INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING
Int. J. Optim. Civil Eng., 2020; 10(4):669-700

SET THEORETICAL VARIANTS OF OPTIMIZATION ALGORITHMS FOR OPTIMAL DESIGN OF SKELETAL STRUCTURES

A. Kaveh[*, †] and K. Biabani Hamedani

School of Civil Engineering, Iran University of Science and Technology, Tehran, Iran

ABSTRACT

In this paper, set theoretical variants of the artificial bee colony (ABC) and water evaporation optmization (WEO) algorithms are proposed. The set theoretical variants are designed based on a set theoretical framework in which the population of candidate solutions is divided into some number of smaller well-arranged sub-populations. The framework aims to improve the compromise between diversification and intensification of the search and makes it possible to design various variants of a P-metaheuristic. In order to verify the stability and robustness of the set theoretical framework, the proposed algorithms are applied to solve three different benchmark structural design optimization problems. The results show that the set theoretical framework improves the performance of the ABC and WEO algorithms, especially in terms of robustness and convergence characteristics.

Keywords: structural optimization; truss structures; frame structures; population-based metaheuristics; set theory; artificial bee colony algorithm; water evaporation optimization algorithm.

Received: 12 August 2020; Accepted:15 November 2020

1. INTRODUCTION

Set theory is a branch of mathematics that studies the sets and theirs properties. A set can be defined as a well-defined collection of objects called elements or members [1]. The elements of a set may or may not have mathematical nature. Georg Cantor, one of the creators of the set theory, defined a set as follows [2]: “A set is a gathering together into a whole of definite, distinct objects of our perception or of our thought, which are called elements of the set.” The language of set theory can be used to represent almost any mathematical concept.

[*] Corresponding author: School of Civil Engineering, Iran University of Science and Technology, Narmak, Tehran, P.O. Box 16846-13114, Iran
[†]E-mail address: alikaveh@iust.ac.ir (A. Kaveh)

This capability can be regarded as one of the most important achievements of mathematics. Accordingly, it can be said that the set theory provides a foundation for mathematics [3]. Set theory has found many applications in engineering. As an example in structural engineering, Behravesh et al. used set theory for configuration processing [4]. Kaveh [5] employed a set of contours and its transversal in nodal ordering for bandwidth reduction, and Kaveh et al. [6] employed set theory concepts for modeling the shuffled shepherd optimization algorithm.

Metaheuristics can be classified based on many different criteria, the most common of which is population-based search versus single-solution based search. Single-solution based algorithms manipulate and move a single candidate solution in the search space, whereas P-metaheuristics start their own search process with a set of candidate solutions, known as initial population. Considering this point, the initial population of P-metaheuristics could be viewed as a set with a certain number of elements. In a similar manner, a P-metaheuristics could be viewed as an iterative improvement of a set of elements. In these algorithms, an iterative search process continues by a set of candidate solutions until a pre-defined stopping criterion is fulfilled. At the first step, the initial population is initialized. Different strategies can be used to generate the initial population, such as random generation, sequential diversification, parallel diversification, heuristic initialization, etc. [7]. Random generation is the most common strategy for generation of the initial population. Next, based on higher level strategies acting on the current set of candidate solutions, a new set of candidate solutions is generated. Using some specific procedures known as replacement strategies, the current set of candidate solutions is replaced with the set of new ones. In other words, the current set of candidate solutions is updated. This process is repeated until a pre-defined stopping criterion is satisfied. The most important point in designing a metaheuristic is to keep a good balance between two conflicting aspects of exploration (global search or diversification) and exploitation (local search or intensification), which is an uneasy task [8]. Exploration means the ability of the metaheuristic to search entirely unknown regions of the search space, whereas exploitation means the ability of the metaheuristic to search the neighborhood of promising regions of the search space which have been visited formerly. Recently, Kaveh et al. [9] proposed a general set theoretical framework for population-based metaheuristics (P-metaheuristics) with the aim of improving the compromise between diversification and intensification of the search. The main idea of the framework is to devide the population of candidate solutions into a number of smaller well-arranged sub-populations through which the search process is done. The framework, which is applicable to almost all the P-metaheuristics, makes it possible to design various variants of a P-metaheuristic.

In this study, the set theoretical framework is employed and set theoretical variants of the artificial bee colony (ABC) and water evaporation optmization (WEO) algorithms are developed. The ABC algorithm is a P-metaheuristic which was developed by Karaboga in 2005 [10] based on the foraging behavior of honey bees. The set theoretical variants of the ABC algorithm are named as OST-ABC and STMP-ABC, which are the acronyms of “ordered set theoretical artificial bee colony” and “set theoretical multi-phase artificial bee colony”, respectively. The WEO algorithm was developed by Kaveh and Bakhshpoori [11] in 2016 inspired by evaporation of a tiny amount of water molecules on the solid surface with different wettability. In a similar way, OST-WEO and STMP-WEO are the acronyms of “ordered set theoretical water evaporation optmization” and “set theoretical multi-phase water evaporation optmization”, respectively.

In order to provide evidence to support the applicability and efficiency of the set theoretical framework, the developed algorithms are tested on three different benchmark structural design optimization problems, including continuous optimization of dome-like truss structures with multiple natural frequency constraints, and discrete size optimization of planar steel frame structures with strength and displacement constraints. The reason for choosing frequency-constrained optimization problems is due to their highly nonlinear, non-convex, and discontinuous search spaces with several local optima [12]. The most common problem with frequency-constrained optimization seems to be the high sensitivity of vibration modes to shape modifications, which means that vibration modes can switch during the optimization process. This can lead to significant changes in natural frequencies, which causes convergence difficulties [13]. In the past few decades, frequency-constrained optimization problems have attracted the attention of many researchers. Bellagamba and Yang [14] were one of the first researchers to study the minimum-mass design of truss structures with natural frequency constraints. Grandhi and Venkayyat [15] presented a design optimization algorithm for structural weight minimization with multiple frequency constraints. Tong and Liu [16] presented an optimization procedure for the minimum weight optimization of truss structures subjected to constraints on stresses, natural frequencies and frequency responses. Sedaghati et al. [17] compared the performance of the displacement and force methods to optimize truss and beam structures with frequency constraints. Lingyun et al. [18] used an enhanced genetic algorithm to solve the truss shape and sizing optimization problems. Gomes [19] investigated the performance of a particle swarm optimization (PSO) algorithm in the field of truss optimization with frequency constraints. In a similar work, Miguel and Fadel Miguel [20] used harmony search (HS) and firefly algorithm (FA) to solve truss shape and size optimization with frequency constraints. Kaveh and Javadi [21] performed size and shape optimization of truss structures using an efficient hybrid algorithm harmony search (HS), ray optimization (RO), and particle swarm optimization (PSO). Kaveh and Ilchi Ghazaan [22] utilized an improved ray optimization algorithm to the optimum design of truss structures subjected to multiple frequency constraints. Kaveh and Ilchi Ghazaan [23] used vibrating particles system (VPS) algorithm for truss optimization with multiple natural frequency constraints. Kaveh and Zolghadr [24] presented a study where the cyclical parthenogenesis algorithm (CPA) was employed for layout optimization of truss structures with frequency constraints. Kaveh and Zolghadr [25] reviewed different different metaheuristic optimization techniques utilized to structural optimization problems with frequency constraints. Kaveh et al. [26] studied the performance of some metaheuristics in frequency-constrained truss optimization problems. Frame structures are among the most common structures in the structural engineering. In the past few decades, the problem of optimal design of steel frame structures has been studied by many researchers using different optimization methods: Pezeshk et al. [27] using genetic algorithm (GA), Camp et al. [28] using ant colony optimization (ACO), Degertekin [29] employing harmony search (HS), Kaveh and Talatahari [30] using charged system search (CSS), Zhou and Yang [31] using an improved genetic algorithm, Kaveh and Bakhshpoori [32] using cuckoo search (CS) algorithm, Maheri and Narimani [33] using an enhanced harmony search (EHS) algorithm, Talatahari et al. [34] using the eagle strategy with differential evolution (DE), Kaveh and Ilchi Ghazaan [35] using CBO and ECBO methods, Kazemzadeh Azad and Hasançebi [36] using guided stochastic search (GSS), Kaveh and Bakhshpoori [37] using an accelerated water evaporation optimization (AWEO), Kaveh and Ilchi Ghazaan [38] using vibrating particles system (VPS) algorithm, and Kaveh et al. [39] employing seven different population-based metaheuristics.

The rest of this paper is organized as follows: Section 2 is devoted to present set theoretical variants of the ABC and WEO algorithms. Furthermore, the optimization problems are defined briefly. In Section 3, the proposed algorithms are tested on three different benchmark structural design optimization problems. Finally, the last section concludes the paper.

2. MATERIALS AND METHODS

2.1 Set theoretical optimization algorithms

Recently, Kaveh et al. [9] employed the concepts of set theory and proposed a general set theoretical framework for population-based optimization algorithms. The main idea of the framework is to divide the population of candidate solutions into a number of smaller well-arranged sub-populations. In this paper, the set theoretical framework is applied to the artificial bee colony (ABC) and water evaporation optmization (WEO) algorithms. For each of the ABC and WEO algorithms, two different set theoretical variants are proposed: (1) ordered set theoretical (OST) variant; and (2) set theoretical multi-phase (STMP) variant. The OST variant is designed as follows: The initial population is divided into a number of smaller well-arranged sub-populations. Next, the search process starts through the sub-populations separately. It should be noted that the well-arranged sub-populations are reformed before the next iteration starts. The process continues until the termination criterion is satisfied. The STMP variant is a multi-phase version of the OST variant. In each phase of the STMP variant, a self-contained OST variant is executed with a specific number of sub-populations. Each phase of the STMP variant uses the output of the previous one as its initial population. In other words, the phases are executed in a sequence without any cooperation. The only difference between the phases of the STMP variant is the number of sub-populations. The number of sub-populations decreases in a few steps from the integer n_1 at the first phase to n_k at the last (e.g., k-th) phase. Obviously, the numbers n_1 and n_k must be chosen among the divisors of nE where nE represents the population size.

2.1.1. Set theoretical variants of the artificial bee colony algorithm

In the ABC algorithm, each candidate solution is represented by a food source. The food sources are modified by honey bees in a repeated manner with the aim of reaching food sources with better quality. In each iteration, the ABC algorithm searches in three sequential phases. Employed bees modify the food sources and share their information with onlooker bees. Onlooker bees select a food source based on the information from employed bees and try to modify it. Scout bees perform random searches in the vicinity of the hive. In the following, the three phases are formulated:

Employed bees phase: Generation of new honey bees ($newHB$) based on the recruited or employed bees strategy. This phase can be stated mathematically as follows:

$$newHB = HB + stepsize \quad (1)$$
$$sstepsize = rand_{(i)(j)} \times \left(HB - HB\left[permute_{(i)(j)}\right]\right) \quad (2)$$

where i is the number of honey bees; j is the number of design variables; $rand_{(i)(j)}$ is a random number chosen from the [-1, 1] interval; HB is the current set of honey bees; $newHB$ is the newly generated set of honey bees; $permute_{(i)(j)}$ is different rows permutation functions; and $stepsize$ is the step size of movement of the honey bees.

Onlooker bees phase: Generate new honey bees ($newHB$) based on the onlooker bees strategy. The onlooker bees phase can be stated mathematically as follows:

$$newHB = \begin{cases} HB_{rws} + stepsize, & \text{if } rand < mr \\ HB_{rws}, & \text{otherwise} \end{cases} \quad (3)$$
$$stepsize = rand_{(i)(j)} \times \left(HB_{rws} - HB\left[permute_{(i)(j)}\right]\right) \quad (4)$$

where HB_{rws} is food sources chosen by onlooker bees based on the roulette wheel selection scheme; and mr is a parameter that controls whether the selected food source by onlooker bee will be modified or not.

Scout bees phase: In the scout bee phase, employed bees who cannot modify their food sources after a specified number of trials (A) become scouts. The corresponding food source will be abandoned, and a random-based new food source will be generated in the vicinity of the hive.

2.1.1.1 Ordered set theoretical artificial bee colony (OST-ABC)

The OST-ABC algorithm is based on the idea of dividing the population of candidate solutions into a number of well-arranged sub-populations of the same size. The OST-ABC algorithm is stated in the following four steps:

Step one (initialization): The initial population is generated randomly.

Step two (forming the sub-populations): The sub-populations are formed based on a procedure proposed by Kaveh et al. [6] as follows: Let us consider an initial population containing nE solutions. The aim is to divide the initial population into a certain number of smaller well-arranged sub-populations (e.g., m) of the same size. To this end, the initial population is evaluated and the candidate solutions are sorted in ascending order of penalized objective function. In the first step of forming the sub-populations, the first m candidate solutions of the sorted population are chosen and each candidate solution is placed in one of the sub-populations randomly. In the second step, the next m candidate solutions are chosen and placed in the sub-populations randomly. The process continues until all candidate solutions of the sorted population are chosen and placed in the sub-populations. At the end of the last step, the sub-populations have an equal number of candidate solutions. If nS is the number of candidate solution of a sub-population, it can be said that:

$$nE = m \times nS \quad (5)$$

Step three (main body of ABC): The main body of the ABC algorithm is performed for all the sub-populations separately. In other words, three phases of the ABC algorithm are executed once for all the sub-populations separately. Therefore, m sub-population, each of them containing nS new candidate solutions, are generated. Next, the new sub-populations collectively form the population of new candidate solutions.

Step four (termination criterion): If the termination criterion of the algorithm is satisfied, the algorithm is terminated; otherwise, the algorithm returns to step two. The maximum number of objective function evaluations ($MaxNFEs$) is considered as the termination criterion of the algorithms.

The pseudo code of the OST-ABC algorithm is provided as follows:

Initialization:

- Set the algorithm parameters: mr, A, nE, m, and $MaxNFEs$.
- Generate random initial population of food sources (HB) and evaluate them.
- $NFEs = 0$;

Cyclic body of the algorithm:

While $NFEs < MaxNFEs$

- Sort the initial population in ascending order of penalized objective function.
- Form the sub-populations based on the procedure described in the second step of the OST-ABC algorithm.
- Generate the sub-populations of new food sources ($newHB$) based on the employed bees strategy using Eqs. (1) and (2).
- Evaluate the newly generated food sources ($newHB$) and apply replacement strategy between old and new food sources.
- Update the number of objective function evaluations ($NFEs = NFEs + nE$).
- Generate the sub-populations of new food sources ($newHB$) based on the onlooker bees strategy using Eqs. (3) and (4).
- Evaluate the newly generated food sources ($newHB$) and apply replacement strategy between old and new food sources.
- Update the number of objective function evaluations ($NFEs = NFEs + nE$).
- Discard each food sources if there is no improvement after the A number of objective function evaluations. Employ scout bees to generate randomly food sources in the vicinity of the hive and then evaluate them.
- Update the number of objective function evaluations.
- Collect the sub-populations and form the unified populations of food sources.
- Monitor the best food source found by the OST-ABC algorithm so far.

End While

The steps of the OST-ABC algorithm is shown in Fig. 1.

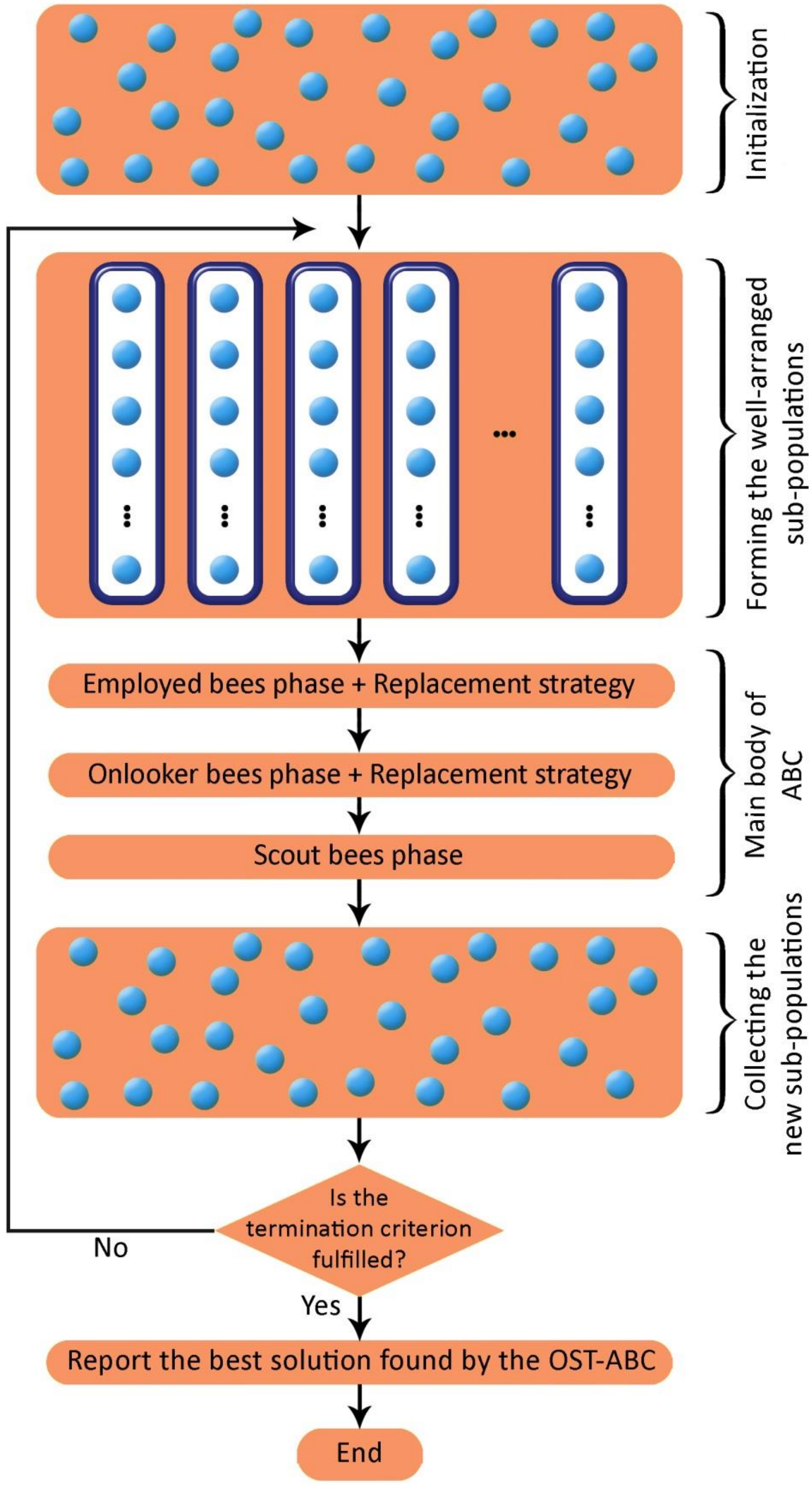

Figure 1. Steps of the OST-ABC algorithm

2.1.1.2 Set theoretical multi-phase artificial bee colony (STMP-ABC)

The STMP-ABC algorithm is a multi-phase version of the OST-ABC algorithm. In each phase of the STMP-ABC algorithm, a self-contained OST-ABC algorithm is executed with a specific number of sub-populations. Each phase uses the output of the previous one as its initial population. In other words, the phases are executed in a sequence without any cooperation. Each phase continues until the number of objective function evaluations ($NFEs$) reaches its predefined limit. The phases differ in the number of sub-populations they work with. The number of sub-populations varies in a decreasing order. The pseudo code of the STMP-ABC algorithm is provided as follows:

Initialization:

- Set the algorithm parameters: mr, A, nE, m, k, and $MaxNFEs$.
- Generate random initial population of food sources (HB) and evaluate them.
- Specify the number of sub-populations in each phase ($n_1, n_2, \cdots, n_k$).
- $i = 0$; $NFEs = 0$;

Cyclic body of the algorithm:

While $i < k$

- $i = i + 1$;

While $NFEs < i \times MaxNFEs/k$

- Sort the initial population in ascending order of penalized objective function.
- Form the sub-populations of i-th phase (n_i sub-populations) based on the procedure described in the second step of the OST-ABC algorithm.
- Generate the sub-populations of new food sources ($newHB$) based on the employed bees strategy using Eqs. (1) and (2).
- Evaluate the newly generated food sources ($newHB$) and apply replacement strategy between old and new food sources.
- Update the number of objective function evaluations ($NFEs = NFEs + nE$).
- Generate the sub-populations of new food sources ($newHB$) based on the onlooker bees strategy using Eqs. (3) and (4).
- Evaluate the newly generated food sources ($newHB$) and apply replacement strategy between old and new food sources.
- Update the number of objective function evaluations ($NFEs = NFEs + nE$).
- Discard each food sources if there is no improvement after the A number of objective function evaluations. Employ scout bees to generate randomly food sources in the vicinity of the hive and then evaluate them.
- Update the number of objective function evaluations.
- Collect the sub-populations and form the unified populations of food sources.
- Monitor the best food source found by the STMP-ABC algorithm so far.

End While

End While

The steps of the STMP-ABC algorithm is shown in Fig. 2.

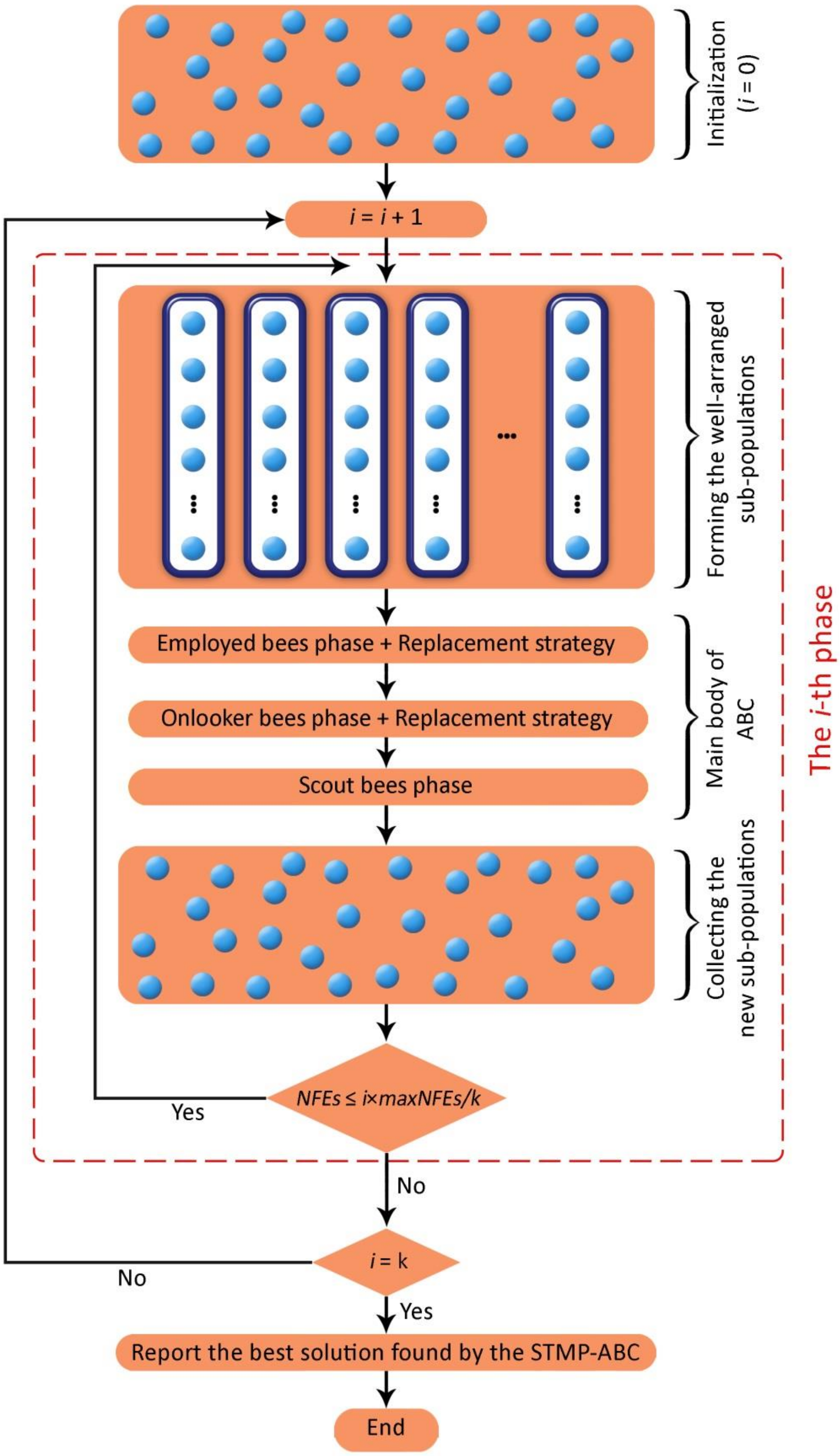

Figure 2. Steps of the STMP-ABC algorithm

2.1.2 Set theoretical variants of the water evaporation optimization algorithm

In the WEO algorithm, each candidate solution is represented by a water molecule. Solid surface or substrate with variable wettability is reflected as the search space. Decreasing the surface wettability (substrate changed from hydrophility to hydrophobicity) reforms the water aggregation from a monolayer to a sessile droplet. Decreasing wettability of the surface can represent the decrease of objective function for a minimizing optimization problem. Evaporation flux rate of the water molecules is considered as the most appropriate measure for updating the individuals which its pattern of change is in good agreement with the local and global search ability of the algorithm. The WEO algorithm searches in two sequential phases of monolayer evaporation and droplet evaporation. In the first half of the iterations ($NFEs \leq MaxNFEs/2$), the water molecules are updated based on the monolayer evaporation strategy, while, within the second half of the iterations ($NFEs > MaxNFEs/2$), the water molecules are updated based on the droplet evaporation strategy. In the following, the two phases are formulated:

Monolayer evaporation phase: Generation of new water molecules ($newWM$) based on the monolayer evaporation strategy. This phase can be stated mathematically as follows:

$$stepsize = rand \times \left(WM\left[permute1_{(i)(j)}\right] - WM\left[permute2_{(i)(j)}\right]\right) \quad (6)$$

$$nexWM = WM + stepsize \times MEP \quad (7)$$

$$MEP_{ij} = \begin{cases} 1, & \text{if } rand_{ij} < exp\left(E_{sub}(i)\right) \\ 0, & \text{if } rand_{ij} \geq exp\left(E_{sub}(i)\right) \end{cases} \quad (8)$$

$$E_{sub}(i) = \frac{3 \times (PFit_i - \min(PFit))}{(\max(PFit) - \min(PFit))} - 3.5 \quad (9)$$

where i is the number of water molecules; j is the number of design variables; $rand_{(i)(j)}$ is a random number chosen from the [-1, 1] interval; WM is the current set of water molecules; $newWM$ is the newly generated set of water molecules; $permute1_{(i)(j)}$ and $permute1_{(i)(j)}$ are different rows permutation functions; $stepsize$ is the step size of movement of the water molecules; $PFit$ is the penalized objective function vector of the current set of water molecules; MEP is the monolayer evaporation probability matrix; and E_{sub} is the subtrate interaction energy vector.

Droplet evaporation phase: Generation of new water molecules ($newWM$) based on the droplet evaporation strategy. This phase can be stated mathematically as follows:

$$nexWM = WM + stepsize \times DEP \quad (10)$$

$$stepsize = rand \times \left(WM\left[permute1_{(i)(j)}\right] - WM\left[permute2_{(i)(j)}\right]\right) \quad (11)$$

$$DEP_{ij} = \begin{cases} 1, & \text{if } rand_{ij} < J(\theta_i) \\ 0, & \text{if } rand_{ij} \geq J(\theta_i) \end{cases} \quad (12)$$

$$\theta_i = \frac{30 \times (PFit_i - \min(PFit))}{(\max(PFit) - \min(PFit))} - 50 \quad (13)$$

$$J(\theta) = \frac{1}{2.6} \times \left(\frac{2}{3} + \frac{(\cos(\theta))^3}{3} - \cos(\theta)\right)^{-2/3} \times (1 - \cos(\theta)) \quad (14)$$

where DEP is the droplet evaporation probability matrix; $J(\theta)$ is the evaporation flux vector; and θ is the contact angle of the water droplet.

2.1.2.1 Ordered set theoretical water evaporation optimization (OST-WEO)

Similar to the OST-ABC algorithm, the OST-WEO algorithm is based on the idea of dividing the population of candidate solutions into a number of well-arranged sub-populations of the same size. The OST-WEO algorithm is stated in the following four steps:

Step one (initialization): The initial population is generated randomly.

Step two (forming the sub-populations): The sub-populations are formed based on the procedure described in the second step of the OST-ABC algorithm.

Step three (main body of WEO): The main body of the WEO algorithm is performed for all the sub-populations separately. Therefore, m sub-population, each of them containing nS new candidate solutions, are generated. Next, the new sub-populations collectively form the population of new candidate solutions.

Step four (termination criterion): If the termination criterion of the algorithm is satisfied, the algorithm is terminated; otherwise, the algorithm returns to step two. The maximum number of objective function evaluations ($MaxNFEs$) is considered as the termination criterion of the algorithms.

The pseudo code of the OST-WEO algorithm is provided as follows:

Initialization:

- Set the algorithm parameters: nE, m, and $MaxNFEs$.
- Generate random initial population of water molecules (WM) and evaluate them.
- $NFEs = 0$;

Cyclic body of the algorithm:

While $NFEs < MaxNFEs$

- Sort the initial population in ascending order of penalized objective function.
- Form the sub-populations based on the procedure described in the second step of the OST-ABC algorithm.

If $NFEs \leq MaxNFEs/2$

- Generate the sub-populations of new water molecules ($newWM$) based on the monolayer evaporation strategy using Eqs. (6) to (9).
- Evaluate the newly generated water molecules ($newWM$) and apply replacement strategy between old and new water molecules.
- Update the number of objective function evaluations ($NFEs = NFEs + nE$).

else

- Generate the sub-populations of new water molecules ($newWM$) based on the droplet evaporation strategy using Eqs. (10) to (14).
- Evaluate the newly generated water molecules ($newWM$) and apply replacement strategy between old and new water molecules.
- Update the number of objective function evaluations ($NFEs = NFEs + nE$).

End if

- Collect the sub-populations and form the unified populations of water molecules.
- Monitor the best water molecule found by the OST-WEO algorithm so far.

End While

The steps of the OST-WEO algorithm is shown in Fig. 3.

2.1.2.2 Set theoretical multi-phase water evaporation optimization (STMP-WEO)

Similar to the STMP-ABC algorithm, the STMP-WEO is a multi-phase version of the OST-WEO algorithm. The pseudo code of the STMP-WEO algorithm is provided as follows:

Initialization:

- Set the algorithm parameters: nE, m, k, and $MaxNFEs$.
- Generate random initial population of water molecules (WM) and evaluate them.
- Specify the number of sub-populations in each phase ($n_1, n_2, \cdots, n_k$).
- $i = 0$; $NFEs = 0$;

Cyclic body of the algorithm:

While $i < k$

- $i = i + 1$;

While $NFEs < i \times MaxNFEs/k$

- Sort the initial population in ascending order of penalized objective function.
- Form the sub-populations of i-th phase (n_i sub-populations) based on the procedure described in the second step of the OST-ABC algorithm.

If $NFEs \leq MaxNFEs/2$

- Generate the sub-populations of new water molecules ($newWM$) based on the monolayer evaporation strategy using Eqs. (6) to (9).
- Evaluate the newly generated water molecules ($newWM$) and apply replacement strategy between old and new water molecules.
- Update the number of objective function evaluations ($NFEs = NFEs + nE$).

else

- Generate the sub-populations of new water molecules ($newWM$) based on the droplet evaporation strategy using Eqs. (10) to (14).
- Evaluate the newly generated water molecules ($newWM$) and apply replacement strategy between old and new water molecules.
- Update the number of objective function evaluations ($NFEs = NFEs + nE$).

End if

- Collect the sub-populations and form the unified populations of water molecules.
- Monitor the best water molecule found by the STMP-WEO algorithm so far.

End While

End While

The steps of the STMP-WEO algorithm is shown in Fig. 4.

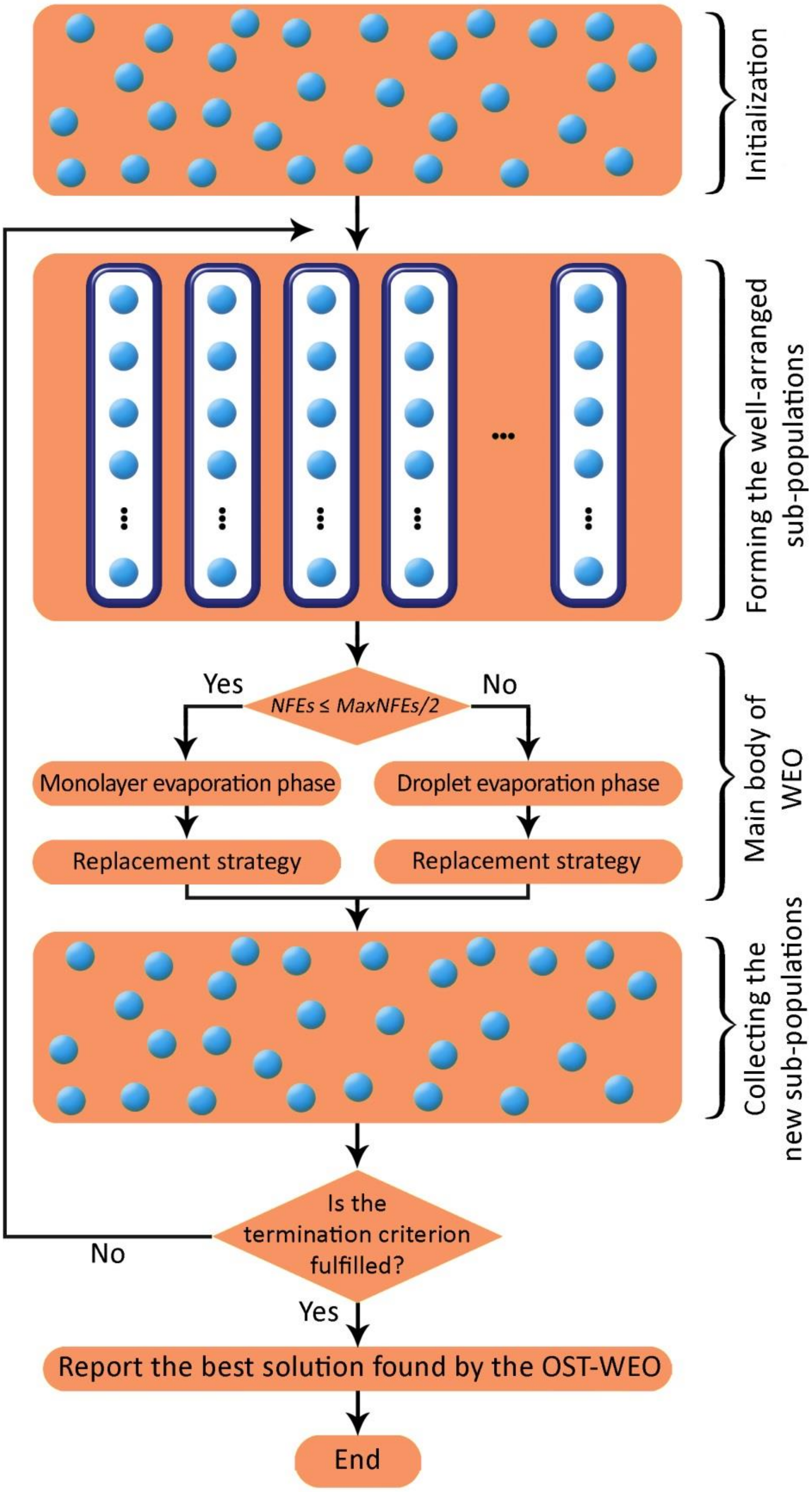

Figure 3. Steps of the OST-WEO algorithm

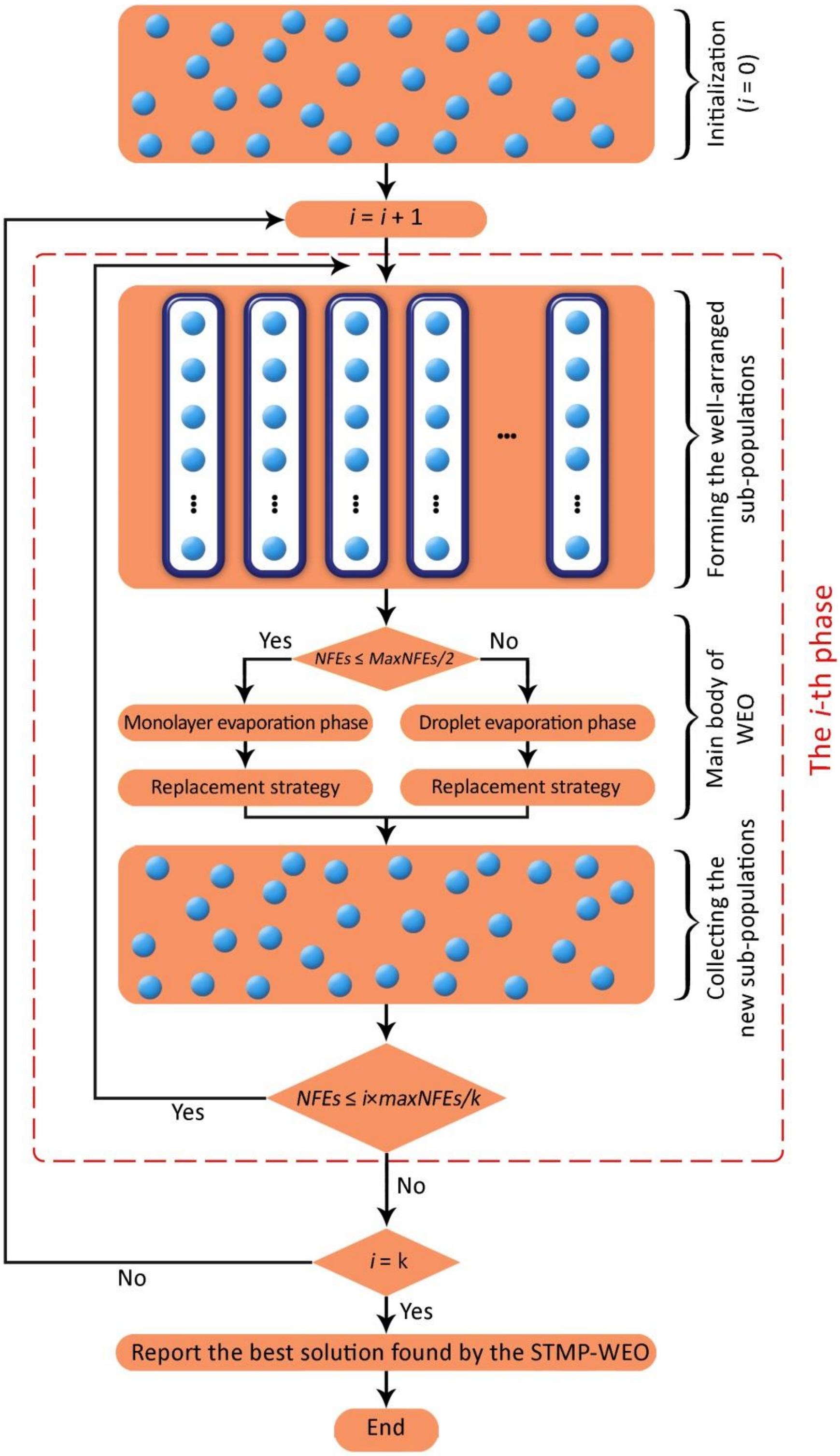

Figure 4. Steps of the STMP-WEO algorithm

2.2 Definition of the optimization problems

In order to investigate the performance of the set theoretical framework, the proposed algorithms are applied to solve three different structural optimization problems, including continuous optimization of dome-like truss structures with multiple frequency constraints and discrete size optimization of steel frame structures with strength and displacement constraints. The frame structures are designed based on the requirements of AISC load and resistance factor design (LRFD) specification for structural steel buildings [40]. The optimization problems are formulated mathematically as follows:

2.2.1 Optimization of truss structures with frequency constraints

Truss optimization problems with multiple frequency constraints are taken into account as complex problems. In such problems, the aim is to minimize the total weight of the truss structure while satisfying some constraints on natural frequencies. In a simultaneous size and shape truss optimization problem with continuous search space, the cross-section area of members and the coordinates of some nodes are considered as design variables, which can vary continuously in the search space. The topology of the structure is assumed to be fixed during the optimization process. The continuous truss optimization problem with frequency constraints can be defined mathematically as follows [9]:
Find

$$\{D\} = [d_1, d_2, \dots, d_{nD}], \qquad d_i \in S_i \tag{15}$$

to minimize

$$PFit(\{D\}) = W(\{D\}) + f_{penalty}(D) \tag{16}$$

where

$$W(\{D\}) = \sum_{i=1}^{nE} \rho_i \times A_i \times L_i \tag{17}$$

subject to

$$\begin{cases} d_i^L \le d_i \le d_i^U \\ \omega_k \le \omega_k^* \\ \omega_j \ge \omega_j^* \end{cases} \tag{18}$$

where $\{D\}$ is the candidate solution vector containing the set of design variables, including both cross-sectional areas and nodal coordinates; d_i is the i-th design variable; nD is the number of design variables; and S_i is the allowable range of i-th design variable. $W(\{D\})$, $f_{penalty}(D)$, and $PFit(\{D\})$ denote the objective function (total structural weight), penalty function, and penalized objective function of the candidate solution $\{D\}$, respectively. nM is the number of truss members; and ρ_i, A_i, and L_i are the material density, cross-sectional area, and length of the i-th truss element, respectively. d_i^L and d_i^U are the lower and upper bounds of the i-th design variable, respectively; ω_j and ω_k denote the j-th and k-th natural frequencies of the structure, respectively; ω_j^* is the lower bound of j-th natural frequency; and ω_k^* is the upper bound of k-th natural frequency. The design variable d_i can vary continuously within the range S_i. The range S_i can be stated as:

$$S_i = \{d_i | d_i \in [d_i^L, d_i^U]\} \tag{19}$$

The free vibration analysis of a structural system results in the eigenvalue problem given by the following equation [9]:

$$K\phi_i = \gamma_i M\phi_i \tag{20}$$

where K and M are the stiffness and mass matrices of the structural system, respectively; and ϕ_i is the i-th eigenvector of the structural system. The i-th period (T_i) and the i-th circular frequency (ω_i) are related to the i-th eigenvalue (γ_i) by the following equation:

$$\gamma_i = \omega_i^2 = (2\pi/T_i)^2 \tag{21}$$

In this research, one of the well-known constraint handling strategies, known as penalizing strategy, is employed to handle the problem constraints. In penalizing strategies, by using a penalty function, the infeasible solutions are penalized [41]. The penalty function is defined as follows:

$$f_{penalty}(D) = (1 + \varepsilon_1 \times \upsilon)^{\varepsilon_2};\ \upsilon = \sum_{i=1}^{nC} \upsilon_i \tag{22}$$

where nC denotes the number of constraints of the problem; υ is the sum of the violation of all constraints; and υ_i is the violation of the i-th constraint. In this research, ε_1 is set to unity. Also, ε_2 is calculated by Eq. (23). The violation of the i-th constraint (υ_i) can be obtained from Eq. (24). This formulation indicates that υ_i is set to zero if i-th constraint is satisfied.

$$\varepsilon_2 = 1.5(1 + \frac{iter}{MaxIter}) \tag{23}$$

$$\upsilon_i = \begin{cases} \left|1 - \frac{\omega_i}{\omega_i^*}\right|, & \text{the } i-\text{th constraint is violated} \\ 0, & \text{otherwise} \end{cases} \tag{24}$$

2.2.2 Discrete sizing optimization of steel frame structures

The problem of discrete sizing optimization of steel frame structures can be expressed mathematically as follows [39]:

Find

$$\{D\} = [d_1, d_2, \dots, d_{nD}], \qquad d_i \in R_i \tag{25}$$

to minimize

$$PFit(\{D\}) = W(\{D\}) + f_{penalty}(D) \tag{26}$$

where

$$W(\{D\}) = \sum_{i=1}^{nE} \rho_i \times A_i \times L_i \tag{27}$$

subject to

$$g_i(\{D\}) \leq 0, \qquad i = 1,2, \dots, nC \tag{28}$$

where $\{D\}$, d_i, nD, $W(\{D\})$, $f_{penalty}(D)$, and $PFit(\{D\})$ are defined similar to those in Eqs. (15), (16), and (17). nM is the number of frame members; ρ_i, A_i, and L_i are the material density, cross-sectional area, and length of the i-th frame element, respectively; R_i denotes the allowable set of values for the design variable d_i; nC is the number of problem constraints; and $g_i(\{D\})$ represents design constraints including strength constraints of the AISC-LRFD specification [40] and displacement constraints. Similar to the truss problems, the penalty function is defined based on Eq. (22). The design variable d_i is selected from a countable set of discrete values (denoted by R_i):

$$R_i = \left\{r_{i,1}, r_{i,2}, r_{i,3}, \dots, r_{i,nV(i)}\right\} \tag{29}$$

where $nV(i)$ is the number of available discrete values for the design variable d_i.
According to the provisions of the AISC-LRFD specification [40], design constraints are summarized as follows:

- Maximum lateral displacement constraint:

$$\frac{\Delta_T}{H} - R_H \leq 0 \tag{30}$$

where Δ_T is the maximum lateral displacement (lateral displacement of the roof); H is the height of the structure; and R_H is the maximum drift index, which is equal to $1/300$.

- Inter-story displacement constraints:

$$\frac{d_i}{h_i} - R_i \leq 0, \qquad i = 1,2, \dots, nS \tag{31}$$

where d_i is the inter-story drift of i-th story of the structure; h_i is the height of i-th story of the structure; nS is the total number of stories; and R_i is the allowable inter-story drift index of i-th story of the structure as given by the code of the practice.

- Strength constraints:

$$\begin{Bmatrix} \dfrac{P_u}{2\phi_c P_n} + \left(\dfrac{M_{ux}}{\phi_b M_{nx}} + \dfrac{M_{uy}}{\phi_b M_{ny}}\right) \leq 1, & \text{for} \quad \dfrac{P_u}{\phi_c P_n} < 0.2 \\ \dfrac{P_u}{\phi_c P_n} + \dfrac{8}{9}\left(\dfrac{M_{ux}}{\phi_b M_{nx}} + \dfrac{M_{uy}}{\phi_b M_{ny}}\right) \leq 1, & \text{for} \quad \dfrac{P_u}{\phi_c P_n} \geq 0.2 \end{Bmatrix} \tag{32}$$

where P_u and P_n are the required axial strength and the nominal axial strength (tension or compression), respectively; M_{ux} and M_{nx} are the required flexural strength and the nominal flexural strength about the x axis, respectively; M_{uy} and M_{ny} are the required flexural strength and the nominal flexural strength about the y axis, respectively; ϕ_c is the resistance factor (ϕ_c= 0.9 for tension, ϕ_c= 0.85 for compression); and ϕ_b is the resistance factor for flexure (ϕ_b= 0.90). It should be noted that $\bar{M}_{ny} = 0$ for two-dimensional frames.

3. RESULTS AND DISCUSSION

3.1 Optimization problems

Three benchmark structural optimization problems are considered from literature: continuous size optimization of a 120-bar dome-like truss structure with multiple frequency constraints; simultaneous size and layout optimization of a 52-bar dome-like truss structure with multiple frequency constraints; and discrete size optimization of a 3-bay 15-story steel frame structure with strength and displacement constraints. The maximum number of objective function evaluations ($MaxNFEs$) is considered as the stopping criterion of the algorithms. The value of $MaxNFEs$ is set to 20000 for all problems. However, in some cases, the number of performed evaluations may be much larger than the required ones. To ensure fair comparison of the performance of the algorithms, ten independent runs have been executed for each algorithm, and the results of the best run have been reported. Initial population is generated randomly. Weight convergence histories are provided and a magnified part is attached to convergence curves to display better curves. For each problem, optimized results at five different stages of the optimization process are provided, which allows comparing the performance of the algorithms. The algorithms, as well as finite elements analysis codes, are implemented in Matlab.

3.1.1 Example 1: 120-bar dome-like truss

The first example is size optimization of a 120-bar dome-like truss shown in Fig. 5. The layout of the structure is kept unchanged during the optimization process. Non-structural masses are attached to all free nodes of the dome as follows: 3000 kg at node one, 500 kg at nodes 2 through 13, and 100 kg at remaining nodes. Table 1 lists the material properties, variable bounds, and frequency constraint for this problem. The first two natural frequencies of the structure are considered as the problem constraints. All the dome members are categorized into seven groups considering the symmetry of the structure. The minimum and maximum allowable cross-sectional area of members are considered to be 1 cm^2 and 129.3 cm^2, respectively. This problem has been studied by different researchers including Kaveh and Ilchi Ghazaan [23] using vibrating particles system (VPS) and Kaveh et al. [9] using enhanced variants of the TLBO algorithm.

Table 2 presents the optimal design results for the 120-bar dome-like truss. The results indicate that OST-ABC performs better than ABC and STMP-ABC in terms of best weight, average weight, and worst weight of 10 independent runs. In addition, the OST-WEO and STMP-WEO have gained better results compared to the basic WEO in terms of best weight, average weight, and worst weight. A close examination of Table 2 confirms that the ABC algorithm and its set theoretical variants have better results than the WEO algorithm and its set theoretical variants in the present example. The first five natural frequencies of the optimal designs obtained by the proposed algorithms for the 120-bar dome-like truss are listed in Table 3. None of the frequency constraints are violated, as expected. Table 4 provides the optimized weights found by the ABC and WEO algorithms and their set theoretical variants at five different stages of the optimization process. It can be seen that the set theoretical variants of ABC and WEO converge faster than their basic versions. Figs. 6 and 7 compares the convergence curves of the average of 10 runs for the ABC and WEO algorithms and their set theoretical variants. As Fig. 6 demonstrates, the set theoretical variants of ABC have higher convergence rates compared to the basic ABC. Similarly, Fig. 7 indicates that the set theoretical variants of WEO converge faster than the basic WEO.

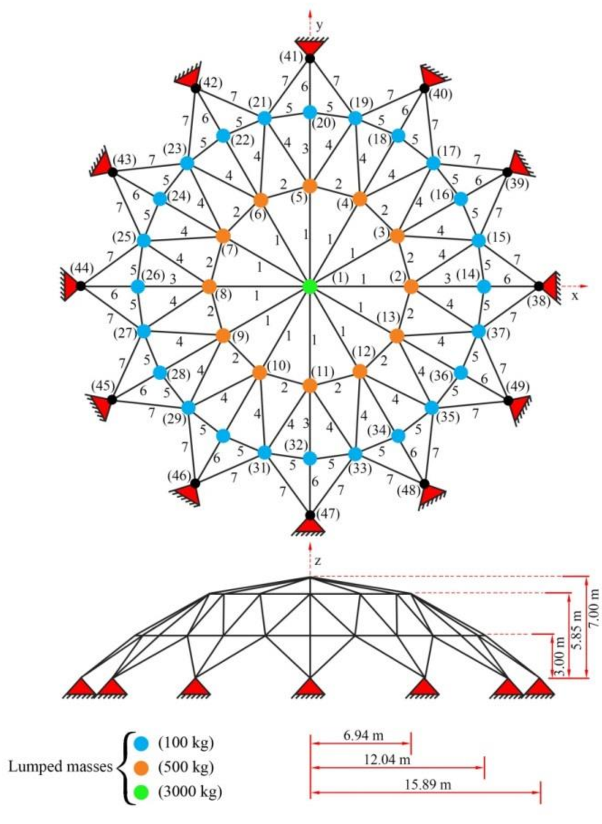

Figure 5. Schematic of the 120-bar dome-like truss

Table 1: Material properties, variable bounds, and frequency constraints of the 120-bar truss

Property / Unit	Value
E (Modulus of elasticity) / GPa	210
ρ (Material density) / kg/m^3	7971.81
Added mass / kg	3000 kg at node 1, 500 kg at nodes 2-13, 100 kg at remaining nodes
Lower bound of design variables / cm^2	1
Upper bound of design variables / cm^2	129.3
Frequency constraints / Hz	$\omega_1 \geq 9$, $\omega_2 \geq 11$

Table 2: Comparison of optimization results for the 120-bar dome-like truss

Element group	Cross-sectional areas (cm^2)					
	ABC	OST-ABC	STMP-ABC	WEO	OST-WEO	STMP-WEO
1	19.35	19.55	19.34	19.44	19.30	19.32
2	40.76	40.30	41.03	40.49	41.04	40.81
3	10.55	10.65	10.62	10.89	10.61	10.76
4	21.15	21.26	21.09	21.29	20.93	21.28
5	9.99	9.78	9.82	9.86	9.97	9.92
6	11.99	11.80	11.86	11.58	11.82	11.77
7	14.66	14.70	14.76	14.63	14.91	14.56
Best weight (kg)	8709.98	8709.18	8709.53	8712.51	8711.68	8710.10
Worst weight (kg)	8718.29	8716.16	8718.77	8726.18	8722.12	8721.82
Average weight (kg)	8713.34	8711.97	8713.30	8717.06	8715.54	8716.77
S.D. (kg)	2.88	2.55	2.44	3.71	3.54	3.82
No. of analyses	20000	20000	20000	20000	20000	20000
No. of runs	10	10	10	10	10	10

Table 3: Natural frequencies (Hz) of the optimal designs for the 120-bar dome-like truss

Frequency number	Natural frequencies (Hz)					
	ABC	OST-ABC	STMP-ABC	WEO	OST-WEO	STMP-WEO
1	9.0003	9.0001	9.0001	9.0023	9.0011	9.0001
2	11.0002	11.0001	11.0003	11.0002	11.0006	11.0000
3	11.0002	11.0002	11.0003	11.0002	11.0006	11.0000
4	11.0006	11.0002	11.0003	11.0011	11.0009	11.0006
5	11.0678	11.0664	11.0669	11.0673	11.0685	11.0671

Table 4: Optimized results at different stages of optimization for the 120-bar dome-like truss (best run)

No. of analyses	Optimized weight (kg)					
	ABC	OST-ABC	STMP-ABC	WEO	OST-WEO	STMP-WEO
4000	10675.73	10132.16	9495.41	9588.41	8816.32	8830.88

8000	8823.50	8750.70	8770.52	8925.64	8731.28	8730.16
12000	8727.99	8712.82	8716.03	8734.53	8719.28	8723.11
16000	8712.40	8709.60	8710.82	8712.51	8713.56	8710.21
20000	8709.98	8709.18	8709.53	8712.51	8711.68	8710.10

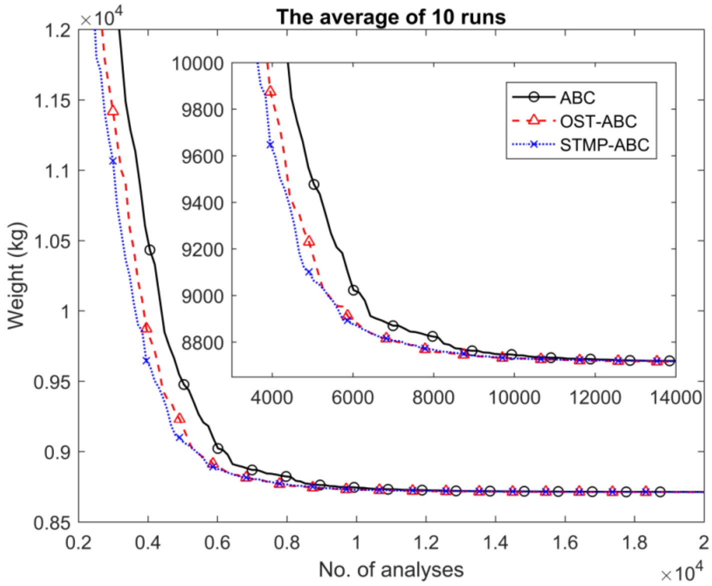

Figure 6. Convergence histories of the 120-bar dome-like truss obtained by the ABC, OST-ABC, and STMP-ABC algorithms

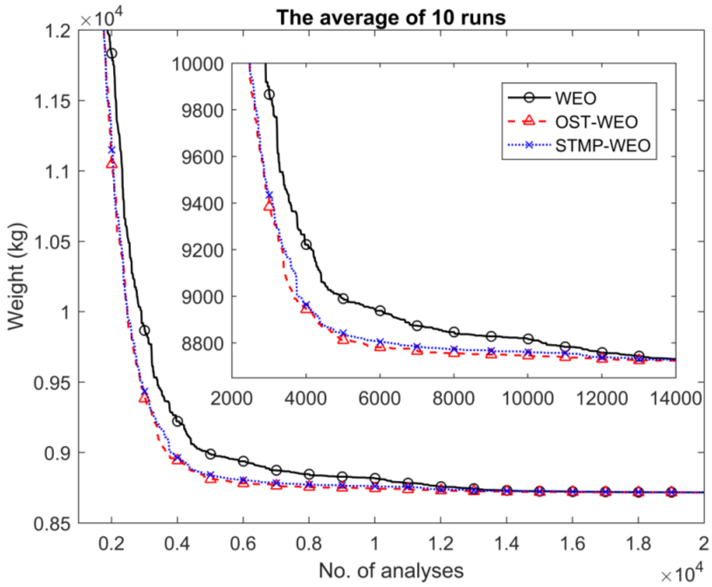

Figure 7. Convergence histories of the 120-bar dome-like truss obtained by the WEO, OST-WEO, and STMP-WEO algorithms

3.1.2 Example 2: 52-bar dome-like truss

Fig. 8 shows initial layout of a 52-bar dome-like truss considered as the second example. This is a simultaneous size and layout optimization problem. Cross-sectional area of members and nodal coordinates are considered as design variables. A non-structural mass of 50 kg is attached to all free nodes. The material properties, variable bounds, and frequency constraints are provided in Table 5. Due to the symmetry of the structure, all members of the dome are categorized into eight groups, as shown in Table 6. All free nodes are allowed to move $\pm 2\ m$ from their initial position in a symmetrical manner. Hence, this is a configuration optimization problem with 13 design variables (five layout variables and eight sizing variables) and two constraints on the first two natural frequencies. The cross-sectional area of the members can vary continuously between 1 cm^2 and 10 cm^2. This is a well-known problem studied by many researchers using different algorithms: Miguel and Fadel Miguel [20] employing harmony search (HS) and firefly algorithm (FA), Kaveh and Ilchi Ghazaan [22] utilizing an improved ray optimization (IRO) algorithm, and Kaveh and Zolghadr [24] using the cyclical parthenogenesis algorithm (CPA).

Table 7 compares the optimal design results for the 52-bar dome-like truss. The results show that OST-ABC and STMP-ABC have gained better results compared to the basic ABC in terms of best weight, average weight, and worst weight of 10 independent runs. In addition, the OST-ABC performs better than STMP-ABC in terms of best weight and average weight. Comparing the final results of the basic WEO with those of its set theoretical variants confirms that OST-WEO perform considerably better than the other methods, especially in terms of average weight, worst weight, and standard deviation. It can be concluded from the optimization results that the WEO algorithm and its set theoretical variants have better performance compared to the ABC algorithm and its set theoretical variants in this example. Table 8 lists the first five natural frequencies of the optimal designs obtained by the proposed algorithms for the 52-bar dome-like truss. The table confirms that all the frequency constraints are fulfilled, as expected. Also, the table indicates that the constraint on the second natural frequency controls the design process. The optimized weights found by the ABC and WEO algorithms and their set theoretical variants at five different stages of the optimization process are provided in Table 9. As the table demonstrates, in the early iterations, the set theoretical variants have slower rates of convergence compared to the basic versions, whereas, within the next iterations, the set theoretical variants converge more effectively than the basic versions. Figs. 9 and 10 provide a comparison between the convergence curves of set theoretical variants with those of the basic versions. As Fig. 6 shows, the set theoretical variants of ABC converge faster than the basic. Fig. 10 indicates that OST-WEO has better convergence rate compared to that of the WEO and STMP-WEO.

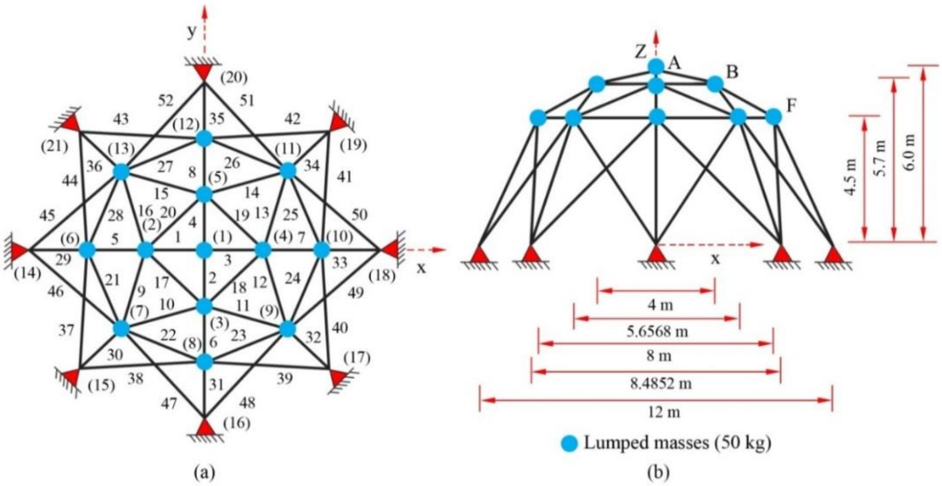

Figure 8. Schematic of the initial layout of the 52-bar dome-like truss: (a) Top view, (b) Side view

Table 5: Material properties, variable bounds, and frequency constraints of the 52-bar truss

Property / Unit	Value
E (Modulus of elasticity) / GPa	210
ρ (Material density) / kg/m^3	7800
Added mass / kg	50
Lower bound of design variables / cm^2	1
Upper bound of design variables / cm^2	10
Frequency constraints / Hz	$\omega_1 \leq 15.916$, $\omega_2 \geq 28.648$

Table 6: Element grouping of the 52-bar dome-like truss

Group number	Elements of the group
A_1	1-4
A_2	5-8
A_3	9-16
A_4	17-20
A_5	21-28
A_6	29-36
A_7	37-44
A_8	45-52

Table 7: Comparison of optimization results for the 52-bar dome-like truss

Design variable	Coordinates (m) and cross-sectional areas (cm^2)					
	ABC	OST-ABC	STMP-ABC	WEO	OST-WEO	STMP-WEO
Z_A (m)	6.19	6.07	5.98	5.94	5.91	5.99
X_B (m)	2.21	2.26	2.30	2.31	2.30	2.24

Z_B (m)	4.05	4.03	4.01	4.04	3.98	4.01
X_F (m)	3.86	3.80	3.72	3.76	3.71	3.80
Z_F (m)	2.50	2.50	2.50	2.50	2.50	2.50
A_1 (cm^2)	1.00	1.00	1.00	1.00	1.00	1.01
A_2 (cm^2)	1.18	1.12	1.14	1.08	1.12	1.13
A_3 (cm^2)	1.27	1.20	1.27	1.20	1.21	1.21
A_4 (cm^2)	1.32	1.39	1.48	1.49	1.46	1.43
A_5 (cm^2)	1.47	1.41	1.48	1.46	1.40	1.40
A_6 (cm^2)	1.00	1.00	1.00	1.01	1.00	1.00
A_7 (cm^2)	1.45	1.48	1.44	1.44	1.49	1.48
A_8 (cm^2)	1.45	1.49	1.43	1.48	1.48	1.49
Best weight (kg)	194.26	193.59	193.81	193.98	193.63	193.91
Worst weight (kg)	204.56	203.93	203.92	204.08	197.28	203.65
Average weight (kg)	200.40	198.70	200.02	197.51	194.87	196.82
S.D. (kg)	3.96	4.17	3.99	2.99	1.00	2.85
No. of analyses	20000	20000	20000	20000	20000	20000
No. of runs	10	10	10	10	10	10

Table 8: Natural frequencies (Hz) of the optimal designs for the 52-bar dome-like truss

Frequency number	Natural frequencies (Hz)					
	ABC	OST-ABC	STMP-ABC	WEO	OST-WEO	STMP-WEO
1	11.1584	11.4701	11.6100	11.8206	11.9010	11.5448
2	28.6485	28.6484	28.6513	28.6500	28.6483	28.6534
3	28.6486	28.6486	28.6517	28.6519	28.6489	28.6534
4	28.6631	28.6509	28.6574	28.6531	28.6681	28.6554
5	28.7050	28.6874	28.7333	29.0592	28.9014	29.0268

Table 9: Optimized results at different stages of optimization for the 52-bar dome-like truss (best run)

No. of analyses	Optimized weight (kg)					
	ABC	OST-ABC	STMP-ABC	WEO	OST-WEO	STMP-WEO
4000	294.24	475.74	406.62	335.82	441.95	350.12
8000	202.89	215.50	200.64	247.94	253.84	257.61
12000	196.22	195.96	194.50	207.63	207.41	202.90
16000	194.53	193.66	194.04	195.43	194.47	195.86
20000	194.26	193.59	193.81	193.98	193.63	193.91

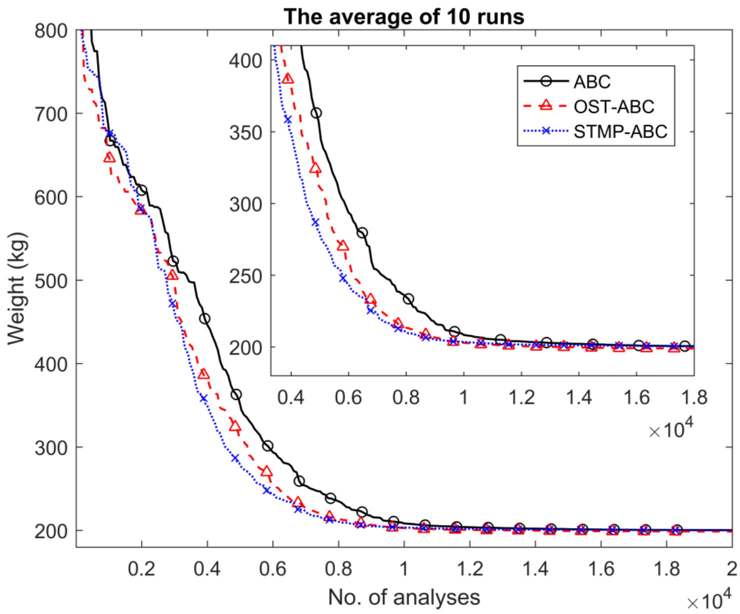

Figure 9. Convergence histories of the 52-bar dome-like truss obtained by the ABC, OST-ABC, and STMP-ABC algorithms

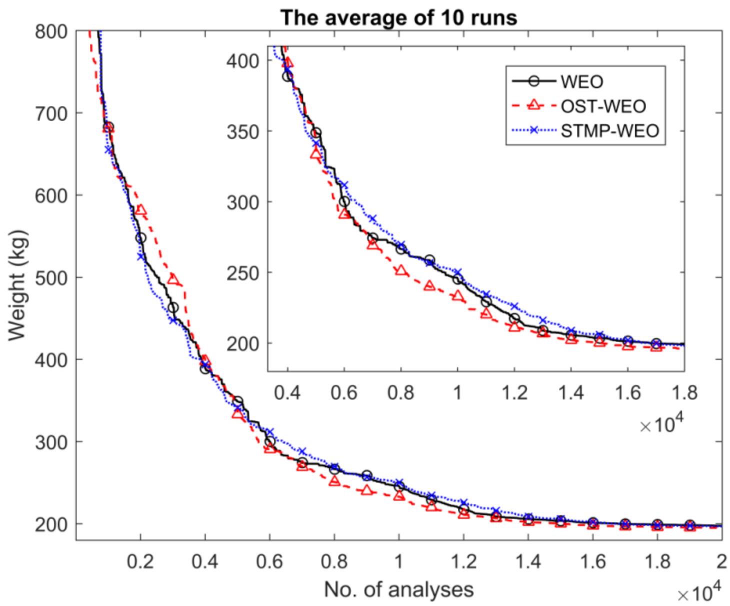

Figure 10. Convergence histories of the 52-bar dome-like truss obtained by the WEO, OST-WEO, and STMP-WEO algorithms

3.1.3 Example 3: 3-bay 15-story steel frame

Fig. 11 shows the configuration and loading conditions of a 3-bay 15-story steel frame structure consisting of 64 joints and 105 members. The columns are grouped into ten column groups, whereas all beams are considered to form only one beam group, as shown in Fig. 11. Column groups are formed in a way that the exterior columns of each three consecutive stories (starting from the foundation) form one column group. In a similar way, the interior columns of each three consecutive stories form one column group. Therefore, the problem has 11 discrete design variables. All design variables (element groups) are chosen from 267 W-sections. The steel has a yield stress of $F_y = 36\ ksi$ and a modulus of elasticity equal to $E = 29000\ ksi$. The frame is subjected to strength and displacement constraints based on the AISC-LRFD requirements. The non-braced length of beam elements is considered to be equal to one-fifth of the span length. Column elements have no lateral restraint along their lengths. The effective length factor of columns in a sway-permitted frame is calculated as $K_x \geq 1$ by using the approximation equation proposed by Dumonteil [42] and the out-of-plane effective length factor is specified as $K_y = 1$. An additional displacement constraint is imposed on the sway of the top story, which is limited to 9.25 in. This problem has been solved with various methods by different researchers: Kaveh and Talatahari using CSS [30], Kaveh and Ilchi Ghazaan using CBO and ECBO [35], Kaveh and Bakhshpoori using an accelerated water evaporation optimization [37], and Kaveh and Ilchi Ghazaan using VPS [38].

Table 10 compares the optimal design results for the 3-bay 15-story steel frame. The results indicate that the weight of the basic ABC is slightly better than those of the OST-ABC and STMP-ABC, but OST-ABC and STMP-ABC have better performance in terms of the average weight, worst weight, and standard deviation compared to those of the basic ABC. In addition, the set theoretical variants of WEO perform better than the basic WEO in all aspects. Comparing the results obtained by the OST-WEO with those of the STMP-WEO confirms that the OST-WEO has better performance. It can be concluded from the results that the WEO algorithm and its set theoretical variants perform better than the ABC algorithm and its set theoretical variants in this example. Table 11 provides the optimized design weights found by the algorithms at five different stages of the optimization process. Figs. 12 and 13 provide a comparison between the convergence curves of the basic version of ABC and WEO with those of their set theoretical variants. The STMP-ABC converges considerably faster than WEO and OST-WEO. In addition, as Fig. 13 shows, in the early iterations, OST-WEO and STMP-WEO have slower rates of convergence compared to the basic WEO, whereas, within the last iterations, the set theoretical variants converge more effectively than the basic WEO. Fig. 14 shows the inter-story drifts for the best design of the 3-bay 15-story frame obtained by OST-WEO. As Fig. 14 confirms, all the inter-story drift constraints are fulfilled. Fig. 15 shows the stress ratios for the best design of the 3-bay 15-story frame obtained by OST-WEO. A close examination of Figs. 14 and 15 reveals that stress constraints control the design of the 3-bay 15-story steel frame.

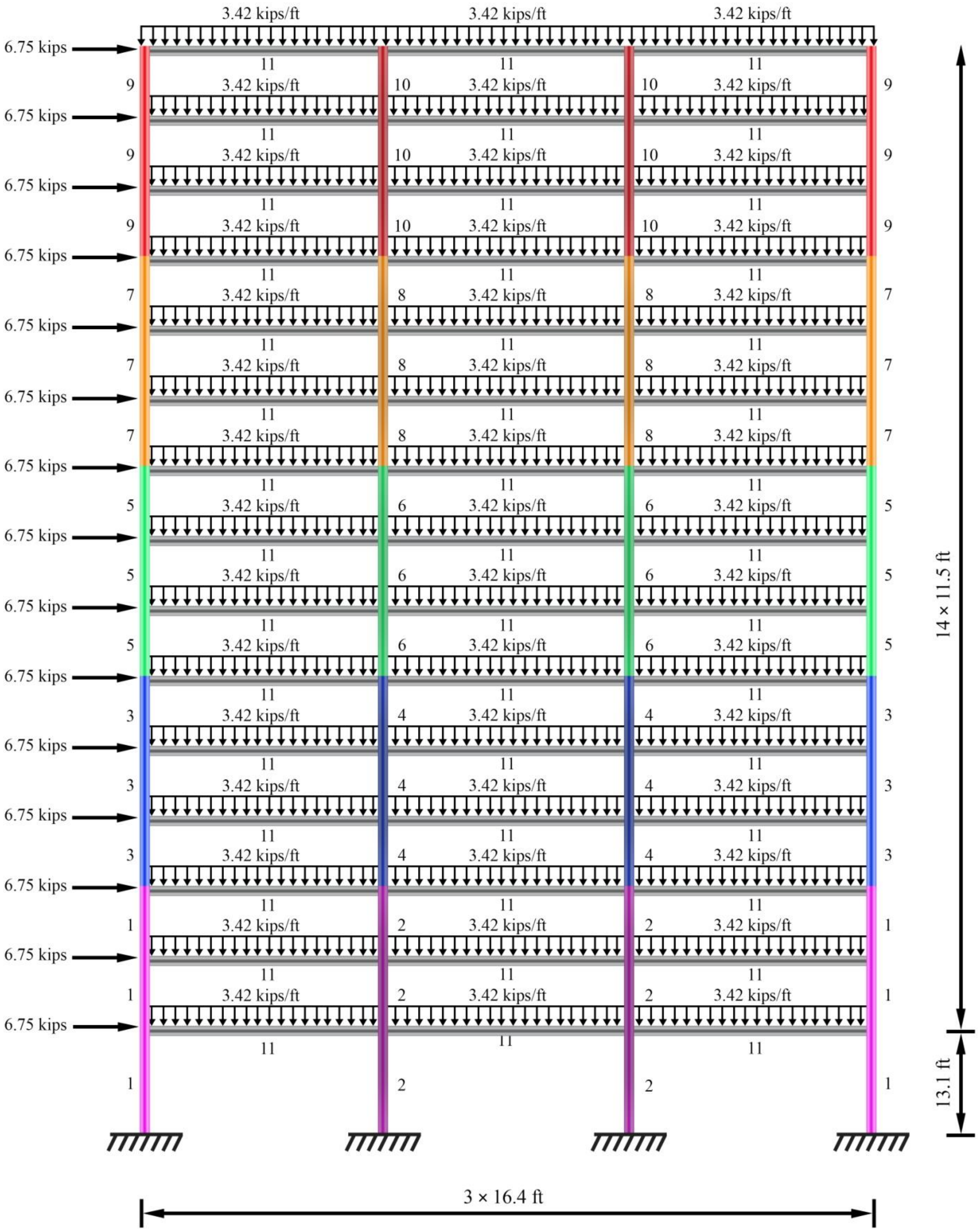

Figure 11. the 3-bay 15-story steel frame structure

Table 10: Comparison of optimization results for the 3-bay 15-story frame

Element group	ABC	OST-ABC	STMP-ABC	WEO	OST-WEO	STMP-WEO
1	W14×99	W14×99	W16×89	W16×89	W14×99	W24×117
2	W27×161	W27×161	W36×170	W36×170	W24×162	W27×146
3	W12×79	W12×79	W12×79	W12×79	W12×79	W27×84
4	W27×114	W27×114	W21×111	W33×118	W24×104	W24×104
5	W24×68	W14×61	W14×61	W24×68	W14×61	W16×67
6	W18×86	W30×90	W30×90	W12×87	W30×90	W18×86
7	W8×48	W10×45	W16×50	W14×48	W21×48	W10×49
8	W21×68	W24×68	W21×68	W24×68	W21×68	W12×65
9	W8×28	W14×34	W8×28	W12×30	W8×35	W8×31
10	W10×39	W18×35	W10×39	W8×40	W14×38	W10×39
11	W21×44	W21×44	W21×44	W21×44	W21×44	W21×44
Best weight (*lb*)	87814	87883	87883	88366	87748	87893
Worst weight (*lb*)	96060	93381	92932	91434	90120	90418
Average weight (*lb*)	90659	90065	90029	89185	88856	89037
S.D. (*lb*)	2595	1869	1907	933	669	777
No. of analyses	20000	20000	20000	20000	20000	20000
No. of runs	10	10	10	10	10	10

Table 11: Optimized results at different stages of optimization for the 3-bay 15-story frame (best run)

No. of analyses	Optimized weight (*lb*)					
	ABC	OST-ABC	STMP-ABC	WEO	OST-WEO	STMP-WEO
4000	118143	122515	113633	97762	95665	101382
8000	97416	99675	95442	91783	89906	97742
12000	91203	91890	91893	91254	89009	93750
16000	88780	89335	88918	89389	88737	88445
20000	87814	87883	87883	88366	87748	87893

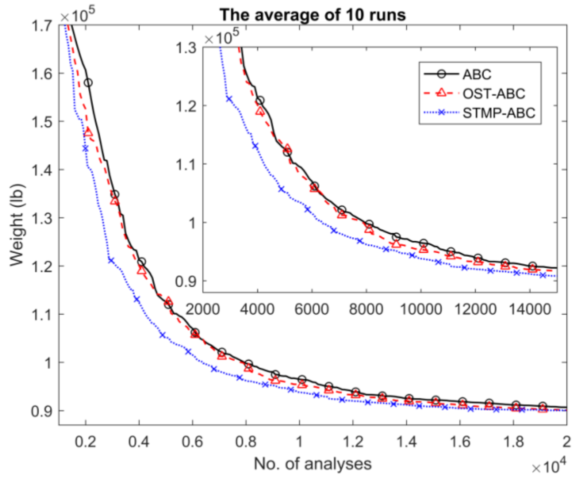

Figure 12. Convergence histories of the 3-bay 15-story steel frame obtained by the ABC, OST-ABC, and STMP-ABC algorithms

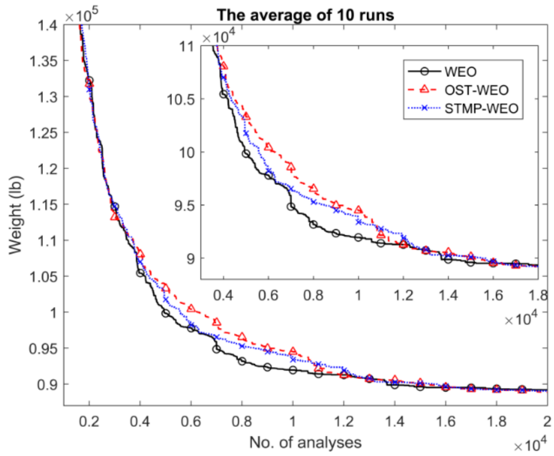

Figure 13. Convergence histories of the 3-bay 15-story steel frame obtained by the WEO, OST-WEO, and STMP-WEO algorithms

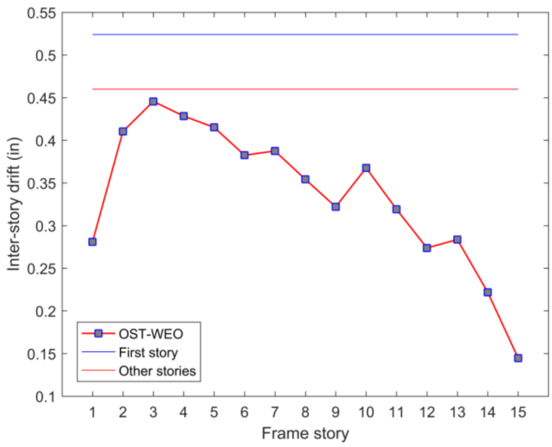

Figure 14. Inter-story drifts for the 3-bay 15-story steel frame obtained by OST-WEO (best run)

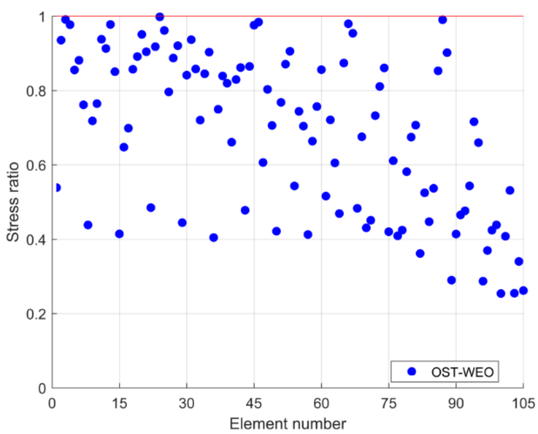

Figure 15. Stress ratios for the 3-bay 15-story steel frame obtained by OST-WEO (best run)

4. CONCLUSION

In this paper, set theoretical variants of the artificial bee colony (ABC) and water evaporation optimization (WEO) algorithms were proposed. The set theoretical variants were designed based on a simple general set theoretical framework proposed by Kaveh et al. [9]. The set theoretical framework, which is applicable to almost all population-based optimization algorithms, makes it possible to design various versions of a P-metaheuristic. The main idea of the set theoretical variants is based on the division of the initial population into a number of smaller well-arranged sub-populations with the aim of improving the balance between the diversification and the intensification of the search space. In order to verify the validity and efficiency of the proposed algorithms, some benchmark structural optimization problems, including two frequency-constrained truss optimization problems and one steel frame structure with displacement and strength constraints, were studied. The results show that the set theoretical variants of ABC and WEO have better the convergence characteristics compared to their basic versions. It can be concluded from the optimization results that the OST variant of the ABC and WEO algorithms have better performance compared to the corresponding STMP variants.

REFERENCES

1. Jain PK, Ahuja OP. *Functional Analysis*, New Academic Science, 2nd edition, England, 2010.
2. Cantor G. *Contributions to the Founding of the Theory of Transfinite Numbers*, Open Court Publishing, Chicago, 1915.
3. Kenneth K. *Set Theory: An Introduction to Independence Proofs*, North-Holland, 1st edition, Amsterdam, 1983.
4. Behravesh A, Kaveh A, Sabet S, Nani M. A set theoretical approach for configuration generation, *Comput Struct* 1988; **30**(6): 1293-302. https://doi.org/10.1016/0045-7949(88)90194-0
5. Kaveh A. *Structural Mechanics: Graph and Matrix Methods*, Research Studies Press, 3rd edition, Baldock, Hertfordshire, England, 2004.
6. Kaveh A, Biabani Hamedani K, Zaerreza A. A set theoretical shuffled shepherd optimization algorithm for optimal design of cantilever retaining wall structures, *Eng Comput* 2020; 1-18. https://doi.org/10.1007/s00366-020-00999-9
7. Descamps B, Filomeno Coelho R. *Graph Theory in Evolutionary Truss Design Optimization*. In: Gandomi AH, Yang XS, Talatahari S, Alavi AH, editors. *Metaheuristic Applications in Structures and Infrastructures*, London: Elsevier; 2013, pp. 241-268.
8. Kaveh A, Bakhshpoori T. *Metaheuristics: Outlines, MATLAB Codes and Examples*, Springer, 1st edition, Cham, Switzerland, 2019.
9. Kaveh A, Biabani Hamedani K, Kamalinejad M. Set theoretical variants of the teaching–learning-based optimization algorithm for optimal design of truss structures with multiple frequency constraints, *Acta Mech* 2020; **231**: 3645-72. https://doi.org/10.1007/s00707-020-02718-3

10. Karaboga D. An idea based on honey bee swarm for numerical optimization, Technical Report, Erciyes University, Engineering Faculty, Computer Engineering Department, Erciyes, Turkey, 2005
11. Kaveh A, Bakhshpoori T. Water evaporation optimization: a novel physically inspired optimization algorithm, *Comput Struct* 2016; **167**: 69-85. https://doi.org/10.1016/j.compstruc.2016.01.008
12. Kaveh A, Zolghadr A. Optimal design of cyclically symmetric trusses with frequency constraints using cyclical parthenogenesis algorithms, *Adv Struct Eng* 2018; **21**(5): 739-55. https://doi.org/10.1177%2F1369433217732492
13. Grandhi R. Structural optimization with frequency constraints-a review, *AIAA J* 1993; **31**(12): 2296-303. https://doi.org/10.2514/3.11928
14. Bellagamba L, Yang TY. Minimum-mass truss structures with constraints on fundamental natural frequency, *AIAA J* 1981; **19**(11): 1452-8. https://doi.org/10.2514/3.7875
15. Grandhi R, Venkayyat VB. Structural optimization with frequency constraints, *AIAA J* 1988; **26**(7): 858-66. https://doi.org/10.2514/3.9979
16. Tong WH, Liu GR. An optimization procedure for truss structures with discrete design variables and dynamic constraints, *Comput Struct* 2001; **79**(2): 155-62. https://doi.org/10.1016/S0045-7949(00)00124-3.
17. Sedaghati R, Suleman A, Tabarrok B. Structural optimization with frequency constraints using the finite element force method, *AIAA J* 2002; **40**(2): 382-8. https://doi.org/10.2514/2.1657
18. Lingyun W, Mei Z, Guangming W, Guang M. Truss optimization on shape and sizing with frequency constraints based on genetic algorithm, *Comput Mech* 2005; **35**(5): 361-8. https://doi.org/10.1007/s00466-004-0623-8
19. Gomez HM. Truss optimization with dynamic constraints using a particle swarm algorithm, *Expert Syst Appl* 2011; **38**(1): 957-68. https://doi.org/10.1016/j.eswa.2010.07.086
20. Miguel LFF, Fadel Miguel LF. Shape and size optimization of truss structures considering dynamic constraints through modern metaheuristic algorithms, *Expert Syst Appl* 2012; **39**(10): 9458-67. https://doi.org/10.1016/j.eswa.2012.02.113
21. Kaveh A, Javadi SM. Shape and size optimization of trusses with multiple frequency constraints using harmony search and ray optimizer for enhancing the particle swarm optimization algorithm, *Acta Mech* 2014; **225**(6): 1595-605. https://doi.org/10.1007/s00707-013-1006-z
22. Kaveh A, Ilchi Ghazaan M. Layout and size optimization of trusses with natural frequency constraints using improved ray optimization algorithm, *IJST-T CIV ENG* 2015; **39**:C2+: 395-408. https://dx.doi.org/10.22099/ijstc.2015.3509
23. Kaveh A, Ilchi Ghazaan M. Vibrating particles system algorithm for truss optimization with multiple natural frequency constraints, *Acta Mech* 2017; **228**(1): 307-22. https://doi.org/10.1007/s00707-016-1725-z
24. Kaveh A, Zolghadr A. Cyclical parthenogenesis algorithm for layout optimization of truss structures with frequency constraints, *Eng Optim* 2017; **49**(8): 1317–34. https://doi.org/10.1080/0305215X.2016.1245730
25. Kaveh A, Zolghadr A. Meta-heuristic methods for optimization of truss structures with vibration frequency constraints, *Acta Mech* 2018; **229**(10): 3971-92.

https://doi.org/10.1007/s00707-018-2234-z

26. Kaveh A, Biabani Hamedani K, Barzinpour F. Optimal size and geometry design of truss structures utilizing seven meta-heuristic algorithms: a comparative study, *Int J Optim Civil Eng* 2020; **10**(2): 231–60.
27. Pezeshk S, Camp CV, Chen D. Design of nonlinear framed structures using genetic optimization, *J Struct Eng* 2000; **126**(3): 382-8. https://doi.org/10.1061/(ASCE)0733-9445(2000)126:3(382)
28. Camp CV, Bichon BJ, Stovall SP. Design of steel frames using ant colony optimization, *J Struct Eng* 2005; **131**(3): 369-79. https://doi.org/10.1061/(ASCE)0733-9445(2005)131:3(369)
29. Degertekin SO. Optimum design of steel frames using harmony search algorithm, *Struct Multidiscipl Optim* 2008; **36**(4): 393-401. https://doi.org/10.1007/s00158-007-0177-4
30. Kaveh A, Talatahari S. Charged system search for optimal design of frame structures *Appl Soft Compu* 2012; **12**(1): 382-93. https://doi.org/10.1016/j.asoc.2011.08.034
31. Zhou S, Yang B. Improved Genetic Algorithm and its application in optimization of frame structures, *Eng Comput* 2012; **9**(3): 245-50. https://doi.org/10.1260/1708-5284.9.3.245
32. Kaveh A, Bakhshpoori T. Optimum design of steel frames using Cuckoo Search algorithm with Lévy flights, *Struct Des Tall Spec Build* 2013; **22**(13): 1023-36. https://doi.org/10.1002/tal.754
33. Maheri MR, Narimani MM. An enhanced harmony search algorithm for optimum design of side sway steel frames, *Comput Struct* 2014; **136**: 78-89. https://doi.org/10.1016/j.compstruc.2014.02.001
34. Talatahari S, Gandomi AH, Yang XS, Deb S. Optimum design of frame structures using the eagle strategy with differential evolution, *Eng Struct* 2015; **91**: 16-25. https://doi.org/10.1016/j.engstruct.2015.02.026
35. Kaveh A, Ilchi Ghazaan M. A comparative study of CBO and ECBO for optimal design of skeletal structures, *Comput Struct* 2015; **153**: 137-47. https://doi.org/10.1016/j.compstruc.2015.02.028
36. Kazemzadeh Azad S, Hasançebi O. Computationally efficient discrete sizing of steel frames via guided stochastic search heuristic, *Comput Struct* 2015; **156**: 12-28. https://doi.org/10.1016/j.compstruc.2015.04.009
37. Kaveh A, Bakhshpoori T. An accelerated water evaporation optimization formulation for discrete optimization of skeletal structures, *Comput Struct* 2016; **177**: 218-28. https://doi.org/10.1016/j.compstruc.2016.08.006
38. Kaveh A, Ilchi Ghazaan M. A new meta-heuristic algorithm: vibrating particles system *Sci Iran, Transact A, Civil Eng* 2017; **24**(2): 551-66. https://dx.doi.org/10.24200/sci.2017.2417
39. Kaveh A, Biabani Hamedani K, Hosseini SM, Bakhshpoori T. Optimal design of planar steel frame structures utilizing metaheuristic optimization algorithms, *Struct* 2020; **25**: 335-46. https://doi.org/10.1016/j.istruc.2020.03.032
40. American Institute of Steel Construction (Ed.). *Load and Resistance Factor Design*, American Institute of Steel Construction, 2001.
41. Talbi EG. *Metaheuristics: from Design to Implementation*, John Wiley & Sons, 1st edition, USA, 2009.
42. Dumonteil P. Simple equations for effective length factors, *Eng J AISC* 1992; **29**(3): 111-5.