# Multiple-Gbest Guided Artificial Bee Colony Optimization Algorithm

Abdul Ghani Abro, and Junita Mohamad-Saleh,

*Abstract*— Artificial Bee Colony (ABC) applies foraging phenomenon of honeybees for evolving the optimal solutions of problems at hand. ABC algorithm is one of many prominent bio-inspired optimization algorithms. Moreover, ABC algorithm is capable on inducting new possible-solutions while the algorithm searches for the optimal solutions. Scout-bee is responsible for inducting the possible-solutions. Nonetheless, ABC suffers from slow convergence and poor exploitation. Furthermore, the scout-bee selects completely random possible-solution to replace the abandoned possible-solution. Hence, there are very little chances of getting higher-quality possible-solution. Therefore, researchers have proposed various solutions to overcome the demerits of ABC algorithm. Nevertheless, the variants are either computationally intensive or could not avert the flaws. This research work proposes two different modifications for the performance enhancement of ABC algorithm. The first modification enhances the neighborhood searching capability of ABC algorithm, whereas, the second modification improves searching capability of the scout-bee by exploiting the so-far best-found possible-solution. The proposed algorithm has been compared with various existing variants of ABC algorithm on a wide set of high-dimensional benchmark functions. The significance of the proposed algorithm has been proved by a statistical test. The results reveal that the proposed algorithm has produced the significantly better convergence in comparison to the compared algorithms.

***Keywords*—**ABC variant, Evolutionary algorithms, Metaheuristic algorithms, Stochastic algorithms, Computational intelligence

## I. Introduction

As various real-world optimization problems are becoming severely difficult, global optimization using traditional techniques is becoming an immensely challenging assignment [1]. Computational intelligence (CI), a driven branch of Artificial Intelligence (AI), is an area which solves real-world problems, which do not have straight-to-the-point solutions [2]. CI has been successfully applied in various disciplines [3-7]. Bio-inspired optimization algorithms are a subfield of CI. Bio-inspired optimization algorithms generate the optimal solutions using principles of various natural phenomena [8-12]. The algorithms are also known as population-based algorithms or meta-heuristic algorithms.

Junita Mohamad-Saleh (Corresponding author email; jms@eng.usm.my, Fax no. +6045941023, Phone no. +6045996027) is with the school of Electrical and Electronic Engineering, Universiti Sains Malaysia; 14300 Nibong Tebal, Seberang Perai Selatan; Penang, Malaysia.

Abdul Ghani Abro (aga10_eee097@student.usm.my) is also with the school of Electrical and Electronic Engineering, Universiti Sains Malaysia.

Artificial Bee Colony (ABC) algorithm, proposed in 2007, generates the optimal solutions using foraging strategy of honeybees [13]. ABC algorithm has outperformed various optimization algorithms such as differential evolution, particle swarm optimization, evolutionary strategies on a wide set of benchmark functions [14-17]. Standard-ABC has not only been able to yield better performance but it also has fewer parameters to tune [14, 16]. ABC optimization algorithm has been applied in a number of engineering applications [18-22]. In-spite of better results, ABC algorithm suffers from few demerits like poor exploitation capability [23], slow convergence on unimodal problems [24, 25] and can easily be trapped local optima while handling complex multimodal problems [26].

To overcome the flaws of ABC algorithm, [1, 24-31] are the well-assessed proposed variants of the standard-ABC algorithm. Variants proposed in [1, 29] have used rosenbrock and chaotic-map techniques for the performance enhancement of the standard-ABC algorithm however, both the variants have been outperformed by a variant proposed in [25]. Another variant proposed in [24] have yielded better performance than the standard-ABC algorithm only on very few benchmark functions. Results given in the respective research works reveal that the variants proposed in [27, 28] also have been outperformed by ABC variant proposed in [25] reference. Nevertheless, an efficient bio-inspired optimization algorithm is required to have the balanced exploration and exploitation capabilities for yielding better performance. The variant proposed in [25] has the better exploration capability than the exploitation. Therefore, the variant has capability to avert local-optima trappings but has slower convergence. Moreover, variants proposed in [24, 25, 27] carry additional control-variables and there is no systematic way to determine the optimal value of the control-variables except trial-and-error. The variants proposed in [26, 30] generate solutions only around the global-best possible-solution and hence, are prone to local-optima trappings [8].

This research work proposes an efficient variant of ABC algorithm, which converges faster and have the capability to avoid local-optima trappings simultaneously. Furthermore, the proposed variant does not require any additional control variables. The proposed algorithm has been compared with various existing variants of ABC algorithm on a wide-set of the complex benchmark functions. The results show better performance of the proposed variant of ABC algorithm in comparison to the existing variants of the standard ABC algorithm.

Rest of the paper is divided into five sections. The immediate section presents the standard-ABC algorithm. The third section presents the proposed variants. The fourth section presents parameter settings of the compared optimization algorithms and the considered benchmark functions for the performance evaluation of the compared algorithms. Then the results have been reported and discussed in the fifth section. The conclusion has been presented in the sixth section.

## II. Artificial Bee Colony Optimization Algorithm

Honeybees in ABC algorithms are divided into three classes [13, 24]. One is called employed-bees, which are actually assigned food-sources at the start of the algorithm. The food-sources represent the possible-solutions. The second type of honeybees is called onlooker-bees. Onlooker-bees wait in a dancing area of the hive for seeking information from the employed-bees about the explored food-sources. The onlooker-bees probably-select the food-sources having higher nectar-amount [15]. The nectar-amount of a food-source symbolizes the quality or fitness of a possible-solution. The third type of bees is known as scout-bees. If any possible-solution does not show any improvement over a preset number of times, the possible-solution is abandoned. The preset number of times is controlled by a user-defined control variable named *limit* [26]. The employed-bee associated with the abandoned food-source becomes scout-bee. Then the scout-bee flies around the hive for picking a new food-source [16].

As the standard-ABC algorithm starts with random initialization of the food-sources around the hive. Each employed-bee is assigned a food-source. Hence, the number of employed-bees and the food-sources is equal. Moreover, the number of employed-bees and onlooker-bees is same. Employed-bees explore the neighborhood of the associated food-sources using the following mutation equation,

$$z_{ij} = y_{ij} + \phi_{ij}(y_{ij} - y_{kj}) \tag{1}$$

where $y_{ij}$ is the $j_{th}$ dimension of the $i_{th}$ food-source, $y_{kj}$ represents the $j_{th}$ dimension of the $k_{th}$ food-source, $z_{ij}$ symbolizes candidate-food-source of the $j_{th}$ dimension of the $i_{th}$ food-source, *i* and *k* are the mutually-exclusive food-sources, j Є [1,2,…. D], *D* is the dimension of search space, *j* and *k* are randomly chosen numbers and $\varphi$ is a random number within [-1, 1].

Equation (1) clearly shows that the neighborhood of every food-source is explored by a randomly chosen food-source. The fitness of food-sources is calculated by the following equation,

$$fit_i = \begin{cases} \dfrac{1}{1+f_i},\ f_i \geq 0, \\ 1+abs(f_i),\ f_i < 0, \end{cases} \tag{2}$$

where $f_i$ is objective function value of $i_{th}$ food-source and $fit_i$ is the corresponding fitness value after transformation.

The fitness value of the candidate and the old food-source is compared. The food-source having higher fitness value is retained [13]. This is known as greedy selection. As mentioned earlier, onlooker-bees wait in the dancing area of the hive for receiving the information of food-sources found by employed-bees. In the standard-ABC, the probability of the food-source having higher fitness value is calculated by the following equation,

$$p_i = \frac{fit_i}{\sum_{j=i}^{NS} fit_j} \tag{3}$$

where *NS* represents maximum number of food-sources and *i* is the selected food-source.

If an abandoned food-source exists then the scout-bee is assigned a randomly-initialized food-source to replace the abandoned food-source. Only one scout-bee has been used in the standard-ABC and its variants. The scout-bee is assigned a food-source using the equation given below,

$$y_{ij} = y_j^{\min} + rand(0,1)(y_j^{\max} - y_j^{\min}) \tag{4}$$

where $y_j^{\min}$ is lower limit of search space, $y_j^{\max}$ is upper bound of search space and *rand* is randomly generated number within [0 1]. The pseudo-code of the standard ABC algorithm is given below,

```
START
Random initialization of food-sources
CYCLE = 1
REPEAT while preset number of generation is reached
    Employed bees explore food-source using equation (1)
        Calculate fitness of explored food-source using equation (2)
        Apply greedy-selection
    Onlooker bees explore selected food-sources using equation (1)
        Calculate fitness of explored food-source using equation (2)
        Apply greedy-selection
    Look for abandoned food-source
            If abandoned food source exist
            Then assign a food-source to scout bee using equation (4)
    Memorize the best food-source found so far
INCREASE cycle counter
TERMINATE if cycle is equal to maximum number of cycles
END
```

## III. Proposed Algorithm

This research work proposes two different modifications for the performance enhancement of the standard ABC optimization algorithm. The proposed algorithm is named Enhanced ABC algorithm (EABC). The modifications are described below sections and the flow chart of the proposed algorithm is portrayed in Figure 1.

### A. Enhancement of ABC neighborhood exploration

In the standard ABC algorithm, neighborhood-exploration of any food-source using the global-best food-source increases the convergence rate. However, it inducts the tendency towards premature convergence. On the other hand, the neighborhood-exploration of any food-source on the basis of a randomly chosen food-source curtails convergence rate. Nevertheless, it inducts capability to avert the local optima. Therefore, this research work proposes a novel variant of the standard-ABC algorithm, which capitalizes on different mutation-equations for availing the benefits of each mutation-equation. The mutations equations are given as;

$$z_{ij} = y_{ij} + \phi_{ij}(y_{ij} - y_{best,j}) \tag{5}$$

$$z_{ij} = y_{ij} + \phi_{ij}(y_{ij} - y_{second-best,j}) \tag{6}$$

$$z_{ij} = y_{ij} + \phi_{ij}(y_{ij} - y_{kj}) \tag{7}$$

where $z_{ij}$ corresponds to candidate-solution of the $j_{th}$ dimension of the $i_{th}$ food-source, $y_{best,\ j}$ is the jth index of the global-best food-source, $y_{second\text{-}best,\ j}$ is the jth index of the global-second-best food-source, $y_{ij}$ symbolizes the $j_{th}$ dimension of the $i_{th}$ food-source, $y_{kj}$ represents the $j_{th}$ dimension of the $k_{th}$ food-source, *i* and *k* are the mutually-exclusive food sources, j Є [1,2,…. D], *D* is the dimension of search space and $\varphi$ is a random number within [-1, 1].

Equation (5) and (6) capitalize on the global-best and the global-second-best food-source. Hence, the proposed algorithm capitalizes on the multiple global-best food-source rather than the single global-best food-source. Thus the proposed algorithm explores around the multiple so-far best-found areas of the search-space and hence, has ability to converge faster and simultaneously avoid local-optima trappings. Moreover, equation (7) capitalizes on a randomly chosen food-source and it further strengthen local-optima avoiding capability of the proposed algorithm.

The proposed algorithm explores the neighborhood of every food-source on the basis of Equations (5), (6) and (7). After calculating the nectar amount of three new food-sources, the best among three is selected for applying the greedy selection. Thus the proposed algorithm uses three different mutation-equations however, it updates every food-source only once in an iteration.

### *B. Enhancement of scout-bee searching capability*

In the standard-ABC and in its various variants, the scout-bee is assigned a randomly initialized food-source using equation (4). Hence, there are little chances of getting a quality food-source. Therefore, this research work proposes a novel scheme to assign a quality food-source to the scout-bee. The scheme capitalizes on the global-best food-source to assign a food-source to the scout-bee. Following is the proposed equation to assign a food-source to scout-bee.

$$z_{ij} = \beta_{nj}(y_{best,j}) \tag{8}$$

where $z_{nj}$ is the $j_{th}$ dimension-magnitude of the newly assigned food-source, $y_{best,\ j}$ is the $j_{th}$ dimension-magnitude of the global-best food-source and $\beta_{nj}$ is random number within [0.90 1.10].

The limits of β are selected so that the dimension-magnitudes of the new food-source may not go out of the chosen boundaries of the function.

## IV. Experimental Setup and Parameter Setting

This research work compares performance of the proposed Enhanced-ABC (EABC) algorithm to gbest-guided-ABC (GABC) [27], Improved-ABC (IABC) [25], global-best ABC (BABC) [26], best-so-far ABC (BSFABC) [28], modified ABC (MABC) [24] and the standard ABC (ABC) on a wide set of benchmark functions. The benchmark functions considered for the comparative analysis carried out in this research work, have also been used in [24, 26, 30, 32]. Functions from $f_1$ to $f_{16}$ are commonly used benchmark functions whereas, $f_{17}$ to $f_{24}$ are expended functions. Method to formulate the expended functions can be obtained in [32]. The benchmarks functions are given in Table 1.

Dimension of all the benchmark functions has been set to fifty. The *colony-size* of all the algorithms has been set to 50, *limit* (i.e. the control variable) has been set to 80 and the *number-of-generations* has been limited to 1000. Each algorithm has been run 30 times on each function to observe the variability of the algorithms. Random initialization has been used for all compared algorithms to make the comparison even. The performance analysis of the algorithms has been carried out on the basis of the best convergence, the worst convergence, average convergence over the thirty runs and standard deviation among the thirty outputs. The standard deviation predicts the robustness, the average value prophecies the convergence of the algorithms.

*C-value* for GABC has been set to 1.5, *ASF* for ModABC has been limited to 0.9 and *P-value* for IABC has been set to 0.25. A statistical test Wilcoxon signed rank has been performed to evaluate the significance of the proposed algorithm. Significance “1 (one)” represents the significantly better results of the proposed algorithm than the respective algorithm, at 95% or more confidence level. Moreover, significance “0 (zero)” corresponds to the insignificant difference whereas “-1” shows that the results of the proposed algorithm are significantly inferior to the respective algorithm. Significance of the results is considered when the statistical test yields *“h-value”* equal to 1 at 95% or more confidence level.

## V. Results and Discussion

Table-2 gives the performance-analysis results of the algorithms on $f_1$ to $f_{16}$. Results reveal worse performance of BSFABC and MABC in comparison to rest of the compared optimization algorithms. BABC and IABC have yielded better performance in comparison to GABC on all functions except $f_{13}$ and $f_{16}$. However, BABC and IABC are very much competitive to each other. Moreover, the performance analysis reveals poor performance of BABC on $f_3$ than IABC and GABC. Function $f_3$ is rosenbrock function and it has global-optimum lying in curving and deep valley. Moreover, BABC has yielded worse performance than IABC on $f_5$. Function $f_5$ represents schwefel function and it has very deep local-optimum far from the global-optimum. As BABC generates solutions only around the global-best food-source, therefore, the algorithm is unable to avoid local optima trappings.

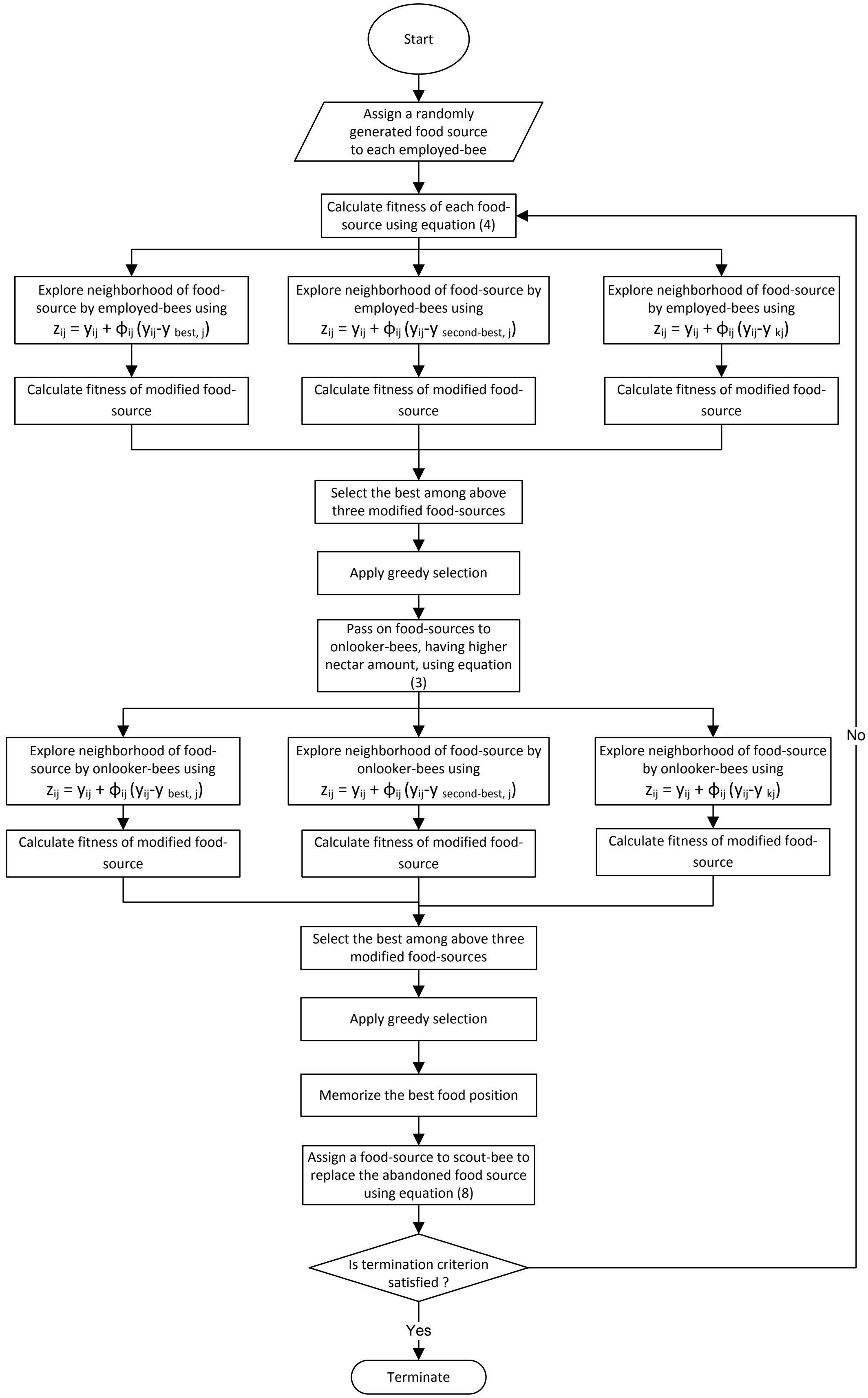


Figure 1 Flow chart of Enhanced ABC algorithm.

Table 1 Benchmark functions for the performance evaluations of the optimization algorithms

| Label | Function | Function Formula | Range | Modality |
|---|---|---|---|---|
| $f_1$ | RS Griewank | $f(x)=\frac{1}{4,000}(\sum_{i=1}^{D}(x-a)_i^2-\prod_{i=1}^{D}\cos(\frac{(x-a)_i}{\sqrt{i}})+1$ | ±600 | MN |
| $f_2$ | RS Rastringin | $f(x)=\sum_{i=1}^{D}(x-a)_i^2-10\cos(2\pi(x-a)_i))+10\times D$ | ±15 | MS |
| $f_3$ | RS Rosenbrock | $f(x)=\sum_{i=1}^{D}100((x-a)_i^2-(x-a)_{i+1})^2+(1-(x-a)_i)^2$ | ±15 | UN |
| $f_4$ | RS Ackley | $f(x)=20+e-20e^{(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}(x-a)_i^2}}-e^{\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi(x-a)_i)}$ | ±32 | MN |
| $f_5$ | RS Schwefel | $f(x)=D\times418.9829+\sum_{i=1}^{D}-(x-a)_i\times\sin(\sqrt{\lvert x-a\rvert_i})$ | ±500 | MN |
| $f_6$ | RS Himmelblau | $f(x)=\frac{1}{D}\sum_{i=1}^{D}((x-a)_i^4-16(x-a)_i^2+5(x-a)_i)$ | ±600 | MS |
| $f_7$ | RS Sphere | $f(x)=\sum_{i=1}^{D}(x-a)_i^2$ | ±100 | US |
| $f_8$ | RS Step | $f(x)=\sum_{i=1}^{D}\left(\lfloor(x-a)_i+0.5\rfloor\right)^2$ | ±100 | US |
| $f_9$ | RS Bohachevsky | $f(x)=\sum_{i=1}^{D-1}\begin{matrix}(x-a)_i^2+2(x-a)_{i+1}^2-\ldots\\ \ldots0.3\cos(3\pi(x-a)_i)\cos(3\pi(x-a)_{i+1})+0.3\end{matrix}$ | ±100 | MN |
| $f_{10}$ | RS Schwefel2 | $f(x)=\sum_{i=1}^{D}\lvert(x-a)_i\rvert+\prod_{i=1}^{D}\lvert(x-a)_i\rvert$ | ±100 | UN |
| $f_{11}$ | RS Schwefel Ridges | $f(x)=\sum_{i=1}^{D}\left(\sum_{j=1}^{i}(x-a)_j\right)^2$ | ±100 | UN |
| $f_{12}$ | RS Schwefel Ridges with Noise | $f(x)=\sum_{i=1}^{D}\left(\sum_{j=1}^{i}(x-a)_j\right)^2+(1+0.4\lvert N(0,1)\rvert)$ | ±15 | UN |
| $f_{13}$ | RS Elliptic | $f(x)=\sum_{i=1}^{D}\left(10^6\right)^{\frac{i-1}{D-1}}\left((x-a)_i^2-10\right)-450$ | ±100 | UN |
| $f_{14}$ | Zekhprice | $f(x)=\sum_{i=1}^{D-1}\begin{matrix}((x-a)_i-1)^2[1+\sin^2(3\pi(x-a)_{i+1})]+\ldots\\ \ldots\sin^2(3\pi(x-a)_1)+\lvert x_D-1\rvert[1+\sin^2(3\pi(x-a)_D)]\end{matrix}$ | ±15 | MN |
| $f_{15}$ | Non-continuous Rastringin | $f(x)=10D+\sum_{i=1}^{D}y_i^2-10\cos(2\pi y_i)$; $y_i=\begin{cases}x_i,\lvert x_i\rvert<1/2\\ round(2x_i)/2,\lvert x_i\rvert\geq1/2\end{cases}$ | ±15 | MS |
| $f_{16}$ | Dixon-Price | $f(x)=(x_1-1)^2+\sum_{i=2}^{D}i\left(2\left(x_i^2\right)-x_{i-1}\right)^2$ | ±100 | UN |
| $f_{17}$ | First expanded function | *Griewank's plus Rosenbrock's Function* | ±15 | MN |
| $f_{18}$ | Second expanded function | *Random-Shiftted-Sphere plus Schwefel* | ±500 | US |

| | | | | |
|---|---|---|---|---|
| $f_{19}$ | Third Expanded Function | *Rosenbrock plus Step* | ±15 | UN |
| $f_{20}$ | Fourth Expanded Function | *Rastringin plus Step* | ±500 | MS |
| $f_{21}$ | Fifth Expanded Function | *Bohachevsky 2 plus Step* | ±100 | MN |
| $f_{22}$ | Sixth Expanded Function | *Bohachevsky plus Schwefel Ridge* | ±100 | MN |
| $f_{23}$ | Seventh Expanded Function | *NonContinuous Rastringin plus Schwefel Ridge* | ±15 | MS |
| $f_{24}$ | Eighth Expanded Function | *Dixon-Price plus Schwefel Ridge* | ±100 | UN |
| RS = Random shifted, M = Multi-Modal, U =Uni-Modal, S = Separable, N = Non-separable | | | | |

On the other hand, the proposed algorithm has yielded significantly best performance among all the compared optimization algorithms on all benchmark functions except $f_6$ and $f_{12}$. The reported results on $f_6$ and $f_{12}$ reveal that all the optimization algorithms have converged to the optimal or very near to the optimal solution. Nevertheless, Figures 2 and 3 clearly depict the best convergence of the proposed algorithm on the functions.

Table 2 Performance results of the compared algorithms on $f_1$ to $f_{16}$.

| Label | Algorithm | Average | St Deviation | Sig |
|---|---|---|---|---|
| $f_1$ | **EABC** | **1.40E-16** | **2.40E-16** | |
| | BABC | 1.13E-12 | 4.43E-13 | 1 |
| | IABC | 1.86E-12 | 8.78E-13 | 1 |
| | GABC | 6.85E-11 | 4.19E-11 | 1 |
| | BSFABC | 3.93E-04 | 6.47E-04 | 1 |
| | MABC | 6.60E-02 | 4.81E-02 | 1 |
| | ABC | 1.76E-07 | 1.98E-07 | 1 |
| $f_2$ | **EABC** | **1.02E-11** | **3.16E-10** | |
| | BABC | 8.06E-06 | 1.05E-05 | 1 |
| | IABC | 1.60E-06 | 2.77E-06 | 1 |
| | GABC | 1.67E+00 | 9.68E-01 | 1 |
| | BSFABC | 3.63E+01 | 4.98E+00 | 1 |
| | MABC | 2.94E+01 | 9.12E+00 | 1 |
| | ABC | 6.08E+00 | 2.14E+00 | 1 |
| $f_3$ | **EABC** | **1.37E-04** | **1.32E-01** | |
| | BABC | 9.80E+01 | 4.27E+01 | 1 |
| | IABC | 3.78E+01 | 3.26E+01 | 1 |
| | GABC | 7.74E+01 | 3.22E+01 | 1 |
| | BSFABC | 3.00E+02 | 4.34E+01 | 1 |
| | MABC | 1.53E+02 | 2.77E+01 | 1 |
| | ABC | 4.51E+01 | 2.88E+01 | 1 |
| $f_4$ | **EABC** | **1.56E-10** | **1.20E-09** | |
| | BABC | 7.16E-06 | 1.62E-06 | 1 |
| | IABC | 1.19E-05 | 3.48E-06 | 1 |
| | GABC | 1.16E-04 | 4.26E-05 | 1 |
| | BSFABC | 3.72E+00 | 6.13E-01 | 1 |
| | MABC | 2.92E+00 | 3.80E+00 | 1 |
| | ABC | 2.75E-02 | 2.10E-02 | 1 |
| $f_5$ | **EABC** | **3.46E+01** | **6.70E+01** | |
| | BABC | 1.30E+02 | 1.00E+02 | 0 |
| | IABC | 3.55E+01 | 5.52E+01 | 0 |
| | GABC | 1.18E+03 | 2.51E+02 | 1 |
| | BSFABC | 3.41E+03 | 2.82E+02 | 1 |
| | MABC | 8.60E+03 | 4.44E+02 | 1 |
| | ABC | 1.39E+03 | 2.54E+02 | 1 |
| $f_6$ | **EABC** | **-7.83E+01** | **2.67E-14** | |
| | BABC | -7.83E+01 | 3.75E-09 | 0 |
| | IABC | -7.83E+01 | 2.10E-08 | 0 |
| | GABC | -7.83E+01 | 1.05E-04 | 0 |
| | BSFABC | 1.24E+03 | 1.31E+03 | 1 |
| | MABC | 9.17E+00 | 1.18E+02 | 1 |
| | ABC | -7.82E+01 | 2.19E-01 | 0 |
| $f_7$ | **EABC** | **1.40E-15** | **1.86E-16** | |
| | BABC | 3.87E-09 | 1.68E-09 | 1 |
| | IABC | 9.83E-09 | 6.27E-09 | 1 |
| | GABC | 2.53E-08 | 2.78E-08 | 1 |
| | BSFABC | 3.88E+02 | 2.34E+02 | 1 |
| | MABC | 2.51E+02 | 1.42E+02 | 1 |
| | ABC | 3.71E-05 | 7.94E-05 | 1 |
| $f_8$ | **EABC** | **1.27E-15** | **1.69E-16** | |
| | BABC | 3.64E-09 | 1.77E-09 | 1 |
| | IABC | 1.29E-08 | 9.08E-09 | 1 |
| | GABC | 3.17E-08 | 3.96E-08 | 1 |
| | BSFABC | 4.82E+02 | 2.18E+02 | 1 |

| | | | | |
|---|---|---|---|---|
| | MABC | 2.83E+02 | 1.77E+02 | 1 |
| | ABC | 5.98E-05 | 1.07E-04 | 1 |
| $f_9$ | **EABC** | **1.53E-15** | **1.64E-15** | |
| | BABC | 2.39E-06 | 6.53E-06 | 1 |
| | IABC | 5.77E-07 | 4.52E-07 | 1 |
| | GABC | 1.00E-06 | 1.11E-06 | 1 |
| | BSFABC | 4.51E+01 | 1.52E+01 | 1 |
| | MABC | 2.84E+01 | 1.07E+01 | 1 |
| | ABC | 4.26E-02 | 1.54E-01 | 1 |
| $f_{10}$ | **EABC** | **4.25E-10** | **2.51E-10** | |
| | BABC | 2.68E-05 | 5.30E-06 | 1 |
| | IABC | 4.87E-05 | 1.12E-05 | 1 |
| | GABC | 8.62E-05 | 2.55E-05 | 1 |
| | BSFABC | 8.18E+00 | 2.34E+00 | 1 |
| | MABC | 8.70E+00 | 1.70E+00 | 1 |
| | ABC | 5.30E-03 | 2.02E-03 | 1 |
| $f_{11}$ | **EABC** | **1.41E-15** | **2.55E-16** | |
| | BABC | 2.54E-09 | 1.20E-09 | 1 |
| | IABC | 8.09E-09 | 5.00E-09 | 1 |
| | GABC | 1.61E-08 | 1.25E-08 | 1 |
| | BSFABC | 4.50E+02 | 2.56E+02 | 1 |
| | MABC | 1.97E+02 | 1.11E+02 | 1 |
| | ABC | 5.67E-05 | 1.04E-04 | 1 |
| $f_{12}$ | **EABC** | **1.39E+00** | **6.10E-02** | |
| | BABC | 1.67E+00 | 9.04E-02 | 0 |
| | IABC | 1.61E+00 | 1.16E-01 | 0 |
| | GABC | 1.65E+00 | 8.96E-02 | 0 |
| | BSFABC | 2.57E+02 | 1.74E+02 | 1 |
| | MABC | 2.45E+02 | 1.52E+02 | 1 |
| | ABC | 1.98E+00 | 2.12E-01 | 0 |
| $f_{13}$ | **EABC** | **2.92E-15** | **2.96E-15** | |
| | BABC | 3.07E-05 | 2.56E-05 | 1 |
| | IABC | 1.72E-04 | 1.90E-04 | 1 |
| | GABC | 2.20E-05 | 2.03E-05 | 1 |
| | BSFABC | 3.22E+01 | 2.51E+02 | 1 |
| | MABC | 1.22E+01 | 1.51E+02 | 1 |
| | ABC | 4.59E-01 | 6.68E-01 | 1 |
| $f_{14}$ | **EABC** | **3.01E-12** | **7.73E-12** | |
| | BABC | 8.22E-04 | 2.99E-03 | 1 |
| | IABC | 3.71E-03 | 2.03E-02 | 1 |
| | GABC | 3.83E-03 | 2.03E-02 | 1 |
| | BSFABC | 5.19E-01 | 4.59E-01 | 1 |
| | MABC | 9.09E-02 | 4.21E-01 | 1 |
| | ABC | 2.02E-02 | 3.53E-02 | 1 |
| $f_{15}$ | **EABC** | **3.68E-09** | **1.27E-08** | |
| | BABC | 5.54E-04 | 7.45E-04 | 1 |
| | IABC | 1.96E-04 | 8.05E-04 | 1 |

| | | | | |
|---|---|---|---|---|
| | GABC | 3.19E+00 | 1.34E+00 | 1 |
| | BSFABC | 3.12E+01 | 3.55E+00 | 1 |
| | MABC | 6.39E+02 | 8.12E+01 | 1 |
| | ABC | 9.69E+00 | 1.90E+00 | 1 |
| $f_{16}$ | **EABC** | **1.17E-04** | **1.19E-02** | |
| | BABC | 7.31E+00 | 2.99E+00 | 1 |
| | IABC | 4.73E+00 | 2.74E+00 | 1 |
| | GABC | 3.96E+00 | 3.19E+00 | 1 |
| | BSFABC | 4.16E+04 | 4.62E+04 | 1 |
| | MABC | 8.59E+03 | 5.28E+03 | 1 |
| | ABC | 8.81E+00 | 4.44E+00 | 1 |

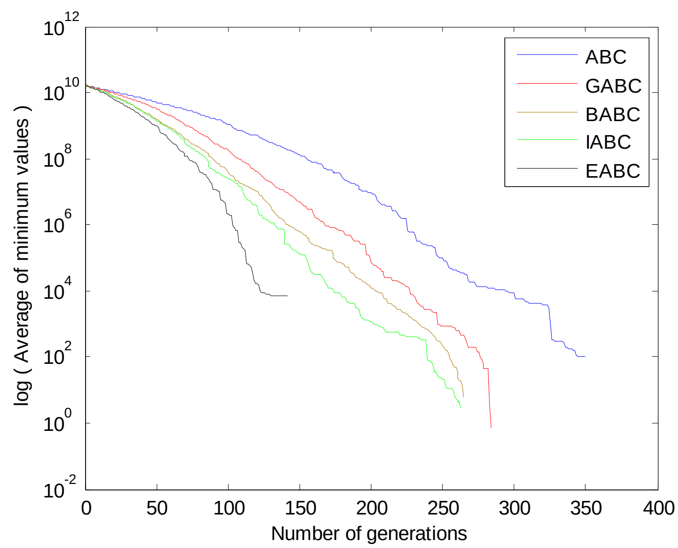


Figure 2 Convergence rates of the compared algorithms on $f_6$.

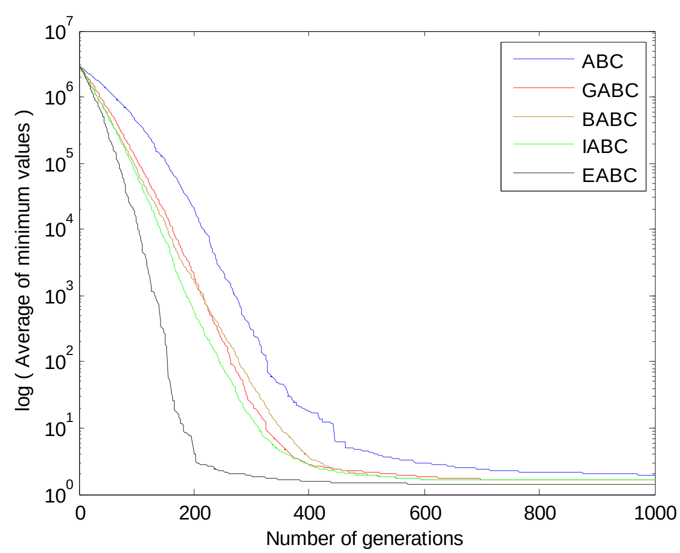


Figure 3 Convergence rates of the compared algorithms on $f_{12}$.

Table 3 gives the comparative analysis results of the optimization algorithms on $f_{17}$ to $f_{24}$. The results vividly prove the dominance of the proposed algorithm on all benchmark functions. Nonetheless, Table 3 results show that there is no significant difference among the results produced by the proposed algorithm and the compared algorithms on $f_{22}$. However, Figure 4 illustrates the best convergence of the proposed algorithm among all the algorithms on $f_{22}$ function.

Table 3 Performance results of the compared on expanded functions ($f_{17}$ to $f_{24}$).

| Label | Algorithms | Average | St Deviation | Sig |
|---|---|---|---|---|
| $f_{17}$ | **EABC** | **1.33E-04** | **2.36E-04** | |
| | BABC | 2.74E-01 | 1.02E-01 | 1 |
| | IABC | 1.55E-01 | 1.02E-01 | 1 |
| | GABC | 2.17E-01 | 1.01E-01 | 1 |
| | BSFABC | 8.52E-01 | 6.57E-01 | 1 |
| | MABC | 7.92E-01 | 1.80E-01 | 1 |
| | ABC | 2.02E-01 | 9.06E-02 | 1 |
| $f_{18}$ | **EABC** | **3.62E-01** | **6.03E-01** | |
| | BABC | 2.01E+01 | 7.61E+00 | 1 |
| | IABC | 6.52E-01 | 9.20E-01 | 0 |
| | GABC | 1.68E+01 | 1.30E+01 | 1 |
| | BSFABC | 3.71E+02 | 2.50E+02 | 1 |
| | MABC | 1.04E+02 | 4.05E+01 | 1 |
| | ABC | 2.84E+01 | 1.67E+01 | 1 |
| $f_{19}$ | **EABC** | **3.27E-04** | **3.24E+00** | |
| | BABC | 1.65E-01 | 3.84E+01 | 1 |
| | IABC | 1.23E-01 | 3.99E+01 | 1 |
| | GABC | 4.25E-03 | 3.66E+01 | 0 |
| | BSFABC | 8.01E+01 | 2.01E+02 | 1 |
| | MABC | 1.96E+01 | 2.67E+01 | 1 |
| | ABC | 3.15E+00 | 2.02E+01 | 1 |
| $f_{20}$ | **EABC** | **0** | **0** | |
| | BABC | 7.70E-16 | 1.66E-15 | 1 |
| | IABC | 9.89E-15 | 1.21E-14 | 1 |
| | GABC | 2.78E-14 | 5.66E-14 | 1 |
| | BSFABC | 2.39E+04 | 2.90E+04 | 1 |
| | MABC | 6.86E+03 | 6.97E+03 | 1 |
| | ABC | 2.17E-06 | 1.02E-05 | 1 |
| $f_{21}$ | **EABC** | **4.29E-19** | **2.84E-16** | |
| | BABC | 1.21E-15 | 4.02E-15 | 1 |
| | IABC | 4.02E-16 | 1.67E-16 | 1 |
| | GABC | 6.57E-16 | 2.24E-16 | 1 |
| | BSFABC | 1.44E+02 | 1.66E+02 | 1 |
| | MABC | 7.57E+01 | 1.25E+02 | 1 |
| | ABC | 1.44E-08 | 6.08E-08 | 1 |
| $f_{22}$ | **EABC** | **4.29E-16** | **2.84E-16** | |
| | BABC | 1.21E-15 | 4.02E-15 | 0 |
| | IABC | 4.02E-16 | 1.67E-16 | 0 |
| | GABC | 6.57E-16 | 2.24E-16 | 0 |
| | BSFABC | 1.44E+02 | 1.66E+02 | 1 |
| | MABC | 7.57E+01 | 1.25E+02 | 1 |
| | ABC | 1.44E-08 | 6.08E-08 | 1 |
| $f_{23}$ | **EABC** | **4.02E-18** | **1.92E-16** | |
| | BABC | 1.29E-13 | 2.32E-13 | 1 |
| | IABC | 1.65E-12 | 2.74E-12 | 1 |
| | GABC | 6.93E-13 | 1.11E-12 | 1 |
| | BSFABC | 4.21E+07 | 4.76E+07 | 1 |
| | MABC | 1.58E+07 | 2.09E+07 | 1 |
| | ABC | 3.75E-04 | 1.04E-03 | 1 |
| $f_{24}$ | **EABC** | **0** | **0** | |
| | BABC | 2.96E-16 | 8.19E-16 | 1 |
| | IABC | 1.00E-14 | 1.82E-14 | 1 |
| | GABC | 4.86E-15 | 1.36E-14 | 1 |
| | BSFABC | 1.71E+02 | 1.15E+02 | 1 |
| | MABC | 5.16E+02 | 7.82E+02 | 1 |
| | ABC | 4.05E-06 | 1.66E-05 | 1 |

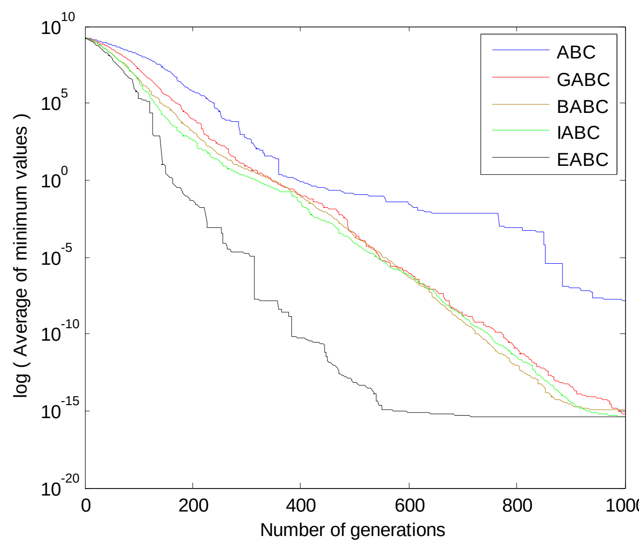


Figure 4 Convergence rates of the compared algorithms on $f_{22}$.

## I. Conclusion

This research work proposes two different modifications for the performance enhancement of the standard ABC optimization algorithm to avert the flaws of the algorithm. The first modification improves neighborhood searching capability of the standard-ABC algorithm. Additionally, the second modification enhances scout-bee stage of the standard-ABC algorithm. The proposed variant has been named Enhanced-ABC (EABC) algorithm. The proposed variant has been compared with various existing variants of ABC algorithm on a wide-set of high-dimensional complex benchmark functions.

The significance of the results has been evaluated using a statistical test. The results prove that the proposed variant has performed significantly better in comparison to the considered optimization algorithms on all the benchmark functions, irrespective of the type of the benchmark functions.

## ACKNOWLEDGMENT

The authors acknowledge Institute of Postgraduate Studies, Universiti Sains Malaysia for the financial support.

## REFERENCES

[1] Kang, F., J. Li, and Z. Ma, *Rosenbrock artificial bee colony algorithm for accurate global optimization of numerical functions.* Information Sciences, Vol. 181, No.16, pp. 3508-3531, 2011.

[2] Yingxu, W., W. Kinsner, and Z. Du, *Contemporary Cybernetics and Its Facets of Cognitive Informatics and Computational Intelligence.* Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, Vol.39, No.4, pp. 823-833, 2009.

[3] Kulkarni, R.V., Forster, A. and Venayagamoorthy, G. K., *Computational Intelligence in Wireless Sensor Networks: A Survey.* Communications Surveys & Tutorials, IEEE, Vol.13, No.1, pp. 68-96, 2011.

[4] Osareh, A., B. Shadgar, and R. Markham, *A Computational-Intelligence-Based Approach for Detection of Exudates in Diabetic Retinopathy Images.* Information Technology in Biomedicine, IEEE Transactions on, Vol.13, No.4, pp. 535-545, 2009.

[5] Werbos, P.J., *Computational Intelligence for the Smart Grid-History, Challenges, and Opportunities.* Computational Intelligence Magazine, IEEE, Vol.6, No.3, pp. 14-21, 2011.

[6] Wu, S.X. and W. Banzhaf, *The use of computational intelligence in intrusion detection systems: A review.* Applied Soft Computing, Vol.10, No.1, pp. 1-35, 2010.

[7] Brabazon, A., M. O'Neill, and I. Dempsey, *An Introduction to Evolutionary Computation in Finance.* Computational Intelligence Magazine, IEEE, Vol.3, No.4, pp. 42-55, 2008.

[8] Castro, L.N.d., *Fundamentals of Natural Computing; Basic Concepts, Algorithms and Applications*. Computer and Information Science Series, New York: Taylor & Francis, 2006

[9] Kumar, M., A. Sharma, and R. Kumar, *Optimization of Test Cases using Soft Computing Techniques: A Critical Review.* WSEAS Transactions on information science and applications, Vol.8, No.11, pp. 440-452, 2011.

[10] Priya, R.D. and S. Kuppuswami, *A Genetic Algorithm Based Approach for Imputing Missing Discrete Attribute values in Databases.* WSEAS Transactions on information science and applications, Vol.9, No.6, pp. 169-178, 2012.

[11] Surekha, P. and Dr.S.Sumathi, *Solving Economic Load Dispatch problems using Differential Evolution with Opposition Based Learning.* WSEAS Transactions on information science and applications, Vol. 9, No.1, pp. 1-13, 2012.

[12] Zhang, C. and J. Yue, *Application of an improved adaptive chaos prediction model in aero-engine performance parameters.* WSEAS Transactions on mathematics, Vol.11, No.2, pp. 114-124, 2012.

[13] Dervis Karaboga, B.B., *A powerful and efficient algorithm for numerical function optimization: artificial bee colony algorithm.* Journal of Global Optimization, Vol. 39, pp. 459-471, 2007.

[14] Mohammed, E.-A., *Performance assessment of foraging algorithms vs. evolutionary algorithms.* Information Sciences, Vol.182, No.1, pp. 243-263, 2012.

[15] Dervis Karaboga, B.A., *On the performance of artificial bee colony algorithm.* Applied Soft Computing, Vol.8, No.1, pp. 687-697, 2008.

[16] Karaboga, D. and B. Akay, *A comparative study of Artificial Bee Colony algorithm.* Applied Mathematics and Computation, Vol.214, No.1, pp. 108-132, 2009.

[17] Castro, L.N.d., *Fundamentals of natural computing: an overview.* Physics of Life Reviews, Vol.4, No.1, pp. 1-36, 2007

[18] Karaboga, D. and C. Ozturk, *A novel clustering approach: Artificial Bee Colony (ABC) algorithm.* Applied Soft Computing, Vol.11, No.1, pp. 652-657, 2011.

[19] Nurhan, K., *A new design method based on artificial bee colony algorithm for digital IIR filters.* Journal of the Franklin Institute, Vol.346, No.4, pp. 328-348, 2009.

[20] Sabat, S.L., S.K. Udgata, and A. Abraham, *Artificial bee colony algorithm for small signal model parameter extraction of MESFET.* Engineering Applications of Artificial Intelligence, Vol.23, No.5, pp. 689-694, 2010.

[21] Samanta, S. and S. Chakraborty, *Parametric optimization of some non-traditional machining processes using artificial bee colony algorithm.* Engineering Applications of Artificial Intelligence, Vol.24, No.6, pp. 946-957, 2011.

[22] Şencan Şahin, A., B. Kılıç, and U. Kılıç, *Design and economic optimization of shell and tube heat exchangers using Artificial Bee Colony (ABC) algorithm.* Energy Conversion and Management, Vol.52, No.11, pp. 3356-3362, 2011.

[23] Gao, W.F., S.Y. Liu, and F. Jiang, *An improved artificial bee colony algorithm for directing orbits of chaotic systems.* Applied Mathematics and Computation, Vol.218, No.7, pp. 3868-3879, 2011.

[24] Akay, B. and D. Karaboga, *A modified Artificial Bee Colony algorithm for real-parameter optimization*. Information Sciences, Vol.192, No.6, pp. 120-142, 2012.

[25] Gao, W. and S. Liu, *Improved artificial bee colony algorithm for global optimization.* Information Processing Letters, Vol.111, No.17, pp. 871-882, 2011.

[26] Gao, W., S. Liu, and L. Huang, *A global best artificial bee colony algorithm for global optimization.* Journal of Computational and Applied Mathematics, 236(11): pp. 2741-2753, 2012.

[27] Zhu, G. and S. Kwong, *Gbest-guided artificial bee colony algorithm for numerical function optimization.* Applied Mathematics and Computation, Vol.217, No.7, pp. 3166-3173, 2010.

[28] Banharnsakun, A., T. Achalakul, and B. Sirinaovakul, *The best-so-far selection in Artificial Bee Colony algorithm.* Applied Soft Computing, Vol.11, No.2, pp. 2888-2901, 2011.

[29] Alatas, B., *Chaotic bee colony algorithm for global numerical optimization.* Expert systems with applications, Vol.37, No.8, pp. 5682-5687, 2010.

[30] Gao, W.-f. and S.y. Liu, *A modified artificial bee colony algorithm.* Computers & Operations Research, Vol.39, No.3, pp. 687-697, 2012.

[31] Li, G., P. Niu, and X. Xiao, *Development and investigation of efficient artificial bee colony algorithm for numerical function optimization.* Applied Soft Computing, Vol.12, No.1, pp. 320-332, 2012.

[32] P.N. Suganthan, N.H., J.J. Liang, K. Deb, Y.-P. Chen, A. Auger, and S. Tiwari., *Problem definitions and evaluation criteria for the CEC 2005, special session on real-parameter optimization.* 2005.

**Abdul Ghani Abro** has received his first and second degree from MUET University – Pakistan and NED University – Pakistan. He is Assistant Professor at NED University – Pakistan. Currently he is pursuing PhD studies at School of Electrical and Electronic Engineering Universiti Sains Malaysia. His research interests include bio-inspired optimization algorithms and their application for power system operation and control.

**Junita Mohamad-Saleh** has received her B.Sc (in Computer Engineering) degree from the Case Western Reserve University, USA in 1994, the M.Sc. degree from the University of Sheffield, UK in 1996 and the Ph.D. degree from the University of Leeds, UK in 2002. She is currently an Associate Professor in the School of Electrical & Electronic Engineering, Universiti Sains Malaysia. Her research interests include computational intelligence and tomographic imaging.