




## Article

# Combined Gaussian Local Search and Enhanced Comprehensive Learning PSO Algorithm for Size and Shape Optimization of Truss Structures

Thu Huynh Van <sup>1</sup>, Sawekchai Tangaramvong <sup>1,\*</sup>, Soviphou Muong <sup>1</sup> and Phuc Tran Van <sup>2</sup>

<sup>1</sup> Center of Excellence in Applied Mechanics and Structures, Department of Civil Engineering, Chulalongkorn University, Bangkok 10330, Thailand

<sup>2</sup> Faculty of Engineering, University of Architecture Ho Chi Minh City, 196 Pasteur Street, District 3, Ho Chi Minh 700000, Vietnam

\* Correspondence: sawekchai.t@chula.ac.th

**Abstract:** This paper proposes the use of enhanced comprehensive learning particle swarm optimization (ECLPSO), combined with a Gaussian local search (GLS) technique, for the simultaneous optimal size and shape design of truss structures under applied forces and design constraints. The ECLPSO approach presents two novel enhancing techniques, namely perturbation-based exploitation and adaptive learning probability, in addition to its distinctive diversity of particles. This prevents the premature convergence of local optimal solutions. In essence, the perturbation enables the robust exploitation in the updating velocity of particles, whilst the learning probabilities are dynamically adjusted by ranking information on the personal best particles. Based on the results given by ECLPSO, the GLS technique takes data from the global best particle and personal best particles in the last iteration to generate samples from a Gaussian distribution to improve convergence precision. A combination of these techniques results in the fast convergence and likelihood to obtain the optimal solution. Applications of the combined GLS-ECLPSO method are illustrated through several successfully solved truss examples in two- and three-dimensional spaces. The robustness and accuracy of the proposed scheme are illustrated through comparisons with available benchmarks processed by other meta-heuristic algorithms. All examples show simultaneous optimal size and shape distributions of truss structures complying with limit state design specifications.

**Keywords:** non-convex optimization; enhanced comprehensive learning; Gaussian local search; particle swarm optimization; perturbation-based exploitation; adaptive learning probability



**Citation:** Van, T.H.; Tangaramvong, S.; Muong, S.; Van, P.T. Combined Gaussian Local Search and Enhanced Comprehensive Learning PSO Algorithm for Size and Shape Optimization of Truss Structures. *Buildings* **2022**, *12*, 1976. <https://doi.org/10.3390/buildings12111976>

Academic Editor: Harry Far

Received: 25 September 2022

Accepted: 7 November 2022

Published: 14 November 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 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/>).

## 1. Introduction

Structural optimization uses computing techniques toward sustainability that achieves not only economical designs of structures and infrastructures (minimum resource consumption) but also ones with integrity (public safety and functionality). Over the past decades, this field has gained increasing interest and achievements in scientific research and engineering applications. Structural optimization has been successfully applied to the designs of many structures, including trusses, beams, plates, and shells. Among them, the optimal design of truss structures is generally divided into the three main problems, namely size, shape, and/or topology optimizations. Specifically, size optimization considers cross-sectional areas of truss members as variables, whilst shape optimization takes nodal coordinates as the design variables. Topology optimization determines whether to remove or maintain discrete elements (volume fraction control) within the domain. The simultaneous optimization of size and shape provides a more economical design than its individual (either size or layout) counterparts. The seminal work of Haftka and Grandhi [1], including the comprehensive reviews [2–5], laid the groundwork for methods

and applications in shape optimization. The generic formulation of these problems minimizes an objective function (e.g., total volume or weight presenting the cost of a structure), subjected to the constraints describing design, ultimate (permissible) strength, and/or serviceability criteria.

Mathematical programming-based approaches [6–8] develop the optimality criteria techniques for the solutions of general optimization problems that involve the calculation of implicit gradient functions. This poses difficulties, especially when considering challenging non-convex (and/or non-smooth) optimization problems. The presence of such conditions often leads to pitfalls such as the premature convergence of local optimal solutions, and the performance of standard solution techniques depends on specific parametric initialization.

With the bypassing of differential (gradient) operations in mathematics, meta-heuristic optimization methods, based on nature-inspired techniques, have been recently developed to approximate the solutions of optimization programs. For instance, Wu and Chow [9] utilized a genetic algorithm (GA) for the combined size and layout optimization of truss structures involving discrete size and continuous configuration variables. Soh and Yang [10] adopted the GA to perform simultaneous size and shape optimization of steel bridge trusses. Kaveh and Talatahari [11] carried out a layout optimization using an improved charged system search (CSS) algorithm. Miguel and Miguel [12] employed the two meta-heuristic harmony search (HS) and firefly algorithm (FA) methods to process the simultaneous size and geometry optimization of steel trusses under dynamic constraints. Ho-Huu et al. [13] proposed a new version of differential evolution (DE) for layout optimization of discrete size trusses under displacement and stress conditions. Azad et al. [14] employed a modified big bang-big crunch algorithm to simultaneously solve the size and shape optimization of truss structures under dynamic excitations. Ho-Huu et al. [15] developed a novel DE to process the size and shape optimization problems for truss structures with frequency constraints. Nguyen-Van et al. [16] hybridized the DE and symbiotic organisms search (SOS) methods to concurrently improve the solution for size and shape truss optimization under multiple frequency constraints. Inspired by the success of AlphaGo, Luo, et al. [17] applied a Markov decision process (MDP) model, and a two-stage Monte Carlo tree search (MCTS) for the optimal truss layout, simultaneously considering topology, geometry, and bar sizes.

Meta-heuristic algorithms generally consist of a series of trial-and-error processes to find the optimal solution of optimization problems within the constructed population. The word “meta” describes beyond or higher level [18,19]. The meta-heuristic algorithm is classified as a population-based (or trajectory-based) technique, containing two subpopulation exploitation and exploration phases [20,21]. The exploitation ability searches for solutions in a local area by using information of the good local solution. Meanwhile, the exploration constructs the search space on global positions to produce the global optimum. Reaching a good balance between the exploration and exploitation of sample positions is important for any meta-heuristic method, to escape the premature convergence of local optima, increasing the likelihood of accurate optima.

As described in no-free-lunch theorems [22], no meta-heuristic method ensures the optimality of solutions in the presence of non-convex and/or non-smooth conditions often encountered when solving practical-scale problems. Three underlying drawbacks are addressed. First, a standard meta-heuristic algorithm is often trapped into local optima leading to the premature convergence of inaccurate design solutions or even failure to attend. Second, the final design depends on the preset initial parameters, as well as some stochastic (random) values constructed during the optimization process, yielding unreliable (nonrepetitive) design solutions. Third, solving complex problems (in the presence of non-convex and/or non-smooth variable domains) requires a high number of particles and numerical simulations. The performance of algorithms is rather problem-dependent. An approach providing a good optimal design in one application does not necessarily yield an optimal solution in another.

Many meta-heuristic algorithms have been introduced with underlying exploitation and exploration abilities. The particle swarm optimization (PSO) [23], being a swarm-intelligence approach, emulates the movement or social behavior of a bird flock. The PSO constructs a set of particles in the population, where their positions are iteratively updated through the movement (velocity functions) learnt from the global best particle. However, the premature local optima are often encountered by the standard PSO method, as its social update components do not sufficiently work. Various new techniques have been incorporated with the original PSO to enhance its global search ability and overcome local optimal pitfalls.

Our recent work successfully applied a variant version of the PSO, called comprehensive learning particle swarm optimization (CLPSO), for the design of steel structures [24]. In the CLPSO [25], the learning technique enables cross-positions between the sets of best swarm particles in each dimensional space, leading to the likelihood of overcoming locally optimal searches and premature termination of undesired non-optimal but feasible solutions. The proposed scheme followed a learning probability function that defined the cooperative responses among swarm populations. However, this strategy did not improve the exploitation ability to perform deep local searches around the best global position.

To improve the searching mechanism and exploitation ability of standard CLPSO, this paper proposes an enhanced CLPSO (ECLPSO), combined with a Gaussian local search (GLS), termed the GLS-ECLPSO method to perform simultaneous size and shape optimization of structures under applied forces. The two enhancing techniques [26] underlying the ECLPSO, namely normative knowledge and new adaptive learning probability, were incorporated with the CLPSO. The former technique determines whether the algorithm needs to improve its exploitation ability using perturbation-based exploitation, whilst the latter replaces the original learning probability with new functions, letting individual particles learn from ranking information of the personal best positions in search spaces. The enhanced exploitation and exploration abilities underpinning the ECLPSO, therefore, obtain improved optimal solutions, even when considering challenging non-convex optimization problems.

In addition to the ECLPSO, the GLS applies a bell-curved shape of Gaussian distribution to exploit the particles in a central field, further improving solution optimality. The GLS adopts information from the global best particle and samples from personal best particles to better converge on an accurate optimal solution. Random particles are constructed around the global best position based on the Gaussian distribution function. We illustrate applications of the proposed GLS-ECLPSO method using a few optimal size and shape designs of two- and three-dimensional truss structures. The results show superior performance in capturing the accurate optimal design for truss structures with modest computing efforts compared with standard techniques.

## 2. Size and Shape Optimization Problem

This section describes the simultaneous size and shape optimization problem that minimizes the cost function, described by the total weight  $W(\mathbf{A}, \mathbf{x}, \mathbf{y}, \mathbf{z})$  of (planar or space) truss structures simultaneously subjected to ultimate strength and serviceability conditions. The structural optimization contains the size (cross-sectional area) variables  $\mathbf{A} \in \mathbb{R}^{ne} = [A_1, \dots, A_{ne}]$  of  $ne$  members (viz.,  $m \in \{1, \dots, ne\}$ ) and the shape variables, including the member lengths  $\mathbf{L} \in \mathbb{R}^{ne} = [L_1, \dots, L_{ne}]$ , written as functions of unknown nodal coordinates,  $(\mathbf{x}, \mathbf{y}, \mathbf{z}) \in \mathbb{R}^{nn \times nn \times nn} = [(x_1, y_1, z_1), \dots, (x_{nn}, y_{nn}, z_{nn})]$ , for  $n \in \{1, \dots, nn\}$ . The practical design classifies the  $ng$  independent groups of members, consisting of similar sizes using technological constraints [27,28], namely  $A_m = \sum_{g=1}^{ng} Q_{m,g} a_g$  for  $\forall m \in \{1, \dots, ne\}$ . The self-evident matrix  $\mathbf{Q} \in \mathbb{R}^{ne, ng}$  collects binary parameters indicating either the  $m$ -th member lying within (viz.,  $Q_{m,g} = 1$ ) or outside ( $Q_{m,g} = 0$ ) the  $g$ -th group.

The optimization formulation is therefore written in terms of the unknown (independent) design area variables  $\mathbf{a} \in \mathbb{R}^{ng} = [a_1, \dots, a_{ng}]$  and nodal coordinates  $(\mathbf{x}, \mathbf{y}, \mathbf{z}) \in \mathbb{R}^{nn \times nn \times nn}$ , as follows:

$$\begin{aligned} & \text{Minimize } W(\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{m=1}^{ne} \left( \sum_{g=1}^{ng} Q_{m,g} \cdot a_g \cdot \rho_g \right) \cdot L_m(\mathbf{x}, \mathbf{y}, \mathbf{z}) \\ & \text{Subject to } \begin{cases} g_j(\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z}) \leq 0 & \forall j \in \{1, \dots, nc\} \\ a_{g,\min} \leq a_g \leq a_{g,\max} & \forall g \in \{1, \dots, ng\} \end{cases} \end{aligned} \quad (1)$$

where  $L_m(\mathbf{x}, \mathbf{y}, \mathbf{z})$  is the  $m$ -th member length. For each  $g$ -th member group, the material properties define the density  $\rho_g$ , minimum available area size  $a_{g,\min}$ , and maximum area size  $a_{g,\max}$ . The design constraints,  $g_j(\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z}) \leq 0$  for  $\forall j \in \{1, \dots, nc\}$ , imply the permissible stress (e.g.,  $\sigma_m \leq \sigma_m^{\text{permissible}}$ ) and serviceability (limited displacement and/or natural frequency, viz.,  $\delta_n \leq \delta_n^{\text{limit}}$ ) criteria. The static (stress,  $\sigma_m$ ) and kinematic (serviceability,  $\delta_n$ ) responses are described as functions of design variables, namely cross-sectional areas ( $\mathbf{a}$ ) and nodal coordinates  $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ .

Whilst the pin-connected trusses form the focus of this study, applications on more general beam and frame structures can also be considered. In fact, the optimization problem in Equation (1) remains rigorous, where the constraints  $g_j(\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z}) \leq 0$  describe the permissible stresses and/or limited displacements at some specified locations. In contrast to pin-connected trusses, stresses are additionally presented by the generalized flexural forces at member ends. A similar set of design variables  $(\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z})$  is processed.

The formulation in Equation (1) belongs to a challenging class of nonlinear programming (NLP) problems in the presence of non-convex and non-smooth constraints [1]. This can be converted to an unconstrained nonlinear equation using the penalty function measuring the constraint violations by:

$$W' = W(\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z})(1 + C)^\epsilon \quad (2)$$

where  $C = \sum_{j=1}^{nc} \max(g_j(\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z}), 0)$  is a constraint penalty function and  $\epsilon$  a positive penalty scalar. The solution to the unconstrained nonlinear formulation in Equation (2) presents the optimal size and shape design of structures written in Equation (1). The ECLPSO method combined with the GLS strategy was developed to provide a solution to the problem in Equation (2).

### 3. Enhanced Comprehensive Learning Particle Swarm Optimization

#### 3.1. Comprehensive Learning PSO

Similar to standard PSO algorithms [23], the CLPSO method [25] randomly constructs the swarm population of  $np$  particles in a stochastic fashion, namely  $\mathbf{X}_p \in \mathbb{R}^{nd} = (X_{p,d} | \forall d \in \{1, \dots, nd\})$  for  $p \in \{1, \dots, np\}$  of the design variables  $\mathbf{X} \in \mathbb{R}^{nd}$  in  $nd$  dimensions. The iterative procedures are performed to update the new position  $\mathbf{X}_p^{\text{next}} \in \mathbb{R}^{nd}$  of the  $p$ -th generic particle by

$$\mathbf{X}_p^{\text{next}} = \mathbf{X}_p + \mathbf{V}_p^{\text{next}} \quad (3)$$

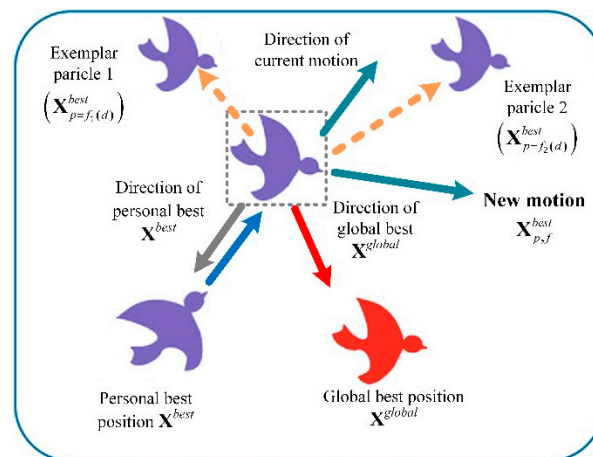
where the new velocity  $\mathbf{V}_p^{\text{next}} \in \mathbb{R}^{nd}$  defines changes in the particle position  $\mathbf{X}_p$  from the current to next step iteration. The velocity  $\mathbf{V}_p^{\text{next}}$  of the  $p$ -th particle satisfies the following random function:

$$\mathbf{V}_p^{\text{next}} = w_p \mathbf{V}_p + c_1 \cdot \text{rand}1_p \cdot (\mathbf{X}_p^{\text{best}} - \mathbf{X}_p) + c_2 \cdot \text{rand}2_p \cdot (\mathbf{X}^{\text{global}} - \mathbf{X}_p) \quad (4)$$

where the acceleration weight  $c_1$  attracts  $\mathbf{X}_p$  toward the best position  $\mathbf{X}_p^{\text{best}} \in \mathbb{R}^{nd} = (X_{p,d}^{\text{best}} | \forall d \in \{1, \dots, nd\})$  (viz., ones associated with the best objective value at its own  $p$ -th particle for all time steps);  $c_2$  is the acceleration weight associated with searches around the global best position  $\mathbf{X}^{\text{global}} \in \mathbb{R}^{nd} = (X_d^{\text{global}} | \forall d \in \{1, \dots, nd\})$ ;  $\text{rand}1_p$

and  $rand2_p$  are the random numbers uniformly selected within the interval of  $[0, 1]$ ; and  $w$  is the inertia weight controlling the excessive momentum in particles.

Comprehensive learning [25] determines the best position  $\mathbf{X}_p^{best}$  in Equation (3) through the new learning exemplar  $\mathbf{X}_{p,f}^{best} = (X_{p,f(d)}^{best} | \forall d \in \{1, \dots, nd\})$ . For the  $p$ -th particle, the learning exemplar  $\mathbf{X}_{p,f}^{best}$  is initialized at the best location  $\mathbf{X}_p^{best}$ . At the  $d$ -th dimension, the location of the best particle  $X_{p=f(d),d}^{best}$  is indicated by the particle index  $f(d)$  through the learning probability searches across all  $np$  particles. The exemplar  $\mathbf{X}_{p,f}^{best}$  then explores for each  $d$ -th dimension its new  $X_{p,f(d)}^{best}$  from one of the two best particles  $(X_{p,f_1(d)}^{best}, X_{p,f_2(d)}^{best})$ , where  $X_{p,f_1(d)}^{best} \neq X_{p,f_2(d)}^{best}$ . Each of  $X_{p,f_1(d)}^{best}$  and  $X_{p,f_2(d)}^{best}$  is randomly selected from the  $\mathbf{X}_p^{best}$ , namely  $(X_{p=1,d}^{best}, \dots, X_{p=np,d}^{best})$ . The index  $f(d)$  is set to either  $f_1(d)$  or  $f_2(d)$  associated with the more optimal objective function,  $\min(W(\mathbf{X}_{p=f_1(d)}^{best}), W(\mathbf{X}_{p=f_2(d)}^{best}))$ . The new exemplar at the  $d$ -th dimension is updated by  $X_{p,f(d)}^{best} = X_{p=f(d),d}^{best}$ . All  $nd$  dimensions are explored to yield the new direction  $\mathbf{X}_{p,f}^{best}$  of the  $p$ -th particle. A schematic expression of the comprehensive learning strategy constructing the updating position  $\mathbf{X}_{p,f}^{best}$  of the  $p$ -th particle is depicted in Figure 1.



**Figure 1.** Comprehensive learning for the updating position  $\mathbf{X}_{p,f}^{best}$ .

For each  $p$ -th particle, the comprehensive searches permit its best exemplars  $\mathbf{X}_p^{best}$  to learn from those of other particles in the same dimension. This feature enables the CLPSO to construct a diverse (better-quality) swarm and have global exploration ability of search spaces. The CLPSO assists particles in avoiding local optima pitfalls, and therefore, increases the likelihood of capturing accurate optimal design solutions.

In comprehensive learning, the exemplar at each dimension is randomly selected according to the learning probability function  $Pc_p$ , described in Equation (5) [25]. More explicitly, for each  $d$ -th dimension, the learning probability function  $Pc_p$  determines the new learning position of the exemplar  $X_{p,f(d)}^{best} = X_{p=f(d),d}^{best}$  of the  $p$ -th particle, only when the random number within an interval  $[0, 1]$  (called  $rand$ ) is less than the value of the function  $Pc_p$ . Otherwise, the exemplar remains at its best position, namely  $X_{p,f(d)}^{best} = X_{p,d}^{best}$ . In a special case when all exemplars of the  $p$ -th particle are at their current best values  $X_{p,d}^{best}$ , the new position  $\mathbf{X}_{p,f}^{best}$  takes one of its exemplars randomly learnt from another particle  $\mathbf{X}_p^{best}$  at the same  $d$ -th dimension. The comprehensive learning strategy iteratively processes the best position  $\mathbf{X}_p^{best}$  (i.e.,  $\mathbf{X}_{p,f}^{best} = \mathbf{X}_p^{best}$ ) of the  $p$ -th particle, updated in Equations (2) and (3) for all  $nd$  dimensions.

The function  $Pc_p$  takes different values for different particles that do not vary during the optimization iteration. For instance, the values of  $Pc_p$  in Equation (5) for  $np = 20, 30, 40,$  and  $50$  particles are plotted in Figure 2.

$$Pc_p = 0.05 + 0.45 \frac{\left(\exp\left(\frac{10(p-1)}{np-1}\right) - 1\right)}{(\exp(10) - 1)} \tag{5}$$

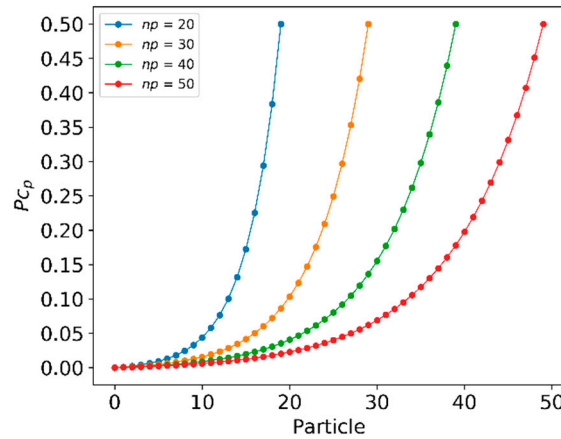


Figure 2. Learning probability function  $Pc_p$ .

The CLPSO performs comprehensive learning when the objective function does not consecutively improve for more than refreshing gap,  $rgap$  (e.g.,  $rgap = 5$ ), iterations [25]. The comprehensive searches for the new learning location  $f(d)$  are described by the flowchart in Figure 3.

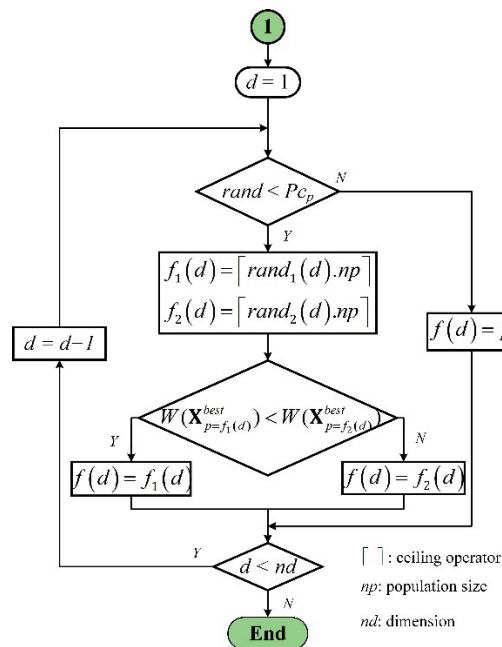


Figure 3. Comprehensive learning procedure for new learning locations  $f(d)$ .

### 3.2. Perturbation-Based Exploitation

To improve the exploitation ability of particles, an enhanced version of the CLPSO [26], called the ECLPSO, implements a normative knowledge structure, detailed in Table 1. The normative knowledge adopts a dimensional interval to all personal best positions of the population. To employ perturbation-based exploitation, a certain condition in the

normative knowledge is made for the decision of when the exploitation is performed effectively, and which region is focused.

**Table 1.** Normative knowledge.

Dimension	1	2	...	nd
Present dimensional lower bound	$\underline{P}_1$	$\underline{P}_2$	...	$\underline{P}_{nd}$
Present dimensional upper bound	$\overline{P}_1$	$\overline{P}_2$	...	$\overline{P}_{nd}$

The two lower  $\underline{P}_d$  and upper  $\overline{P}_d$  bounds to all personal best positions  $\mathbf{X}_p^{best}$  at the  $d$ -th dimension are defined by:

$$\underline{P}_d = \min \{ X_{1,d}^{best}, X_{2,d}^{best}, \dots, X_{np,d}^{best} \} \tag{6}$$

$$\overline{P}_d = \max \{ X_{1,d}^{best}, X_{2,d}^{best}, \dots, X_{np,d}^{best} \} \tag{7}$$

The perturbation-based exploitation improves the exploitation and accuracy of the CLPSO algorithm. This is applied to the standard CLPSO when the conditions stated in Equation (8) are satisfied. Then, the velocity is updated by Equation (9), now incorporating the perturbation-based exploitation term [26].

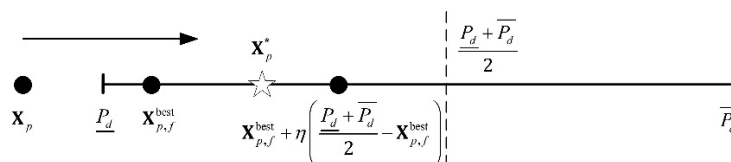
$$\begin{cases} \overline{P}_d - \underline{P}_d \leq \alpha (X_d^{\max} - X_d^{\min}) \\ \overline{P}_d - \underline{P}_d \leq \beta \end{cases} \tag{8}$$

$$\mathbf{V}_p^{next} = w_{pbE} \mathbf{V}_p + c_1 \cdot rand1_p \cdot (\mathbf{X}_{p,f}^{best} + \eta (\Gamma - \mathbf{X}_{p,f}^{best}) - \mathbf{X}_p) + c_2 \cdot rand2_p \cdot (\mathbf{X}^{global} - \mathbf{X}_p), \tag{9}$$

where  $X_s^{\max}$  and  $X_s^{\min}$  are the maximum and minimum positions at the  $d$ -th dimension, respectively;  $\alpha$  is the relative ratio (which equals to 0.01);  $\beta$  is the small absolute bound (which is set as 2);  $\eta$  is the perturbation coefficient (which is constructed randomly from a normal distribution with the mean value of 1 and standard deviation of 0.65); and  $w_{pbE}$  is the inertia weight (which is set as 0.5).

Each  $p$ -th particle moves toward  $\mathbf{X}_{p,f}^{best}$  with a perturbation term of  $\eta (\Gamma - \mathbf{X}_{p,f}^{best})$ , where  $\Gamma = \frac{\overline{P}_d - \underline{P}_d}{2}$  at the  $d$ -th dimension. The perturbation term can be considered in two ways. First, it is obvious that  $\frac{\underline{P}_d - \overline{P}_d}{2} \leq \frac{\overline{P}_d + \underline{P}_d}{2} - \mathbf{X}_{p,f}^{best} \leq \frac{\overline{P}_d - \underline{P}_d}{2}$ , and the perturbation term is proportional to a normative interval size of  $\overline{P}_d - \underline{P}_d$ . The smaller the interval size, the smaller the perturbation. Second, for the global optimal at the  $d$ -th dimension (i.e.,  $\mathbf{X}_p^*$  is often close to the normative interval center of  $\frac{\overline{P}_d + \underline{P}_d}{2}$ ), the perturbation term is proportional to the distance between  $\mathbf{X}_p$  and  $\frac{\overline{P}_d + \underline{P}_d}{2}$ , namely  $\mathbf{X}_{p,f}^{best}$  being closer to the interval center.

As a result, the velocity in Equation (9) adaptively determines the local search granularity based on the interval size as well as the distance between exemplar position and the interval center of the dimension. As Equation (9) pulls each particle toward some position other than the position of the  $d$ -th dimension, a better solution can be expected around the interval. When the description in Equation (8) is not satisfied, the standard velocity given in Equation (4) is adopted. The perturbation-based exploitation process is depicted in Figure 4.



**Figure 4.** Perturbation-based local search process.

### 3.3. Adaptive Learning Probability

Learning probabilities primarily capture an exemplar index in the CLPSO algorithm. Based on the particle index, these do not generally vary during the design iteration. The static learning probability often causes some difficulties in converging to an optimal solution. The ECLPSO method proposes a new adaptive learning probability function that is dynamically adjusted according to ranking information from a set of personal best particles [26]. The new adaptive learning probabilities in Equation (10) are adopted for the replacement of Equation (5) from the standard CLPSO:

$$Pc_p = L_{\min} + (L_{\max} - L_{\min}) \frac{\exp\left(\frac{10(K_p-1)}{np-1}\right) - 1}{\exp(10) - 1} \tag{10}$$

where

$$L_{\max} = L_{\min} + 0.25 + 0.45 \log_{(nd+1)}(M_{iter} + 1) \tag{11}$$

$L_{\min}$  is a positive scalar of 0.05 and  $M_{iter}$  is the number of dimensions (i.e., when Equation (8) is satisfied before or during an iteration  $iter$ ). Similar to Equation (5), the new function in Equation (10) incorporates the ranking parameter  $K_p$  that is defined by sorting the personal best fitness value in an ascending order. When the particle presents the best fitness value compared with others, its rank reads the value of 1 ( $K_p = 1$ ). On the other hand, its rank reads the total number of population  $np$  ( $K_p = np$ ) when the particle shows the worst fitness value.

To avoid premature solution convergence, an adjustment to  $L_{\max}$  in Equation (11) is required to for a good balance between exploration and exploitation of search spaces. More explicitly, a small  $L_{\max}$  value provides good exploration, whilst a large value supports exploitation procedures. In Equation (11),  $L_{\max} = 0.3$  reads the minimum value, when  $M_{iter} = 0$ . Otherwise,  $L_{\max} = 0.75$  takes the maximum value, when  $M_{iter} = nd$ . The presence of adaptive learning probability therefore enhances the exploitative ability of the population and accelerates solution convergence. This yields the main computational advantage of the ECLPSO, which overcomes the poor exploitation problem in standard CLPSO schemes.

### 4. Gaussian Local Search Strategy

The nature of the Gaussian distribution, unlike uniformity, is a bell-curved shape. It exploits more weights in the central field and improves the chance of finding accurate optimal solutions. The Gaussian (called normal) distribution denoted by  $N(\mu, \sigma^2)$  is characterized by the mean  $\mu$  and variance  $\sigma^2$  values, as follows:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right), \quad (\sigma > 0) \tag{12}$$

The present ECLPSO method applies the GLS strategy to define the deep local searches around  $\mathbf{X}^{global} = (X_d^{global} | \forall d \in \{1, \dots, nd\})$  and  $\mathbf{X}_p^{best} = (X_{p,d}^{best} | \forall d \in \{1, \dots, nd\})$ . The GLS maps out the positions  $\mathbf{X}_p^{next}$ , within the interval of  $[X_d^{\min}, X_d^{\max}]$  at the  $d$ -th dimension. A series of the new locations  $\mathbf{X}_p^{next}$  are constructed based on the Gaussian probability density function to test the global best  $\mathbf{X}^{global}$ . In essence, the best global  $\mathbf{X}^{global}$  remains rigorous when its objective function  $W(\mathbf{X}^{global})$  continues to present the most optimum, viz.,  $W(\mathbf{X}^{global}) \leq W(\mathbf{X}_p^{next})$ . Otherwise, the new global best position is updated by  $\mathbf{X}^{global} = \mathbf{X}_p^{next}$ , associated with  $W(\mathbf{X}^{global}) > W(\mathbf{X}_p^{next})$ .



The exemplars  $\mathbf{X}_p^{next}$  generated using the Gaussian probability density function in Equation (12), with the mean  $\mu = \mathbf{X}^{global}$  and variance  $\sigma^2 = |\mathbf{X}^{global} - \mathbf{X}_p^{best}|$  values for 1000 random samples, are depicted in Figure 5, where

$$\mathbf{X}_p^{next} \cong N(\mu, \sigma^2) = N(\mathbf{X}^{global}, |\mathbf{X}^{global} - \mathbf{X}_p^{best}|) \quad (13)$$

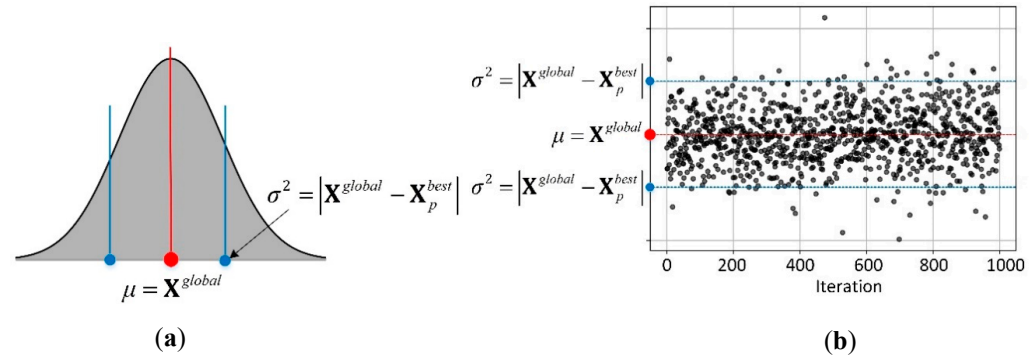


Figure 5. (a) GLS strategy with mean  $\mu$  and variance  $\sigma^2$  and (b)  $\mathbf{X}_p^{next}$  constructed around  $\mathbf{X}^{global}$ .

The figure graphically illustrates the GLS strategy that constructs a series of samples  $\mathbf{X}_p^{next}$  with a normal distribution around  $\mathbf{X}^{global}$ .

The ECLPSO method is turned into the GLS scheme when the current iteration  $iter$  reaches 80% of the total number of iterations ( $max\_iter$ ), namely  $iter \geq 0.8 \times max\_iter$ . The positions  $\mathbf{X}_p^{next}$  given in Equation (13) replace the velocities  $\mathbf{V}_p^{next}$  and the positions  $\mathbf{X}_p^{next}$  to construct the next position. The GLS scheme performs precise local searches around the global best  $\mathbf{X}^{global}$ , and therefore enhances the exploitative ability underlying the ECLPSO. The total number of iterations,  $max\_iter$ , is preset to a sufficiently large value that ensures convergence to accurate optimal solutions.

## 5. Optimization Procedure

The combined ECLPSO and GLS, or GLS-ECLPSO, method, is summarized by the flowchart in Figure 6. The pseudocode is provided in the following section.

### STEP 0: Initial swarm population

- Construct the initial swarm population  $\mathbf{X}_p$  for  $p \in \{1, \dots, np\}$ .
- Perform the comprehensive learning searches.

### STEP 1: Enhanced CLPSO

- Determine the velocities  $\mathbf{V}_p^{next}$  in Equation (9) if the conditions in Equation (8) satisfy; otherwise, the velocities  $\mathbf{V}_p^{next}$  read Equation (4).
- Update the new positions  $\mathbf{X}_p^{next}$  in Equation (3) and associated objective functions  $W(\mathbf{X}_p^{next})$ .
- Determine the best position  $\mathbf{X}_p^{best}$  of the  $p$ -th particle. If there is no improvement in  $W(\mathbf{X}_p^{best})$  for more than  $rgap$  consecutive swarm iterations, perform the comprehensive learning searches for the new  $p$ -th best position  $\mathbf{X}_p^{best}$ .
- Determine the global best position  $\mathbf{X}^{global}$  over the entire swarm population.
- Reiterate the ECLPSO processes in STEP 1. When  $iter \geq 0.8 \times max\_iter$ , perform the GLSs at STEP 2.

### STEP 2: Gaussian local searches

- Perform the GLSs to construct the samples  $\mathbf{X}_p^{next}$  in Equation (13) using the Gaussian distribution function in Equation (12).
- Test the global best position  $\mathbf{X}^{global}$ . If  $W(\mathbf{X}^{global}) > W(\mathbf{X}_p^{next})$ , update  $\mathbf{X}^{global} = \mathbf{X}_p^{next}$ .

- Repeat STEP 2 until  $iter = max\_iter$ .

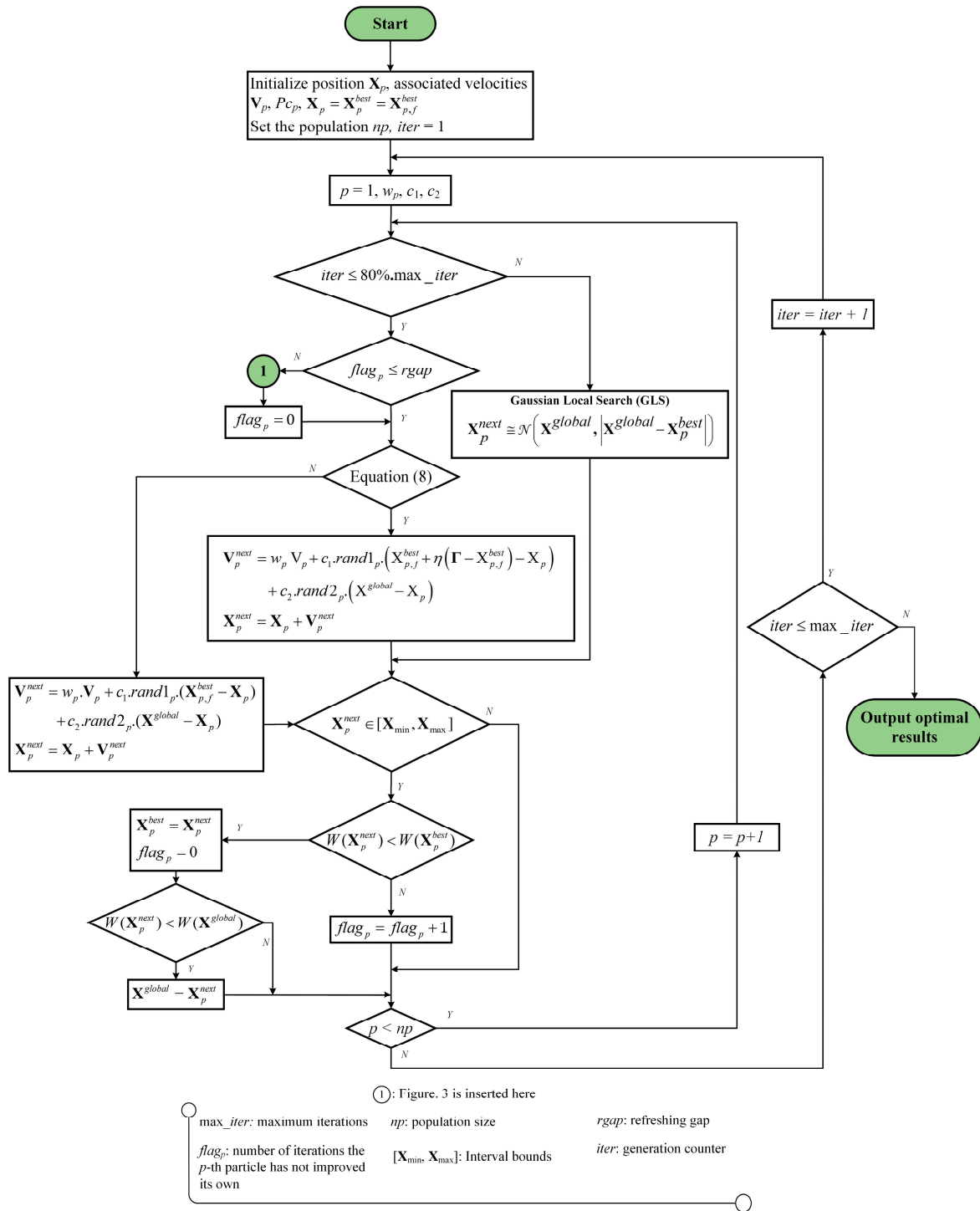


Figure 6. GLS-ECLPSO procedure.

### 6. Illustrative Examples

The applications of the proposed GLS-ECLPSO method are illustrated for the optimal size and shape designs of planar and spatial truss structures. Four examples [13,29–37], consisting of different geometries and sizes, have generally been adopted to validate the accuracy and robustness of various meta-heuristic algorithms. These were formulated as the challenging non-convex and non-smooth NLP problems in Equation (1). The presence of permissible (buckling) stress and serviceability conditions with discrete area variables

posed the major difficulty in processing such a problem. For each example, the solutions (including the total design weight, total number of analyses, and statistical values) collected from 25 independent GLS-ECLPSO solves were reported and compared with those available in the literature. The proposed GLS-ECLPSO algorithm was encoded as a Python code, made available for download at <https://github.com/thuchula6792/GLS-ECLPSO> (accessed on 6 September 2022).

6.1. Example 1: 15-Bar Cantilever Truss

The first example considered a 15-bar planar truss structure [29], subjected to a vertical load (10 kips) at node 8, as shown in Figure 7. The structure was discretized into 15 pin-connected members and 8 nodes ( $ng = 23$ ;  $ne = 15$ ; and  $nn = 8$ ). All structural members were subjected to the stress limits of  $\pm 25$  ksi in tension and compression.

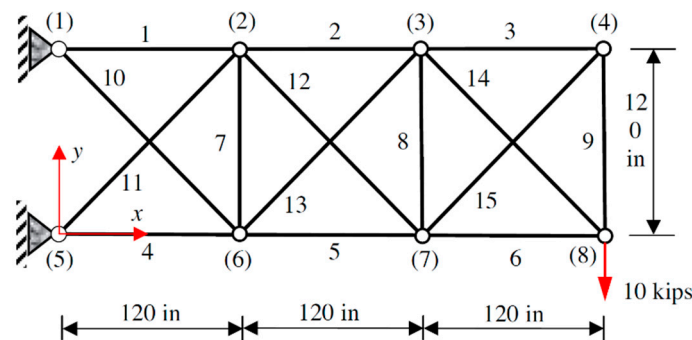


Figure 7. Example 1: 15-bar planar truss.

The material properties adopted had a density of  $0.1 \text{ lb-in}^{-3}$  and modulus of elasticity of  $10^4$  ksi. Member sizes were solely selected from a discrete set of available areas, and nodal coordinate variables were  $(x_n, y_n) \in \mathbb{R}^2$  for  $\forall n \in \{1, \dots, 8\}$ , as detailed in Table 2. For practical designs, a symmetric layout was imposed as a design criterion in the optimization problem. The NLP problem in Equation (1) consisted of 23 design variables  $\mathbf{X} \in \mathbb{R}^{23} = (\mathbf{a}, \mathbf{x}, \mathbf{y})$ , namely 15 unknown member sizes (in<sup>2</sup>-unit) and 8 unknown nodal x-y coordinates (in-unit).

Table 2. Example 1: design variables and constraints.

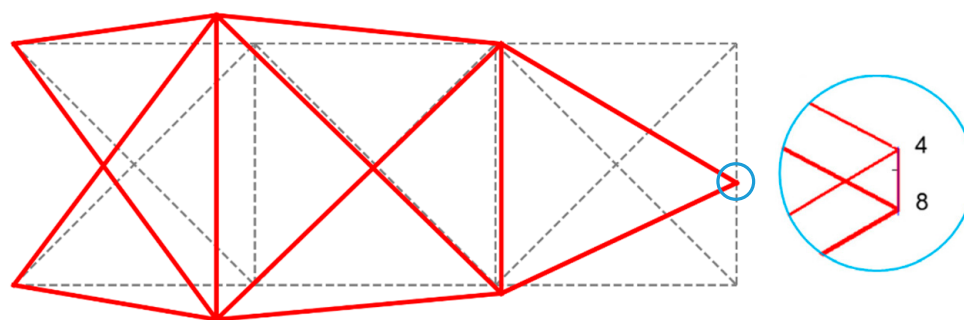
Objective Function:	$W(\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{m=1}^{15} \left( \sum_{g=1}^{23} Q_{m,g} \cdot a_g \cdot \rho_g \right) \cdot L_m(\mathbf{x}, \mathbf{y}, \mathbf{z})$
Stress constraints:	$\begin{cases} \sigma_m \leq 25 \text{ (ksi) in tension} \\ \sigma_m \geq -25 \text{ (ksi) in compression} \end{cases}$ , for all $m \in \{1, 2, \dots, 15\}$
Size variables:	$a_g$ , for all $g \in \{1, 2, \dots, 15\}$
Shape variables:	$x_2 = x_6, x_3 = x_7, y_2, y_3, y_4, y_6, y_7, y_8$
Layout conditions:	$100 \leq x_2 \leq 140$
	$220 \leq x_3 \leq 260$
	$100 \leq y_2 \leq 140$
	$100 \leq y_3 \leq 140$
	$50 \leq y_4 \leq 90$
	$-20 \leq y_6 \leq 20$
	$-20 \leq y_7 \leq 20$
	$20 \leq y_8 \leq 60$
Discrete area variables:	$a_g \in \{0.111, 0.141, 0.174, 0.220, 0.270, 0.287, 0.347, 0.440, 0.539, 0.954, 1.081, 1.174, 1.333, 1.488, 1.764, 2.142, 2.697, 2.800, 3.131, 3.565, 3.813, 4.8055, 952, 6.572, 7.192, 8.525, 9.300, 10.850, 13.330, 14.290, 17.170, 19.180\}$ (in <sup>2</sup> )
Young modulus:	$E = 10^4$ (ksi)
Material density:	$\rho = 0.1$ (lb/in <sup>3</sup> )

The proposed GLS-ECLPSO algorithm was performed on the total 20 particles (i.e.,  $np = 20$ ;  $\max\_iter = 300$ ), and successfully obtained the optimal solution of the truss. As detailed in Table 3, the computed minimum total weight of  $W(\mathbf{a}) = 74.1723$  lb presented the least value, compared with the reference methods [13,29–31]. The optimal layout of the designed truss, which varied from its original shape, is depicted in Figure 8.

**Table 3.** Example 1: optimal size and shape design solutions.

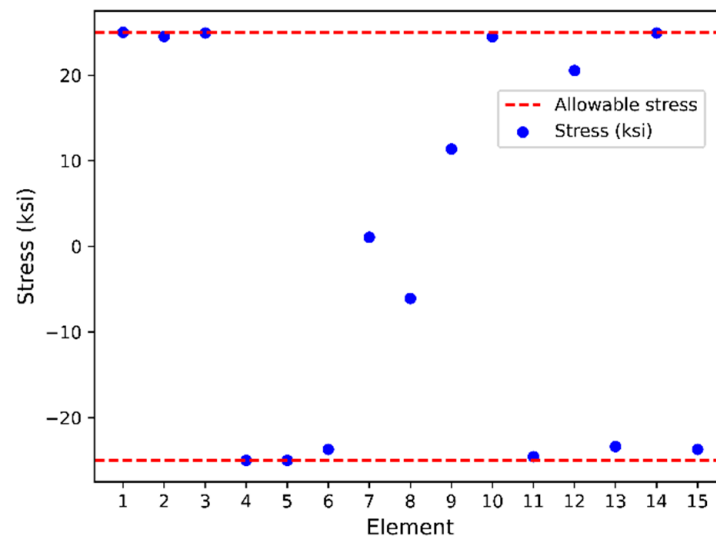
Variables	GA [30]	FA [31]	PSO [29]	CPSO [29]	TLBO [32]	D-ICDE [13]	GLS-ECLPSO (Present)
A1	1.081	0.954	0.954	1.174	1.081	1.081	0.954
A2	0.539	0.539	1.081	0.539	0.954	0.539	0.539
A3	0.287	0.220	0.270	0.347	0.141	0.141	0.220
A4	0.954	0.954	1.081	0.954	1.081	0.954	0.954
A5	0.539	0.539	0.539	0.954	0.539	0.539	0.539
A6	0.141	0.220	0.287	0.141	0.347	0.287	0.220
A7	0.110	0.111	0.141	0.141	0.111	0.111	0.111
A8	0.110	0.111	0.111	0.111	0.174	0.111	0.111
A9	0.539	0.287	0.347	1.174	0.141	0.141	0.440
A10	0.440	0.440	0.440	0.141	0.270	0.347	0.440
A11	0.539	0.440	0.270	0.440	0.220	0.440	0.440
A12	0.270	0.220	0.111	0.440	0.141	0.270	0.270
A13	0.220	0.220	0.347	0.141	0.440	0.270	0.220
A14	0.141	0.270	0.440	0.141	0.347	0.287	0.220
A15	0.287	0.220	0.220	0.347	0.141	0.174	0.220
X2	101.5775	114.967	106.0521	102.287	100.004	100.031	100.9857
X3	227.9112	247.040	239.0245	240.505	241.047	238.701	242.8470
Y2	134.7986	125.919	130.3556	112.584	118.823	132.847	134.2018
Y3	128.2206	111.067	114.273	108.043	100.083	125.367	119.9010
Y4	54.8630	58.298	51.9866	57.795	50.000	60.307	50.8212
Y6	−16.4484	−17.564	1.8135	−6.429	3.141	−10.665	−17.1359
Y7	−13.3007	−5.821	9.1827	−1.800	−9.699	−12.245	−4.1215
Y8	54.8572	31.465	46.9087	57.798	46.896	59.993	50.7841
Best weight (lb)	76.6854	75.55	82.2344	77.615	76.652	74.682	74.1723
Constraint violation (%)	0.0	N/A	0.0	0.0	N/A	0.0	0.0
No. of analyses	8000	8000	4500	4500	N/A	8000	6000
SD	N/A	2.96	N/A	N/A	2.42	N/A	3.22

Note: GA = genetic algorithm; FA = firefly algorithm; PSO = particle swarm optimization; CPSO = cellular automata and particle swarm optimization; TLBO = teaching-learning-based optimization; D-ICDE = discrete improved constrained differential evolution.



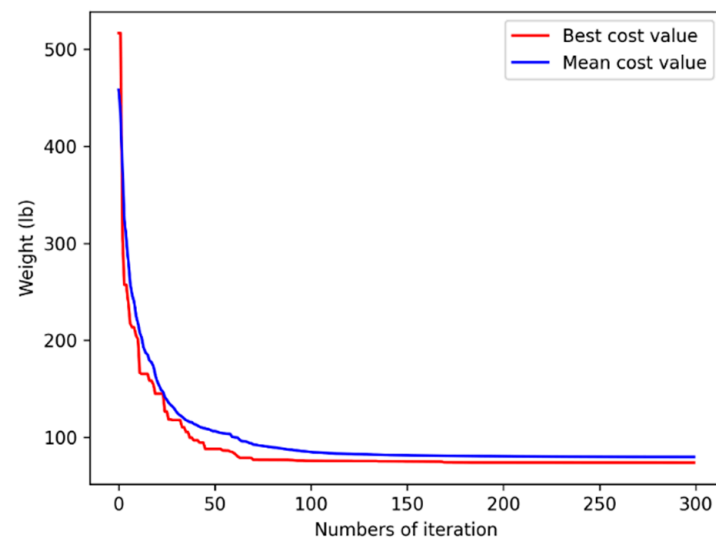
**Figure 8.** Example 1. optimal shape solution.

All 25 independent designs strictly complied with the imposed constraints. The maximum stress developed was 24.9964 ksi, as shown in Figure 9. The statistical values in Table 3 show that the proposed GLS-ECLPSO captured the most minimum weight with a small standard deviation (namely,  $SD = 3.22$  lb).



**Figure 9.** Example 1: permissible and designed member stresses.

The convergence of designed total weights, defined by the unconstrained nonlinear function in Equation (2), was plotted in Figure 10 for both the best and mean values to the optimal solutions. This illustrated that the GLS-ECLPSO quickly converged to the minimum weight of the structure as it approached the first 200 iterations of the maximum preset 300 iterations.



**Figure 10.** Example 1: solution convergence.

### 6.2. Example 2: 18-Bar Planar Truss

The 18-bar planar truss [29] in Figure 11 was applied by the vertical forces of 20 kips at nodes 8, 6, 4, 2, and 1. The structure was modeled as 18 pin-connected members with 11 nodes (i.e.,  $ng = 12$ ;  $ne = 18$ ; and  $nm = 11$ ). The material properties adopted had a density of  $0.1 \text{ lb-in}^{-3}$  and modulus of elasticity of  $10^4 \text{ ksi}$ . The member sizes were solely selected from the available set of discrete areas. Similar area conditions of members within the groups are specified in Table 4. The structure was designed under buckling constraints, with a buckling coefficient of  $K = 4$  in compression and a permissible tensile stress of 25 ksi.

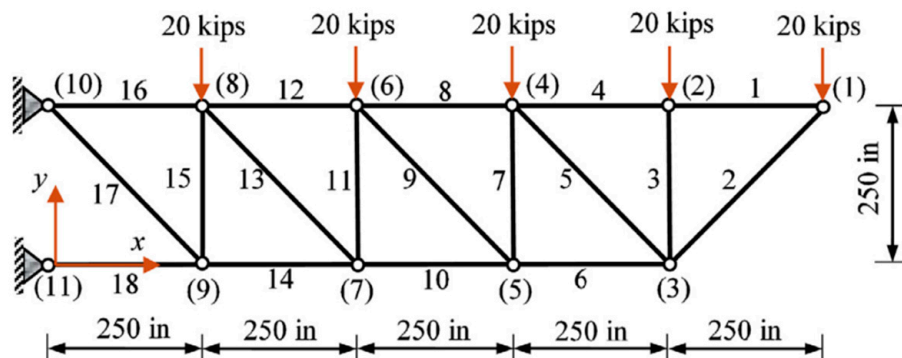


Figure 11. Example 2: 18-bar planar truss.

Table 4. Example 2: design variables and constraints.

Objective Function:	$W(\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{m=1}^{ne=18} \left( \sum_{g=1}^{ng=12} Q_{m,g} \cdot a_g \cdot \rho_g \right) \cdot L_m(\mathbf{x}, \mathbf{y}, \mathbf{z})$
Stress constraints:	$\begin{cases} \sigma_m \leq 25 \text{ (ksi) in tension} \\ \sigma_m \geq -25 \text{ (ksi) in compression} \end{cases}, \text{ for all } m \in \{1, 2, \dots, 18\}$
Buckling constraints:	$ \sigma_m  \leq KEA_m/L_m^2, \text{ for all } m \in \{1, 2, \dots, 18\}$
Size variables:	$A_1 = A_4 = A_8 = A_{12} = A_{16}, A_2 = A_6 = A_{10} = A_{14} = A_{18},$ $A_3 = A_7 = A_{11} = A_{15}, A_5 = A_9 = A_{13} = A_{17}$
Shape variables:	$x_3, y_3, x_5, y_5, x_7, y_7, x_9, y_9$
Discrete area variables:	$a_g \in \mathbb{R}^{ng} = \{2.00, 2.25, 2.50, \dots, 21.25, 21.50, 21.75\} \text{ (in}^2\text{)}$
Layout conditions:	$775 \leq x_3 \leq 1225$ $525 \leq x_5 \leq 975$ $275 \leq x_7 \leq 725$ $25 \leq x_9 \leq 475$ $-225 \leq y_3, y_5, y_7, y_9 \leq 245$
Young modulus:	$E = 10^4 \text{ (ksi)}$
Buckling coefficient:	$K = 4$
Material density:	$\rho = 0.1 \text{ (lb/in}^3\text{)}$

The nodal coordinates  $(x_n, y_n) \in \mathbb{R}^2$  for  $n \in \{1, 2, 4, 6, 8, 10, 11\}$  at the top-chord members and two restrained points were unvaried, whilst those for  $n \in \{3, 5, 7, 9\}$  at the bottom-chord were designed for an optimal layout. This led to a total of twelve design variables  $\mathbf{X} \in \mathbb{R}^{12} = (\mathbf{a}, \mathbf{x}, \mathbf{y})$ , consisting of four unknown member sizes (in<sup>2</sup>-unit) and eight unknown nodal x-y coordinates (in-unit).

The simultaneous size and shape optimization was formulated as the non-convex and non-smooth NLP problem in Equation (1). The GLS-ECLPSO method constructed the total 20 particles with the algorithmic parameters of  $\text{max\_iter} = 300$ . The minimum weight of  $W(\mathbf{a}) = 4175.1425 \text{ lb}$  was successfully determined. All constraints on permissible and buckling forces developed for design members were strictly complied with, as shown in Figure 12. The corresponding optimal layout and normalized buckling force ratios are depicted in Figure 13.

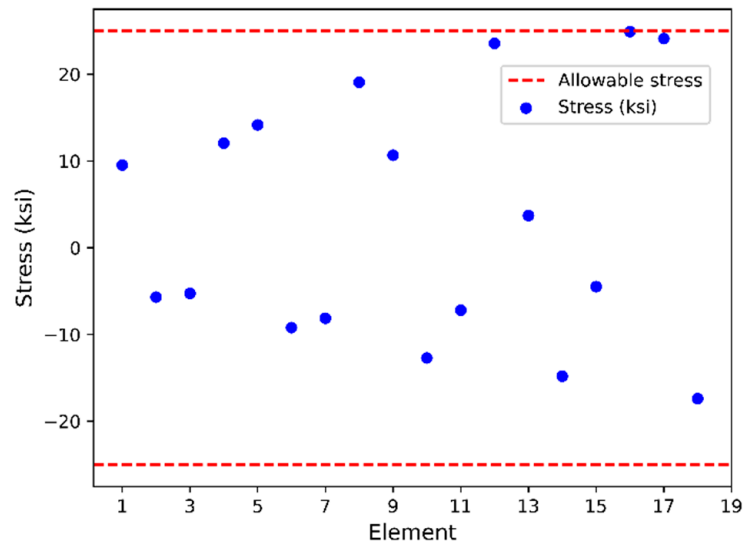


Figure 12. Example 2: permissible and designed member stresses.

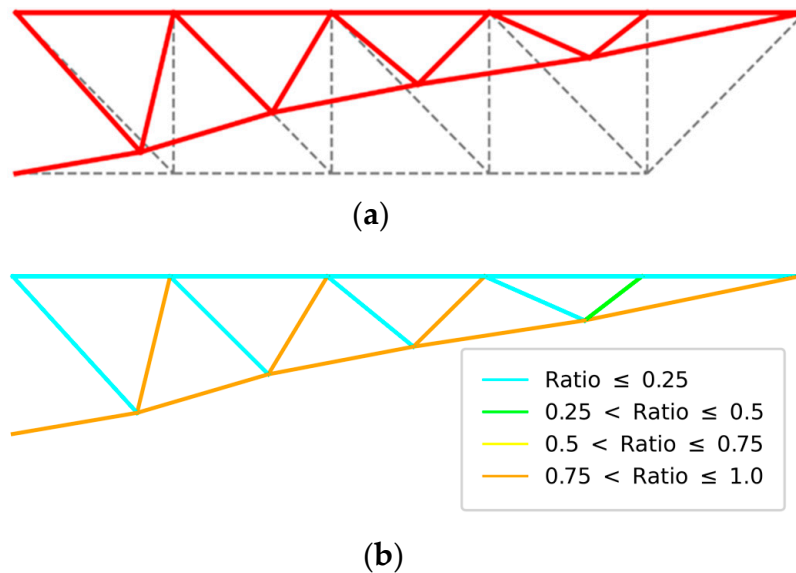


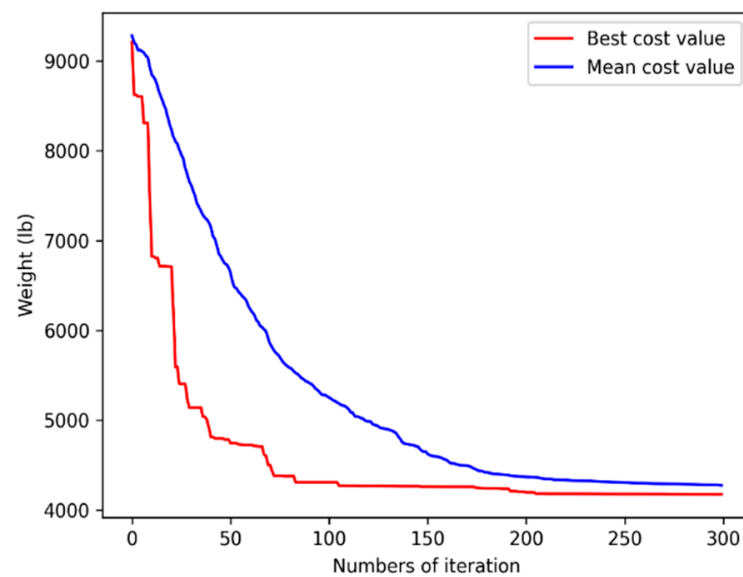
Figure 13. Example 2: optimal design solution with (a) optimal shape and (b) normalized buckling force ratio.

The optimal solution presented a total weight value that was 8% less than those reported in the literature [13,29,30,33]. Table 5 summarizes the optimal solutions, as well as the statistical values obtained from 25 independent runs. In essence, the small standard deviation of 57.32 lb is addressed. This illustrates the superior performance of the proposed method in obtaining accurate and reliable optimal design solutions that are insensitive to the underlying algorithmic random parameters. Good convergence to the (best and mean) optimal design solutions was shown in the first 200 iterations, as shown in Figure 14.

**Table 5.** Example 2: optimal size and shape design solutions.

Variables	GA [30]	PSO [29]	SCPSO [29]	D-ICDE [13]	ABC [33]	GLS-ECLPSO (Present)
A1	12.75	12.00	12.5	13	12.50	10.25
A2	18.50	18.50	17.5	17.5	17.75	17.5
A3	4.75	5.25	5.75	6.5	5.75	6.25
A4	3.25	4.50	3.75	3.0	3.75	2.75
X3	917.4475	903.9806	907.2491	914.06	912.9974	909.0566
Y3	193.7899	185.7807	179.8671	183.46	183.6806	180.1431
X5	654.3243	644.9170	636.7873	640.53	642.7143	636.8079
Y5	159.9436	144.9692	141.8271	133.74	143.8920	138.5523
X7	424.4821	428.2196	407.9442	406.12	411.6918	406.1961
Y7	108.5779	100.5623	94.0559	92.63	97.14763	94.7056
X9	208.4691	209.5415	198.7897	196.69	200.9087	197.5999
Y9	37.6349	24.3748	29.5157	37.06	30.21906	33.6241
Best weight (lb)	4530.68	4609.001	4512.365	4554.29	4537.064	4175.1425
Constraint violation (%)	0.0	0.0	0.0	0.0	0.0	0.0
No. of analyses	8000	4500	4500	8025	2700	6000
SD	N/A	N/A	37.691	N/A	9.7971	57.32

Note: GA = genetic algorithm; PSO = particle swarm optimization; SCPSO = sequential cellular and particle swarm optimization; D-ICDE = discrete improved constrained differential evolution; ABC = artificial bee colony algorithm.

**Figure 14.** Example 2: solution convergence.

### 6.3. Example 3: 47-Bar Tower Truss

The third example designed the simultaneous shape and member sizes of a 47-bar tower truss in Figure 15 [29]. Three different load cases (viz., each case involved similar forces applied at either node 17 or 22, or both nodes) were considered (see Table 6), and the design was subjected to three multiple load cases. The presence of multiple load cases was considered by assigning each member stress and each nodal displacement with their associated maximum (critical) responses developed under the envelop of three load cases.



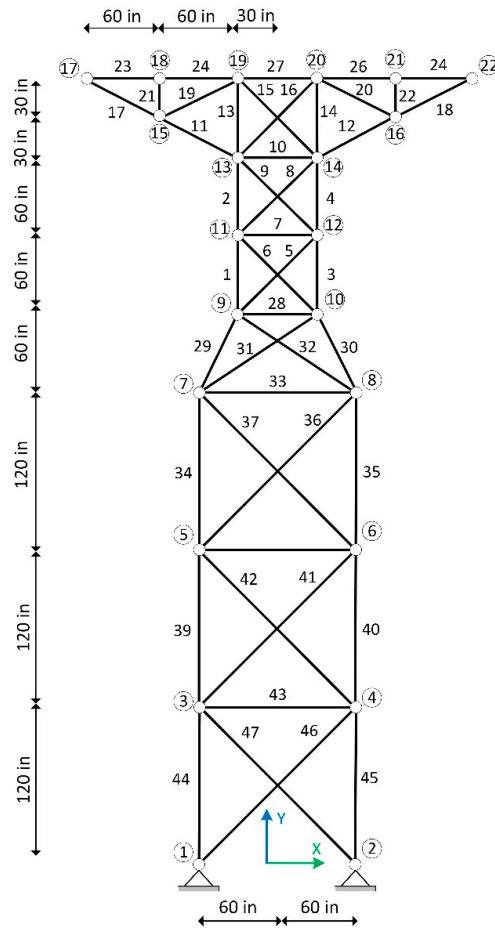


Figure 15. Example 3: 47-bar tower truss.

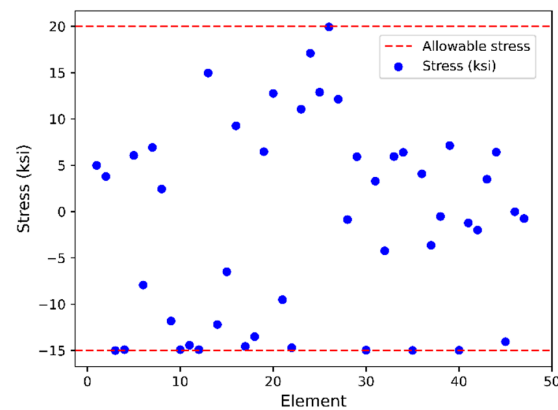
Table 6. Example 3: design variables and constraints.

Objective Function:	$W(\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{m=1}^{ne=47} \left( \sum_{g=1}^{ng=44} Q_{m,g} \cdot a_g \cdot \rho_g \right) \cdot L_m(\mathbf{x}, \mathbf{y}, \mathbf{z})$
Stress constraints:	$\begin{cases} \sigma_m \leq 20 \text{ (ksi) in tension} \\ \sigma_m \geq -15 \text{ (ksi) in compression} \end{cases}, \text{ for all } m \in \{1, 2, \dots, 47\}$
Buckling constraints:	$ \sigma_m  \leq KEA_m / L_m^2, \text{ for all } m \in \{1, 2, \dots, 47\}$
Size variables:	$A_3 = A_1, A_4 = A_2, A_5 = A_6, A_7, A_8 = A_9, A_{10}, A_{12} = A_{11}, A_{14} = A_{13}, A_{15} = A_{16}, A_{18} = A_{17}, A_{20} = A_{19}, A_{22} = A_{21}, A_{24} = A_{23}, A_{26} = A_{25}, A_{27}, A_{28}, A_{30} = A_{29}, A_{31} = A_{32}, A_{33}, A_{35} = A_{34}, A_{36} = A_{37}, A_{38}, A_{40} = A_{39}, A_{41} = A_{42}, A_{43}, A_{45} = A_{44}, A_{46} = A_{47}$
Shape variables:	$x_2 = -x_1, x_4 = -x_3, y_4 = y_3, x_6 = -x_5, y_6 = y_5, x_8 = -x_7, y_8 = y_7, x_{10} = -x_9, y_{10} = y_9, x_{12} = -x_{11}, y_{12} = y_{11}, x_{14} = -x_{13}, y_{14} = y_{13}, x_{20} = -x_{19}, y_{20} = y_{19}, x_{21} = -x_{18}, y_{21} = y_{18}$
Discrete area variables:	$a_g \in \mathbb{R}^{ng} = \{0.1, 0.2, 0.3, \dots, 4.8, 4.9, 5.0\} \text{ (in}^2\text{)}$
Layout conditions:	$x_i, y_i \in \mathbb{R}$
Loads (kips):	Case 1: at node 17: $F_x = 6, F_y = -14$ Case 2: at node 22: $F_x = 6, F_y = -14$ Case 3: at nodes 17 and 22: $F_x = 6, F_y = -14$
Young modulus:	$E = 3 \times 10^4 \text{ (ksi)}$
Buckling coefficient:	$K = 3.96$
Material density:	$\rho = 0.3 \text{ (lb/in}^3\text{)}$

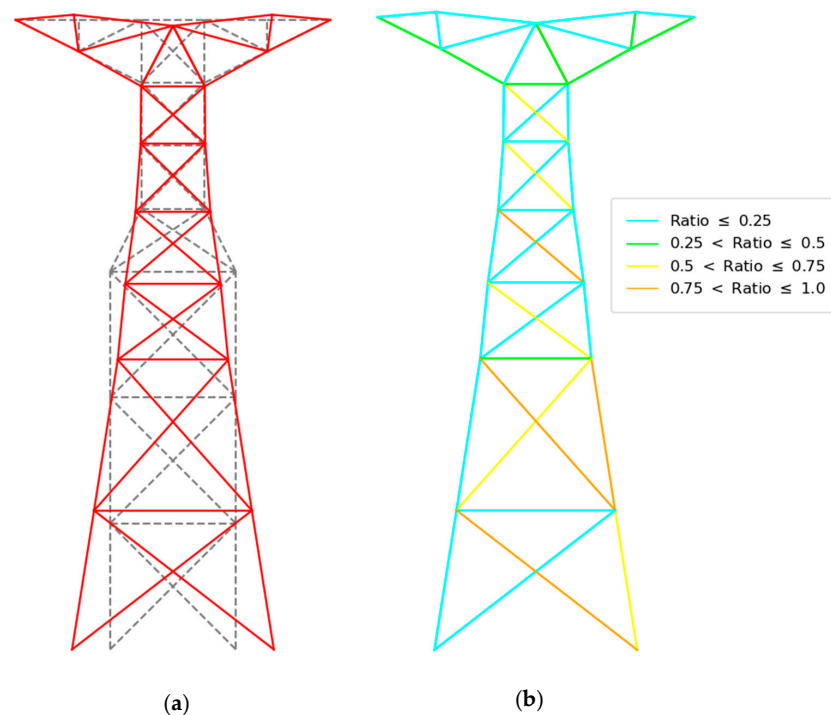
The material properties had a density of  $0.3 \text{ lb-in}^{-3}$  and modulus of elasticity of  $3 \times 10^4 \text{ ksi}$ , with permissible stresses of 20 ksi in tension and 15 ksi in compression.

The structure was discretized into 47-pin-connected members and 22 nodes ( $ng = 44$ ;  $ne = 47$ ; and  $nn = 22$ ). A symmetric layout about the  $y$ -axis was imposed and led to a total of 44 design variables  $\mathbf{X} \in \mathbb{R}^{44} = (\mathbf{a}, \mathbf{x}, \mathbf{y})$ , including 27 unknown member sizes (in<sup>2</sup>-unit) and 17 unknown nodal x-y coordinates (in-unit). Six specific nodes, namely 1 and 2 at supports and 15–17 and 22 at the tower arms, were restrained to their original locations. The design solely selected the member sizes from the set of discrete areas with an incremental step of  $0.1 \text{ in}^2$  and determined the nodal coordinates  $(x_n, y_n) \in \mathbb{R}^2$  for  $\forall n \in \{1, \dots, 22\}$  in the presence of stress constraints, where the Euler's buckling load of  $3.96 EA.L^{-2}$  was enforced to all members.

The proposed GLS-ECLPSO method generated a total of 20 particles (viz.,  $np = 20$ ,  $\text{max\_iter} = 1500$ ) and successfully converged to the minimum total weight of  $W(\mathbf{a}) = 1799.8757 \text{ lb}$  for the best designed truss structure. The optimal design strictly satisfied the permissible stress conditions, drawn in Figure 16. The associated optimal layout with normalized buckling load ratios developed within the members are plotted in Figure 17.



**Figure 16.** Example 3: permissible and designed member stresses.



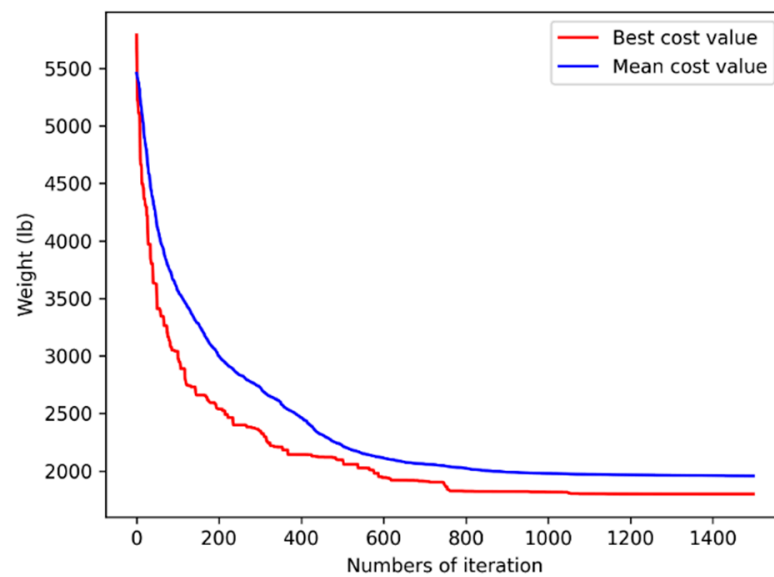
**Figure 17.** Example 3: optimal design solution with (a) optimal shape and (b) normalized buckling force ratio.

The computed results, including member areas, nodal coordinates, and statistical values given by 25 independent solves, were summarized in Table 7. The most minimum design was achieved compared with those reported in the literature [29,34–37]. Moreover, the small standard deviation of 89.53 lb described the reliability of the proposed method in repetitively obtaining accurate designs under population-based random parameters. The fast convergence to the best and mean designed total weights of the structure is depicted in Figure 18 and presents the superior performance of the GLS-ECLPSO method in capturing the near-optimal solution for the non-convex and non-smooth NLP problem in Equation (1).

**Table 7.** Example 3: optimal size and shape design solutions.

Variables	SA [34]	SSO [35]	PSO [29]	SCPSO [29]	ABC [33]	CA-ICEA [36]	GLS-ECLPSO (Present)
A3	2.5	2.8	2.80	2.5	2.4	2.7	2.7
A4	2.5	2.5	2.70	2.5	2.2	2.5	1.9
A5	0.8	0.7	0.80	0.8	1.1	0.7	0.8
A7	0.1	0.1	1.10	0.1	0.1	0.1	0.5
A8	0.7	1.0	0.80	0.7	1.2	0.9	1.1
A10	1.3	1.1	1.30	1.4	1.3	1.1	1.7
A12	1.8	1.8	1.80	1.7	1.7	1.8	2.2
A14	0.7	0.7	0.90	0.8	0.6	0.7	0.5
A15	0.9	0.8	1.20	0.9	0.8	0.9	0.9
A18	1.2	1.5	1.40	1.3	1.6	1.3	1.9
A20	0.4	0.4	0.30	0.3	0.3	0.3	0.4
A22	1.3	1.0	1.40	0.9	0.9	1.1	0.4
A24	0.9	1.1	1.10	1.0	1.2	1.0	1.7
A26	0.9	1.0	1.20	1.1	1.0	0.9	1.5
A27	0.7	5.0	1.60	5.0	1.0	0.8	2.3
A28	0.1	0.1	1.00	0.1	0.6	0.1	0.3
A30	2.5	2.7	2.80	2.5	2.8	2.7	3.1
A31	1.0	0.9	0.80	1.0	0.4	0.8	0.5
A33	0.1	0.1	0.10	0.1	0.1	0.1	0.1
A35	2.9	3.0	3.00	2.8	2.9	3.0	3.3
A36	0.8	0.8	0.90	0.9	1.5	0.9	0.8
A38	0.1	0.1	0.10	0.1	0.6	0.1	0.1
A40	3.0	3.2	3.30	3.0	3.1	3.2	3.4
A41	1.2	1.1	0.90	1.0	0.9	1.0	0.7
A43	0.1	0.1	0.10	0.1	0.1	0.1	0.2
A45	3.2	3.3	3.30	3.2	3.3	3.3	3.7
A46	1.1	1.1	1.20	1.2	0.8	1.1	0.3
X2	104	100.5396	98.8628	101.3393	103.6063	99.8037	96.1045
X4	87	81.0279	78.6595	85.9111	81.5008	81.2026	75.1729
Y4	128	137.2003	146.7331	135.9645	143.0525	137.3511	132.4016
X6	70	63.8334	66.5231	74.7969	67.0169	63.7482	52.6328
Y6	259	254.1838	239.0901	237.7447	252.8466	249.2955	276.0971
X8	62	56.1445	55.6936	64.3115	54.5203	54.2828	45.4036
Y8	3.26	327.9040	327.7882	321.3416	374.0126	338.6518	348.3091
X10	53	48.2708	48.8641	53.3345	39.8226	49.1830	35.6093
Y10	412	407.5132	398.6775	414.3025	443.9461	404.2573	417.0551
X12	47	42.4458	43.1400	46.0277	30.9474	44.0380	30.7598
Y12	486	468.8267	464.7831	489.9216	491.9941	467.5475	482.1343
X14	45	45.8692	37.8993	41.8353	36.7597	44.4605	30.2856
Y14	504	515.2907	511.0450	522.4161	510.000	511.3081	536.6923
X20	2.0	0.0010	18.2341	1.0005	17.6763	4.0141	0.1239
Y20	584	586.9443	594.0710	598.3905	598.8911	590.5903	594.3145
X21	89	80.7351	90.9369	97.8696	77.6661	84.3733	94.5263
Y21	637	621.5769	621.3943	624.0552	619.8911	630.3705	604.8316
Best weight (lb)	1871.17	1869.876	1975.839	1864.10	1871.843	1844.71	1799.8757
Constraint violation (%)	0.0	N/A	0.0	0.0	N/A	$7.8409 \times 10^{-6}$	0.0
No. of analyses	N/A	20,020	25,000	25,000	18,000	5016	30,000
SD	N/A	29.55	N/A	97.478	7.565	N/A	89.53

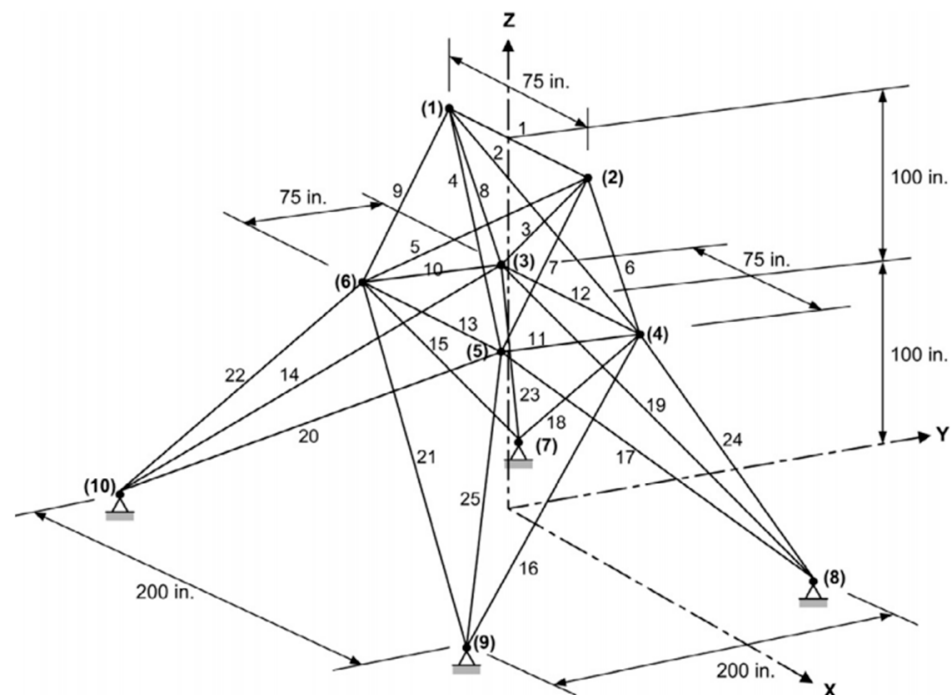
Note: SA = simulated annealing; SSO = shuffled shepherd optimization; PSO = particle swarm optimization; SCPSO = sequential cellular and particle swarm optimization; ABC = artificial bee colony algorithm; CA-ICEA = cellular automata-imperialist competitive algorithm.



**Figure 18.** Example 3: solution convergence.

#### 6.4. Example 4: 25-Bar Space Truss

The final example considered the 25-bar space (three-dimensional) truss structure [29] shown in Figure 19 that was subjected to applied load regimes listed in Table 8. The material properties employed throughout had a density of  $0.1 \text{ lb-in}^{-3}$  and a modulus of elasticity of  $10^4 \text{ ksi}$ . The simultaneous size and shape optimization of this structure determined its optimal layout and distribution of member areas under the design criteria, stating the allowable stresses of 40 ksi in both tension and compression, as well as limited displacements of 0.35 in all x-y-z directions at nodes 1 to 6, simultaneously.



**Figure 19.** Example 4: 25-bar space truss.

**Table 8.** Example 4: design variables and constraints.

Objective Function:	$W(\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{m=1}^{ne=25} \left( \sum_{g=1}^{ng=13} Q_{m,g} \cdot a_g \cdot \rho_g \right) \cdot L_m(\mathbf{x}, \mathbf{y}, \mathbf{z})$
Stress constraints:	$\begin{cases} \sigma_m \leq 40 \text{ (ksi) in tension} \\ \sigma_m \geq -40 \text{ (ksi) in compression} \end{cases}, \text{ for all } m \in \{1, 2, \dots, 25\}$
Displacement constraints:	$\delta_n \leq 0.35 \text{ in, for all } n \in \{1, 2, \dots, 6\}$
Size variables:	$A_1, A_2 = A_3 = A_4 = A_5, A_6 = A_7 = A_8 = A_9, A_{10} = A_{11}$ $A_{12} = A_{13}, A_{14} = A_{15} = A_{16} = A_{17}, A_{18} = A_{19} = A_{20} = A_{21},$ $A_{22} = A_{23} = A_{24} = A_{25}$
Shape variables:	$x_4 = x_5 = -x_3 = -x_6, x_8 = x_9 = -x_7 = -x_{10},$ $y_3 = y_4 = -y_5 = -y_6, y_7 = y_8 = -y_9 = -y_{10},$ $z_3 = z_4 = z_5 = z_6$
Discrete area variables:	$a_g \in \mathbb{R}^{ng} = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1$ $1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1, 2.2,$ $2.3, 2.4, 2.5, 2.6, 2.8, 3.0, 3.2, 3.4\} \text{ (in}^2\text{)}$
Layout conditions:	$20 \leq x_4 \leq 60$ $40 \leq x_8 \leq 80$ $40 \leq y_4 \leq 80$ $100 \leq y_8 \leq 140$ $90 \leq z_4 \leq 130$
Loads (kips):	Node 1: $F_x = 1, F_y = -10, F_z = -10$ Node 2: $F_x = 0, F_y = -10, F_z = -10$ Node 3: $F_x = 0.5, F_y = 0, F_z = 0$ Node 6: $F_x = 0.6, F_y = 0, F_z = 0$
Young modulus:	$E = 10^4 \text{ (ksi)}$
Material density:	$\rho = 0.1 \text{ (lb/in}^3\text{)}$

The structure was modeled as 25-pin-connected members with 10 nodes (i.e.,  $ng = 13$ ;  $ne = 25$ ; and  $nm = 10$ ). The governing NLP problem in Equation (1) consisted of 13 design variables  $\mathbf{X} \in \mathbb{R}^{13} = (\mathbf{a}, \mathbf{x}, \mathbf{y}, \mathbf{z})$ , including 8 unknown member sizes (in<sup>2</sup>-unit) and 5 unknown coordinates  $(x_n, y_n, z_n) \in \mathbb{R}^3$  (in-unit) at nodes 4 and 8. The member sizes were selected solely from the set of discrete areas, and the nodal coordinates were imposed with the conditions stated in Table 8.

The GLS-ECLPSO approach with the total 20 particles ( $np = 20$ ,  $max\_iter = 300$ ) successfully determined the least minimum total weight of  $W(\mathbf{a}) = 118.045$  lb, compared with the designs given in the literature [13,29–31,37]. The plots in Figure 20 illustrated the satisfaction of permissible stresses and limited displacements at nodes 1 to 6. The diagram also showed the limited displacement conditions governing the optimal design. Contrary to the displacements, the member stresses were far from the permissible values.

As detailed in Table 9, the standard deviation associated with the 25 independent design runs was only 4.2 lb, which indicated good performance of the proposed method in reliably obtaining the optimal size and shape designs for the class of problems considered. The optimal layout corresponding to the best design is plotted in Figure 21, presenting geometric variation from its original shape. Good convergence to the best and mean values of the total weights in Figure 22 was observed for a series of GLS-ECLPSO processes. It showed that optimal solutions were achieved in the first 280 iterations of the total preset 300 iterations in the algorithm.

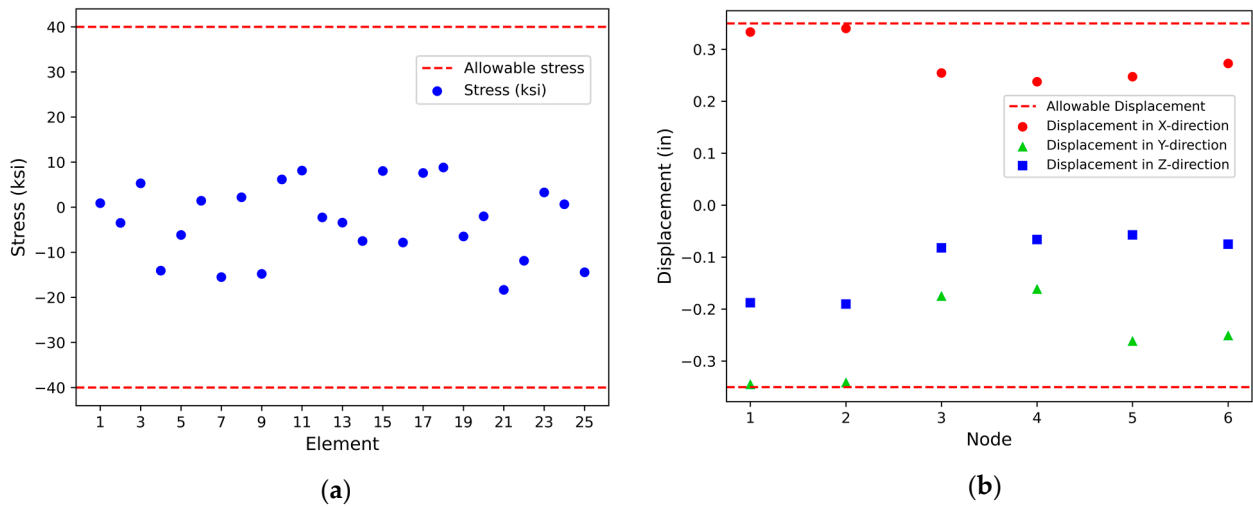


Figure 20. Example 4: constraint satisfaction of (a) permissible member stresses and (b) limited displacements at nodes 1 to 6.

Table 9. Example 4: optimal size and shape design solutions.

Variables	IGA [37]	GA [30]	FA [31]	PSO [29]	D-ICDE [13]	GLS-ECLPSO (Present)
A1	0.1	0.1	0.1	0.1	0.1	0.1
A2	0.1	0.1	0.1	0.1	0.1	0.1
A3	1.1	1.1	0.9	1.1	0.9	0.9
A4	0.1	0.1	0.1	0.1	0.1	0.1
A5	0.1	0.1	0.1	0.4	0.1	0.1
A6	0.2	0.1	0.1	0.1	0.1	0.1
A7	0.2	0.2	0.1	0.4	0.1	0.1
A8	0.7	0.8	1	0.7	1	1
X4	35.47	33.0487	37.32	27.6169	36.83	37.200
Y4	60.37	53.5663	55.74	51.6196	58.53	61.438
Z4	129.07	129.9092	126.62	129.9071	122.67	122.07
X8	45.06	43.7826	50.14	42.5526	49.21	50.270
Y8	137.04	136.8381	136.40	132.7241	136.74	140
Best weight (lb)	124.943	120.115	118.83	129.207	118.76	118.045
Constraint violation (%)	0.0	0.0	N/A	0.0	0.0	0.0
No. of analyses	6000	10,000	6000	4500	6000	6000
SD	N/A	N/A	5.5	N/A	N/A	4.2

Note: GA = genetic algorithm; IGA = improved genetic algorithm; FA = firefly algorithm; PSO = particle swarm optimization; D-ICDE = discrete improved constrained differential evolution.

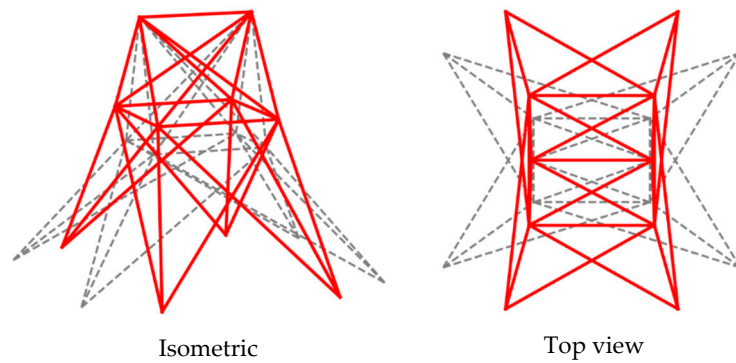
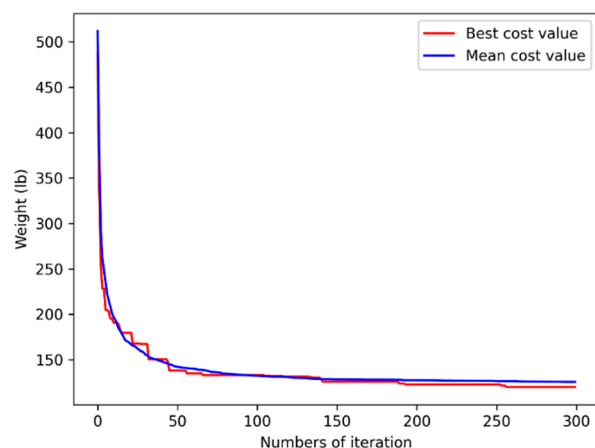


Figure 21. Example 4: optimal shape solution.



**Figure 22.** Example 4: solution convergence.

### 6.5. Discussions of the Results

From all the successfully solved design examples, we discuss three main findings.

(i) The proposed GLS-ECLPSO method incorporated various enhanced search techniques especially designed for different purposes. The perturbation-based local searches enhanced the standard CLPSO using normative information to perform robust exploitative searches (i.e., updating the particle velocity) over the appropriate locations within a feasible domain. The adaptive learning probability enabled dynamic adjustment from the ranking of personable best particles, improving the capability to eliminate the poor exploitation of search spaces. The GLS scheme generated samples in a Gaussian distribution around the global best particle, and thus enhanced explorative searches. A combination of these techniques provided an enhanced likelihood of obtaining an accurate optimal design for simultaneous size and shape optimization of the problems considered, which were casted as challenging convex and/or non-smooth programs.

(ii) The optimal solutions computed by the proposed GLS-ECLPSO are summarized in Table 10, where the lowest minimum weights reported in the literature [13,29–37] are collected. The superior performance of our proposed design method is shown. The lowest minimum weights (indicating that it was the most optimal solution) were obtained by the GLS-ECLPSO approach, compared with the best results found in the literature, namely D-ICDE [13], SCPSO [29], CA-ICEA [36], and D-ICDE [13] for Examples 1 through 4, respectively. The associated member stresses (buckling forces) and nodal displacements strictly complied with the imposed limits, and hence reported zero percent constraint violation. Moreover, the statistical values collected from the 25 independent designs solved by the proposed method (see Tables 3, 5, 7 and 9) showed a small standard deviation for all the examples processed, indicating high reliability of the GLS-ECLPSO approach in repeatedly determining accurate optimal designs for the class of structures considered, being insensitive to the underlying population-based random parameters.

**Table 10.** Result summary for least minimum weight designs.

Example	No. of Variables	GLS-ECLPSO (lb)	No. of Analyses	Reference Method (lb)	No. of Analyses	Reference
1	23	74.172	6000	74.682	8000	D-ICDE [13]
2	12	4175.142	6000	4512.365	4500	SCPSO [29]
3	44	1799.875	30,000	1844.71	5016	CA-ICEA [36]
4	13	118.045	6000	118.76	6000	D-ICDE [13]

(iii) Whilst a fast convergence to minimum weight solutions (viz., for both best and mean values) was presented in Figures 10, 14, 18 and 22, the proposed GLS-ECLPSO method involved slightly greater numbers of analyses compared with methods cited in the

literature. It processed enhanced search strategies, including perturbation exploitative and Gaussian explorative searching schemes. Similar to standard population-based algorithms, the solutions of large-scale problems necessarily require a special high-performance parallel computing framework with sufficient memory storage. In contrast to this, the presence of some machine learning maps out the predictive models of design structures and bypasses the need to iteratively call finite element implementations within a time-consuming meta-heuristic algorithm. This could be a promising solution to robustly handle large-scale optimization problems. This technique is currently under investigation and is an area for future research.

## 7. Concluding Remarks

This paper presents an efficient meta-heuristic GLS-ECLPSO method that reliably performs optimal size and shape design of truss structures in two- and three-dimensional spaces. The governing formulation is cast as a non-convex and/or non-smooth optimization problem in view of the presence of discrete area variables and permissible stress and/or limited displacement conditions. In essence, two main phases underpinning the proposed approach have been incorporated to enhance a good balance between the exploration and exploitation of search spaces.

The first phase enhances the trajectories of particles in the design space using the information across personal best particles, whilst the comprehensive learning scheme enables cross-positions between the best swarm particles in each dimension. The two perturbation-based exploitation (*viz.*, normative knowledge on bounds of the personal best positions) and adaptive learning probability functions are encoded to achieve good exploitative search ability and fast convergence to the optimal solution. The second phase applies the GLS strategy that generates random (following a Gaussian distribution) particles enveloping the global best particle to test its optimality, and hence, an accurate exploitation procedure. This process avoids premature convergence to local optima.

Four size and shape optimization examples illustrate the accurate optimal design solutions using the proposed method, validating the good performance of the proposed GLS-ECLPSO method. The results are consistent with those reported in the literature. The standard deviation collected from various independent runs presents a small variance. This indicates the reliability of the proposed design method that can achieve optimal solutions, whilst being insensitive to population-based random parameters.

**Author Contributions:** T.H.V.: methodology, software, validation, writing—original draft preparation; S.T.: conceptualization, methodology, resources, writing—original draft preparation, supervision, and funding acquisition; S.M.: formal analysis, investigation, and data curation; P.T.V.: visualization. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Thailand Science Research and Innovation Fund, Chulalongkorn University (IND66210025). We also acknowledge the support from Chulalongkorn University under the Ratchadaphiseksomphot Endowment Fund and the Second Century Fund.

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author.

**Acknowledgments:** The authors would like to sincerely thank the reviewers for their careful reviews and constructive comments on the earlier version of the manuscript.

**Conflicts of Interest:** The authors declare that 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. Haftka, R.T.; Grandhi, R.V. Structural shape optimization—A survey. *Comput. Methods Appl. Mech. Eng.* **1986**, *57*, 91–106. [[CrossRef](#)]
2. Wang, D.; Zhang, W.; Jiang, J. Truss shape optimization with multiple displacement constraints. *Comput. Methods Appl. Mech. Eng.* **2002**, *191*, 3597–3612. [[CrossRef](#)]



3. Hsu, Y.-L. A review of structural shape optimization. *Comput. Ind.* **1994**, *25*, 3–13. [[CrossRef](#)]
4. Salajegheh, E.; Vanderplaats, G.N. Optimum design of trusses with discrete sizing and shape variables. *Struct. Multidiscip. Optim.* **1993**, *6*, 79–85. [[CrossRef](#)]
5. Wang, D.; Zhang, W.; Jiang, J. Truss optimization on shape and sizing with frequency constraints. *AIAA J.* **2004**, *42*, 622–630. [[CrossRef](#)]
6. Rao, S.S. *Engineering Optimization: Theory and Practice*; John Wiley & Sons: Hoboken, NJ, USA, 2019.
7. Kochenderfer, M.J.; Wheeler, T.A. *Algorithms for Optimization*; MIT Press: Cambridge, MA, USA, 2019.
8. Martins, J.R.; Ning, A. *Engineering Design Optimization*; Cambridge University Press: Cambridge, UK, 2021.
9. Wu, S.-J.; Chow, P.-T. Integrated discrete and configuration optimization of trusses using genetic algorithms. *Comput. Struct.* **1995**, *55*, 695–702. [[CrossRef](#)]
10. Soh, C.K.; Yang, J. Optimal layout of bridge trusses by genetic algorithms. *Comput. Aided Civ. Infrastruct. Eng.* **1998**, *13*, 247–254. [[CrossRef](#)]
11. Kaveh, A.; Talatahari, S. An enhanced charged system search for configuration optimization using the concept of fields of forces. *Struct. Multidiscip. Optim.* **2011**, *43*, 339–351. [[CrossRef](#)]
12. Miguel, L.F.F.; Miguel, L.F.F. Shape and size optimization of truss structures considering dynamic constraints through modern metaheuristic algorithms. *Expert Syst. Appl.* **2012**, *39*, 9458–9467. [[CrossRef](#)]
13. Ho-Huu, V.; Nguyen-Thoi, T.; Nguyen-Thoi, M.; Le-Anh, L. An improved constrained differential evolution using discrete variables (D-ICDE) for layout optimization of truss structures. *Expert Syst. Appl.* **2015**, *42*, 7057–7069. [[CrossRef](#)]
14. Azad, S.K.; Bybordiani, M.; Azad, S.K.; Jawad, F.K. Simultaneous size and geometry optimization of steel trusses under dynamic excitations. *Struct. Multidiscip. Optim.* **2018**, *58*, 2545–2563. [[CrossRef](#)]
15. Ho-Huu, V.; Nguyen-Thoi, T.; Truong-Khac, T.; Le-Anh, L.; Vo-Duy, T. An improved differential evolution based on roulette wheel selection for shape and size optimization of truss structures with frequency constraints. *Neural Comput. Appl.* **2018**, *29*, 167–185. [[CrossRef](#)]
16. Nguyen-Van, S.; Nguyen, K.T.; Luong, V.H.; Lee, S.; Lieu, Q.X. A novel hybrid differential evolution and symbiotic organisms search algorithm for size and shape optimization of truss structures under multiple frequency constraints. *Expert Syst. Appl.* **2021**, *184*, 115534. [[CrossRef](#)]
17. Luo, R.; Wang, Y.; Xiao, W.; Zhao, X. AlphaTruss: Monte Carlo Tree Search for Optimal Truss Layout Design. *Buildings* **2022**, *12*, 641. [[CrossRef](#)]
18. Hussain, K.; Salleh, M.N.M.; Cheng, S.; Shi, Y. Metaheuristic research: A comprehensive survey. *Artif. Intell. Rev.* **2019**, *52*, 2191–2233. [[CrossRef](#)]
19. Yang, X.-S. *Nature-Inspired Metaheuristic Algorithms*; Luniver Press: Cambridge, UK, 2010.
20. Hussain, K.; Salleh, M.N.M.; Cheng, S.; Shi, Y. On the exploration and exploitation in popular swarm-based metaheuristic algorithms. *Neural Comput. Appl.* **2019**, *31*, 7665–7683. [[CrossRef](#)]
21. Morales-Castañeda, B.; Zaldivar, D.; Cuevas, E.; Fausto, F.; Rodríguez, A. A better balance in metaheuristic algorithms: Does it exist? *Swarm Evol. Comput.* **2020**, *54*, 100671. [[CrossRef](#)]
22. Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. [[CrossRef](#)]
23. Eberhart, R.; Kennedy, J. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Network, Perth, Australia, 27 November–1 December 1995; pp. 1942–1948.
24. Van, T.H.; Tangaramvong, S.; Limkatanyu, S.; Xuan, H.N. Two-phase ESO and comprehensive learning PSO method for structural optimization with discrete steel sections. *Adv. Eng. Softw.* **2022**, *167*, 103102. [[CrossRef](#)]
25. Liang, J.J.; Qin, A.K.; Suganthan, P.N.; Baskar, S. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Trans. Evol. Comput.* **2006**, *10*, 281–295. [[CrossRef](#)]
26. Yu, X.; Zhang, X. Enhanced comprehensive learning particle swarm optimization. *Appl. Math. Comput.* **2014**, *242*, 265–276. [[CrossRef](#)]
27. Tangaramvong, S.; Tin-Loi, F.; Gao, W. Optimal retrofit of moment resisting frames using braces accounting for geometric nonlinearity and serviceability conditions. *Eng. Struct.* **2014**, *80*, 189–199. [[CrossRef](#)]
28. Tangaramvong, S.; Tin-Loi, F. Optimal performance-based rehabilitation of steel frames using braces. *J. Struct. Eng.* **2015**, *141*, 04015015. [[CrossRef](#)]
29. Gholizadeh, S. Layout optimization of truss structures by hybridizing cellular automata and particle swarm optimization. *Comput. Struct.* **2013**, *125*, 86–99. [[CrossRef](#)]
30. Rahami, H.; Kaveh, A.; Gholipour, Y. Sizing, geometry and topology optimization of trusses via force method and genetic algorithm. *Eng. Struct.* **2008**, *30*, 2360–2369. [[CrossRef](#)]
31. Miguel, L.F.F.; Lopez, R.H.; Miguel, L.F.F. Multimodal size, shape, and topology optimisation of truss structures using the Firefly algorithm. *Adv. Eng. Softw.* **2013**, *56*, 23–37. [[CrossRef](#)]
32. Dede, T.; Ayvaz, Y. Combined size and shape optimization of structures with a new meta-heuristic algorithm. *Appl. Soft Comput.* **2015**, *28*, 250–258. [[CrossRef](#)]
33. Jawad, F.K.J.; Ozturk, C.; Dansheng, W.; Mahmood, M.; Al-Azzawi, O.; Al-Jemely, A. Sizing and layout optimization of truss structures with artificial bee colony algorithm. *Structures* **2021**, *30*, 546–559. [[CrossRef](#)]

34. Hasançebi, O.; Erbatur, F. On efficient use of simulated annealing in complex structural optimization problems. *Acta Mech.* **2002**, *157*, 27–50. [[CrossRef](#)]
35. Kaveh, A.; Zaerreza, A. Size/Layout Optimization of Truss Structures Using Shuffled Shepherd Optimization Method. *Period. Polytech. Civ. Eng.* **2020**, *64*, 408–421. [[CrossRef](#)]
36. Dehghani, M.; Mashayekhi, M.; Sharifi, M. An efficient imperialist competitive algorithm with likelihood assimilation for topology, shape and sizing optimization of truss structures. *Appl. Math. Model.* **2021**, *93*, 1–27. [[CrossRef](#)]
37. Tang, W.; Tong, L.; Gu, Y. Improved genetic algorithm for design optimization of truss structures with sizing, shape and topology variables. *Int. J. Numer. Methods Eng.* **2005**, *62*, 1737–1762. [[CrossRef](#)]