Bat algorithm: a novel approach for global engineering optimization

Xin-She Yang

Mathematics and Scientific Computing, National Physical Laboratory, Teddington, UK, and

Amir Hossein Gandomi

Department of Civil Engineering, Tafresh University, Tafresh, Iran

Abstract

Purpose – Nature-inspired algorithms are among the most powerful algorithms for optimization. The purpose of this paper is to introduce a new nature-inspired metaheuristic optimization algorithm, called bat algorithm (BA), for solving engineering optimization tasks.

Design/methodology/approach – The proposed BA is based on the echolocation behavior of bats. After a detailed formulation and explanation of its implementation, BA is verified using eight nonlinear engineering optimization problems reported in the specialized literature.

Findings – BA has been carefully implemented and carried out optimization for eight well-known optimization tasks; then a comparison has been made between the proposed algorithm and other existing algorithms.

Originality/value – The optimal solutions obtained by the proposed algorithm are better than the best solutions obtained by the existing methods. The unique search features used in BA are analyzed, and their implications for future research are also discussed in detail.

Keywords Iterative methods, Programming and algorithm theory, Optimization techniques, Bat algorithm, Engineering optimization, Metaheuristic algorithm

Paper type Research paper

1. Introduction

Design optimization forms an important part of any design problem in engineering and industry. Structural design optimization focuses on finding the optimal and practical solutions to complex structural design problems under dynamic complex loading pattern with complex nonlinear constraints. These constraints often involve thousands of and even millions of members with stringent limitations on stress, geometry as well as loading and service requirements. The aim is not only to minimize the cost and materials usage, but also to maximize their performance and lifetime service. All these designs are of scientific and practical importance (Deb, 1995; Yang, 2010). However, most structural design optimization problems are highly nonlinear and multimodal with noise, and thus they are often NP-hard. Finding the right and practically efficient algorithms are usually difficult, if not impossible. In realistic, the choice of an algorithm requires extensive experience and knowledge of the problem of interest. Even so, there is no guarantee that an optimal or even suboptimal solution can be found.

Amir Hossein Gandomi is now based in the Department of Civil Engineering, The University of Akron, Akron, Ohio, USA.

Metaheuristic algorithms including evolutionary and swarm intelligence algorithms are now becoming powerful methods for solving many tough problems (Gandomi and Alavi, 2011) and especially real-world engineering problems (Gandomi *et al.*, 2011; Alavi and Gandomi, 2011). The vast majority of heuristic and metaheuristic algorithms have been derived from the behavior of biological systems and/or physical systems in nature. For example, particle swarm optimization was developed based on the swarm behavior of birds and fish (Kennedy and Eberhart, 1995) or charged system search inspired from physical processes (Kaveh and Talatahari, 2010). New algorithms are also emerging recently, including harmony search (HS) and the firefly algorithm. The former was inspired by the improvising process of composing a piece of music (Geem *et al.*, 2001), while the latter was formulated based on the flashing behavior of fireflies (Yang, 2008). Each of these algorithms has certain advantages and disadvantages. For example, simulating annealing (Kirkpatrick *et al.*, 1983) can almost guarantee to find the optimal solution if the cooling process is slow enough and the simulation is running long enough; however, the fine adjustment in parameters does affect the convergence rate of the optimization process. A natural question is whether it is possible to combine major advantages of these algorithms and try to develop a potentially better algorithm?

This paper is such an attempt to address this issue. In this paper, we intend to propose a new metaheuristic method, namely, the bat algorithm (BA), based on the echolocation behavior of bats, and preliminary studies show that this algorithm is very promising (Yang, 2010). The capability of echolocation of microbats is fascinating as these bats can find their prey and discriminate different types of insects even in complete darkness. We will first formulate the BA by idealizing the echolocation behavior of bats. We then describe how it works and make comparison with other existing algorithms. Finally, we will discuss some implications for further studies.

2. Echolocation of microbats

Bats are fascinating animals. They are the only mammals with wings and they also have advanced capability of echolocation. It is estimated that there are about 1,000 different species which account for up to about one-fifth of all mammal species (Altringham, 1996). Their size ranges from the tiny bumblebee bat (of about 1.5-2 g) to the giant bats with wingspan of about 2 m and weight up to about 1 kg. Microbats typically have forearm length of about 2.2-11 cm. Most bats uses echolocation to a certain degree; among all the species, microbats are a famous example as microbats use echolocation extensively while megabats do not (Richardson, 2008).

Most microbats are insectivores. Microbats use a type of sonar, called, echolocation, to detect prey, avoid obstacles, and locate their roosting crevices in the dark. These bats emit a very loud sound pulse and listen for the echo that bounces back from the surrounding objects. Their pulses vary in properties and can be correlated with their hunting strategies, depending on the species. Most bats use short, frequency-modulated signals to sweep through about an octave, while others more often use constant-frequency signals for echolocation. Their signal bandwidth varies depends on the species, and often increased by using more harmonics.

Though each pulse only lasts a few thousandths of a second (up to about 8-10 ms); however, it has a constant frequency which is usually in the region of 25-150 kHz. The typical range of frequencies for most bat species are in the region between 25 and 100 kHz, though some species can emit higher frequencies up to 150 kHz. Each ultrasonic

burst may last typically 5-20 ms, and microbats emit about 10-20 such sound bursts every second. When hunting for prey, the rate of pulse emission can be sped up to about 200 pulses per second when they fly near their prey. Such short sound bursts imply the fantastic ability of the signal processing power of bats. In fact, studies show the integration time of the bat ear is typically about 300-400 μs.

As the speed of sound in air is typically $v = 340$ m/s, the wavelength λ of the ultrasonic sound bursts with a constant frequency f is given by $\lambda = v/f$, which is in the range of 2-14 mm for the typical frequency range from 25 to 150 kHz. Such wavelengths are in the same order of their prey sizes.

Amazingly, the emitted pulse could be as loud as 110 dB, and, fortunately, they are in the ultrasonic region. The loudness also varies from the loudest when searching for prey and to a quieter base when homing towards the prey. The travelling range of such short pulses is typically a few meters, depending on the actual frequencies (Richardson, 2008). Microbats can manage to avoid obstacles as small as thin human hairs.

Studies show that microbats use the time delay from the emission and detection of the echo, the time difference between their two ears, and the loudness variations of the echoes to build up three dimensional scenario of the surrounding. They can detect the distance and orientation of the target, the type of prey, and even the moving speed of the prey such as small insects. Indeed, studies suggested that bats seem to be able to discriminate targets by the variations of the Doppler effect induced by the wing-flutter rates of the target insects (Altringham, 1996).

Obviously, some bats have good eyesight, and most bats also have very sensitive smell sense. In reality, they will use all the senses as a combination to maximize the efficient detection of prey and smooth navigation. However, here we are only interested in the echolocation and the associated behavior.

Such echolocation behavior of microbats can be formulated in such a way that it can be associated with the objective function to be optimized, and this makes it possible to formulate new optimization algorithms. In the rest of this paper, we will first outline the basic formulation of the BA and then discuss the implementation and comparison in detail.

3. Bat algorithm

If we idealize some of the echolocation characteristics of microbats, we can develop various bat-inspired algorithms or BAs. For simplicity, we now use the following approximate or idealized rules:

- All bats use echolocation to sense distance, and they also "know" the difference between food/prey and background barriers in some magical way.
- Bats fly randomly with velocity v_i at position x_i with a fixed frequency f_{min}, varying wavelength λ and loudness A_0 to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses and adjust the rate of pulse emission r in the range of [0, 1], depending on the proximity of their target.
- Although the loudness can vary in many ways, we assume that the loudness varies from a large (positive) A_0 to a minimum constant value A_{min}.

Another obvious simplification is that no ray tracing is used in estimating the time delay and three dimensional topography. Though this might be a good feature for the

application in computational geometry, however, we will not use this feature, as it is more computationally extensive in multidimensional cases.

In addition to these simplified assumptions, we also use the following approximations, for simplicity. In general the frequency f in a range $[f_{min}, f_{max}]$ corresponds to a range of wavelengths $[\lambda_{min}, \lambda_{max}]$. For example, a frequency range of [20, 500 kHz] corresponds to a range of wavelengths from 0.7 to 17 mm.

For a given problem, we can also use any wavelength for the ease of implementation. In the actual implementation, we can adjust the range by adjusting the wavelengths (or frequencies), and the detectable range (or the largest wavelength) should be chosen such that it is comparable to the size of the domain of interest, and then toning down to smaller ranges. Furthermore, we do not necessarily have to use the wavelengths themselves; instead, we can also vary the frequency while fixing the wavelength λ. This is because λ and f are related due to the fact λf is constant. We will use this later approach in our implementation.

For simplicity, we can assume f is within $[0, f_{max}]$. We know that higher frequencies have short wavelengths and travel a shorter distance. For bats, the typical ranges are a few meters. The rate of pulse can simply be in the range of [0, 1] where 0 means no pulses at all, and 1 means the maximum rate of pulse emission.

Based on these approximations and idealization, the basic steps of the BA can be summarized as the pseudo code shown in Figure 1.

3.1 Velocity and position vectors of virtual bats

In simulations, we use virtual bats naturally. We have to define the rules how their positions x_i and velocities v_i in a d-dimensional search space are updated. The new solutions x_i^t and velocities v_i^t at time step t are given by:

$$f_i = f_{\min} + (f_{\max} - f_{\min})\beta \tag{1}$$

$$v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i \tag{2}$$

```
Bat Algorithm

Objective function f (x), x = (x₁, ...,x_d)ᵀ
 Initialize the bat population x_i (i = 1,2, ...,n) and v_i
Define pulse frequency f_i at x_i
Initialize pulse rates r_i and the loudness A_i
while (t < Max number of iterations)
   Generate new solutions by adjusting frequency,
   and updating velocities and locations/ solutions [equations (1) to (3)]
   if (rand < r_i)
     Select a solution amoung the best solutions
     Generate a local solution around the selected best solution
   end if
   Generate a new solution by flying randomly
   if (rand < A_i & f (x_i) < f (x*))
     Accept the new solutions
     Increase r_i and reduce A_i
   end if
   Rank the bats and find the current best x*
 end while
postprocess results and visualization
```

Figure 1.
Pseudo code of the BA

$$x_i^t = x_i^{t-1} + v_i^t \tag{3}$$

where $\beta \in [0, 1]$ is a random vector drawn from a uniform distribution. Here x_* is the current globalbest location (solution) which is located after comparing all the solutions among all the n bats. As the product $\lambda_i f_i$ is the velocity increment, we can use either f_i (or λ_i) to adjust the velocity change while fixing the other factor λ_i (or f_i), depending on the type of the problem of interest. In our implementation, we will use $f_{min} = 0$ and $f_{max} = 100$, depending the domain size of the problem of interest. Initially, each bat is randomly assigned a frequency that is drawn uniformly from $[f_{min}, f_{max}]$.

For the local search part, once a solution is selected among the current best solutions, a new solution for each bat is generated locally using a local random walk:

$$x_{new} = x_{old} + \varepsilon A^t \tag{4}$$

where $\varepsilon \in [-1, 1]$ is a random number, while $A^t = <A_i^t>$ is the average loudness of all the bats at this time step.

The update of the velocities and positions of bats have some similarity to the procedure in the standard particle swarm optimization (Geem *et al.*, 2001) as f_i essentially controls the pace and range of the movement of the swarming particles. To a degree, BA can be considered as a balanced combination of the standard particle swarm optimization and the intensive local search controlled by the loudness and pulse rate.

3.2 Variations of loudness and pulse emission
Furthermore, the loudness A_i and the rate r_i of pulse emission have to be updated accordingly as the iterations proceed. As the loudness usually decreases once a bat has found its prey, while the rate of pulse emission increases, the loudness can be chosen as any value of convenience. For example, we can use $A_0 = 100$ and $A_{min} = 1$. For simplicity, we can also use $A_0 = 1$ and $A_{min} = 0$, assuming $A_{min} = 0$ means that a bat has just found the prey and temporarily stop emitting any sound. Now we have:

$$A_i^{t+1} = \alpha A_i^t, \quad r_i^{t+1} = r_i^0[1 - \exp(-\gamma t)] \tag{5}$$

where α and γ are constants. In fact, α is similar to the cooling factor of a cooling schedule in the simulated annealing (Kirkpatrick *et al.*, 1983; Yang, 2008). For any $0 < \alpha < 1, 0 < \gamma$, we have:

$$A_i^t \rightarrow 0, \quad r_i^t \rightarrow r_i^0, \quad as \ t \rightarrow \infty \tag{6}$$

In the simplicity case, we can use $\alpha = \gamma$, and we have in fact used $\alpha = \gamma = 0.9$ in our simulations. The choice of parameters requires some experimenting. Initially, each bat should have different values of loudness and pulse emission rate, and this can be achieved by randomization. For example, the initial loudness A_i^0 can typically be [1, 2], while the initial emission rate r_i^0 can be around zero, or any value $r_i^0 \in [0, 1]$ if using (equation (5)). Their loudness and emission rates will be updated only if the new solutions are improved, which means that these bats are moving towards the optimal solution.

4. Non-linear engineering design tasks
Most real-world engineering optimization problems are nonlinear with complex constraints, sometimes the optimal solutions of interest do not even exist. In order to see

how BA performs, we will now test it against some well-known, tough but yet diverse, benchmark design problems. We have chosen eight case studies as:

(1) mathematical problem;

(2) Himmelblau's problem;

(3) three-bar truss design;

(4) speed reducer design;

(5) parameter identification of structures;

(6) cantilever stepped beam;

(7) heater exchanger design; and

(8) car side problem.

The reason for such choice is to provide a validation and test of the proposed BA against a diverse range of real-world engineering optimization problems. As we will see below, for most problems, the optimal solutions obtained by BA are far better than the best solutions reported in the literature. In all case studies, the statistical measures have been obtained, based on 50 independent runs.

4.1 Case 1: mathematical problem
Now let us start with a nonlinear mathematical benchmark problem. This problem has been used as a benchmark constrained optimization problem with some active inequality constraints (Chen and Vassiliadis, 2003). In this problem, N is the number of variables and it is a multiple of four ($N = 4n, n = 1, 2, 3, \ldots$). This problem has $N/2$ inequality constraints and $2N$ simple bounds or limits. The problem can be stated as follows:

$$\text{Minimize}: \ f(X) = \sum_{i=1}^{N} \sqrt{i}(x_i - 1)^2 + \left[\sum_{i=1}^{N} x_i^2 - 25\right]^2 \qquad (7)$$

Subject to:

$$g_j = x_{4(j-1)+1} + 2x_{4(j-1)+2} + 3x_{4(j-1)+3} + 4x_{4(j-1)+4} - 20 \qquad (8)$$

$$0 \leq g_j \leq 30 \ \ and \ \ j = 1, 2, \ldots, \frac{N}{4} \qquad (9)$$

with simple bounds:

$$0.5 \leq x_i \leq 10 (i = 1, 2, \ldots N).$$

For this problem, the global optimum and best known optimum for $N = 12$ and $N = 60$ obtained by BA is given in Table I. It can clearly be seen from Table I that the BA

ID	Variables no.	Constraints no.	Global optimum	Best BA result
1	12	6	256.75	256.75
2	60	30	30,945.28	30,945.28

Table I.
BA results and global optimums for the mathematical problem

successfully find the global minimum. The statistical results of the mathematical problem are also presented in Table II.

4.2 Case 2: Himmelblau's problem

Now we solve a well-known benchmark problem, namely Himmelblau's problem. This problem was originally proposed by Himmelblau's (1972) 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 simple bounds or limits. The problem can be stated as follows:

$$\text{Minimize}: \quad f(X) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40792.141 \quad (10)$$

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

$$g_1 = 85.334407 + 0.0056858x_1x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 \quad (11)$$

$$g_2 = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 - 0.0021813x_3^2 \quad (12)$$

$$g_3 = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 - 0.0019085x_3x_4 \quad (13)$$

with simple bounds:

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

The best known optimum for the Himmelblau's problem obtained by BA is given in Table III.

The problem was initially solved by Himmelblau (1972) using a generalized gradient method. Since then, this problem has also been solved using several other methods such as GA (Gen and Cheng, 1997; Homaifar *et al.*, 1994), HS algorithm (Lee and Geem, 2004; Fesanghary *et al.*, 2008), and PSO (He *et al.*, 2004; Shi and Eberhart, 1998). Table II summarizes the results obtained by BA, as well as those published in the literature. It can clearly be seen from Table IV that the result obtained by BA is better than the best feasible solution previously reported.

4.3 Case 3: a three-bar truss design

This case study considers a three-bar planar truss structure shown in Figure 2. This problem was first presented by Nowcki (1974). The volume of a statically loaded three-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. The mathematical formulation is

ID	Best	Mean	Worst	SD	No. bats	No. evals.	Ave. time (s)
1	256.752	256.753	256.758	0.0016	10	10,000	1.539
2	30,945.278	35,622.163	46,707.949	4,770.34	25	50,000	32.50

Table II.
Statistical results of the mathematical problem

X [x_1, x_2, x_3, x_4, x_5]	F_{min}	No. bats	No. evals.	Ave. time (s)
[78, 33, 29.995523554, 45, 36.77520645342]	−30,665.4922	15	15,000	2.76224

Table III.
BA results for the Himmelblau's problem

Author(s)	SD[a]	Best	$0 \leq g_1 \leq 92$	$90 \leq g_2 \leq 110$	$20 \leq g_3 \leq 25$
Himmelblau (1972)	N/A	−30,373.9490	N/A	N/A	N/A
Gen and Cheng (1997)	N/A	−30,183.5760	N/A	N/A	N/A
Homaifar et al. (1994)	N/A	−30,005.7000	91.65619	99.53690	20.02553
Lee and Geem (2004)	N/A	−30,665.5000	*92.00004*	98.84051	*19.99994*
Fesanghary et al. (2008)	N/A	−31,024.3160	*93.27834*	100.39612	20.00000
He et al. (2004)	70.0400	−30,643.9900	*93.28536*	100.40478	20.00000
Shi and Eberhart (1998)	N/A	−31,025.5610	*93.28533*[b]	100.40473	*19.99997*
Coello (2000)	73.6335	−31,020.8590	*93.28381*	100.40786	20.00191
Omran and Salman (2009)	N/A	−31,025.5560	*93.28536*	100.40478	20.00000
Present study	444.293	−30,665.4922	91.99990	98.84039	20.00001

Notes: [a]SD is standard deviation; [b]italized sets are violated sets

Table IV.
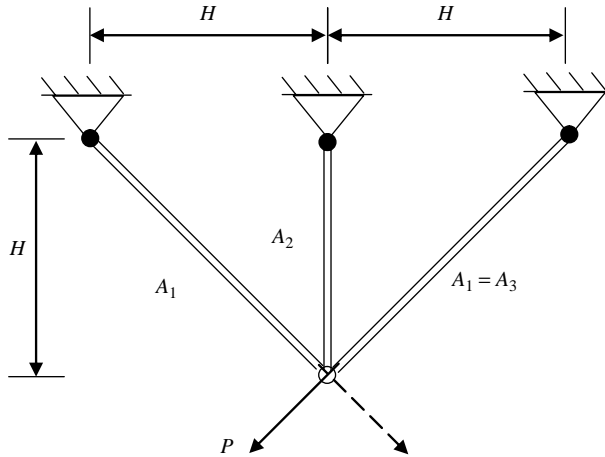Statistical results for the Himmelblau's problem

Figure 2.
Three-bar truss

given as below:

$$\text{Minimize}: \quad f(X) = (2\sqrt{2x_1} + x_2) \times l \tag{14}$$

Subject to:

$$g_1 = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2} P - \sigma \leq 0 \tag{15}$$

$$g_2 = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2} P - \sigma \leq 0 \tag{16}$$

$$g_3 = \frac{1}{x_1 + \sqrt{2}x_2} P - \sigma \leq 0 \tag{17}$$

where:

$0 \leq x_1 \leq 1 \ and \ 0 \leq x_2 \leq 1; \ l = 100\,\text{cm}, P = 2\,KN/\text{cm}^2, \ and \ \sigma = 2\,KN/\text{cm}^2$

This design problem is a nonlinear fractional programming problem. The statistical values of the best solution obtained by BA are given in Table V. The best solution by BA is $(x_1, x_2) = (0.78863, 0.40838)$ with the objective value equal to 263.896248. Table VI presents the best solutions obtained by BA and those reported by Ray and Saini (2001) and Tsai (2005). It can be seen clearly that the best objective value reported by Tsai (2005) is not feasible because the first constraint (g_1) is violated. Hence, it can be concluded that the results obtained by BA are better than those of the previous studies.

4.4 Case 4: speed reducer design

The design of a speed reducer is a more complex case study (Golinski, 1973) and it is one of the benchmark structural engineering problems (Gandomi and Yang, 2011). This problem involves seven design variables, as shown in Figure 3, with the face width b (x_1), module of teeth m (x_2), number of teeth on pinion z (x_3), length of first shaft between bearings l_1 (x_4), length of second shaft between bearings l_2 (x_5), diameter of first shaft d_1 (x_6), and diameter of second shaft d_2 (x_7). The objective is to minimize the total weight of the speed reducer. There are nine constraints, including the limits on the

Table V.
Statistical results of the best three-bar truss model

Best	Mean	Worst	SD	No. bats	No. evals.	Ave. time (s)
263.896248	263.90614	263.9024677	0.003527	10	15,000	0.72

Table VI.
Best solutions for the three-bar truss design example

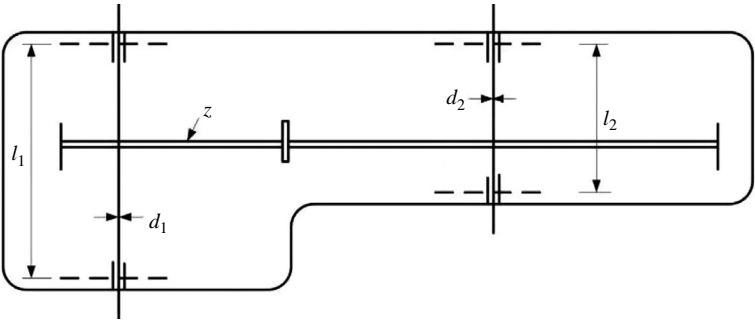|	Park *et al.* (2007)	Ray and Saini (2001)	Tsai (2005)	Present study
$x_1(A_1)$	0.78879	0.79500	0.78800	0.78863
$x_2(A_2)$	0.40794	0.39500	0.40800	0.40838
g_1	0.00000	-0.00169	0.00082 [a]	0.00000
g_2	-0.26778	-0.26124	-0.26740	-0.26802
g_3	-0.73223	-0.74045	-0.73178	-0.73198
f_{min}	263.8965	264.3000	263.6800	263.8962

Note: [a]Italized set is violated sets

Figure 3.
Speed reducer

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:

$$\text{Minimize}: f(X) = 0.7854x1x2^2(3.3333x3^2 + 14.9334x3 - 43.0934) - 1.508x1(x_6^2 + x_7^2)$$
$$+ 7.477(x_6^3 + x_7^3) + 0.7854(x_4x_6^2 + x_5x_7^2)$$

Subject to:

$$g_1 = \frac{27}{x_1x_2^2x_3}P - 1 \le 0 \tag{18}$$

$$g_2 = \frac{397.5}{x_1x_2^2x_3^2} - 1 \le 0 \tag{19}$$

$$g_3 = \frac{1.93}{x_2x_3x_4^3x_6^4} - 1 \le 0 \tag{20}$$

$$g_4 = \frac{1.93}{x_2x_3x_4^3x_7^4} - 1 \le 0 \tag{21}$$

$$g_5 = \frac{\sqrt{(745x_4/x_2x_3)^2 + 1.69 \times 10^6}}{110x_6^3} - 1 \le 0 \tag{22}$$

$$g_6 = \frac{\sqrt{(745x_4x_2x_3)^2 + 157.5 \times 10^6}}{85x_7^3} - 1 \le 0 \tag{23}$$

$$g_7 = \frac{x_2x_3}{40} - 1 \le 0 \tag{24}$$

$$g_8 = \frac{5x_2}{B - 1} - 1 \le 0 \tag{25}$$

$$g_9 = \frac{x_1}{12x_2} - 1 \le 0 \tag{26}$$

In addition, the design variables are also subject to simple bounds list in Table VII. This problem has been solved by using BA, and the corresponding statistical values of the best solutions are also presented in Table VII.

Table VIII summarizes a comparison of the results obtained by BA with those obtained by other methods. Although some of the best objective values are better than those of BA, these reported values are not feasible because some of the constraints are violated. Thus, BA obtained the best feasible solution for this problem.

4.5 Case 5: 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. Sanayei and Saletnik (1996) developed a parameter estimation benchmark using measured strains for simultaneous estimation of the structural

	Bound	Value
x_1 (b)	[2.6-3.6]	3.50000
x_2 (m)	[0.7-0.8]	0.70000
x_3 (z)	[17-28]	17.0000
x_4 (l₁)	[7.3-8.3]	7.30001
x_5 (l₂)	[7.3-8.3]	7.71532
x_6 (d₁)	[2.9-3.9]	3.35021
x_7 (d₂)	[5.0-5.5]	5.28665
Objective function value		2,994.4671
No. bats		15
No. iterations		15,000
Time of each run (s)		2.53317

Table VII.
Statistical results
of the speed reducer
design example

	Kuang et al. (1998)	Akhtar et al. (2002)	Golinski (1973)	Ray and Saini (2001)	Hsu and Liu (2007)	Li and Papalambros (1985)	Present study
Best	2,876.1176	3,008.08	2,985.2	2,732.9006	3,007.8	2,985.22	2,994.4671
x_1	3.6	3.506122	3.5	3.514185	3.5197	3.500243	3.50000
x_2	0.7	0.700006	0.7	0.700005	0.7039	0.70	0.70000
x_3	17	17	17	17	17.3831	17	17.0000
x_4	7.3	7.549126	7.3	7.497343	7.3	7.3	7.30001
x_5	7.8	7.85933	7.3	7.8346	7.7152	7.8	7.71532
x_6	3.4	3.365576	3.35	2.9018	3.3498	3.4	3.35021
x_7	5	5.289773	5.29	5.0022	5.2866	5.0	5.2875
g_1	− 0.100	− 0.076	− 0.074	− 0.078	− 0.246	− 0.074	− 0.074
g_2	− 0.220	− 0.199	− 0.198	− 0.201	− 0.513	− 0.198	− 0.198
g_3	− 0.528	− 0.456	− 0.499	− 0.036	− 0.907	− 0.528	− 0.499
g_4	− 0.877	− 0.899	− 0.919	− 0.875	0.000	− 0.877	− 0.905
g_5	− 0.043	− 0.013	0.000	*0.540*	0.000	− 0.043	0.000
g_6	*0.182*	− 0.002	− 0.002	*0.181*	− 0.694	*0.182*	0.000
g_7	− 0.703	− 0.703	− 0.703	− 0.703	0.000	− 0.703	− 0.703
g_8	− 0.028	− 0.002	0.000	− 0.004	− 0.583	0.000	0.000
g_9	− 0.571	− 0.583	− 0.583	− 0.582	− 0.051	− 0.583	− 0.583
g_{10}	− 0.041	− 0.080	− 0.051	− 0.166	0.000	− 0.041	− 0.051
g_{11}	− 0.051	− 0.018	*0.057*	− 0.055	− 0.246	− 0.051	0.000
Mean	NA	3,012.120	NA	2,758.888	NA	NA	2,994.4671
Max.	NA	3,028.280	NA	2,780.307	NA	NA	4,973.8644
SD	NA	NA	NA	NA	NA	NA	721.51803

Notes: Italized sets are violated sets

Table VIII.
Statistical results
of the speed reducer
design example

parameters. The parameter estimation objective function is defined as follows:

$$\text{Minimize} : \sum_{i=1}^{NMS} \left| \frac{([\varepsilon_a]_{m,i} - [\varepsilon_a]_{a,i})}{[\varepsilon]_{m,i}} \right| \qquad (27)$$

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

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

$$[\varepsilon] = [B][K]^{-1}[F] \tag{28}$$

It is not required to measure all the strains, therefore, equation (32) is partitioned based on measured strain $\boldsymbol{a}$ and unmeasured strain $\boldsymbol{b}$:

$$\begin{bmatrix} \varepsilon_a \\ \varepsilon_b \end{bmatrix} = \begin{bmatrix} B_a \\ B_b \end{bmatrix}[K]^{-1}[F] \tag{29}$$

Since there is no need for unmeasured strains $[\boldsymbol{\varepsilon_b}]$ is eliminated as:

$$[\varepsilon_a] = [B_a][K]^{-1}[F] \tag{30}$$

In this work, the case study is a frame structure presented by Sanayei and Saletnik (1996) (Figure 4). The identified parameter in this example is moment of inertia $I(X)$ 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 section areas are, respectively, 484 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 $X = [869, 869, 869, 869, 869, 1,320, 1,320]$ (cm^4) with corresponding function value equal to $f^*(X) = 0.00000$. The statistical results for this case study provided by BA are presented in Table IX.

The analytical algorithm proposed by Sanayei and Saletnik (1996) is not applicable to this problem due to a singularity. Arjmandi (2010) solved this problem using GA. A comparison of the results obtained by GA and BA with the measured values is shown in Figure 5. The results show that BA has found the global optimum and identified all the parameters without any error.
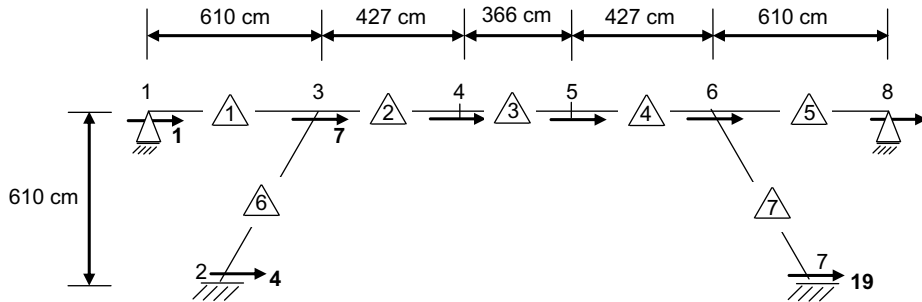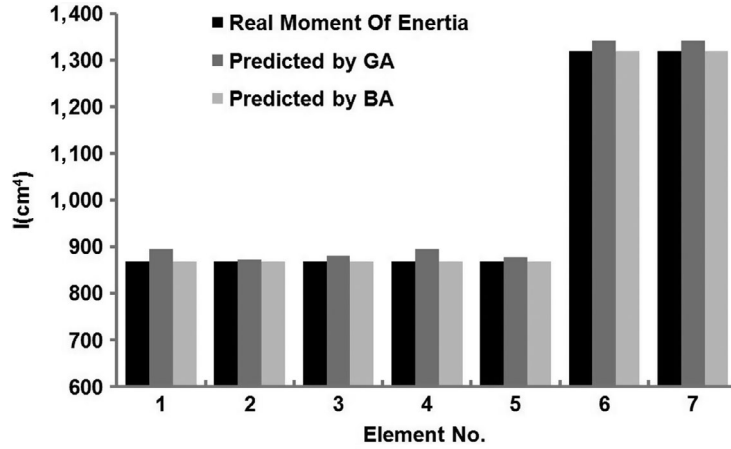

Figure 4.
Frame structure used for parameter identification example

Best	Mean	Worst	SD	No. bats	No. evals.	Ave. time (s)
5.39×10^{-14}	5.1052×10^{-10}	5.1146×10^{-4}	2.491×10^{-5}	25	25,000	14.9

Table IX.
Best solutions for the parameter identification example using BA

Figure 5.
Parameter identification
results using GA and BA

4.6 Case 6: cantilever stepped beam
The capability of BA for continuous and discrete variable design problems are verified
using a design problem with ten variables. The case is originally presented by Thanedar
and Vanderplaats (1995). Figure 6 shows a five-stepped cantilever beam with
rectangular shape. In this case study, the width ($x_1 - x_5$) and height ($x_6 - x_{10}$) of the
beam in all five steps of the cantilever beam are design variables. The volume of the
beam is to be minimized. The objective function is formulated as follows:

$$\text{Minimize} : V = \sum_{i=1}^{5} x_i x_{i+5} l_i \tag{31}$$

where $l_i = 100\,\text{cm}$ (i = 1, 2, ..., 5).
    Subject to the following constraints:

$$g_1 = \frac{600P}{x_5 x_{10}^2} - 14000 \leq 0 \tag{32}$$

$$g_2 = \frac{6P(l_s + l_4)}{x_4 x_9^2} - 14000 \leq 0 \tag{33}$$

$$g_3 = \frac{6P(l_s + l_4 + l_3)}{x_3 x_8^2} - 14000 \leq 0 \tag{34}$$

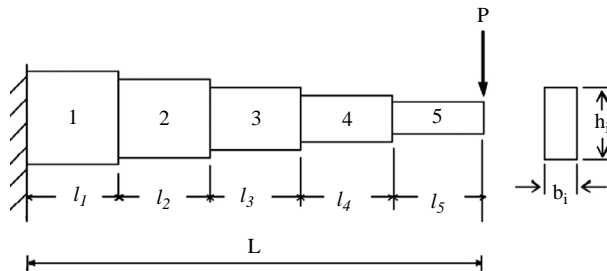**Figure 6.**
A stepped cantilever beam

$$g_4 = \frac{6P(l_s + l_4 + l_3 + l_2)}{x_2 x_7^2} - 14000 \leq 0 \tag{35}$$

$$g_5 = \frac{6P(l_s + l_4 + l_3 + l_2 + l_1)}{x_1 x_6^2} - 14000 \leq 0 \tag{36}$$

$$g_6 = \frac{Pl^3}{3E} \left(\frac{1}{I_s} + \frac{7}{I_4} + \frac{19}{I_3} + \frac{37}{I_2} + \frac{61}{I_1} \right) - 2.7 \leq 0 \tag{37}$$

$$g_7 = \frac{x_{10}}{x_5} - 20 \leq 0 \tag{38}$$

$$g_8 = \frac{x_9}{x_4} - 20 \leq 0 \tag{39}$$

$$g_9 = \frac{x_8}{x_3} - 20 \leq 0 \tag{40}$$

$$g_{10} = \frac{x_7}{x_2} - 20 \leq 0 \tag{41}$$

$$g_{11} = \frac{x_6}{x_1} - 20 \leq 0 \tag{42}$$

where $P = 50,000\,\text{N}$, $E = 2 \times 10^7\,\text{N/cm}^2$ and the initial design space are: $1 \leq x_i \leq 5$ $(i = 1, 2, \ldots, 5)$, and $30 \leq x_j \leq 65$ $(j = 6, 7, \ldots, 10)$.

BA has achieved a solution that satisfies all the constraints and it reaches the best solution, possibly the unique global optimum. BA outperforms the previous other methods in terms of the minimum objective function value. Table X presents the results obtained by BA. We can see that the proposed method requires 25 bats and 1,000 iterations to reach the optimum.

This nonlinear constrained problem has been solved by other researchers shown in Table XI. As it is seen, BA significantly outperforms other studies.

4.7 Case 7: heat exchanger design
As another case study, we now try to solve the heat exchanger design task, which is a difficult benchmark minimization problem since all the constraints are binding. It involves eight design variables and six inequality constraints (three linear and three non-linear). The problem is expressed as follows:

$$\text{Minimize}: \ f(X) = x_1 + x_2 + x_3 \tag{43}$$

Subject to:

$$g_1 = 0.0025(x_4 + x_6) - 1 \leq 0 \tag{44}$$

Best	Mean	Worst	SD	No. bats	No. evals.	Ave. time (s)
61,914.86841	61,914.86842	61,914.86845	0.00001	25	25,000	7.025

Table X.
Best solution results for the stepped cantilever beam examples

	Thanedar and Vanderplaats (1995)	Lamberti and Pappalettere (2003)	Huang and Arora (1997)	Present study
x_1	3.06	NA	NA	2.99204
x_2	2.81	NA	NA	2.77756
x_3	2.52	NA	NA	2.52359
x_4	2.2	NA	NA	2.20455
x_5	1.75	NA	NA	1.74977
x_6	61.16	NA	NA	59.84087
x_7	56.24	NA	NA	55.55126
x_8	50.47	NA	NA	50.4718
x_9	44.09	NA	NA	44.09106
x_{10}	35.03	NA	NA	34.99537
Best objective	63,110	65,352.2	63,108.7	61,914.9

Table XI.
Statistical results of the stepped cantilever beam example using different methods

$$g_2 = 0.0025(x_5 + x_7 - x_4) - 1 \leq 0 \qquad (45)$$

$$g_3 = 0.01(x_8 - x_5) - 1 \leq 0 \qquad (46)$$

$$g_4 = 833.33252x_4 + 100x_1 - x_1x_6 - 83333.333 \leq 0 \qquad (47)$$

$$g_5 = 1250x_5 + x_2x_4 - x_2x_7 - 125x_4 \leq 0 \qquad (48)$$

$$g_6 = x_3x_5 - 2500x_5 - x_3x_8 + 125 \times 10^4 \leq 0 \qquad (49)$$

Table XII shows the best solution for the heat exchanger design obtained by BA as well as the best solutions obtained previously by other methods. The solution shown for BA is the best generated using 25 bats. The solution generated by BA (with $X^* = [579.30675, 1,359.97076, 5,109.97052, 182.01770, 295.60118, 217.98230, 286.41653, 395.60118]$) is better than the best solutions reported in the literature. As shown in Table XII, the SD and the number of evaluations using BA are also much less than those obtained by the other methods. This solution is feasible and

Author(s)	Best	Mean	Worst	SD	No. evalus.
Lee and Geem (2004)	7,057.2744	NA	NA	NA	NA
Joines and Houck (1994)	7,068.688	7,244.2786	NA	107.7516	NA
Jaberipour and Khorram (2010) (1)	7,109.1901	NA	NA	NA	200,000
Jaberipour and Khorram (2010) (2)	7,051.3012	NA	NA	NA	200,000
Deb (2000)	7,060.221	NA	NA	NA	320,080
Michalewicz (1995)	7,377.976	NA	NA	NA	NA
Shopova and Vaklieva-Bancheva (2006)	7,095.485	NA	NA	NA	NA
Chootinan and Chen (2006)	7,049.2607	7,049.5659	7,051.6857	0.57	NA
Koziel and Michaelwicz (1999)	7,147.9	8,163.6	9,659.3	NA	NA
Runarsson and Yao (2000)	7,054.316	7,559.192	8,835.655	530	NA
Farmani and Wright (2003)	7,061.34	7,627.89	8,288.79	373	NA
Amirjanov (2006)	7,054.316	7,372.613	8,835.655	1,000	128,000
Wright and Farmani (2001)	7,152.83	NA	NA	NA	NA
Ben Hamida and Schoenauer (2000)	7,095.15	NA	NA	NA	NA
Present study	7,049.248	7,049.2484	7,049.3307	0.00523	25,000

Table XII.
Statistical results of the heat exchanger design example by different model

the constraint values are $G^* = [0.0000000, \ 0.0000000, \ 0.0000000, \ -0.0071449, \ -0.0061782, \ -0.0020000]$.

4.8 Case 8: car side impact design

Design of car side impact is used as a benchmark problem of the proposed BA. On the foundation of European Enhanced Vehicle-Safety Committee procedures, a car is exposed to a side impact (Youn *et al.*, 2004). 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 formulated as follow:

$$\text{Minimize } f(\boldsymbol{x}) = \textit{Weight}; \tag{50}$$

Subject to:

$$g_1(\boldsymbol{x}) = F_a(\textit{load in abdomen}) \leq 1\,\text{kN}; \tag{51}$$

$$g_2(\boldsymbol{x}) = V \times Cu(\textit{dummy upper chest}) \leq 0.32\,\text{m/s}; \tag{52}$$

$$g_3(\boldsymbol{x}) = V \times Cm(\textit{dummy middle chest}) \leq 0.32\,\text{m/s}; \tag{53}$$

$$g_4(\boldsymbol{x}) = V \times Cl(\textit{dummy lower chest}) \leq 0.32\,\text{m/s}; \tag{54}$$

$$g_5(\boldsymbol{x}) = \Delta_{ur}(\textit{upper rib deflection}) \leq 32\,\text{mm}; \tag{55}$$

$$g_6(\boldsymbol{x}) = \Delta_{mr}(\textit{middle rib deflection}) \leq 32\,\text{mm}; \tag{56}$$

$$g_7(\boldsymbol{x}) = \Delta_{lr}(\textit{lower rib deflection}) \leq 32\,\text{mm}; \tag{57}$$

$$g_8(\boldsymbol{x}) = F_p(\textit{Pubic force}) \leq 4\,\text{kN}; \tag{58}$$

$$g_9(\boldsymbol{x}) = V_{MBP}(\textit{Velocity of } V - \textit{Pillar at middle po\,int}) \leq 9:9\,\text{mm/ms}; \tag{59}$$

$$g_{10}(\boldsymbol{x}) = V_{FD}(\textit{Velocity of front door at } V - \textit{Pillar}) \leq 15:7\,\text{mm/ms}; \tag{60}$$

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, we ran BA with 20 bats and 1,000 iterations. Because this case study has not been solved previously in the literature, we also solved this problem using PSO, DE and GA methods so as to benchmark and compare with the BA method. Table XIII shows the statistical results for the car side impact design problem using the proposed BA method and other well-known methods after 20,000 searches. As it can be seen from Table XIII, in comparison with other heuristic algorithms, the proposed algorithm is better than GA and it seems that the BA method performances similar to the PSO and DE.

5. Discussions and conclusions

We have presented a new BA for solving engineering optimization problems. BA has been validated using several benchmark engineering design problems, and it is found

Method	PSO	DE	GA	BA
Best objective	22.84474	22.84298	22.85653	22.84474
x_1	0.50000	0.50000	0.50005	0.50000
x_2	1.11670	1.11670	1.28017	1.11670
x_3	0.50000	0.5000	0.50001	0.50000
x_4	1.30208	1.30208	1.03302	1.30208
x_5	0.50000	0.50000	0.50001	0.50000
x_6	1.50000	1.50000	0.50000	1.50000
x_7	0.50000	0.50000	0.50000	0.50000
x_8	0.34500	0.34500	0.34994	0.34500
x_9	0.19200	0.19200	0.19200	0.19200
x_{10}	-19.54935	-19.54935	10.3119	-19.54935
x_{11}	-0.00431	-0.00431	0.00167	-0.00431
Mean objective	22.89429	23.22828	23.51585	22.89273
Worst objective	23.21354	24.12606	26.240578	23.21354
SD	0.15017	0.34451	0.66555	0.17383

Table XIII.
Statistical results of the car side design example by different methods

from our simulations that BA is very efficient. The extensive comparison study, carried out over seven different nonlinear constrained design tasks, reveals that BA performs superior to many different existing algorithms used to solve these seven benchmark problems. It is potentially more powerful than other methods such as GA and PSO as well as HS. The primary reason is that BA uses a good combination of major advantages of these algorithms in some way. Moreover, PSO and HS are the special cases of BA under appropriate simplifications. More specifically, if we fix the loudness as $A_i = 0$ and pulse emission rate as $r_i = 1$, BA reduces to the standard particle swarm optimization. On the other hand, if set $A_i = r_i = 0.7$-0.9, BA essentially becomes a HS as frequency change is equivalent to the pitch adjustment in HS.

Sensitivity studies can be an important issue for the further research topics, as the fine adjustment of the parameters α and γ can affect the convergence rate of the BA. This is true for almost all metaheuristic algorithms. In fact, parameter α plays a similar role as the cooling schedule in the simulated annealing. Though the implementation is more complicated than many other metaheuristic algorithms; however, the detailed study of seven engineering design tasks indicates that BA actually uses a balanced combination of the advantages of existing successful algorithms with innovative feature based on the echolocation behavior of microbats. New solutions are generated by adjusting frequencies, loudness and pulse emission rates, while the proposed solution is accepted or not depends on the quality of the solutions controlled or characterized by loudness and pulse rate which are in turn related to the closeness or the fitness of the locations/solution to the global optimal solution.

Theoretically speaking, if we simplify the system with enough approximations, it is possible to analyze the behaviour of the BA using analysis in the framework of dynamical systems. In addition, more extensive comparison studies with a more wide range of existing algorithms using much tough test functions in higher dimensions will pose more challenges to the algorithms, and thus such comparisons will potentially reveal the virtues and weakness of all the algorithms of interest. Furthermore, a natural extension is to formulate a discrete version of BA so that it can directly solve combinatorial optimization problems such as the travelling salesman problem.

On the other hand, for dynamical optimization problems and computational geometry, a further natural extension to the current BA would be to use the directional echolocation and Doppler effect, which may lead to even more interesting variants and new algorithms. These further extensions will help us to design more efficient, often hybrid, algorithms to solve a wider class of even tougher optimization problems.

References

Akhtar, S., Tai, K. and Ray, T. (2002), "A socio-behavioral simulation model for engineering design optimization", *Eng. Optmiz.*, Vol. 34 No. 4, pp. 341-54.

Alavi, A.H. and Gandomi, A.H. (2011), "A robust data mining approach for formulation of geotechnical engineering systems", *International Journal of Computer Aided Methods in Engineering-Engineering Computations*, Vol. 28 No. 3, pp. 242-74.

Altringham, J.D. (1996), *Bats: Biology and Behaviour*, Oxford Univesity Press, Oxford.

Amirjanov, A. (2006), "The development of a changing range genetic algorithm", *Comput. Methods Appl. Mech. Eng.*, Vol. 195, pp. 2495-508.

Arjmandi, P. (2010), "Damage detection of continues steel beams using static data", MSc thesis, Tafresh University, Tafresh.

Ben Hamida, S. and Schoenauer, M. (2000), "An adaptive algorithm for constrained optimization problems", *Proceedings of Parallel Problem Solving from Nature*, Vol. VI, pp. 529-38.

Chen, T.W.C. and Vassiliadis, V.S. (2003), "Solution of general nonlinear optimization problems using the penalty/modified barrier method with the use of exact Hessians", *Comput. Chem. Eng.*, Vol. 27, pp. 501-25.

Chootinan, P. and Chen, A. (2006), "Constraint handling in genetic algorithms using a gradient-based repair method", *Comput. Oper. Res.*, Vol. 33, pp. 2263-81.

Coello, C.A.C. (2000), "Use of a self-adaptive penalty approach for engineering optimization problems", *Comput. Ind.*, Vol. 41 No. 2, pp. 113-27.

Deb, K. (1995), *Optimization for Engineering Design: Algorithms and Examples*, Prentice-Hall, New Delhi.

Deb, K. (2000), "An efficient constraint handling method for genetic algorithms", *Comput. Methods Appl. Mech. Eng.*, Vol. 186, pp. 311-38.

Farmani, R. and Wright, J.A. (2003), "Self-adaptive fitness formulation for constrained optimization", *IEEE T. Evolut. Comput.*, Vol. 7 No. 5, pp. 445-55.

Fesanghary, M., Mahdavi, M., Minary-Jolandan, M. and Alizadeh, Y. (2008), "Hybridizing harmony search algorithm with sequential quadratic programming for engineering optimization problems", *Comput. Methods Appl. Mech. Eng.*, Vol. 197, pp. 3080-91.

Gandomi, A.H. and Alavi, A.H. (2011), "Multi-stage genetic programming: a new strategy to nonlinear system modeling", *Information Sciences*, Vol. 181 No. 23.

Gandomi, A.H. and Yang, X.S. (2011), "Benchmark problems in structural optimization", in Koziel, S. and Yang, X.S. (Eds), *Computational Optimization, Methods and Algorithms*, Chapter 12, Springer, Berlin, pp. 267-91.

Gandomi, A.H., Yang, X.S. and Alavi, A.H. (2011), "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems", *Engineering with Computers*, Vol. 27.

Geem, Z.W., Kim, J.H. and Loganathan, G.V. (2001), "A new heuristic optimization algorithm: harmony search", *Simulation*, Vol. 76, pp. 60-8.

Gen, M. and Cheng, R. (1997), *Genetic Algorithms & Engineering Design*, Wiley, New York, NY.

Golinski, J. (1973), "An adaptive optimization system applied to machine synthesis", *Mech. Mach. Synthesis*, Vol. 8 No. 4, pp. 419-36.

He, S., Prempain, E. and Wu, Q.H. (2004), "An improved particle swarm optimizer for mechanical design optimization problems", *Eng. Optimiz.*, Vol. 36 No. 5, pp. 585-605.

Himmelblau, D.M. (1972), *Applied Nonlinear Programming*, McGraw-Hill, New York, NY.

Homaifar, A., Lai, S.H.V. and Qi, X. (1994), "Constrained optimization via genetic algorithms", *Simulation*, Vol. 62 No. 4, pp. 242-54.

Hsu, Y.L. and Liu, T.C. (2007), "Developing a fuzzy proportional-derivative controller optimization engine for engineering design optimization problems", *Eng. Optimiz.*, Vol. 39 No. 6, pp. 679-700.

Huang, M.-W. and Arora, J.S. (1997), "Optimal design with discrete variables: some numerical experiments", *Int. J. Numer. Meth. Eng.*, Vol. 40, pp. 165-88.

Jaberipour, M. and Khorram, E. (2010), "Two improved harmony search algorithms for solving engineering optimization problems", *Commun. Nonlinear Sci. Numer. Simulat.*, Vol. 15 No. 11, pp. 3316-31.

Joines, J. and Houck, C. (1994), "On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with gas", in Fogel, D. (Ed.), *Proceedings First IEEE Conf. on Evolutionary Computation, Orlando, FL*, IEEE Press, Piscataway, NJ, pp. 579-84.

Kaveh, A. and Talatahari, S. (2010), "A novel heuristic optimization method: charged system search", *Acta Mech.*, Vol. 213 Nos 3/4, pp. 267-89.

Kennedy, J. and Eberhart, R. (1995), "Particle swarm optimization", *Proceedings IEEE Int. Conf. Neural Networks, Perth, Australia*, pp. 1942-5.

Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P. (1983), "Optimization by simulated annealing", *Science*, Vol. 220 No. 4598, pp. 671-80.

Koziel, S. and Michaelwicz, Z. (1999), "Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization", *Evol. Comput.*, Vol. 7 No. 1, pp. 19-44.

Kuang, J.K., Rao, S.S. and Chen, L. (1998), "Taguchi-aided search method for design optimization of engineering systems", *Eng. Optimiz.*, Vol. 30, pp. 1-23.

Lamberti, L. and Pappalettere, C. (2003), "Move limits definition in structural optimization with sequential linear programming. Part II: numerical examples", *Computers & Structures*, Vol. 81, pp. 215-38.

Lee, K.S. and Geem, Z.W. (2004), "A new meta-heuristic algorithm for continues engineering optimization: harmony search theory and practice", *Comput. Methods Appl. Mech. Eng.*, Vol. 194, pp. 3902-33.

Li, H.L. and Papalambros, P. (1985), "A production system for use of global optimization knowledge", *ASME. J. Mech Transm. Autom. Des.*, Vol. 107, pp. 277-84.

Michalewicz, Z. (1995), "Genetic algorithms, numerical optimization, and constraints", in Eshelman, L. (Ed.), *Proceedings Sixth Int. Conf. on Genetic Algorithms*, Morgan Kaufmann, Waltham, MA, pp. 151-8.

Nowcki, H. (1974), "Optimization in pre-contract ship design", in Fujita, Y., Lind, K. and Williams, T.J. (Eds), *Computer Applications in the Automation of Shipyard Operation and Ship Design*, Vol. 2, Elsevier, New York, NY, pp. 327-38.

Omran, M.G.H. and Salman, A. (2009), "Constrained optimization using CODEQ", *Chaos Soliton. Fract.*, Vol. 42, pp. 662-8.

Park, Y.C., Chang, M.H. and Lee, T.Y. (2007), "A new deterministic global optimization method for general twice differentiable constrained nonlinear programming problems", *Eng. Optimiz.*, Vol. 39 No. 4, pp. 397-411.

Ray, T. and Saini, P. (2001), "Engineering design optimization using a swarm with an intelligent information sharing among individuals", *Eng. Optimiz.*, Vol. 33 No. 3, pp. 735-48.

Richardson, P. (2008), *Bats*, Natural History Museum, London.

Runarsson, T.P. and Yao, X. (2000), "Stochastic ranking for constrained evolutionary optimization", *IEEE T. Evolut. Comput.*, Vol. 4 No. 3, pp. 284-94.

Sanayei, M. and Saletnik, M.J. (1996), "Parameter estimation of structures from static strain measurements I: formulation", *J. Struct. Eng-ASCE*, Vol. 122 No. 5, pp. 555-62.

Shi, Y and Eberhart, R.C. (1998), "A modified particle swarm optimizer", *Proceedings Int. Cong. Evolutionary Computation*, IEEE Service Center, Piscataway, NJ.

Shopova, E.G. and Vaklieva-Bancheva, N.G. (2006), "BASIC – a genetic algorithm for engineering problems solution", *Comput. Chem. Eng.*, Vol. 30, pp. 1293-309.

Thanedar, P.B. and Vanderplaats, G.N. (1995), "Survey of discrete variable optimization for structural design journal of structural engineering", *J. Struct. Eng.-ASCE*, Vol. 121 No. 2, pp. 301-6.

Tsai, J. (2005), "Global optimization of nonlinear fractional programming problems in engineering design", *Eng. Optimiz.*, Vol. 37 No. 4, pp. 399-409.

Wright, J.A. and Farmani, R. (2001), "Genetic algorithm: a fitness formulation for constrained minimization", *Proceedings of Genetic and Evolutionary Computation Conf., San Francisco, CA, July 7-11*, pp. 725-32.

Yang, X.-S. (2008), *Nature-inspired Metaheuristic Algorithms*, Luniver Press, Beckington.

Yang, X.-S. (2010), "A new metaheuristic bat-inspired algorithm", in Cruz, C., Gonzalez, J., Krasnogor, N. and Terraza, G. (Eds), *Nature Inspired Cooperative Strategies for Optimization* (*NICSO 2010*), Vol. SCI 284, Springer, Berlin, pp. 65-74.

Youn, B.D., Choi, K.K., Yang, R.-J. and Gu, L. (2004), "Reliability-based design optimization for crashworthiness of vehicle side impact", *Struct. Multidisc. Optim.*, Vol. 26, pp. 272-83.

Corresponding author
Amir Hossein Gandomi can be contacted at: a.h.gandomi@gmail.com