Periodica Polytechnica Civil Engineering

Competition of Three Chaotic Meta-heuristic Algorithms with Physical Inspiration for Optimal Design of Truss Structures

Ali Kaveh[1]*, Hosein Yosefpoor[2]

[1] School of Civil Engineering, Iran University of Science and Technology, P. O. B. 16765-163, Narmak, Tehran 16846-13114, Iran
[2] Department of Technics and Engineering, Maragheh Branch, Islamic Azad University, P. O. B. 345, Maragheh 5519747591, Iran
* Corresponding author, e-mail: alikaveh@iust.ac.ir

Received: 11 March 2024, Accepted: 12 June 2024, Published online: 26 June 2024

Abstract

Chaos maps create a significant improvement in the optimization results of meta-heuristic algorithms by creating a balance between the stages of exploration and exploitation. The optimization algorithms of structures are strongly non-linear and non-convex, having several local optima. Chaotic functions, while creating chaotic jumps, provide the conditions for escaping from local optima to global optima. Most of the meta-heuristic algorithms fall into the trap of local optima and suffer some kind of premature convergence. In this paper, by forming three scenarios, chaos functions can be embedded into the exploration, exploitation or both stages at the same time, and improve the results of meta-heuristic algorithms. The considered algorithms are inspired by physical phenomena, with the possibility of accessing classical and regular relations, the effectiveness of chaos functions in meta-heuristic algorithms are increased. Nowadays, chaotic algorithms are widely utilized by researchers and are considered as a challenging topic. In the present research, the effects of logistic and Gaussian chaos functions on the optimization results of three physically inspired meta-heuristic algorithms are investigated. These algorithms include Chaotic Thermal Exchange Optimization (CTEO), Chaotic Big Bang-Big Crunch (CBB-BC), and Chaotic Tug-of-War Optimization (CTWO).

Keywords

chaos maps, meta-heuristic algorithms, exploration, exploitation, premature convergence, local optima, global optima

1 Introduction

The attractive features of truss structures have increased the use of this group of structures in engineering fields. Thus, nowadays engineers are witnessing the increasing popularity of this group of structures. Covering large openings with beautiful effects, lightness and economy, ease of production and implementation, speed of installation are some of the special features of trusses. In addition to these cases, the large number of members of these structures and their mass production in different uses such as sheds, industrial buildings, airplane hangars, power transmission towers and pedestrian bridges necessitate the optimization of these structures in order to saving resources and costs has been justified. Based on this, in recent decades, the optimization of structures has been the focus of most researchers. Therefore, for engineering designs, in addition to components such as stress, deformation, thinning and buckling, the variables related to cost and efficiency and applying the economic aspects of the design should also be considered. One of the primary methods to achieve this goal is to use traditional gradient-based methods and check the derivative of the objective function. But these solutions always had their own limitations and problems. In most engineering problems, one did not have an explicit relation of the objective function, and had to enter into the complex discussion of partial derivatives and strongly "non-linear and non-convex" states. In some cases, after long calculations, the results lead to local optima. Also, with the increase in the number of decision variables, the complexity of the calculations increased exponentially and terribly. In these algorithms, instead of dealing with the derivatives of the objective function, it itself is evaluated and with inspiration from natural and physical phenomena, it obtains an improving process in successive iterations. The main reason for this choice is the characteristics of natural phenomena that if there is a powerful place in mind, nature has done it in the best way [1]. The first choice of researchers for inspiration is the genetic evolution of living things over millions of years from the

Cite this article as: Kaveh, A., Yosefpoor, H. "Competition of Three Chaotic Meta-heuristic Algorithms with Physical Inspiration for Optimal Design of Truss Structures", Periodica Polytechnica Civil Engineering, 2024. https://doi.org/10.3311/PPci.36853

beginning of their life. In this evolution, the characteristics of living things are improved by the act of crossing and mutation, so that by adapting as much as possible to the surrounding environment, they win in the competition with other living things in the courtyard of life. Examples of these algorithms include Genetic Algorithm (GA) [2] and Differential Evolution (DE) [3]. The second inspiration is to take advantage of the intelligence of animal swarms and their nature in search and access to food. The constituent factors of this inspiration include population, cooperation, communication, information exchange, information flow and self-organization. These components are evident in the swarm life of birds, fish, ants and other animals. A number of these algorithms include Particle Swarm Optimization (PSO) [4], Artificial Bee colony (ABC) [5], Cyclical Parthenogenesis Algorithm (CPA) [6]. Inspired by physical laws, the third group of meta-heuristic algorithms are formed. Examples of these algorithms include: Water Evaporation Optimization (WEO) [7], Thermal Exchange Optimization (TEO) [8], Big Bang-Big Crunch (BB-BC) [9], Tug-of War Optimization (TWO) [10], Charged System Search (CSS) [11], Colliding Bodies Optimization (CBO) [12], Harmony Search (HS) [13], Vibrating Particles System (VPS) [14]. Meta-heuristic algorithms with the origin of physical inspiration have regular classical relationships and are more popular among researchers. These algorithms have also played an important role in improving the optimization results of structures. Today, there is no limit to the scope of inspiration, so the fields of inspiration have expanded on a wide level and remarkable successes have been achieved in the state-of-the-art. Based on the traditional Nelder and Mead method, the Shuffled Complex Evolution (SCE-UA) [15] was proposed at the University of Arizona. The cases of inspiration are the geometric operators of contraction and reflection. The inspiration points in this improved algorithm and the Shuffled Frog-Leaping Algorithm (SFLA) [16] were proposed, which is classified as a memetic algorithm. Other algorithms, such as the Shuffled Shepherd Optimization Algorithm (SSOA) [17], Imperialist Competitive Algorithm (ICA) [18] and Teaching-Learning-Based Optimization (TLBO) [19], inspired by different behaviors, have made a significant improvement in the optimization of structures. For engineering problems, meta-heuristic methods for optimization are more successful. If these methods are compared, pests still affect them. Premature convergence, getting caught in the trap of local optima and slowing down the optimization process are among the plagues that affect meta-heuristic algorithms in standard mode. By embedding chaos functions in the exploration and exploitation parts of these algorithms, the weakness of these algorithms is largely eliminated. Lorenz has done extensive research on the performance of chaos functions. The most important features of chaos functions can be summarized in several cases. The constituent series of these functions are sensitive to the initial conditions, their dynamic state is compact, their behavior is similar to random but in practice they are deterministic, and the structure of their functions is such that they do not have an inverse [20]. In our previous research, several chaos functions were investigated where the results of Logistic and Gaussian functions improved the weakness of the exploration and exploitation in the best conditions [21–24]. Therefore, in the present research, only these functions have been incorporated. By embedding these functions in meta-heuristic algorithms, the balance between exploration and exploitation is achieved and a significant improvement in optimization can be achieved with chaotic modes.

2 Formulation of the optimization problems

The main parts of any optimization problem include the objective function, design constraints, and the bound of decision variables. For optimal design of truss structures, the objective function includes the weight of the structure, that must satisfy all the design requirements such as allowable stress, nodal displacement, slenderness constraints and buckling stress with the lowest possible weight. The main variable in determining the weight of the structure is the cross-sectional area of the members, which should be selected within the lower and upper limits of the decision variables. The general form related to these issues is defined as Eq. (1).

$$\begin{array}{lll} \text{Find} & A=\{A_1,A_2,A_3,\ldots,A_n\} & \\ \text{to Minimize} & W(A)=\sum_{i=1}^{n}\gamma_i\times A_i\times L_i & \\ \text{Subjected to} & g_j(A)\le 0; & j=1,2,3,\ldots,m \\ & h_k(A)\le 0; & k=1,2,3,\ldots,p \\ & \{A_L\}\le\{A\}\le\{A_U\} & \end{array} \tag{1}$$

According to Eq. (1), A is the cross-sectional area of the members, W is the total weight of the structure, n is the number of members of the structure, g_i and h_i are design constraints. These constraints can include stress, member slender and nodal displacement. Also, A_L and A_u are the upper and lower bounds of the decision variables. The initial form of meta-heuristic algorithms for optimization of unconstrained problems is presented. For this reason, the penalty function method with Lagrange coefficients is used in the modeling to convert the bounded function into unbounded one. In this method, if there is no violation and the answers satisfy the restrictions, the amount of the penalty will be zero. But if there is a violation of the design constraints, its value is calculated according to Eqs. (2)–(6) and included in the penalized objective function as

$$\sigma_i \le \sigma^{\max} \Rightarrow V_i = \max\left(0, \frac{\sigma_i}{\sigma^{\max}} - 1\right); \quad i = 1, 2, 3, \ldots, n \tag{2}$$

$$\delta_j \le \delta^{\max} \Rightarrow V_j = \max\left(\frac{\delta_j}{\delta^{\max}} - 1\right); \quad j = 1, 2, 3, \ldots, n \tag{3}$$

$$\lambda_k \le \lambda^{\max} \Rightarrow V_k = \max\left(0, \frac{\lambda_k}{\lambda^{\max}} - 1\right); \quad k = 1, 2, 3, \ldots, p \tag{4}$$

$$F_{\text{penalty}}(A) = 1 + \gamma_p \times \left\{ \sum_{i=1}^{m} V_i + \sum_{j=1}^{n} V_j + \sum_{k=1}^{p} V_k \right\} \tag{5}$$

$$\text{to Minimize} \quad Mer(A) = W(A) \times F_{\text{penalty}}(A) \tag{6}$$

Equations (2)–(4) are related to stresses, displacements and slenderness ratios, respectively. The penalty function is presented in Eq. (5) and the objective function is formed after the penalty (merit function) in Eq. (6).

3 Introduction of selected chaos map

The most important factor of stagnation and premature convergence in meta-heuristic algorithms is the imbalance between the two stages of exploration and exploitation. In such cases, the algorithm stagnates in the initial iterations and gets caught in the trap of local optima. The most appropriate solution to jump from local optima and transfer to global optima is to use chaos functions with different scenarios. Chaotic functions do not have any effect of random behavior, but by creating irregular behavior in the search space, it provides access to the near global optimal position. Among the other characteristics of chaos functions, the following can be mentioned: these functions are very sensitive to initial conditions and their dynamic behaviors are non-periodic, deterministic and ergodic, and the important point is that their functions do not have an inverse. These functions are established with sudden jumps near the global optimal position and create the necessary conditions to reach them. In some chaotic functions, the relation of the function is such that it converges from local minima to global minima with a very high probability. This group of chaos functions are suitable for improving algorithms that are weak in the exploration stage. In another group of chaos functions, the decision space moves towards local optima with a very high probability, this group of chaos functions is suitable for improving the exploitation conditions of algorithms. Therefore, this group of chaos functions can be embedded in algorithms that have weaknesses in the exploitation stage. In the embedding of chaos functions in metaheuristic algorithms, at least one sample from each group must be present, and in the first scenario, the chaos function is embedded in the exploration part and in the second scenario, in the exploitation part. Also, in cases where behavioral imbalance in the algorithm requires a general replacement, chaos functions are used simultaneously in both exploration and exploitation parts of the algorithm. This state is introduced as the third scenario. It can be concluded that the use of Chaotic series for Optimization in meta- heuristic Algorithms (COA) has significant advantages over other methods. In chaotic algorithms, deterministic search replaces random search. With this replacement, a chaotic jump of local optima is performed and early convergence is resolved [25]. In order to illustrate how to embed the chaotic functions in meta-heuristic algorithms, the flowchart of Fig. 1. In recent research, logistic and Gaussian chaos functions have been investigated. The first chaos function moves the search space from local optima to global optima. Therefore, it can be suitable for exploration. But the second chaos function transfers the results to local optima and is suitable for exploitation. By embedding these functions to meta-heuristic algorithms, the problem of algorithms in standard mode of any type is improved. In order to familiarize with these chaos functions, the number of their initial 100 repetitions are displayed in Fig. 2.

3.1 Logistics map

This map appears in nonlinear dynamic behaviors related to biological populations [26]. The statements of chaotic sequences in the logistic function are obtained according to Eqs. (7) and (8):

$$CHM_{k+1} = a \times CHM_k \left(1 - CHM_k\right) \tag{7}$$

$$\begin{gathered} CHM_{k+1} \in (0,1),\ CHM_k \in (0,1), \\ CHM_0 \notin (0, 0.25, 0.50, 0.75, 1) \end{gathered} \tag{8}$$

In the present studies, $a = 4$ has been utilized. The terms CHM_k and CHM_{k+1} are related to series sentences of chaos map in consecutive order.

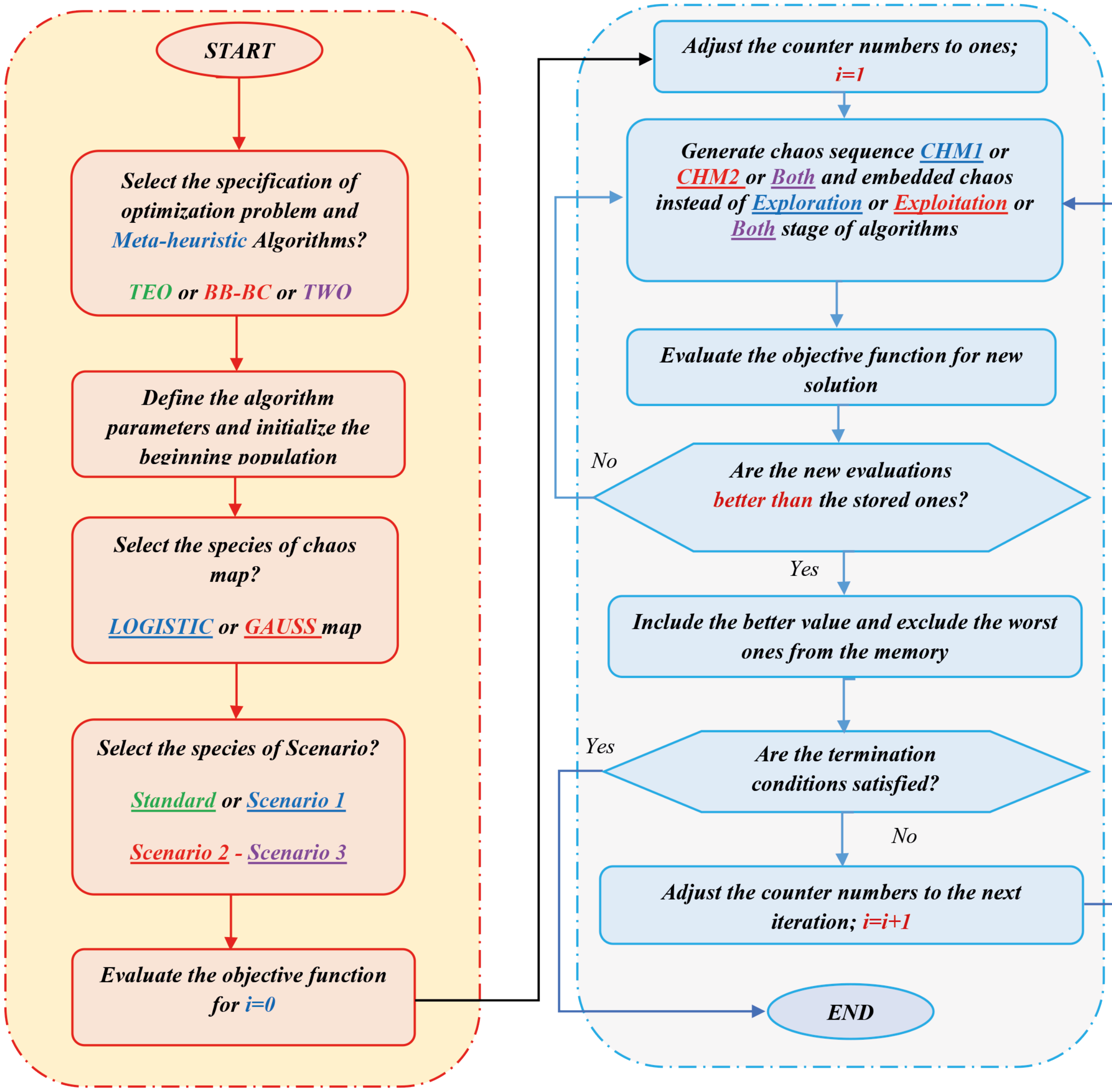
START
Select the specification of optimization problem and Meta-heuristic Algorithms?
TEO or BB-BC or TWO
Define the algorithm parameters and initialize the beginning population
Select the species of chaos map?
LOGISTIC or GAUSS map
Select the species of Scenario?
Standard or Scenario 1
Scenario 2 - Scenario 3
Evaluate the objective function for i=0
Adjust the counter numbers to ones; i=1
Generate chaos sequence CHM1 or CHM2 or Both and embedded chaos instead of Exploration or Exploitation or Both stage of algorithms
Evaluate the objective function for new solution
No
Are the new evaluations better than the stored ones?
Yes
Include the better value and exclude the worst ones from the memory
Yes
Are the termination conditions satisfied?
No
Adjust the counter numbers to the next iteration; i=i+1
END

Fig. 1 Flowchart for the chaos algorithm

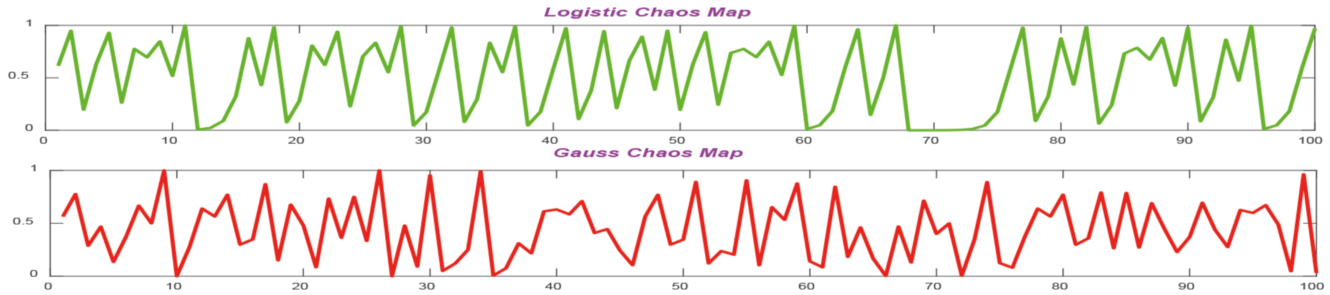

Fig. 2 The chaotic value distribution during 100 iterations

3.2. Gauss map

Using this function in nonlinear dynamic behaviors has shown good results [27]. The statements of chaotic sequences in the Gaussian function are obtained according to Eq. (9):

$$CHM_{k+1} = \begin{cases} 0 & CHM_k = 0 \\ \dfrac{1}{CHM_k} - \left[\dfrac{1}{CHM_k}\right] & CHM_k \neq 0 \end{cases}. \qquad (9)$$

The terms CHM_k and CHM_{k+1} are related to the series sentences of chaos map in a consecutive order.

4 Meta-heuristic algorithms and applying chaos functions

In 2012, Talatahari et al. [18] embedded Logistic and Gossin chaos functions in the Imperialist Competitive Algorithm (ICA) and significantly improved the results of structural optimization. For each meta-heuristic algorithm, two stages of exploration and exploitation are considered. These stages play a very important role in the convergence towards the optimal answer. In the exploration phase, points of the search space that have a special feature are selected, and in the exploitation phase, the neighborhood of the selected points is carefully examined. For comprehensive access to the search space, diversity in reviews is needed. In the standard mode of algorithms, this diversity is provided by random parameters. Therefore, using probabilistic functions with Uniform, Normal, Logistic or Levy distribution, these random parameters are selected. Studies show that random parameters play a big role in increasing or decreasing the speed of convergence. Also, in order to escape the trap of local optima and create a balance between exploration and exploitation, these random parameters should be modified. Some of the advantages of the series created by chaos functions are:

- their values are definite,
- the sentences of the series are dynamic and non-repetitive,
- the behavior of the series sentences is non-linear,
- these sentences do not converge towards a specific limit,
- and finally, the functions.

The generator of these series is not reversible [24]. Embedding chaos functions in meta-heuristic algorithms is done in different ways.

4.1 Standard Thermal Exchange Optimization (TEO)

Inspired by Newton's law of cooling, Kaveh and Dadras [8] presented a new meta-heuristic algorithm called Thermal Exchange Optimization (TEO). According to Newton's law of cooling, the rate of thermal energy dissipation between the object and the surrounding environment is proportional to their temperature difference. If the energy exchange is separated between thermal particles into two separate parts, this exchange between the first half and the second half is similar to the movement of weak responses towards optimal responses during the optimization process. In each iteration, the elitism of the algorithm is provided by allocating a memory to store the best result.

4.1.1 Basic steps in Thermal Exchange Optimization

Step 1 Formation of initial responses: This algorithm, like other population-based algorithms, begins with the introduction of initial responses. The selection range of these answers is based on the lower bound and upper bound of the decision variable. The number of populations is chosen to create heat exchange *nTO*. These initial responses are evaluation based on objective functions and penalized objective functions.

Step 2 Forming groups to start heat exchange: In this step, the formed thermal particles are sorted based on the penalized objective function and then grouped into two equal parts. The constituent members of the first part ($i = 1, 2, \ldots, nTO/2$) exchange heat with the members of the second part ($i = nTO/2 + 1, \ldots, n$). This exchange is based on Newton's law of cooling.

Step 3 Updating the new position of the particles: after the thermal exchange between the particles, their new position is determined by the following relationship.

$$\text{new}TO(i) = envTO(i) + \left(TO(i) - envTO(i)\right)\exp\left(-\beta(i)t\right) \tag{10}$$

In the presented relation, the position of the particles before the thermal exchange is *envTO*(*i*), which is transferred to a new position new*TO*(*i*) by performing the thermal exchange. One can also use the following suggestions to introduce time and the β component.

$$t = \frac{NIT_s}{\max NIT_s}, \quad \beta(i) = \frac{PFit(i)}{\max PFit(i)} \tag{11}$$

The components used in relation to time are the number of iterations in each stage NIT_s and the maximum number of iterations maxNIT_s. Also, in order to determine the β for each moment of time, the amount of the penalized target function and its maximum amount is considered.

Step 4 Probable components to prevent convergence: The previous steps are done according to deterministic relationships. At this stage, in order to escape from local optima and prevent premature convergence, two strategies of exploration and exploitation are applied in thermal exchange between particles. Embedding the Probable conditions of these two solutions is done by choosing the components of C, $C1$, $C2$ and rand command.

$$\text{new}TO = (1 - c \times rand)TO \tag{12}$$

$$C = C1 + C2 \times (1 - t) \tag{13}$$

$$C1 = \text{round}(rand), \quad C2 = \text{round}(rand) \tag{14}$$

Step 5 Applying elitism in the thermal exchange optimization algorithm: The elitism conditions in this algorithm are determined by allocating memories to store the best answers. Also, the best thermal exchange (*TO-M*), the target function (*Fit-M*) and the penalized target function (*PFit-M*) related to it are measured in each iteration with the previous results, and if the conditions improve, they are replaced.

Step 6 By checking the termination conditions, if necessary, the particle thermal exchange operation is repeated from the second step.

4.1.2 Chaos-embedded Thermal Exchange Optimization (CTEO)

By performing thermal exchange and determining the new position of each of the exchanging particles, two stages of exploration and exploitation are considered to create diversity in the space of decision variables Eqs. (12)–(14). In this research, by embedding chaos functions in random choices, a significant improvement in optimization results is achieved. Embedding chaos functions is possible with several scenarios, which is examined below:

- *Scenario 1* Embedding the chaos function in the exploration part: In this scenario, the effectiveness of the chaos function in the exploration phase is investigated. Therefore, the first chaos function *CHM*1 is replaced in Eq. (12). instead of randomly selecting the standard state. With this substitution, the equation changes as follows:

$$\text{new}TO = (1 - c \times CHM1)TO \tag{15}$$

- *Scenario 2* Embedding the chaos function in the exploitation part: In this scenario, the effectiveness of the chaos function in the exploitation stage is checked. Therefore, the second chaos function *CHM*2 is replaced to evaluate the amount of *C*1 and *C*2 in Eq. (14). instead of randomly selecting the standard state. With this substitution, the equation changes as follows:

$$C1 = \text{round}(CHM2), \quad C2 = \text{round}(CHM2) \tag{16}$$

- *Scenario 3* Embedding the chaos function in both parts simultaneously: In this scenario, the effectiveness of the chaos functions is checked in both stages. Therefore, both chaos functions are replaced simultaneously instead of random selections of the standard mode.

4.2 Standard Big Bang-Big Crunch (BB-BC) algorithm

Inspired by the energy dissipation in the transformation from an orderly state to a chaotic state, a meta-heuristic algorithm based on physics has been presented by Erol and Eksin [9]. This algorithm uses two theories related to the evolution of the universe, including the Big Bang and the Big Crunch (BB-BC), and is also known by this name. In this theory, the Big Bang stage is proposed as the model of the beginning of the world and the Big Crunch as a model for its continuation and end. Therefore, the algorithm consists of two main phases. In the Big Bang phase, due to the abundance of energy, the particles are distributed in all areas of space, but in the next phase, that is, the Big Crunch phase, the dispersed particles converge to a point based on a specific order and instructions. In this algorithm, like other population-based algorithms, first, points are randomly selected in scattered parts of the decision space. This step is responsible for exploration for the algorithm. But in the big Crunch stage, by moving towards local optima, the results are compressed in one point. Therefore, after successive repetitions, the space used in the Big Bang converges towards the points resulting from the Big Crunch stage. This step plays the role of exploitation for the algorithm.

4.2.1 Basic steps in Big Bang-Big Crunch algorithm

Step 1 selection of algorithm parameters: Algorithm parameters include the number of initial particles *nP*, the maximum number of function evaluation of *NFEs*, as a stopping criterion, the selection of coefficients *α* and *β* to determine the percentage of participation of the center of mass and the best particle and calculation steps based on the relation proposed by Camp.

Step 2 Formation of the initial population: The initial population of particles is formed by considering the upper and lower bounds of the decision variables. Its formation method will be according to Eq. (17):

$$\begin{aligned} &P_j^0(i) = P_{j,LB} + rand(1, nv) \otimes \left(P_{j,UB} - P_{j,LB}\right) \\ &i = 1, 2, \ldots, nP, \quad j = 1, 2, \ldots, nV \end{aligned} \tag{17}$$

Step 3 Sorting the answers and choosing the best particle: By evaluating the objective function, the results are sorted and the lowest one is introduced as the best particle best*P*.

Step 4 By updating the number of iterations, the center of mass of the particles is determined based on the phase of the big bang. The position of the center of mass of the particles is according to Eq. (18):

$$CM(i)=\frac{\sum_{j=1}^{nP}\left(\frac{P(j,i)}{PFit(j)}\right)}{\sum_{j=1}^{nP}\left(\frac{1}{PFit(j)}\right)},\quad i=1,\ldots,nV \tag{18}$$

Step 5 Determining the new position of each particle: The new position is determined based on the modified Kemp's formula and according to the big crunch phase. Its components include the center of mass and the best weight, each of which contributes a certain weight. The details of determining the new position are given in Eq. (19):

$$\begin{aligned}&\text{new}P(i)=\left(\beta\times CM+(1-\beta)\times\text{best}P\right)\\&+\frac{\alpha\times rand\otimes\left(P_{UB}-P_{LB}\right)}{nIT},\\&i=1,\ldots,nP.\end{aligned} \tag{19}$$

Step 6 the new position of the particles is evaluated and then sorted. By determining the best particle in this iteration, the position of the best particle and the number of iterations are updated.

Step 7 the termination conditions are checked and if necessary, the steps from the fourth step are repeated. Otherwise, the operation is terminated.

4.2.2 Chaos-embedded Big Bang-Big Crunch (CBB-BC) algorithm

This algorithm consists of two important strategies, big bang and big crunch, which play the role of exploration and exploitation in the algorithm. By replacing the chaos maps instead of the random selections of these two steps, there will be a significant improvement in the optimization results. This replacement is done with the following suggested scenarios:

- *Scenario 1* Embedding the chaos map in the big bang stage: in this case, the first chaos map *CHM*1 replaces the random selection of the algorithm. By embedding this map in Eq. (17), the results will be according to Eq. (20):

$$\begin{aligned}&P_j^0(i)=P_{j,LB}+CHM1\otimes\left(P_{j,UB}-P_{j,LB}\right),\\&i=1,2,\ldots,nP,\quad j=1,2,\ldots,nV.\end{aligned} \tag{20}$$

- *Scenario 2* Embedding the chaos function in the big crunch stage: in this case, the second chaos map *CHM*2 replaces the random selection of the algorithm. By embedding this map in Eq. (19), the results will be according to Eq. (21):

$$\begin{aligned}&\text{new}P(i)=\left(\beta\times CM+(1-\beta)\times\text{best}P\right)\\&+\frac{\alpha\times CHM2\otimes\left(P_{UB}-P_{LB}\right)}{nIT},\\&i=1,\ldots,nP.\end{aligned} \tag{21}$$

- *Scenario 3* Embedding the chaos maps in both stages simultaneously: in this case, both chaos maps simultaneously replace the random choices of the algorithm in Eqs. (29) and (30).

4.3 Standard Tug of War Optimization (TWO)

Inspired by the game of tug-of-war between competing teams in a league, an emerging meta-heuristic algorithm has been presented by Kaveh and Zolghadr [10]. In each stage of the game of tug-of-war, two competing teams are pulling the rope, the light team loses the competition and moves to the heavy team. In the competition between teams, the best team has the most weight and the worst team has the least weight. This algorithm is population-based and like other population-based algorithms, the initial answers are randomly selected. These answers are selected as starting teams from the bound of the decision space, and each solution is considered as a team. The total number of competing teams in each period is introduced as the league of the same period. In each iteration of the algorithm, the teams of each level are evaluated based on the merit function. In the following, the basic steps of the algorithm are presented in standard mode.

4.3.1 Basic steps in Tug of War Optimization

Step 1 Selecting the initial components of the algorithm: The initial components include two choices as follows:

- Introducing the number of teams attending in the competition. This component is displayed with *nT* icon.
- Introducing the number of members assigned to each team with *T* icon. This component is known as the league model series.

In each step, the objective function and penalized objective function values are evaluated simultaneously.

Step 2 Estimating each team's score: Each team from the league participating in the tug-of-war competition has a certain weight, which we can estimate based on the Eq. (22):

$$W_i=\frac{PFit_i-\min(PFit)}{\min(PFit)-\max(PFit)}+1\,. \tag{22}$$

The components used in this regard include the maximum and minimum value of the penalized objective function along with the value of the penalized objective function itself. The range for this relationship is between 1 and 2, where the numerical value of 2 belongs to the best and heaviest team.

Step 3 Competing between teams: Each team in the league competes with all other teams. To move to its new position in each period of repetition, the tensile force applied by each team is proportional to the frictional force at rest. In the game of tug-of-war, there is always a competition between two teams who continue to pull the rope on both sides, and here it is meant by two values that result from the scores of the two teams. In the modeling, the value of the coefficient of friction is assumed to be one, and the pulling force between the two teams *i* and *j* can be the maximum of the following two values, respectively:

$$F_{p,ij} = \max\left\{W_i \mu_s, W_j \mu_s\right\} . \tag{23}$$

Therefore, the strength of team *i* in the face of heavier team *j* can be as follows:

$$F_{r,ij} = F_{p,ij} - W_i \mu_k . \tag{24}$$

In order to determine the acceleration for the movement of team *i* towards team *j*, the following relationship is proposed:

$$a_{ij} = \frac{F_{r,ij}}{W_i \mu_k} g_{ij}, \quad g_{ij} = T_j - T_i \tag{25}$$

In the presented relationship, the acceleration of gravity plays an important role, which one can access by determining the difference for the position of the two teams. To determine the amount of displacement in each step, the following relationship is used:

$$\text{stepsize}_{ij} = \frac{1}{2} a_{ij} \Delta T^2 + \alpha\beta(Lb - Ub) \otimes randn \cdot \tag{26}$$

In the second part of the relation of step-size, the random components of the league teams have been applied. In each period of the game, team *i* will pass a part of the decision space before being stopped by team *j*. In order to make this interval, one can use the coefficient of possible *α* effects. This coefficient is in the range of [0.9, 0.99] and large values for this coefficient at the beginning stage create the opportunity to search the decision space. In the final stages, by reducing the numerical value of this coefficient and choosing smaller steps, the convergence towards the optimum increases. Also, *β* is chosen as the scaling factor and its changes are in the range of [0, 1]. The scaling factor is to control the steps of post-suggestion responses. In cases where we the search step is needed more accurately; this parameter is introduced with smaller steps. To select the allowed range to determine the step length, we use the difference between the upper bound and the lower bound of the variables in the search space. In this research, a standard normal random distribution has been proposed, which can create diversity in the search space by multiplying member by member. If in some cases *j* is lighter than *i*, the displacement is not done and its value is assumed to be zero. In the original version, time steps have been applied as 1. At the end of each stage of the periodic game, the total number of places changed for Team *i* is determined according to the following relationship:

$$\text{stepsize}_i = \sum_{j=1}^{nT} \text{stepsize}_{ij}, \quad i \neq j . \tag{27}$$

To determine the new position for a team, the following step-size length is added to its previous position.

$$T_i^{\text{new}} = T_i + \text{stepsize}_i \tag{28}$$

Step 4 Replacing the improved results: With the competition between the league teams, the new results are compared with the existing results, and if the results are improved, the position of the league teams is updated.

Step 5 Controlling the range of the decision space: by applying step-size to the initial position, in some cases the variables go out of the determined ranges. Using the following relationship, these intervals are modified.

$$T_{ij} = \text{best}T_j + \left(\frac{randn}{NIT_s}\right)\left(\text{best}T_j - T_{ij}\right) \tag{29}$$

In the relation proposed by best*T*, the result is the best team so far and the NIT_s counter is related to the repetition.

Step 6 The conditions for the final stage of the games will be checked and if necessary, the competition between the teams will be repeated again.

4.3.2 Chaos-embedded Tug of War Optimization (CTWO)

Like meta-heuristic algorithms, two important strategies of exploration and exploitation are considered in this algorithm. These two strategies are in *β* scaling factor and applying restrictions in the space of decision variables. By embedding chaos functions in the random selection parts of the algorithm in the standard mode, a significant improvement will be achieved in the results of the algorithm. The three suggested scenarios for this installation are as follows:

- *Scenario 1* Embedding the chaos function in the exploration strategy: In this step, the first chaos function *CHM*1 replaces the β scaling factor in Eq. (26). Equation (30) for the chaotic state is as follows:

$$\begin{aligned}\text{stepsize}_{ij} &= \frac{1}{2} a_{ij} \Delta T^2 \\ &+ \alpha \cdot CHM1 \otimes (Lb - Ub) \otimes randn.\end{aligned} \tag{30}$$

- *Scenario 2* Embedding the chaos function in the exploitation strategy: In this step, the second chaos function *CHM*2 replaces the coefficient of the decision space limitation in Eq. (29). Equation (31) for the chaotic state is as follows:

$$T_{ij} = \text{best}T_j + \left(\frac{CHM2}{NIT_s}\right)\left(\text{best}T_j - T_{ij}\right). \tag{31}$$

- *Scenario 3* Embedding chaos functions simultaneously in the stages of exploration and exploitation: In this step, both chaos functions are simultaneously replaced in Eqs. (26), (29).

Chaotic algorithms can be developed for Topologies optimization in the case of probabilistic loading [28, 29], Reliability based topology optimization of thermo-elastic structures [30], and elasto-plastic limit analysis [31, 32]. In these cases, the design with the limit of the minimum penalized weight is practical.

5 Numerical examples of optimal truss design

Each of the introduced meta-heuristic algorithms deal with optimization with different physical inspirations. Embedding chaos functions in these algorithms and forming triple scenarios can make a significant improvement in the optimization of truss structures. Each chaos function can solve the weakness related to exploration, exploitation or both simultaneously in the decision space. Therefore, at least two types of chaos functions should be considered for embedding in algorithms. Considering the formation of chaotic mutations in meta-heuristic algorithms, in any case, a significant improvement in the results is obtained. In order to expand the scope of investigations, 3 standard modes of meta-heuristic algorithms are compared with 18 chaotic modes. This wide diversity provides a challenge and intense competition to move towards overall optimality. Therefore, the possibility of access to optimal answers with high accuracy increases. By comparing these models, the best algorithm, the best chaotic function and the best scenario are selected. Although in the optimal design of truss structures, the main goal is to choose the lowest possible value for the cross-sectional area of the members, but at the same time, the limits related to the permissible stress, the permissible deformation of the nodes and the slenderness of the members must be satisfied according to the regulations. In the following well-known examples are examined.

5.1 A 47-bar power transmission tower

Power transmission tower with truss system is selected according to Fig. 3. Its geometric structure consists of 47 members and 22 nodes. The numbering of the nodes is presented in Fig. 3. In this structure, the specific weight of structural materials is 0.3 lb/in^3 and the modulus of elasticity of the members is 30,000 ksi. Both stress and buckling limits must be satisfied for all members. The allowable

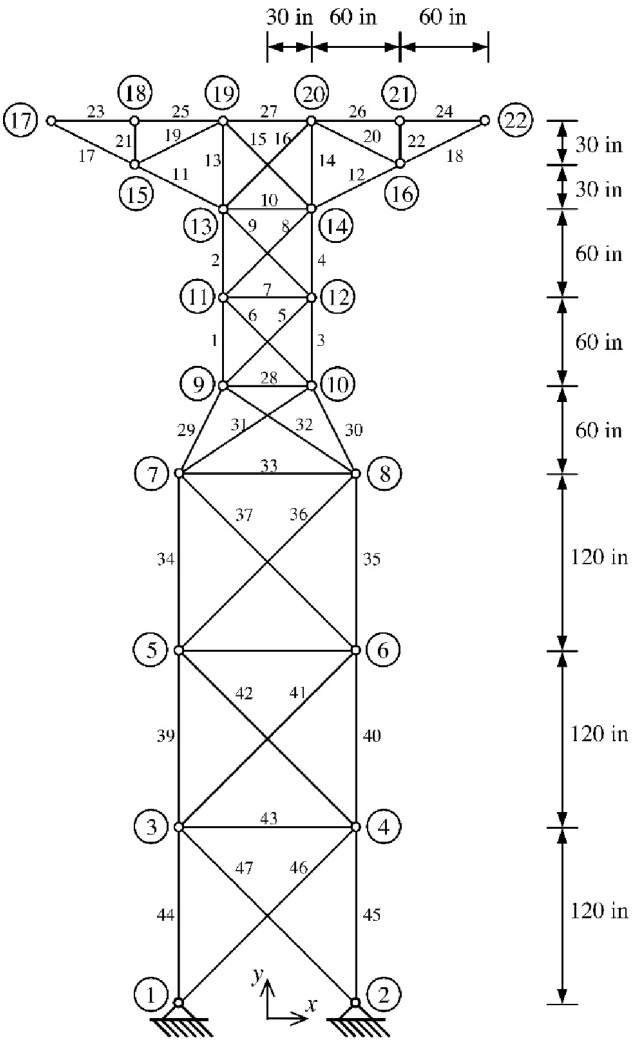

Fig. 3 Schematic of a 47-bar power transmission tower

stress is set at 20 ksi in tension and 15 ksi in compression. The allowable compressive stress for Eulerian buckling is according to the following Eq. (32):

$$\sigma_i^e = \frac{-KEA_i}{L_i^2}; \quad i = 1, 2, \ldots, 47 . \tag{32}$$

In this relationship, K includes a constant coefficient that is selected according to the geometrical shape of the cross-section profile. Also, E is the modulus of elasticity, A_i is the cross-sectional area value, and L_i is the length of the member. In the recent research, the value of K is 3.96. The number of loading combinations has been selected in three groups. The loads of the first group are entered with an intensity of 6 kP along the positive X-axis with 14 kP along the negative Y-axis and at nodes 17 and 22. The loads of the second group with an intensity of 6 kP along the positive X-axis with 14 kP along the negative axis of Y are entered only at node 17.

The loads of the third group with an intensity of 6 kP along the positive X-axis with 14 kP along the negative axis of Y are entered only at node 22. In the first group, diagonal loads related to both power transmission lines are applied in the normal state, but in the second and third groups, it considers the cases where one of the two lines is broken. The structure has geometric symmetry and the members of the structure are classified into 27 groups. AISC codes are used to design the cross-section of the members by 64 discrete values. By performing 20 independent evaluations, statistical samples were modeled and the results are displayed in statistical Table 1. In these models, the results of the best weight, the best average and the coefficient of variation are presented. Comparison of optimization details in standard and chaotic mode is done in Table 2. In Table 2, the most optimal of the three scenarios for the chaotic state is presented.

In order to summarize the results and quick access, bar graphs are shown in Fig. 4. Embedding chaos functions in algorithms has caused a significant improvement in reducing the weight of the structure. In the following, we introduce each of the chaos functions and similar scenarios with the best optimization results:

- For optimization with Chaotic Thermal Exchange Optimization (CTEO), the Logistic chaos map with the first scenario has the most optimal weight with the value of 2328.8298 pounds.
- For optimization with Chaotic Big Bang-Big Crunch (CBB-BC), the Gaussian chaos map with the second scenario has the most optimal weight with the value of 2313.9752 pounds.
- For optimization with Chaotic Tug-of-War Optimization (CTWO), the Gaussian chaos map with the third scenario has the most optimal weight with the value of 2314.5396 pounds.

The results of these investigations are presented in Table 2. By comparing all the Chaotic Algorithms and Scenarios, the meta-heuristic Chaotic Big Bang-Big Crunch (CBB-BC), with Gaussian chaos map and the second scenario and the optimal weight of ***2313.970*** pounds has the most optimal result. For convenience and quick access, the graph of the convergence history of the algorithms for the standard and chaotic mode is presented in Fig. 5.

Table 1 Statistical results of the 47-bar power transmission tower

Algorithms	Statistical Information	TEO Standard	CTEO-21 Logist-1	CTEO-22 Logist-2	CTEO-23 Logist-3	CTEO-31 Gauss-1	CTEO-32 Gauss-2	CTEO-33 Gauss-3
1-TEO	Best	2395.3630	***2328.8298***	2391.1050	2386.9676	2355.9526	2357.1052	2349.0212
	Mean	2423.4305	***2364.7342***	2423.2361	2412.5609	2415.0340	2372.2855	2384.5812
	C.V(%)	1.2528	1.8153	1.2928	0.7157	1.9300	***0.60191***	1.7411
Algorithms	Statistical Information	BB-BC Standard	CBB-BC-21 Logist-1	CBB-BC-22 Logist-2	CBB-BC-23 Logist-3	CBB-BC-31 Gauss-1	CBB-BC-32 Gauss-2	CBB-BC-33 Gauss-3
2-BB-BC	Best	2363.1164	2321.7473	2326.8120	2342.8048	2318.2260	***2313.9752***	2335.3663
	Mean	2379.8142	2382.4864	2381.8078	2400.5620	2336.0767	***2332.9950***	2389.4763
	C.V(%)	0.9005	2.7106	1.7578	1.9148	0.80995	***0.77935***	2.0574
Algorithms	Statistical Information	TWO Standard	CTWO-21 Logist-1	CTWO-22 Logist-2	CTEWO-23 Logist-3	CTWO-31 Gauss-1	CTWO-32 Gauss-2	CTWO-33 Gauss-3
3-TWO	Best	2392.1796	2317.0418	2319.2046	2330.1925	2316.3392	2318.1133	***2314.5396***
	Mean	2400.7506	2349.3671	2361.6789	2339.9130	2337.0366	***2336.8646***	2346.0129
	C.V(%)	0.29636	0.94111	2.0276	0.45009	0.99221	***0.73604***	1.1082

Table 2 Optimal design comparison for the 47-bar power transmission tower

Number group	Element group	TEO Stand	CTEO Logis-1	BB-BC Stand	CBB-BC Gaus-2	TWO Stand	CTWO Gaus-3
1	A_1-A_3	3.763	3.7605	3.799	3.7785	3.8081	3.7728
2	A_2-A_4	3.3323	3.3221	3.3785	3.3634	3.3625	3.3635
3	A_5-A_6	0.84608	0.78404	0.77638	0.76238	0.75403	0.76149
4	A_7	1.2523	0.64742	0.11672	0.10145	0.24984	0.16657
5	A_8-A_9	0.8929	0.84296	0.79343	0.78248	0.83455	0.77622
6	A_{10}	1.9818	1.948	1.9409	1.8336	2.1999	2.9112
7	A_{11}-A_{12}	2.113	2.0899	2.1105	2.0922	2.1053	2.1228
8	A_{13}-A_{14}	1.2655	1.1656	1.237	1.1705	1.2617	1.2034
9	A_{15}-A_{16}	1.589	1.5625	1.5322	1.5526	1.5751	1.52
10	A_{17}-A_{18}	2.2775	2.107	2.0994	2.0886	2.1159	2.0882
11	A_{19}-A_{20}	0.11446	0.10234	0.17467	0.10149	0.11779	0.10034
12	A_{21}-A_{22}	0.34357	1.0034	0.10098	0.1	1.4884	0.59216
13	A_{23}-A_{24}	1.6983	1.7258	1.7229	1.7101	1.7406	1.7097
14	A_{25}-A_{26}	1.7289	1.7076	2.3211	1.7111	1.7179	1.7059
15	A_{27}	1.7633	1.8404	1.3938	1.5844	1.7746	1.6496
16	A_{28}	1.5352	1.3617	0.53918	0.45629	2.0948	0.96308
17	A_{29}-A_{30}	3.6587	3.6588	3.6339	3.5906	3.6682	3.6285
18	A_{31}-A_{32}	1.4112	1.3952	1.461	1.4423	1.3908	1.4211
19	A_{33}	0.21953	0.18927	0.2541	0.27757	0.18351	0.27472
20	A_{34}-A_{35}	3.0104	3.0265	3.0276	3.0249	3.0302	2.9931
21	A_{36}-A_{37}	1.2653	1.2382	1.2272	1.2437	1.2333	1.2779
22	A_{38}	0.28865	0.30927	0.43333	0.30288	0.27483	0.25192
23	A_{39}-A_{40}	3.6707	3.4876	3.693	3.674	3.5788	3.4562
24	A_{41}-A_{42}	1.5246	1.5265	1.5193	1.5277	1.5552	1.5186
25	A_{43}	0.10032	0.10288	0.1	0.1054	0.27612	0.10041
26	A_{44}-A_{45}	4.504	4.0988	4.5456	4.3849	4.3196	4.0756
27	A_{46}-A_{47}	1.4369	1.4275	1.4367	1.4251	1.5	1.4306
Best	Weight (lb)	2395.36	2328.82	2363.11	***2313.97***	2392.17	2314.53
Mean	Weight (lb)	2423.43	2364.73	2379.81	2332.99	2400.75	2346.01
Coefficient	Var (CV)	1.2528	1.8153	0.9005	0.77935	0.29636	1.1082
NFE		12000	12000	12000	12000	12000	12000

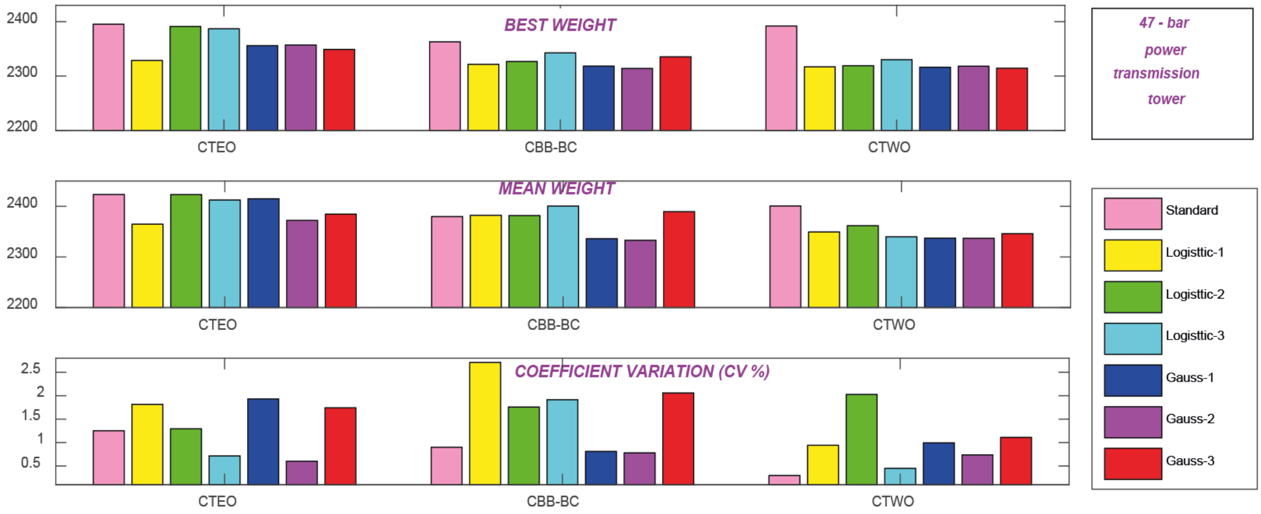

Fig. 4 Optimization results in standard mode and selection of chaos map for the 47-bar power transmission tower

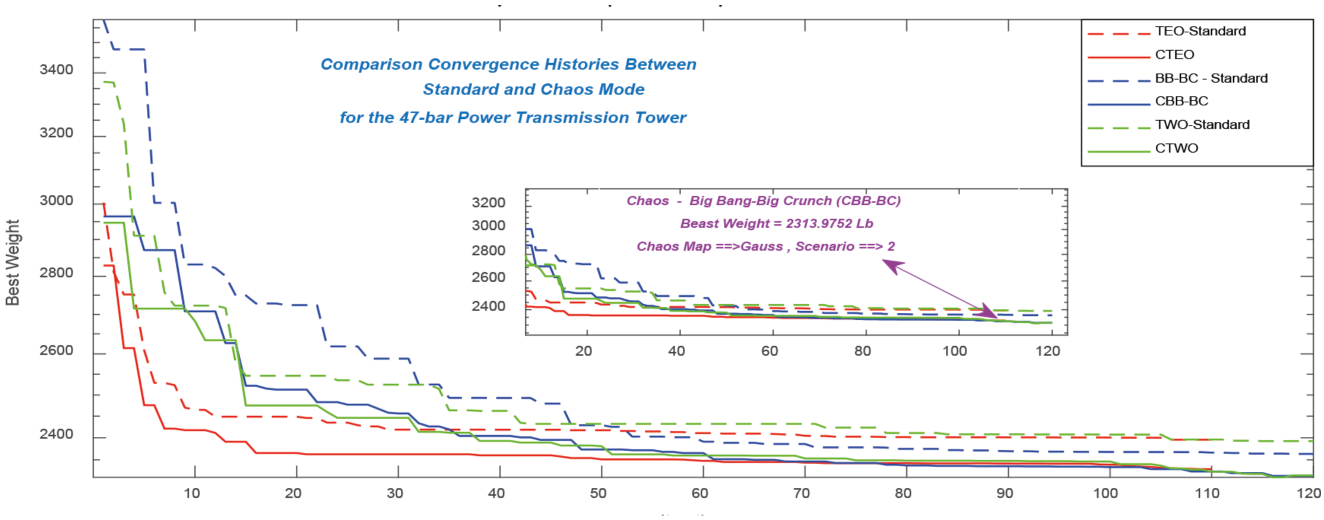

Fig. 5 The convergence histories for the 47-bar power transmission tower

5.2 A 120-bar dome shaped truss

The dome-shaped truss with 120 members is selected according to Fig. 6. According to the geometrical symmetry of the structure, the members are classified into seven groups. To determine the allowable tensile and compressive stresses, it is done according to the AISC ASD code. All nodes of the dome, except the support nodes, are loaded by gravity. The intensity of the incoming load in node 1 is −13.49 kips, in nodes 2 to 14 it is −6.744 kips and in other nodes −2.248 kips. In order to control the deformation of nodes, a limit of 0.1969 in is suggested for all extensions. The lower bound and upper bound for the design of the cross-sectional area of the members are 0.775 in^2 and 20 in^2, respectively. Equation (33) is proposed to calculate the allowable stress in tension and compression.

$$\sigma_i = \begin{cases} 0.6F_y & \text{for} \quad \sigma_i \geq 0 \\ \sigma_i^- & \text{for} \quad \sigma_i < 0 \end{cases} \tag{33}$$

For compressive stresses, the following Eq. (34) can be used

$$\sigma_i^- = \begin{cases} \left(1 - \dfrac{\lambda^2}{2C^2}\right) F_y \Big/ FS & \text{for} \quad \lambda < C \\ \dfrac{12\pi^2 E}{23\lambda^2} & \text{for} \quad \lambda \geq C \end{cases} . \tag{34}$$

In this equation we have the Eq. (35):

$$FS = \left(\frac{5}{3} + \frac{3\lambda}{8C} - \frac{\lambda^3}{8C^3}\right); \; C = \sqrt{\left(\frac{2\pi^2 E}{F_y}\right)}; \; \lambda = \frac{kl}{r}; \tag{35}$$

$$r = aA^b; \; a = 0.4993; \; b = 0.6777.$$

In this relation, E expressing the modulus of elasticity, F_y is the yield stress of the steel, C is the amount of slenderness, which separates to the elastic or inelastic buckling region, compared to the existing slandering λ, and also k is the effective length coefficient, l and r is the radius of rotation of the limb. The minimum and maximum cross-sectional area of all members is 0.775 in^2 and 20 in^2 respectively. By performing 20 independent evaluations, statistical samples were modeled and the results are displayed in statistical Table 3. In these models, the results of the best weight, the best average and the coefficient of variation are presented. Comparison of optimization details in standard and chaotic mode is done in Table 4. In Table 4, the most optimal of the three scenarios for the chaotic state is presented. In order to

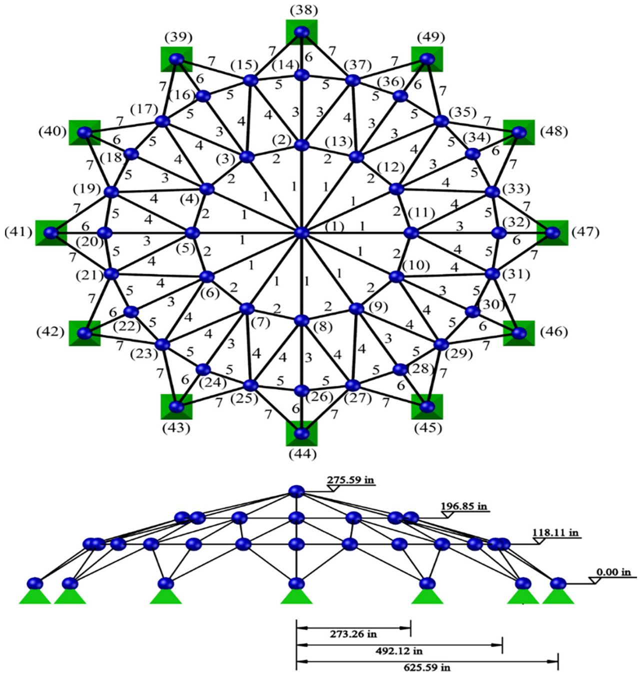

Fig. 6 Schematic of a 120-bar dome shaped truss

Table 3 Statistical results for the 120-bar dome shaped truss

Algorithms	Statistical Information	TEO Standard	CTEO-21 Logist-1	CTEO-22 Logist-2	CTEO-23 Logist-3	CTEO-31 Gauss-1	CTEO-32 Gauss-2	CTEO-33 Gauss-3
	Best	33256.2451	33214.9629	33212.9148	33126.4708	33137.2136	33098.8560	***33086.7020***
1-TEO	Mean	33264.5395	33246.0068	33240.8490	33149.9988	33189.3005	33147.8790	***33136.4610***
	C.V(%)	***0.016597***	0.088786	0.06405	0.068182	0.092985	0.09734	0.08869
Algorithms	Statistical Information	BB-BC Standard	CBB-BC-21 Logist-1	CBB-BC-22 Logist-2	CBB-BC-23 Logist-3	CBB-BC-31 Gauss-1	CBB-BC-32 Gauss-2	CBB-BC-33 Gauss-3
	Best	33241.8290	33083.2930	***33057.6510***	33128.9340	33177.9380	33152.6095	33133.9880
2-BB-BC	Mean	33252.237	33121.750	***33081.0974***	33161.587	33211.0454	33195.5537	33164.132
	C.V(%)	0.036207	0.12069	0.06264	0.09124	***0.056916***	0.095251	0.079824
Algorithms	Statistical Information	TWO Standard	CTWO-21 Logist-1	CTWO-22 Logist-2	CTEWO-23 Logist-3	CTWO-31 Gauss-1	CTWO-32 Gauss-2	CTWO-33 Gauss-3
	Best	33235.303	33159.561	33144.504	33114.604	33093.011	33128.753	***33065.343***
3-TWO	Mean	33261.579	33178.461	33180.142	33150.603	***33122.219***	33186.700	33125.499
	C.V(%)	0.074489	***0.047542***	0.07333	0.10716	0.05806	0.10769	0.12704

Table 4 Optimal design comparison for the 120-bar dome shaped truss

Number group	TEO Stand	CTEO Gaus-3	BB-BC Stand	CBB-BC Logis -2	TWO Stand	CTWO Gaus-3
1	3.02544	3.0246	3.02277	3.02699	3.02596	3.02377
2	14.6042	14.5465	14.9875	14.9501	14.7717	15.0837
3	5.10704	5.21025	4.89262	5.22663	5.07626	5.10404
4	3.13438	2.94134	3.11807	2.87259	3.10532	2.90719
5	8.47955	8.61269	8.52708	8.50858	8.39326	8.48781
6	3.40848	3.39464	3.3282	3.32841	3.48949	3.34531
7	2.49458	2.49612	2.50013	2.50298	2.49375	2.4979
Best Weight (lb)	33256.245	33086.702	33241.829	***33057.651***	33235.303	33065.343
Mean Weight (lb)	33264.539	33136.461	33252.237	33081.097	33261.579	33125.499
Coefficient Var (CV)	0.016597	0.08869	0.036207	0.06264	0.074489	0.12704
NFE	12000	12000	12000	12000	12000	12000

summarize the results and quick access, bar graphs are shown in Fig. 7. Embedding chaos functions in algorithms has caused a significant improvement in reducing the weight of the structure. In the following, each of the chaos functions is introduced and similar scenarios with the best optimization results:

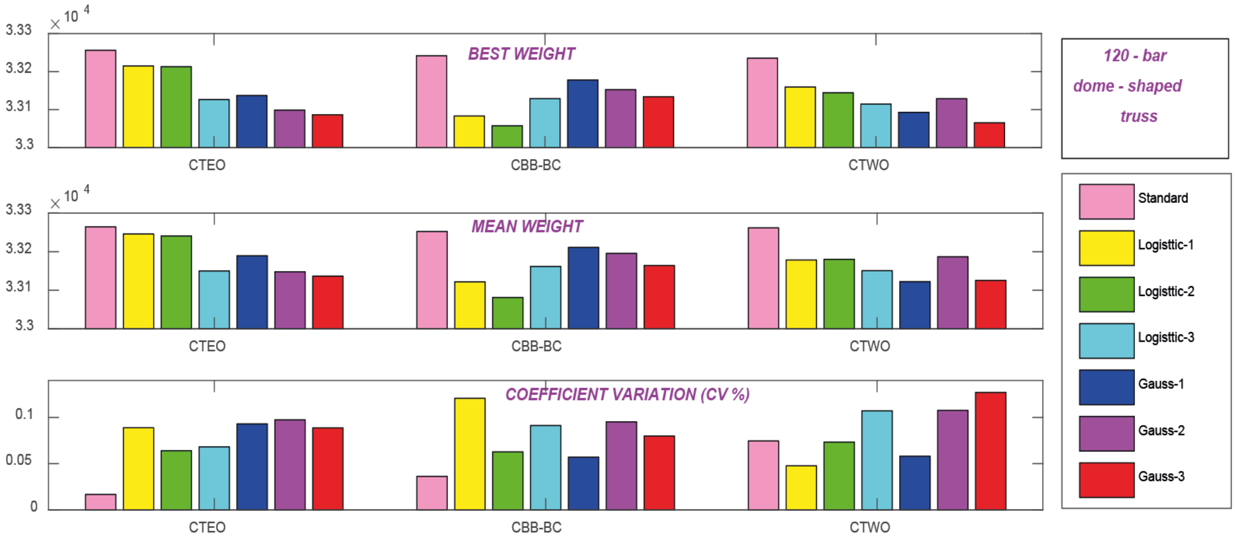

Fig. 7 Optimization results in standard mode and chaos map for the 120-bar dome shaped truss

- For optimization with Chaotic Thermal Exchange Optimization (CTEO), the Gaussian chaos map with the third scenario has the most optimal weight with the value of 33086.7020 pounds.
- For optimization with Chaotic Big Bang-Big Crunch (CBB-BC), the Logistic chaos map with the second scenario has the most optimal weight with the value of 33057.6510 pounds.
- For optimization with Chaotic Tug-of-War Optimization (CTWO), the Gaussian chaos map with the third scenario has the most optimal weight with the value of 33065.3430 pounds.

The results of these investigations are presented in Table 4. By comparing all the Chaotic Algorithms and Scenarios, the meta-heuristic Chaotic Big Bang-Big Crunch (CBB-BC), with Logistic chaos map and the second scenario and the optimal weight of ***33057.6510*** pounds has the most optimal result. For convenience and quick access, the graph of the convergence history of the algorithms for the standard and chaotic mode is presented in Fig. 8.

6 Discussion of the algorithms

Meta-heuristic algorithms inspired by physical laws are more successful compared to other algorithms. The relationships governing physical phenomena are easily modeled and used in meta-heuristic algorithms. To derive the final results regarding the success of chaotic algorithms, the processes related to Tables 1–4 should be first combined and then normalized [22, 23]. In order to derive final results about the success of chaos algorithms, the processes from Tables 1–4 are combined and then normalized. Equation (36) is intended to combine and summarize information about the contribution of all problems.

$$Val^{MV}_{com} = \frac{1}{S}\sum_{i=1}^{S}\left(\frac{Val^{MV}}{Val_{i,\min}} - 1\right) \tag{36}$$

In this regard, the standard mode has been compared with six chaotic modes. In order to increase the range of reviews and access to more accuracy, the percentage of success has been done with the participation of all examples. The components used include the optimal values of Tables 1–4 by Val^{MV}, the combined values of the results by Val^{MV}_{com}, the number of examined samples by S, and also the lowest numerical value in the evaluation of each component by $Val_{i,\min}$ applied [24]. For ease of interpretation of the results in Eq. (37), we have benefited from inverted and normalized values. Therefore, optimal modes belong to chaos functions and scenarios that have achieved a high percentage of success.

$$Val^{MV}_{norm} = \frac{\dfrac{1}{Val^{MV}_{com}}}{\displaystyle\sum_{j=1}^{nopt}\frac{1}{Val^{MV}_{j,com}}} \times 100 \tag{37}$$

In Table 5, the percentage of success of each of the algorithms in the standard mode and six chaotic modes has been analyzed. By examining Table 5, we can introduce the best chaos function and the best scenario for each meta-heuristic algorithm. Also, by comparing the percentage of success of chaotic modes with the standard initial mode, determine the amount of improvement in

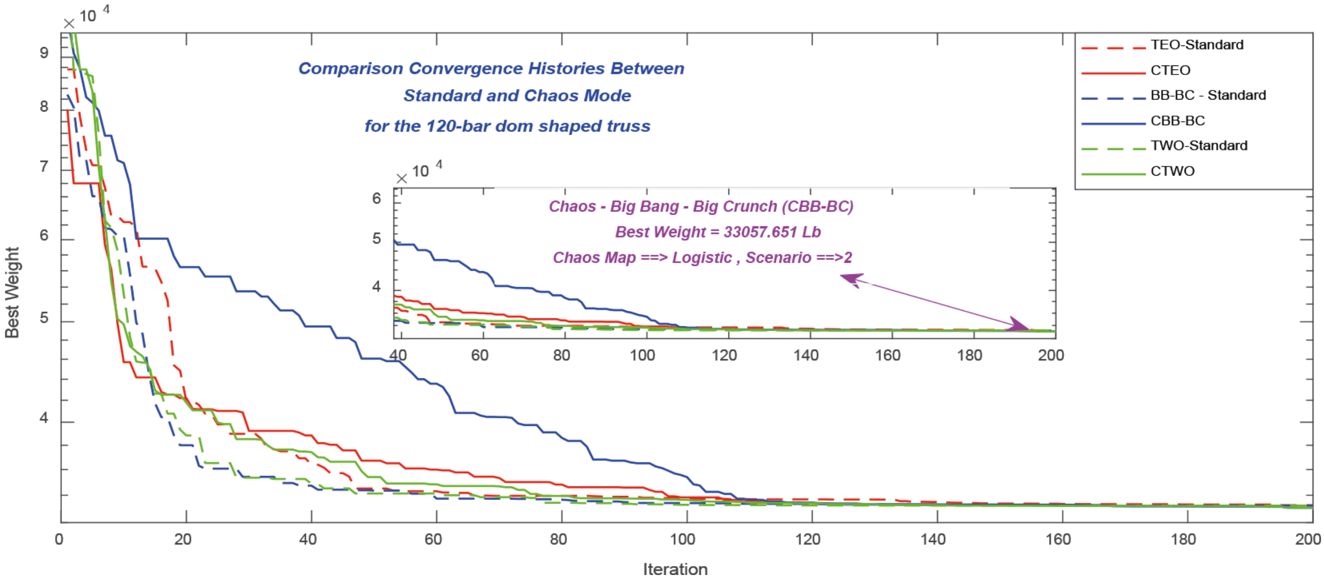

Fig. 8 The convergence histories for the 120-bar spatial dome

Table 5 Final normalized value with the participation of all examples

Category	Algorithms	Standard	Logistic 21	Logistic 22	Logistic 23	Gauss 31	Gauss 32	Gauss 33
Best Weight	TEO	5.429	18.0592	6.2804	6.8034	12.7355	18.7772	***31.9154***
	BB-BC	3.4588	9.5358	8.601	5.5069	10.5997	***50.5523***	11.7455
	TWO	1.5213	6.6924	4.1453	3.5091	12.7274	15.5581	***55.8462***
Mean Weight	TEO	3.5408	7.5698	5.1582	6.7163	3.5288	***57.5362***	15.95
	BB-BC	4.689	3.9614	5.2293	4.0269	16.4002	***58.7054***	6.9877
	TWO	2.1453	10.3675	2.7191	3.6907	23.7779	***49.9395***	7.36
(CV %)	TEO	9.6752	7.206	14.9836	***32.7795***	6.2171	18.2795	10.8592
	BB-BC	25.2467	1.7971	4.0327	3.9469	***50.6501***	10.3931	3.9334
	TWO	9.5916	***59.9434***	2.1418	3.042	11.266	12.0488	1.9663

the optimization results. Next, in order to quickly access the process of improving the results in chaotic situations, a circular diagram of the Table 5's components is formed. These components include the best weight, the best average and the best coefficient of variation.

6.1 Results of optimal design for best weight

According to the combination of the results with the participation of all the examples, the optimal design for determining the best weight in the Chaotic Thermal Exchange Optimization (CTEO), belonging to the Gaussian chaos map with the third scenario has been successful with a value of 31.915%, the algorithm based on Chaotic Big Bang-Big Crunch (CBB-BC), belonging to the Gaussian chaos function with the second scenario has been successful with a value of 50.552%, and finally, the Chaotic Tug-of-War Optimization (CTWO), belongs to the Gaussian chaos function with the third scenario has been successful with a value of 55.846%. The final results of the optimal design for introducing the best weight are displayed in Fig. 9.

6.2 Results of optimal design for best mean

According to the combination of the results with the participation of all the examples, the optimal design for determining the best mean in the Chaotic Thermal Exchange Optimization (CTEO), Chaotic Big Bang-Big Crunch (CBB-BC) and Chaotic Tug-of-War Optimization (CTWO), for all three belongs to the Gaussian chaos function with the second scenario and has been successful with a value of 57.536%, 58.705%, and 49.939% respectively. The final results of the optimal design to introduce the best average is displayed in Fig. 10.

6.3 Results of optimal design for best coefficient of variation

According to the combination of the results with the participation of all the examples, the optimal design for determining the best coefficient of variation in the Chaotic Thermal Exchange Optimization (CTEO), belonging to the Logistic chaos map with the third scenario has been successful with a value of 32.779%, the algorithm based on Chaotic Big Bang-Big Crunch (CBB-BC), belonging

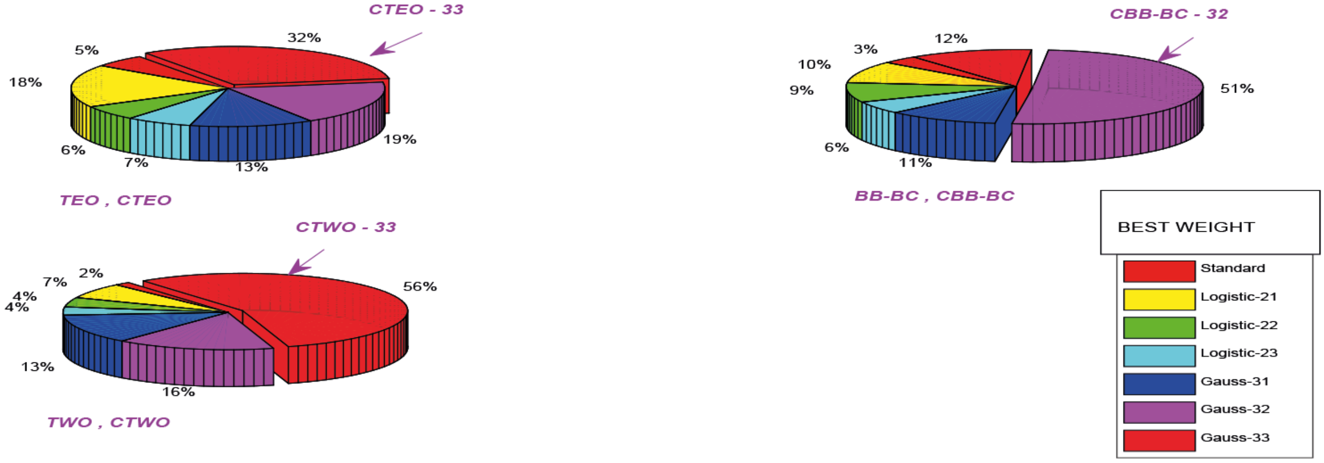

Fig. 9 The final results of the optimal design to determine the best weight

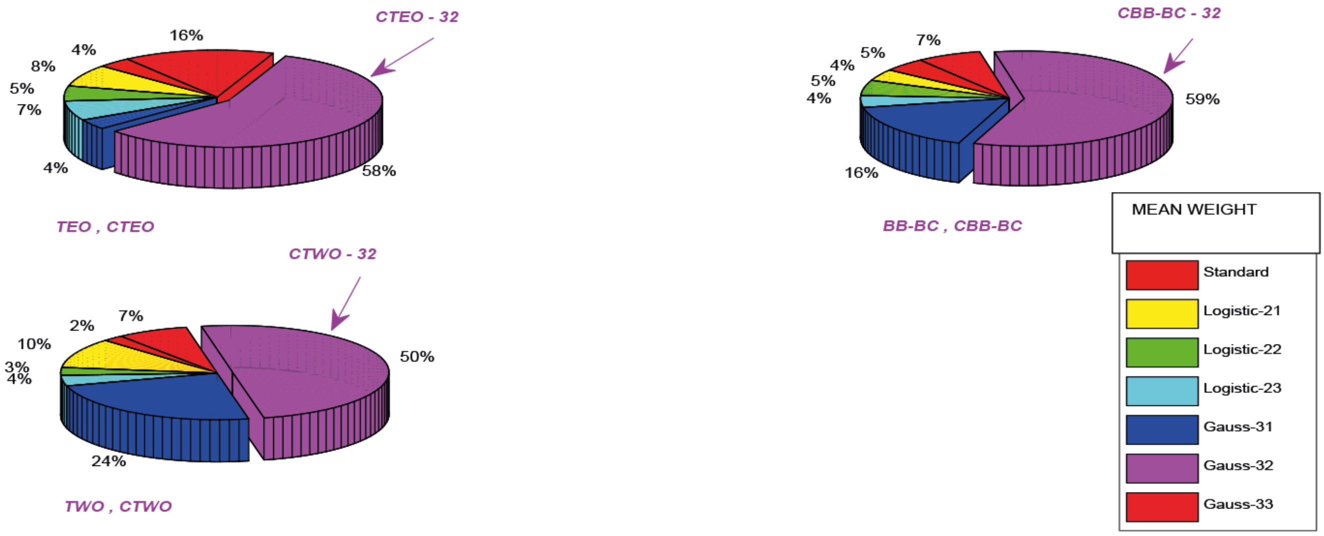

Fig. 10 The final results of the optimal design to determine the best mean

to the Gaussian chaos function with the first scenario has been successful with a value of 50.650%, and finally, the Chaotic Tug-of-War Optimization (CTWO), belongs to the Logistic chaos function with the second scenario has been successful with a value of 55.846%. The final results of the optimal design to introduce the best the coefficient of variation is shown in Fig. 11.

7 Conclusions

In this research, the chaos functions in three well-known algorithms have been inspired by the physical laws of embedding and significant results have been obtained regarding the improvement of the optimization conditions. The main results are as follows:

- Physically inspired meta-heuristic algorithms are more successful. Classical conditions for physical relations make their use as sources of inspiration in meta-heuristic algorithms easy and have better results.
- The selected algorithms include three well-known physically inspired algorithms, which will have the best optimization results for non-linear and non-convex problems, but face early convergence in problems with a large number of decision variables.
- Chaotic functions create the necessary conditions to escape from the local optima trap by making sudden jumps.
- The embedding of chaos functions is done with three scenarios, in scenarios 1 and 2, chaos functions have

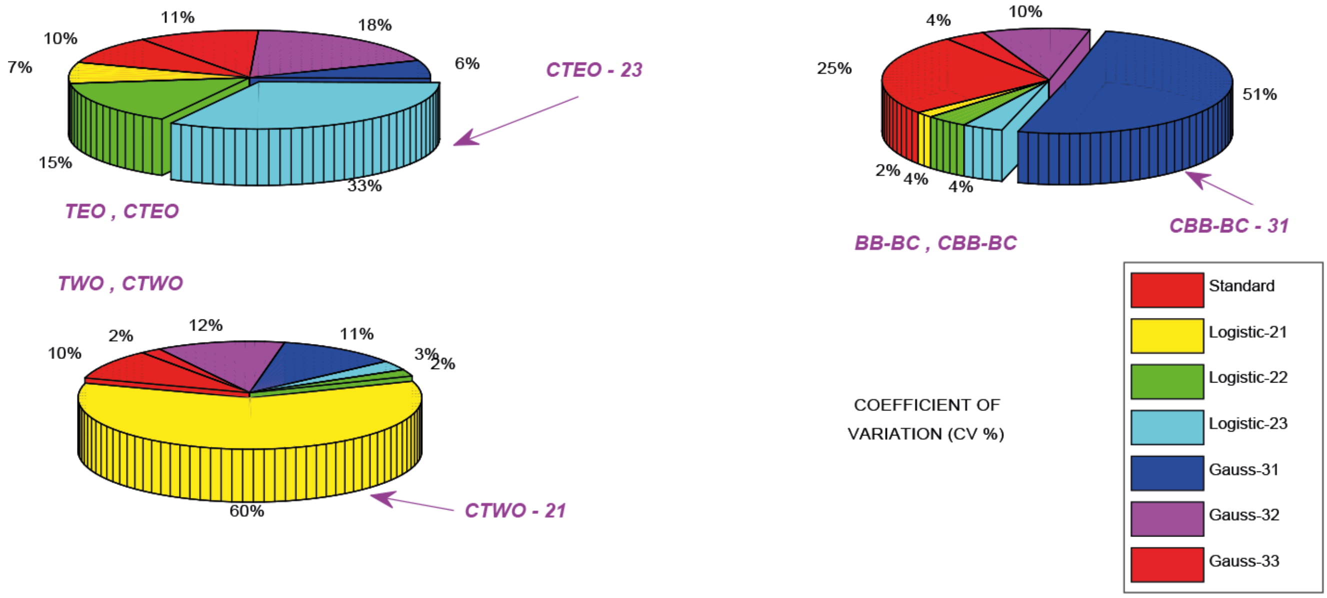

Fig. 11 The final results of the optimal design to determine the best coefficient of variation

replaced the exploration and exploitation steps, respectively. But in the third scenario, these functions simultaneously replace both stages. Therefore, based on this, the algorithms can be classified into three categories.

- In chaos functions, the first term plays a decisive role in the structure of the chaos series. Therefore, performing several initial repetitions before the main repetitions to find the best starting sentence will play a significant role in improving the results.
- Finally, it is interesting to mention that the force method of structural analysis can be employed in place of the displacement method with considerable benefits for frame structures with smaller degrees of indeterminacy than the kinematical indeterminacy [31, 32].

References

[1] Kaveh, A. "Advances in Metaheuristic Algorithms for Optimal Design of Structures", Springer, 2021. ISBN 978-3-030-59391-9 https://doi.org/10.1007/978-3-030-59392-6

[2] Holland, J. H. "Adaptation in Natural and Artificial Systems: An introductory analysis with applications to biology, control, and artificial intelligence", University of Michigan Press, 1975. ISBN 9780472084609

[3] Price, K. V., Storn, R. M., Lampinen, J. A. "Differential Evolution: A Practical Approach to Global Optimization", Springer, 2005. ISBN 978-3-540-20950-8 https://doi.org/10.1007/3-540-31306-0

[4] Kennedy, J., Eberhart, R. "Particle swarm optimization", In: Proceedings of ICNN'95 - International Conference on Neural Networks, Perth, WA, Australia, 1995, pp. 1942–1948. ISBN 0-7803-2768-3 https://doi.org/10.1109/ICNN.1995.488968

[5] Karaboga, D. "An idea based on honey bee swarm for numerical optimization", Erciyes University, Kayseri, Türkiye, Rep. TR06, 2005.

[6] Kaveh, A., Zolghadr, A. "Optimal design of cyclically symmetric trusses with frequency constraints using cyclical parthenogenesis algorithm", Advances in Structural Engineering, 21(5), pp. 739–755, 2018. https://doi.org/10.1177/1369433217732492

[7] Kaveh, A., Bakhshpoori, T. "Water Evaporation Optimization: A novel physically inspired optimization algorithm", Computers and Structures, 167, pp. 69–85, 2016. https://doi.org/10.1016/j.compstruc.2016.01.008

[8] Kaveh, A., Dadras, A. "A novel meta-heuristic optimization algorithm: Thermal exchange optimization", Advances in Engineering Software, 110, pp. 69–84, 2017. https://doi.org/10.1016/j.advengsoft.2017.03.014

[9] Erol, O. K., Eksin, I. "A new optimization method: Big Bang–Big Crunch", Advances in Engineering Software, 37(2), pp. 106–111, 2006. https://doi.org/10.1016/j.advengsoft.2005.04.005

[10] Kaveh, A., Zolghadr, A. "A novel meta-heuristic algorithm: tug of war optimization", International Journal of Optimization in Civil Engineering, 6(4), pp. 469–492, 2016.

[11] Kaveh, A., Talatahari, S. "A novel heuristic optimization method: charged system search", Acta Mechanica, 213, pp. 267–289, 2010. https://doi.org/10.1007/s00707-009-0270-4

[12] Kaveh, A., Mahdavi, V. R. "Colliding bodies optimization: A novel meta-heuristic method", Computers and Structures, 139, pp. 18–27, 2014. https://doi.org/10.1016/j.compstruc.2014.04.005

[13] Alatas, B. "Chaotic harmony search algorithms", Applied Mathematics and Computation, 216(9), pp. 2687–2699, 2010. https://doi.org/10.1016/j.amc.2010.03.114

[14] Kaveh, A., Ilchi Ghazaan, M. "A new meta-heuristic algorithm: vibrating particles system", Scientia Iranica, Transactions A: Civil Engineering, 24(2), pp. 551–566, 2017. https://doi.org/10.24200/sci.2017.2417

[15] Duan, Q. Y., Gupta, V. K., Sorooshian, S. "Shuffled complex evolution approach for effective and efficient global minimization", Journal of Optimization Theory and Applications, 76(3), pp. 501–521, 1993. https://doi.org/10.1007/BF00939380

[16] Eusuff, M., Lansey, K., Pasha, F. "Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization", Engineering Optimization, 38(2), pp. 129–154, 2006. https://doi.org/10.1080/03052150500384759

[17] Kaveh, A., Biabani Hamedani, K., Zaerreza, A. "A set theoretical shuffled shepherd optimization algorithm for optimal design of cantilever retaining wall structures", Engineering with Computers, 37(4), pp. 3265–3282, 2021. https://doi.org/10.1007/s00366-020-00999-9

[18] Talatahari, S., Kaveh, A., Sheikholeslami, R. "Chaotic imperialist competitive algorithm for optimum design of truss structures", Structural and Multidisciplinary Optimization, 46(3), pp. 355–367, 2012. https://doi.org/10.1007/s00158-011-0754-4

[19] Rao, R. V., Savsani, V. J., Vakharia, D. P. "Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems", Computer-Aided Design, 43(3), pp. 303–315, 2011. https://doi.org/10.1016/j.cad.2010.12.015

[20] Peitgen, H.-O., Jürgens, H., Saupe, D. "Chaos and Fractals: New Frontiers of Science", [e-book] Springer, 2006. ISBN 978-0-387-21823-6 https://doi.org/10.1007/b97624

[21] Yousefpoor, H., Kaveh, A. "Chaos-embedded meta-heuristic algorithms for optimal design of truss structures", Scientia Iranica, Transactions A: Civil Engineering, 29(6), pp. 2868–2885, 2022. https://doi.org/10.24200/sci.2022.59812.6441

[22] Kaveh, A., Yosefpoor, H. "Chaotic Meta-heuristic Algorithms for Optimal Design of Structures", Springer, 2024. ISBN 978-3-031-48917-4 https://doi.org/10.1007/978-3-031-48918-1

[23] Kaveh, A., Zarfam, P., Aziminejad, A., Yosefpoor, H. "Comparison of Four Chaotic MetaHeuristic Algorithms for Optimal Design of LargeScale Truss Structures", Iranian Journal of Science and Technology, Transactions of Civil Engineering, 46(6), pp. 4067–4091, 2022. https://doi.org/10.1007/s40996-022-00908-8

[24] Kaveh, A., Yosefpoor, H. "Chaotic optimization of trusses with frequency constraints with three metaheuristic algorithms", Iranian Journal of Science and Technology, Transactions of Civil Engineering, 48(1), pp. 271–294, 2024. https://doi.org/10.1007/s40996-023-01223-6

[25] Lee, K. S., Han, S. W., Geem, Z. W. "Discrete size and discrete-continuous configuration optimization methods for truss structures using the harmony search algorithm", International Journal of Optimization in Civil Engineering, 1(1), pp. 107–126, 2011.

[26] Ott, E. "Chaos in Dynamical Systems", Cambridge University Press, 2002. ISBN 9780521010849 https://doi.org/10.1017/CBO9780511803260

[27] Bucolo, M., Caponetto, R., Fortuna, L., Frasca, M., Rizzo, A. "Does chaos work better than noise?, IEEE Circuits and Systems Magazine, 2(3), pp. 4–19, 2002. https://doi.org/10.1109/MCAS.2002.1167624

[28] Lógó, J., Ghaemi, M., Movahedi Rad, M. "Optimal topologies in case of probabilistic loading: The influence of load correlation", Mechanics Based Design of Structures and Machines, 37(3), pp. 327–348, 2009. https://doi.org/10.1080/15397730902936328

[29] Movahedi Rad, M., Habashneh, M., Lógó, J. "Reliability based bi-directional evolutionary topology optimization of geometric and material nonlinear analysis with imperfections", Computers and Structures, 287, 107120, 2023. https://doi.org/10.1016/j.compstruc.2023.107120

[30] Habashneh, M., Movahedi Rad, M. "Plastic-limit probabilistic structural topology optimization of steel beams", Applied Mathematical Modelling, 128, pp. 347–369, 2024. https://doi.org/10.1016/j.apm.2024.01.029

[31] Kaveh, A., Malakoutirad, S. "Hybrid genetic algorithm and particle swarm optimization for the force method-based simultaneous analysis and design", Iranian Journal of Science and Technology, Transaction B- Engineering, 34(B1), pp. 15–34, 2010.

[32] Kaveh, A. "Improved cycle bases for the flexibility analysis of structures", Computer Methods in Applied Mechanics and Engineering, 9(3), pp. 267–272, 1976. https://doi.org/10.1016/0045-7825(76)90031-1