VOL. 11, NO. 14, JULY 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

COMPARISON OF EVOLUTIONARY ALGORITHMS IN CONSTRAINED ENGINEERING OPTIMIZATION

Ong Pauline[1], Noor Hajar Mhd Khair[1], Muhaimin Yusup[1], Mohamad Amri Mazlan[1], Sia Chee Kiong[1], Badrul Aisham Md Zain[1], Noormaziah Jafferi[1], Hanani Abd Wahab[1], Ismail Nawi[1], Khairulnizam Ngadimon[1] and Zarita Zainuddin[2]
[1]Advanced Dynamic Control Research Group, Faculty of Mechnaical and Manufacturing Engineering, Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, Johor, Malaysia
[2]School of Mathematical Sciences, Universiti Sains Malaysia, USM, Pulau Pinang, Malaysia
E-Mail: ongp@uthm.edu.my

ABSTRACT

Advances in computer technology, coupled with the intention to utilize the limited resources to its best possible way while conforming to the prescribed objective, has led to a wealth of different optimization approaches in engineering problems. Of particular note is the rise of using evolutionary algorithms in obtaining the optimal solutions in the engineering design problems. This paper sets out to compare the optimization performances of three recently developed evolutionary algorithms, namely, bat algorithm, cuckoo search algorithm, and flower pollination algorithm in constrained engineering optimization. Three infamous constrained real world problems, specifically, spring design, welded beam design, and pressure vessel design, are considered in this study. The promising optimization capabilities of all reviewed evolutionary algorithms are shown in the performance assessment.

Keywords: bat algorithm, constrained optimization, cuckoo search algorithm, and flower pollination algorithm.

INTRODUCTION

Constrained optimization, in general, attempts to maximize or minimize the prescribed objective function, while conforming to the presence of set of constraints [1]. Various industrial engineering activities involve constrained optimization in its core, in which the underlying problems are usually unstructured and complex in nature. For instance, the job scheduling that needs to consider the production line efficiency in relevance to the cost, available resource, workflows and recovery of failures [2], the pulping process that requires to maximize the paper strength while minimize the kappa number simultaneously [3], inevitably necessitate constrained optimization of the problems at hand.

The Lagrange multipliers, penalty method and direct approach, are the primary approaches used to handle constrained optimization problem [4]. The Lagrange multiplier, perhaps the easiest solution strategy with meticulous foundation, involves the addition of terms that describe the constraints to the objective function. Meanwhile, the penalty method transforms the constrained problem into unconstrained problem by adding a penalty function. The direct approach locates the feasible solution within the region defined by the constraints; however, it is often subject to slow convergence. On the other hand, the Lagrange multiplier requires both objective function and constraints to have continuous first partial derivatives, which is not always valid in real scenarios.

The evolutionary algorithms, which draw inspiration from how living creatures adjust to constantly changing conditions imposed to them by their living environment, have gained prevalence in solving constrained optimization tasks recently. The upsurge popularity, probably, is attributed to their flexibility to adapt the solution strategies according to different problem requirements. Moreover, since the convergence to optimal solution involves merely a set of rules, the evolutionary algorithms are gradient free and the search space is not confined within the region defined by the constraints [1]. It thus overcomes those hurdles in the Lagrange multiplier and direct approach.

Such superiority of the evolutionary algorithms triggers their vast application in constrained engineering optimization problem. Ong *et al.* applied the adaptive cuckoo search algorithm to determine the optimal experimental milieu of the pulping process [3]. Mittal *et al.* solved the turbine placement problem in wind farm, in which the used strategy was a combination of genetic algorithm (GA) and deterministic gradient based methods [5]. The evolutionary approach, too, has been utilized in optimizing the aerodynamic shape of the tilt-rotor cockpit region [6]. Xia *et al.* optimized the transient stability constrained optimal power flow in AC transmission system with particle swarm optimization (PSO) [7]. In order to achieve the minimum maintenance cost of a highway network, the optimal pavement maintenance and rehabilitation scheduling was generated using GA in [8].

The bat algorithm (BA) [9], cuckoo search algorithm (CSA) [10] and flower pollination algorithm (FPA) [11], are types of evolutionary algorithms. Preserving the simple yet effective features as in other evolutionary algorithms, their implementation in diverse domains; including economic dispatch problem [12], image segmentation [13], traveling salesman problem [14], distribution feeder configuration [15], design of plate fin heat exchanger [16] and design of FIR fractional order differentiator [17], have been investigated.

In this study, the BA, CSA and FPA are used to determine the optimal design of three well-known

 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

constrained engineering problems, specifically, the spring design problem, welded beam design problem and pressure vessel design problem. Performance comparisons with other state-of-art methods are performed subsequently.

The paper is organized as follows. Firstly, the brief introduction of BA, CSA and FPA are given, followed by the problem description and solution. Lastly, some conclusions are drawn.

BAT ALGORITHM

The BA is inspired by the echolocation phenomenon of microbats [9]. Such echolocation behaviour is used for various reasons: to hunt prey, dodge obstacles and find their way back to their habitat. Echolocation is crucial to microbats during the night. The echolocation pulses that bounce back from the surrounding area are the primary way to help the mircobats to fly better in the dark.

In mimicking such echolocation phenomenon of microbats, some idealized rules have been used in [9] for simplicity, which are given as follows:

- All microbats use echolocation to determine distance, different types of prey and the obstacles.
- Microbats fly randomly when looking for prey with velocity v_i at position x_i with a fixed frequency $f_{\min}$, but different wavelength λ and loudness A_0. From the emitted pulses, wavelength or frequency and rate of pulse emission $r \in [0,1]$ can be adjusted automatically depending on the proximity of their target.
- Even though the loudness can be different in many ways, it is assumed that it ranges from a large (positive) A_0 to a minimum constant value $A_{\min}$.

Based on these idealized assumptions, Figure-1 summarizes the basic steps of the BA.

```
Bat Algorithm
Objective function f((x)),  x = (x_1,...,x_d)^T
Initialize the bat population X_i and V_i, where
i = 1,2,...,n
Define pulse frequency f_i at X_i
Initialize pulse rates r_i and the loudness A_i
while (t<MaxGeneration)
Generate new solutions by adjusting frequency and
updating velocities and locations using Eqn. (1)-(3)
if rand >r_i
        Select a solution among 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
```

Figure-1. Solution strategy of the BA [9].

In the BA, the motion of each bat is related to its velocity, v_i, at a position, x_i, fixed frequency $f_{\min}$, inconsistent wavelength, λ, and loudness, A_o. The generated new solution x_i^t at time step t which corresponds to velocity v_i^t is formulated as:

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

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

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

where $\beta \in [0, 1]$ is a random vector drawn from uniform distribution while x_* is the current global best solution among all n bats.

In real world scenario, the loudness of pulse is changing when the microbats are flying. From this, microbats are able to dodge any obstruction even the littlest, like the human hair. Loudness is usually reduced while pulse emission is increased when a bat has targeted its prey. In accordance to such situation, the loudness, A_i and the rate of pulse emission, r_i in the BA are revised iteratively as follows:

$$A_i^{t+1} = \alpha A_i^t, \qquad r_i^{t+1} = r_i^0 \left(1 - e^{-\gamma t}\right) \tag{4}$$

where α and γ are constants. The constant α is taken as the same as the cooling factor in the simulated annealing.

CUCKOO SEARCH ALGORITHM

The CSA was developed by Yang and Deb in 2009, in which in essence, the CSA is inspired by how cuckoo lay their eggs in the nest of other birds that might be from different species [10]. Host birds might find out that the eggs in the nest are not theirs and probably the host bird will destroy the eggs or leave the nest. This results in the evolution of cuckoo eggs in order to increase the similarity between the eggs of cuckoos and host birds. To apply such continuous evolutionary arms race as an optimization tool, Yang and Deb used three ideal rules, which are described as follows:

- Each cuckoo lays one egg at a time in a randomly chosen host nest.
- The nests (solutions) with high quality of eggs will be carried over to the next generation.
- The number of available host nest is fixed. The host bird 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.

Based on these rules, the basic steps of the CSA can be summarized as in Figure-2.

Cuckoo Search Algorithm
Objective function f((**x**), $\mathbf{x} = (x_1,...,x_d)^T$
Initialize a population of n host nests $\mathbf{x}_i$,where $i = 1, 2,..., n$
while (t<MaxGeneration)
Generate a cuckoo (say i) randomly by Lévy flight
Evaluate its fitness F_i
Choose a nest among n nests (say j) randomly
if $F_i>F_j$
Replace j by the new solution
end if
Abandon a fraction (p_a) of worst nests solutions and build new nests
Rank the fitness of all solutions and find the current best x_*
end while

Figure-2. Solution strategy of the CSA [10]

The CSA, alike to other evolutionary algorithms, starts with of an initial population, which can be characterized as:

$$x_{i,j} = x_{\min,j} + rand(0,1)(x_{\max,j} - x_{\min,j}) \tag{5}$$

where $i = 1,2,...,n$ and $j = 1,2,...,d$. n represents the number of eggs (potential solutions) while d is the dimension of the search space. $x_{\min,j}$ and $x_{\max,j}$ denote the lower bound and upper bound of dimension j, respectively.

A new solution (cuckoo egg) x_i^{t+1} is then generated by performing the Lévy flight, which is represented as:

$$x_{i,j}^{\,t+1} = x_{i,j}^{\,t} + \alpha L(\lambda) \tag{6}$$

where α> 0 is the step size which should be related to the scales of the problem of interest. In most cases, α = 1 is applied. $L(\lambda)$ is the random step from the Levy distribution, which is given as:

$$L(\lambda) = \left| \frac{\Gamma(1+\lambda)\times \sin(\pi\lambda/2)}{\Gamma[(1+\lambda)/2]\times\lambda\times 2^{(\lambda-1)/2}} \right|^{1/\lambda} \tag{7}$$

Here, Γ is the gamma function while λ is a constant
($1 < \lambda \leq 3$).

The integration of Lévy flight in the exploitation and exploration process of the CSA is an important component to enhance its optimization ability. The Levy flight is based on a random walk which is characterized by a series of jumps simultaneously with the random selection of the probability function which has the power law tail [10]. This type of random search pattern can be found frequently in nature, for instance, the shark's movement. The Lévy flight performs short jumps at most of the time, while occasionally long jumps, assisting the CSA in avoiding from getting trapped in local optima.

FLOWER POLLINATION ALGORITHM

The idea of FPA comes from the pollination of flowers, which was introduced by Yang in 2012 [11]. The flower pollination process, in general, can be classified into abiotic and biotic. If a pollinator, for instance, the insects and animals, are involved in transferring the pollen, such process is belonged to biotic pollination. On the other hand, if the flower pollination requires no pollinators but occurs with the assist of wind and diffusion in water, this type of pollination process is said as abiotic pollination.

The FPA models the flower pollination process by four idealized rules, which are described as follows [11]:

- The biotic pollination is considered as global searching process, in which the pollen-carrying pollinators perform Lévy flight in their movement
- The abiotic pollination is considered as local searching process.
- Flower constancy is regarded as the reproduction probability, in which it is proportional to the similarity among two flowers involved.
- The interaction or switching of local pollination and global pollination is controlled by a switch probability $p \in [0, 1]$.

Based on these idealized rules, the searching process of the FPA is illustrated in Figure-3 [11]. The first idealized assumption which corresponds to the global pollination is represented mathematically as:

$$x_i^{t+1} = x_i^t + L(x_i^t - x_*) \tag{8}$$

Flower Pollination Algorithm (FPA)
Objective function f((**x**), $\mathbf{x} = (x_1,..., x_d)^T$
Initialize a population of n flowers $\mathbf{x}_i$,where $i = 1, 2,..., n$
Find the best solution x_* in the initial population
Define a switch probability $p \in [0,1]$
while (t<MaxGeneration)
for i = 1 : n (all n flowers in the population)
if rand < p
Perform global pollination
else
Perform local pollination
end if
Evaluate new solution
If new solutions are better, update them in the population
end for
Find the current best solution x_*
end while

Figure-3. Solution strategy of the FPA [11].

where x_i^t represents the solution (pollen) at step time t while x_* denotes the best solution among all available solutions. The parameter L is the step size, defining how far a pollinator can travel over a long

VOL. 11, NO. 14, JULY 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

distance. The value of L is drawn from Lévy distribution, which is characterized as:

$$L(\lambda) = \frac{\lambda\Gamma(\lambda)\times\sin(\pi\lambda/2)}{\pi}\frac{1}{s^{1+\lambda}} \quad (9)$$

where $s >> s_0 > 0$. Here, Γ is the gamma function.

The second idealized assumption which corresponds to the local pollination is represented mathematically as:

$$x_i^{t+1} = x_i^t + \varepsilon(x_j^t - x_k^t) \quad (10)$$

where x_j^t and x_k^t are pollen from different flowers from the same species. The local pollination, indeed, is modeled as local random walk, as ε is drawn from a uniform distribution [0,1].

NUMERICAL EXAMPLES

In order to evaluate the optimization performance of the BA, CSA and FPA, three examples of well-studied constrained engineering design problems, namely, the spring design, welded beam design and pressure vessel design, are solved. For all the concerned case studies, the population size is chosen as 20, while the allowable maximum generation number is set to 2000 generations.

For all BA, CSA and FPA, the control parameters have to be assigned judiciously since they keep the balance between exploitation and exploration of the searching process. In this regard, the setup of parameters of BA are: loudness $A = 0.5$, pulse rate $r = 0.5$, minimum frequency $f_{\min} = 0$ and maximum frequency $f_{\max} = 2$; for CSA: step size $\alpha = 1$; and for FPA: probability switch $p = 0.8$.

EXAMPLE 1: OPTIMIZATION OF SPRING DESIGN

Spring is a product that is widely used in the engineering industry worldwide. In the first case study, a spring design problem with three design variables, which are, the diameter wire *d*, the average coil diameter *w*, and the length (or the number of coils) *L*, as illustrated in Figure-4, is concerned.

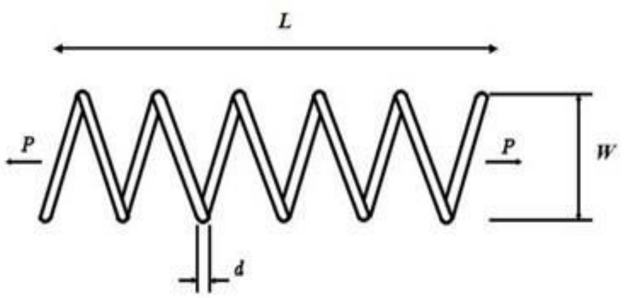

Figure-4. Spring design problem [18].

The main objective of this spring design is to minimize the weight of the spring, which is subject to various constraints such as maximum shear stress, minimum deflection and geometrical limits. This problem can be written compactly as:

$$\text{minimize } f(x) = (L+2)w^2 d \quad (11)$$

subject to the following constraints:

$$g_1(x) = 1 - \frac{d^3 L}{71785 w^4} \le 0 \quad (12)$$

$$g_2(x) = 1 - \frac{140.45 w}{d^2 L} \le 0 \quad (13)$$

$$g_3(x) = \frac{2(w+d)}{3} - 1 \le 0 \quad (14)$$

$$g_4(x) = \frac{d(4d-w)}{w^2(12566d - w)} + \frac{1}{5108 w^2} - 1 \le 0 \quad (15)$$

The design variables are within the ranges of: $0.05 \le w \le 2.0$, $0.25 \le d \le 1.3$ and $2.0 \le L \le 15.0$.

Table-1 presents the optimal design variables obtained by the BA, CSA and FPA, and their performance comparisons to the available results reported in the literature. Several evolutionary algorithms have been used to solve the spring design problem, including harmony search algorithm, mine blast algorithm and PSO, to name but a few. Since the optimization of spring design has been formulated as a minimization problem, the one with the lowest weight $f(x)$ gives the best result, which is given in bold font in Table-1.

It can be concluded that optimizing the spring design with the PSO gives the most satisfactory performance, where the minimum weight of 0.012665 is achieved. However, it can be said that the BA, CSA and FPA are comparable to PSO in this regard, as the obtained minimum weights show merely marginal variations from the best result. The minimum weights, particularly, are 0.127, 0.1277 and 0.1267 for BA, CSA and FPA, respectively, where the differences can be said are not significant. It is worth mentioning that the PSO takes 24,000 generations in order to reach the global optima, but the BA, CSA and FA take 2,000 generations only in this regard. The obtained optimal values of the diameter for the BA, CSA and FPA, basically, are similar to other evolutionary approaches, which is around 0.05. The CSA, however, converges to different optimal values of mean coil diameter and length of wire, which are 0.4076 and 9.0217, respectively.

EXAMPLE 2: OPTIMIZATION OF WELDED BEAM DESIGN

Properly designing of a welded beam is crucial in the industry, in attempting to minimize the fabrication

ARPN Journal of Engineering and Applied Sciences

cost, to tally with the design specification, in addition to ensure the safety issue is addressed during the beam welding.

The main aim of the welded beam design problem is to minimize the fabrication cost, in which the four involved variables are: width of beam *d*, thickness of beam *w*, length of beam *l* and thickness of welding *h* (Figure-5).

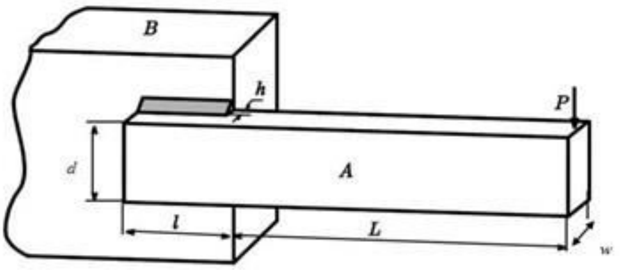

Figure-5. Welded beam design problem [18]

The objective function that represents the fabrication cost is formulated as:

$$\text{minimize } f(x) = 1.10471w^2 l + 0.04811dh(14.0 + l) \quad (16)$$

subject to the constraints of:

$$g_1(x) = w - h \le 0 \quad (17)$$

$$g_2(x) = \frac{65{,}856}{30{,}000hd^3} - 0.25 \le 0 \quad (18)$$

$$g_3(x) = \sqrt{\alpha^2 + \frac{\alpha\beta l}{D} + \beta^2} - 13600 \le 0 \quad (19)$$

$$g_4(x) = \frac{504{,}000}{hd^2} - 30000 \le 0 \quad (20)$$

$$g_5(x) = 0.10471w^2 + 0.04811hd(14 + l) - 5.0 \le 0 \quad (21)$$

$$g_6(x) = 0.125 - w \le 0 \quad (22)$$

$$g_7(x) = 6000 - 0.61423\times10^6 \frac{dh^3}{6}\left(1 - \frac{\sqrt[4]{30/48}}{28}\right) \le 0 \quad (23)$$

where

$$\alpha = \frac{6000}{\sqrt{2wl}} \quad (24)$$

$$\beta = \frac{QD}{J} \quad (25)$$

$$Q = 6000\left(14 + \frac{l}{2}\right) \quad (26)$$

$$D = \frac{1}{2}\sqrt{l^2 + (w+d)^2} \quad (27)$$

$$J = \sqrt{2wl\frac{l^2}{6} + \frac{(w+d)^2}{2}} \quad (28)$$

The design variables are given within the boundaries of $0.1 \le l, d \le 2.0$ and $0.1 \le w, h \le 10.0$. From the aforementioned constraints, $g_1(x)$ assures that the thickness of welding is less than the thickness of beam, while $g_2(x)$ ensures that the beam end deflection is less than the fixed total deviation. The constraint $g_3(x)$ examines the resultant maximum shear stress such that it is lower than the permissible shear stress in weld, while $g_4(x)$ is to ascertain that the produced maximum stress is lower than the allowable normal stress in beam. Since the length of beam and thickness of beam cannot be negative, the constraint $g_5(x)$ is developed for this purpose. Lastly, $g_6(x)$ evaluates the thickness of welding such that it is higher than the predefined minimum thickness, while $g_7(x)$ ensures that allowable bucking stress is not violated.

Table-2 summarizes the optimal welded beam design obtained by the BA, CSA and FPA, and their performance assessment in comparison to the available literature. The evolutionary strategy with the best welded beam design is given in bold font in this table.

Table-2 shows that the best result is given by the improved CSA (ICSA) approach, with the fabrication cost of 1.5757. On the other hand, the attained lowest fabrication cost of BA, CSA and FPA are 1.8973, 1.69277 and 1.7249, respectively. Although the results do not outperform the ICSA, the optimization abilities of the BA, CSA and FPA are satisfactory as they are comparable or even superior than most of the evolutionary algorithms reported in the literature. Moreover, it is pertinent to note that there is no improvement strategy as in the ICSA has been integrated, neither in BA, CSA nor FPA.

EXAMPLE 3: OPTIMIZATION OF PRESSURE VESSEL DESIGN

Figure-6 illustrates an example of pressure vessel design, in which a cylindrical pressure vessel with two hemispherical heads at both ends is concerned.

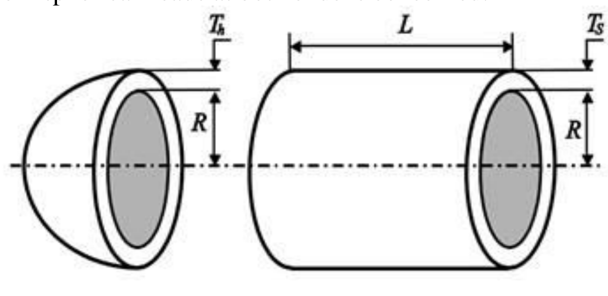

Figure-6. Pressure vessel design problem [18].

The pressure vessel is designed in accordance to the ASME boiler and pressure vessel code, where the working pressure and the minimum volume are assigned as 3000 psi and 750 ft^3, respectively. The main objective of this problem is to minimize the fabrication cost, which combines the welding cost, material and forming cost. There are four variables involved: thickness of pressure line T_s, thickness of the head T_h, inner radius R, and the length of the cylindrical section of the vessel L. The objective function, *i.e.*, the total cost is expressed as:

Table-1. Performance comparison of the BA, CSA and FPA for spring design with studies in literature.

Literature Review	Method	Diameter Wire, d	Mean Coil Diameter, w	Length, L	Weight, $f(x)$
[19]	Evolutionary Strategy	0.0516	0.3553	11.397	0.01270
[20]	**PSO**	0.051690	0.35675	11.28713	**0.012665**
[21]	T-Cell Algorithm	0.0516	0.3551	11.384	0.01267
[22]	Gaussian Quantum-PSO	0.0515	0.3525	11.538	0.01267
[23]	Direct Searching	0.05171	0.3573	11.2570	0.01267
[24]	Harmony Search	0.05171	0.3573	11.2553	0.01267
[25]	Mine Blast Algorithm	0.0516	0.3559	11.344	0.01267
[26]	Fish Swarm Optimization	0.05174	0.3578	11.5613	0.01279
[18]	Improved CSA	0.05170	0.3570	11.2670	0.01267
[27]	Modified BA	0.0519	0.3620	10.980	0.01267
This study	BA	0.0512	0.3455	11.9893	0.0127
This study	CSA	0.0537	0.4076	9.0217	0.01277
This study	FPA	0.051688	0.356695	11.29105	0.01267

Table-2. Performance comparison of the BA, CSA and FPA for welded beam design with studies in literature.

Literature Review	Method	Beam Thickness, w	Length, l	Beam Width, d	Welding Thickness, h	Fabrication Cost, f
[20]	PSO	0.2057	3.4705	9.0366	0.2057	1.7225
[21]	T-Cell Algorithm	0.2444	6.2186	8.2915	0.2444	2.3811
[23]	Direct Searching	0.2057	3.2534	9.0366	0.2057	1.6953
[28]	Discrete PSO	0.2500	1.7821	8.2500	0.1875	1.9553
[29]	Firefly Algorithm	0.2057	3.562	9.0414	0.2015	1.7312
[24]	Harmony Search	0.2085	3.2524	8.9760	0.2067	1.7070
[30]	Fuzzy GA	0.2057	3.4160	9.0139	0.2076	1.7162
[25]	Mine Blast Algorithm	0.2057	3.4704	9.0366	0.2057	1.7248
[26]	Fish Swarm Optimization	0.2100	3.4125	8.9100	0.2088	1.7318
[18]	**ICSA**	0.2317	1.6115	8.5241	0.2277	**1.5757**
[31]	Interior Search Algorithm	0.2443	6.2199	8.2915	0.2443	2.3812
[27]	Modified BA	0.2057	3.5620	9.0414	0.2015	1.7312
This study	BA	0.2172	3.5447	8.2801	0.2450	1.8973
This study	CSA	0.20573	3.23491	9.03661	0.20573	1.69277
This study	FPA	0.20573	3.4705	9.0366	0.20573	1.7249

Table-3. Performance comparison of the BA, CSA and FPA for vessel pressure design with studies in literature.

Literature Review	Method	Thickness of Pressure Vessel, T_s	Head Thickness Hemisphere, T_h	Inner Radius, R	Cylinder Length without Head, L	*Fabrication Cost, f*
[23]	Direct Searching	0.75	0.375	38.8601	221.36547	5850.3831
[24]	Harmony Search	0.7441	0.3678	38.5523	226.1553	5829.5475
[26]	Fish Swarm Optimization	0.8125	0.4375	42.0913	176.7466	6061.0778
[18]	**ICSA**	0.7389	0.3655	38.2793	230.5088	**5823.3730**
This study	BA	0.8088	0.3999	41.9067	185.3666	6087.0846
This study	CSA	0.8480	0.4163	43.5603	171.2527	5970.1795
This study	FPA	0.7782	0.3847	40.3199	199.9968	5885.363

VOL. 11, NO. 14, JULY 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

minimize

$$f(x) = 0.6224T_sRL + 1.7781T_hR^2 + 3.1661T_s^2L + 19.84T_h^2L \tag{29}$$

subject to the constraints of:

$$g_1(x) = -T_s + 0.0193R \leq 0 \tag{30}$$

$$g_2(x) = -T_h + 0.0954R \leq 0 \tag{31}$$

$$g_3(x) = -\pi R^2 L - \frac{4}{3}\pi R^3 + 1296000 \leq 0 \tag{32}$$

$$g_4(x) = L - 240 \leq 0 \tag{33}$$

where the ranges of these design variables are: $0 \leq T_s, T_h \leq 99$ and $10 \leq R, L \leq 200$.

The pressure vessel design is a benchmark structural optimization problem which has been studied widely. Numerous evolutionary optimization approaches, including harmony search and fish swarm optimization, have been applied in getting the optimal design of this problem, in attempting to minimize the fabrication cost. The performance evaluation of several approaches in comparison to the BA, CSA and FPA for the pressure vessel design is presented in Table-3. The best result with the lowest fabrication cost is given in bold font.

As it is seen, the best feasible solution is given by ICSA, with the cost of 5823.3760. However, it can be concluded that the BA, CSA and FPA are capable of effectively approaching to the optimal solution, as evident from the small relative differences in the obtained fabrication cost. The FPA, particularly, achieves the minimum cost of 5885.363, where the noticeable deviation from the best result is merely 1.06%.

Comparing the optimization abilities of the BA, CSA and FPA, it can be seen that all algorithms converge to the similar value in terms of T_s, T_h and R, albeit different evolutionary strategies are utilized in its diversification and intensification. The only large relative difference lies in the design variable L, where the BA, CSA and FPA converge to the optimal value of 185.3666, 171.2527 and 199.9968, respectively.

CONCLUSIONS

In this study, the BA, CSA and FPA are employed for solving constrained optimization problem. The optimization performances of these algorithms have been validated using several benchmark engineering problems, specifically, the spring design, welded beam design and pressure vessel design. The comparative study with the reported literature reveals that the BA, CSA and FPA are found to be efficient. The obtained results are comparable or even superior than most of the evolutionary algorithms.

On the other hand, from the simulations, it can be observed that the optimization strategy of the BA, perhaps, is the least satisfactory method among the three considered evolutionary algorithms. As shown in Table-1 to Table-3, the optimal values of the objective function achieved by the BA in all design problems are higher than the CSA and FPA. This is probably attributed to the fact that there more parameters to be fine-tuned in the BA, in comparison to the CSA and FPA. There is essentially one parameter needs to be inspected in the latter, while for BA, the parameters of loudness, impulse rate, maximum and minimum frequency have to be assigned judiciously for proper convergence towards the optimal solutions.

ACKNOWLEDGEMENT

Financial support from the Malaysian government with cooperation of Universiti Tun Hussein Onn (UTHM) in the form of FRGS Vot 1490 is gratefully acknowledged.

REFERENCES

[1] Ong P. and Zainuddin Z. 2013. An efficient cuckoo search algorithm for numerical function optimization. In: AIP Conference Proceedings 1522. pp. 1378-1384.

[2] Kurdi M. 2015. A new hybrid island model genetic algorithm for job shop scheduling problem. Computers and Industrial Engineering. 88: 273-283.

[3] Ong P., Zainuddin Z., Sia C.K. and Zain B.A.M. 2015. Adaptive Cuckoo Search Algorithm with Two-Parent Crossover for Solving Optimization Problems. In: Advances in Swarm and Computational Intelligence. pp. 427-435.

[4] Yang X.S. 2008. Nature-Inspired Metaheuristic Algorithms. First Edition. Luniver Press.

[5] Mittal P., Kulkarni K. and Mitra K. 2016. A novel hybrid optimization methodology to optimize the total number and placement of wind turbines. Renewable Energy. 86: 133-147.

[6] Comis D.R., Ponza R. and Benini E. 2015. Aerodynamic shape optimization of aircraft components using an advanced multi-objective evolutionary approach. Computer Methods in Applied Mechanics and Engineering. 285: 255-290.

[7] Xia S., Chan K.W., Bai X. and Guo Z. 2015. Enhanced particle swarm optimisation applied for transient angle and voltage constrained discrete optimal power flow with flexible AC transmission system. IET Generation, Transmission and Distribution. 9: 61-74.

[8] Yang C., Remenyte-Prescott R. and Andrews J.D. 2015. Pavement maintenance scheduling using genetic algorithms. International Journal of Performability Engineering. 11: 135-152.

[9] Yang X.-S. 2010. A new metaheuristic bat-inspired algorithm. In: Nature inspired cooperative strategies for optimization. pp. 65-74.

[10] Yang X.S. and Deb S. 2009. Cuckoo Search via Levy flights. In: World Congress on Nature & Biologically Inspired Computing. pp. 210-214.

[11] Yang X.S. 2012. Flower pollination algorithm for global optimization. In: Lecture Notes in Computer Science. pp. 240-249.

[12] Dubey H.M., Pandit M. and Panigrahi B.K. 2015. Hybrid flower pollination algorithm with time-varying fuzzy selection mechanism for wind integrated multi-objective dynamic economic dispatch. Renewable Energy. 83: 188-202.

[13] Emary E., Zawbaa H.M., Hassanien A.E., Tolba M.F. and Snášel V. 2014. Retinal Vessel Segmentation Based on Flower Pollination Search Algorithm. Advances in Intelligent Systems and Computing. 303: 93-100.

[14] Saji Y. and Riffi M.E. 2015. A novel discrete bat algorithm for solving the travelling salesman problem. Neural Computing & Application (In Press).

[15] Kavousi-Fard A., Niknam T. and Fotuhi-Firuzabad M. 2015. A Novel Stochastic Framework Based on Cloud Theory and θ-Modified Bat Algorithm to Solve the Distribution Feeder Reconfiguration. IEEE Transactions on Smart Grid. 99: 1-1.

[16] Wang Z. and Li Y. 2015. Irreversibility analysis for optimization design of plate fin heat exchangers using a multi-objective cuckoo search algorithm. Energy Conversion and Management. 101: 126-135.

[17] Kumar M. and Rawat T.K. 2015. Optimal design of FIR fractional order differentiator using cuckoo search algorithm. Expert Systems with Applications. 42: 3433-3449.

[18] Bulatović R.R., Bošković G., Savković M.M. and Gašić M.M. 2014. Improved Cuckoo Search (ICS) algorthm for constrained optimization problems. Latin American Journal of Solids and Structures. 11: 1349-1362.

[19] Mezura-Montes E. and Coello C.A.C. 2008. An empirical study about the usefulness of evolution strategies to solve constrained optimization problems. International Journal of General Systems. 37: 443-473.

[20] Cagnina L.C., Esquivel S.C. and Coello C.A.C. 2008. Solving Engineering Optimization Problems with the Simple Constrained Particle Swarm Optimizer. 32: 319-326.

[21] Aragón V.S., Esquivel S.C. and Coello C.A.C. 2010. A modified version of a T-Cell Algorithm for constrained optimization problems. International Journal for Numerical Methods in Engineering. 84: 351-378.

[22] Coelho L.d.S. 2010. Gaussian quantum-behaved particle swarm optimization approaches for constrained engineering design problems. Expert Systems with Applications. 37: 1676-1683.

[23] Zou D., Liu H., Gao L. and Li S. 2011. Directed searching optimization algorithm for constrained optimization problems. Expert Systems with Applications. 38: 8716-8723.

[24] Mun S. and Cho Y.-H. 2012. Modified harmony search optimization for constrained design problems. Expert Systems with Applications. 39: 419-423.

[25] Sadollah A., Bahreininejad A., Eskandar H. and Hamdi M. 2013. Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems. Applied Soft Computing, 13: 2592-2612.

[26] Lobato F.S. and Steffen Jr V. 2014. Fish swarm optimization algorithm applied to engineering system design. Latin American Journal of Solids and Structures. 11: 143-156.

[27] Yılmaz S. and Küçüksille E.U. 2015. A new modification approach on bat algorithm for solving optimization problems. Applied Soft Computing. 28: 259-275.

[28] Datta D. and Figueira, J.R. 2011. A real-integer-discrete-coded particle swarm optimization for design problems. Applied Soft Computing. 11: 3625-3633.

[29] Gandomi A.H., Yang X.S. and Alavi A.H. 2011. Mixed variable structural optimization using Firefly Algorithm. Computers & Structures. 89: 2325-2336.

[30] Ashtari P. and Barzegar F. 2012. Accelerating fuzzy genetic algorithm for the optimization of steel structures. Structure Multidiscipline Optimization. 45: 275-285.

[31] Gandomi A.H. 2014. Interior search algorithm (ISA): A novel approach for global optimization. ISA Transactions. 53: 1168-1183.