SOLVING THE ASSOCIATED WEAKNESS OF BIOGEOGRAPHY-BASED OPTIMIZATION ALGORITHM

Ali R. Alroomi[1], Fadhel A. Albasri[2] and Jawad H. Talaq[3]

[1]Electrical Engineer, University of Bahrain, Sakhir, Bahrain
[2,3]Department of Electrical Engineering, University of Bahrain, Sakhir, Bahrain

ABSTRACT

Biogeography-based optimization (BBO) is a new population-based evolutionary algorithm and is based on an old theory of island biogeography that explains the geographical distribution of biological organisms. BBO was introduced in 2008 and then a lot of modifications and hybridizations were employed to enhance its performance. The researchers found that the original version of BBO has some weakness on its exploration. This paper tries to solve the root problems itself instead of solving its effect by using different techniques. It proposes two modifications; firstly, modifying the probabilistic selection process of the migration and mutation stages to give a fairly randomized selection for all the features of the islands. Secondly, the clear duplication process, which is located after the mutation stage, is sized to avoid any corruption on the suitability index variables of the non-mutated islands. The proposed modifications are extensively tested on 120 test functions with different dimensions and complexities. The results proved that the BBO performance can be enhanced effectively without embedding any additional sub-algorithm, and without using any complicated form of the immigration and emigration rates. In addition, the new BBO algorithm requires less CPU time and becomes even faster than the original simplified partial migration-based BBO. These essential modifications have to be considered as an initial step for any other modifications.

KEYWORDS

Biogeography-Based Optimization, BBO, Evolutionary Algorithm, Migration, Partial Migration

1. INTRODUCTION

Biogeography-based optimization (BBO) is a new population-based evolutionary algorithm (EA) that was introduced by Dan. Simon in 2008 [1]. BBO algorithm is based on the theory of island biogeography. It is an old theory that was presented in 1960s by the two ecologists, H. MacArthur and Edward O. Wilson [2,3].

Habitat, in biogeography, is the locality, site and particular type of local environment occupied by an organism [5], where the island is any area of suitable habitat surrounded by an expense of unsuitable habitat and is endowed with exceptionally rich reservoirs of endemic, exclusive, strange and relict species [6].

Each island has its own features as simple biotas, varying combinations of biotic and abiotic factors, and variability in isolation, shape, and size [7,9]. With these characteristics, islands represent themselves as natural experiments, see Fig. 1.

In BBO, the richness of species on any island depends on the availability of the good biotic and abiotic factors which represents the independent variables of such a problem. Thus, if the island characterized with a lot of good features, then it will attract more species, and represents itself as a good solution.

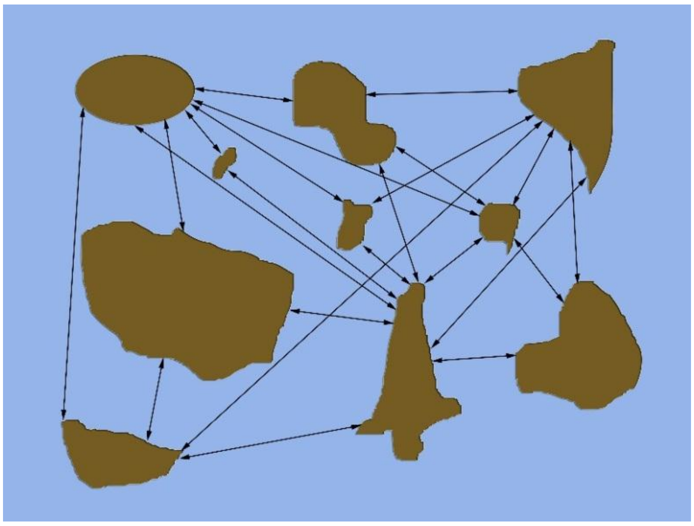


Figure 1. Migration process between islands with different isolation, shape, and size

The researchers found that the raw form of the original BBO algorithm has some weakness in its migration and mutation stages that affects its overall performance. However, the proposed solutions in the literature are focusing on solving the effects of that weakness rather than solving the sources of such weakness. The objective of this paper is to treat that issue by focusing on the root problems instead of using other complicated approaches, like non-linear migration rates in [13,16] and/or hybridization with other optimization techniques [18], which adds new sub-algorithm that needs more acknowledgement and requires extra CPU time.

This paper is organized as follows: Section 2 gives a brief overview about the theory of island biogeography and Section 3 explains the algorithm of BBO. Section 4 shows how BBO performance can be enhanced more by solving the root problems of the migration and mutation stages, and then followed by a performance comparison between the original and the modified BBOs in Section 5. The conclusions and suggestions are presented in Section 6.

2. THE THEORY OF ISLAND BIOGEOGRAPHY

The equilibrium theory of island biogeography proposes that the number of inhabited species on an island is based on the dynamic equilibrium between new immigrated species onto an island and the extinct species out from that island [2,3,8].

Fig. 2 graphically represents the equilibrium model with linear immigration (or speciation) rate λ and emigration (or extinction) rate μ, where they can be plotted as logistic, exponential or any proper function [4,10,11].

I and E are the maximum possible immigration and emigration rates, respectively. I occurs when the island is empty of any species and thus it offers a maximum opportunity to the species on the other islands for immigrating to settle on it; whereas the arrivals on that island increases, the opportunity for settlement will decrease, which means that the immigration rate decreases. Also, as λ decreases, the species density increases, and thus the predation, competition and parasitism factors will increase too; and as a result, the emigration rate μ will increase and reaches its maximum value E when λ reaches its minimum value [12].

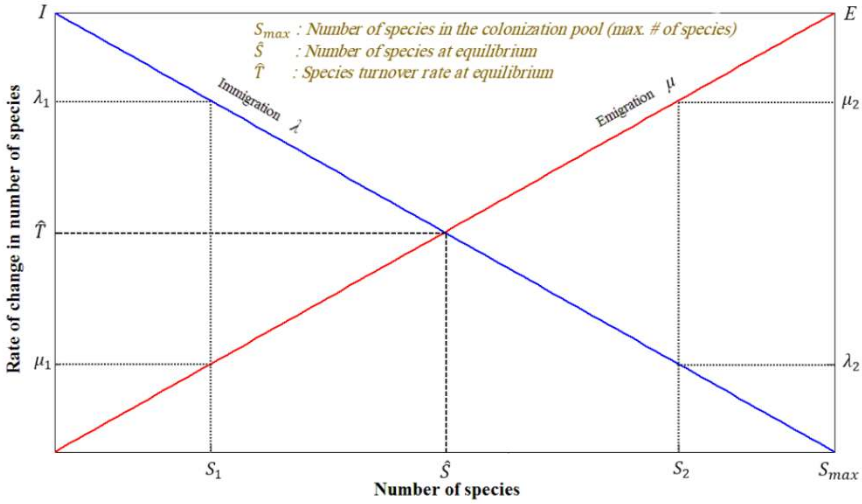


Figure 2. Simplified equilibrium model of a biota of a single island

MacArthur and Wilson, in their early study [2,3], proposed a simplified equilibrium model with $I=E$, where at time t, the recipient island has S species with probability $P_s(t)$, and λ_s and μ_s are the immigration and emigration rates at the present of S species in that island. The variation from $P_s(t)$ to $P_s(t+\Delta t)$ can be described as:

$$P_s(t+\Delta t) = P_s(t)(1-\lambda_s\Delta t - \mu_s\Delta t) + P_{s-1}(t)\lambda_{s-1}\Delta t + P_{s+1}(t)\mu_{s+1}\Delta t \tag{1}$$

From Eq. 1, to have S at time $(t+\Delta t)$, one of the following three conditions should hold:

1. S species at time t, and no immigration or emigration took place during the interval Δt;
2. $(S - 1)$ species at time t, and one species immigrated;
3. $(S + 1)$ species at time t, and one species emigrated.

The Δt has to be set with small value so the probability of more than one immigrated or emigrated species can be ignored. Now, as Δt approaches 0, the ratio $\left(\frac{\Delta P_s}{\Delta t}\right)$ approaches $\dot{P}_s(t)$:

$$\frac{dP_s(t)}{dt} \cong \lim_{\Delta t\to 0}\frac{P_s(t+\Delta t)-P_s(t)}{\Delta t}$$

$$\frac{dP_s(t)}{dt} \cong -(\lambda_s+\mu_s)P_s(t) + \lambda_{s-1}P_{s-1}(t) + \mu_{s+1}P_{s+1}(t) \tag{2}$$

By considering the above three conditions, Eq. 2 can be specified more as:

$$\dot{P}_s(t) = \begin{cases} -(\lambda_s+\mu_s)P_s + \mu_{s+1}P_{s+1}, & S=0 \\ -(\lambda_s+\mu_s)P_s + \lambda_{s-1}P_{s-1} + \mu_{s+1}P_{s+1}, & 1 \le S \le S_{max}-1 \\ -(\lambda_s+\mu_s)P_s + \lambda_{s-1}P_{s-1}, & S = S_{max} \end{cases} \tag{3}$$

If $P_s(t)$ is known, then $\dot{P}_s(t)$ can be obtained from Eq. 3, where the value of $P_s(t+\Delta t)$ in Eq. 1 can be approximated as:

$$P_s(t+\Delta t) \cong P_s(t) + \dot{P}_s(t)\Delta t \tag{4}$$

Eq. 4 is the final confirmed form that has to be used in the program of BBO for calculating $P_s(t+\Delta t)$.

For finding $P_s(t)$, Dan. Simon in [1] gives two methods; either by solving Eq. 3 numerically, or applying the following theorem:

Theorem 1: The steady-state value for the probability of the number of each species is given by:

$$P(\infty) = \frac{v}{\sum_{i=1}^{S_{max}+1} v_i} \tag{5}$$

Where v and v_i can be computed from the following equations:

$$v = [v_1, v_2, \dots, v_{S_{max}+1}]^T \tag{6}$$

$$v_i = \frac{S_{max}!}{(S_{max}+1-i)!\,(i-1)!} \qquad (i = 1, \dots, S_{max}+1) \tag{7}$$

Although the second alternative is easier and $P_s(t)$ can be computed directly without any iteration, this method is not preferable in the most programs, such as C/C++, MATLAB, Octave, Maple, Python, etc, because it is valid only when $S_{max} \leq 170$, otherwise $S_{max}! = \infty$, unless an special sub-algorithm is used to overcome this problem. In addition, this approach will consume extra CPU time for dealing with long product operations.

The remaining terms for finding $P_s(t + \Delta t)$ are μ_s and λ_s, which can be calculated directly as:

$$\mu_s = \frac{E}{S_{max}} S \tag{8}$$

$$\lambda_s = 1 - \mu_s = I\left(1 - \frac{S}{S_{max}}\right) \tag{9}$$

3. BIOGEOGRAPHY-BASED OPTIMIZATION (BBO)

BBO translates the natural distribution of species into a general problem solution [1]. Each island represents one solution, where the good problem solution means that the island has lots of good biotic "*living: diversity of prey, trees, shrubs, meadow, etc*" and abiotic "*non-living: distance of isolation, wind, temperature, humidity, water, area, etc*" factors, which attracts more species than the other islands [4]. Each feature is called suitability index variable (*SIV*), which represents the independent variable of such a problem in BBO. As these features changes, the island suitability index (*ISI*) changes too; thus in BBO, *ISI* is the dependent variable [1,17].
A problem with *n*-independent variables and *k*-islands or individuals can be expressed as:

$$ISI_i = f(SIV_1, SIV_2, \dots, SIV_n) \qquad i = 1,2, \dots, k \tag{10}$$

In the early stages of introducing BBO, Dan. Simon proposed four different types of migration process, these types can be sorted as [1,15]:

1. Partial Migration Based BBO “PMB-BBO”
2. Single Migration Based BBO “SMB-BBO”
3. Simplified Partial Migration Based BBO “SPMB-BBO”
4. Simplified Single Migration Based BBO “SSMB-BBO”

From preceding study [20], it is shown that SMB-BBO and SSMB-BBO give poor performance but with lowest CPU time, while the performance comparison between PMB-BBO and SPMB-BBO shows that PMB-BBO gives better performance as the complexity, side constrains and/or dimensions of a given problem increases and as the number of islands decreases; and vice versa for SPMB-BBO. This judgment could be clearly observed if the mutation algorithm is not activated, because it compensates for the weakness of the migration algorithm. As a result, SPMB-BBO could trap in a local or at least a near-global optima. For this reason, PMB-BBO is selected as a final confirmed BBO model for applying the proposed essential modifications, *which will be explained in the next section*.

The algorithm of BBO consists of two main stages, migration and mutation.

3.1. Migration

Considering Fig. 2 and Eq. 10, if island i has lots of features, then lots of species will colonize it, which means that λ_S becomes low and μ_S becomes high.

Thus, the high *ISI* for island i represents a good solution, and vice versa for a poor solution which has a shortage in its features diversity, and reflected on the total available number of species; where at this condition, λ_S is high and μ_S is low.

From Fig. 2, S_1 is located before $\hat{S}$, where λ_S is high, μ_S is low and the solution ISI_1 is poor; while S_2 is located after $\hat{S}$, where λ_S is low, μ_S is high and the solution ISI_2 is good. Based on that, λ_S and μ_S can be used as indications of poor and good solutions, respectively.

The purpose of the migration process is to use high *ISI* islands as a source of modification to share their features with low *ISI* islands, so the poor solutions can be probabilistically enhanced and may become better than those good solutions.

The migration process of PMB-BBO can be described as:

```
Let ISI_i denote the ith population member and contains n features
For each island ISI_i (where i=1,2,3,...,k)
    For each SIV s (where s=1,2,3,...,n)
        Use λ_i to probabilistically select the immigrating island ISI_i
            If rand < λ_i
                For j=1 to k
                    Use μ_j to probabilistically decide whether to emigrate to ISI_i
                        If ISI_j is selected
                            Randomly select an SIV σ from ISI_j
                            Replace a random SIV s in ISI_i with SIV σ
                        end if
                end for
            end if
        next SIV
next island
```

3.2. Mutation

In island theory, the species at equilibrium point $\hat{S}$ can be deviated dramatically due to some external events. Events such as predators from other islands, tsunamis, volcanos, diseases or earthquakes cause negative deviation, and the total number of species will steeply decrease [11].

On the other hand, there are some other useful events such as wind-carrying seeds (wind pollination) or flotsam (shipwreck) which provide good features to an island, thus giving better solution with a significant enhancement [12]. In BBO, the mutation process is modeled as *SIV* mutation; and through species count probabilities P_s , the mutation rate *m* can be determined as:

$$m = m_{max}\left(1 - \frac{P_s}{P_{max}}\right) \tag{11}$$

m_{max} is a user-defined maximum mutation rate that *m* can reach, and $P_{max} = max\,(P_s)$.

From the previous equation, m reaches to its minimum "*zero*" at the maximum value of P_s, and vice versa. Thus, *m* is inversely proportional to P_s. This process can be graphically described as in Fig. 3, where the species count *S* starts from zero to S_{max}. As m_{max} increases, the chance to let the solutions be mutated increases too.

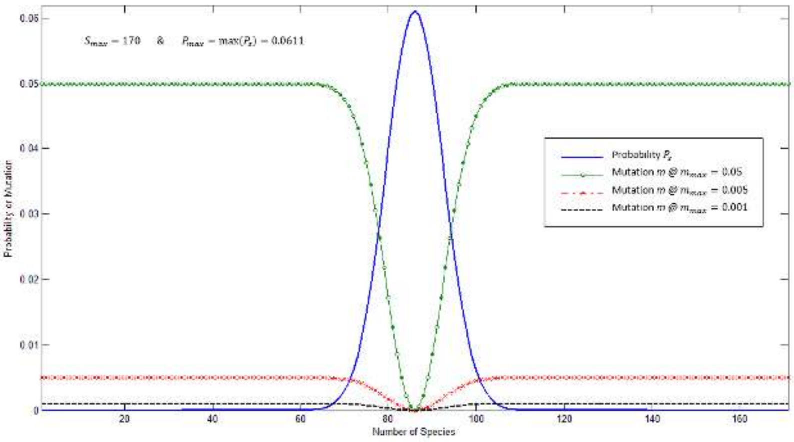


Figure 3. Comparison between $P_s(t)$ and $m(t)$ at different m_{max}

During the mutation stage, the low and high *ISI* solutions are likely to be mutated, and then could be enhanced more than what they already have, where the solutions at the equilibrium point are not mutated [1]. Even if the mutated solutions become worse, the optional stage, called elitism, will store the best solutions from one generation to the next [17].

The mutation process can be described as:

For $i = 1$ to k (where k is the number of islands, see Eq. 10)
    Calculate probability P_s based on λ_S and μ_S (by numerical or direct method)
    Calculate mutation rate m (using Eq. 11)
    Select ISI_i with probability proportional to P_s
    If ISI_i is selected
        Replace *SIV* of ISI_i with a randomly generated *SIV*
    end if
end for

3.3. BBO Algorithm

The BBO algorithm can be summarized through the flowchart of Fig. 4. The algorithm's looping can be terminated either if it reaches to an acceptable tolerance or after completing the desired number of generations.

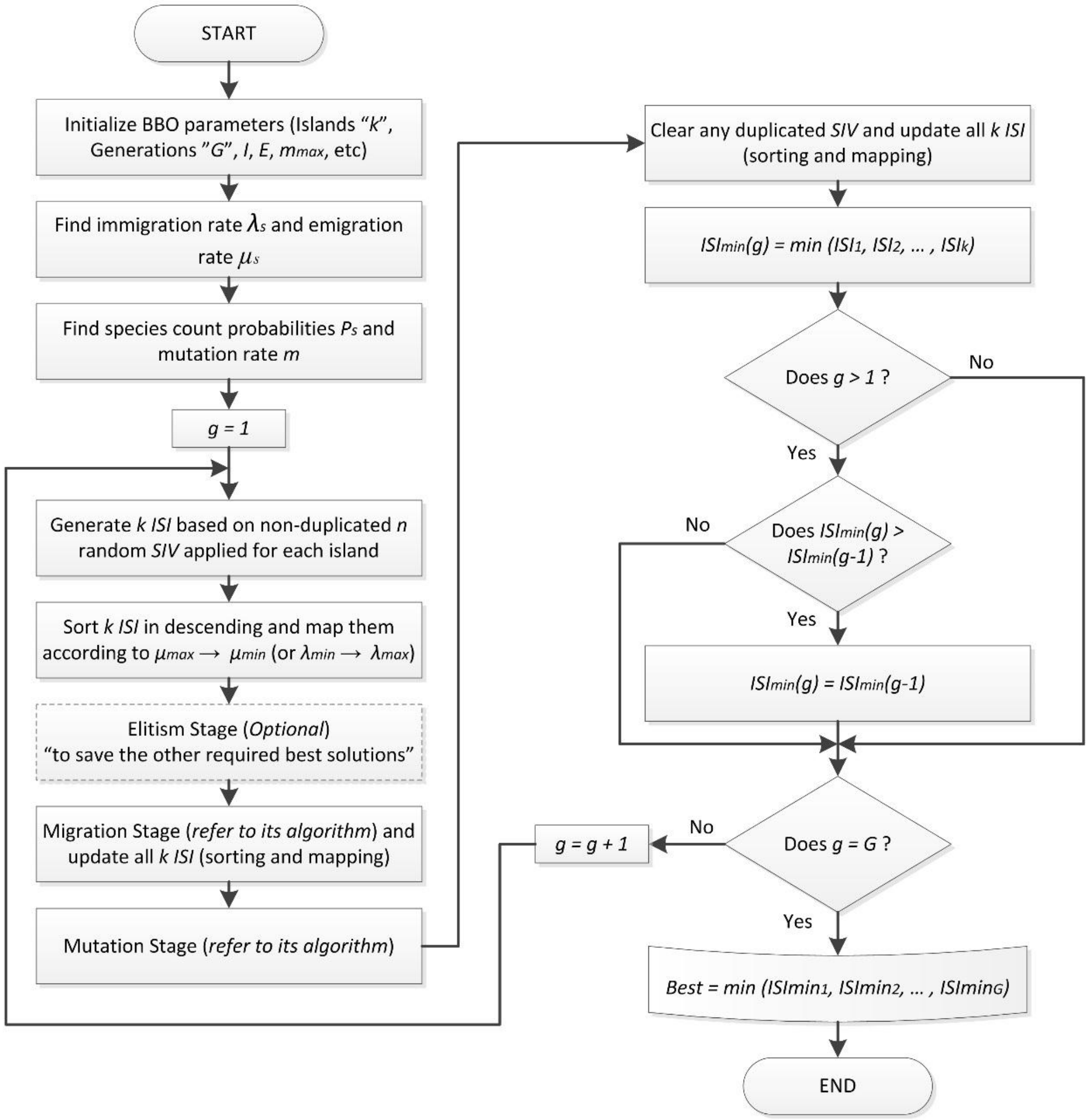


Figure 4. General flowchart of BBO algorithm

4. THE PROPOSED ESSENTIAL MODIFICATIONS ON BBO

This paper tries to solve two fundamental problems that are associated with the original version or the raw form of BBO with using just a linear immigration and emigration rates.

By accomplishing this modification, it can be used a basis for any further modification or hybridization. This study shows that the linear immigration and emigration rates still can do well if the associated root problems are solved.

According to the original BBO program that was designed by Dan. Simon in [21], there are two essential parts that need to be corrected:

4.1. Probabilistic Selection Process of the Migration and Mutation Stages

Referring to the preceding algorithms, the process for selecting *SIV s* of an island *i* that needs to be migrated is done probabilistically, and the general code for this task is:

$$SIV_s(1:k) = round[1 + (n-1) \times rand(1,n)] \quad (12)$$

If the independent variables of each *ISI* are represented as a vector of $[SIV_1, SIV_2, \ldots, SIV_n]$, then the analysis of the above code shows that the *SIV s* at the beginning and at the end, i.e. SIV_1 and SIV_n, have less weight than the other *(n-2) SIV* that are located in between.

Fig. 5a shows how unfair selection be done for each *SIV* of island *i*. In this example; *n=5* and *k=1000*, with *4* trails since the process is done randomly.

In MATLAB, the proposed modification is to use integer random *"randi"* instead of using rounded real random *"rand"* with *n SIV* alignment, as in Eq. 12. This integer random function provides pseudorandom integers from a uniform discrete distribution on *1 to n*.

For getting integer random values with fairly selection for all *n SIV*:

$$SIV_s(1:k) = randi(n, 1, n) \quad (13)$$

This MATLAB code given in Eq. 13 is equivalent to the previous code in Eq. 12, but with significant enhancement. The same analysis is done for this code, and the result is shown in Fig. 5b. It can be clearly seen that the selection process for the migrated and mutated *n SIV* is enhanced.

Note that, this part of modification is not available for 1-dimensional problems, and has less effect for 2-dimensional problems.

4.2. Clear Duplication Process of the Mutation Stage

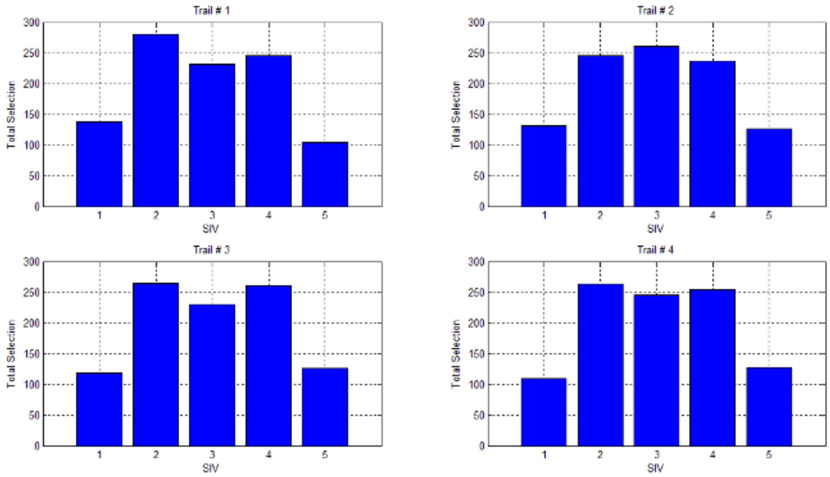
According to the original BBO [21], only the worst solutions are mutated. The range of these mutated solutions can be defined through the preceding mutation algorithm in Section 3.2 as:

For $m = round\left[length\left(\frac{Islands}{d}\right): length(Islands)\right]$
    Do mutation (*refer to its algorithm*)
end for

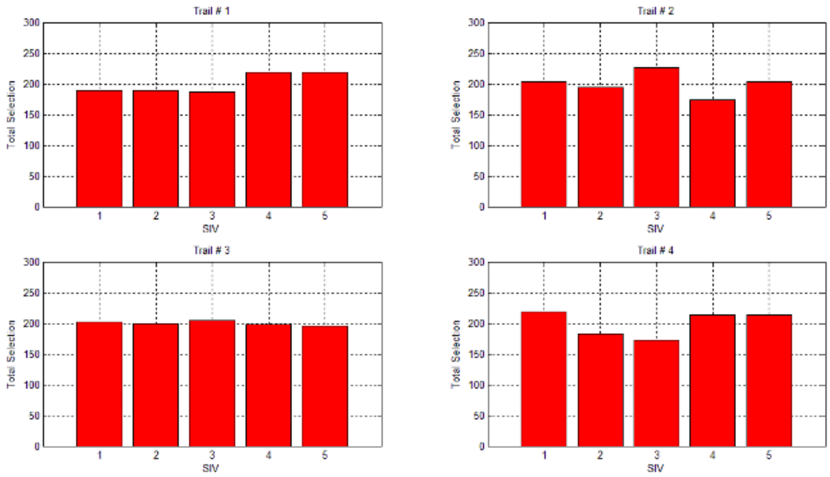
If *d=2*, then the worst half solutions are to be mutated; and as *d* increases, the percentage of the total mutated solutions increases too.

The problem happens when the mutation stage is completed, because the clear duplication process covers all the solutions. This action will corrupt the non-mutated solutions, and can affect the migration performance in the next generations.

The suggested modification to this problem is to size that process to be done only on the mutated solutions, so that the *n SIV* of the non-mutated solutions are kept away from any change.



(a) Original SIV Selection Process for Migration and Mutation Stages



(b) Modified SIV Selection Process for Migration and Mutation Stages

Figure 5. The Original and Modified *SIV*-Selection Process for Migration and Mutation Stages

5. PERFORMANCE COMPARISON

The original and modified versions of PMB-BBO have been extensively tested through 120 of test functions with different dimensions and complexities, where all the details of each test function are given in the Appendix.

Table 1a shows the parameters used for both BBOs. These parameters are similar to those used in [13,16], but with more restriction on the generation limits, which are listed in Table 1b.

The performance comparison are evaluated using 120 test functions. They are split into two equally groups; the first group contains only 2-dimensional test functions and are listed in Table 2a, while the second group contains other n-dimensional test functions and are listed in Table 2b. The reason for this arrangement is because this paper is a part of a project that deals with 2-dimensional engineering problem, and the obtained result from this paper will be used as a basis to determine if these essential modifications give a better performance or not before implementing it for solving that particular problem.

Therefore, more effort was done on 2-dimensional test functions to cover a gradient of difficulties of unimodal and multimodal functions in conjunction with few and many local minima as traps, where the diversity of variable bounds through narrow and wide search spaces provides other challenges.

Table 1a. BBOs' parameters (For more details refer to [13,16])

Parameter	Value
Population size – or k	50
Max. λ – or I	1
Max. μ – or E	1
m_{max}	0.01
Elitism	1
Mutation range	$round\left[\left(\frac{k}{2}\right):k\right]$
Number of trails	30

Table 1b. Required generations for various n-dimensional problems

Problem's dimension	# of generations
1, 2, 3, 4, 5 or 6	1,000
8, 9 or 10	5,000
15, 17 or 20	10,000
30	20,000
60	50,000

Table 2a and Table 2b give the best, mean and standard deviation of the 120 test functions. As an overall, the obtained results shows that the performance of PMB-BBO can be enhanced effectively by applying the proposed essential modifications on its probabilistic selection process and the range of the clear duplication process.

However, for 1-dimensional problems, this proposed modification is not effective because they have only one *SIV*, which means that the migration and mutation stages are processed within one independent variable. Thus, the first part of the proposed modification is absent. Based on this study, the new version of PMB-BBO provides better performance if both parts of the essential modifications are embedded. This can be seen from the result of the 1-dimensional problems that are shown in Table 2b. This issue could be solved if the blinded or/and binary-coded BBO program is used instead of using the real-coded BBO program, although it has not yet been proven. Similarly, for other n-dimensional problems (where $n \geq 2$), the modified PMB-BBO shows better performance and wins in most test functions. From the results of the best error, the mean, and the standard deviation, it can be concluded that the proposed correction on the randomized selection process of the migration and mutation algorithms improves its exploration and exploitation. Fig. 6 shows the curves of fitness functions of both versions for the Generalized Rastrigin's function, Schwefel's problem 1.2, Qing's function and Salomon's function. It can be clearly seen that the modified PMB-BBO can converge to a better solution more than the original version.

The modified PMB-BBO has proved that these essential modifications are highly recommended for problems with dimension higher than 1. Even, for the few test functions, where the original version shows better results, the modified version gives a competitive results.

Although, the modified version of PMB-BBO shows enhanced results, for some very hard test functions, like Price's Transistor, Storn's Chebyshev, Trid "or Neumaier F3", Normalized Rana, Bent Cigar, Qing, Generalized Rosenbrock and Schwefel F1.2 problems, both versions failed to converge to the optimal solutions. However, as a comparison, the proposed version outperform the original version on most of these test functions.

Table 2a. Comparison of results over 30 trails of the original and modified versions of PMB-BBO, where "Best" means the smallest error, "Mean" indicates the mean smallest error, and "StdDev" stands for the standard deviation – It contains just 2-dimensional test functions

Func. #	Function Name	n	Biogeography Based Optimization (BBO)					
			Original Partial Migration Based			Modified Partial Migration Based		
			Best	Mean	StdDev	Best	Mean	StdDev
f2-01	Aluffi-Pentini	2	1.4073E-07	4.3166E-05	5.3028E-05	**1.1278E-07**	**8.8268E-06**	**1.0718E-05**
f2-02	Banana Shape	2	**3.2053E-07**	**3.4468E-04**	**5.4340E-04**	2.9272E-06	7.4032E-04	8.3409E-04
f2-03	Beale	2	**2.8187E-06**	**1.6885E-04**	**1.6853E-04**	3.0033E-06	3.5367E-04	5.3948E-04
f2-04	Becker-Lago	2	9.1117E-08	1.4622E-05	1.7747E-05	**1.6084E-08**	**3.5743E-06**	**6.8117E-06**
f2-05	Bird	2	**7.0155E-07**	2.8262E-03	2.8437E-03	1.2209E-05	**3.5197E-04**	**4.4087E-04**
f2-06	Bohachevsky F1	2	2.6931E-05	1.1511E-03	1.2958E-03	**8.4775E-07**	**2.4441E-04**	**3.2364E-04**
f2-07	Bohachevsky F2	2	5.2095E-06	1.5371E-03	1.7717E-03	**1.5425E-06**	**1.2837E-04**	**1.6563E-04**
f2-08	Bohachevsky F3	2	9.8338E-05	**2.5948E-03**	**4.0705E-03**	**1.8607E-05**	3.4955E-03	4.0741E-03
f2-09	Booth	2	2.3450E-05	1.3871E-03	1.3831E-03	**7.0317E-08**	**2.7318E-04**	**3.8789E-04**
f2-10	Branin RCOS	2	**5.6480E-07**	9.6466E-05	1.6743E-04	6.1344E-07	**5.2541E-05**	**1.3086E-04**
f2-11	Bukin F4	2	**7.9693E-07**	6.6953E-05	6.2600E-05	9.5554E-07	**4.1762E-05**	**5.5448E-05**
f2-12	Bukin F6	2	1.4263E-01	**5.5443E-01**	**2.3743E-01**	**1.2043E-01**	6.5545E-01	3.2602E-01
f2-13	Carrom Table	2	1.6143E-06	2.6772E-04	3.3997E-04	**4.6210E-08**	**5.5445E-05**	**5.1728E-05**
f2-14	Chichinadze	2	4.2929E-06	8.3095E-03	1.6882E-02	**3.6249E-06**	**4.4121E-03**	**6.0434E-03**
f2-15	Complex	2	**3.3448E-09**	1.1847E-05	1.4765E-05	1.1044E-08	**2.4003E-06**	**3.0318E-06**
f2-16	Cosine Mixture	2	6.6629E-08	4.9456E-06	5.9989E-06	**1.7583E-11**	**1.2499E-06**	**2.0690E-06**
f2-17	Cross In Tray	2	1.6206E-08	3.1863E-06	4.7941E-06	**5.1076E-09**	**5.2181E-07**	**5.9861E-07**
f2-18	Cross Leg Table	2	9.9939E-01	**9.9959E-01**	**8.6652E-05**	**9.9933E-01**	9.9960E-01	9.1172E-05
f2-19	Crowned Cross	2	**1.2851E-01**	2.6119E-01	**3.8380E-02**	1.5692E-01	**2.4251E-01**	4.1245E-02
f2-20	Davis	2	1.4899E-01	4.0370E-01	1.3762E-01	**8.7947E-02**	**2.6525E-01**	**1.1171E-01**
f2-21	Decanomial	2	**2.3936E-06**	**1.2007E-02**	**2.0762E-02**	3.6446E-05	3.0835E-02	8.4008E-02
f2-22	Dekkers-Aarts	2	4.9519E-01	2.6947E+00	3.6355E+00	**4.8362E-01**	**1.8882E+00**	**3.5944E+00**
f2-23	Drop Wave	2	**6.5366E-06**	**3.6741E-03**	**1.1802E-02**	1.7502E-05	7.8462E-03	1.9646E-02
f2-24	Easom	2	3.3330E-06	1.0643E-04	8.4788E-05	**8.7507E-08**	**1.8423E-05**	**3.1394E-05**
f2-25	Egg Holder	2	8.1682E-02	4.4751E+00	**2.5844E+00**	**4.1432E-02**	**3.6928E+00**	5.8969E+00
f2-26	EXP2	2	1.3177E-07	2.0891E-05	**3.0858E-05**	**9.5422E-08**	**4.4703E-06**	**6.1713E-06**
f2-27	Freudenstein-Roth	2	5.4859E-05	1.0514E-02	2.1059E-02	**2.0160E-06**	**4.2978E-03**	**6.6492E-03**
f2-28	Giunta	2	2.4751E-08	5.0673E-07	6.3542E-07	**7.6581E-10**	**8.1474E-08**	**1.1653E-07**
f2-29	Goldstein-Price	2	2.7778E-05	1.5749E-03	1.6278E-03	**3.1758E-06**	**2.7174E-04**	**4.1521E-04**
f2-30	Himmelblau	2	5.9826E-06	9.4920E-04	1.2093E-03	**1.4971E-06**	**7.2058E-05**	**9.5358E-05**
f2-31	Holder Table	2	6.7954E-07	7.5566E-05	1.2147E-04	**6.7237E-07**	**1.6798E-05**	**2.3888E-05**
f2-32	Hosaki	2	2.9862E-08	1.2363E-05	1.3592E-05	**1.9118E-08**	**2.2394E-06**	**2.3016E-06**
f2-33	Kearfott	2	1.4138E-06	1.7270E-05	2.3710E-05	**9.0621E-09**	**3.6849E-06**	**6.8312E-06**
f2-34	Inverted Cosine Wave	2	8.6479E-06	5.0150E-04	5.6270E-04	**2.5292E-06**	**4.7085E-05**	**1.0482E-04**
f2-35	Levy F3 (or Hansen)	2	3.4917E-04	1.9752E-02	2.4033E-02	**2.1775E-04**	**1.7541E-03**	**2.2954E-03**
f2-36	Levy F5	2	**5.5062E-05**	1.4691E-01	1.4975E-01	6.4056E-05	**2.2562E-02**	**2.5591E-02**
f2-37	Matyas	2	**2.5988E-07**	6.4073E-05	7.3529E-05	5.2695E-07	**3.8882E-05**	**4.2294E-05**
f2-38	McCormick	2	1.9661E-07	1.9014E-05	2.7088E-05	**9.8424E-08**	**3.2686E-06**	**4.4669E-06**
f2-39	Michalewicz	2	4.4897E-07	1.4864E-05	3.5212E-05	**9.5133E-09**	**3.6163E-06**	**5.3882E-06**
f2-40	Muller-Brown Surface	2	8.2645E-04	2.2639E-02	2.2129E-02	**8.5253E-06**	**8.2214E-03**	**1.0327E-02**
f2-41	Parsopoulos	2	3.3808E-12	4.5059E-08	5.6394E-08	**8.3716E-13**	**2.2079E-08**	**2.7074E-08**
f2-42	Peaks	2	7.0089E-07	2.2400E-04	2.7718E-04	**4.2985E-08**	**3.7351E-05**	**4.3122E-05**
f2-43	Pen Holder	2	8.5556E-09	2.7320E-07	4.6665E-07	**1.9586E-10**	**3.8556E-08**	**3.8049E-08**
f2-44	Powell's Badly Scaled	2	3.3630E-04	**6.4544E-01**	3.7524E-01	**9.7414E-06**	8.9186E-01	**3.1950E-01**
f2-45	Sawtoothxy	2	7.1264E-05	3.2655E-03	6.8834E-03	**1.6672E-06**	**3.1424E-04**	**4.5476E-04**
f2-46	Schaffer's F1	2	9.5860E-04	**8.7166E-03**	2.4978E-03	**1.0451E-04**	9.2323E-03	**1.9361E-03**
f2-47	Schaffer's F2	2	1.4803E+00	6.6148E+00	3.4823E+00	**1.3788E+00**	**3.3010E+00**	**1.2894E+00**
f2-48	Shekel's Foxholes	2	**2.1720E-11**	8.0558E-08	2.1865E-07	6.4642E-11	**6.4142E-10**	**5.9986E-10**
f2-49	Sinusoidal Problem	2	2.2129E-07	4.4691E-05	5.4481E-05	**2.6980E-08**	**3.1974E-06**	**4.2395E-06**
f2-50	Stenger	2	1.0055E-06	1.7379E-04	2.2777E-04	**5.6936E-07**	**6.4535E-05**	**9.4907E-05**
f2-51	Storn	2	**3.9437E-07**	1.3249E-06	1.3478E-06	3.9493E-07	**1.1236E-06**	**1.2236E-06**
f2-52	Stretched V	2	**2.9346E-26**	**1.4161E-16**	**4.4560E-16**	5.9429E-24	4.1241E-16	1.3925E-15
f2-53	Test Tube Holder	2	1.0791E-06	8.0866E-05	9.8752E-05	**7.8935E-08**	**2.8773E-05**	**4.7662E-05**
f2-54	Treccani	2	9.4387E-08	2.0880E-05	2.7157E-05	**3.8026E-08**	**2.2693E-06**	**1.8565E-06**
f2-55	Trefethen F4	2	**4.6587E-04**	**9.5690E-02**	**8.1159E-02**	9.1985E-04	1.3107E-01	1.1657E-01
f2-56	Tripod	2	5.2656E-03	1.0925E-01	**6.3076E-02**	**3.5156E-03**	**5.3680E-02**	9.0101E-02
f2-57	Zakharov	2	**1.4286E-07**	6.8890E-05	6.9831E-05	6.0584E-07	**6.6225E-06**	**6.2012E-06**
f2-58	Zettl	2	2.6937E-08	8.7643E-06	1.0503E-05	**9.2581E-09**	**1.9735E-06**	**4.2604E-06**
f2-59	3-Hump Camel-Back	2	**5.7882E-08**	2.8687E-05	5.8124E-05	9.8715E-08	**5.3205E-06**	**7.9850E-06**
f2-60	6-Hump Camel-Back	2	2.3455E-07	9.1217E-05	1.2481E-04	**3.4081E-08**	**1.5711E-05**	**2.2781E-05**

Table 2b. Comparison of results over 30 trails of the original and modified versions of PMB-BBO, where "Best" means the smallest error, "Mean" indicates the mean smallest error, and "StdDev" stands for the standard deviation - (where $n \neq 2$)

Func. #	Function Name	n	Biogeography Based Optimization (BBO)					
			Original Partial Migration Based			Modified Partial Migration Based		
			Best	Mean	StdDev	Best	Mean	StdDev
f1-01	Mineshaft F1	1	6.2998E-01	7.5377E-01	**7.4896E-02**	**4.4885E-01**	**7.4142E-01**	1.0203E-01
f1-02	Mineshaft F2	1	**2.8719E-09**	4.6768E-04	**6.6885E-04**	3.9974E-08	**3.9916E-04**	8.2865E-04
f1-03	Shekel's F1	1	9.8030E-08	**2.8723E-05**	**4.2939E-05**	**6.7958E-09**	1.2670E-04	4.0902E-04
f1-04	Shekel's F2	1	**8.1958E-10**	**2.1546E-05**	**5.2185E-05**	8.7502E-09	5.4862E-05	8.6120E-05
f1-05	Shekel's F3	1	1.7454E-08	7.4624E-05	1.8726E-04	**1.4736E-10**	**5.1717E-05**	**8.7380E-05**
f1-06	Stron.-Zilin.-Shalt.	1	**5.7915E-11**	**1.0865E-07**	**2.4327E-07**	1.0768E-10	2.5748E-07	6.8094E-07
f1-07	Suharev	1	1.5944E-11	**9.3043E-08**	**1.2955E-07**	**1.1613E-11**	1.8424E-07	4.0888E-07
f1-08	Zilinskas F2	1	1.4002E-09	**2.2730E-06**	**5.2141E-06**	**1.0064E-09**	9.9106E-06	2.2533E-05
f3-01	Box-Betts	3	2.3366E-08	1.4450E-06	**1.1712E-06**	**2.3614E-09**	**1.0091E-06**	1.2014E-06
f3-02	Hartman's F1	3	7.6177E-06	5.0374E-04	3.8211E-04	**1.0262E-06**	**3.8549E-05**	**7.4211E-05**
f3-03	Helical Valley	3	8.1299E-02	8.4147E-01	**6.2500E-01**	**2.3847E-03**	**5.2274E-01**	6.8708E-01
f3-04	Levy F8	3	6.4096E-06	1.2455E-03	1.2789E-03	**4.8947E-07**	**5.2271E-05**	**6.1623E-05**
f3-05	Meyer and Roth	3	9.1336E-06	1.0188E-04	**8.3202E-05**	**4.5474E-06**	**7.0623E-05**	9.3791E-05
f3-06	Perm No.1	3	1.6149E-01	1.2030E+00	7.6733E-01	**4.0283E-03**	**5.7516E-01**	**4.9994E-01**
f4-01	Corana (or Ingber)	4	3.5444E+00	8.6925E+01	6.1928E+01	**0.0000E+00**	**1.1788E+01**	**1.2559E+01**
f4-02	Kowalik	4	3.9927E-04	8.0573E-04	3.2529E-04	**2.5735E-04**	**6.2274E-04**	**3.0950E-04**
f4-03	Miele and Cantrell	4	8.0518E-09	**1.2520E-06**	**1.2340E-06**	**1.2172E-10**	1.4661E-06	3.9810E-06
f4-04	Powell's Quartic	4	2.8518E-02	1.8301E+00	2.0186E+00	**5.9062E-03**	**2.6144E-01**	**3.1940E-01**
f4-05	Neumaier F2	4	9.7017E-03	3.9273E-02	2.4992E-02	**6.8860E-03**	**2.8289E-02**	**2.3559E-02**
f4-06	Wood (or Colville)	4	1.5438E+00	7.1389E+00	3.4273E+00	**1.2867E-01**	**1.9675E+00**	**1.3293E+00**
f5-01	AMGM	5	9.8524E-10	1.4399E-07	2.2968E-07	**3.1111E-11**	**4.1812E-09**	**8.7922E-09**
f5-02	Osborne No.1	5	1.1839E-02	**1.1988E-01**	**9.1466E-02**	**1.1325E-02**	1.3801E-01	1.0496E-01
f5-03	SODP	5	7.2433E-07	6.9565E-05	8.3596E-05	**6.3699E-09**	**1.6674E-05**	**2.4309E-05**
f5-04	Styblinski-Tang	5	8.1139E-02	8.6368E-01	8.0724E-01	**3.4485E-03**	**6.1622E-02**	**5.0114E-02**
f6-01	Hartman's F2	6	1.8781E-03	6.9158E-02	6.0706E-02	**7.1304E-04**	**3.8071E-02**	**5.4616E-02**
f6-02	Perm No.2	6	2.9814E-01	1.0219E+00	6.1973E-01	**1.8974E-02**	**5.7573E-01**	**6.0020E-01**
f9-01	ANNs XOR	9	**7.5664E-04**	**5.7103E-03**	3.6158E-03	7.9878E-04	6.8751E-03	**2.2983E-03**
f9-02	Price's Transistor	9	2.5525E+01	1.3334E+02	5.4565E+01	**9.1948E+00**	**9.0453E+01**	**4.1274E+01**
f9-03	Storn's Chebyshev	9	9.7939E+03	5.4462E+04	3.6762E+04	**3.5246E+03**	**2.6181E+04**	**2.2022E+04**
f10-01	Epistatic Michalewicz	10	4.2122E-01	1.2966E+00	6.2987E-01	**1.5307E-01**	**6.7944E-01**	**3.4882E-01**
f10-02	Katsuura	10	2.1784E-01	5.0836E-01	1.4062E-01	**8.0454E-02**	**2.9182E-01**	**1.1279E-01**
f10-03	Odd Square	10	**9.0095E-01**	**1.0167E+00**	4.3942E-02	9.6634E-01	1.0366E+00	**3.0877E-02**
f10-04	Paviani	10	6.2946E-03	2.0765E-02	1.3830E-02	**1.3197E-03**	**5.5884E-03**	**4.0332E-03**
f15-01	Dixon-Price	15	**5.6066E-01**	2.1042E+00	1.3961E+00	7.3983E-01	**1.4613E+00**	**5.4173E-01**
f15-02	Neumaier F3 (or Trid)	15	1.0026E+02	9.5503E+02	**6.7827E+02**	**8.9477E+01**	**8.7253E+02**	7.7831E+02
f15-03	Normalized Rana	15	**2.9324E+01**	5.6653E+01	1.2546E+01	2.9923E+01	**4.6883E+01**	**9.8307E+00**
f17-01	Bent Cigar	17	9.2454E+05	2.6185E+06	1.2377E+06	**5.3826E+05**	**1.3379E+06**	**5.7265E+05**
f17-02	Defl. Corrug. Spring	17	6.2664E-01	1.3368E+00	5.0800E-01	**6.2664E-01**	**9.9217E-01**	**3.9077E-01**
f17-03	Infinity (or Csendes)	17	4.6802E-14	1.6281E-11	2.4605E-11	**4.1422E-15**	**2.4643E-12**	**4.6031E-12**
f20-01	Alpine	20	1.7629E-02	3.6322E-02	1.2919E-02	**1.5741E-02**	**2.8971E-02**	**1.0286E-02**
f20-02	Quintic	20	2.0141E+00	3.2993E+00	7.3579E-01	**1.4342E+00**	**2.6384E+00**	**5.6576E-01**
f20-03	Pathological	20	**1.7044E+00**	2.6537E+00	4.2252E-01	2.1295E+00	**2.5968E+00**	**3.8864E-01**
f30-01	Ackley	30	6.3026E-01	9.9236E-01	2.3065E-01	**5.6622E-01**	**9.3734E-01**	**2.1466E-01**
f30-02	Gen. Griewank	30	8.6708E-01	1.0263E+00	**3.4402E-02**	**8.4581E-01**	**1.0131E+00**	4.7757E-02
f30-03	Gen. Penalized F1	30	5.0934E-03	2.9591E-02	3.2559E-02	**1.6288E-03**	**1.7078E-02**	**1.6484E-02**
f30-04	Gen. Penalized F2	30	9.3001E-02	**1.6875E-01**	**6.0757E-02**	**8.2945E-02**	1.8237E-01	7.5631E-02
f30-05	Gen. Rastrigin	30	9.4594E-01	1.9351E+00	**6.4580E-01**	**7.5521E-01**	**1.7540E+00**	7.1453E-01
f30-06	Gen. Rosenbrock	30	1.0517E+02	2.6058E+02	**7.9555E+01**	**8.5320E+01**	**2.4504E+02**	9.3473E+01
f30-07	Gen. Schwefel F2.26	30	**4.4291E-06**	1.3876E-05	5.7466E-06	4.6460E-06	**1.0766E-05**	**3.8181E-06**
f30-08	Mishra F1	30	1.3229E-01	1.8511E-01	3.4109E-02	**1.1668E-01**	**1.6764E-01**	**2.7709E-02**
f30-09	Mishra F2	30	1.2195E-01	1.9327E-01	3.4082E-02	**1.1973E-01**	**1.6742E-01**	**3.0336E-02**
f30-10	Quartic	30	2.8562E-07	1.2406E-06	1.2872E-06	**4.9473E-08**	**9.6111E-07**	**1.1137E-06**
f30-11	Schwefel F1.2	30	4.2433E+02	1.9527E+04	1.1268E+04	**1.0214E+00**	**2.6767E+02**	**2.9316E+02**
f30-12	Schwefel F2.21	30	**3.2007E+00**	6.2387E+00	1.1469E+00	4.3096E+00	**5.5198E+00**	**9.0466E-01**
f30-13	Schwefel F2.22	30	**4.0024E-01**	**7.0184E-01**	1.3458E-01	4.5480E-01	7.0286E-01	**1.3019E-01**
f30-14	Sphere	30	1.8518E+00	3.8843E+00	1.3964E+00	**1.5508E+00**	**3.3657E+00**	**1.2515E+00**
f30-15	Step	30	2.0000E+00	4.7333E+00	**1.8245E+00**	**0.0000E+00**	**4.1000E+00**	1.8859E+00
f60-01	Hyper-Ellipsoid	60	5.2404E-01	1.0131E+00	3.8482E-01	**5.1024E-01**	**9.0328E-01**	**2.6241E-01**
f60-02	Qing	60	3.5937E+03	**5.9785E+03**	**1.6586E+03**	**3.5094E+03**	6.4488E+03	2.0255E+03
f60-03	Salomon	60	2.2999E+00	3.0172E+00	3.5403E-01	**2.2999E+00**	**2.7949E+00**	**2.5704E-01**

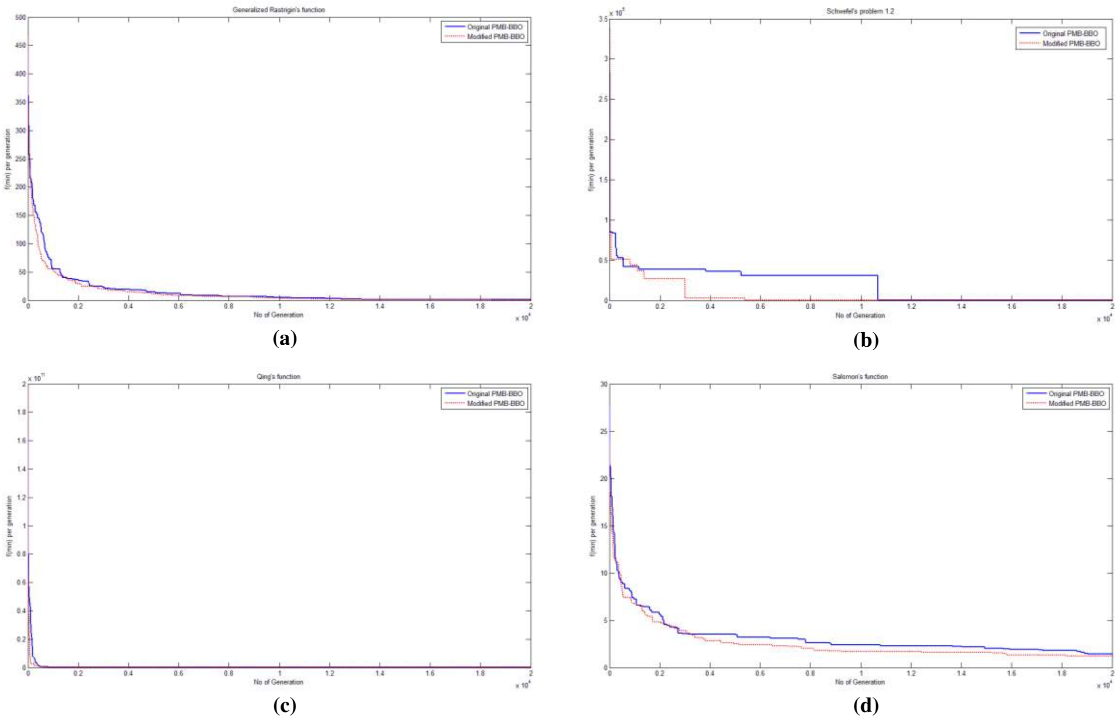


Figure 6. Curves of fitness functions of the original and modified PMB for some selected functions. **(a)** f30-05, **(b)** f30-11, **(c)** f60-02, **(d)** f60-03

Table 3 shows an extended comparison when the mutation stage is not considered, as it has been done in the preceding study between PMB-BBO and SPMB-BBO [20]. Here the modified BBO is the best model.

Table 3. The overall performance of PMB and modified PMB with/without mutation stage

f #	Function Name	n	Mutation Stage	Biogeography Based Optimization (BBO)					
				Partial Migration Based			Modified Partial Migration Based		
				Best	Mean	StdDev	Best	Mean	StdDev
f05	Generalized Rosenbrock	30	Activated	1.0517E+02	2.6058E+02	**7.9555E+01**	**8.5320E+01**	**2.4504E+02**	9.3473E+01
			Not-Activated	1.7036E+02	6.0492E+02	6.4746E+02	**1.6188E+02**	**2.5669E+02**	**7.3475E+01**
f08	Generalized Schwefel F2.26	30	Activated	**4.4291E-06**	1.3876E-05	5.7466E-06	4.6460E-06	**1.0766E-05**	**3.8181E-06**
			Not-Activated	8.1593E-06	2.9257E-05	1.1255E-05	**4.4413E-06**	**1.1069E-05**	**4.1198E-06**
f14	Shekel's Foxholes	2	Activated	**2.1720E-11**	8.0558E-08	2.1865E-07	6.4642E-11	**6.4142E-10**	**5.9986E-10**
			Not-Activated	5.4152E-08	2.2467E-03	8.6667E-03	**5.2873E-11**	**3.9904E-08**	**1.3873E-07**
f18	Goldstein-Price	2	Activated	2.7778E-05	1.5749E-03	1.6278E-03	**3.1758E-06**	**2.7174E-04**	**4.1521E-04**
			Not-Activated	1.4627E-04	2.5692E-02	2.2876E-02	**4.3193E-06**	**1.0116E-03**	**1.1351E-03**

Table 4 shows the CPU time comparison between the original PMB-BBO and the modified PMB-BBO. It can be clearly seen that the modified version can save around 32.32% of the CPU time, which means that it is faster than even the simplified partial migration model (SPMB-BBO) in [20] by around 24.76%.

5.1. Discussions

As a comparison between the four original models of BBO (PMB, SMB, SPMB and SSMB), PMB-BBO gives the best performance when the given problem is hard, has large upper and lower limits of search space, high-dimensional and/or the number of islands or population size is small [20]. However, PMB-BBO lacks the exploration [18]. Therefore, in this study, the root problem that causes the poor exploration is solved by using an integer random function which provides a pseudorandom integers from a uniform discrete distribution. Furthermore, the exploitation is improved by keeping the non-mutated solutions away from any corruption by clear duplication process.

Table 4. Normalized CPU times on 60-dimensional test functions

Function	PMB-BBO Versions		CPU Time Saving (%)
	Original ver.	Modified ver.	
Hyper-Ellipsoid	1.4617E+00	**1.0000E+00**	31.58437697
Qing	1.4605E+00	**1.0000E+00**	31.5305639
Salomon	1.5104E+00	**1.0000E+00**	33.79377207
Avg. CPU Time	1.4775E+00	**1.0000E+00**	32.3195104

6. Conclusions and Suggestions

This study proposed some modifications to improve the performance of the original form of PMB-BBO without using any complicated models for immigration and emigration rates. It is shown that the simplified linear model still can give good results if the root problems of the migration and mutation stages are solved. An extensive testing of the original and the proposed modified versions of PMB-BBO through 120 test functions shows that the performance of the modified version of PMB-BBO is better on Best, Mean, and StdDev than that of the original version.

The proposed modification can be used as a basis for modifying the existing modified BBOs in literature. For example, the blended-BBO which is presented in [14] can be a great add-on to this modified PMB-BBO. Furthermore, even if it is required to employ the complicated models of immigration and emigration rates, such as the generalized sinusoidal migration model in [16]. The generalized sinusoidal migration model is based on an old study that was done by James A. MacMahon in 1987 [19], and was mentioned in ch.3 of [4]. This complicated model shows a great performance in [16], and it is very interesting to re-test this model with considering the proposed essential modifications that are described in this study.

Appendix

This appendix contains a complete list of all the involved test functions that are collected from various sources where some of them are hard to be found while the other, especially the old functions, are corrected analytically before being used here. This is because they are available in their original sources with approximated global solutions. Perhaps due limited computing capability in that time. These 120 test functions are spread through references [22-64], and some of these references have a collection of test functions, which means that the popular test functions can be found in different locations, while the others are located in one or few locations. In addition, sometimes the information about test functions are available in different references. For more information regarding any test function, please refer to the related reference(s).

f #	Function Name	Dimension (n)	Variables Bounds	Global Optimum	References
f1-01	Mineshaft F1	1	$0 \leq x \leq 10$	1.380487165157852	[63]
f1-02	Mineshaft F2	1	$-10 \leq x \leq 10$	−1.416353520337699	[63]
f1-03	Shekel's F1	1	$0 \leq x \leq 10$	−10.1531987550848817763568 39400251	[23,24]

f1-04	Shekel's F2	1	$0 \le x \le 10$	$-10.402822044707775329070518200751$	[23,24]
f1-05	Shekel's F3	1	$0 \le x \le 10$	$-10.536290299294717105427357601002$	[23,24]
f1-06	Strongin-Zilinskas-Shaltyanis	1	$3 \le x \le 7.5$	-1.601307546494396	[25]
f1-07	Suharev	1	$0 \le x \le 1$	-1	[25]
f1-08	Zilinskas F2	1	$0 \le x \le 1$	-1.125	[25]
f2-01	Aluffi-Pentini	2	$-10 \le x_i \le 10$	-0.352386073800034	[26]
f2-02	Banana Shape	2	$-1.5 \le x_1 \le 1.5$ $-2.5 \le x_2 \le 0.5$	-25	[27]
f2-03	Beale	2	$-4.5 \le x_i \le 4.5$	0	[28,50]
f2-04	Becker-Lago	2	$-10 \le x_i \le 10$	0	[26]
f2-05	Bird	2	$-2\pi \le x_i \le 2\pi$	-106.7645367198034	[29,33,34]
f2-06	Bohachevsky F1	2	$-50 \le x_i \le 50$	0	[26,30]
f2-07	Bohachevsky F2	2	$-50 \le x_i \le 50$	0	[26,30]
f2-08	Bohachevsky F3	2	$-50 \le x_i \le 50$	0	[30]
f2-09	Booth	2	$-10 \le x_i \le 10$	0	[28,31]
f2-10	Branin RCOS	2	$-5 \le x_1 \le 10$ $0 \le x_2 \le 15$	0.39788735772973816	[31,32]
f2-11	Bukin F4	2	$-15 \le x_i \le 5$	0	[33]
f2-12	Bukin F6	2	$-15 \le x_i \le 5$	0	[33]
f2-13	Carrom Table	2	$-10 \le x_i \le 10$	-24.15681551650653	[34]
f2-14	Chichinadze	2	$-30 \le x_i \le 30$	-42.94438701899098	[34]
f2-15	Complex	2	$-2 \le x_i \le 2$	0	[35]
f2-16	Cosine Mixture	2	$-1 \le x_i \le 1$	0.2	[38]
f2-17	Cross In Tray	2	$-15 \le x_i \le 15$	-2.062611870822739	[33,34]
f2-18	Cross Leg Table	2	$-10 \le x_i \le 10$	-1	[33,34]
f2-19	Crowned Cross	2	$-10 \le x_i \le 10$	0.0001	[33,34]
f2-20	Davis	2	$-100 \le x_i \le 100$	0	[35]
f2-21	Decanomial	2	$-10 \le x_i \le 10$	0	[34]
f2-22	Dekkers-Aarts	2	$-20 \le x_i \le 20$	-24777	[26]
f2-23	Drop Wave	2	$-5.12 \le x_i \le 5.12$	-1	[34,36]
f2-24	Easom	2	$-10 \le x_i \le 10$	-1	[26,31,32,36,37]
f2-25	Egg Holder	2	$-512 \le x_i \le 512$	$-959.640662711\ (for\ n = 2)$	[31,33,34]
f2-26	EXP2	2	$0 \le x_i \le 20$	0	[31,34]
f2-27	Freudenstein-Roth	2	$-10 \le x_i \le 10$	0	[39,50]
f2-28	Giunta	2	$-1 \le x_i \le 1$	0.06447042053690566 $(for\ n = 2)$	[33,34]
f2-29	Goldstein-Price	2	$-2 \le x_i \le 2$	3	[26,31,32,34,35,36,40,41]
f2-30	Himmelblau	2	$-6 \le x_i \le 6$	0	[25,31,34,35,41,43]
f2-31	Holder Table	2	$-10 \le x_i \le 10$	-19.20850256788675	[33,34]
f2-32	Hosaki	2	$0 \le x_i \le 10$	-2.345811576101292	[26,31,34]
f2-33	Kearfott	2	$-3 \le x_i \le 4$	0	[43,44,45]
f2-34	Inverted Cosine Wave	2	$-5 \le x_i \le 5$	$-n + 1$	[37]
f2-35	Levy F3 (Shubert or Hansen)	2	$-10 \le x_i \le 10$	-176.5417931365915	[26,31,42]
f2-36	Levy F5	2	$-10 \le x_i \le 10$	-176.1375	[31,46]
f2-37	Matyas	2	$-10 \le x_i \le 10$	0	[28,31,34,37]
f2-38	McCormick	2	$-1.5 \le x_1 \le 4$ $-3 \le x_2 \le 4$	-1.913222954981037	[26,31,33,34]
f2-39	Michalewicz	2	$0 \le x_i \le \pi$	-1.801303228593281 $(for\ n = 2)$	[36,37]
f2-40	Muller-Brown Surface	2	$-1.5 \le x_1 \le 1$ $-0.5 \le x_2 \le 2.5$	-146.6995172099539	[47,48]
f2-41	Parsopoulos	2	$-5 \le x_i \le 5$	0	[41]
f2-42	Peaks	2	$-4 \le x_i \le 4$	-6.551133332622496	[49]
f2-43	Pen Holder	2	$-11 \le x_i \le 11$	-0.9635348327265058	[33,34]
f2-44	Powell's Badly Scaled	2	$-10 \le x_i \le 10$	0	[50,51,52]
f2-45	Sawtoothxy	2	$-20 \le x_i \le 20$	0	[49,54]
f2-46	Schaffer's F1	2	$-100 \le x_i \le 100$	0	[26,31,33,48]
f2-47	Schaffer's F2	2	$-100 \le x_i \le 100$	0	[26,31,33,48]
f2-48	Shekel's Foxholes	2	$-65.536 \le x_i \le 65.536$	0.998003837794449325873406851315	[23,36,54]
f2-49	Sinusoidal Problem	2	$0 \le x_i \le 180°$	-3.5	[26]
f2-50	Stenger	2	$-1 \le x_i \le 4$	0	[35]
f2-51	Storn	2	$-4 \le x_i \le 4$	-18.0587	[41]
f2-52	Stretched V	2	$-10 \le x_i \le 10$	$0\ (for\ n = 2)$	[31,34]
f2-53	Test Tube Holder	2	$-10 \le x_i \le 10$	-10.872299901558	[33,34]
f2-54	Treccani	2	$-5 \le x_i \le 5$	0	[34,55]
f2-55	Trefethen F4	2	$-6.5 \le x_1 \le 6.5$ $-4.5 \le x_2 \le 4.5$	-3.3068686474	[31,34,54]
f2-56	Tripod	2	$-100 \le x_i \le 100$	0	[37]
f2-57	Zakharov	2	$-5 \le x_i \le 10$	0	[32,37]
f2-58	Zettl	2	$-1 \le x_i \le 5$	$-0.00379123722046865 6$	[31,33,34]
f2-59	3-Hump Camel-Back	2	$-5 \le x_i \le 5$	0	[26,31,33,40]

f2-60	6-Hump Camel-Back	2	$-5 \leq x_i \leq 5$	-1.031628453489877	[23,26,31,34,36, 37,40]
f3-01	Box-Betts	3	$0.9 \leq x_1, x_3 \leq 1.2$ $9 \leq x_2 \leq 11.2$	0	[31,34]
f3-02	Hartman's F1	3	$0 \leq x_i \leq 1$	-3.86278214782076	[23,26,30,31,32, 34,37,40]
f3-03	Helical Valley	3	$-100 \leq x_i \leq 100$	0	[26,34,39,50,52, 56]
f3-04	Levy F8	3	$-10 \leq x_i \leq 10$	0	[31,41]
f3-05	Meyer and Roth	3	$-20 \leq x_i \leq 20$	0.4×10^{-4}	[26]
f3-06	Perm F1	3	$-n \leq x_i \leq n+1$	0	[34,55]
f4-01	Corana (or Ingber)	4	$-100 \leq x_i \leq 100$	0	[31,34,57]
f4-02	Kowalik	4	$-5 \leq x_i \leq 5$	0.00030748598780560421684 04344971009	[23,26,30,31,34, 37]
f4-03	Miele and Cantrell	4	$-1 \leq x_i \leq 1$	0	[26]
f4-04	Powell's Quartic	4	$-10 \leq x_i \leq 10$	0	[26,50]
f4-05	Neumaier F2	4	$0 \leq x_i \leq n$	0	[26,34,55]
f4-06	Wood (or Colville)	4	$-10 \leq x_i \leq 10$	0	[30,31,37,50,58]
f5-01	AMGM	5	$0 \leq x_i \leq 10$	0	[34]
f5-02	Osborne F1	5	$0 \leq x_1, x_2, x_4, x_5 \leq 3$ $-3 \leq x_3 \leq 0$	5.46×10^{-5}	[63,64]
f5-03	SODP	5	$-1 \leq x_i \leq 1$	0	[34,36,37]
f5-04	Styblinski-Tang	5	$-5 \leq x_i \leq 5$	$-39.16616570377142n$	[33,34]
f6-01	Hartman's F2	6	$0 \leq x_i \leq 1$	-3.32236801141551	[23,26,30,31,32, 34,37,40]
f6-02	Perm F2	6	$-1 \leq x_i \leq 1$	0	[55]
f9-01	ANNs XOR	9	$-1 \leq x_i \leq 1$	0.959759	[55]
f9-02	Price's Transistor	9	$-10 \leq x_i \leq 10$	0	[26]
f9-03	Storn's Chebyshev	9	$-2^n \leq x_i \leq 2^n$	0	[26,59]
f10-01	Epistatic Michalewicz	10	$0 \leq x_i \leq \pi$	-9.660152	[26,59]
f10-02	Katsuura	10	$-1000 \leq x_i \leq 1000$	$1\ (for\ n = 10)$	[57]
f10-03	Odd Square	10	$-15 \leq x_i \leq 15$	$-1.143833\ (for\ n = 10)$	[26,31]
f10-04	Paviani	10	$-2.001 \leq x_i \leq 9.999$	-45.77848	[26,31,54,60]
f15-01	Dixon-Price	15	$-10 \leq x_i \leq 10$	0	[28,34]
f15-02	Neumaier F3 (or Trid)	15	$-n^2 \leq x_i \leq n^2$	$-665\ (for\ n = 15)$	[26,31,34,48]
f15-03	Normalized Rana's Function + Diagonal Wrap	15	$-520 \leq x_i \leq 520$	$-512.753162426239100568636$ 786193	[60,65]
f17-01	Bent Cigar	17	$-100 \leq x_i \leq 100$	0	[34]
f17-02	Deflected Corrugated Spring	17	$0 \leq x_i \leq 10$	-1	[22,58]
f17-03	Infinity (or Csendes)	17	$-1 \leq x_i \leq 1$	0	[34]
f20-01	Alpine	20	$-10 \leq x_i \leq 10$	0	[34,37]
f20-02	Quintic	20	$-10 \leq x_i \leq 10$	0	[34,55]
f20-03	Pathological	20	$-100 \leq x_i \leq 100$	0	[37]
f30-01	Ackley	30	$-32 \leq x_i \leq 32$	0	[23,26,28,30,31, 34,36,37,46,57]
f30-02	Generalized Griewank	30	$-600 \leq x_i \leq 600$	0	[23,26,28,30]
f30-03	Generalized Penalized F1	30	$-50 \leq x_i \leq 50$	0	[23,62]
f30-04	Generalized Penalized F2	30	$-50 \leq x_i \leq 50$	0	[23,62]
f30-05	Generalized Rastrigin	30	$-5.12 \leq x_i \leq 5.12$	0	[22,23,26,28,30, 31,36,37,46,53,57, 62]
f30-06	Generalized Rosenbrock	30	$-30 \leq x_i \leq 30$	0	[23,26,28,31,32, 36,37,46,62]
f30-07	Generalized Schwefel F2.26	30	$-500 \leq x_i \leq 500$	$-418.982887272433799807913$ $601398n$	[22,23,26,28,30, 31,34,36,53,60,62, 65]
f30-08	Mishra F1	30	$0 \leq x_i \leq 1$	2	[34]
f30-09	Mishra F2	30	$0 \leq x_i \leq 1$	2	[34]
f30-10	Quartic (or De Jong's F4)	30	$-1.28 \leq x_i \leq 1.28$	0	[23,31,37,46,56]
f30-11	Schwefel F1.2	30	$-100 \leq x_i \leq 100$	0	[23,31,37,53,62]
f30-12	Schwefel F2.21	30	$-100 \leq x_i \leq 100$	0	[23,31,37,62]
f30-13	Schwefel F2.22	30	$-10 \leq x_i \leq 10$	0	[23,30,31,37,53, 62]
f30-14	Sphere (Square Sum, Harmonic or De Jong's F1)	30	$-100 \leq x_i \leq 100$	0	[23,28,31,36,37, 46,53,57,62]
f30-15	Step	30	$-100 \leq x_i \leq 100$	0	[23,28,31,37,53, 57]
f60-01	Hyper-Ellipsoid	60	$-1 \leq x_i \leq 1$	0	[57]
f60-02	Qing	60	$-500 \leq x_i \leq 500$	0	[53]
f60-03	Salomon	60	$-100 \leq x_i \leq 100$	0	[26,53]

REFERENCES

[1] Dan. Simon, "Biogeography-based optimization," *IEEE Trans. on Evolutionary Computation*, vol. 12, no. 6, pp. 702-713, Dec. 2008.

[2] Robert H. MacArthur and Edward O. Wilson, "An Equilibrium Theory of Insular Zoogeography," *Journal of Evolution*, vol. 17, no. 4, pp. 373-387, Dec. 1963.

[3] Robert H. MacArthur and Edward O. Wilson, *The Theory of Island Biogeography*. Princeton, New Jersey: Princeton University Press, 1967.

[4] Mark V. Lomolino, Brett R. Riddle and James H. Brown, *Biogeography*, 3rd ed. Sunderland, Massachusetts: Sinauer Associates Inc., 2009.

[5] R. J. Lincoln, G. A. Boxshall and P. F. Clark, *A dictionary of Ecology, Evolution and Systematics*. Cambridge, UK: Cambride University Press, 1982.

[6] Paulami Maiti and Prabodh K. Maiti, *Biodiversity: Perception, Peril and Preservation*. Prentice-Hall of India, 2011.

[7] William A. Nierenberg, *Encyclopedia of Environmental Biology*, vol. 2. San Diego, California: Academic Press Inc., 1995.

[8] Jonathan B. Losos and Robert E. Ricklefs, *The Theory of Island Biogeography Revisited*. Princeton, New Jersey: Princeton University Press, 2010.

[9] Alan A. Myers and Paul S. Giller, *Analytical Biogeography: An Integrated Approach to the Study of Animal and Plant Distributions*. London, UK: Chapman and Hall, 1990.

[10] Martin L. Cody, *Plants on Islands: Diversity and Dynamics on a Continental Archipelago*. Berkeley and Los Angeles, California: University of California Press, 2006.

[11] Robert H. MacArthur and Joseph H. Connell, *The Biology of Populations*. New York: John Wiley & Sons Inc., 1966.

[12] Robert H. MacArthur, *Geographical Ecology: Patterns in the Distribution of Species*. New York: Harper & Row Publishers Inc., 1972.

[13] Haiping Ma, "An analysis of the equilibrium of migration models for biogeography-based optimization," *Information Sciences*, vol. 180, no. 18, pp. 3444-3464, 15 Sept. 2010.

[14] Haiping Ma and Dan. Simon, "Blended biogeography-based optimization for constrained optimization," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 3, pp. 517-525, Apr. 2011.

[15] Dan. Simon. (2009, Feb.). A Probabilistic Analysis of a Simplified Biogeography-Based Optimization Algorithm. Cleveland State University. Cleveland, Ohio. Available: http://academic.csuohio.edu/simond/bbo/simplified/bbosimplified.pdf

[16] Haiping Ma and Dan. Simon, "Analysis of migration models of biogeography-based optimization using Markov theory," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 6, pp. 1052-1060, Sept. 2011.

[17] Dan. Simon, M. Ergezer and Dawei Du, "Population distributions in biogeography-based optimization algorithms with elitism," *IEEE International Conference on Systems, Man and Cybernetics,* pp. 991-996, Oct. 2009.

[18] Wenying Gong, Zhihua Cai and Charles X. Ling, "DE/BBO: A hybrid differential evolution with biogeography-based optimization for global numerical optimization," *Soft Computing*, vol. 15, no. 4, pp. 645-665, Apr. 2011.

[19] James A. McMahon, "Disturbed lands and ecological theory: an essay about a mutualistic association," in: *Restoration Ecology, a Synthetic Approach to Ecological Research*, William R. Jordan III, Michael E. Gilpin and John D. Aber. Cambridge, UK: Cambride University Press, 1987, pp. 221–237.

[20] Ali R. Alroomi, Fadhel A. Albasri and Jawad H. Talaq, “Performance Comparison between the Original Forms of Biogeography-Based Optimization,” *Second International Conference on Advanced Information Technologies and Applications (ICAITA-2013)*, pp. 121-140, Nov. 2013.

[21] Dan. Simon, "The Matlab Code of Biogeography-Based Optimization," [online] Aug. 2008, http://academic.csuohio.edu/simond/bbo/ (Accessed: 01 Feb. 2013).

[22] Wen Wan and Jeffrey B. Birch, "Using a modified genetic algorithm to find feasible regions of a desirability function," *Quality and Reliability Engineering International*, vol. 27, no. 8, pp. 1173-1182, Dec. 2011.

[23] Xin Yao, Yong Liu and Guangming Lin, "Evolutionary Programming Made faster," *IEEE Trans. on Evolutionary Computation*, vol. 3, no. 2, pp. 82-102, Jul. 1999.

[24] Chien-Wen Chao, Shu-Cherng Fand and Ching-Jong Liao, "A Tropical cyclone-based method for global optimization," *Journal of Industrial and Management Optimization*, Vol. 8, no. 1, Feb. 2012.

[25] Elena Kiseleva and Tatyana Stepanchuk, "On the Efficiency of a Global Non-differentiable Optimization Algorithm Based on the Method of Optimal Set Partitioning," *Journal of Global Optimization*, vol. 25, no. 2, pp. 209–235, Feb. 2003.

[26] M. Montaz Ali, Charoenchai Khompatraporn and Zelda B. Zabinsky, "A Numerical Evaluation of Several Stochastic Algorithms on Selected Continuous Global Optimization Test Problems" *Journal of Global Optimization*, vol. 31, no. 4, pp. 635-672, Apr. 2005.

[27] Ray-Bing Chen, Weichung Wang and Fuhuang Tsai, "A Basis-based Response Surface Method for Computer Experiment Optimization," 2006

[28] Milos Subotic, Milan Tuba and Nadezda Stanarevic, "Different approaches in parallelization of the artificial bee colony algorithm," *International Journal of Mathematical and Methods in Applied Sciences*, vol. 5, no. 4, Mar. 2011.

[29] Celso A. G. Santos, Paula K. M. M. Freire and Sudhanshu K. Mishra, "Cuckoo search via Lévy Flights for optimization of a physically-based runoff-erosion model," *Journal of Urban and Environmental Engineering*, vol. 6, no. 2 , pp. 123-131, Dec. 2012

[30] Xinchao Zhao and Xiao-Shan Gao, "Affinity genetic algorithm," *Journal of Heuristics*, vol. 13, no. 2, pp. 133-150, Apr. 2007.

[31] Ernesto P. Adorio. (2005, Jan.), MVF - Multivariate Test Functions Library in C for Unconstrained Global Optimization. University of the Philippines Diliman. Quezon City, Metro Manila, Philippines. Available: http://geocities.ws/eadorio/mvf.pdf

[32] Rachid Chelouah and Patrick Siarry, "Tabu Search applied to global optimization," *European Journal of Operational Research*, vol. 123, no. 2, pp. 256–270, Jun. 2000.

[33] Sudhanshu Mishra. (2006, Aug.). Some new test functions for global optimization and performance of repulsive particle swarm method. North-Eastern Hill University. Shillong, India. Available: http://mpra.ub.uni-muenchen.de/2718/1/MPRA_paper_2718.pdf

[34] Andrea Gavana, "Test Functions Index," [online] Feb. 2013, http://infinity77.net/global_optimization/test_functions.html (Accessed: 01 April 2013).

[35] Z. Emami, M. H. Farahi and E. Ansari. (2012, Feb.). Global Optimization with Electro Weak. *The Open Industrial and Manufacturing Engineering Journal*. [Online]. vol. 5, pp. 1-7. Available: http://benthamscience.com/open/toimej/articles/V005/1TOIMEJ.pdf

[36] Marcin Molga and Czesław Smutnicki. (2005, Apr.). Test functions for optimization needs. North-Eastern Hill University. Shillong, India. Available: http://www.zsd.ict.pwr.wroc.pl/files/docs/functions.pdf

[37] Shahryar Rahnamayan, Hamid R. Tizhoosh and Magdy M. A. Salama, "A novel population initialization method for accelerating evolutionary algorithms," *Computers & Mathematics with Applications*, vol. 53, no. 10, pp. 1605–1614, May 2007

[38] Leo Breiman and Adele Cutler, "A deterministic algorithm for global optimization," *Mathematical Programming*, vol. 58, no.1-3, pp. 179-199, Jan. 1993.

[39] Mohamed B. Trabia and Xiao Bin Lu, "A Fuzzy Adaptive Simplex Search Optimization Algorithm," *Journal of Mechanical Design*, vol. 123, no. 2, pp. 1-10, Jun. 2001.

[40] J. B. Lee and B. C. Lee, "A global optimization algorithm based on the new filled function method and the genetic algorithm," *Engineering Optimization*, vol. 27, no. 1, pp. 1-20, 1996.

[41] M. Fernanda P. Costa, Florbela P. Fernandes, Edite Manuela da G. P. Fernandes, "A deterministic-stochastic method for nonconvex MINLP problems," *2nd International Conference on Engineering Optimization*, Sept. 6-9, 2010, Lisbon, Portugal.

[42] Kaj Madsen, "Test problems for global optimization," [online] Jun. 2008, http://www2.imm.dtu.dk/~kajm/Test_ex_forms/test_ex.html (Accessed: 09 April 2013).

[43] M. N. Vrahatis, D. G. Sotiropoulos and E. C. Triantafyllou, "Global optimization for imprecise problems," in: *Developments in Global Optimization, Nonconvex Optimization and Its Applications*, Immanuel M. Bomze, Tibor Csendes, Reiner Horst, Panos M. Pardalos. Springer Science+Business Media Dordrecht: Kluwer Acdemic Publishers, 1997, vol. 18, pp. 37-54.

[44] Baker Kearfott, "An efficient degree-computation method for a generalized method of bisection," *Numerische Mathematik*, vol. 32, no. 2, pp. 109-127, 1979.

[45] Esin Onbaşoğlu and Linet Özdamar, "Parallel Simulated Annealing Algorithms in Global Optimization," *Journal of Global Optimization*, vol. 19, no. 1, pp. 27-50, Jan. 2001.

[46] S. Rahnamayan, H. R. Tizhoosh and M. M. A. Salama, "Opposition-Based Differential Evolution for Optimization of Noisy Problems," *IEEE Congress on Evolutionary Computation, CEC 2006*, pp.1865-1872, Jul. 2006.

[47] Klaus Müller and Leo D. Brown, "Location of saddle points and minimum energy paths by a constrained simplex optimization procedure," *Theoretica chimica acta*, vol. 53, no. 1, pp. 75-93, 1979.

[48] Wan-Ching Chou, "Multiple Optima Problems in Higher Dimensions," [online] Mar. 2013, http://www.math.ntu.edu.tw/~wwang/cola_lab/test_problems/multiple_opt/multiopt_prob/ (Accessed: 25 April 2013).

[49] MathWorks, "Global Optimization Toolbox," [online] 2013, http://www.mathworks.com/products/global-optimization/index.html (Accessed: 25 April 2013).

[50] Singiresu S. Rao, *Engineering Optimization, Theory and Practice*, 4th ed. Hoboken, New Jersey: John Wiley & Sons Inc., 2009.

[51] Gisela C.V. Ramadas and Edite M.G.P. Fernandes, "Solving Nonlinear Equations by a Tabu Search Strategy," *11th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE2011*, vol. 4, no. 1, pp. 1578-1589, Jun. 2011.

[52] Jorge J. Moré, Burton S. Garbow and Kenneth E. Hillstrom, "Testing Unconstrained Optimization," *ACM Transactions on Mathematical Software (TOMS)*, vol. 7, no. 1, pp. 17-41, Mar. 1981.

[53] Anyong Qing, *Differential Evolution, fundamentals and Applications in Electrical Engineering*. Singapore: Wiley-IEEE Press, Sept. 2009.

[54] Muhammad Aria, "Educational Simulator for Teaching of Particle Swarm Optimization in LabVIEW," *TELEKONTRAN*, vol. 1, no. 1, Jan. 2013.

[55] Sudhanshu Mishra. (2006, Oct.). Repulsive Particle Swarm Method on Some Difficult Test Problems of Global Optimization. North-Eastern Hill University. Shillong, India. Available: http://mpra.ub.uni-muenchen.de/1742/1/MPRA_paper_1742.pdf

[56] V. Zitko. (1984, Sept.), "Simplex Optimization," Canadian Technical Report of Fisheries and Aquatic Sciences, No. 1308. Fisheries and Oceans. Biological Station (St. Andrews, N.B.), Canada. Available: http://www.dfo-mpo.gc.ca/Library/20701.pdf

[57] Rainer Storn and Kenneth Price, "Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341-359, Dec. 1997.

[58] Ashok D. Belegundu and Tirupathi R. Chandrupatla, *Optimization Concepts and Applications in Engineering*, 2nd ed. New York: Cambridge University Press, 2011. Page331

[59] Kenneth V. Price, Rainer M. Storn and Jouni A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*, Berlin, Germany: Springer Science+Business Media, 2005.

[60] Sudhanshu Mishra. (2012, Aug.). Global optimization of some difficult benchmark functions by cuckoo-host co-evolution meta-heuristics. North-Eastern Hill University. Shillong, India. Available: http://mpra.ub.uni-muenchen.de/40666/1/MPRA_paper_40666.pdf

[61] Darrell Whitley, Deon Garrett and Jean-Paul Watson, "Quad Search and Hybrid Genetic Algorithms," in: *Genetic and Evolutionary Computation—GECCO 2003*, G. Goos, J. Hartmanis and J. Van Leeuwen. Berlin, Germany: Springer Science+Business Media, 2003, vol. 2, pp. 1469-1480.

[62] A. K. Qin, V. L. Huang and P. N. Suganthan, "Differential Evolution Algorithm with Strategy Adaptation for Global Numerical Optimization," *IEEE Trans. On Evolutionary Computation*, vol. 13, no. 2, pp. 398 - 417, Apr. 2009.

[63] David R. Monismith Jr., "The Uses of the Slime Mold Lifecycle as a Model for Numerical Optimization," Ph.D. dissertation, Dept. Elect. Eng., Oklahoma State University, Oklahoma City, OK, 2010.

[64] Debasis Kundu and Amit Mitra, "Estimating the parameters of the linear compartment model," *Journal of Statistical Planning and Inference,* vol. 70, no. 2, pp. 317-334, May 1998.

[65] Lappeenranta University of Technology, "The function testbed," [online] May. 2007, http://www.it.lut.fi/ip/evo/functions/functions.html (Accessed: 10 April 2013).

Authors

Ali Ridha Al-Roomi received the B.Sc. degree in Process Instrumentation and Control Engineering from University of Bahrain, Bahrain, in 2006. In the period between 2009-2013, he covered the electrical courses of the B.Sc. program in Electrical Engineering, and followed by M.Sc. degree from the same university. After B.Sc. graduation, he was selected to be as a project engineer in Moore Control & Engineering “MCE Middle East” till the mid of 2007, when he joined Yokogawa Middle East as a DCS subsystem and graphics engineer. From 2008 to 2012, he worked in Aluminum Bahrain (ALBA B.SC.) as an instrumentation and control engineer in its power plants, generation and auxiliary C&I maintenance. He is currently working as a researcher in the field of electrical power systems and evolutionary computation.

Fadhel Abbas Al-Basri received the B.Sc. and M.Sc. degrees in Electrical Engineering from University of Bahrain, Bahrain, and Ph.D. degree in the same field from University of Western Ontario, Canada, in 1992, 1997 and 2007, respectively. He worked in Ministry of Electricity and Water, Bahrain, as an electrical engineer from 1993 to 1994. In 1994, he joined the University of Bahrain as teaching and research assistant and currently an assistant professor in the department of electrical engineering. His research interest is power systems protection, power systems analysis and FACTS-devices.

Jawad Hasan Talaq received B.Sc. degree from University of Technology, Baghdad, Iraq (1981), M.Sc. degree from University of Strathclyde, Glasgow, UK (1987), and Ph.D. degree from Technical University of Nova Scotia, Halifax, Canada (1993); all in electrical engineering. He worked in Ministry of Electricity and Water, Bahrain, as a graduate engineer in Riffa power plant from 1981 to 1985, and as a shift charge engineer from 1985 to 1989, when he moved from industrial field to academic field. He is now an associate professor in the department of electrical engineering, University of Bahrain. His teaching covers power systems operation and control, power systems dynamics, and advanced power systems analysis; and his research topics include power systems operation, control and dynamics, and applied soft computing.