

Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems

Amir Hossein Gandomi · Xin-She Yang ·
Amir Hossein Alavi

Received: 28 December 2010 / Accepted: 11 July 2011 / Published online: 29 July 2011
© Springer-Verlag London Limited 2011

Abstract In this study, a new metaheuristic optimization algorithm, called cuckoo search (CS), is introduced for solving structural optimization tasks. The new CS algorithm in combination with Lévy flights is first verified using a benchmark nonlinear constrained optimization problem. For the validation against structural engineering optimization problems, CS is subsequently applied to 13 design problems reported in the specialized literature. The performance of the CS algorithm is further compared with various algorithms representative of the state of the art in the area. The optimal solutions obtained by CS are mostly far better than the best solutions obtained by the existing methods. The unique search features used in CS and the implications for future research are finally discussed in detail.

Keywords Cuckoo search · Structural optimization · Engineering design · Metaheuristic algorithm

1 Introduction

Most design optimization problems in structural engineering are highly nonlinear, involving many different design

variables under complex constraints. These constraints can be written either as simple bounds such as the ranges of material properties or as nonlinear relationships including maximum stress, maximum deflection, minimum load capacity, and geometrical configuration. Such nonlinearity often results in multimodal response landscape. Subsequently, local search algorithms such as hill-climbing and Nelder-Mead downhill simplex methods are not suitable; only global algorithms should be used so as to obtain optimal solutions [1].

Metaheuristic algorithms can be defined as upper level general methodologies (templates) that can be used as guiding strategies in designing underlying heuristics to solve specific optimization problems [2]. Two important characteristics of metaheuristics are intensification and diversification [3]. Intensification intends to search around the current best solutions and select the best candidates or solutions. Diversification makes sure that the algorithm can explore the search space more efficiently, often by randomization.

Modern metaheuristic algorithms have been developed with an aim to carry out global search with three main purposes: solving problems faster, solving large problems, and obtaining robust algorithms [2]. Genetic algorithms (GA) and particle swarm optimization (PSO) are typical examples of these algorithms. The efficiency of metaheuristic algorithms can be attributed to the fact that they imitate the best features in nature, especially the selection of the fittest in biological systems which have evolved by natural selection over millions of years. Timeline of main metaheuristic algorithms is shown in Fig. 1.

As shown in Fig. 1, cuckoo search (CS) is a new metaheuristic search algorithm. This algorithm is based on the obligate brood parasitic behavior of some cuckoo species in combination with the Lévy flight behavior of

A. H. Gandomi (✉)
Department of Civil Engineering, Tafresh University,
Tafresh, Iran
e-mail: a.h.gandomi@gmail.com

X.-S. Yang
Mathematics and Scientific Computing, National Physical Lab,
Teddington TW11 0LW, UK
e-mail: xy227@cam.ac.uk

A. H. Alavi
School of Civil Engineering, Iran University of Science
and Technology, Narmak, Tehran, Iran
e-mail: ah_alavi@hotmail.com

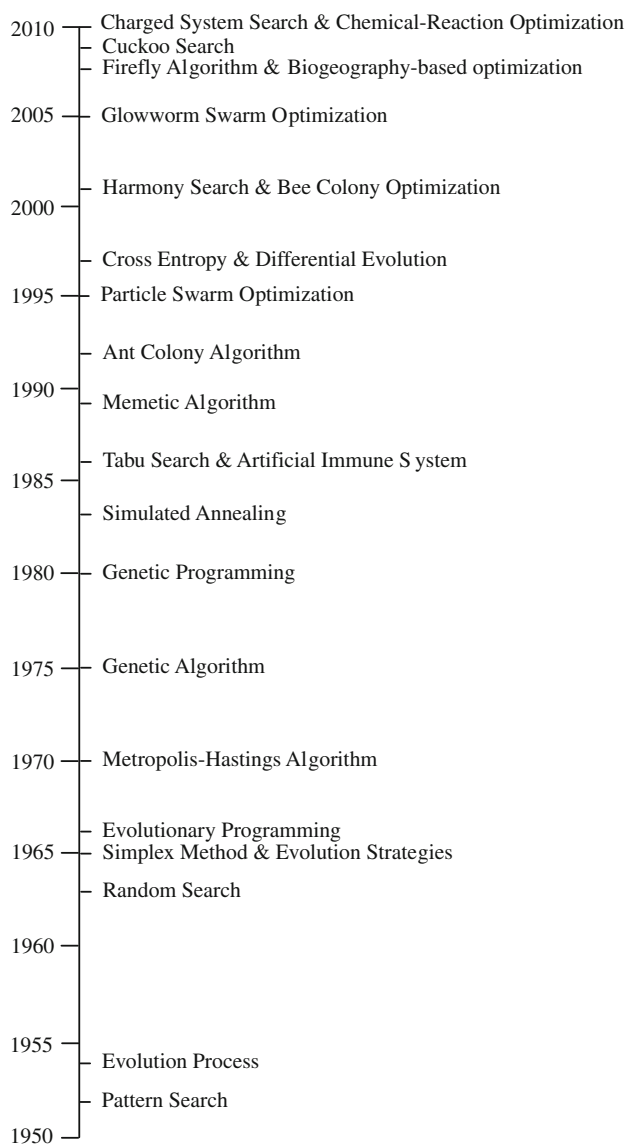


Fig. 1 Timeline of main metaheuristics algorithms

some birds and fruit flies. It is developed by Yang and Deb [4] and the preliminary studies show that it is very promising and could outperform existing algorithms such as GA and PSO [4]. In this paper, the CS algorithm is further validated against various structural engineering optimization problems including stochastic test functions. The introduced search strategy is compared with other popular optimization algorithms. Finally, the unique features of CS are discussed and topics for further studies are proposed.

2 Cuckoo search algorithm

In order to describe the CS algorithm more clearly, let us briefly review the interesting breed behavior of certain

Cuckoo Search Algorithm

begin

Objective function $f(x)$, $x=(x_1, \dots, x_d)^T$;

Initial a population of n host nests x_i ($i=1, 2, \dots, n$);

while ($t < \text{Maximum Generation}$) or (stop criterion);

 Get a cuckoo (say i) randomly

 and generate a new solution by Lévy flights;

 Evaluate its quality/fitness; F_i

 Choose a nest among n (say j) randomly;

if ($F_i > F_j$),

 Replace j by the new solution;

end

 Abandon a fraction (P_a) of worse nests

 [and build new ones at new locations via Lévy flights];

 Keep the best solutions (or nests with quality solutions);

 Rank the solutions and find the current best;

end while

 Post process results and visualization;

end

Fig. 2 Pseudo code of the Cuckoo Search method

cuckoo species. Then, we will outline the basic ideas and steps of the CS algorithm.

2.1 Cuckoo breeding behavior

Cuckoos are fascinating birds, not only because of the beautiful sounds they can make, but also because of their aggressive reproduction strategy. Some species such as the *ani* and *guira* cuckoos lay their eggs in communal nests, though they may remove others' eggs to increase the hatching probability of their own eggs [5]. Quite a number of species engage the obligate brood parasitism by laying their eggs in the nests of other host birds (often other species). There are three basic types of brood parasitism: intraspecific brood parasitism, cooperative breeding, and nest takeover. Some host birds can engage direct conflict with the intruding cuckoos. If a host bird discovers the eggs are not its own, it will either throw these alien eggs away or simply abandon its nest and build a new nest elsewhere. Some cuckoo species such as the new world brood-parasitic *Tapera* have evolved in such a way that female parasitic cuckoos are often very specialized in the mimicry in color and pattern of the eggs of a few chosen host species [5]. This reduces the probability of their eggs being abandoned and thus increases their reproductively.

Furthermore, the timing of egg-laying of some species is also amazing. Parasitic cuckoos often choose a nest where the host bird just laid its own eggs. In general, the cuckoo eggs hatch slightly earlier than their host eggs. Once the first cuckoo chick is hatched, the first instinctive action it will take is to evict the host eggs by blindly propelling the

eggs out of the nest, which increases the cuckoo chick’s share of food provided by its host bird [5]. Studies also show that a cuckoo chick can also mimic the call of host chicks to gain access to more feeding opportunity.

2.2 Lévy flights

In nature, animals search for food in a random or quasi-random manner. In general, the foraging path of an animal is effectively a random walk because the next move is based on the current location/state and the transition probability to the next location. Which direction it chooses depends implicitly on a probability which can be modeled mathematically. For example, various studies have shown that the flight behavior of many animals and insects has demonstrated the typical characteristics of Lévy flights [6, 7].

A recent study by Reynolds and Frye [8] shows that fruit flies or *Drosophila melanogaster* explore their landscape using a series of straight flight paths punctuated by a sudden 90° turn, leading to a Lévy-flight-style intermittent scale-free search pattern. Studies on human behavior such as the Ju/’hoansi hunter-gatherer foraging patterns also show the typical feature of Lévy flights. Even light can be related to Lévy flights [9]. Subsequently, such behavior has been applied to optimization and optimal search, and preliminary results show its promising capability [10].

2.3 Cuckoo search

For simplicity in describing the new CS method [4], we now use the following three idealized rules:

- Each cuckoo lays one egg at a time, and dumps it in a randomly chosen nest.
- The best nests with high quality of eggs (solutions) will carry over to the next generations.
- The number of available host nests is fixed, and a host can discover an alien egg with a probability $P_a \in [0, 1]$. In this case, the host bird can either throw the egg away or abandon the nest so as to build a completely new nest in a new location.

For simplicity, this last assumption can be approximated by a fraction P_a of the n nests being replaced by new nests (with new random solutions at new locations). For a maximization problem, the quality or fitness of a solution can simply be proportional to the objective function. Other forms of fitness can be defined in a similar way to the fitness function in GA.

Based on these three rules, the basic steps of the CS can be summarized as the pseudo code as follows:

When generating new solutions $x^{(t+1)}$ for, say cuckoo i , a Lévy flight is performed

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus \text{Lévy}(\lambda) \tag{1}$$

where $\alpha > 0$ is the step size which should be related to the scales of the problem of interest. In most cases, we can use $\alpha = O(1)$. The product \oplus means entry-wise multiplications. Lévy flights essentially provide a random walk, while their random steps are drawn from a Lévy distribution for large steps

$$\text{Lévy} \sim u = t^{-\lambda}, (1 < \lambda \leq 3) \tag{2}$$

which has an infinite variance with an infinite mean. Here, the consecutive jumps/steps of a cuckoo essentially form a random walk process which obeys a power-law step-length distribution with a heavy tail. Pseudo code of the cuckoo search algorithm is presented in Fig. 2.

3 Implementation and numerical experiments

3.1 Validation

Before solving the structural engineering problems, the CS was benchmarked using a well-known problem, namely, Himmelblau’s problem. This problem has originally been proposed by Himmelblau’s [11] and it has been widely used as a benchmark nonlinear constrained optimization problem. In this problem, there are five design variables $[x_1, x_2, x_3, x_4, x_5]$, six nonlinear inequality constraints, and ten boundary conditions. The problem can be stated as follows:

$$\begin{aligned} \text{Minimize } : f(X) \\ = 5.3578547x_3^2 + 0.8356891x_1x_5 \\ + 37.293239x_1 - 40792.141 \end{aligned} \tag{3}$$

Subject to $0 \leq g_1 \leq 92, 90 \leq g_2 \leq 110$, and $20 \leq g_3 \leq 25$ where

$$g_1 = 85.334407 + 0.0056858x_1x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 \tag{4}$$

$$g_2 = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 - 0.0021813x_3^2 \tag{5}$$

$$g_3 = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 - 0.0019085x_3x_4 \tag{6}$$

and

$$78 \leq x_1 \leq 102, 33 \leq x_2 \leq 45, \text{ and } 27 \leq x_3, x_4, x_5 \leq 45.$$

The best-known optimum for the Himmelblau’s problem obtained by CS is depicted in Table 1.

The problem was solved by Himmelblau [11] using generalized gradient method. This problem was also solved using several other methods such as GA (Gen and Cheng [12], Homaifar et al. [13]), harmony search algorithm (Lee and Geem [14], Fesanghary et al. [15]), and PSO (He et al.

Table 1 CS results for the Himmelblau's problem

| $X [x_1, x_2, x_3, x_4, x_5]$ | F_{\min} | No. cuckoos | No. evals. | Ave. time (s) |
|--|------------|-------------|------------|---------------|
| [78.00000, 33.00000, 29.99616, 45.00000, 36.77605] | -30665.233 | 25 | 5,000 | 3.2664 |

[16], Shi and Eberhart [17]). Table 2 illustrates the results obtained by CS, as well as those published in the literature. It is obvious from Table 2 that the result obtained using CS is better than the best feasible solution previously reported.

3.2 Structural optimization problems

Structural optimization problems are complex, sometimes even the optimal solutions of interest do not exist. In order to see how the CS algorithm performs, 12 standard structural engineering test problems are solved.

3.2.1 Case I. Structural design of a pin-jointed plane frame

The first case is design of a pin-jointed plane frame with a fixed base for minimum weight utilizing CS. This case study is originally presented by Majid [20]. Figure 3a shows the frame. The vertical deflections at joints 1 and 2 are limited to 5 mm. All members of the frame have the same cross-sectional area (A) equal to 100 mm²; the vertical forces P1 and P2 are, respectively, 100 and 50 kN, and the length of the base (l) is 1,000 mm. The frame is Symmetric and therefore, just half of the frame is considered, as illustrated in Fig. 3b. Thus, member 3 has half the cross-sectional area of the other members. Joints 1 and 2 move vertically, and the joint displacement vector is $\Delta = (\Delta_1, \Delta_2)^T$. As shown in Fig. 3b, the angles of members 1 and 2 (θ_1 and θ_2) specify the design and define the structure of this frame.

Then the lengths of the members can be calculated using the following equations:

$$l_1 = \frac{l}{2 \cos(\theta_1)} \quad (7)$$

$$l_2 = \frac{l}{2 \cos(\theta_2)} \quad (8)$$

$$l_3 = \frac{l}{2 \cos(\theta_1) \cos(\theta_2) \sqrt{\cos^2(\theta_1) + \cos^2(\theta_2) - 2 \cos(\theta_1) \cos(\theta_2) \cos(\theta_1 - \theta_2)}} \quad (9)$$

The materials and cross-sectional areas of all the members are assumed to be the same. Hence, the weight of the frame is a linear function of the total length of the members. Thus, the objective function of the problem for the three member frame (NM is the Number of Members) will be as follows:

$$\text{Minimize : } f(\theta_1, \theta_2) = \sum_{i=1}^{NM} l_i \quad (10)$$

subject to displacements less than 5:

$$|\Delta_1(\theta_1, \theta_2)| \leq 5 \quad (11)$$

$$|\Delta_2(\theta_1, \theta_2)| \leq 5 \quad (12)$$

where $-\pi/3 \leq \theta_1, \theta_2 \leq \pi/3$ and the displacements $\Delta = (\Delta_1, \Delta_2)^T$ can be derived as

$$K\Delta = F \text{ or } F = K^{-1}F \text{ and } K = B^T k B$$

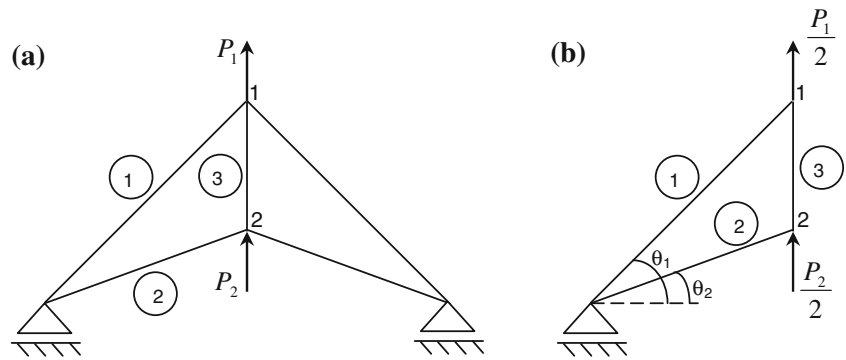
Table 2 Statistical results for the Himmelblau's problem

| Author(s) | Year | SD | Best | $0 \leq g_1 \leq 92$ | $90 \leq g_2 \leq 110$ | $20 \leq g_3 \leq 25$ |
|------------------------|------|---------|-------------|-----------------------------|------------------------|-----------------------|
| Himmelblau [11] | 1972 | N/A | -30373.9490 | N/A | N/A | N/A |
| Homaifar et al. [13] | 1994 | N/A | -30005.7000 | 91.65619 | 99.53690 | 20.02553 |
| Gen and Cheng [12] | 1997 | N/A | -30183.5760 | N/A | N/A | N/A |
| Shi and Eberhart [17] | 1998 | N/A | -31025.5610 | 93.28533^a | 100.40473 | 19.99997 |
| Coello [18] | 2000 | 73.6335 | -31020.8590 | 93.28381 | 100.40786 | 20.00191 |
| He et al. [16] | 2004 | 70.0400 | -30643.9900 | 93.28536 | 100.40478 | 20.00000 |
| Lee and Geem [14] | 2004 | N/A | -30665.5000 | 92.00004 | 98.84051 | 19.99994 |
| Fesanghary et al. [15] | 2008 | N/A | -31024.3160 | 93.27834 | 100.39612 | 20.00000 |
| Omran and Salman [19] | 2009 | N/A | -31025.5560 | 93.28536 | 100.40478 | 20.00000 |
| Present study | | 11.6231 | -30665.2327 | 91.99996 | 98.84067 | 20.00036 |

SD standard deviation

^a Bold sets are violated sets

Fig. 3 Pin-jointed plane frame example



where

$$k = \begin{bmatrix} \frac{EA}{l_1} & 0 & 0 \\ 0 & \frac{EA}{l_2} & 0 \\ 0 & 0 & \frac{EA}{l_3} \end{bmatrix} \tag{13}$$

and

$$B = \begin{bmatrix} \sin(\theta_1) & 0 \\ 0 & \sin(\theta_2) \\ 1 & -1 \end{bmatrix} \tag{14}$$

Therefore, the displacements Δ for the problem are given by $K\Delta = F$, which is

$$EA \begin{bmatrix} \frac{\sin^2(\theta_1)}{l_1} + \frac{1}{2l_3} & -\frac{1}{2l_3} \\ -\frac{1}{2l_3} & \frac{\sin^2(\theta_2)}{l_2} + \frac{1}{2l_3} \end{bmatrix} \begin{bmatrix} \Delta_1 \\ \Delta_2 \end{bmatrix} = \begin{bmatrix} \frac{P_1}{2} \\ \frac{P_2}{2} \end{bmatrix} \tag{15}$$

The assumption is that two frames are remarkably different if their shapes are significantly different. This criterion is characterized by the angles of the members. Thus, the distance between two frames can be specified using the Euclidean distance between them in design space.

The CS algorithm was run to find the global optima of the above design problem. With 25 cuckoos, CS had located two global optima at a computational cost of 500 function evaluations per solution. The results of this study and the best results reported in [21] are presented in Table 3 and the frame shapes are shown in Fig. 4. Note that $\theta_1 = \theta_2$ for both global solutions. This is to be expected, because the weight (total length) is selected as the objective function. Before CS was used to solve this problem, its objective function was thought to be multimodal with two global solutions. The results indicate that the CS algorithm has successfully obtained two global solutions, as expected. As it can be seen from Table 3, the CS results are better than the solution obtained by Li et al. [21] using GA.

3.2.2 Case II. Minimize the vertical deflection of an I-beam

The capability of the CS algorithm in solving real engineering design problems is tested using another design problem including four variables. This case is modified

from the original problem reported in [22]. The goal is to minimize the vertical deflection of an I-beam (see Fig. 5). It simultaneously satisfies the cross-sectional area and stress constraints under given loads.

Minimize the vertical deflection $f(x) = PL^3/48EI$ when length of the beam (L) and modulus of elasticity (E) are, respectively, 5,200 cm and 523,104 kN/cm². Thus, the objective function of the problem is considered to be as follows:

$$\text{Minimize : } f(b, h, t_w, t_f) = \frac{5000}{\frac{t_w(h-2t_f)^3}{12} + \frac{bt_f^3}{6} + 2bt_f\left(\frac{h-t_f}{2}\right)^2} \tag{16}$$

Subject to cross section area less than 300 cm²

$$g_1 = 2bt_w + t_w(h - 2t_f) \leq 300 \tag{17}$$

If allowable bending stress of the beam is 56 kN/cm² the stress constraint is as follows:

$$g_1 = \frac{18h \times 10^4}{t_w(h - 2t_f)^3 + 2bt_w(4t_f^2 + 3h(h - 2t_f))} + \frac{15b \times 10^3}{(h - 2t_f)t_w^3 + 2t_w b^3} \leq 6 \tag{18}$$

where the initial design spaces are $10 \leq h \leq 80$, $10 \leq b \leq 50$, $0.9 \leq t_w \leq 5$, and $0.9 \leq t_f \leq 5$.

For this case study, by increasing the number of function evaluations to 5,000 or 25,000, the results do not improve much. The CS method achieved a solution that satisfies the constraints and it reaches the best solution, possibly the unique global optimum. CS outperforms the previous other methods in terms of minimum objective function value. Table 4 presents the results obtained by CS. As it is seen, the CS method requires 25 cuckoos and 200 iterations to reach the optimum.

This nonlinear constrained problem has been solved with other optimization methods such as adaptive response surface method (ARSM) and improved ARSM [23]. A comparison of the results obtained CS, ARSM, and improved ARSM is shown in Table 5. As it is seen, CS performs superior than the ARSM-based methods.

Table 3 Statistical results of the pin-jointed plane frame example

| | Method | Shape (a) | | | Shape (b) | | |
|----------------|--------|------------------|------------------|-----------|------------------|------------------|-----------|
| | | θ_1 (rad) | θ_2 (rad) | F_{min} | θ_1 (rad) | θ_2 (rad) | F_{min} |
| Li et al. [21] | GA | 0.475784 | 0.472764 | 1125.98 | -0.478625 | -0.479701 | 1127.59 |
| Present study | CS | 0.477459 | 0.477446 | 1125.92 | -0.478069 | -0.476997 | 1126.64 |

Fig. 4 Optimal frame shapes

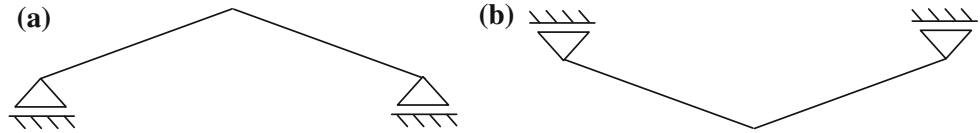


Fig. 5 A beam design problem ($P = 5600$ kN and $Q = 550$ kN)

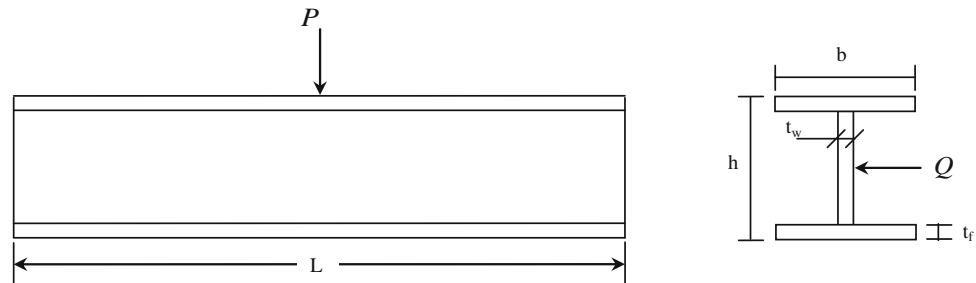


Table 4 Best solution results for the vertical deflection of an I-beam example using CS

| Best | Mean | Worst | SD | No. cuckoos | No. evals. |
|-----------|-----------|------------|-----------|-------------|------------|
| 0.0130747 | 0.0132165 | 0.01353646 | 0.0001345 | 25 | 5,000 |

Table 5 Statistical results of the vertical deflection of an I-beam example

| | CS | ARSM | Improved ARSM |
|-----------|-----------|--------|---------------|
| h | 80.000000 | 80.00 | 79.99 |
| b | 50.000000 | 37.05 | 48.42 |
| t_w | 0.900000 | 1.71 | 0.90 |
| t_f | 2.3216715 | 2.31 | 2.40 |
| F_{min} | 0.0130747 | 0.0157 | 0.131 |

3.2.3 Case III. Piston lever

This problem was first presented in [24]. The main objective is to locate the piston components, H , B , D , and X by minimizing the oil volume when the lever of the piston is lifted up from 0° to 45° as shown in Fig. 6. The objective function of the problem is given as follows:

$$\text{Minimize : } f(H, B, D, X) = \frac{1}{4} \pi D^2 (L_2 - L_1) \tag{19}$$

Subject to:

$$g_1 = QL \cos \theta - RF \leq 0 \quad \text{at } \theta = 45^\circ \tag{20}$$

$$g_2 = Q(L - X) - M_{max} \leq 0 \tag{21}$$

$$g_3 = 1.2(L_2 - L_1) - L_1 \leq 0 \tag{22}$$

$$g_4 = D/2 - B \leq 0 \tag{23}$$

where

$$R = \frac{|-X(X \sin \theta + H) + H(B - X \cos \theta)|}{\sqrt{(X - B)^2 + H^2}} \tag{24}$$

$$F = \pi PD^2/4 \tag{25}$$

$$L_1 = \sqrt{(X - B)^2 + H^2} \tag{26}$$

$$L_2 = \sqrt{(X \sin 45 + H)^2 + (B - X \cos 45)^2} \tag{27}$$

where the payload is $P = 10,000$ lbs, the lever is $L = 240$ in, the maximum allowable bending moment of the lever is $6 \text{ max } M = 1.8 \times 10$ lbs in, and the oil pressure is given as 1,500 psi. A number of inequality constraints are imposed. Force equilibrium, maximum bending moment of the lever, minimum piston stroke, and geometrical conditions are considered. The best solutions obtained by CS are presented in Table 6.

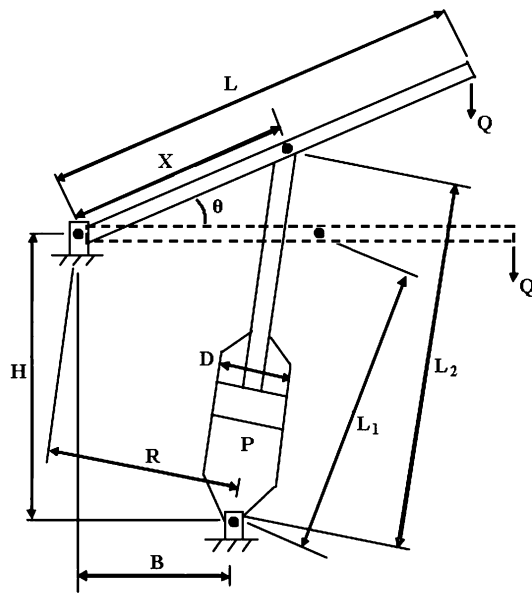


Fig. 6 Piston problem

The best results obtained by the CS algorithm with linear adjusting over 50 independent runs have been compared with those reported in [25]. The performance of each of the algorithms is summarized in Table 7. In this example problem, the performance of CS is remarkable in terms of maximum, minimum, and mean values.

3.2.4 Case IV. Corrugated bulkhead design

This problem is as an example of minimum-weight design of the corrugated bulkheads for a tanker (Kavlie et al. [26]). The variables of the problem are width (b), depth (h), length (l), and plate thickness (t). The minimum-weight

requires the solution of the following optimization problem:

$$\text{Minimize : } f(b, h, l, t) = \frac{5.885t(b + l)}{b + \sqrt{(l^2 - h^2)}} \tag{28}$$

Subject to:

$$g_1 = th \left(0.4b + \frac{l}{6} \right) - 8.94(b + \sqrt{(l^2 - h^2)}) \geq 0 \tag{29}$$

$$g_2 = th^2 \left(0.2b + \frac{l}{12} \right) - 2.2 \left(8.94 \left(b + \sqrt{(l^2 - h^2)} \right) \right)^{\frac{4}{3}} \geq 0 \tag{30}$$

$$g_3 = t - 0.0156b - 0.15 \geq 0 \tag{31}$$

$$g_4 = t - 0.0156l - 0.15 \geq 0 \tag{32}$$

$$g_5 = t - 1.05 \geq 0 \tag{33}$$

$$g_6 = l - h \geq 0 \tag{34}$$

where $0 \leq b, h, l \leq 100$ and $0 \leq t \leq 5$. The minimum-weight and the statistical values of the best solution obtained by the CS algorithm are given in Table 8. CS requires 25 cuckoos and 200 iterations to reach the optimum.

For this problem, Ravindran et al. [27] reported the minimum value of 6.84241 using a random search method. A comparison of the results clearly shows that CS notably improves the results obtained by the random search method.

3.2.5 Case V. Cantilever beam

This case is related to the weight optimization of a cantilever beam with square cross section (see Fig. 7) [28]. The beam is rigidly supported at node 1, and there is a given vertical force acting at node 6. The design variables are the

Table 6 Best solution results for the piston lever example using CS

| H (in) | B (in) | X (in) | D (in) | F_{min} | No. cuckoos | No. evals. | Time of each run (s) |
|--------|--------|---------|--------|-----------|-------------|------------|----------------------|
| 0.0500 | 2.0430 | 120.000 | 4.0851 | 8.4271 | 50 | 50,000 | 14.90 |

Table 7 Statistical results of the piston lever example

| | CS | PSO | DE | GA | HPSO | HPSO with Q-learning |
|------|----------|------|------|------|------|----------------------|
| Max | 168.5920 | 294 | 199 | 216 | 197 | 168 |
| Min | 8.4271 | 122 | 159 | 161 | 162 | 129 |
| Mean | 40.2319 | 166 | 187 | 185 | 187 | 151 |
| SD | 59.0552 | 51.7 | 14.2 | 18.2 | 13.4 | 13.4 |

DE differential evolution, HPSO hybrid PSO

Table 8 Statistical results of the corrugated bulkhead design example

| <i>b</i> (cm) | <i>h</i> (cm) | <i>l</i> (cm) | <i>t</i> (cm) | Best | Average | Worst | SD |
|---------------|---------------|---------------|---------------|----------|----------|----------|----------|
| 37.1179498 | 33.0350210 | 37.1939476 | 0.7306255 | 5.894331 | 5.988257 | 6.126749 | 0.064360 |

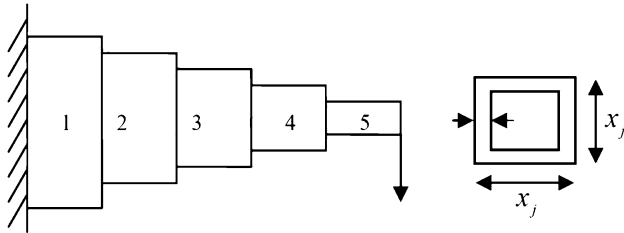


Fig. 7 Cantilever beam

heights (or widths) of the different beam elements, and the thickness is held fixed (here $t = 2/3$). The bound constraints are set as $0.01 \leq x_j \leq 100$. This problem can be expressed analytically as follows:

$$\text{Minimize : } f(X) = 0.0624(x_1 + x_2 + x_3 + x_4 + x_5) \quad (35)$$

$$\text{Subject to : } g(X) = \frac{61}{x_1^3} + \frac{37}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} - 1 \leq 0 \quad (36)$$

The best solutions obtained using CS and various methods [29] for solving this problem are listed in Table 9. As it is seen, the solution found out by the CS approach is slightly better than those of other methods. In this problem, the CS terminated after 2500 searches with 50 cuckoos.

3.2.6 Case VI. Tubular column design

Figure 8 presents an example for designing a uniform column of tubular section to carry a compressive load $P = 2,500$ kgf at minimum cost [30]. The column is made of a material with a yield stress (σ_y) of 500 kgf/cm², a modulus of elasticity (E) of 0.85×10^6 kgf/cm², and a density (ρ) equal to 0.0025 kgf/cm³. The length (L) of the column is 250 cm. The stress included in the column

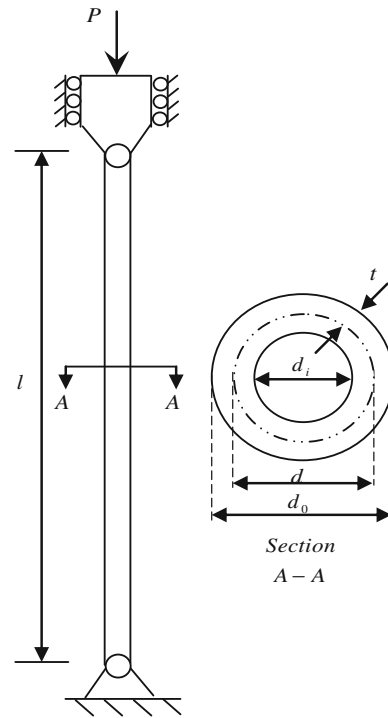


Fig. 8 The tubular column

should be less than the buckling stress (constraint g_1) and the yield stress (constraint g_2). The mean diameter of the column is restricted between 2 and 14 cm (constraint g_3 and g_4), and columns with thickness outside the range 0.2–0.8 cm are not commercially available (constraint g_5 and g_6). The cost of the column includes material and construction costs. It is taken as the objective function. The optimization model of this problem is given as follows:

$$\text{Minimize : } f(d, t) = 9.8dt + 2d \quad (37)$$

Table 9 Best results of the cantilever beam design example

| Methods | x_1 | x_2 | x_3 | x_4 | x_5 | F_{\min} |
|---------|--------|--------|--------|--------|--------|------------|
| CONLIN | 6.0100 | 5.3000 | 4.4900 | 3.4900 | 2.1500 | NC |
| MMA | 6.0100 | 5.3000 | 4.4900 | 3.4900 | 2.1500 | 1.3400 |
| GCA(I) | 6.0100 | 5.3000 | 4.4900 | 3.4900 | 2.1500 | 1.3400 |
| GCA(II) | 6.0100 | 5.3000 | 4.4900 | 3.4900 | 2.1500 | 1.3400 |
| CS | 6.0089 | 5.3049 | 4.5023 | 3.5077 | 2.1504 | 1.33999 |

CONLIN CONvex LINearization, MMA method of moving asymptotes, GCA generalized convex approximation, NC not converge

Table 10 Statistical results of the best model for tubular column design example

| No. cuckoos | No. evals. | Best | Average | Worst | SD | Time of each run |
|-------------|------------|----------|----------|----------|---------|------------------|
| 25 | 15,000 | 26.53217 | 26.53504 | 26.53972 | 0.00193 | 2.56919 |

Subject to

$$g_1 = \frac{P}{\pi dt\sigma_y} - 1 \leq 0 \tag{38}$$

$$g_2 = \frac{8PL^2}{\pi^3 Edt(d^2 + t^2)} - 1 \leq 0 \tag{39}$$

$$g_3 = \frac{2.0}{d} - 1 \leq 0 \tag{40}$$

$$g_4 = \frac{d}{14} - 1 \leq 0 \tag{41}$$

$$g_5 = \frac{0.2}{t} - 1 \leq 0 \tag{42}$$

$$g_6 = \frac{t}{0.8} - 1 \leq 0 \tag{43}$$

Table 10 illustrates the statistical results for the best objective value by CS. Table 11 compares the results obtained by CS with those reported in the literature [30, 31]. It can be observed from Table 11 that the best objective values by other methods are not feasible because the second constraint (g_2) is violated. Thus, the CS algorithm provides the best results.

3.2.7 Case VII. A three-bar truss design

This case considers a 3-bar planar truss structure shown in Fig. 9. This problem was first presented by Nowcki [32]. The volume of a statically loaded 3-bar truss is to be minimized subject to stress (σ) constraints on each of the truss members. The objective is to evaluate the optimal cross sectional areas (A_1, A_2). The mathematical formulation is given as below:

Table 11 Best solutions for the tubular column example

| | Hsu and Liu [31] | Rao [30] | Present study |
|------------|---------------------------|---------------|---------------|
| d | 5.4507 | 5.44 | 5.45139 |
| T | 0.292 | 0.293 | 0.29196 |
| g_1 | -7.8×10^{-5} | -0.8579 | -0.0241 |
| g_2 | 0.1317^a | 0.0026 | -0.1095 |
| g_3 | -0.6331 | -0.8571 | -0.6331 |
| g_4 | -0.6107 | 0 | -0.6106 |
| g_5 | -0.3151 | -0.7500 | -0.3150 |
| g_6 | -0.6350 | 0 | -0.6351 |
| F_{\min} | 25.5316 | 26.5323 | 26.53217 |

^a Bold sets are violated sets

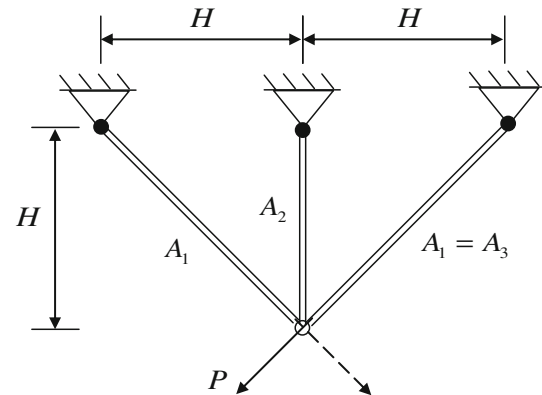


Fig. 9 Three-bar truss

$$\text{Minimize : } f(A_1, A_2) = (2\sqrt{2A_1} + A_2) \times l \tag{44}$$

Subject to

$$g_1 = \frac{\sqrt{2A_1 + A_2}}{\sqrt{2A_1^2 + 2A_1A_2}} P - \sigma \leq 0 \tag{45}$$

$$g_2 = \frac{A_2}{\sqrt{2A_1^2 + 2A_1A_2}} P - \sigma \leq 0 \tag{46}$$

$$g_3 = \frac{1}{A_1 + \sqrt{2A_2}} P - \sigma \leq 0 \tag{47}$$

where

$$0 \leq A_1 \leq 1 \text{ and } 0 \leq A_2 \leq 1; l = 100 \text{ cm, } P = 2\text{KN/cm}^2, \text{ and } \sigma = 2\text{KN/cm}^2$$

This design problem is a nonlinear fractional programming problem. The statistical values of the best solution obtained by the CS algorithm are given in Table 12. The solution by CS is $(A_1, A_2) = (0.78867, 0.40902)$ with the objective value equal to 263.97156. Table 13 presents the solutions obtained by CS and those reported by Ray and Saini [33] and Tsai [34]. As it is seen, the best objective value reported by Tsai [34] is not feasible because the first constraint (g_1) is violated. Hence, it can be concluded that the results obtained by CS are better than those of the previous studies.

3.2.8 Case VIII. Gear train design

Gear train design problem is an unconstrained optimization. This problem has four integer variables and was

Table 12 Statistical results of the best three-bar truss model

| No. cuckoos | No. evals. | Best | Average | SD | Time of each run (s) |
|-------------|------------|-----------|----------|---------|----------------------|
| 25 | 15,000 | 263.97156 | 264.0669 | 0.00009 | 0.52808 |

Table 13 Best solutions for the three-bar truss design example

| | Ray and Saini [33] | Tsai [34] | Present study |
|-----------|--------------------|----------------------------|---------------|
| A_1 | 0.795 | 0.788 | 0.78867 |
| A_2 | 0.395 | 0.408 | 0.40902 |
| g_1 | -0.00169 | 0.00082^a | -0.00029 |
| g_2 | -0.26124 | -0.2674 | -0.26853 |
| g_3 | -0.74045 | -0.73178 | -0.73176 |
| f_{min} | 264.3 | 263.68 | 263.9716 |

^a Bold sets are violated sets

introduced by Sandgran [35]. It consists of the minimization of the cost of the gear ratio of the gear train shown in Fig. 10. The gear ratio is defined as follows:

$$\text{Gear ratio} = \frac{T_d T_b}{T_a T_f} \tag{48}$$

where T_i denotes the number of teeth of the gearwheel i and they are all integers varying in the range 12–60. The mathematical formulation is as follows:

$$f(T_d, T_b, T_a, T_f) = \left(\frac{1}{6.931} - \frac{T_d T_b}{T_a T_f} \right) \tag{49}$$

Table 14 shows the statistical results for the best objective value by CS. The obtained solution by CS is $(T_d, T_b, T_a, T_f) = (19, 16, 43, 49)$ with the objective value $2.70095.712 \times 10^{-12}$. The computational results obtained using different methods are shown in Table 15. As can be observed from this table, the CS results are significantly better than those reported by Sandgren [35] and Kannan

and Kramer [36]. It can also be seen that CS provide similar results to those obtained by Deb and Goyal [37].

3.2.9 Case IX. Speed reducer design

Cs is applied to the design of a speed reducer which is a benchmark structural optimization problem [38] (see in Fig. 11), with the face width (b), module of teeth (m), number of teeth on pinion (z), length of shaft 1 between bearings (l_1), length of shaft 2 between bearings (l_2), diameter of shaft 1 (d_1), and diameter of shaft 2 (d_2). The objective is to minimize the total weight of the speed reducer. The constraints involve limitations on the bending stress of the gear teeth, surface stress, transverse deflections of shafts 1 and 2 due to transmitted force, and stresses in shafts 1 and 2.

The mathematical formulation can be summarized as follows:

$$\begin{aligned} \text{Minimize } & f(b, m, z, l_1, l_2, d_1, d_2) \\ & = 0.7854bm^2(3.3333z^2 + 14.9334z - 43.0934) \\ & \quad - 1.508b(d_1^2 + d_2^2) + 7.477(d_1^3 + d_2^3) \\ & \quad + 0.7854(l_1 d_1^2 + l_2 d_2^2) \end{aligned} \tag{50}$$

Subject to :

$$g_1 = \frac{27}{bm^2z}P - 1 \leq 0 \tag{51}$$

$$g_2 = \frac{397.5}{bm^2z^2} - 1 \leq 0 \tag{52}$$

$$g_3 = \frac{1.93}{mzl_1^3d_1^4} - 1 \leq 0 \tag{53}$$

$$g_4 = \frac{1.93}{mzl_2^3d_2^4} - 1 \leq 0 \tag{54}$$

$$g_5 = \frac{\sqrt{\left(\frac{745l_1}{mz}\right)^2 + 1.69 \times 10^6}}{110d_1^3} - 1 \leq 0 \tag{55}$$

Fig. 10 Gear train

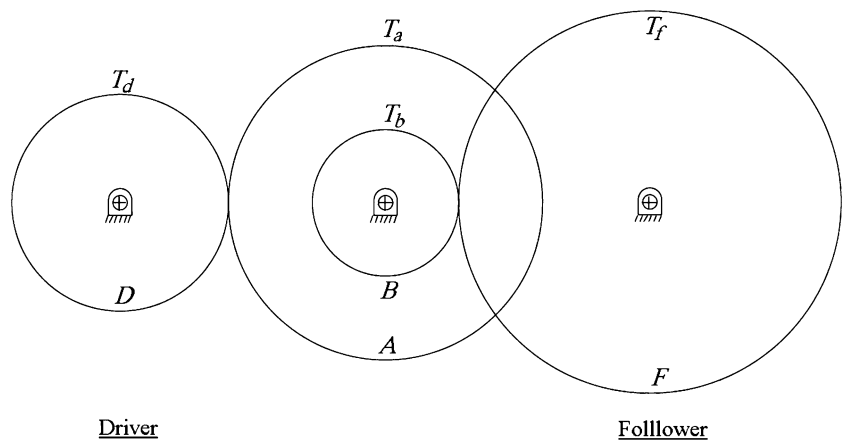


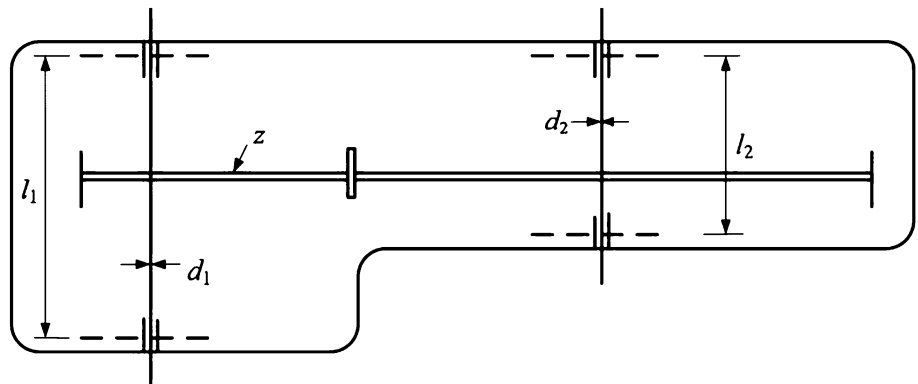
Table 14 Statistical results of the gear train model by CS

| No. cuckoos | No. evals. | Best | Average | Worst | SD | Time of each run |
|-------------|------------|--------------------------|-------------------------|-------------------------|-------------------------|------------------|
| 25 | 5,000 | 2.7009×10^{-12} | 1.9841×10^{-9} | 2.3576×10^{-9} | 3.5546×10^{-9} | 3.25003 |

Table 15 Comparison results of the best solutions for the gear train design example

| | Sandgren [35] | Kannan and Kramer [36] | Deb and Goyal [37] | Present study |
|------------|------------------------|------------------------|-------------------------|-------------------------|
| T_d | 18 | 13 | 19 | 19 |
| T_b | 22 | 15 | 16 | 16 |
| T_a | 45 | 33 | 49 | 43 |
| T_f | 60 | 41 | 43 | 49 |
| Gear ratio | 0.146667 | 0.144124 | 0.144281 | 0.144281 |
| f_{min} | 5.712×10^{-6} | 2.146×10^{-8} | 2.701×10^{-12} | 2.701×10^{-12} |

Fig. 11 Speed reducer



$$g_6 = \frac{\sqrt{\left(\frac{745l_1}{mz}\right)^2 + 157.5 \times 10^6}}{85d_2^3} - 1 \leq 0 \tag{56}$$

$$g_7 = \frac{mz}{40} - 1 \leq 0 \tag{57}$$

$$g_8 = \frac{5m}{B-1} - 1 \leq 0 \tag{58}$$

$$g_9 = \frac{b}{12m} - 1 \leq 0 \tag{59}$$

The corresponding statistical values of the Best CS model and the simple bounds of the speed reducer problem are presented in Table 16.

Table 17 presents a comparison of the results obtained by CS and other methods. As it is seen, the CS results are better than those reported by Akhtar et al. [41]. Although the best objective values derived by Ray and Saini [39] and Kuang et al. [40] are better than those of CS, the reported values are not feasible. This is because the fifth and sixth constraints (g_5, g_6) are significantly violated in the results

Table 16 Statistical results of the speed reducer design example

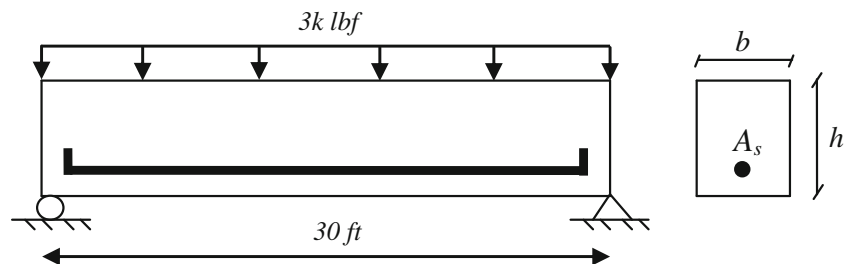
| | Bound | Value |
|--------------------------|-----------|-----------|
| b | (2.6–3.6) | 3.5015 |
| m | (0.7–0.8) | 0.7000 |
| z | (17–28) | 17.0000 |
| l_1 | (7.3–8.3) | 7.6050 |
| l_2 | (7.8–8.3) | 7.8181 |
| d_1 | (2.9–3.9) | 3.3520 |
| d_2 | (5.0–5.5) | 5.2875 |
| Objective function value | | 3000.9810 |
| No. cuckoos | | 50 |
| No. iterations | | 5,000 |
| Time of each run (s) | | 0.65081 |

of Ray and Saini [33]. The 6th constraint (g_6) was violated in the results by Kuang et al. [40]. It should be noted that some previous studies consider the simple bound of x_5 like x_4 so they are not considered in the comparison study.

Table 17 Statistical results of the speed reducer design example

| | Kuang et al. [40] | Akhtar et al. [41] | Ray and Saini [33] | Montes and Coello [39] | Present study |
|----------|---------------------------|--------------------|--------------------|------------------------|---------------|
| Best | 2876.117623 | 3008.08 | 2732.9006 | 3025.005 | 3000.9810 |
| b | 3.6 | 3.506122 | 3.514185 | 3.506163 | 3.5015 |
| m | 0.7 | 0.700006 | 0.700005 | 0.700831 | 0.7000 |
| z | 17 | 17 | 17 | 17 | 17.0000 |
| l_1 | 7.3 | 7.549126 | 7.497343 | 7.460181 | 7.6050 |
| l_2 | 7.8 | 7.85933 | 7.8346 | 7.962143 | 7.8181 |
| d_1 | 3.4 | 3.365576 | 2.9018 | 3.3629 | 3.3520 |
| d_2 | 5 | 5.289773 | 5.0022 | 5.3090 | 5.2875 |
| g_1 | -0.0996 | -0.0755 | -0.0777 | -0.0777 | -0.0743 |
| g_2 | -0.2203 | -0.1994 | -0.2012 | -0.2013 | -0.1983 |
| g_3 | -0.5279 | -0.4562 | -0.0360 | -0.4741 | -0.4349 |
| g_4 | -0.8769 | -0.8994 | -0.8754 | -0.8971 | -0.9008 |
| g_5 | -0.0433 | -0.0132 | 0.5395 | -0.0110 | -0.0011 |
| g_6 | 0.1821^a | -0.0017 | 0.1805 | -0.0125 | -0.0004 |
| g_7 | -0.7025 | -0.7025 | -0.7025 | -0.7022 | -0.7025 |
| g_8 | -0.0278 | -0.0017 | -0.0040 | -0.0006 | -0.0004 |
| g_9 | -0.5714 | -0.5826 | -0.5816 | -0.5831 | -0.5832 |
| g_{10} | -0.0411 | -0.0796 | -0.1660 | -0.0691 | -0.0890 |
| g_{11} | -0.0513 | -0.0179 | -0.0552 | -0.0279 | -0.0130 |
| Mean | NA | 3012.1200 | 2758.8878 | 3088.7778 | 3007.1997 |
| Max | NA | 3028.2800 | 2780.3071 | 3078.5918 | 3.0090 |
| SD | NA | NA | NA | NA | 4.9634 |

^a Bold sets are violated sets

Fig. 12 Illustration of reinforced concrete beam

3.2.10 Case X. A reinforced concrete beam design

A simplified optimization of the total cost of a reinforced concrete beam, shown on Fig. 12, was presented by Amir and Hasegawa [42]. The beam is assumed to be simply supported with a span of 30 ft and subjected to a live load of 2.0 klf and a dead load of 1.0 klf, including the weight of the beam. The concrete compressive strength (σ_c) is 5 ksi, and the yield stress of the reinforcing steel (σ_y) is 50 ksi. The cost of concrete is \$0.02/in²/linear ft and the cost of steel is \$1.0/in²/linear ft. The area of the reinforcement (A_s), the width of the beam (b), and the depth of the beam (h) have to be determined such that the total cost of structure is minimized. Herein, the cross-sectional area of the reinforcing bar (A_s) is taken as a

discrete type variable that must be chosen from the standard bar dimensions listed in [42]. The width of concrete beam (b) is assumed to be an integer variable. The variable h denoting the depth of the beam is a continuous variable. The effective depth is assumed to be $0.8x_2$.

The structure should be proportioned to have a required strength based upon the ACI building code 318-77 as follows:

$$Mu = 0.9A_s\sigma_y(0.8h) \left(1.0 - 0.59 \frac{A_s\sigma_y}{0.8bh\sigma_c} \right) \geq 1.4M_d + 1.7M_l \quad (60)$$

in which M_u , M_d and M_l are, respectively, the flexural strength, dead load, and live load moments of the beam. In

Table 18 Statistical results of the reinforced concrete beam example

| Reference | Amir and Hasegawa [42] | Shih and Yang [44] | | Yun [45] | | Present study |
|-----------|------------------------|----------------------|---------------------|----------|-----------------------|---------------|
| Method | SD-RC ^a | GHN-ALM ^b | GHN-EP ^c | GA | FLC-AHGA ^d | CS |
| f_{min} | 374.2 | 362.2455 | 362.00648 | 366.1459 | 364.8541 | 359.2080 |
| A_s | 7.8 | 6.6 | 6.32 | 7.20 | 6.16 | 6.32 |
| b | 31 | 33 | 34 | 32 | 35 | 34 |
| h | 7.79 | 8.495227 | 8.637180 | 8.0451 | 8.7500 | 8.5000 |
| g_1 | -4.2012 | 0.0159 ^e | -0.7745 | -2.8779 | -3.6173 | -0.2241 |
| g_2 | -0.0205 | -0.1155 | -0.0635 | -0.0224 | 0 | 0 |

- ^a Hybrid discrete steepest descent and rotating coordinate directions methods (SD-RC)
- ^b Generalized Hopfield network based augmented Lagrange multiplier approach (GHN-ALM)
- ^c GHN based extended penalty approach (GHN-EP)
- ^d Adaptive hybrid GA with fuzzy logic controller (flc-aHGA)
- ^e Violated set

this case, $M_d = 1,350$ in kip and $M_l = 2,700$ in kip. The depth to width ratio of the beam is restricted to be less than or equal to 4. The optimization problem can be expressed as

$$\text{Minimize : } f(A_s, b, h) = 2.9A_s + 0.6bh \tag{61}$$

subject to:

$$g_1 = \frac{h}{b} - 4 \leq 0 \tag{62}$$

$$g_2 = 180 + 7.375 \frac{A_s^2}{b} - A_s h \leq 0 \tag{63}$$

The variables bound are A_s : {6.0, 6.16, 6.32, 6.6, 7.0, 7.11, 7.2, 7.8, 7.9, 8.0, 8.4} in², b with fuzzy logic controller: {28, 29, 30, 31, ..., 38, 39, 40} in, and $5 \leq h \leq 10$ in. The constrained functions of g_1 and g_2 as derived by Liebman et al. [43], then, is used in by Amir and Hasegawa [42] and here.

The CS method requires 25 cuckoos and 200 iterations to reach the optimum. Table 18 presents the optimum designs of this problem and the parameters used. One can see that the performance of the CS method is very good, as compared with the results reported in [42].

3.2.11 Case XI. Parameter identification of structures

Estimation of structural parameter is the art of reconciling an a priori finite-element model (FEM) of the structure with nondestructive test data. It has a great potential for use in FEM updating. Saltenik and Sanayei [46] developed a parameter estimation example using measured strains for simultaneous estimation of the structural parameters. The parameter estimation objective function is defined as follows:

$$\text{Minimize : } \sum_{i=1}^{NMS} \left| \frac{([\varepsilon_a]_{m,i} - [\varepsilon_a]_{a,i})}{[\varepsilon]_{m,i}} \right| \tag{64}$$

where $[\varepsilon_a]_m$ is the measured strains, $[\varepsilon_a]_m = \text{number of measurements (NMS)} \times \text{number of loading states (NLS)}$, and $[\varepsilon_a]_a$ is the analytical strains.

The static FEM equation for a structural system is $[F] = [K][U]$. Thus, the analytical strains are calculated as follows:

$$[\varepsilon] = [B][K]^{-1}[F] \tag{65}$$

It is not required to measure all the strains, therefore, Eq. (65) is partitioned based on measured strain a and unmeasured strain b :

Fig. 13 Frame structure used for parameter identification example

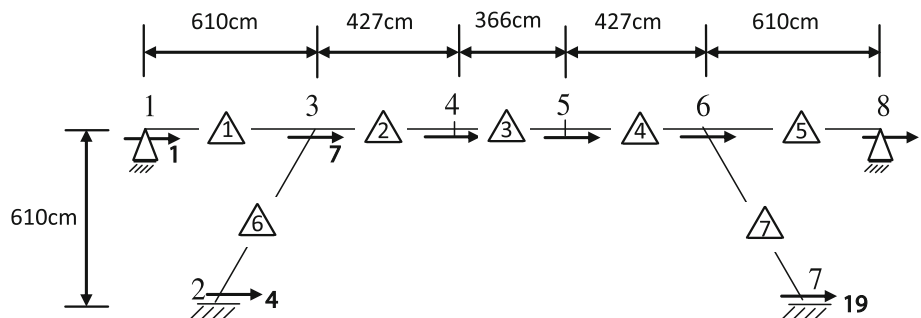


Table 19 Best solutions for the parameter identification example using CS

| Best | Mean | Worst | SD | No. cuckoos | No. evals. |
|------------------------|-------------------------|------------------------|------------------------|-------------|------------|
| 5.39×10^{-14} | 4.970×10^{-10} | 2.289×10^{-4} | 2.302×10^{-5} | 50 | 25,000 |

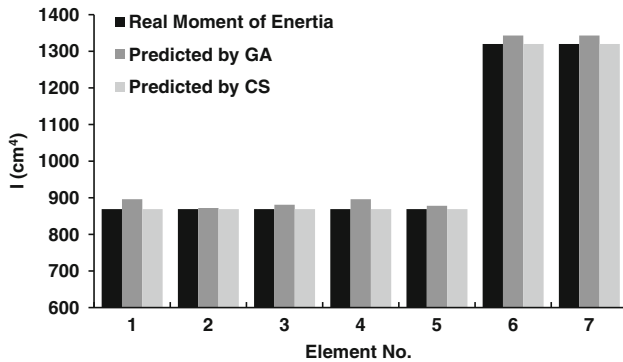


Fig. 14 Parameter identification results using GA and CS

$$\begin{bmatrix} \varepsilon_a \\ \varepsilon_b \end{bmatrix} = \begin{bmatrix} B_a \\ B_b \end{bmatrix} [K]^{-1} [F] \tag{66}$$

Since there is no need for unmeasured strains $[\varepsilon_b]$ is eliminated as

$$[\varepsilon_a] = [B_a][K]^{-1}[F] \tag{67}$$

In this work, the case study is a frame example presented by Saltenik and Sanayei [46] (see Fig. 13). The identified parameter in this example is moment of inertia (I) for each member.

A 445-N load is applied to degrees of freedom of 2, 5, 8, and 11, and each load set is composed of only one force. Strains are measured on 3, 6, and 7 for each load set. The cross-sectional areas are, respectively, 484 cm^2 and 968 cm^2 for the horizontal and inclined members. The Elastic modulus is 206.8 GPa for all elements. The optimal solution is obtained at $I \text{ (cm}^4\text{)} = [869, 869, 869, 869, 869, 1,320, 1,320]$ with corresponding function value equal to $f^*(x) = 0.00000$. The statistical results for this case study provided by CS are presented in Table 19.

The analytical algorithm proposed by Saltenik and Sanayei [46] is not applicable to this problem since a singularity. Arjmandi [47] solved this problem using GA. A comparison of the results obtained by GA and CS with the measured values is illustrated in Fig. 14. The results show that CS has found the global optimum and identified all the parameters without any error.

3.2.12 Case XII. Pressure vessel design

A cylindrical pressure vessel capped at both ends by hemispherical heads is presented in Fig. 15. This compressed air

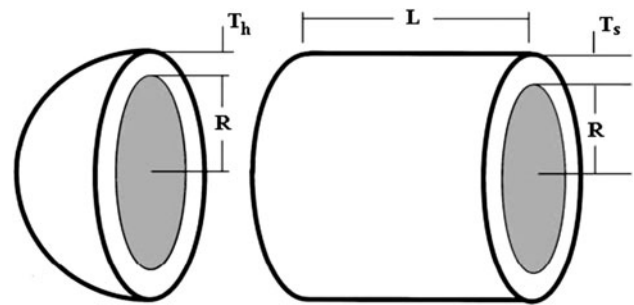


Fig. 15 Pressure vessel

tank has a working pressure of 3000 psi and a minimum volume of 750 ft^3 , and is designed according to the ASME boiler and pressure vessel code. The total cost, including a combination of single 60° welding cost, material and forming cost, is to be minimized. The involved variables are the thickness (T_s), thickness of the head (T_h), the inner radius (R), and the length of the cylindrical section of the vessel (L). The thicknesses of the variables are discrete values which are integer multiples of 0.0625 inch.

Then, the optimization problem can be expressed as follows:

$$\begin{aligned} \text{Minimize : } f(T_s, T_h, R, L) &= 0.6224T_sRL + 1.7781T_hR^2 \\ &+ 3.1661T_s^2L + 19.84T_h^2L \end{aligned} \tag{68}$$

Subject to:

$$g_1 = -T_s + 0.0193R \leq 0 \tag{69}$$

$$g_2 = -T_h + 0.0095R \leq 0 \tag{70}$$

$$g_3 = -\pi R^2L - \frac{4}{3}\pi R^3 + 1,296,000 \leq 0 \tag{71}$$

$$g_4 = L - 240 \leq 0 \tag{72}$$

where $1 \times 0.0625 \leq T_s, T_h \leq 99 \times 0.0625$, and $10 \leq R, L \leq 200$. The minimum cost and the statistical values of the best solution obtained by CS are reported in Table 20. It is notable that the CS method requires 25 cuckoos and 15,000 searches to reach the optimum.

This problem has previously been solved by many researchers as a benchmark structural optimization problem. The comparisons of results for several approaches for the pressure vessel design example are shown in Tables 21, 22. As it is seen, the results obtained by the CS algorithm are better than the available solutions in nearly all cases.

Table 20 Statistical results of the best pressure vessel model obtained by CS

| T_s | T_h | R | L | Best | Ave. time (s) |
|--------|--------|------------|-------------|--------------|---------------|
| 0.8125 | 0.4375 | 42.0984456 | 176.6365958 | 6059.7143348 | 2.50472 |

Table 21 Statistical results of the pressure vessel design example by different model

| Author(s) | Best | Mean | Worst | SD |
|-------------------------------|----------|----------|----------|----------|
| Lee and Geem [14] | 7198.433 | NA | NA | NA |
| Coello [18] | 6288.745 | 6293.843 | 6308.15 | 7.4133 |
| Sandgren [35] | 7980.894 | NA | NA | NA |
| Kannan and Kramer [36] | 7198.042 | NA | NA | NA |
| Akhtar [41] | 6171 | 6335.05 | 6453.65 | NA |
| Yun [45] | 7198.424 | NA | NA | NA |
| Tsai et al. [48] | 7079.037 | NA | NA | NA |
| Cao and Wu [49] | 7108.616 | NA | NA | NA |
| Deb [50] | 6410.381 | NA | NA | NA |
| Coello [51] | 6228.744 | NA | NA | NA |
| Sandgren [52] | 8129.104 | NA | NA | NA |
| Coelho [53] | 6059.714 | NA | NA | NA |
| Zhang and Wang [54] | 7197.7 | NA | NA | NA |
| Coello and Cortés [55] | 6061.123 | 6734.085 | 7368.06 | 457.9959 |
| Homaifar et al. | 6295.11 | 8098.03 | 9528.07 | 831.69 |
| Joines and Houck [56] | 6273.28 | 8092.87 | 10382.1 | 1017.99 |
| Michalewicz and Attia [57] | 6572.62 | 8164.56 | 9580.51 | 789.65 |
| Hadj-Alouane and Bean [58] | 6303.5 | 8065.66 | 10569.65 | 821.3 |
| Fu et al. [59] | 8048.6 | NA | NA | NA |
| Li and Chou [60] | 7127.3 | NA | NA | NA |
| Cai and Thierauf [61] | 7006.931 | NA | NA | NA |
| He et al. [62] | 6059.714 | 6289.929 | NA | 305.78 |
| Coello and Montes [63] | 6059.946 | 6177.253 | 6469.322 | 130.9297 |
| Cao and Wu [64] | 7108.616 | NA | NA | NA |
| Hu et al. [65] | 6059.131 | NA | NA | NA |
| Huang et al. [66] | 6059.734 | 6085.23 | 6371.046 | 43.013 |
| Zahara and Kao [67] | 5930.314 | 5946.79 | 5960.056 | 9.1614 |
| He and Wang [68] | 6061.078 | 6147.133 | 6363.804 | 86.4545 |
| Litinetski and Abramzon [69] | 7197.7 | NA | NA | NA |
| Wu and Chow [70] | 7207.494 | NA | NA | NA |
| Lee and Geem [71] | 7198.433 | NA | NA | NA |
| Cagnina et al. [72] | 6059.714 | NA | NA | NA |
| Coello [55] | 6734.085 | NA | NA | 457.9959 |
| Coello [73] | 6263.793 | NA | NA | 97.9445 |
| Coello [74] | 6177.253 | NA | NA | 130.9297 |
| Ray and Liew [75] | 6171 | 6335.05 | NA | NA |
| Montes et al. [76] | 6059.702 | 6059.702 | NA | 0 |
| Parsopoulos and Vrahatis [77] | 6544.27 | 9032.55 | 11638.2 | 995.573 |
| Shih and Lai [78] | 7462.1 | NA | NA | NA |
| Li and Chang [79] | 7127.3 | NA | NA | NA |
| Kaveh and Talatahari [80] | 6059.73 | 6081.78 | 6150.13 | 67.2418 |
| Present study | 6059.714 | 6447.736 | 6495.347 | 502.693 |

Table 22 Statistical results of the car side impact design example by different methods

| Method | PSO | DE | GA | FA | CS |
|-----------------|-----------|-----------|-----------|-----------|-----------|
| Best objective | 22.84474 | 22.84298 | 22.85653 | 22.84298 | 22.84294 |
| x_1 | 0.50000 | 0.50000 | 0.50005 | 0.50000 | 0.50000 |
| x_2 | 1.11670 | 1.11670 | 1.28017 | 1.36000 | 1.11643 |
| x_3 | 0.50000 | 0.50000 | 0.50001 | 0.50000 | 0.50000 |
| x_4 | 1.30208 | 1.30208 | 1.03302 | 1.20200 | 1.30208 |
| x_5 | 0.50000 | 0.50000 | 0.50001 | 0.50000 | 0.50000 |
| x_6 | 1.50000 | 1.50000 | 0.50000 | 1.12000 | 1.50000 |
| x_7 | 0.50000 | 0.50000 | 0.50000 | 0.50000 | 0.50000 |
| x_8 | 0.34500 | 0.34500 | 0.34994 | 0.34500 | 0.34500 |
| x_9 | 0.19200 | 0.19200 | 0.19200 | 0.19200 | 0.19200 |
| x_{10} | -19.54935 | -19.54935 | 10.3119 | 8.87307 | -19.54935 |
| x_{11} | -0.00431 | -0.00431 | 0.00167 | -18.99808 | -0.00431 |
| Mean objective | 22.89429 | 23.22828 | 23.51585 | 22.89376 | 22.85858 |
| Worst objective | 23.21354 | 24.12606 | 26.240578 | 24.06623 | 23.25998 |
| SD | 0.15017 | 0.34451 | 0.66555 | 0.16667 | 0.07612 |

**Fig. 16** Car side impact model [82]

Note that the best objective value obtained by Zahara and Kao [67] is not feasible since the first and second constraints are violated. The results obtained by Hu et al. [65] and Mezura-Montes et al. [76] also violate the first and third constraints, respectively. Thus, the best feasible solution is $f = 6059.714$ which is provided by CS and also obtained by Coelho [53], He et al. [62] and Cagnina et al. [72].

3.2.13 Case XIII. Car side impact design

Design of car side impact is also used as a benchmark problem of the proposed FA. The FEM model of this problem is illustrated in Fig. 16. On the foundation of European Enhanced Vehicle-Safety Committee (EEVC) procedures, a car is exposed to a side-impact. Here, we

want to minimize the weight using nine influence parameters including, thicknesses of B-Pillar inner, B-Pillar reinforcement, floor side inner, cross members, door beam, door beltline reinforcement and roof rail (x_1 – x_7), materials of B-Pillar inner and floor side inner (x_8 and x_9) and barrier height, and hitting position (x_{10} and x_{11}). The car side problem is stated by Gu et al. [81] and as an optimization problem it can be formulated as follows:

$$\text{Minimize : } f(x) = \text{Weight}; \quad (73)$$

Subject to:

$$g_1(x) = F_a(\text{load in abdomen}) \leq 1 \text{ kN}; \quad (74)$$

$$g_2(x) = V \times Cu (\text{dummy upper chest}) \leq 0 : 32 \text{ m/s}; \quad (75)$$

$$g_3(x) = V \times Cm (\text{dummy middle chest}) \leq 0 : 32 \text{ m/s}; \quad (76)$$

$$g_4(x) = V \times Cl (\text{dummy lower chest}) \leq 0 : 32 \text{ m/s}; \quad (77)$$

$$g_5(x) = \Delta_{ur}(\text{upper rib deflection}) \leq 32 \text{ mm}; \quad (78)$$

$$g_6(x) = \Delta_{mr}(\text{middle rib deflection}) \leq 32 \text{ mm}; \quad (79)$$

$$g_7(x) = \Delta_{lr}(\text{lower rib deflection}) \leq 32 \text{ mm}; \quad (80)$$

$$g_8(x) = F(\text{Pubic force})_p \leq 4 \text{ kN}; \quad (81)$$

$$g_9(x) = V_{MBP}(\text{Velocity of V-Pillar at middle point}) \leq 9 : 9 \text{ mm/ms}; \quad (82)$$

$$g_{10}(x) = V_{FD}(\text{Velocity of front door at V-Pillar}) \leq 15 : 7 \text{ mm/ms}; \quad (83)$$

with simple bounds

$$0.5 \leq x_1, x_3, x_4 \leq 1.5; 0.45 \leq x_2 \leq 1.35;$$

$$0.875 \leq x_5 \leq 2.625; 0.4 \leq x_6, x_7 \leq 1.2;$$

$$x_8, x_9 \in \{0.192, 0.345\}; 0.5 \leq x_{10}, x_{11} \leq 1.5;$$

For solving this problem CS run with 20 fireflies and 1000 iterations. This case study has also been solved using well-known GA, PSO, DE, and firefly algorithm (FA) methods by Gandomi et al. [83] and the results are used to compare with the CS method. Table 22 shows the statistical results for the car side impact design problem using the proposed CS method and other well-known methods after 20,000 searches. As it can be seen from Table 22, in comparison with other heuristic algorithms, the proposed algorithm is far better than GA and slightly better PSO, DE, and FA with the same number of evaluations and runs.

4 Discussions and conclusions

In the present study, the new CS algorithm in combination with Lévy flights is employed for solving structural optimization problems. The CS algorithm has been validated first using several benchmark structural engineering problems and found to be very efficient. The extensive comparative study conducted reveals that CS performs superior to different existing algorithms. This is partly due to the fact that there are fewer parameters to be fine-tuned in CS than in other algorithms such as GA and PSO. In fact, apart from the population size n , there is essentially one parameter P_a . Inspecting the CS algorithm carefully, there are essentially three components: selection of the best, exploitation by local random walk, and exploration by randomization via Lévy flights globally. The selection of the best by keeping the best nests or solutions is equivalent to some form of elitism commonly used in GAs, which ensures the best solution is passed onto the next iteration and there is no risk that the best solutions are cast out of the population. The exploitation around the best solutions is performed by using a local random walk:

$$x^{t+1} = x^t + \alpha \varepsilon_t \quad (84)$$

If ε_t obeys a Gaussian distribution, this becomes a standard random walk indeed. This is equivalent to the crucial step in pitch adjustment in harmony search. If ε_t is drawn from a Lévy distribution, the step of move is larger and could be potentially more efficient. However, if the step is too large, there is risk that the move is too far away. Fortunately, the elitism by keeping the best solutions makes sure that the exploitation moves are within the neighborhood of the best solutions locally. Another way of improving this local search slightly is to replace x^t by the current best, so that the intensive local exploitation step is focused on the local search around the current best found

so far, and the radius or the step of such local search can be chosen appropriately to reflect the scalings of the problem.

On the other hand, in order to sample the whole search space effectively so that new solutions to be generated are diverse enough, a simple way of achieving this is to ensure the generated search locations/solutions are uniformly distributed in the search space. However, this near-uniform approach is not necessarily efficient. A better and more efficient way to carry out the exploration step is to use Lévy flights. In contrast, most metaheuristic algorithms use either uniform distributions or Gaussian to generate new explorative moves [3, 4]. If the search space is large, Lévy flights are usually more efficient. A good combination of the above three components can thus lead to an efficient algorithm such as CS.

Furthermore, from our simulations by varying the algorithm-dependent parameters, we observed that the convergence rate is insensitive to the algorithm-dependent parameters such as P_a . This can be viewed from the balanced combination of the selection of the best and efficient exploration of possible new solutions. The elitism-style selection makes sure that a fraction of the best solutions in the population are guaranteed to remain the population, while the discovery of alien eggs with non-zero probability makes sure that some of the worse solutions are discarded and replaced by new ones. Such insensitivity also means that there is no need to fine tune these parameters for a specific problem. Subsequently, CS is more generic and robust for many optimization problems, compared with other metaheuristic algorithms.

In principle, this potentially powerful optimization strategy can easily be extended in the similar fashion as extending population-based algorithms such as PSO and genetic algorithms to study multi-objective optimization applications with various constraints, including NP-hard problems. Further studies can focus on the sensitivity and parameter studies and their possible relationships with the convergence rate of the algorithm. In addition, hybridization with other popular algorithms such as PSO will also be potentially fruitful. Another possible and yet easy extension is to formulate a discrete version of Cuckoo Search so as to solve combinatorial optimization problems such as traveling salesman problem and job scheduling.

From an analytical point view, as for most metaheuristic algorithms, mathematical analysis of the algorithm structures is highly needed. At the moment, no such framework exists for analyzing metaheuristics in general. The analysis of trajectory-based algorithms such as simulated annealing can be dealt with in the framework of Markov chains Monte Carlo (MCMC). In some way, CS can be viewed as an evolutionary system with multiple interacting Markov chains selected and biased towards global optimality. Any progress in this area will potentially provide new insight

into the understanding of how and why metaheuristic algorithms work. It will also help us to design more efficient, often hybrid, algorithms to solve a wider class of tough optimization problems.

Acknowledgments The authors gratefully acknowledge the work and help of Engineer Parvin Arjmandi (Tafresh University).

References

1. Yang X-S (2008) Nature-inspired metaheuristic algorithms. Luniver Press, UK
2. Talbi E (2009) Metaheuristics: from design to implementation. John Wiley & Sons, Hoboken
3. Yang X-S (2009) Harmony search as a metaheuristic algorithm. In: Geem ZW (ed) Music-inspired harmony search: theory and applications. Springer, Berlin, pp 1–14
4. Yang X-S, Deb S (2009) Cuckoo search via Lévy flights. In: Proceedings of World Congress on Nature & Biologically Inspired Computing. IEEE Publications, USA, pp 210–214
5. Payne RB, Sorenson MD, Klitz K (2005) The Cuckoos. Oxford University Press, New York
6. Brown C, Liebovitch LS, Glendon R (2007) Lévy flights in Dobe Ju/hoansi foraging patterns. *Human Ecol* 35:129–138
7. Pavlyukevich I (2007) Lévy flights, non-local search and simulated annealing. *J Comput Phys* 226:1830–1844
8. Reynolds AM, Frye MA (2007) Free-flight odor tracking in *Drosophila* is consistent with an optimal intermittent scale-free search. *PLoS One* 2:e354
9. Barthelemy P, Bertolotti J, Wiersma DS (2008) A Lévy flight for light. *Nature* 453:495–498
10. Shlesinger MF (2006) Search research. *Nature* 443:281–282
11. Himmelblau DM (1972) Applied nonlinear programming. McGraw-Hill, New York
12. Gen M, Cheng R (1997) Genetic algorithms & engineering design. Wiley, New York
13. Homaifar A, Lai SH-V, Qi X (1994) Constrained optimization via genetic algorithms. *Simulation* 62(4):242–254
14. Lee KS, Geem ZW (2004) A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice. *Comput Methods Appl Mech Eng* 194:3902–3933
15. Fesanghary M, Mahdavi M, Minary-Jolandan M, Alizadeh Y (2008) Hybridizing harmony search algorithm with sequential quadratic programming for engineering optimization problems. *Comput Methods Appl Mech Eng* 197:3080–3091
16. He S, Prempan E, Wu QH (2004) An improved particle swarm optimizer for mechanical design optimization problems. *Eng Optimiz* 36(5):585–605
17. Shi Y, Eberhart RC (1998) A modified particle swarm optimizer, in: Proceedings of the International Congress on Evolutionary Computation, IEEE Service Center, Piscataway, New Jersey
18. Coello CAC (2000) Use of a self-adaptive penalty approach for engineering optimization problems. *Comput Ind* 41(2):113–127
19. Omran MGH, Salman A (2009) Constrained optimization using CODEQ. *Chaos Soliton Fract* 42:662–668
20. Majid KI (1974) Optimum design of structures. Newnes-Butterworth, London
21. Li J-P, Balazs ME, Parks GT (2007) Engineering design optimization using species-conserving genetic algorithms. *Eng Optimiz* 39(2):147–161
22. Gold S, Krishnamurty S (1997) Trade-offs in Robust Engineering Design, Proceedings of the 1997 ASME Design Engineering Technical Conferences, DETC97/DAC3757, September 14–17, Sacramento, California
23. Wang GG (2003) Adaptive response surface method using inherited latin hypercube design points. *Trans ASME* 125:210–220
24. Vanderplaats GN (1995) DOT (Design Optimization Tools) Users Manual, Version 4.20, VR&D
25. Kim P, Lee J (2009) An integrated method of particle swarm optimization and differential evolution. *J Mech Sci Technol* 23:426–434
26. Kvalie D (1967) Optimization of plane elastic grillages. PhD Thesis, Norges Teknisk Naturvitenskapelige Universitet, Norway
27. Ravindran A, Ragsdell KM, Reklaitis GV (2006) Engineering optimization: methods and applications, 2nd edn. John Wiley & Sons, NJ
28. Fleury C, Braibant V (1986) “Structural optimization: a new dual method using mixed variables”. *Int J Numer Meth Eng* 23:409–428
29. Chickermane H, Gea HC (1996) Structural optimization using a new local approximation method. *Int J Numer Method Eng* 39:829–846
30. Rao SS (1996) Engineering optimization: theory and practice, 3rd edn. John Wiley & Sons, Chichester
31. Hsu Y-L, Liu T-C (2007) Developing a fuzzy proportional-derivative controller optimization engine for engineering design optimization problems. *Eng Optimiz* 39(6):679–700
32. Nowcki H (1974) Optimization in pre-contract ship design. In: Y Fujita, K Lind, TJ Williams (eds) Computer applications in the automation of shipyard operation and ship design, vol 2. North-Holland, Elsevier, New York, pp 327–338
33. Ray T, Saini P (2001) Engineering design optimization using a swarm with an intelligent information sharing among individuals. *Eng Optimiz* 33(6):735–748
34. Tsai J (2005) Global optimization of nonlinear fractional programming problems in engineering design. *Eng Optimiz* 37(4):399–409
35. Sandgren E (1990) Nonlinear integer and discrete programming in mechanical design optimization. *J Mech Design* 112(2):223–229
36. Kannan BK, Kramer SN (1994) An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design. *J Mech Des Trans* 116:318–320
37. Deb K, Goyal M (1996) A combined genetic adaptive search (geneas) for engineering design. *Comput Sci Informat* 26(4):30–45
38. Gandomi AH, Yang XS (2011) Benchmark problems in structural optimization. chapter 12 in computational optimization, methods and algorithms, (S Koziel, XS Yang eds) Springer-Verlag, Berlin, 267–291
39. Montes EM, Coello CAC, Ricardo L (2003) Engineering optimization using a simple evolutionary algorithm, in 15th Intl. Conf. on Tools with Art. Intelligence—ICTAI’2003, CA, USA, pp 149–156
40. Kuang JK, Rao SS, Chen L (1998) Taguchi-aided search method for design optimization of engineering systems. *Eng Optimiz* 30:1–23
41. Akhtar S, Tai K, Ray T (2002) A socio-behavioural simulation model for engineering design optimization. *Eng Optimiz* 34(4):341–354
42. Amir HM, Hasegawa T (1989) Nonlinear mixed-discrete structural optimization. *J Struct Eng* 115(3):626–645
43. Liebman JS, Khachaturian N, Chanaratna V (1981) Discrete structural optimization. *J Struct Div* 107(ST11):2177–2197
44. Shih CJ, Yang YC (2002) Generalized Hopfield network based structural optimization using sequential unconstrained

- minimization technique with additional penalty strategy. *Adv Eng Softw* 33:721–729
45. Yun YS (2005) Study on Adaptive Hybrid Genetic Algorithm and Its Applications to Engineering Design Problems. Waseda University, MSc Thesis
 46. Sanayei M, Saletnik MJ (1996) Parameter estimation of structures from static strain measurements. I: Formulation. *J Struct Eng* 122(5):555–562
 47. Arjmandi P (2010) Damage Detection of continuous steel beams using static data. MSc Thesis, Tafresh University
 48. Tsai J-F, Li H-L, Hu N-Z (2002) Global optimization for signomial discrete programming problems in engineering design. *Eng Optimiz* 34(6):613–622
 49. Cao YJ, Wu QH (1997) Mechanical design optimization by mixed variable evolutionary programming. In: Proceedings of the 1997 International Conference on Evolutionary Computation, Indianapolis, pp 443–446
 50. Deb K, Gene AS (1997) A robust optimal design technique for mechanical component design. *Evolutionary algorithms in engineering applications*. Springer-Verlag, Berlin, pp 497–514
 51. Coello CAC (1999) Self-adaptive penalties for GA based optimization. *Proc Congr Evol Comput* 1:573–580
 52. Sandgren E (1988) Nonlinear integer and discrete programming in mechanical design. Proceedings of the ASME Design Technology Conference, Kissimmee, FL, pp 95–105
 53. dos Santos Coelho L (2010) Gaussian quantum-behaved particle swarm optimization approaches for constrained engineering design problems. *Expert Syst Appl* 37(2):1676–1683
 54. Zhang C, Wang HP (1993) Mixed-discrete nonlinear optimization with simulated annealing. *Eng Optimiz* 17(3):263–280
 55. Coello CAC, Cortés NC (2004) Hybridizing a genetic algorithm with an artificial immune system for global optimization. *Eng Optimiz* 36(5):607–634
 56. Joines J, Houck C (1994) On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GAs. In: Proceedings of the first IEEE Conference on Evolutionary Computation, Orlando, Florida. D. Fogel (ed.). IEEE Press, pp 579–584
 57. Michalewicz Z, Attia N (1994) Evolutionary optimization of constrained problems. Proceedings of the 3rd Annual Conference on Evolutionary Programming, World Scientific, pp 98–108
 58. Hadj-Alouane AB, Bean JC (1997) A genetic algorithm for the multiple-choice integer program. *Oper Res* 45:92–101
 59. Fu J, Fenton RG, Cleghorn WL (1991) A mixed integer-discrete-continuous programming method and its application to engineering design optimization. *Eng Optimiz* 17:263–280
 60. Li H-L, Chou C-T (1994) A global approach for nonlinear mixed discrete programming in design optimization. *Eng Optimiz* 22:109–122
 61. Cai J, Thierauf G (1997) Evolution strategies in engineering optimization. *Eng Optimiz* 29(1):177–199
 62. He S, Prempan E, Wu QH (2004) An improved particle swarm optimizer for mechanical design optimization problems. *Eng Optimiz* 36(5):585–605
 63. Coello CAC, Mezura Montes E (2001) Use of dominance-based tournament selection to handle constraints in genetic algorithms. In: *Intelligent Engineering Systems through Artificial Neural Networks (ANNIE2001)*, vol 11. ASME Press, St. Louis, pp 177–182
 64. Cao YJ, Wu QH (1999) A mixed variable evolutionary programming for optimization of mechanical design. *Int J Eng Intel Syst Elect Eng Commun* 7(2):77–82
 65. Hu X, Eberhart RC, Shi Y (2003) Engineering optimization with particle swarm. In: *Proc. 2003 IEEE Swarm Intelligence Symposium*. 53–57
 66. Huang FZ, Wang L, He Q (2007) An effective co-evolutionary differential evolution for constrained optimization. *Appl Math Comput* 186:340–356
 67. Zahara E, Kao YT (2009) Hybrid Nelder–Mead simplex search and particle swarm optimization for constrained engineering design problems. *Expert Syst Appl* 36:3880–3886
 68. He Q, Wang L (2006) An effective co-evolutionary particle swarm optimization for engineering optimization problems. *Eng Appl Artif Intel* 20:89–99
 69. Litinetski VV, Abramzon BM (1998) Mars—a multistart adaptive random search method for global constrained optimization in engineering applications. *Eng Optimiz* 30(2):125–154
 70. Wu SJ, Chow PT (1995) Genetic algorithms for nonlinear mixed discrete-integer optimization problems via meta-genetic parameter optimization. *Eng Opti* 24:137–159
 71. Seok K, Geem ZW (2005) A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice. *Comput Methods Appl Mech Engrg* 194:3902–3933
 72. Cagnina LC, Esquivel SC, Coello CAC (2008) Solving engineering optimization problems with the simple constrained particle swarm optimizer. *Inform* 32:319–326
 73. Coello CAC (2000) Constraint-handling using an evolutionary multiobjective optimization technique. *Civil Engrg Environ Syst* 17:319–346
 74. Coello CAC (2002) Constraint-handling in genetic algorithms through the use of dominance-based tournament selection. *Adv Engrg Inform* 16:193–203
 75. Ray T, Liew K (2003) Society and civilization: An optimization algorithm based on the simulation of social behavior. *IEEE Trans Evol Comput* 7(4):386–396
 76. Montes EM, Coello CAC, Velázquez-Reyes J, Muñoz-Dávila L (2007) Multiple trial vectors in differential evolution for engineering design. *Eng Optimiz* 39(5):567–589
 77. Parsopoulos KE, Vrahatis MN (2005) Unified particle swarm optimization for solving constrained engineering optimization problems. In: *Lecture Notes in Computer Science (LNCS)*, vol 3612, pp 582–591
 78. Shih CJ, Lai TK (1995) Mixed-discrete fuzzy programming for nonlinear engineering optimization. *Eng Optimiz* 23(3):187–199
 79. Li HL, Chang CT (1998) An approximate approach of global optimization for polynomial programming problems. *Eur J Oper Res* 107(3):625–632
 80. Kaveh A, Talatahari S (2010) An improved ant colony optimization for constrained engineering design problems. *Eng Comput* 27(1):155–182
 81. Gu L, Yang RJ, Cho CH, Makowski M, Faruque M, Li Y (2001) Optimization and robustness for crashworthiness. *Int J Vehicle Design* 26(4):348–360
 82. Youn BD, Choi KK, Yang R-J, Gu L (2004) Reliability-based design optimization for crashworthiness of vehicle side impact. *Struct Multidisc Optim* 26:272–283
 83. Gandomi AH, Yang XS, Alavi AH, Mixed variable structural optimization using firefly algorithm. *Computers & Structures (in press)*