## Phased Bee Colony Optimization Algorithm for Solving Mathematical Function

Rahul Kumar Mishra Enterprise Solutions Supply Chain Tata Consultancy Services New Delhi, India

#### ABSTRACT

In this article, we have proposed an efficient Bee Colony Optimization method, namely Phased Bee Colony Optimization (PBCO) technique for solving mathematical functions in a multidimensional space. The search process of the optimization approach is directed towards a region or a hypercube in a multidimensional space to find a global optimum or near global optimum after a predefined number of iterations. The process in the entire search area to another region (new search area) surrounding the optimum value found so far after a few iterations and restarts the search process in the new region. However, the search area of the new region is reduced compared to previous search area. Thus, the search process finding advances and jumps to a new search space (with reduced area space) in several phases until the algorithm is terminated. The PBCO technique has tested on a set of mathematical benchmark functions with number of dimensions up to 100 and compared with several relevant optimizing approaches to evaluate the performance of the algorithm. It has observed that the proposed technique performs either better or similar to the performance of other optimization methods.

#### **Keywords**

Bee Colony Optimization; phased; optimization; hypercube; mathematical functions

#### **1. INTRODUCTION**

The Artificial Bee Colony (ABC) method is a metaheuristic population based evolutionary search algorithm for solving various optimization problems. The algorithm is inspired by the intelligent foraging behavior of honeybee swarms in nature. The ABC algorithm basically imitates the behavior of real bees searching for food. The ABC produces new solution according to the stochastic variance process. It is an iterative algorithm in which the magnitudes of the perturbation are important for finding new solution. The ABC has been applied for solving different problems like scheduling, timetabling, travelling salesman problem, data mining and clustering and many more. Apart from these, the researchers have always made an attempt to enhance the performance of the ABC optimization with or without hybridization for solving problems in discrete as well as continuous domains. Pham and Castellani [1] proposed a continuous exploitative neighborhood search process combined with random explorative search, for increasing the speed and accuracy of the optimization method. Zou, Chen and Zhang [2] had developed an improved ABC algorithm to solve multiobjective optimization problems. Akbari, Mohammadi and Ziarati [3] developed a novel bee swarm optimization algorithm to solve numerical function optimization. Aydin, Liao, Oca and Stutzle [4] modified the ABC by making the population growth over time to improve the performance of algorithm. Zhang [5] used this algorithm to solve the job shop

scheduling problem. Yildih [6] proposed an approach with ABC algorithm for optimal selection of cutting parameters in multipass turning operation. Karaboga and Ozturk [7] applied ABC technique in the application of clustering.We have proposed a new technique based on the ABC algorithm for solving mathematical optimization problems in both discrete and continuous domains. The name of proposed algorithm is Phased Bee Colony Optimization (PBCO). The search in PBCO advances in multiple phases depending on the quality of nectar in the flower patches in a particular region of search space where the best quality of nectar till found after the end of a few number of iterations in each phase. In PBCO the search always jumps to a new region (i.e. the neighboring hypercube in a multidimensional space) with same number of bees. However, the space of new search area is smaller than the previous (hypercubic) search area as shown in Fig. 1. The reduction of search space is done along all dimensions. We have compared our proposed method with some relevant optimization techniques testing on a set of standard benchmark functions with problem dimensions upto 100. The rest of the paper is organized as follows. Section II describes the algorithms of simple ABC and PBCO techniques. It also discusses the search space partitioning process with progress of search. The experimental results along with discussion are provided in Section III. Section IV concludes the paper.

# 2. FUNCTION OPTIMIZATION USING ABC

A global optimization problem can generally be formulated as a pair (S, f) where  $S \subseteq R^n$  is a bounded set on  $R^n$  and f: S  $\mapsto R$  is an n- dimensional real- valued function. The objective of the problem is to find a point opt  $\in S$  on  $R^n$  such that  $(x_t)$ is a global optimum on S. We have to find  $x_t \in S$  according to the following equations for minimization or maximization problems, respectively:

$\forall x \subseteq s:$	$(\mathbf{x}_t)$	f x
$\forall x \subseteq s:$	$(\mathbf{x}_{t})$	f x

where f may not be a continuous function but it must be bounded.

#### 2.1 Conventional ABC Algorithm

Artificial Bee Colony (ABC) algorithm has been proposed by Karaboga in 2005 for optimizing the solutions of different problems [8]. ABC algorithm introduces principle of optimization inspired from the foraging behavior of a bee colony. According to ABC algorithm, there are three categories of artificial bees, known as employed bees, onlooker bees and scout bees. Colony consists of equal numbers of employed bees and onlooker bees. Employed bees search the food in the food source bounded with search space and after that sharing of food information with onlooker bees has done. Onlooker bee's tries to select possibly the best food sources from those discovered by the employed bees, and then further search for foods around the selected food source. Bees whose food sources get abandoned got changed from employed bees to Scout bees, assigned with new random food source [8, 9, and 10]. The simple ABC is a global optimization method, which used for solving various optimization problems [1 - 7]. The ABC algorithm represented algorithmically as follows.

- 1. Initialize the population of solutions  $(X_{i,j})$  in search space and , here denotes lower limit and upper limit of search space and i,j here denotes lower limit and upper limit of search space.
- 2. Evaluate the population
- 3. Repeat
- 4. Produce new solutions  $(U_{i,j})$  in the neighborhood of  $X_{i,j}$  and evaluate. $U_{i,j} = X_{i,j} + O_{i,j} (X_{i,j} X_{k,j})$ , where k= solution in the neighborhood of I andO= random number in the range [-1,1].
- 5. Greedy selection process will be applied in the selection between  $X_{i,j} \mbox{ and } U_{i,j}$  .
- Calculate value ( [*fit*] \_*i*) of solutions followed by the calculation of probability (*p*<sub>-</sub>*i*) where

$$fit_{i} = \begin{cases} \frac{1}{1+f}, & f \ge 0\\ 1+abs(f), & f < 0 \end{cases}$$
$$p_{i} = \frac{fit_{i}}{\sum_{i}^{N} fit_{i}} \text{ where } N = Dimension$$

- 7. Greedy selection is again applied and the abandoned solution is determined, if exists and it is replaced by the new randomly produced solution.
- 8. Memorize the best food source found so far until requirements has met.
- 9. Stop

#### 2.2 Optimization using PBCO

We have modified the simple ABC technique into a phased ABC approach by directing the search towards the area where the possibility of best result is maximum and reducing the search space accordig to the user's choice. The repetitive reduction of the search area in a 2-D space is described in the following section. Initially, the PBCO algorithm starts searching to find the optimum value in the entire search space. The search process then runs for a certain number of iterations (say, Ik) and gives the best possible solutions after the completion of  $I_k$  iterations. The search process is directed towards the area where closed to optimum solution is reached after a few iterations in each phase. The PBCO iterates executes iteratively in multiple phases and in each phase, we find the area i n which the best optimum value lies. The search space is reduced to corresponding area along all dimensions. Once the optimization methods completes Ik the best solution and the position of that solution is also calculated in the search space. The space where the best solution is achieved is reduced along all dimensions (discussed in detail in Section II C). The bee population is regenerated in that reduced search space. The PBCO approach restarts in the new search area and continues again for  $I_k$  times before it is transferred to another new search space considering the best solution approaching

the optimum. The PBCO finally terminates after the completion of  $I_{max}$  iterations.

Algorithmic representation of the PABC technique

- 1. Initialize the population of bees, other parameters and iteration I=1.
- 2. Create solutions for all bees.
- 3. Find the area where the best solution lies of current phase.
- 4. Redefine the search space surrounding the area where the best solution achieved and regenerate the population of bees following the elitist model.
- 5. Move the search to the new search space (See Fig. 1) which is smaller in size than the previous search space and restart the search process.
- 6. Increment  $I = I + I_k$ . If  $I \le I_{max}$ , then go to step 2.
- 7. Stop.

# 2.3 Search space reduction and transferring of the PBCO method

In Fig. 1, we have shown the search space reduction after every (=5, say) iterations in three stages and the shifting of PBCO to the new space at each stage in 2-D space. The search continued for i.e. 5 times after generating the population of bees. The points in the small box of search space ABCD (Fig. 1(a)) show the assumed best values of first stage. Similarly, the best optimum value of 2<sup>nd</sup> and 3<sup>rd</sup> stage are also shown by boxing the points as shown in Fig. 1(b) and Fig. 1(c). The remaining three best values also depicted in FIG. 1 showing various phases. The best solutions in phase 1 (Fig. 1(a)) are boxed in EFGH. The area (search space) of acquiring this solution redefined in the following way. We reduce the search space by a specific percentage (say, u%) along all dimensions. In Fig, 1(b), the new search space has shown only. Along the coordinate axis, the length of AB and DC is reduced by u% such that it adjusted to the new search space named EFGH, as shown in Fig. 1(a). The PBCO method restarts the search process at stage 2 with a new population of bees following the elitist model. In the space EFGH, let us consider the search finds two best values (as boxed in IJKL) out of five iterations and each of the remaining three iterations lie at other positions as shown in Fig. 1(b). Similar to stage 1, the search area corresponding to the positions of best values of stage 2 redefined as previously described. In the final stage i.e. Stage 3, the newly defined search space that is smaller (compared to the area of EFGH) is denoted by IJKL. The search restarts in the space IJKL. Similar to the previous phases, we find the best solutions in the next five iterations. The best solution at the end stage 3 has shown by a small box (Fig. 1(c)). Thus, the point (boxed in Fig. 1(c)) becomes the global or nearglobal optimum solution of the problem at the end of final stage or at the end of the pre-specified number of iterations (which in this case is 15).



**Fig. 1(a)** 

Partitioning of space and best solution of first five iterations (1-5) at phase 1.



Fig. 1(b)

Partitioning newly defined search space and best solutions of next five iterations at phase 2.

Best solution of last five iterations and extraction of global or near-global optimum solution at final stage.



Fig. 1(c)

Fig. 1 Reduction of search space and transferring of the PBCO technique in 3 phases.

#### 3. EXPERIMENTAL STUDIES

The performance of the PBCO technique on a set of 20 mathematical functions evaluated in two parts. In the first part, we have compared the results of PBCO with several other relevant optimization algorithms to establish it as an efficient optimization algorithm. In the second part, we have optimized functions  $F_2 - F_{15}$  to show the effectiveness of the proposed method in optimizing problems with higher dimensions (up to 100). We have considered the set of 20 standard benchmark functions taken from [20], which has listed in Table 1. The results of the optimization methods except the PBCO in Table 3 have also taken from [20]. The functions  $F_1 - F_{20}$  are a combination of unimodal and multimodal functions. The functions  $F_{16} - F_{20}$  are low dimensional compared to other functions  $(F_1 - F_{15})$  (see Tables 1 and 2). Table 2 depicts the basic parameters of the tested functions  $(F_1 - F_{20})$ .

#### 3.1 Parameter Setup.

In the experiment, we have considered that the population size of bees is  $\mu$  (= 50) until the end of the proposed optimization algorithm. For each function in Table 1, the algorithm PBCO ran 50 times and the average of the optimum results of 50 runs tabulated in Table 3 for the PBCO method. The dimension of the function has denoted by n as described in Table 2 for each function. We have run the PBCO technique for I iterations to calculate the optimum value in each phase (hypercube) for directing the search in redefined search space for the next phase. The new search space is generated by reducing u% of the length along all dimensions of the previous search space (see Figs. 1(a) -1(c)). For all tested functions in Table 1, we have considered the value of I to be 10 and the value of u is set to 25. If the value of I is increased towards the higher side of the range, the search process slows down i.e. more number of iterations are required to converge (by the algorithm). On the other hand, if u is set to a high value (more than 25), the algorithm (PBCO) may not converge to a global or near - global optimum solution since the optimization method may have the possibility to miss the global or near - global optimum due to fast reduction of the search space after repetitive I iterations. In the experiment, we have tried to maintain the values of I and u within the range such that the PBCO approach converges faster to the global optimum for all the functions in Table 1.

#### Table 1. Functions tested for optimization

$F1 = \sum_{i=1}^{n} -xsin(\sqrt{ xi })$
$F2 = \sum_{i=1}^{n} x_i^2 - 10\cos(2\pi x_i) + 10$
$F3 = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^{n} \cos(2\pi x_i)\right) +$
$20 + \exp(1)$
$F4 = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$
$F5=\frac{\pi}{2}\{10sin^2(\pi y_1) +$
$\frac{\sum_{i=1}^{n-1} (y_i - 1)^2 \cdot [1 + 10sin^2(\pi y_{i+1})] + (y_n - 1)^2}{\sum_{i=1}^n u(x_i, 10, 100, 4)} $ where
$y_i = 1 + \frac{1}{4}(x_i + 1)$ and
$u(x_{i}, a, k, m) = \begin{cases} k(x_{i} - a)^{n}, x_{i} > a \\ 0, -a \le x_{i} \le a \\ k(-x_{i} - a)^{m}, x < -a \end{cases}$

 $F6 = \frac{1}{10} \{ sin^2 (3\pi x_1) + \sum_{i=1}^{n-1} (x_i - 1)^2 \times [1 + sin^2 (3\pi x_{i+1})] + (x_n - 1)^2 [1 + sin^2 (3\pi x_{i+1})] \}$  $sin^{2}(2\pi x_{n})$ ] +  $\sum_{i=1}^{n} u(x_{i}, 5, 100, 4)$  $\overline{\text{F7}=-\sum_{i=1}^n \sin(x_i)\sin^{20}\left(\frac{i\times x_i^2}{\pi}\right)}$  $F8=\sum_{i=1}^{n}\left[\sum_{j=1}^{n} (\chi_{i,j} sin \omega_j + \psi_{i,j} cos \omega_j) - \sum_{j=1}^{n} (\chi_{i,j} sin x_j$  $\psi_{i,i} cos x_i)]^2]$  $F9 = \frac{1}{n} \sum_{i=1}^{n} (x_i^4 - 16x_i^2 + 5x_i)$  $F10 = \sum_{i=1}^{n-1} [100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2]$ F11= $\sum_{i=1}^{n} x_i^2$  $F12 = \sum_{i=1}^{n} x_i^4 + random[0,1)$  $F13 = \sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i|$  $F14 = \sum_{i=1}^{n} (\sum_{j=1}^{i} x_j)^2$ F15=max{ $|x_i|, i=1, 2, ..., n$ } F16= $4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$ F17= $(x_2 - \frac{5.1}{4\pi}x_1^2 + \frac{5}{\pi}x_1 - 6)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos x_1 + 10$ F18=[1 +  $(x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 12x_1^2 + 12x_1^2)]$  $48x_2 - 36x_1x_2 + 27x_2^2)$ ] F19= $\sum_{i=1}^{11} [a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}]^2$  where  $a_1, a_2, \dots, a_{11} =$ [0.1957, 0.1947, 0.1735, 0.1600, 0.0844, 0.0627, 0.0456, 0.0342] ,0.0323,0.0235,0.0246]  $[b_1, b_2, \dots \dots \dots, b_{11}] =$  $[4,2,1,\frac{1}{2},\frac{1}{4},\frac{1}{6},\frac{1}{8},\frac{1}{10},\frac{1}{12},\frac{1}{14},\frac{1}{16}]$  $F20 = -\sum_{i=1}^{4} c_i \exp\left[-\sum_{i=1}^{6} a_{i,i} (x_i - p_{i,i})^2\right]$  where  $[c_1, c_2, c_3, c_4] = [1, 1.2, 3, 3.2]$  $\begin{bmatrix} a_{i,j} \end{bmatrix}_{4 \times 6} = [10\ 3\ 17\ 3.5\ 1.7\ 8], [0.05, 10, 17, 0.1, 8, 14], [3\ 3.5\ 1.7\ 10\ 17\ 8], [17\ 8\ 0.05\ 10\ 0.1\ 14]$  $[p_{i,j}]_{4 \times 6} = [0.1312 \ 0.1696 \ 0.5569 \ 0.0124 \ 0.8283]$ 0.5886],[0.2329 0.4135 0.8307 0.3736 0.1004

### 0.9991],[0.2348 0.1415 0.3522 0.2883 0.3047 0.6650],[0.4047 0.8828 0.8732 0.5743 0.1091 0.0381]

#### Table 2. Basic parameters of the tested functions

Func.	Search Space	Dimension(n)	Optimal value
F1	[-500,500] <sup>n</sup>	30	-12569.5
F2	[-5.12,5.12] <sup>n</sup>	30	0
F3	$[-32,32]^{n}$	30	0
F4	[-600,600] <sup>n</sup>	30	0
F5	[-50,50] <sup>n</sup>	30	0
F6	[-50,50] <sup>n</sup>	30	0

International Journal of Computer Applications (0975 – 8887) Volume 177 – No. 8, October 2019

F7	$[0,\pi]^n$	100	-99.2784
F8	$[-\pi,\pi]^n$	100	0
F9	[-5,5] <sup>n</sup>	100	-78.33236
F10	[-5,10] <sup>n</sup>	100	0
F11	[-100,100] <sup>n</sup>	30	0
F12	[-1.28,1.28] <sup>n</sup>	30	0
F13	[-10,10] <sup>n</sup>	30	0
F14	[-100,100] <sup>n</sup>	30	0
F15	$[-100, 100]^{n}$	30	0
F16	[-5,5] <sup>n</sup>	2	-1.0316285
F17	[-5,10],[0,15]	2	0.398
F18	[-2,2] <sup>n</sup>	2	3
F19	[-5,-5] <sup>n</sup>	4	0.0003075
F20	$[0,1]^n$	6	-3.32

### Table 3. Comparative results of PBCO with relevant optimization algorithms

Fun.	Algo.	M.F.E	G <sub>min</sub>	S.D
F1	ALEP	150,000	-11469.2	58.2
	FEP	900,000	-12554.5	52.6
	OGA/Q	302,116	-12569.453	6.447×10 <sup>-4</sup>
	M-L	655,895	-5461.826	275.15
	LEA	287,365	-12569.454	4.831×10 <sup>-4</sup>
	GACO	8300	-12569.48	1.953×10 <sup>-2</sup>
	PBCO	3900	-12569.48	1.126×10 <sup>-4</sup>
F2	ALEP	150,000	5.85	2.07
	FEP	500,000	4.6×10 <sup>-2</sup>	1.2×10 <sup>-2</sup>
	OGA/Q	224,710	0	0
	M-L	305,899	121.7575	7.7572
	LEA	223,803	2.103×10 <sup>-18</sup>	3.359×10 <sup>-18</sup>
	GACO	19,650	0	0
	PBCO	4,480	1.19080×10 <sup>-14</sup>	1.191×10 <sup>-14</sup>
F3	ALEP	150,000	1.9×10 <sup>-2</sup>	1.0×10 <sup>-3</sup>
	FEP	150,000	1.8×10 <sup>-2</sup>	2.1×10 <sup>-2</sup>
	OGA/Q	112,421	4.440×10 <sup>-16</sup>	3.989×10 <sup>-17</sup>
	M-L	121,435	2.5993	9.425×10 <sup>-2</sup>
	LEA	105,926	3.274×10 <sup>-16</sup>	3.001×10 <sup>-17</sup>
	GACO	30,550	3.641×10 <sup>-14</sup>	4.47×10 <sup>-14</sup>
	PBCO	6,325	6.32397×10 <sup>-12</sup>	3.236×10 <sup>-14</sup>
F4	ALEP	150,000	2.4×10 <sup>-2</sup>	2.8×10 <sup>-2</sup>
	FEP	200,000	1.6×10 <sup>-2</sup>	2.2×10 <sup>-2</sup>
	OGA/Q	134,000	0	0

International Journal of Computer Applications (0975 - 88	87)
<i>Volume 177 – No. 8, October 20</i>	)19

	M-L	151,281	1.1894×10 <sup>-1</sup>	1.040×10 <sup>-2</sup>
	LEA	130,498	6.104×10 <sup>-16</sup>	2.513×10 <sup>-17</sup>
	GACO	9,950	0	1.1841×10 <sup>-2</sup>
	PBCO	4,475	3.08999×10 <sup>-12</sup>	1.598×10 <sup>-14</sup>
F5	ALEP	150,000	6.0×10 <sup>-6</sup>	1.0×10 <sup>-6</sup>
	FEP	150,000	9.2×10 <sup>-6</sup>	3.6×10 <sup>-6</sup>
	OGA/Q	134,556	6.019×10 <sup>-6</sup>	1.159×10 <sup>-6</sup>
	M-L	146,209	2.105×10 <sup>-1</sup>	3.609×10 <sup>-2</sup>
	LEA	132,642	2.482×10 <sup>-6</sup>	2.276×10 <sup>-6</sup>
	GACO	16,900	1.571×10 <sup>-32</sup>	1.570×10 <sup>-32</sup>
	РВСО	6,175	2.65491×10 <sup>-16</sup>	2.655×10 <sup>-16</sup>
F6	ALEP	150,000	9.8×10 <sup>-5</sup>	1.2×10 <sup>-5</sup>
	FEP	150,000	1.6×10 <sup>-4</sup>	7.3×10 <sup>-5</sup>
	OGA/Q	134,143	1.869×10 <sup>-4</sup>	2.615×10 <sup>-5</sup>
	M-L	147,928	1.50534	2.25564
	LEA	130,213	1.734×10 <sup>-4</sup>	1.205×10 <sup>-4</sup>
	GACO	18,550	1.3497×10 <sup>-32</sup>	1.350×10 <sup>-32</sup>
	PBCO	8,670	8.2983×10 <sup>-15</sup>	8.298×10 <sup>-15</sup>
F7	OGA/Q	302,773	-92.83	2.626×10 <sup>-2</sup>
	M-L	329,087	-23.97544	6.2875×10 <sup>-1</sup>
	LEA	289,863	-93.01	2.314×10 <sup>-2</sup>
	GACO	9,250	-99.26889	6.279×10 <sup>-1</sup>
	PBCO	3,975	-99.26889	2.165×10 <sup>-2</sup>
F8	OGA/Q	190,031	4.672×10 <sup>-7</sup>	1.293×10 <sup>-7</sup>
	M-L	221,547	$2.58778 \times 10^4$	1.73975×10 <sup>3</sup>
	LEA	189,427	1.627×10 <sup>-6</sup>	6.527×10 <sup>-7</sup>
	GACO	3,350	1.7886×10 <sup>-7</sup>	2.38×10 <sup>-7</sup>
	PBCO	1,580	3.341×10 <sup>-10</sup>	1.42×10 <sup>-10</sup>
F9	OGA/Q	245,930	-78.300029	6.288×10 <sup>-3</sup>
	M-L	251,199	-35.80995	8.9146×10 <sup>-1</sup>
	LEA	243,895	-78.310	6.127×10 <sup>-3</sup>
	GACO	8,550	-78.332275	9.223×10 <sup>-5</sup>
	РВСО	3,750	-78.33233	3.245×10 <sup>-6</sup>
F10	OGA/Q	167,863	7.520×10 <sup>-1</sup>	1.140×10 <sup>-1</sup>
	M-L	137,100	2935.93	134.8186
	LEA	168,910	5.609×10 <sup>-1</sup>	1.078×10 <sup>-1</sup>
	GACO	15,450	0	1.776×10 <sup>-13</sup>
	PBCO	6,500	3.421×10 <sup>-13</sup>	3.421×10 <sup>-13</sup>
F11	ALEP	150,000	6.32×10 <sup>-4</sup>	7.6×10 <sup>-5</sup>
	FEP	150,000	5.7×10 <sup>-4</sup>	1.3×10 <sup>-4</sup>
	OGA/Q	112,559	0	0
	M-L	162,010	3.19123	2.9463×10 <sup>-1</sup>

	LEA	110,674	4.727×10 <sup>-16</sup>	6.218×10 <sup>-17</sup>
	GACO	49,650	0	0
	PBCO	18,840	6.703×10 <sup>-14</sup>	6.703×10 <sup>-14</sup>
F12	OGA/Q	112,982	6.301×10 <sup>-3</sup>	4.069×10 <sup>-4</sup>
	M-L	124,982	1.703986	5.2155×10 <sup>-1</sup>
	LEA	111,093	5.136×10 <sup>-3</sup>	4.432×10 <sup>-4</sup>
	GACO	6,350	6.518×10 <sup>-5</sup>	7.6622×10 <sup>-4</sup>
	PBCO	2,550	7.660×10 <sup>-12</sup>	5.763×10 <sup>-12</sup>
F13	FEP	200,000	8.1×10 <sup>-3</sup>	7.7×10 <sup>-4</sup>
	OGA/Q	112,612	0	0
	M-L	120,176	9.74160	4.6376×10 <sup>-1</sup>
	LEA	110,031	4.247×10 <sup>-19</sup>	4.236×10 <sup>-19</sup>
	GACO	42,350	7.816×10 <sup>-20</sup>	8.336×10 <sup>-20</sup>
	РВСО	19,280	4.495×10 <sup>-15</sup>	1.23×10 <sup>-17</sup>
F14	ALEP	150,000	4.185×10 <sup>-2</sup>	5.969×10 <sup>-2</sup>
	FEP	500,000	1.6×10 <sup>-2</sup>	1.4×10 <sup>-2</sup>
	OGA/Q	112,576	0	0
	M-L	155,783	2.21994	5.0449×10 <sup>-1</sup>
	LEA	110,604	6.783×10 <sup>-18</sup>	5.429×10 <sup>-18</sup>
	GACO	379,600	2.512×10 <sup>-20</sup>	3.316×10 <sup>-20</sup>
	РВСО	35,600	7.427×10 <sup>-15</sup>	7.427×10 <sup>-15</sup>
		500.000	2 0 10-1	5 0 10-l
F15	FEP	500,000	3.0×10 <sup>-1</sup>	5.0×10 <sup>-5</sup>
F15	FEP OGA/Q	500,000 112,893	3.0×10 <sup>-</sup>	5.0×10 °
F15	FEP OGA/Q M-L	500,000 112,893 125,439	0 5.5755×10 <sup>-1</sup>	0 3.9968×10 <sup>-2</sup>
F15	FEP OGA/Q M-L LEA	500,000 112,893 125,439 111,105	0 5.5755×10 <sup>-1</sup> 2.683×10 <sup>-16</sup>	0 3.9968×10 <sup>-2</sup> 6.257×10 <sup>-17</sup>
F15	FEP OGA/Q M-L LEA GACO	500,000 112,893 125,439 111,105 28,750	0 5.5755×10 <sup>-1</sup> 2.683×10 <sup>-16</sup> 1.834×10 <sup>-17</sup>	$\begin{array}{c} 5.0 \times 10^{-10} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \end{array}$
F15	FEP OGA/Q M-L LEA GACO PBCO	500,000 112,893 125,439 111,105 28,750 2,610	$\begin{array}{c} 3.0 \times 10^{-1} \\ 0 \\ 5.5755 \times 10^{-1} \\ 2.683 \times 10^{-16} \\ 1.834 \times 10^{-17} \\ 5.412 \times 10^{-14} \end{array}$	$5.0 \times 10^{-10}$ $0$ $3.9968 \times 10^{-2}$ $6.257 \times 10^{-17}$ $1.681 \times 10^{-17}$ $5.23 \times 10^{-15}$
F15 F16	FEP OGA/Q M-L LEA GACO PBCO ALEP	500,000 112,893 125,439 111,105 28,750 2,610 3,000	$\begin{array}{c} 0\\ 5.5755 \times 10^{-1}\\ 2.683 \times 10^{-16}\\ 1.834 \times 10^{-17}\\ 5.412 \times 10^{-14}\\ -1.031\end{array}$	$\begin{array}{c} 5.0 \times 10^{-1} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \end{array}$
F15 F16	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000	$\begin{array}{c} 0\\ 5.5755 \times 10^{-1}\\ 2.683 \times 10^{-16}\\ 1.834 \times 10^{-17}\\ 5.412 \times 10^{-14}\\ -1.031\\ -1.03\end{array}$	$5.0 \times 10^{-7}$ $0$ $3.9968 \times 10^{-2}$ $6.257 \times 10^{-17}$ $1.681 \times 10^{-17}$ $5.23 \times 10^{-15}$ $0.00$ $4.9 \times 10^{-7}$
F15 F16	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000 13,592	$\begin{array}{c} 3.0 \times 10^{-1} \\ 0 \\ 5.5755 \times 10^{-1} \\ 2.683 \times 10^{-16} \\ 1.834 \times 10^{-17} \\ 5.412 \times 10^{-14} \\ -1.031 \\ -1.03 \\ -1.02662 \end{array}$	$5.0 \times 10^{-17}$ $0$ $3.9968 \times 10^{-2}$ $6.257 \times 10^{-17}$ $1.681 \times 10^{-17}$ $5.23 \times 10^{-15}$ $0.00$ $4.9 \times 10^{-7}$ $5.265 \times 10^{-3}$
F15	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000 13,592 10,823	$\begin{array}{c} 0\\ 5.5755 \times 10^{-1}\\ 2.683 \times 10^{-16}\\ 1.834 \times 10^{-17}\\ 5.412 \times 10^{-14}\\ -1.031\\ -1.03\\ -1.02662\\ -1.03108\end{array}$	$\begin{array}{c} 5.0 \times 10^{-17} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \end{array}$
F15	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA GACO	500,000 112,893 125,439 111,105 28,750 2,610 3,000 13,592 10,823 7,050	$\begin{array}{c} 3.0 \times 10^{-1} \\ 0 \\ 5.5755 \times 10^{-1} \\ 2.683 \times 10^{-16} \\ 1.834 \times 10^{-17} \\ 5.412 \times 10^{-14} \\ \hline -1.031 \\ -1.03 \\ -1.02662 \\ -1.03108 \\ -1.031628 \end{array}$	$\begin{array}{c} 5.0 \times 10^{-7} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \\ 1.050 \times 10^{-8} \end{array}$
F15	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA GACO PBCO	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000 13,592 10,823 7,050 400	$\begin{array}{c} 3.0 \times 10^{-1} \\ 0 \\ 5.5755 \times 10^{-1} \\ 2.683 \times 10^{-16} \\ 1.834 \times 10^{-17} \\ 5.412 \times 10^{-14} \\ \hline -1.031 \\ -1.03 \\ -1.02662 \\ -1.03108 \\ -1.031628 \\ -1.031628 \\ -1.031628 \end{array}$	$\begin{array}{c} 5.0 \times 10^{-7} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \\ 1.050 \times 10^{-8} \\ 1.029 \times 10^{-9} \end{array}$
F15 F16 F17	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA GACO PBCO FEP	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000 13,592 10,823 7,050 400 10,000	$\begin{array}{c} 3.0 \times 10^{-1} \\ 0 \\ 5.5755 \times 10^{-1} \\ 2.683 \times 10^{-16} \\ 1.834 \times 10^{-17} \\ 5.412 \times 10^{-14} \\ \hline -1.031 \\ -1.03 \\ -1.02662 \\ -1.031628 \\ -1.031628 \\ \hline 0.398 \end{array}$	$\begin{array}{c} 5.0 \times 10^{-7} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \\ 1.050 \times 10^{-8} \\ 1.029 \times 10^{-9} \\ \hline 1.5 \times 10^{-7} \end{array}$
F15 F16 F17	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA GACO PBCO FEP M-L	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000 13,592 10,823 7,050 400 10,000 12,703,	$\begin{array}{c} 3.0 \times 10^{-1} \\ 0 \\ 5.5755 \times 10^{-1} \\ 2.683 \times 10^{-16} \\ 1.834 \times 10^{-17} \\ 5.412 \times 10^{-14} \\ -1.031 \\ -1.03 \\ -1.02662 \\ -1.03108 \\ -1.031628 \\ -1.031628 \\ 0.398 \\ 0.403297 \end{array}$	$\begin{array}{c} 5.0 \times 10^{-7} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \\ 1.050 \times 10^{-8} \\ 1.029 \times 10^{-9} \\ \hline 1.5 \times 10^{-7} \\ 8.832 \times 10^{-3} \end{array}$
F15 F16 F17	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA GACO PBCO FEP M-L LEA	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000 13,592 10,823 7,050 400 10,000 12,703, 10,538,	$3.0 \times 10^{-1}$ 0 $5.5755 \times 10^{-1}$ $2.683 \times 10^{-16}$ $1.834 \times 10^{-17}$ $5.412 \times 10^{-14}$ -1.031 -1.03 -1.02662 -1.031628 -1.031628 0.398 0.403297 0.398	$\begin{array}{c} 5.0 \times 10^{-7} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \\ 1.050 \times 10^{-8} \\ 1.029 \times 10^{-9} \\ \hline 1.5 \times 10^{-7} \\ 8.832 \times 10^{-3} \\ 2.652 \times 10^{-8} \end{array}$
F15 F16 F17	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA GACO FEP M-L LEA GACO	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000 13,592 10,823 7,050 400 10,000 12,703, 10,538, 2,900	$\begin{array}{c} 3.0 \times 10^{-1} \\ 0 \\ 5.5755 \times 10^{-1} \\ 2.683 \times 10^{-16} \\ 1.834 \times 10^{-17} \\ 5.412 \times 10^{-14} \\ \hline -1.031 \\ -1.03 \\ -1.02662 \\ -1.031628 \\ -1.031628 \\ -1.031628 \\ 0.398 \\ 0.403297 \\ 0.398 \\ 0.39806762 \end{array}$	$\begin{array}{c} 5.0 \times 10^{-7} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \\ 1.050 \times 10^{-8} \\ 1.029 \times 10^{-9} \\ \hline 1.5 \times 10^{-7} \\ 8.832 \times 10^{-3} \\ 2.652 \times 10^{-8} \\ 7.939 \times 10^{-5} \end{array}$
F15 F16 F17	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA GACO FEP M-L LEA GACO PBCO	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000 13,592 10,823 7,050 400 10,000 12,703, 10,538, 2,900 520	0 $5.5755 \times 10^{-1}$ $2.683 \times 10^{-16}$ $1.834 \times 10^{-17}$ $5.412 \times 10^{-14}$ -1.031 -1.02662 -1.031628 -1.031628 0.398 0.403297 0.398 0.39806762 0.398	$\begin{array}{c} 5.0 \times 10^{-7} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \\ 1.050 \times 10^{-8} \\ 1.029 \times 10^{-9} \\ \hline 1.5 \times 10^{-7} \\ 8.832 \times 10^{-3} \\ 2.652 \times 10^{-8} \\ 7.939 \times 10^{-5} \\ 3.673 \times 10^{-8} \end{array}$
F15 F16 F17 F18	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA GACO PBCO FEP M-L LEA GACO PBCO ALEP	500,000 112,893 125,439 111,105 28,750 2,610 3,000 13,592 10,823 7,050 400 10,000 12,703, 10,538, 2,900 520 3,000	$3.0 \times 10^{-1}$ 0 $5.5755 \times 10^{-1}$ $2.683 \times 10^{-16}$ $1.834 \times 10^{-17}$ $5.412 \times 10^{-14}$ -1.031 -1.02662 -1.031628 -1.031628 0.398 0.403297 0.398 0.39806762 0.398 3.000	$\begin{array}{c} 5.0 \times 10^{-7} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \\ 1.050 \times 10^{-8} \\ 1.029 \times 10^{-9} \\ \hline 1.5 \times 10^{-7} \\ 8.832 \times 10^{-3} \\ 2.652 \times 10^{-8} \\ 7.939 \times 10^{-5} \\ 3.673 \times 10^{-8} \\ \hline 0.000 \end{array}$
F15 F16 F17 F18	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA GACO PBCO FEP M-L LEA GACO PBCO	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000 13,592 10,823 7,050 400 10,000 12,703, 10,538, 2,900 520 3,000 10,000	0 $5.5755 \times 10^{-1}$ $2.683 \times 10^{-16}$ $1.834 \times 10^{-17}$ $5.412 \times 10^{-14}$ -1.031 -1.02662 -1.031628 -1.031628 0.398 0.403297 0.398 0.39806762 0.398 3.000 3.02	$\begin{array}{c} 5.0 \times 10^{-1} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \\ 1.050 \times 10^{-8} \\ 1.029 \times 10^{-9} \\ \hline 1.5 \times 10^{-7} \\ 8.832 \times 10^{-3} \\ 2.652 \times 10^{-8} \\ 7.939 \times 10^{-5} \\ 3.673 \times 10^{-8} \\ \hline 0.000 \\ 1.1 \times 10^{-1} \end{array}$
F15 F16 F17 F18	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA GACO FEP M-L LEA GACO PBCO ALEP FEP M-L	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000 13,592 10,823 7,050 400 10,000 12,703, 10,538, 2,900 520 3,000 10,000 16,325	$3.0 \times 10^{-1}$ 0 $5.5755 \times 10^{-1}$ $2.683 \times 10^{-16}$ $1.834 \times 10^{-17}$ $5.412 \times 10^{-14}$ -1.031 -1.02662 -1.031628 -1.031628 0.398 0.403297 0.398 0.39806762 0.398 3.000 3.02 3.048855	$\begin{array}{c} 5.0 \times 10^{-7} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \\ 1.050 \times 10^{-8} \\ 1.029 \times 10^{-9} \\ \hline 1.5 \times 10^{-7} \\ 8.832 \times 10^{-3} \\ 2.652 \times 10^{-8} \\ 7.939 \times 10^{-5} \\ 3.673 \times 10^{-8} \\ \hline 0.000 \\ 1.1 \times 10^{-1} \\ 6.037 \times 10^{-2} \end{array}$
F15 F16 F17 F18	FEP OGA/Q M-L LEA GACO PBCO ALEP FEP M-L LEA GACO PBCO ALEP FEP M-L LEA FEP M-L LEA	500,000 112,893 125,439 111,105 28,750 2,610 3,000 10,000 13,592 10,823 7,050 400 10,000 12,703, 10,538, 2,900 520 3,000 10,000 16,325 11,721	$3.0 \times 10^{-1}$ 0 $5.5755 \times 10^{-1}$ $2.683 \times 10^{-16}$ $1.834 \times 10^{-17}$ $5.412 \times 10^{-14}$ -1.031 -1.02662 -1.031628 0.398 0.403297 0.398 0.403297 0.398 0.39806762 0.398 3.000 3.02 3.048855 3.00003	$\begin{array}{c} 5.0 \times 10^{-1} \\ 0 \\ 3.9968 \times 10^{-2} \\ 6.257 \times 10^{-17} \\ 1.681 \times 10^{-17} \\ 5.23 \times 10^{-15} \\ \hline 0.00 \\ 4.9 \times 10^{-7} \\ 5.265 \times 10^{-3} \\ 3.364 \times 10^{-7} \\ 1.050 \times 10^{-8} \\ 1.029 \times 10^{-9} \\ \hline 1.5 \times 10^{-7} \\ 8.832 \times 10^{-3} \\ 2.652 \times 10^{-8} \\ 7.939 \times 10^{-5} \\ 3.673 \times 10^{-8} \\ \hline 0.000 \\ 1.1 \times 10^{-1} \\ 6.037 \times 10^{-2} \\ 6.248 \times 10^{-5} \end{array}$

	GACO	5,300	3.0000	0
	PBCO	3,000	3.000025	0
F19	FEP	400,000	5.0×10 <sup>-4</sup>	3.2×10 <sup>-4</sup>
	M-L	186,768	1.344×10 <sup>-3</sup>	2.977×10 <sup>-4</sup>
	LEA	55,714	3.512×10 <sup>-4</sup>	7.361×10 <sup>-5</sup>
	GACO	12,850	6.802×10 <sup>-4</sup>	3.978×10 <sup>-4</sup>
	PBCO	2,500	0.0003075	0
F20	FEP	20,000	-3.27	5.9×10 <sup>-2</sup>
	M-L	92,516	-3.12696	6.739×10 <sup>-2</sup>
	LEA	28,428	-3.301	7.832×10 <sup>-3</sup>
	GACO	4,250	-3.3194523	5.061×10 <sup>-4</sup>
	PBCO	840	-3.32	0

- a. M.F.E denotes the average number of function evaluations to reach the optimum value
- b.  $G_{min}$  indicates the mean of the best value found in the last generation of 50 runs.
- c. S.D stands for Standard Deviation

 Table 4. Results of PBCO for the functions with 100 dimensions

Func.	Algo.	M.F.E	G <sub>min</sub>	S.D
F2	GACO	20,600	0	0
	PBCO	5,180	3.870×10 <sup>-13</sup>	3.870×10 <sup>-13</sup>
F3	GACO	30,150	1.358×10 <sup>-13</sup>	1.58×10 <sup>-13</sup>
	PBCO	7,325	6.3239×10 <sup>-12</sup>	6.3239×10 <sup>-12</sup>
F4	GACO	12,950	0	0
	PBCO	4,625	1.9809×10 <sup>-14</sup>	1.980×10 <sup>-14</sup>
F5	GACO	17,850	4.7116×10 <sup>-33</sup>	4.7116×10 <sup>-33</sup>
	PBCO	6,625	3.4379×10 <sup>-16</sup>	3.4379×10 <sup>-16</sup>
F6	GACO	18,550	1.3497×10 <sup>-32</sup>	1.3497×10 <sup>-32</sup>
	PBCO	8,370	4.3298×10 <sup>-15</sup>	4.3841×10 <sup>-15</sup>
F7	GACO	9,250	-99.26889	6.279×10 <sup>-1</sup>
	PBCO	3,975	-99.26893	2.165×10 <sup>-2</sup>
F8	GACO	3,350	1.788×10 <sup>-7</sup>	2.384×10 <sup>-7</sup>
	PBCO	1,580	3.3412×10 <sup>-10</sup>	1.42×10 <sup>-10</sup>
F9	GACO	8,550	-78.332275	9.2238×10 <sup>-5</sup>
	PBCO	3,750	-78.33233	3.245×10 <sup>-6</sup>
F10	GACO	15,450	0	1.776×10 <sup>-13</sup>
	PBCO	6,500	3.4214×10 <sup>-13</sup>	3.4214×10 <sup>-13</sup>
F11	GACO	50,800	0	0
	PBCO	19,170	6.4037×10 <sup>-14</sup>	6.7037×10 <sup>-14</sup>
F12	GACO	29,650	2.861×10 <sup>-5</sup>	10.746×10 <sup>-4</sup>
	PBCO	4,575	3.8765×10 <sup>-10</sup>	5.763×10 <sup>-12</sup>

International Journal of Computer Applications (0975 – 8887) Volume 177 – No. 8, October 2019

F13	GACO	42,100	$1.08 \times 10^{-19}$	3.14×10 <sup>-19</sup>
	PBCO	18,240	4.4954×10 <sup>-15</sup>	2.34×10 <sup>-17</sup>
F14	GACO	829,450	22.659	28.9189
	PBCO	45,150	2.4276×10 <sup>-6</sup>	1.3243×10 <sup>-7</sup>
F15	GACO	24,900	4.417×10 <sup>-18</sup>	1.177×10 <sup>-7</sup>
	PBCO	3,840	9.7651×10 <sup>-14</sup>	4.298×10 <sup>-13</sup>

#### 3.2 Results on Function Optimization

We have tested and compared the performance of the PBCO technique with some optimization approaches and the results have shown in the Table 3. Table 4 shows the optimum value achieved by the PBCO method for the function  $F_2 - F_{15}$  when their dimensions increased to 100. The benchmark functions listed in Table 1 has tested on various other optimization techniques. Among them, ALEP (Adaptive Levy Evolutionary Programming) [16] uses evolutionary programming with adaptive Levy mutation in order to generate an offspring for each new generation. It has also noted in [16] that the nonadaptive algorithm can never outperform ALEP. The OGA/Q (Orthogonal Genetic Algorithm with Quantization) [17] uses an orthogonal design to construct a crossover operator. Another evolutionary approach based method; FEP (Fast Evolutionary Programming) [18] uses evolutionary programming with Cauchy mutation to generate an offspring for each new generation. Hong and Quan [19] proposed a theoretical approach called the mean-value-level-set method (M-L method) by improving the mean of the objective function value on the level set. Wang and Dang [20] designed the LEA (Level-set Evolutionary Algorithm) for global optimization with Latin squares. Its application leads to an effective crossover operator. Table 3 contains the results of various optimization methods along with that of the PBCO technique. We can see from Table 3 that for high dimensional functions  $(F_1 - F_{15})$ , the proposed method PBCO performs better than or similar to the other optimization techniques. The PBCO also reaches the global or near-global optimum faster than the other optimization approaches. For F<sub>14</sub>, the PBCO algorithm reaches the global optimum solution if we set the value of u to 10 for search space reduction in multiple stages. Hence in case of F14, the proposed method takes longer compared to other techniques (except FEP) to achieve the best solution. For the remaining low dimensional functions (F<sub>16</sub> - $F_{20}$ ) in Table 3, it has noticed that the proposed method (PBCO) can reach the global or near-global optimum. However, the PBCO approach takes more time (in terms of mean function evaluation) to reach the global optimum for functions F<sub>16</sub> and F<sub>18</sub> compared to the optimization technique ALEP. Finally, it has observed that the PBCO technique performs either better or similar for most of the benchmark functions (F1 -F20) in terms of achieving the global or nearglobal optimum in lesser time compared to the other optimization methods.

We have further tested the functions F2 –F15 with 100 dimensions (see Table 4) to evaluate the effectiveness of the proposed method to achieve the global optimum for Problems with higher dimensionality. We have not considered the other functions (except F2 – F15) since the optimum value would be different for them with higher dimensions and we had no gold standard for the others. It was noted that the PBCO approach can reach the global or near-global optimum easily without spending much time compared to the timings for the functions F2 –

F15 with 30 dimensions as shown in Table 4. However for F14 with 100 dimensions, the technique cannot achieve the global optimum but it is near to the global optimum value. Here, we have used u=10 but it still cannot fully converge to near-global optimum. In Table 4 also, we have provided comparative results on the set of functions with higher dimensionally (100). finally, it is concluded that the PBCO achieves the global or near-global optimum value efficiently for all functions with low and high (30 and 100) dimensionality.

#### 4. CONCLUSION

The PBCO method performs better than other relevant optimization techniques for the mathematical functions with 30 dimensions as observed by the results in Table 3. In case of F14, we have considered u=10 so that the proposed technique converges to the global optimum. It is observed (for dimension = 30) that the PBCO algorithm converges to the global optimum, but is slower than the other optimization techniques except FEP for F14. For low dimensional functions (F16 - F20), the proposed method performs better in achieving the global or near-global optimum but is slower (in terms of mean function evaluation) for F16 and F18 compared to ALEP. We have also evaluated the performance of the PBCO approach for the mathematical functions F2 - F15 with 100 dimensions. It is noticed that the proposed method can reach the global or near-global optimum smoothly as it reached for the same set of functions with 30 dimensions. However, the PBCO approach cannot reach the near-global optimum for F14, even if we set the value of u to 10.

#### 5. REFERENCES

- [1] DT Pham and MCastellani The Bees Algorithm: modelling foraging behaviour to solve continuous optimization problems, PROCEEDINGS OF THE INSTITUTION OF MECHANICAL ENGINEERS PART C JOURNAL OF NGINEERING SCIENCE 1989-1996 (VOLS 203-210) · DECEMBER 2009.
- [2] Wenping Zou, Yunlong Zhu, Hanning Chen, and Beiwei Zhang Solving Multiobjective Optimization Problems Using Artificial Bee Colony Algorithm, Hindawi Publishing Corporation Discrete Dynamics in Nature and Society Volume 2011, Article ID 569784.
- [3] Reza Akbari, Alireza Mohammadi, Koorush Ziarati A novel bee swarm optimization algorithm for numerical function optimization, Department of Computer Science and Engineering, Shiraz University, Shiraz, Iran.
- [4] Dogan Aydn, Tianjun Liao, Marco A. Montes de Oca, Thomas St•utzleI Improving Performance via Population Growth and Local Search: The Case of the Artificial Bee Colony Algorithm, IRIDIA Technical Report Series Technical Report No. TR/IRIDIA/2011-015 August 2011.
- [5] Rui Zhang, An Artificial Bee Colony Algorithm Based on Problem Data Properties for Scheduling Job Shops, School of Economics and Management, Nanchang University, Nanchang 330031, PR China
- [6] Ali R. Yildiz, Optimization of cutting parameters in multi-pass turning using artificial bee colony-based approach, Bursa Technical University, Department of Mechanical Engineering, Bursa, Turkey

- [7] Dervis Karaboga, Celal Ozturk\*, A novel clustering approach: Artificial Bee Colony (ABC) algorithm, Erciyes University, Intelligent Systems Research Group, Department of Computer Engineering, Kayseri, Turkey
- [8] S.A.M. Fahad, M.E.El-Hawary, Overview of Artificial Bee Colony (ABC) algo-rithm and its applications, in: Proceedings of IEEE Conference on Systems, 2012, pp.1–6.
- [9] D. Karaboga, An idea based on honey bee swarm for numerical optimization , TechnicalReport-TR06, ErciyesUniversity Kayseri, Turkey, 2005, pp.1–10
- [10] Y. Xin -She, Engineering Optimization: An Introduction with Metaheuristic Applications, Wiley.com, Hoboken, NJ, 2010.
- [11] Dervis Karaboga · Bahriye Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, © Springer Science+Business Media B.V. 2007
- [12] Efr'en Mezura-Montes\_ and Omar CetinaDo '1 guez, Empirical Analysis of a Modified Artificial Bee Colony for Constrained Numerical Optimization, Laboratorio Nacional de Informatics Avanade (LANIA A.C.) R'ebsamen 80, Centro, Xalapa, Veracruz, 91000, MEXICO
- [13] Miloš Nikolic', Duša Teodorovic, Empirical study of the Bee Colony Optimization (BCO) algorithm, Faculty of Transport and Traffic Engineering, University of Belgrade, Vojvode Stepe 305, 11000 Belgrade, Serbia
- [14] Bai Li a, Ya Li b,n, Ligang Gong, Protein secondary structure optimization using an improved artificial bee colony algorithm based on AB off-lattice model, Engineering Applications of Artificial Intelligence
- [15] Chin Soon Chong, Malcolm Yoke Hean Low, Appa Iyer Sivakumar, Kheng Leng Gay, A BEE COLONY OPTIMIZATION ALGORITHM TO JOB SHOP SCHEDULING, Proceedings of the 2006 Winter Simulation Conference
- [16] C. Y. Lee and X. Yao, Evolutionary programming using mutations based on the levy probability distribution, IEEE Trans. Evol. Comput., vol. 8, no. 1, pp. 1-13, Feb. 2004
- [17] Y. W. Leung and Y. P. Wang, An orthogonal genetic algorithm with quantization for global numerical optimization, IEEE Trans. Evol. Comput., vol. 5, no. 1, pp. 41-53, Feb 2001
- [18] X. Yao, Y.Liu, and G. Lin, Evolutionary programming made faster, IEEE Trans. Evol. Comput., vol. 3, no. 2, pp. 82-102, July 1999
- [19] C. S. Hong and Z Quan, 1988. Integeral global optimization: Theory implementation and application . SpringerVerlag, Berlin, Germany
- [20] Y. Wang and C. Dang, An evolutionary algorithm for global optimization based on Level-set evolution and Latin squares IEEE Trans. On Evol. Comput., vol. 11, no. 5, October 2007, pp. 579-595