Accepted by “Engineering Optimization”

Biogeography-Based Particle Swarm Optimization with Fuzzy Elitism and Its Applications to Constrained Engineering Problems

Weian Guo[1,2,5], Wuzhao Li[1,4], Qun Zhang[3], Lei Wang[1], Qidi Wu[1], Hongliang Ren[2]

1. School of Electronics and Information, Tongji University, Shanghai, 201804, China
2. Department of Bioengineering, National University of Singapore, 117575, Singapore
3. NUS Graduate School for Integrative Sciences and Engineering, National University of Singapore, 117456, Singapore
4. Jiaxing Vocational Technical College, Zhejiang, Jiaxing, 314036, China
5. Social Robotics Laboratory, Interactive Digital Media Institute, and Department of Electrical and Computer Engineering, National University of Singapore, 119077, Singapore

Abstract: In Evolutionary Algorithms (EA), elites are crucial to maintain good features in solutions. However, too many elites could make the evolutionary process stagnate and cannot enhance the performance. In this paper, we employ Particle Swarm Optimization (PSO) and Biogeography-Based Optimization (BBO) to propose a hybrid algorithm termed Biogeography-Based Particle Swarm Optimization (BPSO) which could make a large number of elites effective in searching optimum. In this algorithm, the whole population is split into several subgroups, which BBO is employed for intra-group and PSO is employed for inter-groups. Since not all the population is used in PSO, this structure overcomes the premature in original PSO. By the time complexity analysis, the novel algorithm will not increase the time consuming. Fourteen numerical benchmarks and four engineering problems with constraints are used to test BPSO. To better deal with constrains, a fuzzy strategy for number of elites is investigated. The simulations results validate the feasibility and effectiveness of the proposed algorithm.

Keywords: Evolutionary Algorithms, elites, Biogeography-Based Optimization, Particle Swarm Optimization, Fuzzy Strategy

1. Introduction

As a branch of artificial intelligence, Evolutionary Algorithms (EAs) play an important role. It uses iterative progress of competitive and random variation based on evolution of population to solve optimization problems. Starting from 1950s and 1960s, many independent efforts had been made for evolutionary algorithms [1,2,3,4]. In contrast to conventional numerical optimization methods, EAs have the advantages such robustness, reliability and little information requirement. Especially, for non-deterministic polynomial (NP) problems, such as Travelling Salesman Problem [5], N-Queens [6], Bin packing [7] and etc., EAs perform much better to obtain satisfactory solutions while the conventional methods cannot achieve the global optimum.

During the past decades, various kinds of evolutionary algorithms inspired from natural phenomena [8,9] were proposed, including Genetic Algorithm (GA) [10,11], Ant Colony Optimization (ACO) [12] and Particle Swarm Optimization (PSO) [13,14,15,16]. Most of EAs are. In recent years, many novel algorithms are also proposed subsequently and implemented successfully in practical applications [17,22]. To combine the advantages in different algorithms, hybrid evolutionary algorithms have been a hot topic in the field of optimization [18,19]. It exhibits a huge ability in optimization and has been well implemented in sciences, engineering, medicine and etc.

Among Evolutionary Algorithms, PSO achieves a great success in optimization. It makes use of birds’ flocking behavior, which each bird tends to improve itself by emulating other better members. The optimization procedure mainly relies on the update mechanism of velocity and position. Hence PSO only needs a few parameters including velocity and position needed, which make this approach easy to implement. However, a drawback in standard PSO is that the swarm may prematurely converge. The reason is that PSO particles converge to a single point which is even not guaranteed as a local optimum [20], let alone a global optimum [21]. To overcome this drawback, a hierarchical structure is proposed and another optimization algorithm Biogeography-Based Optimization (BBO) [22,23,24] is employed to hybridize with PSO. The hybrid algorithm is termed Biogeography-based Particle Swarm Optimization (BPSO). In BPSO, the whole population is split into several subgroups, BBO is used for group-inside search in each subgroup and select elites by fuzzy strategy. The selected elites are used for global search by PSO.

BBO is novel evolutionary algorithm inspired by species. This approach has been investigated not only in theory [25,26,27,28] but also in applications [29, 30]. Simon compared this algorithm with Genetic Algorithm by Markov analysis [25, 31]. Ma studied the performance of BBO by numerical simulation to show the dominant ability in several classical algorithms [32]. Guo et al. investigated migration models of BBO mathematically and the conclusions are helpful to design migration models [33]. The reason to employ BBO is given as follows. For PSO, the information fluid is unidirectional [34], which the global best solution will influence others while others rarely influence the best solution. Hence PSO could lead to a local optimum. However, the information fluid in BBO is global so that each individual can influence others in a probabilistic way. Hence BBO could increase the information fluidity in BPSO.

In addition, elites play a most important role in Evolutionary Algorithms. However, a huge number of elites could make the algorithms stagnate due to the premature. Since BPSO increase the information fluidity in population, it will help algorithms with a huge number of elites avoid stagnation and give a full play of all the elites.

The remainder of this paper is organized as follows. Section 2 introduces BBO and PSO briefly. Section 3 presents the mechanism of BPSO and fuzzy elitism. The numerical simulation and relevant analysis are shown in Section 4. The time complexity analysis is discussed in Section 5 and BPSO is applied to several classic constrained engineering optimizations in Section 6. To solve the constrained optimization problems, an adaptive penalty function is employed. The conclusion and future work are presented in Section 7.

2. Preliminary of BBO and PSO

2.1 Biogeography-Based Optimization

Biogeography-based Optimization is inspired from the science of Biogeography. The habitats in biogeography are analogous to the solutions in optimization problems. The suitability of one island for living is evaluated by High Suitability Index (HSI). The variables that characterize habitability are called Suitability Index Variables (SIVs). In optimization problems, a good solution is analogous to an island with a high HSI, while a poor solution represents an island with a low HSI. An island with high HSI has more species than an island with low HSI. This means the immigration to a good island is difficult for the aliens. Therefore, the emigration is dominant in good islands. In contrast, few species live in poor islands. Hence immigration is dominant in poor islands. Based on this idea, good solutions tend to share their features with poor solutions while poor solutions tend to accept features from good solutions.

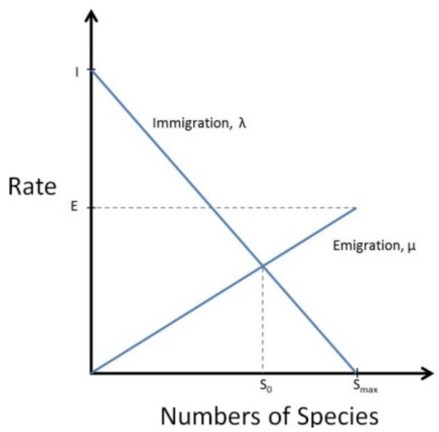


Figure 1 Species model of a single habitat

Figure 1 illustrates a linear model of species distribution in a single habitat [22]. The researches on migration models have been investigated in [25, 33]. In this linear migration model, I is the maximum possible immigration rate and E represents the maximum possible emigration rate. The immigration rate and the emigration rate are functions of the number of species in the habitat. For an island, the immigration rate increases and the emigration rate decreases as the number of species increases. The pseudo-codes of migration are shown in Figure 2.

Algorithm for BBO Migration
Select H_i according to immigration rate λ_i

If H_i is selected

For $j=1$ to n

Select H_j according to emigration rate μ_i

If H_j is selected

Replace SIV in H_i with SIV in H_j

End

End

End

Figure 2 Pseudo-codes of Biogeography-Based Optimization Migration

Besides migration, BBO has the probabilistic property as most evolutionary algorithms do. Mutations may occur during the progress of iteration with certain probability. The pseudo-codes of mutation are shown as follows,

Algorithm for BBO Mutation

For $j=1$ to m

Use λ_i and μ_i to compute the probability P_i

Select SIV $H_i(j)$ with probability P_i

If $H_i(j)$ is selected

Replace $H_i(j)$ with a randomly generated SIV

End

End

Figure 3 Pseudo-codes of Biogeography-Based Optimization Mutation

2.2 Particle Swarm Optimization

Particle Swarm Optimization (PSO) was proposed by Kennedy and Eberhart [35] in 1995 and has been investigated theoretically and practically [36,37,38], where birds flocking simulated around food resources. In PSO, the population is represented by particles, which move around in a search space according to particles’ position and velocity. The position and velocity of the particles will be updated during iterations. The current velocity of each particle is calculated based on three factors: its previous velocity, the distance from its previous best position and the distance from the global best position [39]. The updated position is calculated based on current position and velocity. The mathematical formulas are shown as follows,

$$v_{id}(t)=v_{id}(t-1)+c_1\varphi_1\left(p_{id}-x_{id}(t-1)\right)+c_2\varphi_2\left(p_{gd}-x_{id}(t-1)\right) \tag{1}$$

$$x_{id}(t)=x_{id}(t-1)+v_{id}(t) \tag{2}$$

where $v_{id}(t)$ is the velocity of the dth dimension of the ith particle at the tth generation,

$x_{id}(t)$ presents the current position of d[th] dimension of the i[th] particle at the t[th] generation, p_{id} is the i[th] individual's best position found so far at d[th] dimension, p_{gd} is the global best position found so far at d[th] dimension, φ_1 and φ_2 are two random values in interval (0,1), c_1 and c_2 are learning factors. The pseudo-codes of PSO are shown in Figure 4, where m is the population size, G_p is the own best known fitness, G_g is the entire swarm's best known fitness, P_p is the own best known position and P_g is the entire swarm's best known position.

```
Algorithms for Particle Swarm Optimization
While stopping criteria is not met
    For j = 1 to m
        For each individual j, Evaluate the fitness G_j
        If G_i is better than G_p          %Update the local optimum
            P_p = P_j
        End
        If G_i is better than G_g          %Update the global optimum
            P_g = P_j
        End
        Update the velocity and position of individual j
    End
End
```

Figure 4 Pseudo-codes of Particle Swarm Optimization

3. Biogeography-based Particle Swarm Optimization with Fuzzy Elitism

In this section, a novel algorithm termed Biogeography-based Particle Swarm Optimization is proposed based on the idea of BBO and PSO. In BBO, species migrate from one habitat to another. However, in nature, it is not possible that each kind of species has the ability to migrate everywhere. A part of species rely on other species to migrate. An example is that a seed of a plant may be eaten by a bird. The bird migrates to other islands and excretes the seed. After that, the plant completes migration. In addition, according to the evolutionary mechanism, PSO rarely share information from poor solutions to good solutions, while all individuals in BBO can do this. Hence the BPSO can help make use of their respective advantages.

By assuming that there are only strong individuals who have the ability to migrate globally and the other individuals migrate locally, the population is split into several subgroups. In each subgroup, BBO is used for local search and obtain local elites by fuzzy set. Then the local elites can be composed as the initial population of PSO for global search.

In BPSO, fuzzy set is used to confirm the amount of elites due to the following reasons. The fixed number elites may cost a waste of searching time. At the beginning of evolution, since the quantity of each individual is lower, a huge number of elites will lead to stagnation. Hence, in this stage, only a few individuals will be selected as elites for population in PSO. However, at the later period, the quantities of individuals are so high that a small number of elites cannot afford the high precision searching. Thus, the number of elites to compose initial population in PSO cannot be too small. To quantify the amount of elites selected by BBO, fuzzy strategy is employed. First, the whole process is split into several segments, where the interval of each segment is m generations. At the beginning of each segment, the best fitness is recorded as a_0, while at the end of the segment, the best fitness is recorded as a_1. Then the slope is calculated as follows,

$$slop = \frac{a_1 - a_0}{m} \tag{3}$$

And the percent to be chosen is set as 30%, 40% and 50% respectively. As shown in Figure 5 The membership function of the probability is defined as in (4), (5), (6). $P_1(X)$ is for 30%, $P_2(X)$ for 40%, $P_3(X)$ for 50%. In algorithm, the initial slop can be predefined by users from generation 1 to generation m.

$$P_1(X) = \begin{cases} 1, \ if \ \ 0 \le X < \frac{\pi}{8} \\ \frac{8}{\pi}\left(-X + \frac{\pi}{4}\right), \ if \ \ \frac{\pi}{8} \le X < \frac{\pi}{4} \\ 0, \ otherwise \end{cases} \tag{4}$$

$$P_2(X) = \begin{cases} \frac{8}{\pi}\left(X - \frac{\pi}{8}\right), \ if \ \ \frac{\pi}{8} \le X < \frac{\pi}{4} \\ \frac{8}{\pi}\left(-X + \frac{3\pi}{8}\right), \ if \ \ \frac{\pi}{4} \le X < \frac{3\pi}{8} \\ 0, \ otherwise \end{cases} \tag{5}$$

$$P_3(X)=\begin{cases}\frac{8}{\pi}\left(X-\frac{\pi}{4}\right), & if\ \frac{\pi}{4}\le X<\frac{3\pi}{8}\\ 1, & if\ \frac{3\pi}{8}\le X<\frac{\pi}{2}\\ 0, & otherwise\end{cases} \tag{6}$$

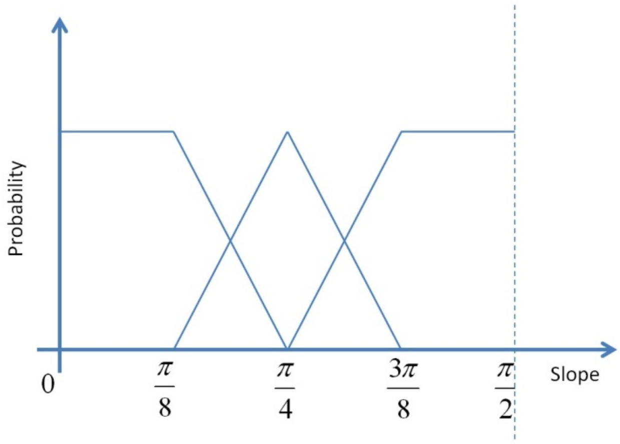


Figure 5The figure of membership function

The schedule of BPSO is described as follows and the flowchart is given in Figure 6.

Step 1. For PSO, the inertia weight is set as ω , learning factors are c_1 and c_2. For BBO, the maximum migration rate is 1. The migration model is linear. The mutation rate is 0.01. The limit generation is l and the segment in fuzzy strategy is m. The initial slop is r , the population size is n, and the number of subgroups is s.

Step 2. The fitness of each individual is calculated and rank the individuals in each subgroup according to the fitness.

Step 3. In each subgroup, BBO is used for local search. The emigration rate and the immigration rate for each individual are calculated based on the migration model. Individuals migrate in each group according to the migration rates.

Step 4. In each subgroup, the elites are selected by fuzzy set. calculate the amount of elites for global search by PSO. The number of elites is set as k. After m generation, the slop can be calculated by (3). Then according to (4), (5) and (6) the probabilities of elite percent can be obtained. After that, the value of k can be calculated by fuzzy set.

Step 5. Choose the top ranking k individuals as elites in each subgroup. For the total $s\times k$ elites, the best velocity and position of each individual and record the global best solution are recorded. Then according to formula (1) and (2), velocity and position can be updated.

Step 6. Record the best solution as the global solution and update the each individual's best velocity and position.

Step 7. All the elites return to the original subgroups respectively.

Step 8. If the stopping criteria are met, then end the optimization search. Otherwise go to Step 2.

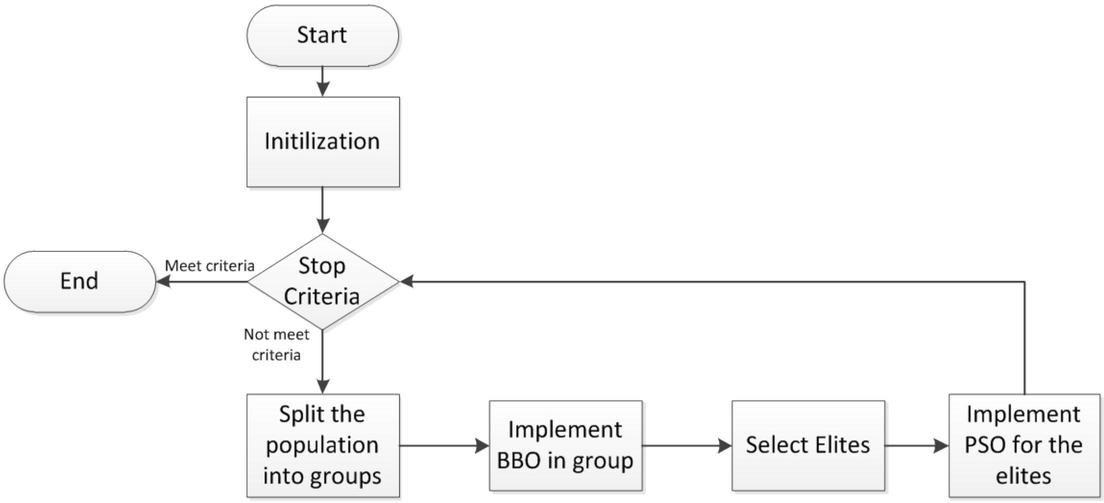


Figure 6 Flowchart of BPSO

4. Analysis of Time Complexity

In this section, the time complexity of the BPSO is analyzed. First, for PSO, suppose that there are N individuals in the population. According to the description in Figure 4, the time complexity of PSO is $O(N)$. Second, for BBO, suppose that there are N islands. According to the design of BBO [22], the worst time complexity of BBO is $O(N^2)$ and the best time complexity of BBO is $O(N)$.

For BPSO, suppose there are N individuals and M groups. There are γ percent of population are considered as elites. Then in BPSO, there are N/M individuals for BBO and γN individuals for PSO. Hence the worst time complexity is

$$O\left((N/M)^2\right)+O(\gamma N)\rightarrow O\left(N^2\right) \tag{7}$$

The best time complexity is

$$O(N/M)+O(\gamma N)\rightarrow O(N) \tag{8}$$

To sum up, the time complexity of PSO, BBO and BPSO are shown in Table 1. In the table, the best time complexities of the three algorithms are the same. For the worst time complexity, BPSO is as same as BBO. Hence the BPSO will not increase the time complexity.

Table 1 Time complexity of PSO, BBO and BPSO

Time Complexity	PSO	BBO	BPSO
Best	$O(N)$	$O(N)$	$O(N)$
Worst	$O(N)$	$O(N^2)$	$O(N^2)$

5. Numerical Simulation and Analysis

5.1 Benchmark Functions and Simulation Results

In this section, the performance of BPSO in 14 well known benchmarks [40,41,42] is investigated. The details about the benchmarks are given in Table 2. In addition, the property of these benchmarks is shown in Table 3.

Table 2 Benchmark Functions

Function	Name	Dimension	Domain
F1	Ackley’s Function	20	$[-30,30]^D$
F2	Fletcher-Powell	20	$[-\pi, \pi]^D$
F3	Generalized Griewank’s function	20	$[-600,600]^D$
F4	Generalized Penalized function 1	20	$[-50,50]^D$
F5	Generalized Penalized function 2	20	$[-50,50]^D$
F6	Quartic function	20	$[-1.28,1.28]^D$
F7	GeneralisedRastrigin’s function	20	$[-5.12,5.12]^D$
F8	Generalized Rosenbrock’s function	20	$[-2.048,2.048]^D$
F9	Schwefel’ Problem 1.2	20	$[-65.535,65.535]^D$
F10	Schwefel’ Problem 2.21	20	$[-100,100]^D$
F11	Schwefel’ Problem 2.22	20	$[-10,10]^D$
F12	Schwefel’ Problem 2.26	20	$[-512,512]^D$
F13	Sphere Model	20	$[-5.12,5.12]^D$
F14	Step Function	20	$[-200,200]^D$

Table 3 Property of benchmarks

Function	Multimodal	Separable	Regular
Ackley’s Function	√		√
Fletcher-Powell	√		
Generalized Griewank’s function	√		√
Generalized Penalized function 1	√		√
Generalized Penalized function 2	√		√
Quartic function		√	√
GeneralisedRastrigin’s function	√	√	√
Generalized Rosenbrock’s function			√
Schwefel’ Problem 1.2			√
Schwefel’ Problem 2.21			
Schwefel’ Problem 2.22	√		
Schwefel’ Problem 2.26	√	√	

Sphere Model	√	√
Step Function	√	

First, the performances of optimization method with different number of elites are compared to investigate the impact of large number of elites on the performance. The simulation results are shown in **Error! Reference source not found.**. For PSO, the two learning factors are both set as 1.296 and the inertial weight is 0.726. For BBO, a linear migration model is employed. The maximum migration rate is 1 and mutation rate is 0.01. The parameters of BPSO are set as same as in BBO and PSO. For each algorithm, the population size is set as 100 and the maximum generation is 1000. For each benchmark, 100 Monte Carlo simulations of each algorithm are conducted to obtain an average performance. PSO-50 means there are 50 elites (half of the population) in PSO. PSO-2 means there are only 2 elites in PSO. BBO-50 means there are 50 elites (half of the population) in BBO. BBO-2 means there are 2 elites in BBO. The comparison results are shown in **Error! Reference source not found.**, where “BEST” records the best optimum in 100 simulations and “MEAN” records the average optimal value in 100 simulations.

In Table 4, it is obvious that increasing the number of elites cannot help algorithms enhance the performance. Compared PSO-50 with PSO-2, for all 14 benchmarks, PSO-50 is inferior to PSO-2 in “BEST” volume and wins for F1, F2, F5, F8, F10, F13, F14, totally 7 benchmarks, in “MEAN” volume. Compared with BBO-50 with BBO-2, BBO-50 only wins F8, F9, F10, F12 in “BEST” volume and is inferior to BBO-2 for all 14 benchmarks in “MEAN” volume. The results shows that, increasing the number of elites cannot help algorithms enhance performance. Oppositely, a large number of elites will be harmful to the performance. However, the performances of BPSO are superior to all of others only except the “MEAN” values in F6, F7 and F14. Hence it can be concluded that although BPSO increase the number of elites, it could exploit the elites to play an invasive role in optimization, which show the design of BPSO is effective and successful.

Table 4 Comparisons of Performance with different number of elites

F	PSO-50		PSO-2		BBO-50		BBO-2		BPSO	
	BEST	MEAN	BEST	MEAN	BEST	MEAN	BEST	MEAN	BEST	MEAN
F1	1.1818E+01	8.2750E+00	4.5664E-01	9.6385E+00	0.0000E+00	7.8612E+00	0.0000E+00	1.2250E-01	0.0000E+00	9.9583E-14
F2	2.6665E+04	1.2539E+04	1.6238E+03	2.1071E+04	1.9403E+02	9.9175E+03	7.4832E+01	1.0850E+03	1.3846E-26	1.0429E+03
F3	1.4390E+01	9.7031E+00	1.0026E+00	6.6945E+00	1.0002E+00	3.9544E+00	1.0001E+00	1.0005E+00	1.0000E+00	1.0000E+00
F4	1.7030E+04	1.5317E+01	5.7211E-03	2.4697E+01	6.5450E-05	3.9231E+00	1.5705E-32	1.0230E-03	1.5705E-32	1.5705E-32
F5	6.0761E+05	6.8459E+03	3.8360E-02	4.3916E+04	2.0000E-03	3.7861E+01	1.3498E-32	6.2922E-03	1.3498E-32	2.2000E-03
F6	1.1059E-01	3.5794E-02	2.1100E-07	3.1193E-02	0.0000E+00	1.1893E-03	0.0000E+00	0.0000E+00	0.0000E+00	1.3613E-63
F7	4.8289E+01	3.8586E+01	0.0000E+00	3.8483E+01	0.0000E+00	1.7048E+01	0.0000E+00	0.0000E+00	0.0000E+00	5.9700E-02
F8	8.2160E+01	3.9849E+01	6.7730E+00	4.2502E+01	3.4596E+00	2.5859E+01	6.0693E+00	3.4596E+00	2.3863E-29	7.8634E-20
F9	1.1740E+03	1.1479E+03	5.1587E-01	1.0354E+03	3.8241E-02	1.0381E+03	1.6882E-01	2.1108E-02	1.2728E-04	1.2728E-04
F10	1.1705E+03	4.7333E+02	1.7763E+01	6.2849E+02	2.8131E+00	2.7383E+02	6.6400E+00	4.6051E-01	0.0000E+00	9.2088E-28
F11	9.5841E+00	6.9292E+00	1.3333E-02	5.4078E+00	0.0000E+00	4.3210E+00	0.0000E+00	0.0000E+00	0.0000E+00	1.4211E-16

F12	2.1289E+01	1.5337E+01	4.8667E-01	1.3925E+01	2.0000E-01	1.0838E+01	6.5667E-01	2.0000E-01	1.8242E-10	2.1364E-04
F13	3.7292E+00	1.0865E+00	5.0393E-03	1.8526E+00	0.0000E+00	1.0083E+00	0.0000E+00	0.0000E+00	0.0000E+00	1.5777E-32
F14	1.4407E+03	6.6200E+02	0.0000E+00	7.2817E+02	0.0000E+00	4.2800E+02	0.0000E+00	0.0000E+00	0.0000E+00	6.0000E-02

Second, BPSO is compared with other classic evolutionary algorithms including GA [10], PBIL [43], ACO [12], PSO [13], DE [44], ES [1, 3] and BBO [22]. For GA, roulette wheel selection and single point crossover are employed. The mutation rate is set as 0.01. For PBIL, the learning rate and mutation rate are set as 0.05 and 0 respectively. For ACO, the initial pheromone value is 1e-6, the update constant being 20 and the pheromone evaporate rate being 0.1. The pheromone sensitivity is 1, visibility sensitivity being 5 and heuristic factor being 1. For PSO, the two learning factors are both set as 1.296 and the inertial weight is 0.726. For DE, the weighting factor and a crossover constant are both set as 0.5. For ES, there are 10 offspring in each generation and a standard deviation is set as 1 for changing solutions. For BBO, a linear migration model is employed. The maximum migration rate is 1 and mutation rate is 0.01. The parameters of BPSO are set as same as in BBO and PSO. The account of subgroups of BPSO is 2 and in each subgroup, half population is considered as elites. For each algorithm, the population size is set as 100. The population is split into 2 groups and the limitation generation is set as 1000. For each benchmark, 100 Monte Carlo simulations of each algorithm are conducted to obtain an average performance. The simulation results are shown in Table 5 and Table 6.

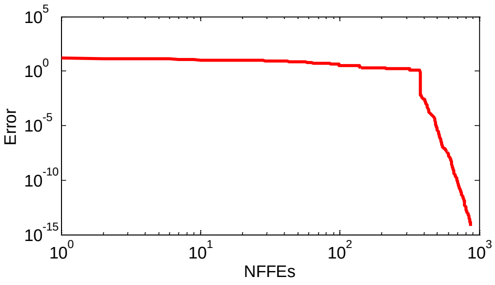
Table 5 Comparisons of Average Performance

MEAN	GA	PBIL	PSO	BBO	BPSO	ACO	DE	ES
F1	6.9608E+00	1.8565E+01	1.4364E+01	1.0080E-01	**9.9583E-14**	5.8940E+00	1.1036E-10	8.8717E+00
F2	2.1971E+04	3.4424E+05	3.3274E+05	1.0911E+03	**1.0429E+03**	4.2977E+05	3.3541E+04	5.5153E+05
F3	1.2836E+00	1.7142E+02	5.1664E+01	1.0006E+00	**1.0000E+00**	1.1508E+00	1.0000E+00	9.7999E+01
F4	5.4000E-02	3.9213E+07	1.1654E+06	1.3000E-03	**1.5705E-32**	9.0719E+07	3.0804E-20	3.9527E+07
F5	6.0420E-01	1.1791E+08	8.1410E+06	6.0000E-03	2.2000E-03	1.7821E+08	**1.3414E-19**	1.1274E+08
F6	1.9270E-06	1.0906E+01	1.4892E+00	2.0000E-10	**1.3613E-63**	3.4000E-03	1.7750E-35	1.4652E+01
F7	2.6268E+01	1.9976E+02	1.3706E+02	7.0800E-02	**5.9700E-02**	7.3926E+01	4.9991E+01	2.1786E+02
F8	3.8829E+01	1.2917E+03	3.4295E+02	5.7873E+00	**7.8634E-20**	8.4145E+02	1.3652E+01	2.4847E+03
F9	4.4959E+01	4.2772E+03	3.6341E+03	1.9230E-01	**1.2728E-04**	3.1628E+01	8.5879E+02	2.9441E+03
F10	2.8954E+03	9.8219E+03	5.4146E+03	7.9055E+00	**9.2088E-28**	1.9605E+03	1.5025E+03	1.2854E+04
F11	3.8860E+00	5.3668E+01	2.7329E+01	1.3700E-01	**1.4211E-16**	2.2011E+01	7.1620E-12	7.5355E+01
F12	1.8590E+01	5.9016E+01	3.5233E+01	7.5000E-01	**2.1364E-04**	2.0203E+01	7.1000E-03	2.1030E+01
F13	9.2300E-02	5.3076E+01	1.5228E+01	2.4000E-03	**1.5777E-32**	8.0228E+00	2.7517E-22	6.7719E+01
F14	1.3000E+01	1.9146E+04	5.6690E+03	1.7100E+00	6.0000E-02	2.1090E+01	**0.0000E+00**	1.5877E+04

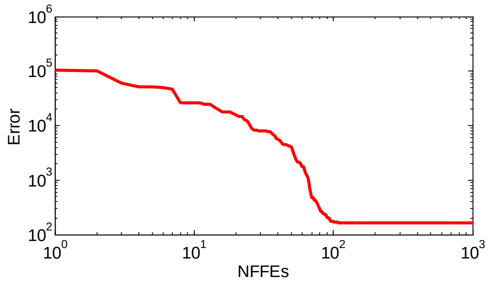
Table 6 Comparisons of Best Performance

BEST	GA	PBIL	PSO	BBO	BPSO	ACO	DE	ES
F1	3.4992E+00	1.6660E+01	1.0874E+01	2.8000E-01	**8.8818E-16**	4.0479E+00	3.3277E-11	6.6575E+00
F2	1.8292E+03	1.8433E+05	1.8692E+05	1.4380E+03	**1.3846E-26**	2.6176E+05	2.1049E+02	1.9665E+05
F3	1.0625E+00	1.0218E+02	2.6322E+01	1.0052E+00	**1.0000E+00**	1.0517E+00	**1.0000E+00**	5.9660E+01
F4	4.1000E-03	4.3454E+06	5.2511E+04	1.1000E-03	**1.5705E-32**	2.2930E-01	1.5951E-21	2.6615E+06
F5	1.2280E-01	2.4029E+07	1.1037E+06	2.1700E-02	**1.3498E-32**	1.3498E-32	1.2231E-20	1.9057E+07
F6	4.5000E-07	4.6795E+00	3.8030E-01	**0.0000E+00**	**0.0000E+00**	1.3000E-03	4.0815E-37	4.8652E+00
F7	1.1434E+01	1.7439E+02	1.0888E+02	**0.0000E+00**	**0.0000E+00**	5.1806E+01	3.3332E+01	1.6970E+02
F8	9.8404E+00	4.0387E+02	1.7236E+02	3.3268E+00	**2.3863E-29**	3.9377E+02	1.2220E+01	1.0282E+03
F9	3.8333E+00	3.1352E+03	2.6831E+03	1.3277E+00	**1.2728E-04**	9.8516E+00	3.9502E+02	2.3009E+03
F10	1.0062E+03	5.5851E+03	3.0947E+03	1.0832E+02	**0.0000E+00**	4.6161E+02	4.8517E+02	7.3296E+03
F11	1.4000E+00	4.3900E+01	1.4803E+01	**0.0000E+00**	**0.0000E+00**	5.7000E+00	2.4656E-12	4.8024E+01
F12	7.0000E+00	4.2600E+01	2.7504E+01	2.2000E+00	**1.8242E-10**	6.4000E+00	3.2000E-03	1.2877E+01
F13	1.0100E-02	2.9359E+01	8.6434E+00	**0.0000E+00**	**0.0000E+00**	2.7313E+00	3.2117E-23	3.5910E+01
F14	1.0000E+00	1.2695E+04	2.6990E+03	**0.0000E+00**	**0.0000E+00**	9.0000E+00	**0.0000E+00**	1.0164E+04

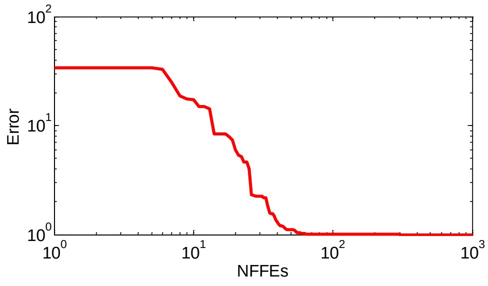
Table 5 shows the average performance of each algorithm on the 100 Monte Carlo simulations. Table 6 shows the best outcome of each algorithm on the 100 Monte Carlo simulations. In both two tables, the best solutions are marked by bold type. In Table 5, it lists the average performance of 9 algorithms on each benchmark, which could show the general ability of optimization. It is obvious that BPSO performs the best over 12 benchmarks, which is about 90 percent of total benchmarks. For F5 and F14, although the performance of BPSO is inferior to that of DE, this BPSO performs the second best. To sum up, BPSO is successful to take the advantages of BBO and PSO accordingly because it performs better than majoring of other classic optimization methods. In addition, according to the results, it is obvious that the performances of other algorithms are far behind BPSO. Table 6 compares the best solution of each algorithm on each benchmark. According to the results, it is obvious that BPSO performs the best for overall 14 benchmarks. Especially for F2, F4, F5 and F8, the superiority is remarkable. BPSO shares the throne with BBO for F6, F7, F11, F13, F14 and with DE for F3, F14.Forthe other benchmarks, the superiority of BPSO is remarkable as well. Hence, BPSO is competent to solve optimization problems. The convergence trend of BPSO for the above fourteen benchmarks is given in Figure 7.



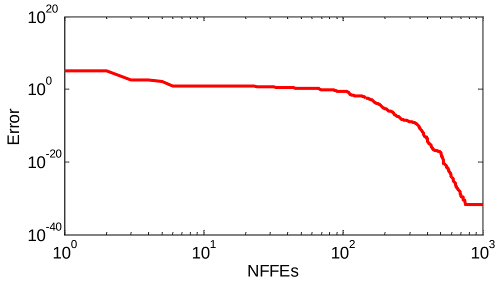
Ackley



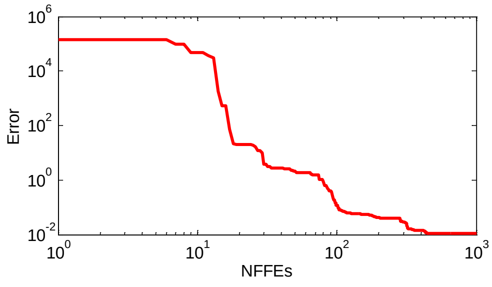
Fletcher



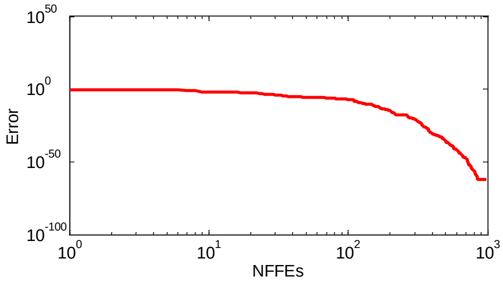
Griewank



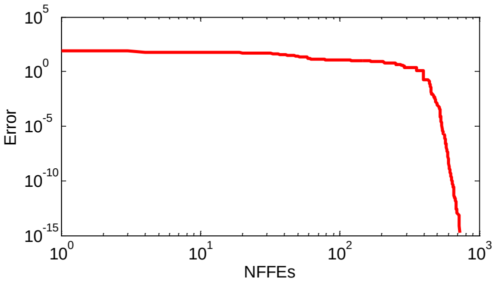
Penalty1



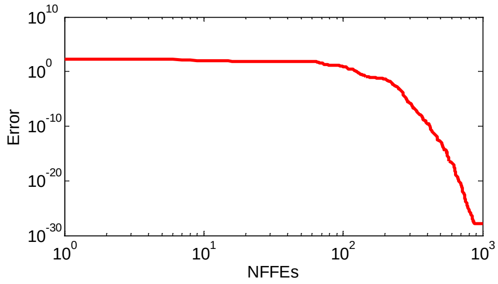
Penalty2



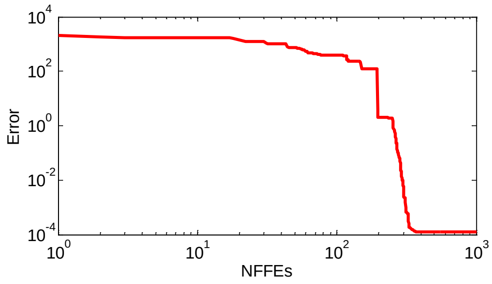
Quartic



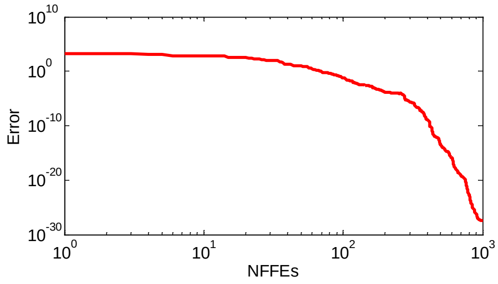
Rastrigin



Rosenbrock



Schwefel



Schwefel2

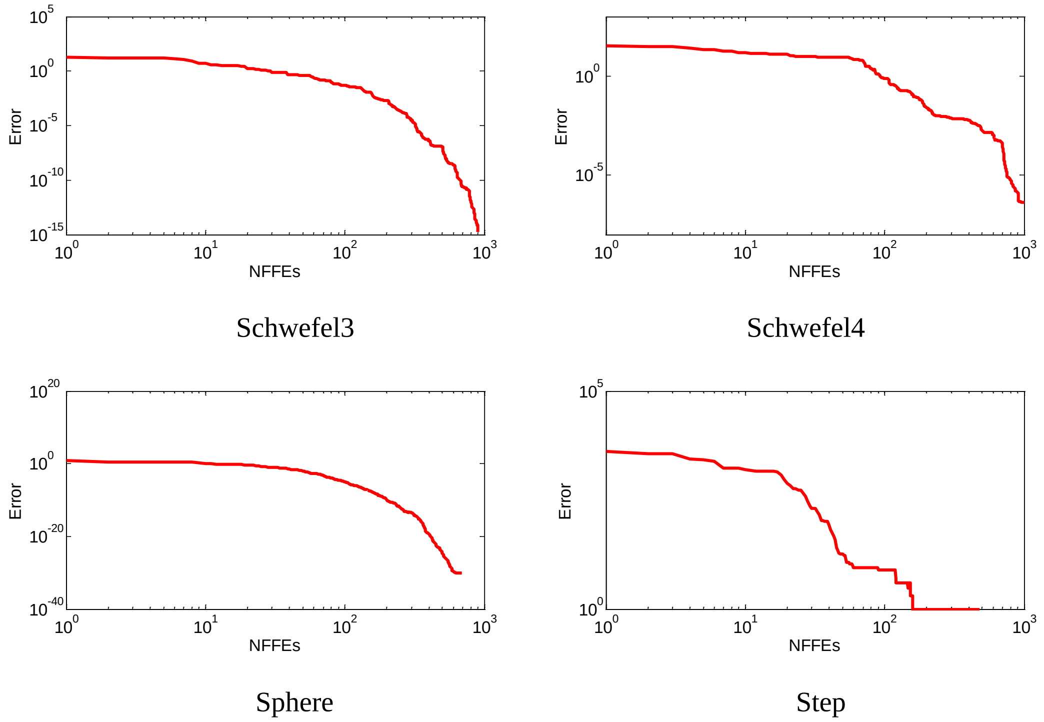


Figure 7 Mean error curves of BPSO for selected functions. ‘‘NFFEs” denotes the number of fitness function evaluations.

6. Optimization in Engineering Problems with Constraints

In this section, BPSO is applied to several classical engineering optimization problems with constraints. The results are used to compare with previous work. The standard form of constrained optimization problem can be described as follows,

$$\text{Minimize:} \quad f(X) \tag{9}$$

$$\text{Subject to:} \quad \begin{array}{l} g_i(X) \le 0, i = 1, 2, \ldots, p \\ h_j(X) = 0, j = 1, 2, \ldots, q \end{array}$$

where $f(X)$ is an objective function, $g_i(X)$ are inequality constraints and $h_j(X)$ are equality constraints.

For constrained optimization problem, a huge number of methods have been proposed [45,46]. Among them, the penalty function approaches very popular because of its ease of implementation. However, since the evolution is dynamical, a fixed penalty function is not suitable for the whole progress. Michalewicsz et al. [47] have recognized the importance of using adaptive penalty in constrained optimization. For (9), the penalty function is designed as follows,

$$\Phi(X) = f(X) + \left(\sum_{i=1}^{p} r_i \times g_i(X) + \sum_{j=1}^{q} s_j \times h_j(X) \right) \tag{10}$$

where r_i and s_j are penalty factors. The objective function can be formulated as follows,

$$F(X) = \begin{cases} f(X) & , \quad if \quad constraints \quad are \quad met \\ f(X) + \Phi(X), & \quad otherwise \end{cases} \tag{11}$$

where $\Phi(X)$ is a penalty function defined in (10). In the remainder of this section, the adaptive penalty functions methods are employed to solve the constrained engineering optimization problems. In the simulations, the population is set as 100 and split into 2 subgroups. In BBO, the linear migration model is employed. For PSO, the two learning factors are both set as 1.296 and the inertial weight is set as 0.726. The mutation rates for BBO and PSO are all 0.01. For each simulation, the limitation generation is set as 50000 and for each problem 100 simulations are conducted.

6.1 Design a Gear Train

The first case is a design of a compound gear train arrangement shown in Figure 8. The gear ratio for a reduction gear train is defined as the ratio of angular velocity of the output shaft to that of the input shaft. The overall gear ratio between input and output is shown as follows,

$$ratio = \frac{\omega_{output}}{\omega_{input}} = \frac{T_d T_b}{T_a T_f} \tag{12}$$

where ω_{output} and ω_{input} are the angular velocities of the output and input shafts respectively and T denotes the number of teeth on each gearwheel. The ratio should be designed as close as possible to 1/6.931. The number of teeth in each gear should be an integer and lie between 12 and 60. This problem can be described mathematically as follows:

$$\text{Minimize:} \quad F(X) = \left\{ \frac{1}{6.931} - \frac{T_a T_b}{T_c T_d} \right\}^2 = \left\{ \frac{1}{6.931} - \frac{x_1 x_2}{x_3 x_4} \right\}^2 \tag{13}$$

$$\text{Subject to:} \quad 12 \le x_i \le 60 \text{ ,where } \quad i \in \{1,2,3,4\} \quad \text{and} \quad x_i \in Z$$

where $X = [x_1, x_2, x_3, x_4]^T = [T_a, T_b, T_c, T_d]^T$ in (12) and Z presents integer set.

Table 7 lists the various design solutions and compare BPSO's outcome with several other pervious works including [48, 49, 50, 51, 52, 53, 54].

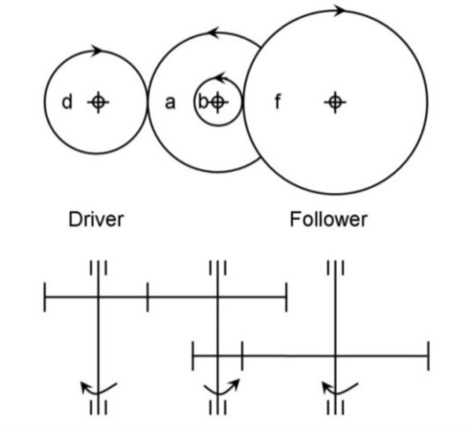


Figure 8 Compound gear train

Table 7 Optimal solutions for the gear train problem.

Item	Optimum Solution					Type
	Sandgren[48]	Fu et al.[49]	Davidet al.[50]	Kannan et al.[51]	Cao&Wu[52]	
x1	18	14	18	13	30	Integer
x2	22	29	18	15	15	Integer
x3	45	47	42	33	52	Integer
x4	60	59	53	41	60	Integer
F(X)	5.7123e-06	4.5476e-06	1.6211e-06	2.1246e-08	2.3576e-9	
Gear Ratio	0.1466	0.146411	0.1456	0.1441	0.144230	
%Error	1.65%	38.3%	0.88%	0.11%	0.03366%	
Item	Optimum Solution					Type
	Lin et al.[53]	Wu&Chow[54]	PSO	BBO	BPSO	
x1	19	19	12	17	16	Integer
x2	16	16	12	22	19	Integer
x3	49	43	28	48	43	Integer
x4	43	49	36	54	49	Integer
F(X)	2.3576e-9	2.7009e-12	2.0226e-6	1.1661e-10	2.7009e-12	
Gear Ratio	0.144230	0.144281	0.142857	0.144290	0.144281	
%Error	0.03366%	0.00011%	0.98571%	0.00748%	0.00011%	

In Table 7, the results in paper [54] and in BPSO are both the best. The two algorithms outperform others. In addition, BBO performs the second best, which means this novel algorithm has a certain potential to solve optimization problems.

6.2 Design of a Pressure Vessel

The second problem is the design of a pressure vessel. A cylindrical pressure vessel is capped at both ends by hemispherical heads as shown in Figure 9. To minimize the total cost including material cost, forming and welding cost, the thicknesses of the shell and the head, the inner radius and the length of the cylindrical section should be optimized. The mathematical model of pressure vessel design is described in (14), with T_s, T_h, R and L parameters denoted by x_1, x_2, x_3 and x_4, respectively.

$$\text{Minimize: } F(X) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3 \quad (14)$$

$$\text{Subject to: } \begin{aligned} g_1(X) &= -x_1 + 0.0193x_3 \le 0 \\ g_2(X) &= -x_2 + 0.00954x_3 \le 0 \\ g_3(X) &= -\pi x_3^2x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \le 0 \\ g_4(X) &= x_4 - 240 \le 0 \end{aligned}$$

where $0.0625 \le x_i \le 6.1875, (i = 1, 2)$ and $10 \le x_i \le 200, (i = 3, 4)$. The values of x_1 and x_2 were considered as integer multiples of 0.0625, and the values of x_3 and x_4 were considered as real numbers.

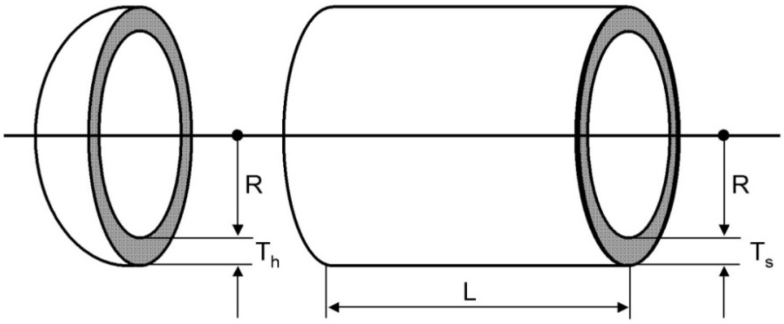


Figure 9 Schematic pressure vessel

Table 8 Optimal solutions for the pressure vessel problem

Item	Optimum Solution						Type
	Sandgren[48]	Fu et al.[49]	Wu&Chow[54]	PSO	BBO	BPSO	
x1	1.125	1.125	1.125	1	1	1.125	Discrete
x2	0.625	0.625	0.625	1	0.5	0.625	Discrete
x3	48.97	48.38070	58.1978	51	188.018	58.2902	Continuous
x4	106.72	111.7449	44.2930	90.6267	168.286	43.69269	Continuous
F(X)	7982.5	8048.619	7207.497	8800.321	6592.758	7198	

For this problem, it is reported in paper [53, 55] that the optimal value 7197.7 was reached. However, neither a more accurate result nor details of the results were provided, so the result is not shown in this comparison. In addition, in paper [56], the optimal value 7195.799 were reached, but the parameters used in the paper variables are too inaccuracy. Hence it is not list as well. Although the fitness of BBO is the best, it violates the constraints. Hence the result is invalid. According to the results in Table

8, the BPSO performs the best to solve the engineering optimization problem. In addition, BPSO performs better than both BBO and PSO, which means the proposed optimization approach is feasible and effective.

6.3 Design of Welded Beam

The third case is a well-studied welded beam design problem. The structure of the welded beam, which is shown in Figure 10, consists of a beam A and a beam B.

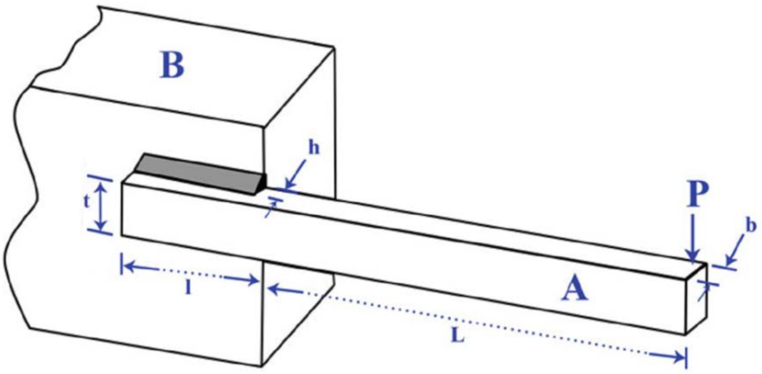


Figure 10 Schematic welded beam

To minimize the fabrication cost, four variables h, l, t and b, denoted by x_1, x_2, x_3 and x_4, should be optimized with several constraints. The mathematical formulation of the objective function is shown as follows,

$$\text{Minimize:} \quad F(X) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14 + x_2) \tag{15}$$

$$\text{Subject to:} \quad \begin{aligned} &g_1(X) = \tau(X) - \tau_{\max} \le 0, \\ &g_2(X) = \sigma(X) - \sigma_{\max} \le 0, \\ &g_3(X) = x_1 - x_4 \le 0, \\ &g_4(X) = 0.10471x_1^2 + 0.04822x_3x_4(14 + x_2) - 5.0 \le 0, \\ &g_5(X) = 0.125 - x_1 \le 0, \\ &g_6(X) = \delta(X) - \delta_{max} \le 0, \\ &g_7(X) = P - P_c(X) \le 0, \end{aligned}$$

where $\tau(X) = \sqrt{\tau_1^2 + 2\tau_1\tau_2\frac{x_2}{2R} + \tau_2^2}$, $\tau_1 = \frac{P}{\sqrt{2}x_1x_2}$, $\tau_2 = \frac{MR}{J}$, $M = P(L + \frac{x_2}{2})$,

$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2}$, $J = 2\left(\sqrt{2}x_1x_2\left(\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2}\right)^2\right)\right)$, $\sigma(x) = \frac{6PL}{x_3^2x_4}$,

$\delta(x) = \frac{4PL^3}{Ex_3^3x_4}$, $P_c(X) = \frac{4.013E\sqrt{\frac{x_3^2x_4^6}{36}}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)$, $P = 6000$, $L = 14$,

$E = 30*10^6$, $G = 12*10^6$, $\tau_{max} = 13600$, $\sigma_{max} = 30000$, $\delta_{max} = 0.25$.

In the formulas, X is a set of design variables (i.e., h, l, t and b), and L is the overhang length of the beam (14 inch). $\tau(X)$ is the allowable design shear stress of weld (13600 psi) and τ_{max} is the weld shear stress. $\sigma(x)$ is the allowable design yield stress for the bar material (30000 psi) and σ_{max} is the maximum barb ending stress. $P_c(x)$ is the bar buckling load. P is the loading condition (6,000 lb), and $\delta(x)$ is the bar end deflection. The optimization problem (15) is considered as an engineering benchmark to test BPSO and the results are compared with the previous work [56,57,58,59,60,61] shown in Table 9.

Table 9 Optimal solutions for the welded beam problem

Item	Optimum Solution				Type
	Coello[57]	Mahdavi[58]	Lee[59]	NSHS[56]	
x1	0.2088	0.20573	0.2442	0.2057	Continuous
x2	3.4205	3.47049	6.2231	3.4704	Continuous
x3	8.9975	9.03662	8.2915	9.0366	Continuous
x4	0.2100	0.20573	0.2443	0.2057	Continuous
g1	-0.337812			-4.5944e-5	
g2	-353.902604			-9.6037e-4	
g3	-0.00120			-7.3973e-9	
g4	-3.411865			-3.4330	
g5	-0.08380			-0.0807	
g6	-0.235649			-0.2355	
g7	-363.232384			-1.7294e-4	
F(X)	1.748309406	1.7248	2.3800	1.7249	
Item	Optimum Solution				Type
	IPHS[60]	Deb[61]	BBO	BPSO	
x1	0.20573	0.2489	0.2287	0.2064	Continuous
x2	3.47049	6.1730	3.2003	3.4715	Continuous
x3	9.03662	8.1789	8.5666	8.9985	Continuous
x4	0.20573	0.2533	0.2289	0.2075	Continuous
g1	0.0		-7.2974e-1	-7.8354e-2	
g2	0.0		3.2131	-3.0818	
g3	-5.55e-17		-2.0000e-4	-1.1302e-3	
g4	-3.4329		-3.3682	-3.4225	
g5	-0.0807		-1.0370e-1	-8.1367e-2	
g6	-0.2355		-2.3475e-1	-2.3548e-1	
g7	9.09e-13		-1.9739e+3	-1.3889e+2	
F(X)	1.7248	2.43	1.8077	1.7328	

From Table 9, although BPSO does not perform the best, it still has high ranking to solve welded beam problem. In addition, it is obvious that BPSO outperforms

BBO.

6.4 Design of Water Pumping System

The fourth case is a water pumping system, which consists of two parallel pumps drawing water from a lower reservoir to another 40m higher one, as shown in Figure 11. In addition, to overcome the pressure difference due to the elevation, the friction in the pipe is 7.2 w^2 kPa, where w is the combined flow rate in kilograms per second. Lieman et al. [62] did some rough tuning on this problem and gave the mathematical description as follows,

$$\text{Minimize:} \quad F(X) = 150 + 0.5(x_1 + x_2)^2 \tag{16}$$

such that,

$$x_3 = 250 + 30x_1 - 6_x^2$$
$$x_3 = 300 + 20x_2 - 12x_2^2$$
$$x_3 = 150 + 0.5\left(x_1 + x_2\right)^2$$

where $x_1 \in [0, 9.422]$, $x_2 \in [0, 5.903]$ and $x_3 \in [0, 267.42]$. Babu and Angira [63] reformulate (16) to obtain the following optimization problem.

$$\text{Minimize:} \quad F(X) = 150 + 0.5(x_1 + x_2)^2 \tag{17}$$

$$\text{Subject to:} \quad \begin{aligned} g_1(X) &= 6x_1^2 - 30x_1 - 249.999 + 150 + 0.5(x_1 + x_2)^2 \geq 0, \\ g_2(X) &= 12x_2^2 - 20x_2 - 299.999 + 150 + 0.5(x_1 + x_2)^2 \geq 0. \end{aligned}$$

where $0 \leq x_1 \leq 9.422$ and $0 \leq x_2 \leq 5.903$.

According to the results in Table 10, BPSO has comparable performance with previous work to solve this problem.

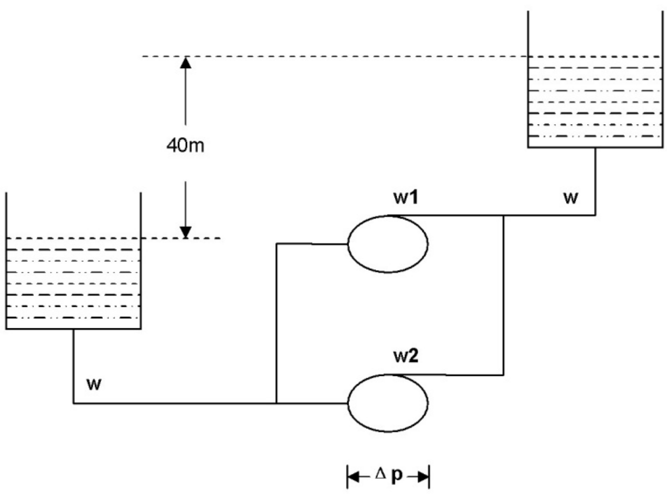


Figure 11 Schematic water pumping system

Table 10 Optimal solutions for the water pumping system problem

Item	Optimum Solution		
	DE[63]	Branch & Reduce[63]	BPSO
x1	6.293430	6.293429	6.293430
x2	3.821839	3.821839	3.821839
x3	201.159334	201.159334	201.159334
F(X)	201.159334	201.159334	201.159334

7. Conclusions

In this paper, an evolutionary algorithm BPSO is proposed by taking the advantages of Biogeography-Based Optimization and Particle Swarm Optimization. The whole population is split into several subgroups and in each subgroup BBO is adopted for intra-group search. Based on the search results, the algorithm uses fuzzy strategy to select a certain number of individuals as elites for global search which is conducted by PSO. After that, the individuals in PSO will return to its original subgroups for the next generation. This hierarchical structure employs a big proportion of population as elites and gives a full play of them. In numerical simulation, BPSO performs better than BBO and PSO along whatever a large number or a small number of elites are employed. The results also show BPSO performs better than other algorithms. In addition, four classic engineering optimization problems with constraints are studied to validate the feasibility and effectiveness of BPSO. Compared with previous work, BPSO performs in top ranking, which means this algorithm is competent to solve practical engineering optimization problems with constraints.

In future, the parameters in BPSO could be adjusted. Meanwhile, improved versions of BBO and PSO could be hybridized to obtain a better performance. For optimization problems, the parallel computation and multi-objectives problems should be considered by using BPSO. The stability of the novel algorithms should also be considered.

Acknowledgement:

This work is supported by the National Natural Science Foundation of China under Grant No.70871091, No.61075064, No.61034004and No.61005090, Program for New Century Excellent Talents in University of Ministry of Education of China, Ph.D. Programs Foundation of Ministry of Education of China (20100072110038), scientific research project of science and technology bureau of Jiaxing(2011BY7003), Singapore Academic Research Fund, under Grant R397000139133, R397000157112 and NUS Teaching Enhancement Grant C397000039001. The author Weian Guo would like to express his best regards to Shuzhi Sam Ge, in Interactive Digital Media Institute, National University of Singapore, for his guidance and help. In addition, Weian Guo appreciates China Scholarship Council (CSC) to support his expense in Singapore.

References:

1 T. Niknam, M. Zare, J. Aghaei, E.A. Farsani, A New Hybrid Evolutionary Optimization Distribution Feeder Reconfiguration, Applied Artificial Intelligence, 25 (10) (2011) 951-971

2 H. Beyer, The Theory of Evolution Strategies . New York: Springer, 2001.

3 E. Mezura-Montes, C. Coello, A simple multimembered evolution strategy to solve constrained optimization problems, IEEE Transaction on Evolutionary Computation, 9 (1) (2005) 1-17.

4 F. B. David, Evolutionary computation: the fossil record, New York: IEEE Press, 1998.

5 P. Chang, W. Huang, C. Ting,Dynamic diversity control in genetic algorithm for mining unsearched solution space in TSP problems, Expert Systems with Application 37 (3) (2010) 1863-1878.

6 R.Shahryar, D. Paul, Efficiency competition on N-queen problem: DE vs. CMA-ES, In: Canadian Conference on Electrical and Computer Engineering, Niagara Falls, Canada, May 04-07, 2008.

7 D. S. Liu, K. C. Tan, S. Y. Huang, C. X. Goh, W. K. Ho, On solving multi-objective bin packing problems using evolutionary particle swarm optimization, European Journal of Operational Research, 190 (2) (2008) 357-382.

8 X. Cai, S. Fan, Y. Tan, Light responsive curve selection for photosynthesis operator of APOA, International Journal of Bio-inspired Computation, 4(6) (2012) 373-379.

9 L.Xie, J.Zeng, R. A. Formato, Selection strategies for gravitational constant G in artificial physics optimisation based on analysis of convergence properties, International Journal of Bio-inspired Computation, 4(6) (2012)380-391.

10 J. H. Holland, Adaptation in Natural and Artificial Systems, Ann Arbor,MI: Univ. Michigan Press, 1975.

11 A. Sadegheih, P. R. Drake, A New Experimental Study of Genetic Algorithm and Sumulated Annealing with Bounded Variables, Applied Artificial Intelligence, 25 (10) (2011) 927-950

12 M. Dorigo, T. Stützle, Ant Colony Optimization. Cambridge, MA:MIT Press, 2004.

13 J. Kennedy, R. C. Eberhart, Y. Shi, Swarm Intelligence. SanFrancisco, CA: Morgan Kaufmann Publishers, 2001.

14 R. Ahila, V. Sadasivam, K. Manimala, Particle Swarm Optimization-Based Feature Selection and Parameter Optimization for Power System Disturbances Classification, Applied Artificial Intelligence, 26 (9) (2012)832-861
15 Z. Cui, F.Gao, Z. Cui, J. Qu, A second nearest-neighbor embedded atom method interatomic potential for Li-Si alloys, Journal of Power Sources, 207 (1) (2012) 150-159.
16 Z. Cui, F.Gao, Z. Cui, J.Qu, Developing a second nearest-neighbor modified embedded atom method interatomic potential for lithium, Modelling and simulation in materials science and engineering, 20(1) (2012) 015014.
17 M. K. Apalaka, D. Karabogab, B. Akayb, The Artificial Bee Colony algorithm in layer optimization for the maximum fundamental frequency of symmetrical laminated composite plates, Engineering Optimization, 10 May 2013
18 G. Kanthaswamy, J. Jovitha, Control of Dead Time System Using Hybrid Ant Colony Ant Optimization, Applied Artificial Intelligence, 25 (7) (2011) 609-634
19 N. Siddique, B. Amavasai, A. Ikuta, Editorial: Hybrid Techniques in AI, Artificial Intelligence Review, 27 (2-3) (2007) 77-78.
20 F. Van den Bergh, A. P.Engelbrecht, A Cooperative Approach to Particle Swarm Optimization, IEEE Transactions on Evolutionary Computation, 8 (3) (2004) 225-239.
21 M. E. H. Pedersen, A. J.Chipperfield, Simplifying Particle Swarm Optimization, Applied Soft Computing, 10 (2) (2010) 618-628.
22 D. Simon, Biogeography-Based Optimization, IEEE Transactions on Evolutionary Computation 12 (6) (2008) 702-713
23 M.R. Lohokare, S. Devi, S.S. Pattnaik, B.K. Panigrahi, J.G. Joshi, Modified biogeography-based optimisation (MBBO), International Journal of Bio-Inspired Computation, 3(4) (2011) 252-266.
24 T.S. JeyaliLaseetha, R. Sukanesh, Investigations on the synthesis of uniform linear antenna array using biogeography-based optimisation techniques, International Journal of Bio-Inspired Computation, 4(2) (2012) 119-130.
25 H. Ma and D. Simon, Analysis of Migration Models of Biogeography-Based Optimization using Markov Theory,Engineering Applications of Artificial Intelligence, 24 (6) (2011) 1052-1060.
26 D. Simon, A Probabilistic Analysis of a Simplified Biogeography-Based Optimization Algorithm, Evolutionary Computation, 19 (2) (2011) 167−188.
27 D. Simon, M. Ergezer, D. Du, and R. Rarick, Markov Models for Biogeography Based Optimization, IEEE Transactions on Systems, Man, and Cybernetics −Part B, 41 (1) (2011) 299-306.
28 R. MacArthur and E. Wilson, The Theory of Biogeography.Princeton, NJ: Princeton Univ. Press, 1967.
29 B. Khokhar, K. Parmar, and S. Dahiya, Application of Biogeography-based Optimization for Economic Dispatch Problems, International Journal of Computer Applications, 47 (13)(2012) 25-30.
30 W. Guo, D. Yang, J. Hu, C. Huang, L. Wang, Q. Wu, Optimal management of a home smart grid scheduling, Journal of Computational Information Systems, 8 (5) (2012), 1921-1928.
31 D. Simon, M. Ergezer, D. Du, and R. Rarick, Markov models for biogeography-based optimization, IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, 41 (1) (2011) 299-306

32 H. Ma, An analysis of the equilibrium of migration models for biogeography-based optimization, Information Sciences 176 (8) (2010) 3444-3464.

33 W. Guo, L. Wang and Q. Wu, An analysis of the migration rates of Biogeography-based Optimization, Information Sciences, Accept.

34 V.Gazi, K. M. Passino, Swarm Stability and Optimization, Springer, 2011, pp. 263

35 J. Kennedy and R.Eberhart, Particle Swarm Optimization. In: Proceedings of IEEE International Conference on Neural Networks. 1995, IV. pp. 1942–1948.

36 J. Robinson, Y. Rahmat-Samii, Particle swarm optimization in electromagnetics, IEEE Transactions on Antennas and Propagation 52 : 2 (2004) 397-407

37 R. Poli, An analysis of publications on particle swarm optimisation applications, Technical Report CSM-469 (Department of Computer Science, University of Essex, UK). (2007)

38 R. Poli, Analysis of the publications on the applications of particle swarm optimisation. Journal of Artificial Evolution and Applications, Volume 2008, 1–10.

39 Matthew Settles, An Introduction to Particle Swarm Optimization University of Idaho, Moscow (2005) November, 1-8

40 T. Back, Evolutionary Algorithms in Theory and Practice. Oxford, U.K.: Oxford Univ. Press, 1996.

41 Z. Cai and Y. Wang, A multi-objective optimization-based evolutionary algorithm for constrained optimization, IEEE on Evolutionary Computation 10 (6) (2006) 658-675.

42 X. Yao, Y. Liu, and G. Lin, Evolutionary Programming made faster, IEEE Transactions on Evolutionary Computation. 3 (2) (1999) 82-102.

43 I. Parmee, Evolutionary and Adaptive Computing in Engineering Design. New York: Springer, 2001.

44 K. Price and R. Storn, Differential evolution, Dr. Dobb’s Journal, 22 (24) (1997) 18-20.

45 C.Coello, Theoretical and numerical constrainthandling techniques usedwith evolutionary algorithms: a survey of the state of the art, ComputerMethods in Applied Mechanics and Engineering, 191(11-12) (2002) 1245-87.

46 Z. Michalewicz, M.Schoenauer. Evolutionary algorithms for constrainedparameter optimization problems, Evolutionary Computation, 4(1) (1996) 1-32.

47 Z. Michalewicz, D. Dasgupta, R. Le Riche, M. Schoenauer,Evolutionary algorithms for constrained engineering problems, Computers and Industrial Engineering Journal 30 (4) (1996) 851–870.

48 E. Sandgren, Nonlinear Integer and Discrete Programming in Mechanical Design Optimization, Journal of Mechanical Design, 112 (2) (1990) 223-229.

49 J. Fu, R. G. Fenton, W. L. Cleghorn, A mixed integer discrete continuous programming method and its application to engineering design optimization, Engineering Optimization, 17 (4) (1991) 263-280.

50 F. W. Y. David, S. P. Koh, S. K. Tiong, S. K. Prajindra, A hybrid artificial immune systems for multimodalfunction optimization and its application in engineeringproblem, Artificial Intelligence Review, 38 (4) (2012) 291-301

51 B. K. Kannan, S. N. Kramer, An augmented lagrange multiplier based method for mixed integer discretecontinuous optimization and its applications to mechanical

design, Journal of Mechanical Design, 116 (2) (1994) 405-411
52 Y. Cao, Q. Wu, Mechanical design optimization by mixed variables evolutionary programming. Proceedings of the 1997 IEEE Conference onEvolutionary Computation, pp. 443-446, 1997.
53 S. Lin, C. Zhang, H. Wang, On mixed discrete nonlinear optimization problems: A comparative study, Engineering Optimization, 23 (4) (1995) 287-300
54 S. Wu, P. Chow,Genetic algorithms for nonlinear mixed discrete integer optimization problems via meta-genetic parameter optimization, Engineering Optimization, 24 (2) (1995) 137-159.
55 H. Loh, P. Y. Papalambros, A sequential linearization approach for solving mixed-discrete nonlinear design optimization problems, Journal of Mechanical Design, 113 (3) (1991) 325 -334
56 H.Sarvari, K.Zamanifar,Improvement of harmony search algorithm by usingstatistical analysis, Artificial Intelligence Review, 37 (3) (2012) 181-215
57 C.A.C Coello, Use of a self-adaptive penalty approach for engineering optimization problems, Computers in Industry, 41(2) (1999) 113–127
58 M. Mahdavi, M. Fesanghary, E.Damangir, An improved harmony search algorithm for solving optimization problems, Applied Mathematicsand Computation,188 (2) (2007) 1567-1579
59 K. S. Lee, Z. W.Geem, A new meta-heuristic algorithm for continues engineering optimization: harmonysearch theory and practice,Computer Methods in Applied Mechanicsand Engineering, 194 (36-38) (2005) 3902-3933
60 M.Jaberipour, E.Khorram, Two improved harmony searchalgorithms for solving engineering optimizationproblems. Communication in Nonlinear Science and Numerical Simulation, 15 (11) (2010) 3316-3331
61 K. Deb, Optimal design of a welded beam via genetic algorithms, AIAA Journal, 29 (11) (1991) 2013-2015.
62 J. Leibman, L. Lasdon, L. Schrage and A. Waren, Modeling and optimization with GINO. The Scientific Press, Palo Alto, CA, 1986.
63 B.V. Babu, R. Angira, Optimization of Water Pumping system using differential evolution strategy. The second International Conference on Computational Intelligence, Robotics, and autonomous Systems, Singapore, December 15, 2003.