Artificial Bee Colony Algorithm: A Survey

2 authors, including:

Pawn Bhambu
arya college of eng and it

5 PUBLICATIONS   **2** CITATIONS

Some of the authors of this publication are also working on these related projects:

Project   Self balanced particle swarm optimization View project

Artificial Bee Colony Algorithm: A Survey

Sangeeta Sharma
Arya College of Engineering and IT,
Jaipur, India

Pawan Bhambu
Arya College of Engineering and IT,
Jaipur, India

ABSTRACT

Artificial bee colony optimization algorithm is one of the popular swarm intelligence technique anticipated by D. Karaboga in year 2005. Since its inception, this algorithm was modified by a number of researchers and applied in different areas of engineering, science and management to solve very complex problems. This algorithm is very simple to implement and has the least number of control parameters. In the last two decades, a large number of new algorithm based on natural phenomenon like artificial bee colony algorithm are developed and used to find solution of many real world problems. This paper provides a state of the art survey of ABC algorithm and analysis of its performance with different size of population.

General Terms

Algorithms

Keywords

Nature Inspired Algorithm, Memetic algorithm, Swarm intelligence, Evolutionary computation.

1. INTRODUCTION

Swarm Intelligence based algorithm are very popular now a days while solving complex problem in field of engineering, management and science. These algorithms are inspired by some natural phenomenon and called Nature Inspired Algorithms (NIAs). The NIAs mimics the intelligent behaviour of social insects like bees, ants, termites, fish, birds, etc. Swarm Intelligence getting popularity now days and become a rising and fascinating area. It depends on the cooperative behaviour of societal living thing. Societal individual makes use of their skill of societal wisdom to crack multifaceted everyday jobs. The main power of swarm based optimization strategy is multiple interactions in societal colonies. Swarm intelligence strategies have the potential to solve complex factual world optimization problems as the preceding study have exposed.

Recently D. Karaboga proposed a very simple and easy to implement strategy motivated by extraordinary food foraging behaviour of honey bee insects and named it as artificial bee colony algorithm [1]. Resembling to other population based optimization algorithms, this algorithm also has a population of budding solutions. Food source for a honey bee represent one possible solution. Fitness of a particular food source, computes its quality that represents the amount of nectar in a food source. Performance of the ABC algorithm depends on the steadiness between searching of local search space and utilization of best feasible outcomes. Every so often it is practical that the ABC prevents proceeding headed for the global most favourable despite the fact that the local optimum not achieved.

Nature teaches us that how to solve very compound problems with different degree of complexity resourcefully and successfully. NIAs encouraged by various natural perceptible facts and one can classify NIAs according to per their source of motivation. Most important types of NIA are: Evolutionary Algorithms, Physical Algorithms, Immune Algorithms, Neural Algorithms, Probabilistic Algorithms, Stochastic Algorithms and Swarm Algorithms [2].

Rest of the paper is organized as follows: Sect. 2 illustrates concise overview of the basic ABC. Recent modifications in ABC algorithm are discussed in Sect. 3. In Sect. 4, discuss applications of ABC algorithm. Performance analysis of ABC is done in section 5. Finally, in Sect. 6, paper is concluded.

2. ARTIFICIAL BEE COLONY ALGORITHM

The ABC strategy is somewhat a clear-cut, swift and population based stochastic search strategy in the area of nature-inspired optimization strategies. The position of swarm updates in ABC by two contradictory activities: first one is a process of adaptation, which empowers exploring the diverse search space, and the second one is a process of selection, which ensures the exploitation of the earlier experience. Sometimes it is observed that ABC stops moving in the direction of global optimum despite the fact that the population has not congregate to a restricted most advantageous [6]. It can be experiential to facilitate the solution investigation equation of ABC is fine at exploration, however pitiable at exploitation [3]. For that reason, it is enormously enviable to develop a new approach which is able to exploit better solutions in its neighborhood and also able to explore the search space for less fit solutions in order to uphold the appropriate balance among exploration and exploitation activities of ABC.

The ABC metaheuristic technique is stimulated through the spontaneous food foraging behavior of the honey bee creature. Honey bee insect most intuitive creation of nature; it shows combined intellectual behavior at the same time as penetrating the food. The honey bee can memorize the ecological circumstances, can accumulate and distribute the information and can decide according to these observations. As per the changes in the surroundings, the bee updates its position, assign the responsibilities dynamically and go on further by means of societal erudition and education. This extraordinary conduct of honey bees motivates research scientists to imitate the intellectual food foraging behavior of the bees.

2.1 Phases of ABC Algorithm

The investigation process of ABC has three key steps [1]:

- Employ the employed bees to a source of food and compute it's amount of nectar;
- According to information collected from employed bees onlooker bees select the food sources and calculate approximately their quality of nectar;
- Discover the scout bees and take advantage of them on promising food sources for the purpose of exploitation.

2.1.1 Initialization of Swarm

The ABC algorithm that mimics the extraordinary food foraging behaviour of real honey bees has three key constraints: first is a population that is considered as a number

of food sources, the second constraint is a limit that is the number of tries following to which a food source is rejected and third constraint is the norm for stop the process also known as maximum number of cycles. D. Karaboga [1] suggested that ABC will give best results if the number of employed bees and onlooker bees are taken equal that are half of the initial population. It is suggested that the initial population must by evenly distributed. Each food source is symbolized by a D-dimensional vector x_i (i = 1, 2 ...SN) and all food sources are initialized using Eq.(1)[3]:

$$x_{ij} = x_{\min j} + rand[0,1](x_{\max j} - x_{\min j})$$
(1)

Where, rand[0,1] is a function that randomly engenders a regularly scattered arbitrary number in the range [0,1].

2.1.2 Employed Bee Phase
The second phase of ABC algorithm is employed bee phase. The position of current solutions modernized with the help of the knowledge of an individual's understanding and the appropriateness of the recently established solutions. Existing food sources replaced with the innovative food source having better value of fitness. The position of jth dimension of ith candidate modernizes using Eq. (2) [3]:

$$v_{ij} = x_{ij} + \phi(x_{ij} - x_{kj})$$
(2)

Here $\phi(x_{ij} - x_{kj})$ is identified as step size. In this formula j and k are two arbitrarily selected indices. Such that $k \in \{1, 2, ..., SN\}$, $j \in \{1, 2, ...,D\}$ and $k \neq i$ make certain that there must be some significant enhancement in step size.

2.1.3 Onlooker Bee Phase
The third key phase of ABC algorithm is onlooker bee phase. The counting of onlooker bees is identical with the quantity of employed bees. For the period of this segment all employed bee share quality of novel food sources through onlooker bees in form of fitness. Every food source judged based on its probability of selection. There are various methods to compute the probability, generally it calculated in term of fitness. The highly fitted solution gets elected by the onlooker. There are various techniques for computation of possibility; however, it have to be a function of fitness. The fitness of each food source is determined by its function value as per Eq. (3) [3]:

$$
\begin{aligned}
&if \ (function \ value > 0) \\
&Fitness = \frac{1}{2 \times FunctionValue + 1} \\
&else \\
&Fitness = 1 + fabs(\frac{1}{FunctionValue})
\end{aligned}
$$
(3)

The selection of probability for each food source is determined by its fitness as per Eq. (4) [3]:

$$P_{ij} = \frac{fit_i}{\sum_{i=1}^{SN} fit_i}$$
(4)

Algorithm 1: Artificial Bee Colony Algorithm

Initialize all the essential parameters;

Reiterate till the criteria for termination do not meet

Step 1: Employed bee phase for calculating new food sources.

Step 2: Onlooker bees phase for position update of the food sources according to the quality of food sources.

Step 3: Scout bee phase generate new food sources in place of discarded food sources.

Step 4: Remember the most excellent food source recognized up to now.

2.1.4 Scout Bee Phase
The final phase of ABC algorithm is a scout bee phase in which some new food sources engendered as opposed to discarded food sources due to low quality. In case when the location of a particular food source is not modernized for a threshold (in term of number of cycles), that food source is derelict and a new phase starts named scout bees phase. The bees that are allied in the midst of the deserted source of food transformed into a scout bee and the food source is substituted by the variably elected source of food within the exploration space. New food sources generated using Eq. (5) [3].

$$x_{ij} = x_{\min j} + rand[0,1](x_{\max j} - x_{\min j})$$
(5)

As discussed in above four phases, it can be concluded that the ABC algorithm has three key constraints. That is the maximum number of cycles, the limit and SN (that is a number of food sources). The ABC algorithm is summarized in algorithm 1.

3. RECENT MODIFICATIONS IN ABC ALGORITHM
The Artificial Bee Colony algorithm is very popular and simple algorithm to solve complex optimization problems. There are lot of complex real world problem that are not solvable by conventional methods. In order to solve this type of problem population base techniques are very helpful. The use of intelligence appeared from collective actions of a swarm that is by and large used to crack multifaceted problems when an individual is not able to find solution of a meticulous problem. G. Yavuz and D. Aydin [7] proposed an Angle modulated Artificial Bee Colony algorithms for feature selection using angle modulation technique. Y. Marinakis et al. proposed A Hybrid Discrete Artificial Bee Colony Algorithm for the Multicast Routing Problem [8] implemented with Variable Neighbourhood Search. N. Sharma et al. [9] proposed a new variant of ABC namely Modified Artificial Bee Colony Algorithm Based on Disruption Operator. This algorithm added a new phase in ABC algorithm namely disruption phase. This algorithm modify all solutions except best one using the distance from current best solution and disrupt or attract new solution. TK Sharma and Millie Pant [10] Shuffled artificial bee colony algorithm. The Shuffled-ABC is hybrid of ABC algorithm and shuffled frog-leaping algorithm. The Shuffled-ABC divides the initial population into two different groups based on their fitness. The ABC applied in best fitted group and SFLA applied on rest population.

A. Yurtkuran and Erdal Emel investigated a discrete artificial bee colony algorithm and used it to solve the problem of scheduling on a single machine [11]. X. Li and Guangfei Yang introduced artificial bee colony algorithm with memory [11]. This algorithm memorizes the previous successful foraging experiences. U. Saif et al.[13] proposed a new hybrid of ABC for assembly line balancing with task time variations and named it Hybrid Pareto artificial bee colony algorithm. This algorithm make use of Pareto concepts, used diverse neighbours of food sources for each employee bee and also make use of crossover and mutation operation in its structure. C. Caraveo et al. [14] anticipated a new bee colony algorithm with fuzzy dynamic parameter adaptation and applied for optimization of fuzzy controller design. L. Lv et al. [15]

developed an artificial bee colony algorithm with accelerating convergence.

S. Kumar et al. [16] modified the onlooker bee phase of ABC algorithm in order to improve performance for function optimization and also introduced improved memetic search in ABC [17]. D. Karaboga and Selcuk Aslan [18] proposed a new emigrant creation strategy for parallel artificial bee colony algorithm. This algorithm presents a new technique for increasing the quality of the distributed source by combining best solutions. C. Ozturk et al. [19] proposed a genetic operator based binary artificial bee colony algorithm. This modification does not follow classical neighbourhood search method. It arbitrarily select two food sources and employ two point crossover between them then select one of them after swapping. S. Kumar et al. [20][21] introduced two new strategy in basic ABC algorithm by enhancing local search strategy and randomised memetic search. Both algorithms improve balance between exploration and exploitation of local search space. Randomized memetic ABC introduced a new parameter in ABC algorithm. P. K. Singhal et al. [22] introduces a new strategy in ABC algorithm for unit commitment problem which is based on measure of dissimilarity between binary solutions. A crossover based ABC algorithm proposed by S. Kumar et al. [23]. This algorithm combine crossover operator from genetic algorithm into ABC algorithm. S. Pandey and S. Kumar introduced an enhanced version of ABC algorithm [24] and deployed it to get solution of travelling salesman problem. A Novel Hybrid Memetic Search in Artificial Bee Colony Algorithm [25] developed by S. Kumar et al. by combining levy flight search and golden section search in order to improve exploration of local search space. MS Kıran and Oğuz Fındık [25] incepted a directed artificial bee colony algorithm. This algorithm introduced a new parameter namely modification rate and position updates based on directional information. A memetic search in ABC with position update based on fitness of individual was proposed by S. Kumar et al. [27]. This algorithm modifies position of solutions based on their fitness and includes one more phase as local search phase using golden section search. P Mansouri et al. [27] combined bisection method with ABC algorithm in order to find fixed point of non linear functions. List of new variants of ABC algorithm is endless there are a large number of research paper based on ABC algorithm exits in literature.

4. APPLICATIONS OF ABC ALGORITHM

Since its origination, the ABC algorithm has become extraordinarily fashionable as it has less number of control parameters; it is robust and easy to apply. It is successfully applied to the problems from different application areas by many researchers. Applications of ABC algorithms listed by [28] with subject area of use. The ABC algorithm has application in field of computer science, electrical engineering, mechanical engineering and electronics engineering. Artificial bee colony algorithm applied for feature selection [7], multicast routing [8], single machine scheduling [11], balancing in assembly line [13], designing of controller [14] and to find solution of travelling salesman problem [24].

List of applications of ABC algorithm are endless, there are almost every field including engineering, science and management where ABC algorithm is in use with priority over other competitive optimization algorithm.

5. ANALYSIS OF ARTIFICIAL BEE COLONY ALGORITHM FOR OPTIMIZATION PROBLEMS

This chapter presents performance analysis of original ABC algorithm. It includes comparison performance of ABC algorithm with different values of parameters. This chapter also provides detailed comparative analysis of ABC with other nature inspired algorithms.

5.1 Analysis of ABC algorithm

The major parameters in ABC algorithm are: Number of Food Sources, Limit and a uniform random number. The uniformly generated random number does not depend on any component as it is randomly generated during each iteration. The number of food sources analogous to population size. Performance of ABC algorithm depends on size of initial population. If population size increases then after a certain limit the performance of ABC deteriorates sharply. The parameter limit also changes with variation in population size as it is product of dimension and population size. Thus change in limit has some effect on performance of ABC algorithm.

The set of 35 standard test problems are given in table 1. It contains unimodal-multimodal problems, constrained and unconstrained problem. Benchmark functions considered in this paper for the purpose of experiment are of diverse uniqueness like uni-model or multi-model and separable or non-separable and of diverse dimensions. Uni-modal function is function having single optimum value like sphere function while multimodal function having more than one optimum value like Rastrigin functions. A function is separable if it can be separable in to components functions otherwise it is non separable. This set of problem includes simple problems as well as complex problems also.

6. CONCLUSION

In the era of science where accuracy and efficiency are major issue while solving complex real world problem from almost each and every field of life demands for robust, fast, and accurate optimizers. The ABC algorithm is one of the simplest swarm intelligence algorithms that offer good result with accuracy for optimization problems with different level of complexity. The ABC algorithm proves that it is best choice when tested for standard benchmark problems and complex real world problem. It can be applied for large class of problems. The ABC algorithm is not insightful to primary parameter values and not affected by the increasing dimension of the problem. The ABC algorithm has some drawbacks also, as sometimes it suffers from premature convergence or stagnation that results in loss of balance between intensification and diversification capabilities. This paper endeavors to present a state of art survey of ABC algorithm. First, it describes the cooperative behavior of honey bees and then the imitation of honey bees for artificial bee colony algorithms is shown. Some latest modification in ABC algorithm and recent applications of it are discussed. Performance of the ABC analyzed with variation in population.

7. REFERENCES

[1] D Karaboga, "An Idea based on honey bee swarm for numerical optimization," Techn. Rep. TR06, Erciyes Univ. Press, Erciyes, 2005.

[2] Brownlee, Clever algorithms: nature-inspired programming recipes. Jason Brownlee, 2011.

[3] D. Karaboga and B. Akay. A comparative study of artificial bee colony algorithm. Applied Mathematics and Computation, 214(1):108–132, 2009.

[4] K. Price, "An Introduction to differential evolution," New Ideas in Optimization. D. Corne, M. Dorigo, and F. Glover (Eds.), London, UK: McGraw Hill, 1999.

[5] Z Michalewicz and DB Fogel. How to Solve It: Modern Heuristics. Springer, 2004.

[6] JC Bansal, H Sharma and SS Jadon "Artificial bee colony algorithm: a survey." Int. J. of Advanced Intelligence Paradigms 5.1 (2013): 123-159.

[7] Yavuz, Gürcan, and Doğan Aydin. "Angle modulated Artificial Bee Colony algorithms for feature selection." Applied Computational Intelligence and Soft Computing 2016 (2016): 7.

[8] Marinakis, Yannis, Magdalene Marinaki, and Athanasios Migdalas. "A Hybrid Discrete Artificial Bee Colony Algorithm for the Multicast Routing Problem." Applications of Evolutionary Computation. Springer International Publishing, 2016. 203-218.

[9] Sharma, Nirmala, et al. "Modified Artificial Bee Colony Algorithm Based on Disruption Operator." Proceedings of Fifth International Conference on Soft Computing for Problem Solving. Springer Singapore, 2016.

[10] Sharma, Tarun Kumar, and Millie Pant. "Shuffled artificial bee colony algorithm." Soft Computing (2016): 1-20.

[11] Yurtkuran, Alkın, and Erdal Emel. "A discrete artificial bee colony algorithm for single machine scheduling problems." International Journal of Production Research (2016): 1-19.

[12] X. Li and Guangfei Yang. "Artificial bee colony algorithm with memory." Applied Soft Computing (2016).

[13] U. Saif et al. "Hybrid Pareto artificial bee colony algorithm for assembly line balancing with task time variations." International Journal of Computer Integrated Manufacturing (2016): 1-16.

[14] C. Caraveo, Fevrier Valdez, and Oscar Castillo. "Optimization of fuzzy controller design using a new bee colony algorithm with fuzzy dynamic parameter adaptation." Applied Soft Computing 43 (2016): 131-142.

[15] L. Lv et al. "Artificial bee colony algorithm with accelerating convergence." International Journal of Wireless and Mobile Computing 10.1 (2016): 76-82.

[16] S. Kumar, V. K. Sharma, and R. Kumari. "Improved Onlooker Bee Phase in Artificial Bee Colony Algorithm." International Journal of Computer Applications 90.6 (2014): 20-25.

[17] S. Kumar, V. K. Sharma, and R. Kumari, "An Improved Memetic Search in Artificial Bee Colony Algorithm," International Journal of Computer Science and Information Technology (0975 9646), 5(2): 1237-1247, 2014.

[18] Karaboga, Dervis, and Selcuk Aslan. "A new emigrant creation strategy for parallel artificial bee colony algorithm." 2015 9th International Conference on Electrical and Electronics Engineering (ELECO). IEEE, 2015.

[19] Ozturk, Celal, Emrah Hancer, and Dervis Karaboga. "A novel binary artificial bee colony algorithm based on genetic operators." Information Sciences 297 (2015): 154-170.

[20] S. Kumar, V. K. Sharma, and R. Kumari, "Enhanced Local Search in Artificial Bee Colony Algorithm," International Journal of Emerging Technologies in Computational and Applied Sciences, 7(2): 177-184, March 2014.

[21] S. Kumar, V. K. Sharma, and R. Kumari, "Randomized Memetic Artificial Bee Colony Algorithm," International Journal of Emerging Trends and Technologies in Computer Science, 3(1): 52-62, March 2014.

[22] Singhal, Prateek K., et al. "A new strategy based artificial bee colony algorithm for unit commitment problem." Recent Developments in Control, Automation and Power Engineering (RDCAPE), 2015 International Conference on. IEEE, 2015.

[23] S. Kumar, R. Kumari and V. K. Sharma, "A Novel Hybrid Crossover Based Artificial Bee Colony Algorithm for Optimization Problems," International Journal of Computer Applications (0975 8887) 82(8):18-25, November 2013.

[24] Shailesh Pandey and Sandeep Kumar, "Enhanced Artificial Bee Colony Algorithm and It's Application to Travelling Salesman Problem," HCTL Open International Journal of Technology Innovations and Research, Vol 2, March 2013, Pages 137-146, ISSN: 2321-1814, ISBN: 978-1-62776-111-6.

[25] S. Kumar, A. Kumar, V. K. Sharma and H. Sharma, "A Novel Hybrid Memetic Search in Artificial Bee Colony Algorithm," In Proceedings of IC3 2014 – The Seventh IEEE International Conference on Contemporary Computing. 7 -9 Aug 2014. pp 68 - 73. DOI: 10.1109/IC3.2014.6897149.

[26] Kıran, Mustafa Servet, and Oğuz Fındık. "A directed artificial bee colony algorithm." Applied Soft Computing 26 (2015): 454-462.

[27] S. Kumar, V. K. Sharma, and R. Kumari, "Memetic Search in Artificial Bee Colony Algorithm with Fitness based Position Update," In Proceedings of IEEE International Conference On Recent Advances and Innovations in Engineering (ICRAIE-2014). 09-11, May 2014. pp 1-6, DOI: 10.1109/ICRAIE.2014.6909301

[28] Mansouri, P., B. Asady, and N. Gupta. "The Bisection–Artificial Bee Colony algorithm to solve Fixed point problems." Applied Soft Computing 26 (2015): 143-148.

[29] J. C. Bansal, H. Sharma, and S. S. Jadon. "Artificial bee colony algorithm: a survey." International Journal of Advanced Intelligence Paradigms 5.1-2 (2013): 123-159.

8. APPENDIX

Table 1. Test Problem

Test Problem	Objective Function	Search Range	Optimum Value	D	Acceptable Error
Sphere	$f_1(x) = \sum_{i=1}^{n} x_i^2$	[-5.12, 5.12]	f(0) = 0	30	1.0E-05
De-Jong's	$f_2(x) = \sum_{i=1}^{n} i \times (x_i)^4$	[-5.12, 5.12]	f(0) = 0	30	1.0E-05
Griewank	$f_3(x) = \frac{1}{4000}\left(\sum_{i=1}^{D}(x_i^2)\right) - \left(\prod_{i=1}^{D}\cos\left(\frac{x_i}{\sqrt{i}}\right)\right) + 1$	[-600, 600]	*f(0) = 0*	30	*1.0E-05*
Rosenbrock	$f_4(x) = \sum_{i=1}^{i=D-1} 100(x_i^2 - x_{i+1})^2 + (1-x_i)^2$	[-30, 30]	*f(0) = 0*	30	*1.0E-01*
Rastrigin	$f_5(x) = \sum_{i=1}^{D}\left[x_i^2 - 10\cos(2\pi x_i) + 10\right]$	[-5.12, 5.12]	*f(0) = 0*	30	*1.0E-05*
Alpine	$f_6(x) = \sum_{i=1}^{n}\left|x_i \sin x_i + 0.1x_i\right|$	[-10, 10]	*f(0) = 0*	30	1.0E-05
Michalewicz	$f_7(x) = -\sum_{i=1}^{D}\sin x_i (\sin(\frac{ix_i^2}{\pi})^{20})$	[0, π]	fmin = -9.66015	10	1.0E-05
Cosine Mixture	$f_8(x) = \sum_{i=1}^{D} x_i^2 - 0.1(\sum_{i=1}^{D}\cos(5\pi x_i)) + 0.1 \times D$	[-1, 1]	f(0) = -D*0.1	30	1.0E-05
Exponential Problem	$f_9(x) = \exp(-0.5\sum_{i=1}^{n} x_i^2)$	[-1, 1]	f(0) = 1	10	1.0E-05
Zakharov	$f_{10}(x) = \sum_{i=1}^{D} x_i^2 + (\sum_{i=1}^{D}\frac{ix_i}{2})^2 + (\sum_{i=1}^{D}\frac{ix_1}{2})^4$	[-5.12, 5.12]	*f(0) = 0*	30	*1.0E-02*
Brown 3	$f_{11}(x) = \sum_{i=1}^{D-1}(x_i^{2(x_{i+1})^2+1} + x_{i+1}^{2x_i^2+1})$	[-1, 4]	f(0) = 0	30	1.0E-05

Table 1: Test Problems (Cont….)

Test Problem	Objective Function	Search Range	Optimum Value	D	Acceptable Error
Schewel	$f_{12}(x) = \sum_{i=1}^{D}\left|x_i\right| + \prod_{i=1}^{D}\left|x_i\right|$	[-10, 10]	f(0) = 0	30	1.0E-05
Salomon Problem	$f_{13}(x) = 1 - \cos(2\pi p) + 0.1 \times p,$ $where, p = \sqrt{\sum_{i=1}^{D} x_i^2}$	[-100, 100]	*f(0) = 0*	30	*1.0E-01*

Axis Parallel hyper-ellipsoid	$f_{14}(x) = \sum_{i=1}^{D} i x_i^2$	[-5.12, 5.12]	f(0) = 0	30	1.0E-05
Sum of different Powers	$f_{15}(x) = \sum_{i=1}^{D} \lvert x_i \rvert^{i+1}$	[-1, 1]	$f(0) = 0$	30	1.0*E*-05
Step Function	$f_{16}(x) = \sum_{i=1}^{D} (\lfloor x_i + 0.5 \rfloor)^2$	[-100, 100]	*f(-0.5≤x≤0.5) = 0*	30	1.0E-05
Inverted Cosine wave function	$f_{17}(x) = -\sum_{i=1}^{D-1} (\exp(\frac{-(x_i^2 + x_{i+1}^2 + 0.5 x_i x_{i+1}}{8},$ $Where. I = \cos(4\sqrt{x_i^2 + x_{i+1}^2 + 0.5 x_i x_{i+1}})$	[-5. 5]	f(0) = -D+1	10	1.0E-05
Neumaier 3 Problem (NF3)	$f_{18}(x) = \sum_{i=1}^{D} (x_i - 1)^2 - \sum_{i=2}^{D} x_i x_{i-1}$	[-100, 100]	f(0) = -210	10	1.0E-01
Rotated Hypere-ellipsoid	$f_{19}(x) = \sum_{i=1}^{D} \sum_{j=1}^{i} x_j^2$	[-65.536, 65.536]	f(0) = 0	30	1.0E-05
Levy montalvo -1	$f_{20}(x) = \frac{\pi}{D}(10\sin^2(\pi y_1) + \sum_{i=1}^{D-1}(y_i - 1)^2(1 + 10\sin^2(\pi y_{i+1})) + (y_D - 1)^2), Where\ y_i = 1 + \frac{1}{4}(x_i + 1)$	[-10, 10]	$f(-1) = 0$	30	1.0*E*-05
Levy montalvo -2	$f_{21}(x) = 0.1(\sin^2(3\pi x_1) + \sum_{i=1}^{D-1}(x_i - 1)^2(1 + \sin^2(3\pi x_{i+1})) + (x_D - 1)^2(1 + \sin^2(2\pi x_D))$	[-5, 5]	$f(1) = 0$	30	1.0*E*-05
Ellipsoidal	$f_{22}(x) = \sum_{i=1}^{D} (x_i - 1)^2$	[-D, D]	f(1, 2,..,D) = 0	30	1.0*E*-05
Beale function	$f_{23}(x) = (1.5 - x_1(1 - x_2))^2 + (2.25 - x_1(1 - x_2^2))^2 + (2.625 - x_1(1 - x_2^3))^2$	[-4.5, 4.5]	$f(3. 0.5) = 0$	2	1.0*E*-05

Table 1: Test Problems (Cont…)

Test Problem	Objective Function	Search Range	Optimum Value	D	Acceptable Error
Colville function	$f_{24}(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2 + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2] + 19.8(x_2 - 1)(x_4 - 1)$	[-10, 10]	$f(1) = 0$	4	1.0*E*-05
Braninss Function	$f_{25}(x) = a(x_2 - b x_1^2 + c x_1 - d)^2 + e(1 - f)\cos x_1 + e$	$x_1 \in [-5,10], x_2 \in [0,15]$	*f(-π, 12.275) = 0.3979*	2	1.0*E*-05
Kowalik function	$f_{26}(x) = \sum_{i=1}^{11} (a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4})^2$	[-5, 5]	*f*(0.1928, 0.1908, 0.1231, 0.1357) = 3.07*E*-04	4	1.0*E*-05
2D Tripod Function	$f_{27}(x) = p(x_2)(1 + p(x_1)) + (x_1 + 50 p(x_2)(1 - 2p(x_1))) + (x_2 + 50(1 - 2p(x_2)))$	[-100, 100]	f(0, -50)=0	2	1.0E-04

Name	Function	Range	Optimum	D	Accuracy
Shifted Rosenbrock	$f_{28}(x) = \sum_{i=1}^{D-1}(100(z_i^2 - z_{i+1})^2 + (z_i - 1)^2 + f_{bias}$, $z = x - o + 1, x = [x_1, x_2, ...x_D], o = [o_1, o_2, ..o_D]$	[-100, 100]	f(o)=fbias=390	10	1.0E-01
Shifted Sphere	$f_{29}(x) = \sum_{i=1}^{D} z_i^2 + f_{bias}, z = (x - o) x = [x_1, x_2, ...x_D]$, $o = [o_1, o_2, ...o_D]$	[-100, 100]	f(o)=fbias=-450	10	1.0E-05
Shifted Rastrigin	$f_{30}(x) = \sum_{i=1}^{D}(z_i^2 - 10\cos(2\pi z_i) + 10) + f_{bias}$, $z = (x - o), x = [x_1, x_2, ...x_D], o = [o_1, o_2, ...o_D]$	[-5, 5]	*f(o)=f_{bias}=-330*	10	1.0*E*-02
Shifted Schwefel	$f_{31}(x) = \sum_{i=1}^{D}(\sum_{j=1}^{i} z_j)^2 + f_{bias}, z = (x - o)$, $x = [x_1, x_2, ...x_D], o = [o_1, o_2, ...o_D]$	[-100, 100]	*f(o)=f_{bias}=-450*	10	1.0E-05
Shifted Griewank	$f_{32}(x) = \sum_{i=1}^{D} \frac{z_i^2}{4000} - \prod_{i=1}^{D} \cos(\frac{z_i}{\sqrt{i}}) + 1 + f_{bias}$, $z = (x - o), x = [x_1, x_2, ...x_D], o = [o_1, o_2, ...o_D]$	[-600, 600]	*f(o)=f_{bias}=-180*	10	1.0E-05
Shifted Ackley	$f_{33}(x) = -20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} z_i^2}) - $ $\exp(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi z_i)) + 20 + e + f_{bias}$, $z = (x - o), x = [x_1, x_2, ...x_D], o = [o_1, o_2, ...o_D]$	[-32,32]	*f(o)=f_{bias}=-140*	10	1.0*E*-05
Goldstein-Price	$f_{34}(x) = (1 + (x_1 + x_2 + 1)^2 \times (19 - 14x_1 + 3x_1^2 - 14x_2$ $+6x_1x_2 + 3x_2^2)) \times (30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1$ $+12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2))$	[-2, 2]	*f(0, -1)=3*	2	1.0*E*-14
Six-hump camel back	$f_{35}(x) = (4 - 2.1x_1^2 + \frac{1}{3}x_1^4)x_1^2 + x_1x_2 + (-4 + 4x_2^2)x_2^2$	[-5, 5]	f(-0.0898, 0.7126) = -1.0316	2	1.0E-05

Table 2: Comparison of success rate with different population size

Problem No	NP10	NP50	NP100	NP200	NP300	NP500	NP1000
f_1	100	100	100	94	28	0	0
f_2	100	100	100	100	100	0	0
f_3	48	68	22	1	0	0	0
f_4	0	11	0	0	0	0	0
f_5	0	2	2	0	0	0	0
f_6	0	0	0	0	0	0	0
f_7	0	0	0	0	0	0	0
f_8	0	0	0	0	0	0	0
f_9	0	0	0	0	0	0	17
f_{10}	0	0	0	0	0	0	0
f_{11}	100	100	100	99	71	0	0
f_{12}	100	100	0	0	0	0	0
f_{13}	0	1	0	0	0	0	0
f_{14}	100	100	100	99	9	0	0
f_{15}	100	100	100	100	51	1	0
f_{16}	91	100	100	99	93	0	0
f_{17}	0	0	0	0	0	0	0
f_{18}	0	0	0	0	0	0	0
f_{19}	100	100	100	24	0	0	0

f_{20}	100	100	100	57	1	0	0
f_{21}	100	100	100	39	0	0	0
f_{22}	100	100	100	90	0	0	0
f_{23}	22	100	100	100	82	88	2
f_{24}	0	0	5	4	7	100	0
f_{25}	92	84	89	92	83	87	5
f_{26}	2	23	33	19	12	12	0
f_{27}	32	99	100	100	100	100	2
f_{28}	2	7	5	3	1	0	0
f_{29}	0	0	0	0	0	0	0
f_{30}	0	0	0	0	0	0	0
f_{31}	0	0	0	0	0	0	0
f_{32}	0	0	0	0	0	0	0
f_{33}	0	0	0	0	0	0	0
f_{34}	0	15	88	100	100	0	0
f_{35}	0	0	0	0	0	0	0

Table 3: Comparison of AFE with different population size

Problem	NP10	NP50	NP100	NP200	NP300	NP500	NP1000
f_1	7474.3	23218	41108	77502	98526	100000	100000
f_2	4233	10977.5	19843	34990	51096	100000	100000
f_3	74969.69	76412	96375	99996	100200	100000	100000
f_4	100004.8	98135.5	100000	100000	100200	100000	100000
f_5	100005.5	99490	99897	100000	100200	100000	100000
f_6	100005.5	100000	100000	100000	100200	100000	100000
f_7	100004.2	100011.2	100013.7	100000.2	100200	100000	100000
f_8	100003.4	100041.9	100007.1	100000.1	100200	100000	100000
f_9	100005	100007.9	100011.3	100000	100200	100000	88240
f_{10}	100000.1	100000	100000	100000	100200	100000	100000
f_{11}	8264.5	20586	36233	66536	93615	100000	100000
f_{12}	17138.4	64664.5	100000	100000	100200	100000	100000
f_{13}	100006	99967.23	100000	100000	100200	100000	100000
f_{14}	8592.31	23131	41354	77006	99894	100000	100000
f_{15}	7060.68	22335	38080	68526	91464	99875	100000
f_{16}	42413.24	18030.84	28836.02	51718	72894	100000	100000
f_{17}	100004.4	100009.7	100067.4	100006.9	100201	100000	100000
f_{18}	100004	100038	100016.5	100001.8	100200.2	100000	100000
f_{19}	11808.51	30859.5	55503	99290	100200	100000	100000
f_{20}	14396.56	30117.5	56183	93652	100155	100000	100000
f_{21}	11175.91	29919	58517	95480	100200	100000	100000
f_{22}	10640.7	27362.5	48612	90820	100200	100000	100000
f_{23}	90373.98	16245.47	32265.75	58013.33	65515.32	64149.79	98020
f_{24}	100005.1	100031.6	98888.62	98843.03	97026.09	83466	100000
f_{25}	14133.09	17737.41	13861.76	12666.13	22611.02	26596.98	97630
f_{26}	99304.7	85643.17	89380.96	89358.05	94542	96290	100000

f_{27}	83013.72	8412.64	7470.02	13020	18852	45215.18	98020
f_{28}	98758.38	96364.35	97196.14	99416	99984	100000	100000
f_{29}	100004.6	100015.8	100057.6	100004.1	100200.3	100000	100000
f_{30}	100005	100017.5	100067.3	100036.3	100201.1	100000	100000
f_{31}	100004.5	100031.4	100015.3	100037.3	100200.7	100000	100000
f_{32}	100004.8	100010.5	100072.1	100007.4	100200.8	100000	100000
f_{33}	100004.6	100001	100000	100000	100200	100000	100000
f_{34}	100005.3	89760.06	35235.83	44356.34	63666.3	100051.6	100000
f_{35}	100004.5	100028.4	100043	100183.8	100155.5	100002.9	100000