


Article

An Automated Design Method for Plane Trusses Based on User Preference Information

Xianzhong Zhao ^{1,2}, Tao Zhang ^{1,2} and Weifang Xiao ^{1,*} ¹ College of Civil Engineering, Tongji University, Shanghai 200092, China² Shanghai Qi Zhi Institute, Shanghai 200232, China

* Correspondence: weifangxiao@tongji.edu.cn

Abstract: Structural design is a complicated decision-making process involving multiple qualitative and quantitative factors. Currently, most automated design methods consider only quantitative objectives and constraints, ignoring the qualitative design information that is difficult to define mathematically, such as the user preference for structural shapes. This limits the functionality and efficiency of such design methods. In this study, a design method named STSA-P is proposed for plane trusses to incorporate user preference into the automatic design process. Two main problems are addressed, i.e., how to quantify user preference information and how to coordinate it with other quantitative design objectives. A prediction model of user preference is developed for the first problem by generating the data set and selecting an appropriate machine learning (ML) algorithm. Specifically, a set of truss features quantitatively representing the structural shapes are identified for the truss sample population. Furthermore, an interactive system is developed for collecting user evaluation information as data labels. Strategies for reducing user fatigue are also considered during the evaluation process. A set of numerical experiments are conducted to select the suitable ML algorithm. Regarding the second problem, the physical programming method is modified to construct a new aggregate function which effectively coordinates user preference with other design objectives. A cost function is designed by considering the design constraints. On this basis, the prediction model is incorporated into the Structural Topology and Shape Annealing (STSA) method to form the STSA-P method. Two students are invited to perform a design case using the STSA-P method. It is demonstrated that the results verify the practicality and validity of the proposed method.



Citation: Zhao, X.; Zhang, T.; Xiao, W. An Automated Design Method for Plane Trusses Based on User Preference Information. *Appl. Sci.* **2023**, *13*, 1543. <https://doi.org/10.3390/app13031543>

Academic Editor: Mingfeng Huang

Received: 8 January 2023

Revised: 20 January 2023

Accepted: 23 January 2023

Published: 25 January 2023



Copyright: © 2023 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) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: structural design; plane truss; prediction model; multi-objective optimization; user preference; user fatigue

1. Introduction

Computational methods have been widely applied in various aspects of structural design, including both structural analysis and construction drawings. To further improve design quality and efficiency, plenty of automated design methods, which focused on performing conceptual design independently, have been developed in recent years. A representative one is Structural Topology and Shape Annealing (STSA) [1], which simulates the truss design process well and has been successfully used in engineering practice [2,3]. However, most automated methods such as STSA consider only quantitative objectives and constraints that can be defined mathematically, e.g., minimum weight or stress limit. The ability to analyze qualitative factors reflecting the designer's intuition, experience, and preference is limited. One example is the user's subjective preference for structural shapes. Thus, the qualitative factors can only be considered by designers, which limits the functionality and efficiency of the automated design methods. To overcome this limitation, two main problems are to be solved: first, how to quantify the effects of qualitative design factors so that they can be considered in computational methods; and second, how to coordinate different design goals, including both subjective and objective, so that they can guide the search towards an optimal design.

This study aims at solving these two problems and proposing an innovative conceptual design method for plane trusses based on user preference. Note that, in this paper, user preference means the subjective satisfaction with the design appearance. Considering the first problem, a prediction model that can quantify user preference is proposed to transform the qualitative information into a quantifiable objective. Regarding the second problem, a cost function based on modified physical programming (MPP) is introduced. It coordinates different design objectives effectively and integrates the prediction model into a multi-objective optimization procedure (STSA). This paper is organized as follows: Section 2 reviews the related studies on applying user preference and multi-objective optimization methods. Section 3 gives the main steps to build a prediction model for user preference. The cost function is described in Section 4. After that, a design example is given to demonstrate the effectiveness of the proposed method in Section 5. Finally, conclusions are presented in Section 6.

2. Related Work

2.1. Research on User Design Preference

User design preference is one of the typical qualitative factors. It plays a significant role in the design process. Considerable studies have been devoted to studying the effects of user preference, including research on modeling and quantifying the preference information. In earlier research, user preference was abstracted as some simple rules such as balance, rhythm, and proportionality [4]. Among these rules, the golden ratio is the best-known one and has been widely applied in architectural design. In the field of industrial design, the Kansei Engineering method [5,6] is a popular design method that assesses user preference via questionnaires, interviews, etc., and utilizes the information to meet the user's expectations in the product design. This method provides a reference for the quantitative research of subjective information in other fields.

The research on user preference in structural design can be grossly divided into two categories according to the application form of the information. Some researchers believe that the preference information is almost impossible to quantify. Thus, they integrate the user into the optimization system to replace the fitness function. The preference information is expressed through the user's judgments and choices during the optimization process. Interactive evolutionary computation (IEC) is a typical method of this category, which combines human subjective evaluation with an evolutionary optimization algorithm [7]. Von Buelow et al. [8,9] performed an interactive design for discrete structures with a genetic algorithm (GA). During the process, a user can select some preferred solutions in every generation as the parents for the subsequent generation until the optimal design is achieved. Muller et al. [10,11] extended this method to the design of rigid frame structures. Other than GA, other heuristic algorithms were also used. Felkner et al. [12] proposed an interactive framework for truss design based on the particle swarm optimization (PSO) algorithm. Within the framework, a user can use NURBS curves to draw the initial design and evaluate the solutions iteratively through an interface until the final design is completed [12–14].

Other researchers tend to use user preference as fundamental information and quantify it into formulas or models before the search process. Furuta et al. [15] developed a supporting system for the aesthetic design of dam structures. They used the analytic hierarchy process (AHP) method to represent the user preference in a fitness function. Chikata et al. [16] simulated user aesthetic evaluation information in the design of concrete retaining walls based on neural networks. For the design of discrete structures, a remarkable work was conducted by Bailey and Raich [17]. The authors modeled the user preference based on neural networks and combined it with the GA method for generating Pareto-optimal designs.

There are some disadvantages to the interactive design research. Firstly, user preference is not quantified nor modeled, thus the preference information cannot be reused. Secondly, the lengthy interaction can cause user fatigue problems, which reduce the design quality and efficiency [18]. By contrast, the quantitative research can avoid these problems.

But how to ensure the accuracy of the user preference represented by the formulas or models is still challenging. Therefore, it seems viable to develop a model that can accurately predict user preference and then incorporate the preference information into the structural design process.

2.2. Multi-Objective Optimization Methods

The essence of coordinating different design objectives is to solve a multi-objective optimization problem (MOP) [19]. Inevitable conflicts between different objectives often make it difficult to find an optimal solution that performs best in all objectives. Thus, the solution of MOPs is generally a set of compromise solutions, i.e., the so-called Pareto optimal solutions or non-dominated solutions. There are a variety of methods for solving MOPs, which can be classified based on different criteria [19–22]. Regarding the number of solutions when the search process terminates, some methods obtain only one solution, and these one-solution methods solve the problem mainly through three ways. Firstly, all objective functions are converted into a single objective function, e.g., the weighted-sum method and utopia point method. Secondly, the objective functions are optimized one by one in order of importance, e.g., the hierarchical method [23]. Thirdly, users are allowed to participate in the search process, e.g., interactive optimization methods [10,12]. The other methods obtain more than one solution at the end of the search process. These multi-solution methods are commonly used to provide the Pareto-optimal set (or subset) for user selection, including not only the classical methods such as the normal boundary intersection method [24] and the normal constraint method [25], but also the evolutionary multi-objective optimization methods such as MOGA [26], SPEA [27], and NSGA-II [28].

The difference between the two types of methods is whether the user information works during the search process. In the multi-solution methods, the user information works only when the optimization ends, that is, users select a satisfying solution from the optimal set. This study expects the user information to play a positive role to guide the search process. Therefore, the one-solution methods are more suitable. However, such methods have some limitations, including the difficulty in determining parameters and the sensitivity to the shape or continuity of the Pareto optimal front (objective values of Pareto optimal set) [19]. Taking the weighted-sum method as an example, it is rather difficult for users to determine the weighting values that integrate all objectives with different physical meanings and orders of magnitude, especially when a MOP involves both subjective and qualitative objectives. Furthermore, it is not possible to obtain the solutions on the non-convex portion of the Pareto optimal front by the weighted-sum method [29]. To solve this problem, physical programming (PP) was proposed [30]. Instead of setting the weights, PP enables users to express their ideas or expectations on each objective in an understandable and meaningful way. Based on this information, the class functions for all objectives are built on the same numerical scale. These functions are combined into one function named the aggregate function. Thus, a MOP is transformed into a single-objective optimization problem (SOP). More details about PP are described in Section 4.1. Owing to its simplicity and ability to represent the complete Pareto optimal set [31,32], PP has been applied in various engineering problems [33,34].

3. User Preference Prediction Model

A suitable machine learning (ML) technique and a valid data set for training and testing the model are two essential components of developing a prediction model using artificial intelligence. A data set generally comprises a number of samples with features and labels. The features, which are often represented as a vector, reflect a set of sample attributes of concern. The labels generally represent the target in terms of values or categories. The samples considered in this study are a group of plane truss structures. Accordingly, the features need to reflect the geometric information, and the labels are expected to show the user evaluation information. This section presents the steps of developing a prediction model of user preference for plane truss designs, mainly including

generating samples, identifying features, obtaining labels while reducing user fatigue, and selecting an ML technique.

3.1. Generation of Random Samples

The user preference for plane truss shapes is of concern in this study; thus, a diversity of truss shapes is necessary for the generation of samples. The truss samples are generated by the STSA method in a search space of 2000-unit length and 1000-unit height. In order to meet the shape requirement and generate the universal samples, random loadings with random direction, value, and distribution range are adopted. Figure 1 shows the sample population with 200 truss structures.

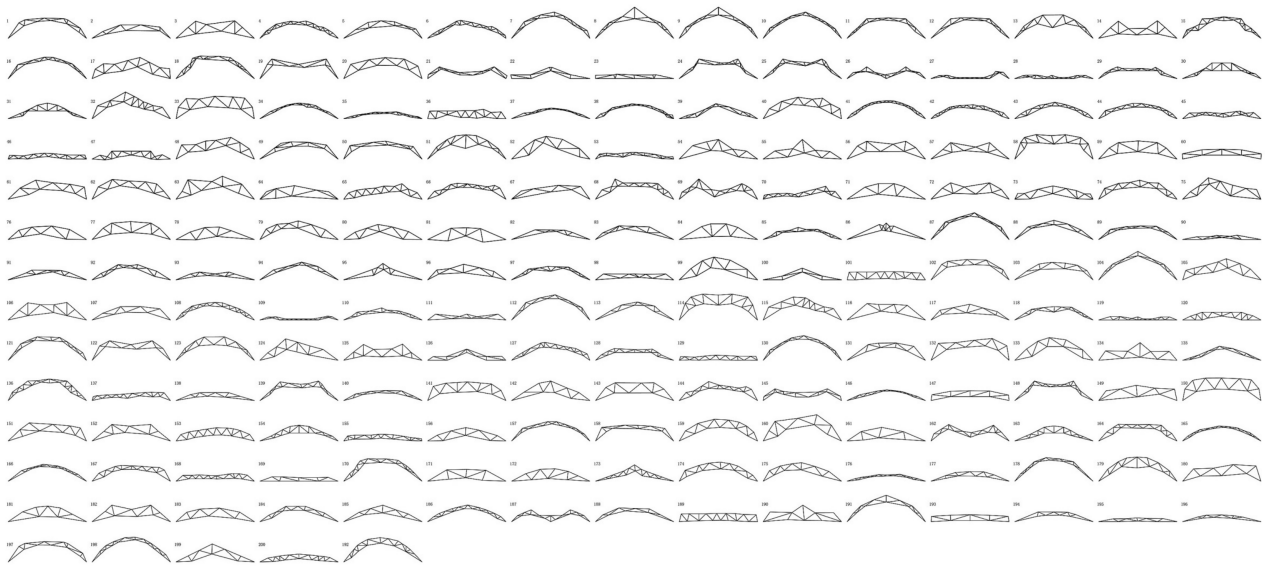


Figure 1. Sample population of truss structures.

3.2. Identification of Truss Features

Visual discrimination is the basis of expressing user preference for truss shapes. Hence, the selected features are required to reflect the geometric characteristics of truss shapes clearly. Then, the users can visually discern the difference between truss samples based on these features. In this study, 26 initial features are defined, and a selection method is proposed to get a group of effective features that adequately represent the truss shapes.

Each feature is defined as a dimensionless value to reflect one aspect of geometric shapes. It can be calculated using structural information such as node coordinates. The definition of the features can be as simple as height-span ratio (HSR), as shown in Equation (1). It can also be as complicated as symmetry rate (SyR), which describes the symmetry of structures, as defined in Equation (2). Figure 2 illustrates the calculation of symmetry rate. A truss structure is divided into two parts by the central axis. The overlapping area can be calculated by folding one part to the other along the central axis. Then, the ratio of the overlapping area to the maximum area of the two parts can be obtained. It is evident that the larger the ratio, the higher the similarity between the two parts, and the more symmetrical the truss structure.

$$HSR = \frac{h_{\max}}{L_{\text{span}}}, \quad (1)$$

where h_{\max} = maximum height of truss nodes, L_{span} = span of truss structures.

$$SyR = \frac{S_o}{\max(S_L, S_R)}, \quad (2)$$

where S_o = overlapping area, S_L = area on the left side of the central axis, and S_R = area on the right side of the central axis.

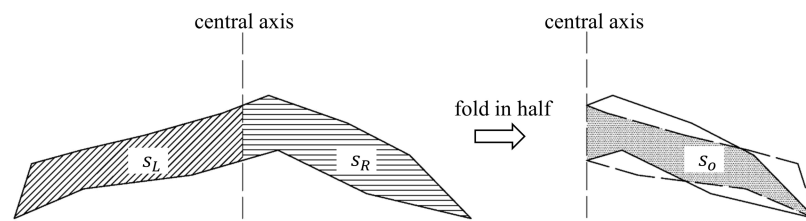


Figure 2. Principle of symmetry rate.

To find the effective features, the K-means clustering method [35] is applied to classify the sample population into nine groups. The classification is based on each truss feature separately. Note that the value of each feature is normalized to 0–1. In addition, the average and the standard deviation of the feature value for each group are calculated. A resulting graph (Figure 3) is plotted as an auxiliary evaluation tool, in which the truss samples belonging to the same group are collected. The truss features are evaluated one by one. Three structural designers were invited to participate in the evaluation process. There are three criteria to select an effective feature. Firstly, visually identifiable similarities should exist between the truss samples within the same group, whereas visually identifiable differences should exist between different groups. Secondly, the average for each group needs to be evenly distributed and the standard deviation needs to be as small as possible. Thirdly, suggestions from the designers need to be taken into account.

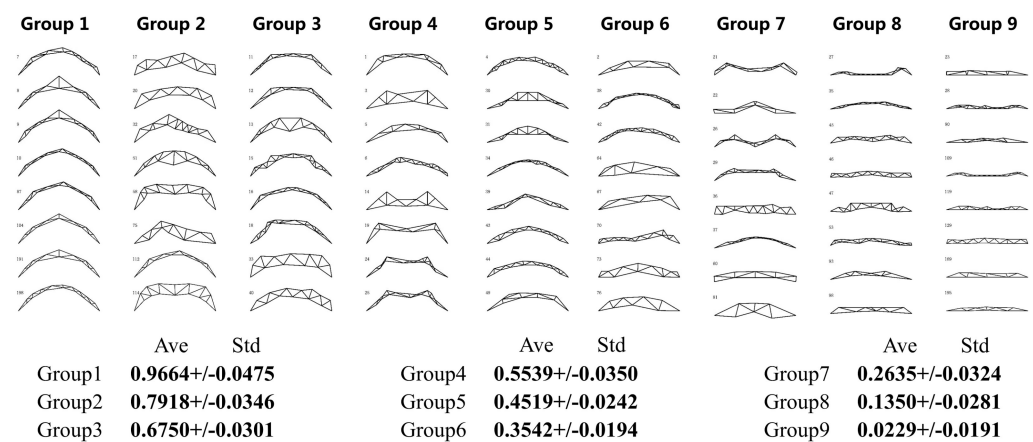


Figure 3. Part of the resulting graph of height-span ratio (HSR).

Taking HSR as an example, Figure 3 shows a part of the resulting graph of the feature. The maximum height of truss samples gradually decreases from group 1 to group 9, whereas the samples in the same group have similar heights. On the other hand, the average for each group is evenly distributed between 0 and 1, and the standard deviation of each group is small. This indicates the good numerical performance of the feature. Furthermore, the designers had no objection to the definition of the feature and affirmed its value to structural design. Therefore, HSR is identified as an effective feature.

After evaluating all the initial features, nine effective features are identified and included in a feature vector, summarized in Table 1.

Table 1. Feature vector.

Basic Size	Complexity	Mid-Span Width	Flatness	Symmetry
Height-Span Ratio	Number of Members	Truss Depth	Top Chord Height Differences	Symmetry Rate
Mid-span Clear Height	Average Length of Web Members	Dispersion of Nodes	Bottom Chord Height Differences	

These features describe five geometric attributes of truss shapes (basic size, complexity, mid-span width, flatness, and symmetry), in which Number of Members (NM), Mid-span Clear Height (MCH), and Average Length of Web Members (AL) can be easily calculated from the basic structural information. Besides, other effective features are expressed by Equations (3)–(5).

$$TD = \frac{\sum_{i=1}^{n_t} h_{tbi} + \sum_{j=1}^{n_b} h_{btj}}{n_t + n_b} \cdot \frac{1}{L_{span}}, \quad (3)$$

where n_t = number of top chord nodes, n_b = number of bottom chord nodes, h_{tbi} = vertical distance from the i th top chord node to the bottom chord, and h_{btj} = vertical distance from the j th bottom chord node to the top chord.

Truss Depth (TD) can be calculated by using Equation (3), which reflects the average width of a truss sample. L_{span} is introduced in the equation to eliminate the size effect. Dispersion of Nodes (DN, Equation (4)) presents the maximum deviation from the average height among all the top and bottom nodes, which reflects the local width of a truss sample. Note that fixed nodes at both ends are excluded from the calculation.

$$DN = \max \left(\max_{1 \leq i \leq n_t} \left| h_{ti} - \frac{\sum_{i=1}^{n_t} h_{ti} + \sum_{j=1}^{n_b} h_{bj}}{n_t + n_b} \right|, \max_{1 \leq j \leq n_b} \left| h_{bj} - \frac{\sum_{i=1}^{n_t} h_{ti} + \sum_{j=1}^{n_b} h_{bj}}{n_t + n_b} \right| \right) \cdot \frac{1}{L_{span}}, \quad (4)$$

where h_{ti} = height of the i th top chord node, and h_{bj} = height of the j th bottom chord node.

$$THD = \left(\sum_{i=2}^{n_t} |h_{ti} - h_{t(i-1)}| + h_{t1} + h_{tn_t} \right) \cdot \frac{1}{L_{span}}, \quad (5)$$

Top Chord Height Differences (THD, Equation (5)) sums the absolute value of the difference in heights between two adjacent nodes (including the two fixed nodes) along the top chord. This feature reflects the flatness of the top chord. Bottom Chord Height Differences (BHD) is defined similarly.

3.3. Strategies for Reducing User Fatigue

Getting the evaluation information of user preference is an interactive process. It is well-known that user fatigue is a huge obstacle to obtaining effective information during this evaluation process. To solve such a problem, two measures are taken, i.e., limiting the number of interactions and reducing the interaction complexity. This section focuses on the former measure and the latter one will be introduced in Section 3.4.

Previous studies have demonstrated that users prefer to group the samples based on their similarity before evaluation [36,37]. Therefore, the user evaluation method “choose first, evaluate later” is proposed. This means that the users first choose a preferred group of samples. Then, they only evaluate the samples in this group. This effectively reduces the number of interactions. Thus, it is vital to find an appropriate clustering method

to group the truss sample population before the user evaluation. This study selects six specific clustering methods as candidates, including K-means, DBSCAN [38], CLIQUE [39], K-medoids [40], SOM [41], and CURE [42] methods. These methods are evaluated from two perspectives, including subjective consistency and numerical performance.

To detect the consistency of the clustering results with user judgment, a subjective evaluation experiment is conducted. Seven students of civil engineering or computer science were invited to compare the truss samples in pairs and rate them based on their similarities. The score varies from 1 to 5. A value of 5 means that the two samples are highly similar, whereas 1 means that they are completely different. The truss samples are randomly selected from the sample population. In total, five random groups (numbered from 1 to 5) are generated, each of which contains 20 samples. Each student needs to evaluate all the groups, in which the samples are presented in the form of pictures, as shown in Figure 4. The final score of each sample pair is calculated from the average of the seven students' scores.

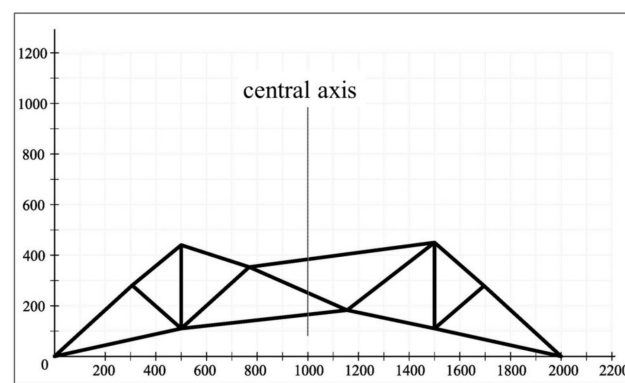


Figure 4. Picture of a truss sample.

After the experiment, the candidate methods are used to cluster the truss samples of each random group. A rating rule is presented to evaluate the clustering results according to the experimental results. Assigning a similar sample pair with a higher score to the same group will result in a better reward, e.g., a clustering method will get 5 points if it assigns a 5-score sample pair to the same group; otherwise it will lose 5 points. Based on this rule, the grade of each candidate method can be calculated. The higher the grade, the better the performance.

The evaluation results are summarized in Table 2. The SOM method achieves the highest grade in groups 1, 3, 4, and 5, whereas the K-means method obtains the highest grade in group 2. By contrast, the CLIQUE method shows the worst performance, with the lowest grades in groups 2, 3, and 4. Generally speaking, the SOM method consistency is most consistent with user judgment.

Table 2. Evaluation results based on the rating rule.

Random Group	CLIQUE	CURE	DBSCAN	K-Means	K-Medoids	SOM
1	611	642	642	635	599	666
2	594	619	619	669	629	656
3	550	639	639	650	627	653
4	632	662	666	677	650	681
5	429	415	480	504	497	510

Regarding the numerical performance, the existing indexes for evaluating clustering results are generally based on two criteria, i.e., Compactness and Separation [43]. Few indexes consider Uniformity, which measures the consistency of the sample size for each cluster. Uniformity is an important evaluation criterion because a large difference in the

sample size between clusters can lead to some problems: for example, too little information will be obtained from a small cluster, whereas user fatigue can be caused by a large cluster. Therefore, a new index, named the Coefficient of Variation and Maximum of Manhattan Distance (CV-MMD), is proposed to evaluate the performance of the candidate methods. It can be computed as follows:

$$CV - MMD = CV \cdot MMD, \quad (6)$$

$$CV = \frac{\sigma(n)}{\bar{n}}, \quad (7)$$

$$MMD = \max_{k \in N} \left[\frac{1}{n_k} \sum_{i=1}^{n_k} d_{Md}(x_{k,i}, c_k) \right], \quad (8)$$

where $s(n)$ = standard deviation of the number of samples in each cluster, \bar{n} = average of the number of samples in each cluster, n_k = number of samples in the k th cluster, c_k = center of the k th cluster, $x_{k,i}$ = the i th sample in the k th cluster, and $d_{Md}(x, y)$ = Manhattan distance between x and y .

The CV-MMD index consists of two parts. CV calculates the coefficient of variation of the sample size to measure the difference in sample size between clusters, whereas MMD calculates the maximum average Manhattan distance of all clusters to reflect the compactness of clustering results. The reason for using the Manhattan distance instead of the more commonly used Euclidean distance is that the latter is easier to fail in multi-dimensional space. Note that a lower value means better performance for both CV and MMD.

The truss sample population is divided into nine groups by each candidate method. Table 3. summarizes the results evaluated by the CV-MMD index.

Table 3. Evaluation results based on CV-MMD index.

Index	CLIQUE	CURE	DBSCAN	K-Means	K-Medoids	SOM
CV-MMD	4.1216	1.1881	4.6888	0.2952	0.3092	0.2675
CV	2.3049	1.3264	2.6536	0.3264	0.3080	0.2922
MMD	1.7882	1.5759	1.7670	0.9043	1.0040	0.9153

The SOM method achieves the minimum value of the CV-MMD index, which indicates the best numerical performance. The K-means and K-medoids methods also have relatively good performance in the CV-MMD index. Regarding the performance of sub-indexes, the SOM method achieves the minimum CV value and performs best in Uniformity, whereas the K-means method performs best in Compactness with a minimum MMD value. In contrast, the CLIQUE and DBSCAN methods get large CV values, which implies that the clustering results are unreasonable. This is most likely relates to the algorithm principles. Both CLIQUE and DBSCAN methods are density-based. In these methods, samples in two adjacent regions are gathered to the same cluster if t numbers of samples in the two regions meet the pre-set quantity requirement. This can easily lead to the generation of some small clusters and super clusters. Therefore, these methods' performance in both Compactness and Uniformity is poor.

Considering the results of both the two evaluation measures, the SOM method has an outstanding performance. Hence, it is selected as the effective clustering method for the subsequent sections.

3.4. Collection of User Preference Information

To collect the user preference information effectively and reduce the interaction complexity, a user preference interaction system written in the C# programming language is designed (Figure 5).

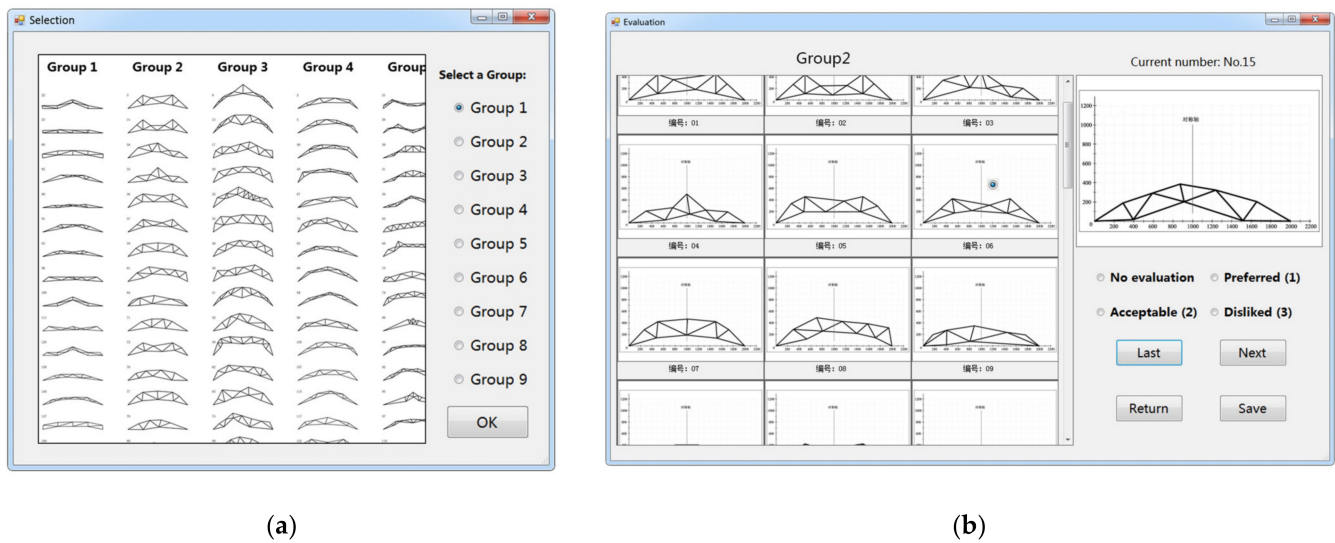


Figure 5. Screenshots of the interaction system: (a) Selection interface; (b) Evaluation interface.

Corresponding to the principle of “choose first, evaluate later”, the main user interfaces of the system includes both the Selection interface and the Evaluation interface. In the Selection interface (Figure 5a), users can preview the clustering results of the SOM method, where the image can be zoomed or dragged. After that, users need to select a preferred group. Then, it is moved to the Evaluation interface (Figure 5b), in which users need to evaluate every sample in the selected group. Three simple evaluation options, i.e., “Preferred”, “Acceptable” and “Disliked”, which correspond to 1, 2, and 3 points, are set. Users can choose an appropriate option for the current sample. For reference, all the truss samples in the group are presented in the left window. To ensure the effectiveness of the evaluation information, the current sample is presented in a random order, and some samples will be presented twice. The two evaluation results for the repeated sample will be checked for consistency to judge the validity of the evaluation information.

When the interaction process ends, a training group with labels is automatically output. Furthermore, the proportion of the importance of each truss feature is calculated by the permutation importance method [44]. It is also output to analyze the user preference information.

Five students (numbered from 1 to 5) were invited to participate in the interaction process, which was divided into two phases. In the first phase, each student selected a preferred group and evaluated the samples via the interactive system. Then, another 30 truss samples, similar to the samples in the preferred group, were generated randomly and evaluated by the same student to form a validation group. Furthermore, the range of their feature values needed to be consistent with the ones in the preferred group, which is called Permissible Feature Range (PFR). In the second phase, each student needed to evaluate the remaining eight groups and their pertinent validation groups. Finally, a total of 90 training and validation groups were obtained.

3.5. Selection of Machine Learning (ML) Techniques

To develop the prediction models of preference, two classical ML techniques, i.e., back propagation neural network (BPNN) and random forest (RF), are used. They are briefly explained in the following.

BPNN is a common type of artificial neural network. The basic network structure of BPNN includes an input layer, one or more hidden layers, and an output layer. Each layer contains several neurons that connect to neurons in the adjacent layers with connection weights. During the training phase, the input data is presented to the input neurons. The signals received by each neuron are processed through a weighted sum and converted as an output by an activation function. Then, the output is transmitted to the neurons

in the adjacent layer. This process continues until the outputs are produced through the output neurons [45]. BPNN uses a function to gradually reduce the error between the actual outputs and the target outputs (labels in this study) by adjusting the connection weights.

RF [46] is an ensemble learning method composed of a number of decision trees. Each tree is constructed based on a subset of features that are randomly selected and a data subset that is randomly selected from the training data set. The output of RF is the majority or the average of the results of all decision trees. RF shows effective and robust performance since it can overcome the overfitting problem of decision trees.

Both BPNN and RF use the 45 training groups to generate the prediction models. In order to compare the performance between the two methods, two numerical experiments are conducted. In the first one, the reappearance ability of the methods is evaluated. The training groups are employed as input again for the corresponding prediction models to get the outputs, which are continuous values ranging from 1 to 3. On the one hand, the mean square error (MSE) between the predicted and targeted outputs is calculated (Equation (9)). On the other hand, the outputs are rounded to integers and then compared with the labels to obtain the reappearance accuracy (RA, Equation (10)).

$$MSE = \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{N}, \quad (9)$$

$$RA = \frac{n_c}{N} \times 100\%, \quad (10)$$

where N = number of samples, y_i = targeted value of the i th sample, \hat{y}_i = predicted value of the i th sample, and n_c = number of correct predictions.

In the second experiment, the validation groups are used as input to evaluate the prediction ability of the methods. Besides the MSE index, prediction accuracy (PA), which has a similar definition to RA, and preference accuracy (PreA, Equation (11)) are also calculated. PreA focuses on the model's ability to correctly identify the "Preferred" and "Acceptable" samples for users, which is conducive to generating satisfactory structures.

$$PreA = \frac{n_{pc}}{n_p} \times 100\%, \quad (11)$$

where n_p = number of samples with "Preferred" or "Acceptable" labels, and n_{pc} = number of samples that are correctly predicted.

Based on the evaluation information provided by the students, the results of the indexes are summarized for each experiment. Taking the No.2 student as an example, Figure 6 illustrates the index values calculated with the nine training groups in the first experiment, whereas Figure 7 depicts the results based on the nine validation groups in the second experiment.

Figure 6 compares the reappearance ability of the two methods based on the No. 2 student's evaluation information. It is observed in Figure 6a that the PA values of the RF models are not less than those of the BPNN models in eight groups. The BPNN model achieves a higher PA value only in Group 5. In Figure 6b, obvious differences exist in the MSE values between the two methods for Group 2 and Group 5. This is consistent with the performance of the two methods indicated by the RA index. The MSE values of the BPNN models are generally greater than the ones of the RF models. Figure 7 displays the evaluation results of the prediction ability of the two methods. As seen in Figure 7a, RF models have better performance indicated by higher values of the PA index for seven groups. Similarly, the RF models are advantageous over the BPNN models due to the lower MSE values for all nine groups (Figure 7b). Regarding the PreA index, the RF models also obtain better results than the BPNN models. The RF models obtain a higher PreA value in one more group than the BPNN models (Figure 7c).

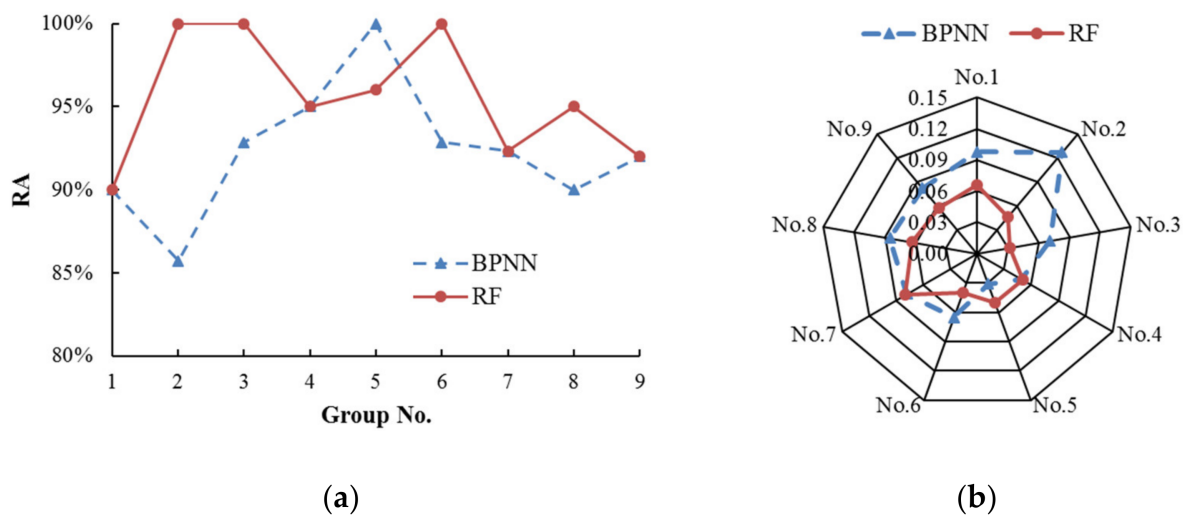


Figure 6. Index results of the No. 2 student in the first experiment: (a) RA; (b) MSE.

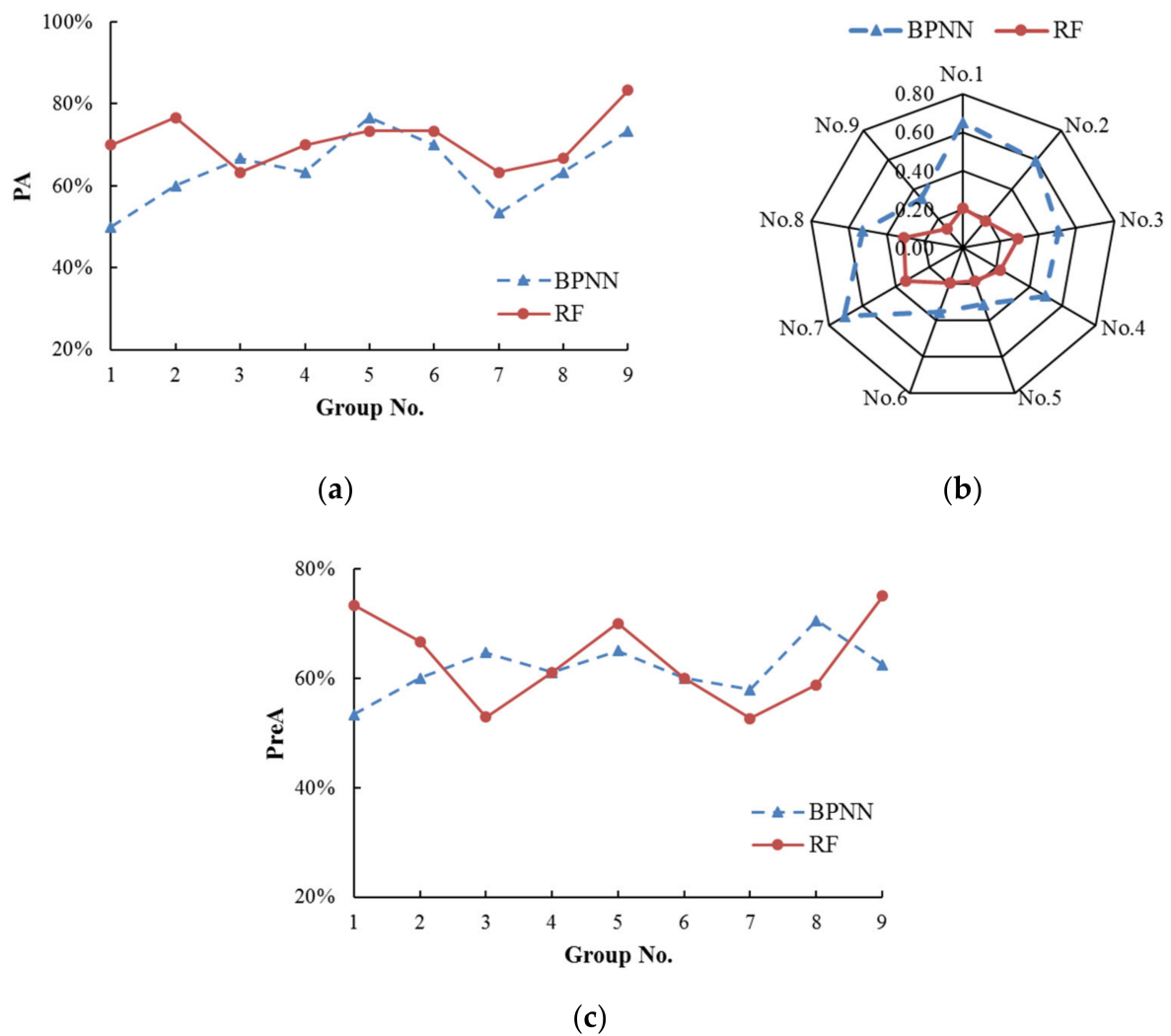


Figure 7. Index results of the No. 2 student in the second experiment: (a) PA; (b) MSE; (c) PreA.

To show the evaluation results comprehensively, the average index values of all nine groups are calculated for each student. The results of the two experiments are summarized in Tables 4 and 5. The RF models are overall advantageous over the BPNN models in all

indexes due to higher RA, PA, and PreA values and lower MSE values. This indicates better reappearance and prediction abilities. Besides, the parameters of the RF models are more general than those of the BPNN models. Nearly all the RF models use the same parameters, whereas the parameters have to be adjusted for each BPNN model. Therefore, RF is selected in this study.

Table 4. Evaluation results of BPNN and RF models in the first experiment.

Student No.	BPNN		RF	
	MSE	RA	MSE	RA
1	0.0698	93.84%	0.0571	96.48%
2	0.0807	92.30%	0.0540	95.59%
3	0.0796	94.67%	0.0434	98.10%
4	0.0774	93.07%	0.0477	97.11%
5	0.0691	93.75%	0.0466	98.05%

Table 5. Evaluation results of BPNN and RF models in the second experiment.

Student No.	BPNN			RF		
	MSE	PA	PreA	MSE	PA	PreA
1	0.3644	62.96%	58.61%	0.2141	74.44%	69.25%
2	0.4986	64.07%	61.68%	0.2303	71.11%	63.39%
3	0.4544	66.67%	59.95%	0.2586	71.48%	69.54%
4	0.4064	62.22%	60.38%	0.2640	71.85%	68.16%
5	0.4628	62.59%	60.82%	0.3366	69.26%	68.47%

In addition, a redundant feature elimination criterion is proposed to improve the accuracy of the prediction model. When a student finishes the evaluation in the first phase (Section 3.4), the proportion of the importance of each feature can be obtained. There would be some features with small proportions of importance, which affect the model performance. In order to explore this effect, the preferred group of each student obtained in the first phase is converted into three groups. The first group includes all nine features, whereas the second and third group eliminate the minimum proportion feature and the last two minimum proportion features. The three groups are used to train the prediction models, respectively. The corresponding validation group is used as input to calculate the PA values. Based on the results, a conservative criterion is obtained, namely, if the proportion of the minimum proportion feature is less than 1%, it can be identified as a redundant feature and eliminated. Using this conservative criterion, the redundant feature is automatically checked and eliminated when a user finishes the evaluation. After that, the prediction model of user preference can be developed.

4. Cost Function Based on Modified Physical Programming (MPP)

This section focuses on the combination of the prediction model of user preference and the STSA method. An aggregate function based on MPP is constructed to coordinate the user preference objective with other objectives, such as structural weight. On this basis, a cost function is further designed by taking the design constraints into account. Finally, an innovative design method named STSA-P is proposed.

4.1. Physical Programming (PP) Synopsis

PP is a user-friendly multi-objective optimization method that eliminates the process of assigning subjective weights. According to the PP procedure, the users' expectations for design objectives can be classified into four classes: smaller is better (Class 1), larger is better (Class 2), value is better (Class 3) and range is better (Class 4). Each class comprises two cases, hard (H) and soft (S), depending on the sharpness of the expectations [31]. In

the soft classes, users need to specify several ranges for each objective to show the degrees of satisfaction. A corresponding class function is built to map the objective function to a non-dimensional space, where all the objectives with different physical meanings and orders of magnitude have the same numerical scale. Figure 8. shows one possible class function of 1-S (soft case of Class 1).

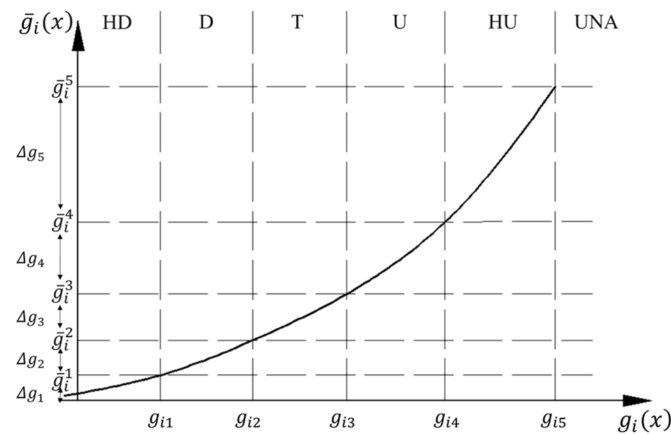


Figure 8. A class function of 1-S.

The objective function $g_i(x)$ is illustrated on the horizontal axis. The corresponding class function $\bar{g}_i(x)$ is shown on the vertical axis. Six ranges for the objective are given: highly desirable (HD), desirable (D), tolerable (T), undesirable (U), highly undesirable (HU), and unacceptable (UNA). The parameters g_{i1} to g_{i5} are the boundary values of the ranges, which are specified by users and directly reflect the users' expectations. To ensure the same scale for each class function, the parameters \bar{g}_i^1 to \bar{g}_i^5 , corresponding to g_{i1} to g_{i5} , should be the same for different class functions. These parameters can be calculated as:

$$\bar{g}_i^1 = \Delta g_1, \text{ take a small value} \quad (12)$$

$$\Delta g_k = \beta \cdot n_{sc} \cdot \Delta g_{k-1}, k = 2, 3, 4, 5 \quad (13)$$

$$\bar{g}_i^k = \bar{g}_i^{k-1} + \Delta g_k, k = 2, 3, 4, 5, \quad (14)$$

where n_{sc} = number of objective functions, and β = parameter of convexity, generally greater than 1.

The curve of the 1-S class function has two parts. The first part, in the HD range ($g_i(x) < g_{i1}$), is represented by a decaying exponential. The second part, in the D, T, U and HU ranges, takes the form of a spline segment. The class function should meet the requirements including continuous first derivative and strictly positive second derivative [30]. These settings are beneficial for gradient-based methods in searching for the optimal solution [47]. However, they also lead to complicated calculations of the class function.

The aggregate function $G(x)$ is formed by combing all the class functions (Equation (15)). Thus, a MOP can be transformed into an SOP.

$$\min G(x) = \lg\left(\frac{1}{n_{sc}} \sum_{i=1}^{n_{sc}} \bar{g}_i[g_i(x)]\right), \quad (15)$$

subject to

$$g_i(x) \leq g_{i5} \text{ (for } 1 - S). \quad (16)$$

4.2. Coordination of Multiple Objectives with MPP

The prediction model of user preference is integrated into the STSA method to guide the generation of a satisfactory structure. The STSA method combines structural grammar and performance evaluation with a stochastic optimization technique (simulated annealing,

SA) [1]. It can be used for topology generation and optimization of discrete structures. Structural grammar defines the basic units (e.g., triangles in 2D, tetrahedrons in 3D) and rules for the structural modification in sectional size, global shape, and topology, which can modify and transform the structures more flexibly. Similarly to human languages, the basic units can form a structure based on structural grammar. STSA improves the initial design continuously with structural grammar based on the feedback from performance evaluation until the final design meets the design goals of safety, efficiency, and economy. The workflow is shown in Figure 9 [1].

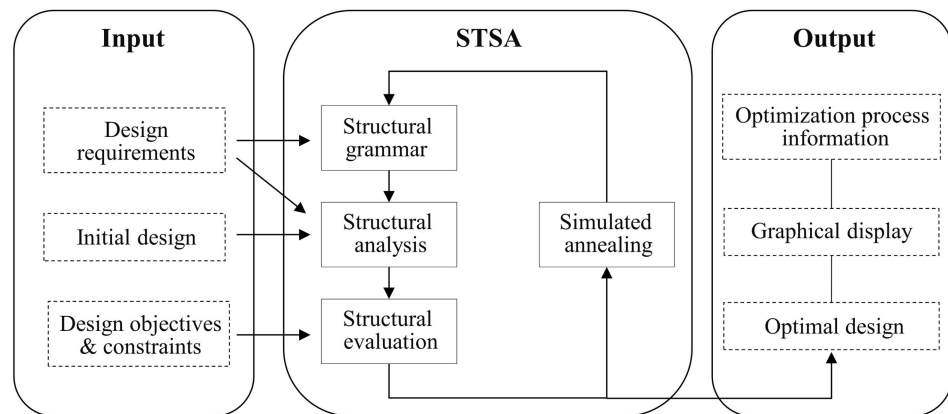


Figure 9. Flow chart of STSA.

Generally, many design factors, such as weight, displacement, stiffness, stress, frequency, etc., can be employed as the design objectives when STSA is applied to design structures. In this study, weight and displacement are selected as the first and second design objectives. Meanwhile, user preference is introduced as a third objective based on the prediction model. These objective functions are defined as follows:

$$g_1 = \min \sum_{i=1}^n \rho l_i s_i, \quad (17)$$

$$g_2 = \min \left[\max_{1 \leq j \leq m} (disp_j) \right], \quad (18)$$

$$g_3 = \min pref, \quad (19)$$

where ρ = material density, n = number of members, l_i = length of the i th member, s_i = sectional area of the i th member, m = number of nodes, $disp_j$ = displacement of the j th node, and $pref$ = preference prediction value of the structure.

Structural weight (Equation (17)) is a classical design objective in structural optimization. The smaller the weight, the better the structure. The objective of node displacement (Equation (18)) minimizes the maximum displacement at all the structural nodes. Similarly, the user preference objective (Equation (19)) minimizes the preference prediction value of the generated structures. A smaller value means a higher user satisfaction with the structural appearance. All three objectives conform to the feature of “smaller is better”, which can be represented by 1-S.

For the objectives of structural weight and node displacement, the class function of the PP method is redesigned and shown in Figure 10.

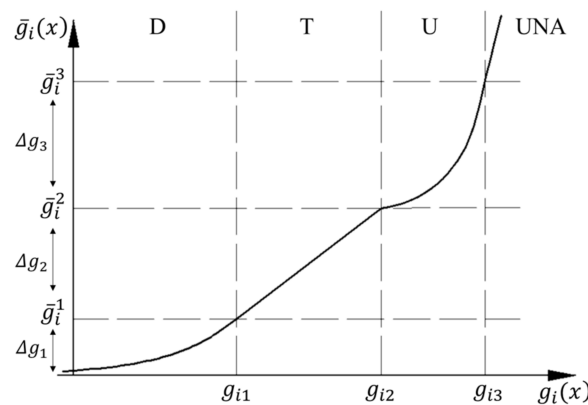


Figure 10. Class function for objectives of weight and displacement.

The number of ranges for each objective is simplified to four (D, T, U, UNA), in which D, T, and U are pertinent to the evaluation options “Preferred”, “Acceptable” and “Disliked” in the interactive system, respectively. The curve of the new class function expands to the UNA range, which is originally regarded as the “infeasible region” in the PP method. This is beneficial for the full exploration of the optimal structure, especially in the early optimization phase. The new class function contains three expressions. The first one is represented by a continuous quadratic function in the U and UNA ranges, where the objective value is expected to enter the U range from the UNA range quickly at a larger slope. Then the slope at the left boundary of the U range becomes smaller to reduce the impact of the current objective. It is conducive to the uniform optimization of all objectives. The second expression in the T range is set to be a linear function to keep a high slope throughout the range. This makes the objective values move quickly to the D range. The third expression in the D range is represented by an exponential function. The curves in the D and T ranges are continuous and have the same slope at the boundary. The new class function is defined as:

$$\bar{g}_i(x) = \begin{cases} \bar{g}_i^2 + \Delta g_3 \left[\frac{g_i(x) - g_{i2}}{g_{i3} - g_{i2}} \right]^2 & , g_i(x) > g_{i2} \\ \frac{\Delta g_2}{g_{i2} - g_{i1}} [g_i(x) - g_{i1}] + \bar{g}_i^1 & , g_{i1} \leq g_i(x) \leq g_{i2} \\ \bar{g}_i^1 \exp \left\{ \frac{\Delta g_2}{(g_{i2} - g_{i1}) \bar{g}_i^1} [g_i(x) - g_{i1}] \right\} & , g_i(x) < g_{i1} \end{cases} \quad (20)$$

The parameters g_{i1} to g_{i3} need to be specified by users, whereas the parameters on the vertical axis are calculated by Equations (12)–(14).

For the user preference objective, the expressions of class function are the same as those of the weight and displacement objectives in the D, T, and U ranges, whereas a special expression is used in the UNA range. The boundary values g_{i1} , g_{i2} and g_{i3} are determined as 1.5, 2.5, and 3, according to the evaluation options and the rounding rule. The prediction model runs only when all the feature values of the current structure are in the Permissible Feature Range (PFR, defined in Section 3.4) of the training data. Therefore, the class function in the UNA range is constructed based on the differences between the feature values and PFR:

$$\bar{g}_i(x) = \bar{g}_i^3 + \zeta + \sum_{j=1}^{N_f} v_j^2, \exists x^j \notin [x_{\min}^j, x_{\max}^j], \quad (21)$$

$$v_j = \begin{cases} \eta \frac{x^j - x_{\max}^j}{x_{\max}^j - x_{\min}^j}, & , x^j > x_{\max}^j \\ 0 & , x_{\min}^j \leq x^j \leq x_{\max}^j \\ \eta \frac{x_{\min}^j - x^j}{x_{\max}^j - x_{\min}^j}, & , x^j < x_{\min}^j \end{cases}, \quad (22)$$

where N_f = number of features without redundant features, ζ = a penalty constant, η = a penalty coefficient, x^j = value of the j th feature of the current structure, x_{\max}^j = maximum value of the j th feature of the samples in the training group, and x_{\min}^j = minimum value of the j th feature of the samples in the training group.

Equation (22) calculates the difference between the j th feature of the current structure and PFR. Then, the result is substituted into Equation (21) to calculate the value of the class function in the UNA range. The parameters ζ and η generally take large values to penalize the structure when feature values are outside the PFR. The class function of the user preference objective is defined as:

$$\bar{g}_i(x) = \begin{cases} \bar{g}_i^3 + \zeta + \sum_{j=1}^{N_f} v_j^2 & , \exists x^j \notin [x_{\min}^j, x_{\max}^j] \\ \bar{g}_i^2 + \Delta g_3 \left[\frac{g_i(x) - g_{i2}}{g_{i3} - g_{i2}} \right]^2 & , g_{i2} < g_i(x) \leq g_{i3} \\ \frac{\Delta g_2}{g_{i2} - g_{i1}} [g_i(x) - g_{i1}] + \bar{g}_i^1 & , g_{i1} \leq g_i(x) \leq g_{i2} \\ \bar{g}_i^1 \exp \left\{ \frac{\Delta g_2}{(g_{i2} - g_{i1}) \bar{g}_i^1} [g_i(x) - g_{i1}] \right\} & , g_i(x) < g_{i1} \end{cases} \quad (23)$$

A simplified form is adopted for the aggregate function:

$$\min G(x) = \sum_{i=1}^{n_{sc}} \bar{g}_i [g_i(x)], \quad (24)$$

subject to

$$r_{nr}(x) \leq 0, \quad (25)$$

where nr = number of constraint functions, and $r_{nr}(x)$ = the nr th inequality constraint function.

During the original design process of the PP method, if the final design result or the performance of an objective fails to meet the user's needs, the user can improve them by adjusting or resetting the boundary values. However, this manner has some drawbacks. Firstly, the adjusted boundary values may be against the original expectations of the user. Secondly, the boundary values for some objectives are unchangeable. For example, the boundary values for the user preference objective are fixed according to the rounding rule and the evaluation options in the interactive system. To solve this problem, an adjustment coefficient α is introduced to avoid the modification of boundary values. Then, the modified aggregate function $G(x)$ is given as:

$$\min G(x) = \sum_{i=1}^{n_{sc}} \alpha_i \bar{g}_i [g_i(x)], \quad (26)$$

subject to

$$r_{nr}(x) \leq 0. \quad (27)$$

Each objective corresponds to one adjustment coefficient. The default value is 1. A user can improve the performance of an objective by increasing the pertinent coefficient, which is equivalent to increasing the weight of the objective. To verify the effectiveness of the coefficient, a simple numerical example is presented:

$$\min g(x) = [g_1(x), g_2(x), g_3(x)], \quad (28)$$

subject to

$$x_1 + x_2 + x_3 \leq 8, 0 \leq x_i \leq 5, i = 1, 2, 3, \quad (29)$$

where

$$g_1(x) = 2x_1^3 - 2x_2^2 - x_3^2 + 1, \quad (30)$$

$$g_2(x) = 3x_2^3 - x_1^2 - 2x_3^2 - 2, \quad (31)$$

$$g_3(x) = 4x_3^3 - 3x_1^2 - x_2^2 - 4, \quad (32)$$

The class function for the three objectives can be calculated by Equation (20). The boundary values are given in Table 6. The SA method is used to solve the optimization problem. The results are shown in Equations (33) and (34).

$$(x_1, x_2, x_3) = (0.9921, 0.4639, 1.0061), \quad (33)$$

$$g(x) = (1.5103, -4.7093, -3.0941). \quad (34)$$

Table 6. Boundary values of objectives.

Objective Function	Type	Boundary Values						
		D ←	g_{i1}	← T →	g_{i2}	← U →	g_{i3}	→ UNA
$g_1(x)$	1-S		−5			0		5
$g_2(x)$	1-S		−4			1		6
$g_3(x)$	1-S		−7			4		10

According to Equation (34) and Table 6 the three objective values lie in the U, D, and T ranges, respectively. The result of the first objective $g_1(x)$ in the U range is not satisfactory. Therefore, the user can increase the adjustment coefficient of $g_1(x)$. Then, the new results are shown in Equations (35) and (36).

$$(x_1, x_2, x_3) = (0.6893, 0.6460, 1.1625), \quad (35)$$

$$g(x) = (-0.5310, -4.3690, -0.4408). \quad (36)$$

It is observed that the three objective values lie in the T, D, and T ranges, respectively. Compared to the initial results, the first objective is significantly improved, whereas the other two objective values still lie in good ranges. Thus, the adjustment coefficient can improve the result effectively without changing the boundary values.

4.3. Treatment of Design Constraints

Many design constraints, e.g., stress constraints and geometric constraints, need to be considered in the design process of truss structures. Generally, these design constraints are regarded as hard constraints, i.e., they cannot be violated. This study tends to handle the constraints as soft constraints, which means that they can be violated during the optimization process. Meanwhile, the final result needs to satisfy the constraints. This is beneficial to the full exploration of the optimal result.

The penalty parameter r of constraint violation (Equation (37)) is set to be associated with the iteration steps. In the early stage of optimization, the penalty for the constraint violation is small, i.e., the value of r is small. As the optimization progresses, the value of r gradually increases, which indicates a larger penalty for the constraint violation. At the late stage of optimization, the penalty is sufficiently large to make the final result satisfy the constraints.

$$r = T^2, \quad (37)$$

where T = number of the current step of STSA during the iteration.

For the i th constraint, the constraint violation $v_i(x)$ is defined as the proportion of the current constraint value exceeding the constraint limit:

$$v_i(x) = \begin{cases} \sum_k \max(\frac{val_k}{val_{limit}} - 1, 0), & \text{if } val_{limit} \text{ is the upper limit value} \\ \sum_k \max(1 - \frac{val_k}{val_{limit}}, 0), & \text{if } val_{limit} \text{ is the lower limit value} \end{cases}, \quad (38)$$

where val_k = value of constraint function of the k th unit, for example, the tensile stress of the k th member, val_{limit} = value of constraint limit.

Thus, the penalty term $V(x)$ of constraint violation is defined by Equation (39):

$$V(x) = 1 + r \sum_{i=1}^{nr} v_i(x), \quad (39)$$

$$\text{mincost} = V(x)G(x) = \left[1 + r \sum_{i=1}^{nr} v_i(x) \right] \cdot \sum_{j=1}^{n_{sc}} \alpha_j \bar{g}_j[g_j(x)]. \quad (40)$$

The new cost function (Equation (40)) is obtained by multiplying $V(x)$ and $G(x)$, which describes the quality of a new design during the optimization process. It combines the prediction model with the STSA method. On this basis, a new automated design method named STSA-P is established for the plane trusses, which applies user preference information. The design process using STSA-P is shown in Figure 11.

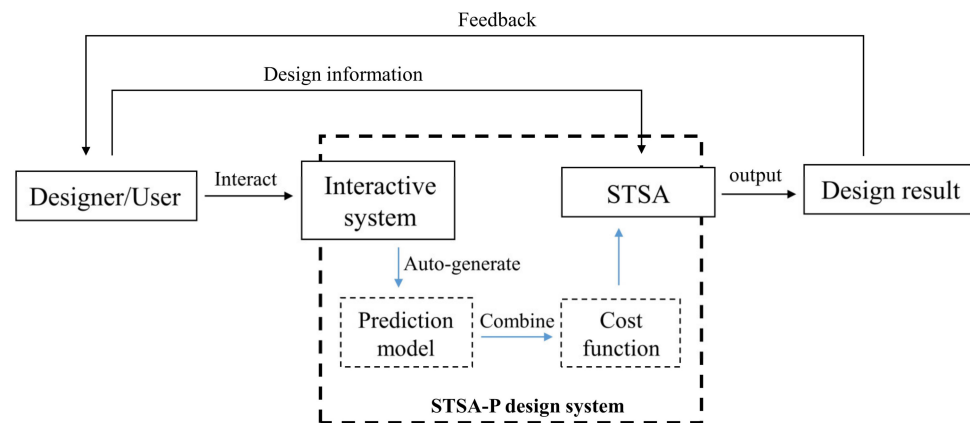


Figure 11. Design process based on STSA-P.

The combination of the interactive system and the STSA algorithm forms the STSA-P design system. Designing a truss structure with this system, designers or users can preset some design information, such as boundary values, and they need to complete the evaluation using the interactive system. Then, the prediction model of user preference is automatically developed and combined with the cost function. After that, the optimization process starts. Finally, the design result is obtained. If the result fails to meet the user's needs, the designer or user can improve the result by modifying the adjustment coefficient until a satisfactory result is achieved.

5. Design Example

To verify the practicality of the STSA-P method, a design example of a plane truss structure is presented. Five students were invited to participate in the design process. In fact, the design results generated by the STSA-P method include only two conditions, no matter how many students were surveyed. In the former, students obtain a satisfactory result without any adjustment. In the latter, students are not satisfied with the initial result and they modify the adjustment coefficient α (proposed in Section 4.2) to get the satisfactory result. Therefore, only two students' results representing the two conditions are presented, for brevity.

5.1. Case Information

Figure 12 shows the initial steel truss structure with a span of 20 m. Both the concentrated and distributed loads are applied to the top chord. Note that the loading points can move freely during the optimization process. The design objectives include minimizing

the structural weight, node displacement, and user preference. The detailed information is summarized in Table 7.

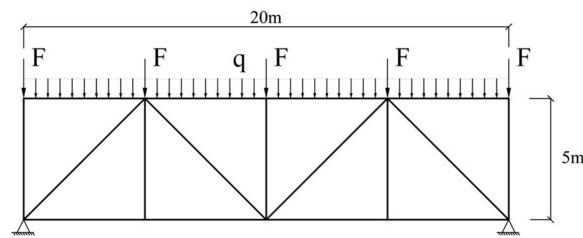


Figure 12. Initial layout of the truss structure.

Table 7. Design parameters.

Type	Name	Value	Name	Value
Material and geometrical properties	Material density	7.85 g/cm ³	Elasticity modulus	210 Gpa
	Cross section type	Solid cylinder	Initial cross-sectional area	100 cm ²
SA parameters	Initial temperature	2000	Final temperature	0.0001
	Number of iterations	170	Number of designs per iteration	150
Loading parameters	Concentrated load	150 kN	Distributed load	100 N/cm
	Self-weight load	Yes	Loading type	Dead load
Design constraints	Minimum member length	100 cm	Maximum member length	15 m
	Minimum area	10 cm ²	Maximum area	1084 cm ²
	Maximum connections per node	20	Minimum angle	20°
	Maximum compressive stress	31 kN/cm ²	Maximum tension stress	31 kN/cm ²
	Maximum height	20 m	Maximum number of members	60
	Intersections between members	No	Movable loading node	Yes

The two students completed the interactive evaluation independently. The user preference information was analyzed and no redundant feature was found based on the evaluation results of both students. The boundary values of the four ranges (D, T, U, UNA) of each objective are summarized in Table 8.

Table 8. Boundary values of objective functions.

Objective Function	Type	Boundary Values						
		D ←	g_{i1}	← T →	g_{i2}	← U →	g_{i3}	→ UNA
Weight	1-S		1500 kg		2100 kg		2500 kg	
Displacement	1-S		2 cm		5 cm		8 cm	
User Preference	1-S		1.5		2.5		3	

5.2. Design Results

Table 9 shows the proportion of importance of the truss features based on the evaluation results of the two students. Both students pay more attention to the symmetry of the truss structure, although they selected different groups to evaluate. MCH obtains the minimum proportion of 2.14% according to the first student's preference, whereas NM obtains the minimum proportion of 4.83% for the second student. Since both the two proportions are more than 1%, there is no redundant feature.

Table 9. Proportion of importance of the truss features.

Feature	First Student	Second Student
HSR	5.75%	8.90%
MCH	2.14%	8.50%
NM	2.89%	4.83%
AL	21.81%	15.25%
TD	16.30%	6.68%
DN	4.92%	12.28%
THD	4.71%	9.32%
BHD	2.78%	7.22%
SyR	38.71%	27.02%

For the first student, the design results are summarized in Table 10. The final truss structure is shown in Figure 13.

Table 10. Objective information for the first student.

Objective	Value	Range
Weight	1817.365 kg	T
Displacement	1.385 cm	D
Preference	1.798	T

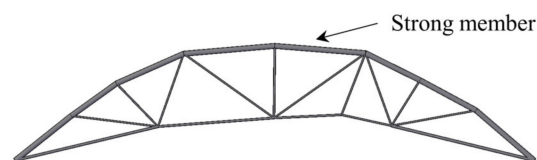


Figure 13. Truss structure for the first student.

It is seen that all the objective values are within the proper ranges, in which the displacement objective values lie in the D range. The truss structure is arched with strong top chord members, which conforms to the mechanical requirements. In addition, the whole structure has a certain degree of symmetry. In general, the student is satisfied with the design of this truss structure.

The results of the second student are shown in Table 11 and Figure 14. Note that an asymmetric optimization result can be generated (e.g., in Figure 14a) since the loading point of the initial structure can move freely and become asymmetric during the optimization process.

Table 11. Objective information for the second student.

Objective	Initial Result		Improved Result	
	Value	Range	Value	Range
Weight	1842.368 kg	T	1896.065 kg	T
Displacement	2.614 cm	T	4.259 cm	T
Preference	2.396	T	1.335	D

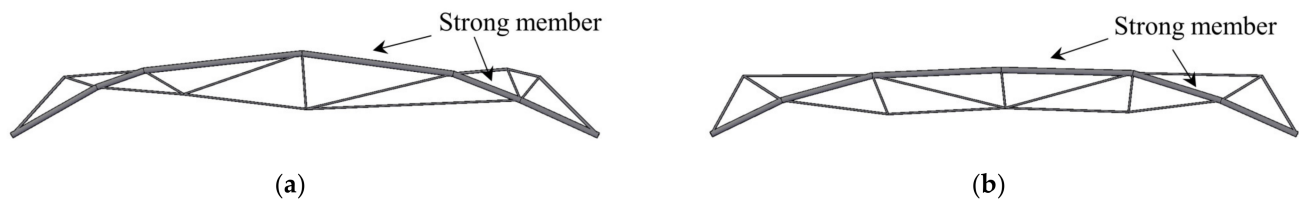


Figure 14. Truss structures for the second student: (a) initial result; (b) improved result.

All the objective values of the initial results are in the T range. However, the symmetry of the truss structure is poor, as shown in Figure 14a, and the student is not satisfied with the structure, especially the structural shape. Thus, the adjustment coefficient of the user preference objective is changed to improve the results. The new result achieves a great improvement in the user preference objective. The value of the objective enters into the D range from the T range, whereas the values of the other two objectives become greater, but still in the T range. As depicted in Figure 14b, the symmetry of the new structure is better than the initial one. In addition, the strong members also form an arch. Finally, the student is satisfied with the new truss structure.

To measure the impact of user preference on the optimization results, a simple index named Influence Degree (I) is proposed. It is calculated based on the change in the structural weight with and without the consideration of the user preference under the same displacement limit:

$$I = \frac{m_p - m_o}{m_o} \times 100\%, \quad (41)$$

where m_p = structural weight with the consideration of user preference, m_o = structural weight without the consideration of user preference.

The index I reflects the cost of considering the user preference. Taking the two design results mentioned above as examples, the index I is summarized in Table 12.

Table 12. Index I of the results for both students.

Student No.	Disp Limit	m_p	m_o	I
1	4.259 cm	1896.065 kg	1206.802 kg	57.11%
2	1.385 cm	1817.365 kg	1577.735 kg	15.19%

Regarding the result of the first student, the weight of the structure considering the user preference is 57.11% more than the one without considering the user preference. This indicates a large increase in construction costs. By contrast, the value is 15.19% based on the results of the second student. Index I can help the users weigh the cost and decide whether to accept the design result.

6. Conclusions

This paper proposes a new design method named STSA-P for plane trusses which incorporates user preference into the structural design process. The design method quantifies the preference information by developing a prediction model which transforms the information into an additional design objective. It realizes the coordination of multiple qualitative and quantitative objectives, allowing user preference to guide the search process to explore various designs. The user only needs to set boundary values and perform a short interaction via the interactive system. Then, the final design is generated automatically. Due to the short interaction and the reusable preference information, this method effectively reduces user fatigue and dependence on users. A design example is illustrated to confirm the effectiveness of the proposed method.

Specifically, the construction of the prediction model is split into two steps, i.e., generating a valid data set and selecting an appropriate machine learning (ML) technique. A series of subjective and numerical experiments are designed and carried out to ensure

the rationality of the steps and the accuracy of the prediction model. To coordinate user preference with other objectives of structural performance, an aggregate objective function is proposed based on the Modified Physical Programming (MPP) method. In addition, the cost function is further designed by considering design constraints, which effectively incorporates the prediction model into the STSA method.

Besides the user preference objective, only two structural performance objectives, i.e., structural weight and displacement, are considered in the STSA-P method. In order to extend the functionality of the method, further studies can be carried out by incorporating more design objectives, such as structural stiffness and natural frequency, into the STSA-P method.

Author Contributions: Conceptualization, T.Z. and X.Z.; methodology, T.Z. and W.X.; software, T.Z.; validation, T.Z., X.Z. and W.X.; formal analysis, T.Z.; investigation, T.Z.; resources, X.Z.; data curation, T.Z.; writing—original draft preparation, T.Z.; writing—review and editing, W.X. and X.Z.; visualization, T.Z. and W.X.; supervision, W.X. and X.Z.; project administration, X.Z.; funding acquisition, X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Shanghai Qi Zhi Institute, grant number SYXF0120020110.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Shea, K.; Smith, I.; Asce, F. Improving Full-Scale Transmission Tower Design through Topology and Shape Optimization. *J. Struct. Eng.* **2006**, *132*, 781–790. [\[CrossRef\]](#)
2. Shea, K.; Zhao, X. A novel noon mark cantilever support: From design generation to realization. In Proceedings of the IASS 2004: Shell and Spatial Structures from Models to Realization, Montpellier, France, 20–24 September 2004.
3. Zhao, X.; Shea, K. Intelligent generation and design of spatial truss structures. *J. Build. Struct.* **2010**, *31*, 63–69.
4. Kelly, J.C.; Maheut, P.; Petiot, J.; Papalambros, P.Y. Incorporating user shape preference in engineering design optimisation. *J. Eng. Des.* **2011**, *22*, 627–650. [\[CrossRef\]](#)
5. Nagamachi, M. Kansei engineering approach to automotive. *J. Soc. Automot. Eng. Jpn.* **1989**, *43*, 94–100.
6. Nagamachi, M. An image technology expert system and its application to design consultation. *Int. J. Hum.-Comput. Interact.* **1991**, *3*, 267–279. [\[CrossRef\]](#)
7. Takagi, H. Interactive evolutionary computation: Fusion of the capabilities of EC optimization and human evaluation. *Proc. IEEE* **2001**, *89*, 1275–1296. [\[CrossRef\]](#)
8. Von Buelow, P. Suitability of genetic based exploration in the creative design process. *Digit. Creat.* **2008**, *19*, 51–61. [\[CrossRef\]](#)
9. Turrin, M.; Von Buelow, P.; Stouffs, R. Design explorations of performance driven geometry in architectural design using parametric modeling and genetic algorithms. *Adv. Eng. Inform.* **2011**, *25*, 656–675. [\[CrossRef\]](#)
10. Mueller, C.; Ochsendorf, J. An Interactive Evolutionary Framework for Structural Design. In Proceedings of the 7th International Seminar of the IASS Structural Morphology Group, London, UK, 17–18 September 2011.
11. Mueller, C.T.; Ochsendorf, J.A. Combining structural performance and designer preferences in evolutionary design space exploration. *Autom. Constr.* **2015**, *52*, 70–82. [\[CrossRef\]](#)
12. Felkner, J.; Chatzi, E.; Kotnik, T. Interactive particle swarm optimization for the architectural design of truss structures. In Proceedings of the 2013 IEEE Symposium on Computational Intelligence for Engineering Solutions (CIES), Singapore, 16–19 April 2013; IEEE: Piscataway, NJ, USA, 2013.
13. Felkner, J.; Chatzi, E.; Kotnik, T. Architectural feedback in the structural optimization process. In *Structures and Architecture: Concepts, Applications and Challenges*; CRC Press: London, UK, 2013.
14. Felkner, J.; Chatzi, E.; Kotnik, T. Interactive truss design using Particle Swarm Optimization and NURBS curves. *J. Build. Eng.* **2015**, *4*, 60–74. [\[CrossRef\]](#)
15. Furuta, H.; Hase, H.; Watanabe, E.; Tonegawa, T.; Morimoto, H. Applications of genetic algorithm to aesthetic design of dam structures. *Adv. Eng. Softw.* **1996**, *25*, 185–195. [\[CrossRef\]](#)

16. Chikata, Y.; Yasuda, N.; Matsushima, M.; Kobori, T. Inverse analysis of aesthetic evaluation of planted concrete structures by neural networks. *Comput. -Aided Civ. Infrastruct. Eng.* **1998**, *13*, 255–264. [\[CrossRef\]](#)
17. Bailey, B.; Raich, A.M. Modeling of user design preferences in multiobjective optimization of roof trusses. *J. Comput. Civil. Eng.* **2012**, *26*, 584–596. [\[CrossRef\]](#)
18. Machwe, A.T.; Parmee, I.C. Reducing user fatigue within an interactive evolutionary design system using clustering and case-based reasoning. *Eng. Optimiz.* **2009**, *41*, 871–887. [\[CrossRef\]](#)
19. Jaimes, A.L.; Martinez, S.Z.; Coello, C.A.C. An introduction to multiobjective optimization techniques. In *Optimization in Polymer Processing*; Nova Science Publishers: Hauppauge, NY, USA, 2009; pp. 29–57.
20. Marler, R.T.; Arora, J.S. Survey of multi-objective optimization methods for engineering. *Struct. Multidiscip. Optim.* **2004**, *26*, 369–395. [\[CrossRef\]](#)
21. Deb, K. Multi-objective optimization. In *Search Methodologies*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 403–449.
22. Gunantara, N. A review of multi-objective optimization: Methods and its applications. *Cogent Eng.* **2018**, *5*, 1502242. [\[CrossRef\]](#)
23. Taylor, S. *Multicriterion Optimization in Engineering*; Andrzej Osyczka; John Wiley & Sons: New York, NY, USA, 1984; Pergamon: Bergama, Turkey, 1985.
24. Das, I.; Dennis, J.E. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM J. Optim.* **1998**, *8*, 631–657. [\[CrossRef\]](#)
25. Messac, A.; Ismail-Yahaya, A.; Mattson, C.A. The normalized normal constraint method for generating the Pareto frontier. *Struct. Multidiscip. Optim.* **2003**, *25*, 86–98. [\[CrossRef\]](#)
26. Fonseca, C.M.; Fleming, P.J. Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization. In Proceedings of the ICGA-93: Fifth International Conference on Genetic Algorithms, San Mateo, CA, USA, 17–22 July 1993.
27. Zitzler, E.; Thiele, L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Trans. Evol. Comput.* **1999**, *3*, 257–271. [\[CrossRef\]](#)
28. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [\[CrossRef\]](#)
29. Das, I.; Dennis, J.E. A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. *Struct. Optim.* **1997**, *14*, 63–69. [\[CrossRef\]](#)
30. Messac, A. Physical Programming: Effective Optimization for Computational Design. *AIAA J.* **1996**, *34*, 149–158. [\[CrossRef\]](#)
31. Messac, A.; Sukam, C.P.; Melachrinoudis, E. Mathematical and pragmatic perspectives of physical programming. *AIAA J.* **2001**, *39*, 885–893. [\[CrossRef\]](#)
32. Messac, A.; Mattson, C.A. Generating well-distributed sets of Pareto points for engineering design using physical programming. *Optim. Eng.* **2002**, *3*, 431–450. [\[CrossRef\]](#)
33. Messac, A.; Van Dessel, S.; Mullur, A.A.; Maria, A. Optimization of large-scale rigidified inflatable structures for housing using physical programming. *Struct. Multidiscip. Optim.* **2004**, *26*, 139–151. [\[CrossRef\]](#)
34. Ilgin, M.A. Aircraft selection using linear physical programming. *J. Aeronaut. Space Technol.* **2019**, *12*, 121–128.
35. Macqueen, J.B. Some Methods for Classification and Analysis of Multivariate Observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Los Angeles, CA, USA, 21 June–18 July 1965.
36. Ohsaki, M.; Takagi, H. Improvement of presenting interface by predicting the evaluation order to reduce the burden of human interactive EC operators. In Proceedings of the SMC'98 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 98CH36218), San Diego, CA, USA, 14 October 1998; IEEE: Piscataway, NJ, USA, 1998.
37. Bailey, B.; Raich, A. Capturing user aesthetic design preferences during multi-objective roof truss optimization. In Proceedings of the Joint International Conference on Computing and Decision Making in Civil and Building Engineering (ICCCBE), Montreal, QC, Canada, 14–16 June 2006.
38. Ester, M.; Kriegel, H.; Sander, J.; Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, OR, USA, 2–4 August 1996.
39. Agrawal, R.; Gehrke, J.E.; Gunopulos, D.; Raghavan, P. Automatic Subspace Clustering of High Dimensional Data for data mining applications. *Data Min. Knowl. Discov.* **1998**, *27*, 94–105.
40. Lacko, D.; Huysmans, T.; Vleugels, J.; De Bruyne, G.; Van Hulle, M.M.; Sijbers, J.; Verwulgen, S. Product sizing with 3D anthropometry and k-medoids clustering. *Comput.-Aided Des.* **2017**, *91*, 60–74. [\[CrossRef\]](#)
41. Kohonen, T. The self-organizing map. *Proc. IEEE* **1990**, *78*, 1464–1480. [\[CrossRef\]](#)
42. Guha, S. CURE: An efficient clustering algorithm for large databases. In Proceedings of the SIGMOD'98, Seattle, WA, USA, 2–4 June 1998.
43. Liu, Y.; Li, Z.; Xiong, H.; Gao, X.; Wu, J. Understanding of internal clustering validation measures. In Proceedings of the 2010 IEEE International Conference on Data Mining, Sydney, NSW, Australia, 13–17 December 2010; IEEE: Piscataway, NJ, USA, 2010.
44. Altmann, A.; Toloşi, L.; Sander, O.; Lengauer, T. Permutation importance: A corrected feature importance measure. *Bioinformatics* **2010**, *26*, 1340–1347. [\[CrossRef\]](#)
45. Suliman, A.; Zhang, Y. A review on back-propagation neural networks in the application of remote sensing image classification. *J. Earth Sci. Eng.* **2015**, *5*, 52–65.

46. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
47. Sanchis, J.; Martínez, M.A.; Ferragud, X.B.; Reynoso-Meza, G. Modeling preferences in multi-objective engineering design. *Eng. Appl. Artif. Intell.* **2010**, *23*, 1255–1264. [[CrossRef](#)]

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.