

Chapter 16

Water Evaporation Optimization Algorithm

16.1 Introduction

Efficient metaheuristic optimization algorithms are developed to overcome the drawbacks of some traditional methods in highly nonlinear engineering optimization problems with high complexity, high dimension, and multimodal design spaces gaining increasing popularity nowadays [1]. Performance assessment of a metaheuristic algorithm may be used by solution quality, computational effort, and robustness [2] directly affected by its two contradictory criteria: exploration of the search space (diversification) and exploitation of the best solutions found (intensification).

A novel metaheuristic algorithm, imitating the evaporation process of a tiny amount of water molecules adhered on a solid surface with different wettability, has been developed very recently and named water evaporation optimization (WEO) [3]. WEO was successfully utilized in real parameter optimization [3] and continuous structural optimization problems [4] presenting: (1) competitive behavior with other algorithms in terms of accuracy and robustness, (2) advantages over other algorithms in the aspect of parameter tuning (except population size, other parameters are set based on the molecular dynamics), and (3) imposing a rational number of physical rules that lead to significantly good convergence behavior and simple algorithmic structure. In structural optimization problems, high computational cost of the algorithm has been found to be the only drawback of the algorithm. In view of this, the present study will propose an accelerated version of WEO potentially able to solve multidisciplinary and engineering optimization problems regardless of the type of optimization problem at hand.

The basic WEO is initialized with a population of random designs named water molecules that are updated through an iterative process to search the optimum. Updating is governed by evaporation rules based on molecular dynamic simulation results obtained for evaporation process of water molecules from a solid surface with different wettability. The candidate solutions are updated globally and locally

in two independent sequential phases: monolayer and droplet evaporation phases. Rules driving each phase are in a good agreement with the local and global search ability of the algorithm. In this study, it is shown that the WEO can be improved to reduce computational cost by simultaneous utilization of both phases.

The improvement to WEO hence involves a simultaneous use of both phases. Two scenarios are developed for considering simultaneous evaporation phases based on (1) water molecules' distances and (2) objective function values of molecules. The effects of these enhancements are tested in very classical discrete structural optimization problems. The first scenario which is developed from the physics of water evaporation phenomena is efficient in enhancing the convergence rate. Although the second scenario (which is developed greedily) also is efficient, the first scenario is accepted and developed in this study in view of its physical interpretation and higher performance.

This chapter is based on the accelerated version of the water evaporation optimization algorithm developed by Kaveh and Bakhshpoori [5], and it is organized as follows: Section 16.2 outlines the basic WEO algorithm. Section 16.3 deals with the development of accelerated version of WEO with mixed phases. Section 16.4 discusses the performance of the new formulation of WEO in discrete structural optimization problems. Finally, conclusions are given in Sect. 16.5.

16.2 Basic Water Evaporation Optimization Algorithm

Evaporation of water restricted on the surface of solid materials is the inspiration basis of WEO which is different from the water evaporation of bulk surface. This type of water evaporation is essential in the macroscopic world such as the water loss through the surface of soil [6]. Wang et al. [7] presented molecular dynamics (MD) simulations on the evaporation of water from a solid substrate with different surface wettability. MD simulations were carried out by adhering nanoscale water aggregation in a neutral substrate which is chargeable. By varying the value of charge ($0 \text{ e} \leq q \leq 0.7 \text{ e}$), a substrate with tunable surface wettability can be obtained. It is found that as the surface changed from hydrophobicity ($q < 0.4 \text{ e}$) to hydrophilicity ($q \geq 0.4 \text{ e}$), the evaporation speed did not show a monotonic decrease from intuition, but increased first and then decreased after reaching a maximum value. Figure 16.1 depicts the MD simulation method: (a) side view of the initial system (the upward arrow denotes the accelerating region), (b) snapshot of water on the substrate with low wettability (the water molecules accumulate into the form of a sessile spherical cap with a contact angle θ to the surface), (c) snapshot of water on the substrate with high wettability (the adhered water forms a flat single-layer molecule sheet), and (d) theoretical topology of water molecules with respect to substrate wettability used for MD simulations.

Considering MD simulation results from end to beginning, a fine analogy can be found between this type of water evaporation phenomena and a population-based

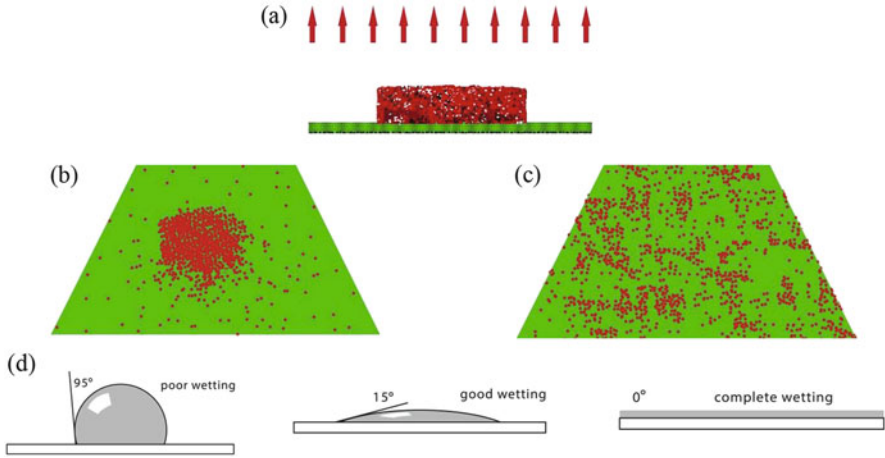


Fig. 16.1 (a) Side view of the initial system. (b) snapshot of water on the substrate with low wettability ($q = 0 \text{ e}$). (c) snapshot of water on the substrate with high wettability ($q = 0.7 \text{ e}$). (d) theoretical topology of water molecules with respect to substrate wettability used for MD simulations

Water molecules	~	Algorithm individuals
Substrate with decreasing wettability	~	Search space with different objective function values
Water aggregation reforms from a monolayer to a sessile droplet with decreasing the surface wettability	~	Change mutual positions of individuals as the algorithm progresses
Decreasing q from 0.7 e to 0.0	~	Decreasing the objective function value
$q = 0.4 \text{ e}$	~	The algorithm reaches the middle of the optimization process
Evaporation in two phases (Monolayer and droplet) with different evaporation flux	~	Global and local search ability of the algorithm

Fig. 16.2 Analogy between water evaporation from a solid surface and a population-based metaheuristic algorithm

metaheuristic algorithm. This analogy led us to developing the basic WEO algorithm [4, 5] depicted in Fig. 16.2.

Water molecules and substrate with decreasing wettability are considered as algorithm individuals and search space, respectively. Decreasing the surface wettability reforms the water aggregation from a monolayer to a sessile droplet. Similarly, mutual positions of individuals in the design space change as the search process progresses. Decreasing q from 0.7 e to 0.0 e can represent the reduction of the objective function for a minimization problem. Evaporation flux variation is considered as the most appropriate measure for updating the algorithm individuals which is in a good agreement with the local and global search ability.

Evaporation flux reaches its maximum around $q=0.4$ e. This situation is considered in the basic WEO [4] until the algorithm reaches the middle of the optimization process. In other words, basic WEO updates the individuals in two independent sequential phases: monolayer and droplet evaporation phases. These two phases were explained in detail in [4]. In each phase the corresponding rules are in a good agreement with the local and global search ability of the algorithm. The steps entailed by the basic WEO are now outlined:

Step 1: Initialization

Algorithm parameters are set in the first step. These parameters are the number of water molecules (nVM), maximum number of algorithm iterations (t_{max}), minimum (MEP_{min}) and maximum (MEP_{max}) values of monolayer evaporation probability, and minimum (DEP_{min}) and maximum (DEP_{max}) values of droplet evaporation probability. Evaporation probability parameters are determined efficiently for WEO based on the MD simulation results ($MEP_{min}=0.03$ and $MEP_{max}=0.6$; $DEP_{min}=0.6$ and $DEP_{max}=1$). The initial positions of all water molecules are generated randomly within the n -dimensional search space ($WM^{(0)}$) as follows and are evaluated based on the objective function of the problem at hand:

$$WM_{i,j}^{(0)} = Round(x_{j,min} + rand_{i,j} \cdot (x_{j,max} - x_{j,min})) \quad (16.1)$$

where $WM_{i,j}^{(0)}$ determines the initial value of the j th variable for the i th water molecule; $x_{j,min}$ and $x_{j,max}$ are the minimum and maximum allowable values for the j th variable; and $rand_{i,j}$ is a random number in the interval $[0, 1]$. The rounding function is used for discrete problems and rounds the value of design variable to the nearest discrete available value. The best water molecule as the output of the algorithm will be returned in this step.

Step 2: Generating Water Evaporation Matrix

Every water molecule follows the evaporation probability rules specified for each phase of the algorithm.

For $t \leq t_{max}/2$ or in the monolayer evaporation phase, in each iteration the objective function of individuals Fit_i^t is scaled to the interval $[-3.5, -0.5]$ and represents the corresponding $E_{sub}(i)^t$ inserted to each individual (substrate energy vector) via the following scaling function:

$$E_{sub}(i)^t = \frac{(E_{max} - E_{min}) \times (Fit_i^t - Min(Fit))}{(Max(Fit) - Min(Fit))} + E_{min} \quad (16.2)$$

Min and Max are the minimum and maximum functions, respectively. After generating the corresponding substrate energy vector, water molecules are globally evaporated based on the monolayer evaporation probability (MEP) rule:

$$MEP_{ij}^t = \begin{cases} 1 & \text{if } rand_{i,j} < \exp(E_{sub}(i)^t) \\ 0 & \text{if } rand_{i,j} \geq \exp(E_{sub}(i)^t) \end{cases} \quad (16.3)$$

where MEP_{ij}^t is the updating probability for the j th variable of the i th individual or water molecule in the t th optimization iteration and $rand_{i,j}$ is a random number generated by the uniform distribution in the interval $[0, 1]$. In this way, in the monolayer evaporation phase, the best and worst candidate solutions will be updated by the probability equal to $\exp(-3.5) = 0.03$ and $\exp(-0.5) = 0.6$, respectively. These values can be considered as minimum (MEP_{min}) and maximum (MEP_{max}) values of monolayer evaporation probability. Algorithm performance evaluations show that considering $MEP_{min} = 0.03$ and $MEP_{max} = 0.6$ based on the MD simulation results is logical.

For $t > t_{max}/2$ or in the droplet evaporation phase, the objective function of individuals Fit_i^t is scaled to the interval $[-50^\circ, -20^\circ]$ via the following scaling function which represents the corresponding contact angle $\theta(i)^t$ (contact angle vector):

$$\theta(i)^t = \frac{(\theta_{max} - \theta_{min}) \times (Fit_i^t - \text{Min}(Fit))}{(\text{Max}(Fit) - \text{Min}(Fit))} + \theta_{min} \quad (16.4)$$

After generating contact angle vector, evaporation occurs based on the droplet evaporation probability (DEP) matrix:

$$DEP_{ij}^t = \begin{cases} 1 & \text{if } rand_{i,j} < J(\theta_i^{(t)}) \\ 0 & \text{if } rand_{i,j} \geq J(\theta_i^{(t)}) \end{cases} \quad (16.5)$$

$$J(\theta) = J_0 P_0 \left(\frac{2}{3} + \frac{\cos^3 \theta}{3} - \cos \theta \right)^{-2/3} (1 - \cos \theta), \quad J_0 P_0 = \frac{1}{24}$$

where DEP_{ij}^t is the updating probability for the j th variable of the i th water molecule in the t th optimization iteration. In this way, minimum (DEP_{min}) and maximum (DEP_{max}) values of droplet evaporation probability are obtained equal to 0.6 and 1, respectively. Algorithm performance evaluation results show that these values are suitable.

Step 3: Generating Random Permutation Based Step Size Matrix

A random permutation based step size matrix is generated according to:

$$S = rand \cdot \left(WM^{(t)}[permute1(i)(j)] - WM^{(t)}[permute2(i)(j)] \right) \quad (16.6)$$

where $rand$ is a random number in the $[0, 1]$ range, $permute1$ and $permute2$ are different row permutation functions, i is the number of water molecule, and j is the number of design variables of the problem at hand. It should be noted that random permutation based step size is used for two purposes. In the first phase, water

molecules are more far from each other than in the second. In this way, the generated permutation based step size will guarantee global and local search capability in each phase. The random part will guarantee the algorithm to be sufficiently dynamic. These two aspects are emphasized by considering two specific evaporation probability mechanisms for each phase.

Step 4: Generating Evaporated Water Molecules and Updating the Matrix of Water Molecules

The evaporated set of water molecules $WM^{(t+1)}$ is generated by adding the product of step size matrix and evaporation probability matrix to the current set of molecules $WM^{(t)}$ according to:

$$WM^{(t+1)} = Round \left(WM^{(t)} + S \times \begin{cases} MEP^{(t)} & t \leq t_{max}/2 \\ DEP^{(t)} & t > t_{max}/2 \end{cases} \right) \quad (16.7)$$

The rounding function is used for discrete problems and rounds the value of design variables to the nearest discrete available value. These water molecules are evaluated based on the objective function. For the molecule i ($i = 1, 2, \dots, nWM$), if the newly generated molecule i ($i = 1, 2, \dots, nWM$) is better than the old one, it will replace it. The best water molecule (best-WM) is returned.

Step 5: Terminating Condition

If the number of iterations of the algorithm (t) becomes larger than the maximum number of iterations (t_{max}), the algorithm terminates. Otherwise go to Step 2.

16.3 Water Evaporation Optimization with Mixed Phases

The basic WEO algorithm updates candidate designs in two independent sequential phases: monolayer and droplet evaporation phases. Design updating strategies selected in each phase are consistent with the local and global search ability of the algorithm. The accelerated version of WEO involving the simultaneous use of both phases is now described.

Two scenarios are developed for considering simultaneous evaporation phases based on (1) water molecules' distances and (2) objective function values of molecules. First scenario is physically based and interpreted from water evaporation phenomena. The second one is considered in a biased way without any physics-based interpretation to investigate the possible improvements of the algorithm. It should be noted that the accelerated WEO is presented here for minimization problems. If the first scenario is adopted, after each individual has been evaluated, water molecules are sorted in ascending order based on their distance ($dist_i$) from the worst current water molecule ($worst-WM$) by the following equation:

$$dist_i = |worstWM - WM_i|, \quad i = 1, 2, \dots, nWM \tag{16.8}$$

If the second scenario is adopted, water molecules are sorted in ascending order based on the objective function value. For both scenarios, water molecules are divided in two groups with the same number of molecules ($nWM/2$) based on the distance from the worst current molecule and objective function value, respectively. In the case of the first scenario, the first half of water molecules in each optimization iteration will be updated by the droplet evaporation probability, and the second half are evaporated based on the monolayer evaporation probability. The second scenario would be the contrary of first one: the groups of molecules will be updated by monolayer and droplet evaporation probability, respectively.

Our simulation results show that the first scenario is effective and successful in enhancing the convergence rate, considering three variants (basic WEO, WEO with mixed phases using the first scenario, and WEO with mixed phases using the second scenario). Table 16.1 presents the statistical results obtained for 20 independent optimization runs carried out from different initial populations randomly generated by the three variants of WEO for the first test problem (a spatial 25-bar truss) solved in this study. Average penalized weight optimized histories obtained by these three variants for 20 independent optimization runs starting from a different population randomly generated are depicted in Fig. 16.3 for this test case. It appears that the accelerated WEO using the first scenario is efficient in enriching the WEO to work with less number of structural analyses.

According to the previous paragraph and because of the physical interpretation of the first scenario, this strategy is developed here. Figures 16.4(a) and (b) show monolayer evaporation flux variation with different substrate energy and droplet evaporation probability variation with different contact angles, respectively. The rationale behind the first scenario is that the worst water molecule and molecules nearby are updated globally ($0.6 \leq J(\theta) \leq 1.0$) by the droplet evaporation probability (*DEP*) and the molecules far from the worst molecule are updated locally ($0.03 \leq exp(E_{sub}) \leq 0.6$) with the monolayer evaporation flux (*MEP*). The second scenario also is inspired by this rationale but greedily divides the molecules for local and global search using the criteria of objective function value in a way that better molecules are updated using *MEP*, and others are updated via *DEP*. Molecules get closer together with the progress of algorithm. Consequently, random

Table 16.1 Comparison of relative efficiency of three WEO variants for the 25-bar problem

Algorithm	Weight (lb)			Difference best–average solution (%)	Difference best–worst solution (%)	SD
	Best	Average	Worst			
WEO-basic	484.854	485.598	489.158	0.15	0.73	1.149
WEO-scenario 1	484.854	485.252	487.989	0.08	0.56	0.715
WEO-scenario 2	484.854	487.584	509.351	0.56	4.27	6.119

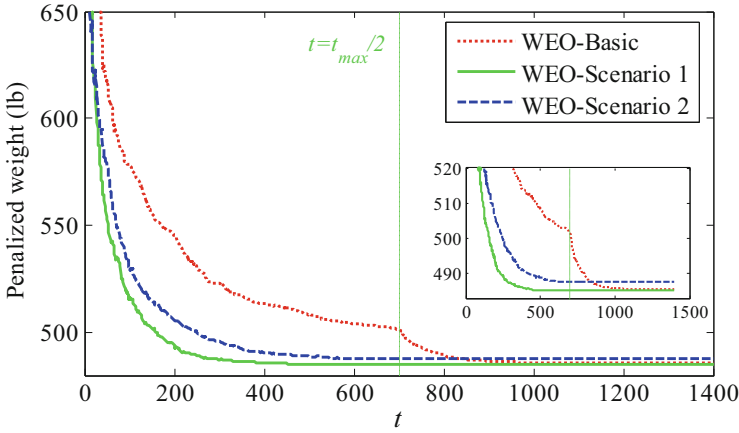


Fig. 16.3 Convergence curves recorded for the spatial 25-bar truss using three variants of WEO

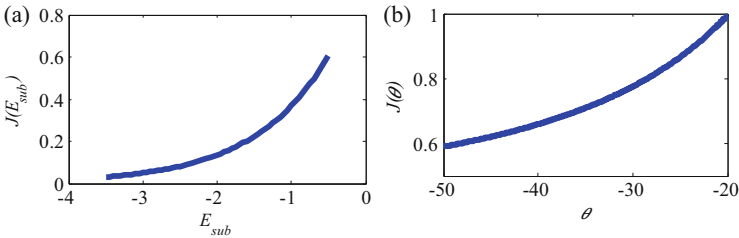


Fig. 16.4 (a) Monolayer evaporation flux with different substrate energy and (b) droplet evaporation flux with different contact angles [3, 4]

permutation based step size (Eq. (16.6)) used for generating evaporated molecules, along with different evaporation mechanisms, will enhance global and local search capability. It should be noted that in the basic WEO, utilizing the random permutation based step size (Eq. (16.6)), all molecules are updated globally using MEP and locally using DEP in the first and second half of the optimization process, respectively.

The steps involved in the first scenario of the accelerated WEO are as follows:

Step 1: Initialization

This step is the same as the one described in the basic WEO. Additionally the worst water molecule (*worst-WM*) will be monitored in this step.

Step 2: Generating Water Evaporation Matrix

Calculate the distance vector between all water molecules and the worst current one using Eq. (16.8). Sort molecules based on their distance values in an ascending order. Generate the *DEP* matrix for updating the first half of the molecules (*droplet-WM*) using the droplet evaporation rule (Eq. (16.5)). Generate the *MEP* matrix for

```

for  $i=1:nWM$ 
     $Dist(i)=norm(WM(i,:)-worst-WM)$ ;
end
 $[a,b]=sort(Dist)$ ;

for  $i=1:nWM/2$ 
     $droplet-WM(i,:)=WM(b(i),:)$ ;
end
Generate the corresponding  $\theta$  vector and  $DEP$  matrix using Eqs. (16.4) and (16.5),
respectively.

for  $i=1:nWM/2$ 
     $monolayer-WM(i,:)=WM(b(size(nWM/2+i),:))$ ;
end
Generate the corresponding  $E_{sub}$  vector and  $MEP$  matrix using Eqs. (16.2) and (16.3),
respectively.

for  $i=1:size(WM,1)$ 
    if  $i \leq nWM/2$ 
         $MDEP(b(i),:)=DEP(i,:)$ ;
    else
         $MDEP(b(i),:)=MEP(i-size(WM,1)/2,:)$ ;
    end
end
end

```

Fig. 16.5 Pseudo code for generating the mixed evaporation matrix ($MDEP$)

updating the next half of the molecules ($monolayer-WM$) based on the monolayer evaporation probability (Eq. (16.3)). In order to generate droplet and evaporation probability matrices, it is also necessary to define the corresponding contact angle vector (Eq. (16.4)) and substrate energy vector (Eq. (16.2)), respectively. It should be noted that in the accelerated version of WEO droplet and evaporation probability matrices and corresponding substrate energy and contact angle vectors include $nWM/2$ rows. The mixed evaporation matrix ($MDEP$) including evaporation vectors of each molecule is assembled from MEP and DEP matrices. The pseudo code shown in Fig. 16.5 can be used for generating the mixed evaporation matrix ($MDEP$).

Step 3: Generating Random Permutation Based Step Size Matrix

A random permutation based step size matrix is generated using Eq. (16.6).

Step 4: Generating Evaporated Water Molecules and Updating the Matrix of Water Molecules

The evaporated set of water molecules $WM^{(t+1)}$ is generated by adding the product of step size matrix and mixed evaporation probability matrix ($MDEP$) to the current set of molecules $WM^{(t)}$ according to:

$$WM^{(t+1)} = Round \left(WM^{(t)} + S \times MDEP^{(t)} \right) \quad (16.9)$$

The rounding function is used for discrete problems and rounds the value of design variables to the nearest discrete available value. These water molecules are evaluated based on the objective function. For the molecule i ($i = 1, 2, \dots, nWM$), if the newly generated molecule is better than the current one, it should be replaced. The best water molecule is returned in this step.

Step 5: Terminating Condition

If the number of iterations of the algorithm (t) becomes larger than the maximum number of iterations (t_{max}), the algorithm terminates. Otherwise go to Step 2.

16.4 Test Problems and Optimization Results

Finding optimum design of the skeletal structures is an effective way to test new optimization algorithms as these problems may include many design variables and constraints as well as large search spaces. The developed accelerated version of WEO algorithm is tested on four well-known discrete weight minimization problems: 25-bar spatial truss, 72-bar spatial truss, 3-bay 15-story frame, and 3-bay 24-story frame.

These test cases included 8, 16, 11, and 20 sizing variables, respectively. In truss problems with discrete variables, search space is defined using a series of discrete values between two lower and upper bounds. The steel members used for the design of steel frames consist of 267 W -shaped sections from the LRFD–AISC database, from $W44 \times 335$ to $W4 \times 13$. These sections with their properties are used to prepare a design pool. The sequence numbers assigned to this pool that are sorted with respect to the area of sections are considered as design variables. In other words, the design variables represent a selection from a set of integer numbers between 1 and the number of sections. Truss problems are designed under stress and nodal displacement constraints. Statement of the truss weight minimization problem can be found in [8]. Strength constraints of AISC load and resistance factor design (LRFD) specification [9] and displacement constraints are imposed on frames. Statement of the planar frame weight minimization problem can be found in [10]. In order to handle optimization constraints, a penalty approach was utilized in this study by introducing the following pseudo cost function:

$$f_{cost}(\{X\}) = (1 + \varepsilon_1 \cdot \nu)^{\varepsilon_2} \times W(\{X\}) \quad (16.10)$$

where $\{X\}$ is the set of design variables; $W(\{X\})$ is the weight of the structure; and ν is the total constraint violation. Constants ε_1 and ε_2 must be selected considering the exploration and the exploitation rate of the search space. In this study, ε_1 was set equal to one while ε_2 was selected so as to decrease the total penalty yet reducing

cross-sectional areas. Thus, ε_2 increased from the value of 1.5 set in the first steps of the search process to the value of 3 set toward the end of the optimization process. It should be noted that in the discrete structural optimization problems the method selected for constraints handling affects the algorithms in the aspect of accuracy. Here we used the penalty approach as one of the simplest methods and with fixed constants for all test problems according to previous studies.

WEO algorithm is compared in two variants: basic WEO and WEO with mixed phases. In order to better evaluate performance of the developed accelerated version, the most effective available state-of-the-art metaheuristic optimization methods based on the authors knowledge are used here as basis of comparison. Since the search process is governed by random rules, each test problem was solved by carrying out 20 independent optimization runs considering different initial populations to obtain statistically significant results. During each run, the maximum number of structural analyses (NSA) of 14×10^3 is used. Other parameters are as follows [4, 5]: the number of water molecules, $nWM = 10$; minimum and maximum value of monolayer evaporation probabilities, $MEP_{min} = 0.03$ and $MEP_{max} = 0.6$; and minimum and maximum value of droplet evaporation probabilities, $DEP_{min} = 0.6$ and $DEP_{max} = 1$. The WEO is coded in the MATLAB software environment. Structural analyses entailed by the optimization process were performed by means of the direct stiffness method.

16.4.1 A Spatial 25-Bar Tower Truss

The 25-bar transmission tower is used widely in structural optimization to verify various metaheuristic algorithms. Truss geometry including node and element numbering is shown in Fig. 16.6. The material density is 0.1 lb/in^3 and the modulus of elasticity is 10^7 psi . Twenty-five members are categorized into eight groups as follows: (1) A_1 , (2) A_2 – A_5 , (3) A_6 – A_9 , (4) A_{10} – A_{11} , (5) A_{12} – A_{13} , (6) A_{14} – A_{17} , (7) A_{18} – A_{21} , and (8) A_{22} – A_{25} . The problem is described in detail in [8].

Table 16.2 tabulates the best optimized designs found by basic WEO and accelerated WEO with mixed phases based on the first scenario. All optimized designs are fully feasible. Selected algorithms to evaluate the accelerated WEO are artificial bee colony (ABC) [11], mine blast algorithm (MBA) [12], cuckoo search (CS) [8], and colliding bodies optimization (CBO) [13]. As it is clear, the accelerated WEO requires only half of the NSA of the basic WEO. Accelerated WEO needs twice the NSA needed for MBA and CS and needs less NSA compared to the CBO. The most interesting point is that the accelerated WEO not only is modified significantly in terms of accuracy but also shows better performance in terms of the computational time. The mean optimized weight found by the accelerated WEO is lower than those found by the basic WEO. However, the basic WEO can reach the same average results by considering higher NSA (20×10^3).

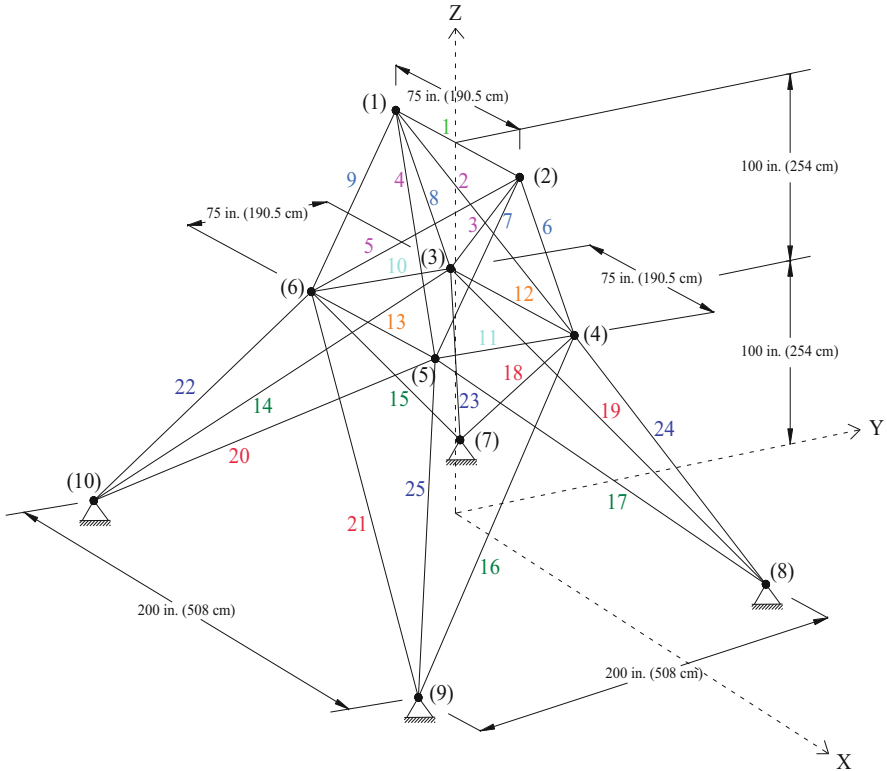


Fig. 16.6 Schematic of the spatial 25-bar truss

Table 16.2 Comparison of optimization results obtained by WEO and some other metaheuristic algorithms for the 25-bar tower problem

Element group	Optimal cross-sectional areas (in ²)					
	ABC [11]	MBA [12]	CS [8]	CBO [13]	Basic WEO Present study [5]	WEO-scenario 1
1	0.1	0.1	0.1	0.1	0.1	0.1
2	0.3	0.3	0.30	0.3	0.3	0.3
3	3.4	3.4	3.4	3.4	3.4	3.4
4	0.1	0.1	0.1	0.1	0.1	0.1
5	2.1	2.1	2.1	2.1	2.1	2.1
6	1.0	1.0	1.0	1.0	1.0	1.0
7	0.5	0.5	0.5	0.5	0.5	0.5
8	3.4	3.4	3.4	3.4	3.4	3.4
Best weight (lb)	484.85	484.85	484.85	484.85	484.85	484.85
Mean weight (lb)	485.05	484.89	485.01	486.87	485.598	485.252
NSA	NA	2150	2000	7050	9610	5060

16.4.2 A Spatial 72-Bar Truss

For the spatial 72-bar truss structure shown in Fig. 16.7, the material density is 0.1 lb/in^3 and the modulus of elasticity is 10^7 psi. The 72 bars are categorized into 16 groups using symmetry: (1) A_1 – A_4 , (2) A_5 – A_{12} , (3) A_{13} – A_{16} , (4) A_{17} – A_{18} , (5) A_{19} – A_{22} , (6) A_{23} – A_{30} , (7) A_{31} – A_{34} , (8) A_{35} – A_{36} , (9) A_{37} – A_{40} , (10) A_{41} – A_{48} , (11) A_{49} – A_{52} , (12) A_{53} – A_{54} , (13) A_{55} – A_{58} , (14) A_{59} – A_{66} , (15) A_{67} – A_{70} , and (16) A_{71} – A_{72} . The structure must be designed for multiple loading conditions. The problem is described in detail in [8].

Table 16.3 presents the statistical results obtained for 20 independent runs carried out from different initial populations randomly generated by the three

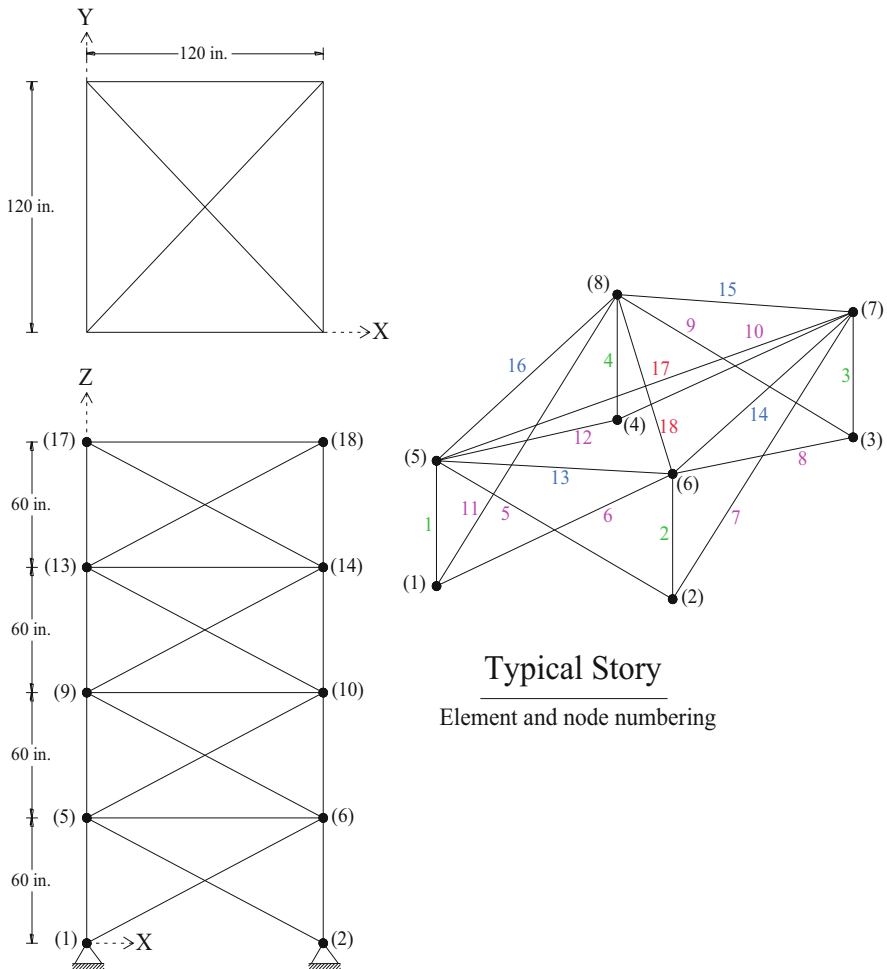


Fig. 16.7 Schematic of the spatial 72-bar truss

Table 16.3 Robustness and reliability analyses of WEO in 72-bar truss problem

Algorithm	Weight (lb)			Difference best–average solution (%)	Difference best–worst solution (%)	SD
	Best	Average	Worst			
WEO-basic	389.33	391.20	395.84	0.48	1.17	1.81
WEO-scenario 1	389.33	390.94	397.74	0.41	1.71	1.93
WEO-scenario 2	389.46	391.75	399.34	0.58	1.90	2.18

variants of WEO. Average penalized weight histories obtained by the basic and accelerated WEO for the 20 independent optimization runs are shown in Fig. 16.8. It appears that accelerated WEO with the first scenario allows to reduce the computational cost of optimization search compared to the basic WEO.

Table 16.4 lists the best optimized designs found by basic WEO and accelerated WEO with mixed phases based on the first scenario. All optimized designs are fully feasible. Both variants of WEO are compared with imperialist competitive algorithm (ICA) [14], CS [8], and CBO [15]. Accelerated WEO found the lightest design overall but needs 7500 NSA, nearly twice the number of analyses needed by other algorithms. It should be noted that the NSA required by accelerated WEO to find intermediate designs better than the optimized designs found by ICA, CS, and CBO (391.26 lb) is equal to 6200. Basic WEO also can find the lightest design, but it is clear that considering mixing phases formulation enables WEO to work with less computational cost (30 %). The most interesting point is that WEO with mixed phases results in averagely lightest optimum design than the basic WEO. It should be noted that the basic WEO can yield the same average results considering more NSA (20×10^3) as the stopping criteria.

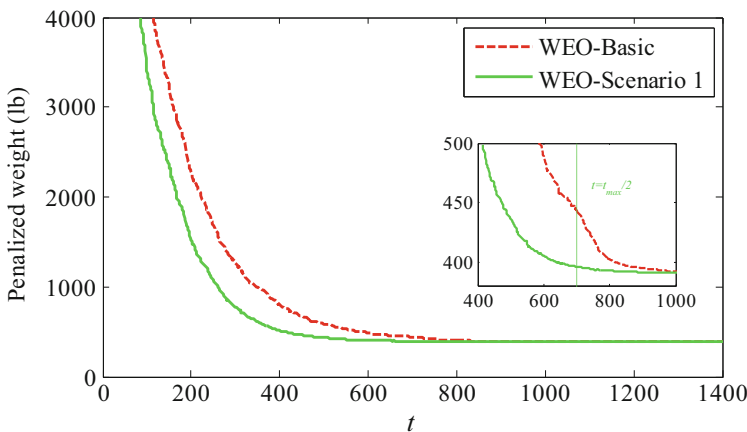


Fig. 16.8 Convergence curves recorded for the 72-bar truss problem

Table 16.4 Comparison of optimization results obtained by WEO and some other metaheuristic algorithms for the 72-bar truss problem

Element group		Optimal cross-sectional areas (in ²)				
		ICA [14]	CS [8]	CBO [15]	Basic WEO Present work [5]	Accelerated WEO
1	A ₁ ~ A ₄	1.99	1.800	1.62	1.99	1.99
2	A ₅ ~ A ₁₂	0.442	0.563	0.563	0.563	0.563
3	A ₁₃ ~ A ₁₆	0.111	0.111	0.111	0.111	0.111
4	A ₁₇ ~ A ₁₈	0.141	0.111	0.111	0.111	0.111
5	A ₁₉ ~ A ₂₂	1.228	1.266	1.457	1.228	1.228
6	A ₂₃ ~ A ₃₀	0.602	0.563	0.442	0.442	0.442
7	A ₃₁ ~ A ₃₄	0.111	0.111	0.111	0.111	0.111
8	A ₃₅ ~ A ₃₆	0.141	0.111	0.111	0.111	0.111
9	A ₃₇ ~ A ₄₀	0.563	0.563	0.602	0.563	0.563
10	A ₄₁ ~ A ₄₈	0.563	0.442	0.563	0.563	0.563
11	A ₄₉ ~ A ₅₂	0.111	0.111	0.111	0.111	0.111
12	A ₅₃ ~ A ₅₄	0.111	0.111	0.111	0.111	0.111
13	A ₅₅ ~ A ₅₈	0.196	0.196	0.196	0.196	0.196
14	A ₅₉ ~ A ₆₆	0.563	0.602	0.602	0.563	0.563
15	A ₆₇ ~ A ₇₀	0.307	0.391	0.391	0.391	0.391
16	A ₇₁ ~ A ₇₂	0.602	0.563	0.563	0.563	0.563
Best weight (lb)		392.84	389.87	391.07	389.33	389.33
Mean weight (lb)		N/A	N/A	403.71	391.20	390.94
NSA		4500	5450	4500	10,510	7490

16.4.3 A 3-Bay 15-Story Frame

The configuration, service loading conditions, and numbering of member groups for the 3-bay 15-story frame are shown in Fig. 16.9. All 64 columns are grouped into nine groups while all 45 beams are considered as a beam element group. All element groups are chosen from 267 W-sections. Performance constraints, material properties, and other conditions are available in [10].

Table 16.5 compares the accelerated WEO with basic WEO and other state-of-the-art metaheuristic algorithms which efficiently solved this test problem: ICA [14], charged system search (CSS) [16], and enhanced version of CBO (ECBO) [13]. All optimized designs are fully feasible. The lightest design is obtained by ECBO algorithm which needs nearly the same NSA as accelerated WEO. Accelerated WEO is better than basic WEO considering the best run and average of 20 independent runs starting from different populations randomly

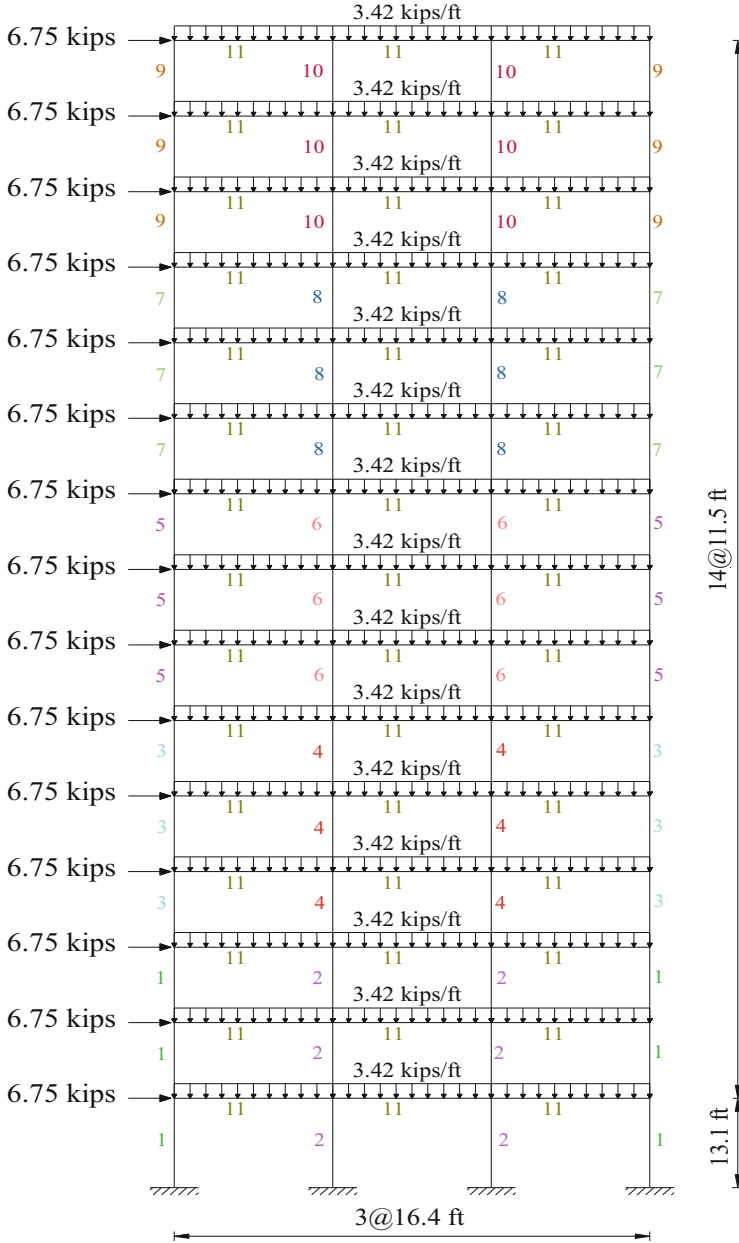


Fig. 16.9 Schematic of the 3-bay 15-story frame

generated, and computational cost. Accelerated WEO leads to lighter design than ICA and CSS but needs nearly twice NSA.

Table 16.5 Comparison of optimization results obtained by WEO and some other metaheuristic algorithms for the 3-bay 15-story frame problem

Element group	Optimal cross-sectional areas (W types)				
	ICA	CSS	ECBO	Basic WEO	Accelerated WEO
	[14]	[16]	[13]	Present study [5]	
1	W24 × 117	W21 × 147	W14 × 99	W14 × 90	W14 × 99
2	W21 × 147	W18 × 143	W27 × 161	W36 × 170	W27 × 161
3	W27 × 84	W12 × 87	W27 × 84	W30 × 90	W27 × 84
4	W27 × 114	W30 × 108	W24 × 104	W24 × 104	W24 × 104
5	W14 × 74	W18 × 76	W14 × 61	W24 × 68	W14 × 61
6	W18 × 86	W24 × 103	W30 × 90	W12 × 87	W30 × 90
7	W12 × 96	W21 × 68	W14 × 48	W8 × 48	W16 × 50
8	W24 × 68	W14 × 61	W14 × 61	W14 × 68	W21 × 68
9	W10 × 39	W18 × 35	W14 × 30	W10 × 33	W14 × 34
10	W12 × 40	W10 × 33	W12 × 40	W16 × 45	W8 × 35
11	W21 × 44	W21 × 44	W21 × 44	W21 × 44	W21 × 44
Best weight (lb)	93,846	92,723	86,986	88,710.97	87,537.96
Mean weight (lb)	N/A	N/A	88,410	90,649.49	88,893.09
NSA	6000	5000	9000	13,580	10,670

Figure 16.10 compares average convergence histories for different formulations of the WEO, CSS [16], and ICA [14] algorithms. It should be noted that the weight optimization convergence history is monitored because of the available results in the literature. For more clarity, the y-axis is in units of kips and its upper bound is limited to 300. Best convergence rate is obtained by CSS algorithm. As it is clear, the new formulation based on the scenario 1 makes the WEO competitive with other algorithms also in terms of convergence rate.

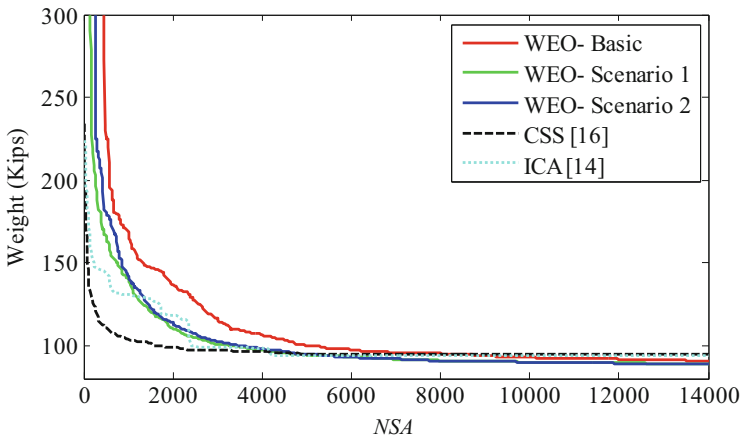


Fig. 16.10 Average convergence histories of algorithms for the 3-bay 15-story frame problem

16.4.4 A 3-Bay 24-Story Frame

The 3-bay 24-story frame optimized in the last test problem is shown in Fig. 16.11. The structure is comprised of 168 members that are collected in 20 groups (16 column groups and four beam groups). Each of the four beam element groups is chosen from all 267 W-shapes, while the 16 column element groups are limited to W14 sections (37 W-shapes). Additional information about the problem can be found in [13].

Table 16.6 compares the accelerated WEO with basic WEO and other state-of-the-art metaheuristic algorithms which efficiently solved this test problem: harmony search (HS) [17], CSS [16], and ECBO [13]. All optimized designs are fully feasible. ECBO shows the best performance. Accelerated WEO performs better than the basic WEO and HS, in relation to both accuracy and computational cost, and leads to a lighter design than that of the CSS. However, the present algorithm requires more structural analyses to complete the optimization process.

Figure 16.12 compares average convergence histories for different formulations of the WEO, CSS [16], and HS [17] algorithms. For more clarity, the y-axis is in the units of kips and its upper bound is limited to 700. Best convergence rate is again obtained by CSS algorithm. The new formulation based on the scenario 1 makes WEO enough competitive in terms of convergence rate.

16.5 Concluding Remarks

This chapter presented an accelerated version of water evaporation optimization algorithm (WEO). WEO is a recently developed population physics-based metaheuristic algorithm which follows the well-known rules governing the evaporation process of water molecules from a solid surface with different wettability. In the basic WEO, molecules are updated globally and locally, respectively, in two independent sequential phases: monolayer and droplet evaporation phases. It was shown that computational cost of the WEO optimizations can be reduced by simultaneously using the two phases. For that purpose, we developed two scenarios based on the (1) water molecules' distance and (2) objective function value of molecules.

The new algorithm was successfully tested in four well-known discrete structural optimization problems. The first scenario which is in a good agreement with physical aspects of the water evaporation is more efficient than the second one and hence developed in this study. It turns out that the developed framework provides a significant enhancement in convergence rate compared to the basic WEO. However, further investigations should be carried out in order to reduce the computational cost of the accelerated WEO in frame problems with respect to other metaheuristic algorithms such as ECBO.

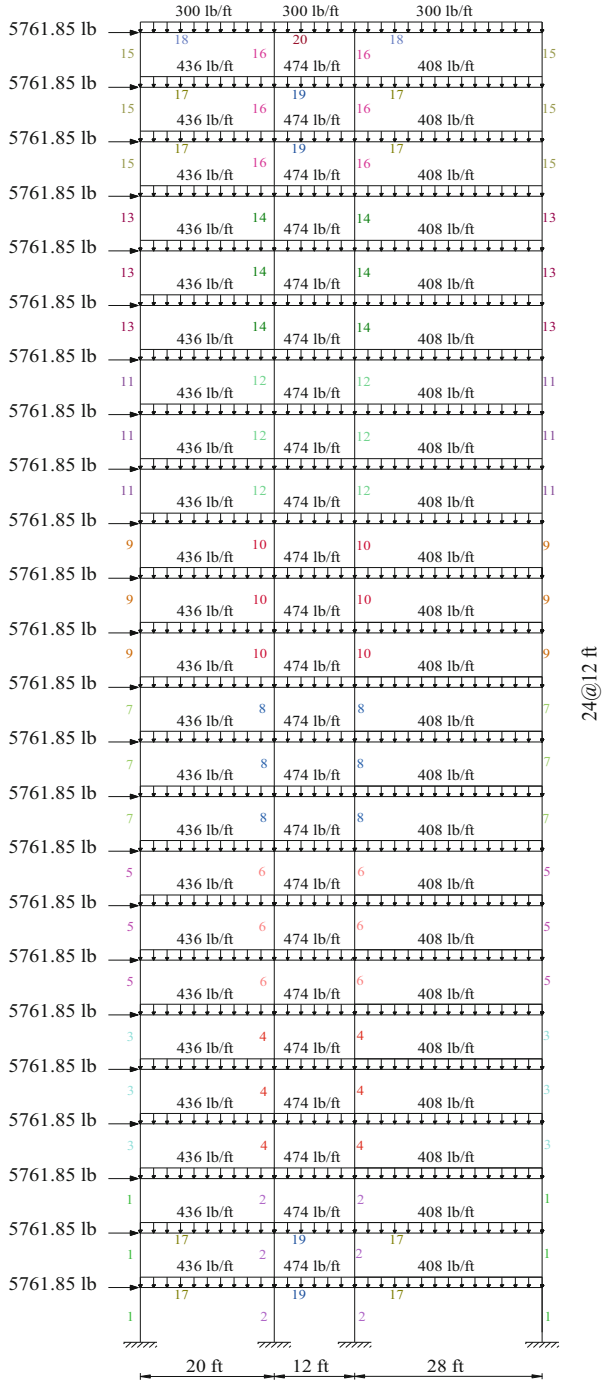


Fig. 16.11 Schematic of the 3-bay 24-story frame

Table 16.6 Comparison of optimization results obtained by WEO and some other metaheuristic algorithms for the 3-bay 24-story frame problem

Element group	Optimal cross-sectional areas (W types)				
	HS	CSS	ECBO	Basic WEO	Accelerated WEO
	[17]	[16]	[13]	Present study [5]	
1	W14 × 176	W14 × 176	W14 × 145	W14 × 132	W14 × 159
2	W14 × 145	W14 × 145	W14 × 99	W14 × 109	W14 × 132
3	W14 × 176	W14 × 145	W14 × 132	W14 × 109	W14 × 99
4	W14 × 132	W14 × 132	W14 × 99	W14 × 90	W14 × 109
5	W14 × 132	W14 × 109	W14 × 99	W14 × 61	W14 × 68
6	W14 × 109	W14 × 109	W14 × 99	W14 × 38	W14 × 38
7	W14 × 109	W14 × 90	W14 × 90	W14 × 38	W14 × 30
8	W14 × 82	W14 × 82	W14 × 82	W14 × 22	W14 × 22
9	W14 × 82	W14 × 74	W14 × 74	W14 × 109	W14 × 90
10	W14 × 61	W14 × 68	W14 × 68	W14 × 109	W14 × 99
11	W14 × 74	W14 × 61	W14 × 38	W14 × 99	W14 × 99
12	W14 × 48	W14 × 43	W14 × 61	W14 × 90	W14 × 74
13	W14 × 34	W14 × 34	W14 × 38	W14 × 82	W14 × 68
14	W14 × 30	W14 × 34	W14 × 30	W14 × 68	W14 × 61
15	W14 × 22	W14 × 34	W14 × 22	W14 × 34	W14 × 34
16	W14 × 22	W14 × 22	W14 × 22	W14 × 22	W14 × 22
17	W30 × 90	W30 × 90	W30 × 90	W30 × 90	W30 × 90
18	W10 × 22	W21 × 50	W6 × 15	W6 × 15	W8 × 18
19	W18 × 40	W21 × 48	W24 × 55	W24 × 55	W24 × 55
20	W12 × 16	W12 × 19	W6 × 8.5	W6 × 8.5	W6 × 8.5
Best weight (lb)	214,860	212,364	201,618	202,626.00	202,194.02
Mean weight (lb)	222,620	215,226	209,644	204,954.03	203,412.88
NSA	13,924	5500	2800	13,510	11,300

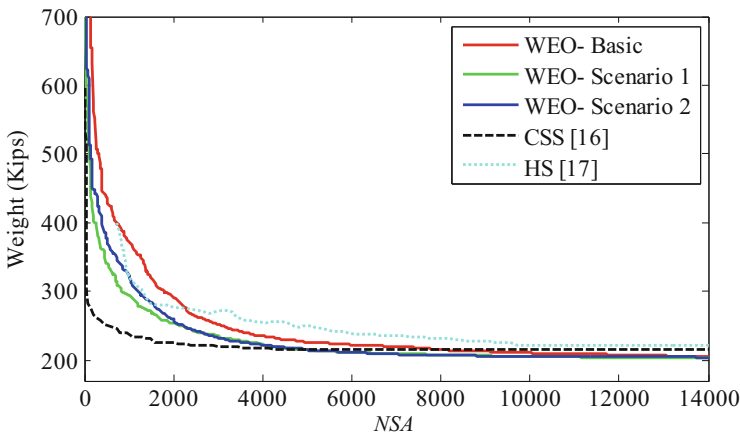


Fig. 16.12 Average convergence histories of algorithms for the 3-bay 24-story frame problem

References

1. Xiong N, Molina D, Ortiz ML, Herrera F (2015) A walk into metaheuristics for engineering optimization: principles, methods and recent trends. *Int J Comput Intell Syst* 8:606–636
2. Talbi E-G (2009) *Metaheuristics: from design to implementation*. Wiley, Hoboken, NJ
3. Kaveh A, Bakhshpoori T (2016) Water Evaporation Optimization: a novel physically inspired optimization algorithm. *Comput Struct* 167:69–85
4. Kaveh A, Bakhshpoori T (2016) A novel metaheuristic for continuous structural optimization: Water Evaporation Optimization. *Struct Multidiscip Optim* 54(1):23–43
5. Kaveh A, Bakhshpoori T (2016) An accelerated water evaporation optimization formulation for discrete optimization of skeletal structures. *Comput Struct* 177:218–228
6. Zarei G, Homaei M, Liaghat AM, Hoorfar AH (2010) A model for soil surface evaporation based on Campbell's retention curve. *J Hydro* 380:356–361
7. Wang S, Tu Y, Wan R, Fang H (2012) Evaporation of tiny water aggregation on solid surfaces with different wetting properties. *J Phys Chem B* 116:13863–13867
8. Kaveh A, Bakhshpoori T (2013) Optimum design of space trusses using cuckoo search algorithm with Lévy flights. *Iranian J Sci Technol Trans Civil Eng* 37:1–15
9. AISC (2001) *Manual of steel construction: load and resistance factor design*. AISC, Chicago, IL
10. Kaveh A, Bakhshpoori T (2013) Optimum design of steel frames using Cuckoo Search algorithm with Lévy flights. *Struct Des Tall Spec Build* 22:1023–1036
11. Sonmez M (2011) Discrete optimum design of truss structures using artificial bee colony algorithm. *Struct Multidiscip Optim* 43:85–97
12. Sadollah A, Bahreininejad A, Eskandar H, Hamdi M (2012) Mine blast algorithm for optimization of truss structures with discrete variables. *Comput Struct* 102–103:49–63
13. Kaveh A, Ilchi Ghazaan M (2015) A comparative study of CBO and ECBO for optimal design of skeletal structures. *Comput Struct* 153:137–147
14. Kaveh A, Talatahari S (2010) Optimum design of skeletal structures using imperialist competitive algorithm. *Comput Struct* 88:1220–1229
15. Kaveh A, Mahdavi VR (2014) Colliding Bodies Optimization method for optimum discrete design of truss structures. *Comput Struct* 139:43–53
16. Kaveh A, Talatahari S (2012) Charged system search for optimal design of frame structures. *Appl Soft Comput* 12:382–393
17. Degertekin SO (2008) Optimum design of steel frames using harmony search algorithm. *Struct Multidiscip Optim* 36:393–401