook include the controller in the front compartment, the battery assembled under the floor, and five adult passengers with their corresponding seats. The controller, the battery pack, each passenger and the seats weigh 40 kg, 200 kg, 75 kg and 15 kg, respectively. All loads are represented by concentrated mass placed at their centers of mass and are connected to the corresponding members of the BIW. During optimization, six common load cases are considered, namely, bending, twisting, bending and twisting, cornering, braking, and free modal. For simplicity, as shown in Table 3, the five load cases, except the free modal condition, are described as Cases I to V, respectively. In each case, the degrees of freedom and loads at the four suspension points (SS1-SS4 in Fig. 9)

are shown in Table 3, where 1, 2, and 3 represent the translational displacements of suspension points in the x, y, and z directions that are restrained, respectively. The symbol '-' indicates suspension point is not restrained. The directions of the loads are consistent with the coordinate system shown in Fig. 9. 2g/-z denotes that the BIW is subjected to twice the downward gravity along the z direction. g represents the gravitational acceleration, and in this study, g is 9.8 m/s². In Case II, the corresponding degrees of freedom of suspension points SS3 and SS4 are restrained, and a pair of forces of equal magnitude but opposite directions along the z-axis are applied at suspension points SS1 and SS2 to simulate the 2000 N·m torque set in the test. The entire computational framework, from the beam-element FE modeling of the BIW to the ISSA-RL optimization and BPNN surrogate training, was implemented using our in-house Python code.

To verify the accuracy of the beam FE model of the BIW, the corresponding detailed shell FE model is established, which includes 99,470 four-node shell elements and 8 three-node shell elements. The boundary conditions and loads of the shell FE model are consistent with those of the beam FE model. The total mass of the structure represented by the beam FE model and the shell FE model, the first-order natural frequency f_1, the maximum displacement of the node representing the driver's center of mass in Case I, $d_{\text{I, max}}$, and the maximum displacement of SS2 in Case II and III, $d_{\text{II, max}}$, $d_{\text{III, max}}$, are shown in Table 4. It is found that the errors between the shell and beam FE models are relatively small. Errors of $d_{\text{II, max}}$ and $d_{\text{III, max}}$ are bigger because the warping and transverse effects are not considered in the beam finite elements. However, the factors that led to these errors are not the focus of this study. Thus, the beam FE model shown in Fig. 9 can be used for the subsequent structural optimization.

Optimization formulation

The structural optimization dealing with four different types of design variables, i.e., the cross-sectional shape, size, material type of members, and topology design variable, can be formulated in Eq. (20):

$$\mathbf{Prop}(\mathbf{topo}, \mathbf{mat}, \mathbf{sect}, \mathbf{dimen}) = [\mathbf{Prop}_1, \mathbf{Prop}_2, \cdots, \mathbf{Prop}_{46}, \theta_j]^T$$

$$\mathbf{x}(i) = [p_1, p_2, \cdots, p_{46}]^T$$

$$\min : \ MSE + \beta \times \text{Var}[MSE] + \lambda \times (1 - I_d(\mathbf{x}))$$

$$\text{s.t.} \ \mathbf{mat} \in [\text{Fe, Al, Mg}]^T$$

$$\mathbf{sect} \in [\text{B}, T, \text{H,C}]^T$$

$$\mathbf{dimen} \in [10 \times 10 \times 1, \cdots]^T$$

$$\mathbf{topo} = 0$$

$$p_j \in \mathbf{Prop}_j \tag{20}$$

$$\text{Cost} = \sum_{i=1}^{46} A_i l_i \text{Price}_k$$

$$\mathbb{E}[d_{m,max}(\mathbf{x})] \leqslant d_m, m = [1, 2, 3, 4, 5]$$

$$\mathbb{E}[\sigma_{m,max}(\mathbf{x})] \leqslant \sigma_m$$

$$\theta_j \sim \mathcal{U}(\theta_{\min,j}, \theta_{\max,j})$$

$$I_d(\mathbf{x}) \geqslant 0.6$$

$$\Delta Q(\mathbf{x}) \leqslant 0.1$$

where $\mathbf{Prop}(\mathbf{topo}, \mathbf{mat}, \mathbf{sect}, \mathbf{dimen}, \theta_j)$ denotes the property library of the member to be optimized. $\mathbf{topo}$ denotes the topology design variable and when a member is assigned the topology design variable, it will be removed from the BIW. $\mathbf{mat}$, $\mathbf{sect}$ and $\mathbf{dimen}$ denote the material, cross-sectional shape and size library, respectively. $\mathbf{Prop}_j$ denotes the property library of the j-th member. $\mathbf{x}(i)$, an d-digit string, denotes one solution and $d = 46$ in this case. p_j denotes the property of the j-th member, which determines the topology, material type, cross-sectional shape, and size of the j-th member. The structure represented by strings can be illustrated in Fig. 10, where $d = 4$. A new collaboration efficiency penalty item, $\lambda \times (1-I_d(\mathbf{x}))$, has been added in the objective function, with $\lambda = 0.05$. A_i, l_i and Price_k denote the cross-sectional area, length and unit price of the material, respectively. All performance indexes adopt the expectation E(·), and are calculated by the probabilistic BPNN model. The Q-table is used to update the stability constraint $\Delta Q \leq 0.1$ (ΔQ represents the rate of change of the Q between adjacent iterations) to ensure the convergence of the multi-agent policy. θ_j is an uncertain parameter

Performance	Shell FE model	Beam FE model	Error
mass (kg)	167.70	176.90	5.49%
f_1(Hz)	15.86	16.11	1.58%
$d_{\text{I, max}}$(mm)	3.85	3.97	3.12%
$d_{\text{II, max}}$(mm)	3.23	2.63	-18.58%
$d_{\text{III, max}}$(mm)	9.56	7.92	-17.15%

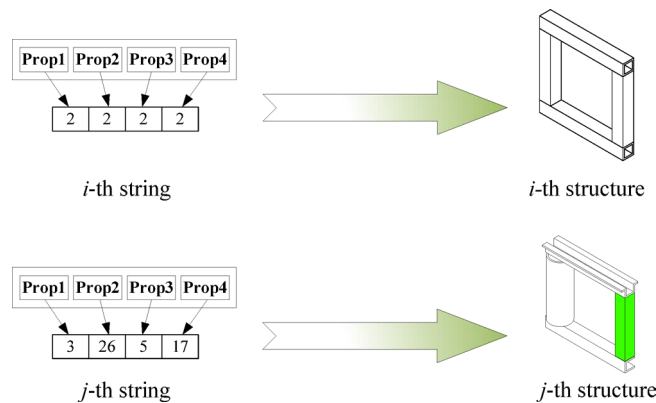**Table 4.** The performance indices of the shell FE model and the beam FE model of the BIW.

Fig. 10. Schematic of the structures represented by strings.

Parameter	Values
Learning rate	0.005
Dropout rate	0.3
Batch size	32
Hidden layer	64 units
Activation function in the hidden layer	ReLU
Activation function in the output layer	sigmoid

Table 5. Network parameters of the LSTM network.

to describe the fluctuations of material properties. d_m denotes the maximal displacement and σ_m denotes the maximal von Mises stress value of the m-th load condition, respectively.

Mean squared error (*MSE*) is used to represent the objective function of the optimization. The smaller the *MSE* value, the higher the accuracy of the model. The expression of *MSE* is shown in Eq. (21):

$$MSE = \frac{1}{q}\sum_{i=1}^{q}\left(f_i(x) - \bar{f}_i(x)\right)^2 \qquad (21)$$

where q represents the number of outputs. $f_i(x)$ represents the i-th output, and $\bar{f}_i(x)$ represents the arithmetic mean of all output responses. The response includes: mass, f_1, $d_{\text{I, max}}$, $d_{\text{II, max}}$, $d_{\text{III, max}}$, maximum von Mises stress value in five load cases: $\sigma_{\text{I, max}}$, $\sigma_{\text{II, max}}$, $\sigma_{\text{III, max}}$, $\sigma_{\text{IV, max}}$, $\sigma_{\text{V, max}}$ and price of the BIW: *Cost*.

The integration of UQ fundamentally changes the optimization process from a deterministic to a robust one. For each candidate design $\mathbf{x}(i)$, the probabilistic BPNN does not output a single performance value but rather a distribution of possible outcomes. The robust objective function (Eq. (19)) is then evaluated based on this distribution.

Crucially, this process automatically penalizes design choices that are highly sensitive to uncertainties. For instance, a component made from magnesium alloy (with its higher 8% variability) might have a promising nominal performance but a large variance in its stress response. The term $\beta \times \text{Var}[f]$ in Eq. (19) will increase the objective value, making this design less favorable unless its mean performance is exceptionally high.

Similarly, a thin-walled section might perform well under ideal loads but could become infeasible (violating stress constraints) under a+10% load perturbation. The expectation operator $\text{E}(\cdot)$ for constraints (e.g., $\text{E}[\sigma_{m,\,\max}(\mathbf{x})] \leq \sigma_m$) requires the constraint to be satisfied not just nominally, but on average across all uncertain scenarios. This can lead to the algorithm excluding such high-risk, discrete design options even if they are deterministically optimal.

To verify the proposed method, the ISSA, ISSA-RL and the genetic algorithm-based BPNN are used to solve the problem, too, and another two types of deep learning methods are employed, i.e., the long short-term memory (LSTM) network and the generative adversarial network (GAN). LSTM is selected for its superior ability to model complex non-linear dependencies between design variables and performance outputs. GAN is chosen because it can learn the underlying data distribution in the performance space, thereby achieving highly accurate predictions. To make the algorithm more convincing, in all cases, 30 independent trials are performed. The maximum number of iterations is 1000 and the population size, n, is set to 100 in each trial. The parameters of the SSA are set as follows: the number of producers and the number of sparrows who perceive the danger account for 20% and 10%, respectively, and $S_T = 0.8$. The parameters of the GA are set as follows: $n_p = 100$, the mutation rate is 0.7, and the crossover rate is 0.2. The parameters of the LSTM network and GAN are optimized based on the ISSA-RL and can be seen in Tables 5 and 6, respectively. The activation function in the output layer of the generator of GAN is a linear function and can be expressed as:

Parameter	Generator	Discriminato
Learning rate	0.0001	0.0001
Hidden layer 1	128 neurons	64 neurons
Hidden layer 2	128 neurons	64 neurons
Activation function in the hidden layers	LeakyReLU	LeakyReLU
Activation function in the output layer	Linear	sigmoid
Optimizer	Adam	Adam

Table 6. Network parameters of the GAN.
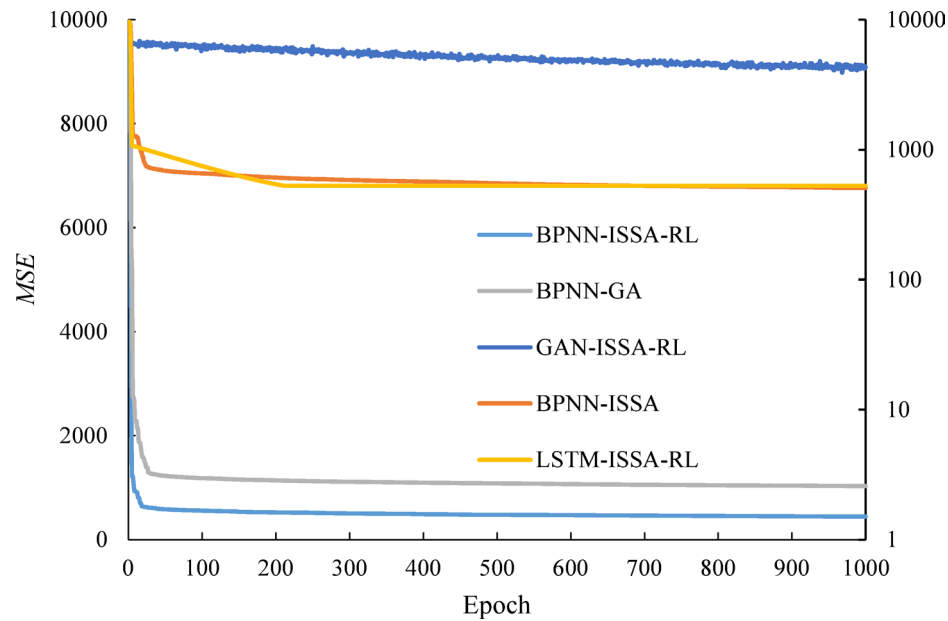
Fig. 11. Fitness values of different neural network models.

$$y = w \cdot x + b \tag{22}$$

where y, w, x and b denote the output, weight, input and bias, respectively.

Results and discussion

The fitness values obtained by different neural network models can be seen in Fig. 11. To facilitate a clear comparison across all models, the figure employs a dual *y*-axis system. The primary axis (left, linear scale) displays the results for BPNN-ISSA-RL, BPNN-GA and GAN-ISSA-RL, highlighting the latter's profound instability and unsuitability for this task. The performance values of the BPNN-ISSA-RL, BPNN-ISSA and LSTM-ISSA-RL models are tightly clustered, making their convergence curves nearly indistinguishable on a linear scale. To resolve this overlap and enable a precise comparison, data obtained by BPNN-ISSA and LSTM-ISSA-RL models are plotted against a secondary *y*-axis with a logarithmic scale. It is clear that neural network models BPNN-ISSA, BPNN-ISSA-RL and LSTM-ISSA-RL are better than those based on BPNN-GA and GAN-ISSA-RL for faster convergence speed. The steeper initial convergence of BPNN-ISSA-RL indicates more effective global exploration, while its ability to reach and refine a superior final fitness value demonstrates strong local exploitation.

As shown in Fig. 11, the optimization result of the surrogate model constructed by LSTM is slightly worse than that of the BPNN-ISSA-RL. Though fine-tuning is implemented, e.g., changing the number of hidden units, using different activation functions and learning rates, the performance of the LSTM network cannot go beyond those shown in Fig. 11. As regard the GAN, the discriminator and generator training rounds are increased, and the learning rate is carefully adjusted, the surrogate model constructed by GAN still failed to simulate the real responses. Despite the sophistication of all networks, the architectural bias of LSTM towards sequential data offered no advantage for this static regression task, often leading to overfitting. Meanwhile, GAN suffered from profound training instability, including mode collapse and difficulty converging, making their predictions unreliable. The BPNN, optimized by ISSA-RL, proved superior due to its direct error minimization, stability, and better suitability for the tabular data structure of the design optimization problem.

The comparison among the performances of the original design and those of the optimal solutions by different methods is shown in Table 7. The optimal solution obtained by BPNN-ISSA-RL shows better performance,

Performance indices	Initial design	Solution by BPNN-ISSA-RL	Solution by BPNN-ISSA	Solution by BPNN-GA	Solution by LSTM-ISSA-RL	Solution by GAN-ISSA-RL
mass (kg)	176.90	125.18	130.70	146.10	137.67	146.92
f_1 (Hz)	16.11	17.02	18.98	16.78	18.49	15.06
$d_{I, z}$ (mm)	3.97	2.39	2.53	4.32	2.40	5.78
$d_{II, z}$ (mm)	2.63	2.39	2.42	3.28	2.29	1.54
$d_{III, z}$ (mm)	7.92	6.86	7.07	9.65	7.54	5.34
$\sigma_{I, max}$ (MPa)	164.87	115.67	144.39	106.25	172.27	126.97
$\sigma_{II, max}$ (MPa)	92.73	138.28	68.93	119.02	113.21	327.35
$\sigma_{III, max}$ (MPa)	203.01	156.22	149.43	189.89	101.17	192.98
$\sigma_{IV, max}$ (MPa)	113.47	110.02	132.36	67.03	129.46	98.88
$\sigma_{V, max}$ (MPa)	154.79	161.72	149.83	178.90	199.10	228.05
Cost (¥)	799.59	910.80	948.49	675.35	986.13	978.91

Table 7. Performance of solutions obtained by different neural network models.
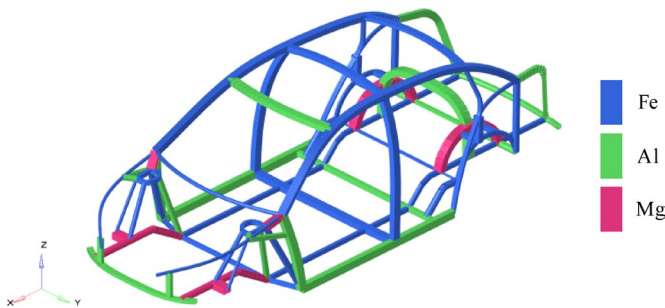
Fig. 12. Optimal solution obtained by the BPNN-ISSA surrogate model.

e.g., lower mass and maximum displacements. Though the solution obtained by BPNN-ISSA-RL gets a larger maximum von Mises stress in Case V, it does not violate the yield stress of all three materials used in this study, and the values can be reduced by a fine mesh and areas have higher maximum stress values can be reinforced in the detailed design stage. The maximum von Mises values considered as objective functions are to avoid thin members appearing in the solution.

In the optimal solution obtained by BPNN-ISSA-RL, members 38, 39, 40, 43 and 46 are retained, and the total mass is reduced by 51.72 kg compared to the initial design, that is, the total mass is reduced by 29.2%. Furthermore, the robust optimization driven by UQ leads to non-intuitive choices that enhance reliability. For example, consider the front anti-collision beam (member group 1). While magnesium alloy offers the lightest weight deterministically, its higher performance variability (8%) resulted in a less robust outcome under load uncertainties. The algorithm ultimately selects a high-strength aluminum alloy (5% variability) for this safety-critical member, demonstrating how the UQ framework automatically prioritizes design stability over risky optimality.

As can be seen from Table 7, most of the performances of the optimal solution are also better than those of the initial design. In combination with Table 1, it can be seen that the materials of the struts located in the suspension support, front anti-collision beam, A-pillar and some transverse members are replaced with aluminum alloy or magnesium alloy, and these pipes are the main body bearing parts, as can be seen from Fig. 12. The mass of the remaining members is mostly reduced, which is also the main reason why the total mass of the structure is significantly reduced, and the performance is improved. Compared with literature[1] (5000 finite element analyses were performed), the computational efficiency of this method is increased by 400% because only 1000 finite element analyses are performed, where 700 groups of results are used as the training set and the rest are used as the verification set.

The displacement plots of the BIW and values of $d_{I, max}$, $d_{II, max}$ and $d_{III, max}$ are shown in Fig. 13.

Conclusion

The optimization design method of framed BIW based on the back propagation neural network proposed in this study reduces the computational cost. Up to four types of design variables are considered simultaneously, i.e., the cross-sectional sizes, shapes, material types, and the presence and absence of the members, which fully exploits the design space and a 29.2% weight reduction is achieved. Therefore, the structure optimization method using multiple types of design variables is conducive to finding an optimal design with lighter weight and better performance. The proposed method shows great potential for complex BIW design. Future work will aim to extend its application to dynamic crash scenarios for more comprehensive validation.
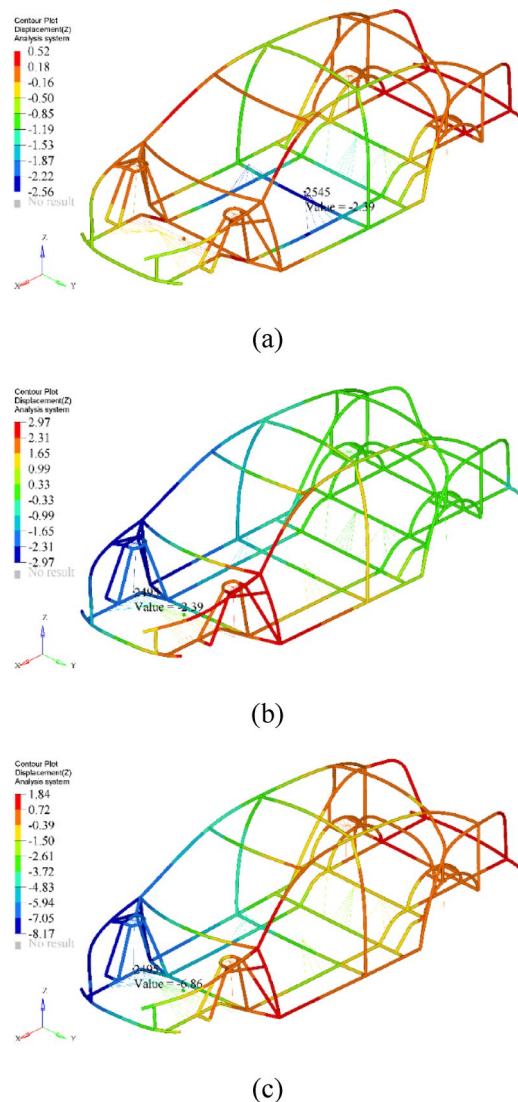
17

(a)

(b)

(c)

Fig. 13. The displacement plots of the BIW in different load cases: (**a**) Case I, (**b**) Case II, and (**c**) Case III.

Data availability

References

1. Ma, C. Discrete sizing, cross-sectional shape, topology optimization, and material selection of a framed automotive body. *Proc. Inst. Mech. Eng., Part D: J. Automob. Eng.* **236**(10–11), 2244–2258 (2022).
2. He, L., Gilbert, J. M. & Song, X. A python script for adaptive layout optimization of trusses. *Struct. Multidiscip Optim.* **60**, 835–847 (2019).
3. Degertekin, S. O., Lamberti, L. & Ugur, I. B. Discrete sizing/layout/topology optimization of truss structures with an advanced Jaya algorithm. *Appl. Soft Comput.* **79** (6), 363–390 (2019).
4. Qin, H., Guo, Y., Liu, Z., Liu, Y. & Zhong, H. Shape optimization of automotive body frame using an improved genetic algorithm optimizer. *Adv. Eng. Softw.* **121** (7), 235–249 (2018).
5. Lieu, Q. X., Do, D. T. T. & Lee, J. An adaptive hybrid evolutionary firefly algorithm for shape and size optimization of truss structures with frequency constraints. *Comp. Struct.* **195** (1), 99–112 (2018).
6. Panagant, N. & Bureerat, S. Truss topology, shape and sizing optimization by fully stressed design based on hybrid grey Wolf optimization and adaptive differential evolution. *Eng. Optim.* **50** (10), 1–17 (2018).

7. Cheng, M. Y., Prayogo, D., Wu, Y. W. & Lukito, M. M. A hybrid harmony search algorithm for discrete sizing optimization of truss structure. *Autom. Constr.* **69** (9), 21–33 (2016).
8. Xu, Y., Gao, Y., Wu, C., Fang, J. & Li, Q. On design of carbon fiber reinforced plastic (CFRP) laminated structure with different failure criteria. *Int. J. Mech. Sci.* 106251. (2021).
9. Sadollah, A., Eskandar, H., Bahreininejad, A. & Kim, J. H. Water cycle, mine blast and improved mine blast algorithms for discrete sizing optimization of truss structures. *Comp. Struct.* **49** (C), 1–16 (2015).
10. Qiu, N., Jin, Z., Liu, J., Fu, L. & Kim, N. H. Hybrid multi-objective robust design optimization of a truck cab considering fatigue life. *Thin-Walled Struct.* **162** (4), 107545 (2021).
11. Fang, J., Sun, G., Qiu, N., Kim, N. H. & Li, Q. On design optimization for structural crashworthiness and its state of the Art. *Struct. Multidiscip Optim.* **55** (3), 1091–1119 (2017).
12. Wu, C. et al. Topology optimisation for design and additive manufacturing of functionally graded lattice structures using derivative-aware machine learning algorithms. *Addit. Manuf.* 103833. (2023).
13. Yoon, H., Lee, K., Lee, J., Kwon, J. & Seo, T. The stiffness adjustable wheel mechanism based on compliant spoke deformation. *Sci. Rep.* **14**, 773 (2024).
14. Xu, Y., et al. Topology optimization for additive manufacturing of CFRP structures. *Int. J. Mech. Sci.* **269** (000), 108967 (2024).
15. Li, S., Wang, D. & Zhou, C. Multi-level structural optimization of thin-walled sections in steel/aluminum vehicle body skeletons. *Appl. Math. Modell.* **132**, 187–210 (2024).
16. Ma, C., Gao, Y., Liu, Z., Duan, Y. & Tian, L. Optimization of multi-material and beam cross-sectional shape and dimension of skeleton-type body. *J. Jilin University(Engineering Technol. Edition)*. **51** (5), 1583–1592 (2021).
17. Li, S., Wang, D., Wang, S. & Zhou, C. Structure-connection-performance integration lightweight optimisation design of multi-material automotive body skeleton. *Struct. Multidiscip Optim.* **66** (198), 03656 (2023).
18. Li, S., Zhou, D. & Pan, A. Integrated lightweight optimization design of wall thickness, material, and performance of automobile body side structure. *Struct. Multidisc Optim.* **67** (95), 03810 (2024).
19. Wu, C. et al. A machine learning-based multiscale model to predict bone formation in scaffolds. *Nat. Comput. Sci.* **1**, 532–541 (2021).
20. Liu, Y., Gu, Z., Sun, M., Guo, C. & Ding, X. Machine learning-based fracture failure analysis and structural optimization of adhesive joints. *Appl. Sci.* **15**, 9041 (2025).
21. Qiu, N., Gao, Y., Fang, J., Sun, G. & Kim, N. H. Topological design of multi-cell hexagonal tubes under axial and lateral loading cases using a modified particle swarm algorithm. *Appl. Math. Modell.* **53**, 567–583 (2018).
22. Xu, X. et al. A feasible identification method of uncertainty responses for vehicle structures. *Struct. Multidiscip Optim.* **64**, 3861–3876 (2021).
23. Yang, H., Geng, X., Xu, H. & Shi, Y. An improved least squares (LS) channel Estimation method based on CNN for OFDM systems. *Electron. Res. Arch.* **31** (9), 5780–5792 (2023).
24. Hong, H. C., Hong, J. Y., Apolito, L. D. & Xin, Q. F. Optimizing lightweight and rollover safety of bus superstructure with multi-objective evolutionary algorithm. *Int. J. Automot. Technol.* **25** (4), 731–743 (2024).
25. Li, Y., Zhang, Y. M., Wang, R. Q. & Tang, Z. Artificial neural network-based sound insulation optimization design of composite floor of high-speed train. *Proc. Inst. Mech. Eng., Part C: J. Mech. Eng. Sci.* **238**(23), 10964–10977 (2024).
26. Homsnit, T., Jongpradist, P., Kongwat, S., Jongpradist, P. & Thongchom, C. Crashworthiness design of an automotive S-rail using ANN-based optimization to enhance performance and safety. *Struct. Multidisc Optim.* **67** (93), 03083 (2024).
27. Xue, J. & Shen, B. A novel swarm intelligence optimization approach: sparrow search algorithm. *Syst. Sci. Control Eng.* **8** (1), 22–34 (2020).
28. Xue, J. & Shen, B. A survey on sparrow search algorithms and their applications. *Int. J. Syst. Sci.* **55** (4), 814–832 (2024).
29. Xue, J., Zhang, C., Wang, M. & Dong, X. MOSSA: an efficient swarm intelligent algorithm to solve global optimization and carbon fiber drawing process problems. *IEEE Internet Things J.* **12** (9), 11940–11953 (2025).
30. Wu, C. et al. Dynamic optimisation for graded tissue scaffolds using machine learning techniques. *Comput. Methods Appl. Mech. Eng.* **425**, 116911 (2024).
31. Qiu, N., Ding, Y., Guo, J. & Fang, J. Energy dissipation of sand-filled TPMS lattices under Cyclic loading. *Thin Wall Struct.* **209** (000), 112848 (2025).
32. Naser, M. Z. et al. A.D. A review of benchmark and test functions for global optimization algorithms and metaheuristics. *WIREs Comput. Stat.* **17**, e70028 (2025).
33. Malakouti, S. M. Leveraging SHapley additive explanations (SHAP) and fuzzy logic for efficient rainfall forecasts. *Sci. Rep.* **15**, 36499 (2025).
34. Zitzler, E., Deb, K. & Thiele, L. Comparison of multiobjective evolutionary algorithms: empirical results. *IEEE Trans. Evol. Comput.* **8** (2), 173–195 (2000).

Author contributions

Conceptualization, Hong Fu and Chao Ma; methodology, Hong Fu; software, Yutan Li; validation, Guang Ma and Pengcheng Lu; formal analysis, Chao Ma; writing—original draft preparation, Hong Fu; writing—review and editing, Chao Ma.

Funding

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to H.F.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.