Int. J. Advance. Soft Comput. Appl., Vol. 5, No. 1, March 2013
ISSN 2074-8523; Copyright © SCRG Publication, 2013

A Comparison Study of Biogeography based Optimization for Optimization Problems

N. F. Hordri[1], S. S. Yuhaniz[1] and Dewi Nasien[1]

[1]*Soft Computing Research Group*, Faculty of Computing
Universiti Teknologi Malaysia,
Skudai, Johor Bahru
email: missfarna@gmail.com; sophia@utm.my

Abstract

Most optimization problems have constraints. The solutions of the problem are obtained from the final results of the search space that have satisfied the given constraints. In such cases, heuristic algorithms are capable to find the estimated solutions, but sometimes they have some limitations. This paper investigates the performance of three heuristic optimization methods: Biogeography Based Optimization (BBO), Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) for solving the optimization problems. We compare these algorithms in terms of their convergence time and their performance in avoiding local minima on fourteen benchmark functions. These benchmark functions are used to test optimization procedures for multidimensional and continuous optimization task. The findings reveal that BBO is a promising optimization tool that can deal with the complex optimization problems.

Keywords: *Benchmark Functions, Biogeography Based Optimization, Genetic Algorithm, Heuristic Algorithm, Optimization Problem, Particle Swarm Optimization.*

1 Introduction

Optimization problem is a computational problem in which the object is to find the potential solutions. The solutions will be found in the feasible region of the minimum or maximum value of the objective function [1]. Optimization problems are common in many disciplines and various fields [2]. We have to find the optimal solution to some objectives. Unfortunately, we are not able to solve the problem in one step. Therefore, several processes that will guide us through the problem solving have to follow. Fig. 1 shows the planning process for optimization problems.

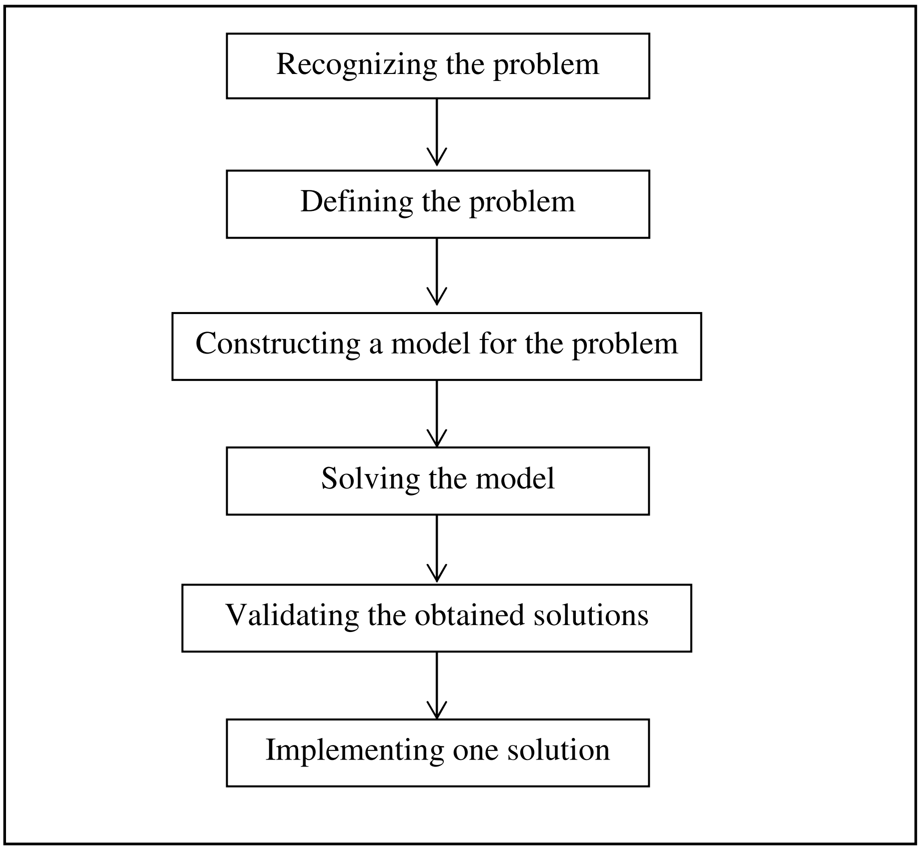

Fig. 1: Planning Process

There are two types of optimization problems: combinatorial optimization problem and continuous optimization problem. Combinatorial optimization problems focus on the limited resources to meet the desired objectives. The decision variables can yield the values from bounded, discrete sets and additional constraints on basic resources. Examples for basic resources are labour, supplies, or capital, restrict the possible alternatives that are considered feasible. There are many possible alternatives to consider, while a goal determines which of these alternatives is the best. However, the situation is different for continuous optimization problems. The continuous optimization problem is more focussed on the optimal setting of parameters or continuous decision variables. There are no limited numbers of alternatives existing but optimal values for continuous variables have to be determined.

Many problem-solving processes tend to be heuristic throughout the human history, for example the heuristic for scientific method for optimization [3]. Heuristic algorithms can suggest some valuation to the solution of optimization problems [4]. As known, the problem is to find the optimal of all possible solutions which is to minimize or maximize the objective function. The objective function is a function to calculate the quality of the generated solution. There have two types of heuristic method. The heuristic method that can optimally solve small problems is cutting planes and branch and bound [5]. While the heuristic method such as 2-opt [6], 3-opt, Markov chain [7], simulated annealing [8] and *Tabu* search [9] are good for solving the large problems.

2 Heuristic Techniques

There are varieties of existing computational tasks and the numbers of algorithms that have been developed to solve the tasks. The heuristic algorithm provides a solution or nearly right solution not for all instances of the problem. The group of the heuristic algorithms have an overflowing spectrum of methods based on traditional techniques as well as specific ones.

Population-based optimization algorithms are inspired by the nature based optimization algorithms. For designing and inventing new systems and algorithms in science and technology, researchers are inspired by interesting and valuable sources of the creatures and natural systems. Evolutionary Algorithms [10] and Swarm Intelligence [11] are among the problem solving techniques inspired from observing the nature.

Evolutionary Algorithms (EAs) succeed in undertaking the premature convergence by considering a number of solutions simultaneously. The algorithms exploit the ideas of biological evolution for examples, reproduction, mutation and recombination for searching the optimal solution. The principle of survival on a set of potential solutions to produce the ongoing approximations to the optimum is applied. EA conducts a search using a population of solutions. Competitive selections among all the solutions in the population are involved for each iteration. This will result in survival of the fittest and detection of the poor solutions from the population. The swapping parts of a solution with another one is known as recombination. Recombination performs and produces the new solutions that might be better than the previous solution, and later the solution will be mutated. Recombination and mutation are used to grow the population towards regions of the space which good solutions may reside. One of the famous methods in the evolutionary algorithms is GA [12]. BBO has been developed since 2008 [13]. The migration strategy of BBO is similar to the global recombination approach of the breeder GA [14] and evolutionary strategies [15] where the parents can contribute to a new single offspring. Migration of BBO is used to change the existing solutions because BBO is an adaptive process which will modify the existing islands [13].

Swarm Intelligence (SI) is an artificial intelligence technique which based on the study of collective behaviour in decentralized, self-organized, systems. SI consists typically of a population of simple agents or *boids* interacting locally with one another and with their environment. The main advantage of this technique is that the SI is very impressively resistant to the local optima problem. Particle Swarm Optimization (PSO) is the most successful technique for this approach. PSO deals with problems in which a best solution can be denoted as a point or surface in an n-dimensional space. [4] PSO solutions are more likely to clump together in similar groups. Each solution is PSO represents as a point in space, and represents the change over time of each solution as a velocity vector. PSO do not change directly their solution but their velocity is actually changed. The velocity is acknowledged as a communication channel between particles [16] [17].

2.1 Genetic Algorithm

Genetic Algorithm is a part of evolutionary computing, which is rapidly growing area of artificial intelligence. GA is an iterative algorithm that is parallel and global. Based on the theory of GA, the solution of each problem is considered to be an individual or a chromosome [18]. There are three processes in GA which are selection, crossover and mutation. In standard GA, each individual represents the chromosome. There is a function to determine the fitness of the individual and another function to selects individuals from the population to reproduce. Several terms used in GA such as *Fitness* (measure how well the chromosome fits the search space or solves the problem), *Selection* (process for choosing a pair of organisms to reproduce while crossover), *Crossover* (process of exchanging the genes between the two individuals that are reproducing) and *Mutation* (process of randomly altering the chromosomes). Fig. 2 shows the crossover process in GA.

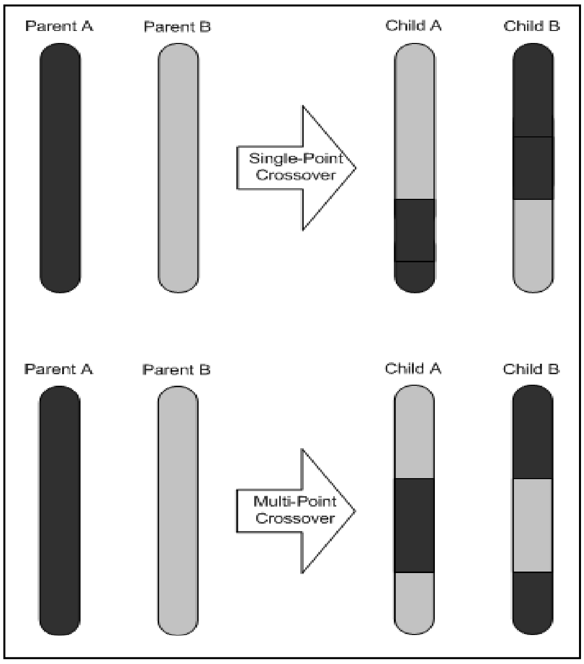

Fig. 2: GA Crossover [31]

2.2 Particle Swarm Optimization

The original PSO algorithm is discovered through simplified social model simulation [19]. PSO is a computational method that optimizes a problem iteratively and it is a simple concept adapted from nature decentralized and self-organized system. PSO is a population-based algorithm in which individual particles work together to solve a given problem. The particle is initialized by assigning random positions and velocities and potential solutions are then flown through the hyperspace. The particles learn over time in response to their own experience and the experience of the other particles in their group [20]. The particle will keep track of its best fitness position and this value is called personal best. The overall best value obtained by any particle in the population is called global best. The basic procedure for PSO is shown in Fig. 3.

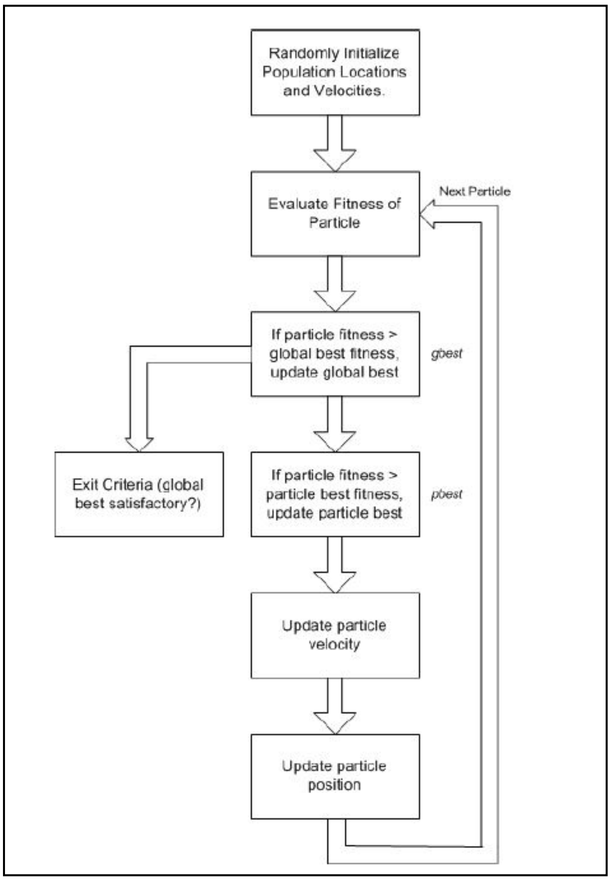

Fig. 3: Basic PSO Procedure [31]

2.3 Biogeography Based Optimization

BBO is the new approach to problem solving, and shares some features with other biology-based algorithms; just as GA and PSO, BBO has a way of sharing information between solutions. However, GA solution only lasts until the end of each generation. PSO and BBO solutions can survive forever although the characteristics change as the optimization process progresses. According to the theory of BBO, a good solution is related to an island with a high, High Suitability

Index (HSI), and a poor solution signifies an island with a low HSI. High HSI solutions resist change more effectively than low HSI solutions.

BBO is the study of migration, speciation and extinction of species. Mathematical models of BBO describes how a species migrates from one island (habitat) to another, how new species arise and how species become extinct. The BBO optimization algorithm is the first presented as an example of how a natural process can be generalized to solve optimization problems [13].

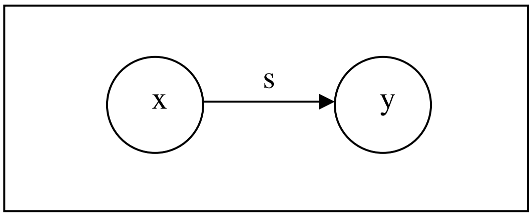

Fig. 4: Migration of the Solution of the Species (s)

From Fig. 4, we can say that an individual (solution) of the species has emigrated from *x* and immigrated to *y*. Normally, when *s* has immigrated to *y*, *s* will probabilistically share information from *x* based on fitness values with *y*. Each individual has a set of features. Just as species migrate back and forth between islands, BBO operates by sharing information between individuals in a species of candidate solutions.

Immigration is the replacement of an old solution feature in an individual with a new solution feature from another individual. The immigrating solution feature replaces a feature in the immigrating individual. While for the emigration is the sharing of a solution feature in BBO from one individual to another. The immigrating solution feature remains in the emigrating individual. Each habitat has its own features such as rainfall, diversity of vegetation, diversity of topographic features, land area and temperature. The characteristics are denoted as suitability index variable of a habitat.

In natural biogeography, a very habitable island is unlikely to accept immigrants from a less habitable island [21]. This is due to two reasons:

1. The very habitable island is already saturated with species and does not have many additional resources to support immigrants; and
2. The inhabitable island does not have very many species to begin with and does not have many potential emigrants.

Although more complicated and life-like migration curves can give better optimization results, we use linear migration curves like those in Fig. 5 for the sake of simplicity to illustrate two individuals in BBO.

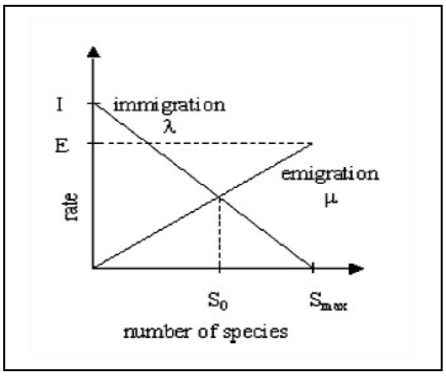

Fig. 5: Species model of a single habitat based on [9]

S_{max} represents a good solution and S_0 represents a poor solutions. The immigration rate for S_{max} will therefore is higher than the immigration rate for S_0. Note that if a good solution is obtained in the population, then there may be a high probability that the population will converge towards that solution, resulting in premature convergence.

2.4 Comparisons of the Methods

GA, PSO and BBO perform information sharing between the solutions i.e. genes for GA, particles for PSO and habitats for BBO. Three of them are inspired by nature and have been by themselves to be effective solutions to optimization problems. Table 1 provides the comparison between BBO, PSO and GA.

Table 1: Comparison between BBO, PSO and GA

Description	BBO [13]	PSO [22]	GA [12]
Develop	2008	1995	Early 1970s
Introduced by	Dan Simon	J. Kennedy and R. Eberhart	John Henry Holland
Optimization Technique (OT)	Bio-inspired OT that used the idea of probabilistically sharing features between solutions based on the solutions' fitness values.	Robust Stochastic OT based on the movement and cooperation.	Intelligent OT that relies on the parallelism found in nature.
At the end of each generation	BBO solutions survive forever.	PSO solutions survive forever.	GA solutions die out.
Grouped	Does not form the grouping of the habitat having identical	Are grouped into their similar characteristic.	Not necessarily have any built-in tendency to cluster.

	characteristic.		
Mutation	Combination of two ideas of global recombination and uniform crossover which are use the entire of the population as potential contributors to the next generation and use fitness-based selection for each solution feature in each offspring.	PSO does not have genetic operators such as crossover or mutation. Particle update themselves with the internal velocity, they also have a memory that is important to the algorithm.	All GAs require some form of recombination, as this allows the creation of new solutions that have, by virtue of their parent success, a higher probability of exhibiting a good performance.

3 Standard Benchmark Functions

In this study, we used fourteen benchmark functions to calculate the quality of the optimization algorithm (Table 2). In the experiment of test function, the three algorithms will be compared based on their convergence time and number of generations.

Table 2: Benchmark Functions [23] [24] [25]

Benchmark Function	Definition	Formula
Ackley	The Ackley Problem is a minimization problem. Originally this problem was defined for two dimensions, but the problem has been generalized to N dimensions.	$f_{ACK} = 20 + e - 20\exp(-20\sqrt{\frac{1}{p}\sum_{i-1}^{p} x_i^2}$
Fletcher	The Fletcher is to finds the solution to the secant equation that is closest to the current estimate and satisfies the curvature condition. Is generalizing the secant method to a	$f_{Fle} = \sum_{i=1}^{p}(A_i - B_i)^2$

	multidimensional problem.	
Griewank	The Griewank function is a function widely used to test the convergence of optimization functions.	$f_{Gri} = 1 + \sum_{i=1}^{p} \frac{x_i^2}{4000} - \prod_{i=1}^{p} cos\left(\frac{x_i}{\sqrt{i}}\right)$
Quartic	Quartic function is a polynomial of even degree; it has the same limit when the argument goes to positive or negative infinity.	$f_{Qua}\ (x) = ax^4 + \ bx^3 + cx^2 + dx + e$
Rastrigin	Rastrigin function is a non-convex function used as a performance test problem for optimization algorithms. It is typical example of non-linear multimodal function	$f_{Ras}(x) = 10p + \sum_{i-1}^{p} \left(x_i^2 - 10cos(2\pi x_i)\right)$
Rosenbrock	Rosenbrock function is a non-convex function used as a performance test problem for optimization algorithms.	$f_{Ros}(x) = \sum_{i=1}^{p-1} [100(x_{i+1} - \ x_i^2)^2 + \ (x_i - 1)^2]$
Step	Step function is using only real number and it can write as a finite linear combination of indicator functions of intervals.	$f_{Ste}(x) = \sum_{i-0}^{n} \propto_i \chi_{A_i}(x)$
Penalty	Penalty function is no infeasible soluble solution considered and it will applied near to feasibility boundary	$f_P(x) = \ f(x) + \sum_{i=1}^{n} C_i \delta_i$
Schwefel	Schwefel function is deceptive in that the global minimum is geometrically distant over the parameter space, from the next best local minima	$f_{Sch}(x) = \sum_{i=1}^{n} \left[-x_i sin\left(\sqrt{\lvert x_i \rvert}\right)\right]$
Sphere	Sphere function is simplest test benchmark	$f_{Sph}(x) = \sum_{i=1}^{n} i \cdot x_i^2$

3.1 Convergence Time and Local Minima

BBO, PSO and GA algorithms will be tested on the benchmark functions to see which optimization algorithm can perform better solving the optimization problems. The experiment is conducted accordingly and the algorithms are compared to each other based on the convergence time and local minima.

In evolutionary computing, convergence means a modeling of tendency for genetic characteristic of populations to stabilize over time. The convergence time is closely connected to the concept of speed that the optimizations can travel over a specific amount of time.

Normally, standard optimization algorithms that use the gradient to find a minimum may become trapped in a local minimum. This will result in less optimal restored image. The presence of multiple local minima during optimization will drag the time become more longer to finish the process of optimization. The approach to get around this difficulty is to run the algorithm many times with different initial guesses which may produce different local minima.

4 Experimental Study

This section is divided into four phases: (i) the early planning phase: (ii) the problem analysis phase; (iii) comparison BBO, PSO and GA; and (iv) result analysis. Fig. 6 shows all the phases involved in the development of this study.

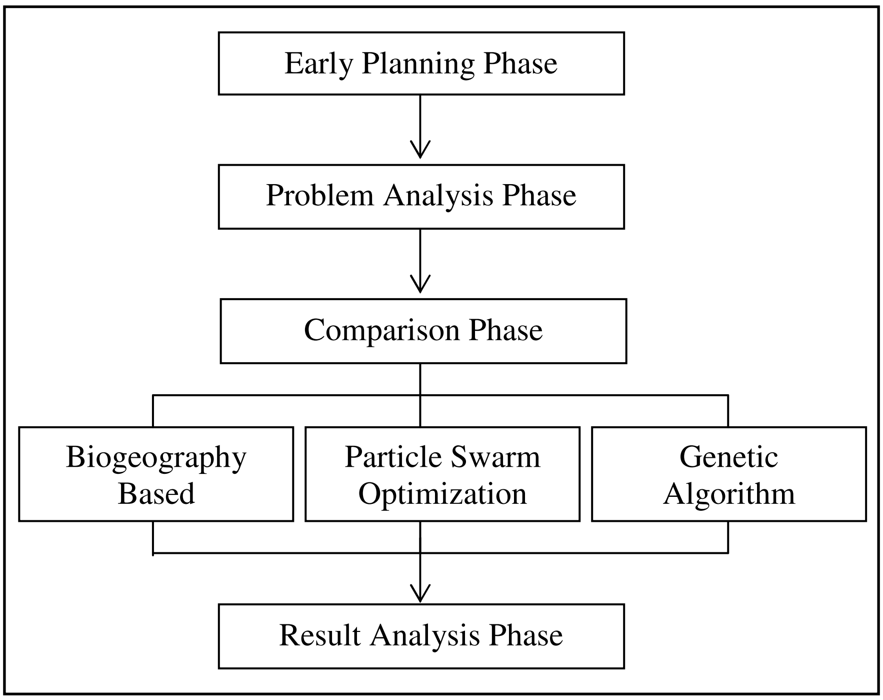

Fig. 6: The Methodology of the Study's Development

The purpose of this study is to investigate the best optimization method of BBO, PSO and GA. The next phase is the problem analysis. Two main tasks are carried out in this phase. The first task is to find out the other optimization method to compare with BBO. The second task is to collect all the related information on optimization problem, BBO and other optimization methods. The third phase is the comparison phase. In this phase, BBO, PSO and GA are compared using fourteen benchmark functions. The comparison is based on local minimum and convergence time. The final phase of the study is the result analysis in which the results of the experiment will be analysed and discussions will be drawn either BBO can be a good tool for optimization problems.

4.1 Parameter Setting for Experiments

Before conducting the experiment, we have to set the population size, the elitism parameter and the number of generation runs as in the Table 3. The settings are the same for each experiment.

Table 3: Initial Setting for Experiments

Criterions	Settings
Population size	50
Elitism parameter	2
The number of generation	50

Fourteen experiments have been conducted for the research. The fourteen benchmark functions are Ackley, Fletcher, Griewank, Penalty 1, Penalty 2, Quartic, Rastrigin, Rosenbrock, Schwefel, Schwefel 2, Schwefel 3, Schwefel 4, Sphere and Step.

For the fourteen experiments, we did not change the settings because different tuning the parameter values might cause different performance. Based on [13], we will be able to have different conclusions if the generation limit has been changed. These experiments are basically to prove that BBO can be a better optimization algorithm compared to PSO and GA.

The results will then be analysed based on the convergence time and local minima. If the algorithm has always been trapped with the local minima, it will slow the convergence time and therefore cannot produce a better result for optimization. The lesser value of best normalized optimization is the best optimization algorithm.

5 Results of the Experiments

The minimum cost at the end of the generation for fourteen experiments are shown in Fig. 7 below. It can be seen that BBO algorithm performs better compared to GA and PSO in all the fourteen experiments. The minimum cost that has been tested for fourteen benchmark functions are provided by the literatures. Based on [23] [24] [25], half of the benchmark functions are multimodal. Multimodal means they have multiple local minima. While, some of them are nonseparable and cannot be written as a sum of functions of individual variables. Some of them are also regular which become logical at each point of their area.

As we can see in Fig. 7, the Quartic function has very low minimum cost for BBO, PSO and GA among the fourteen benchmark functions. Quartic is not multimodal, separable and regular [13]. Hence, BBO, PSO and GA can easily be optimized for each variable in turn. Similar to Quartic function, the Sphere function is not multimodal, separable and regular. Sphere also gives the minimum cost for the compared algorithms. Sphere has been used in the development of the theory of evolutionary strategies and in the evaluation of GA. This function is a simple and strongly convex [26].

However, the Penalty2 has a very high minimum cost compared with other benchmark functions. Penalty2 is z multimodal, nonseparable and regular [25]. Hence, the Penalty2 has many local minima and make it as the most complex case in search process due to its randomness in the search space. The function of Fletcher [27] is highly multimodal. The function is non-symmetrical and their local optima are randomly distributed. Therefore, the objective function has no implicit symmetry advantages that might simplify optimization for BBO, PSO and GA. Although Ackley [28] also is multimodal, nonseparable and regular, Ackley still can have lowest minimum cost among the rest. This is because Ackley has an exponential term that covers its surface with numerous local minima.

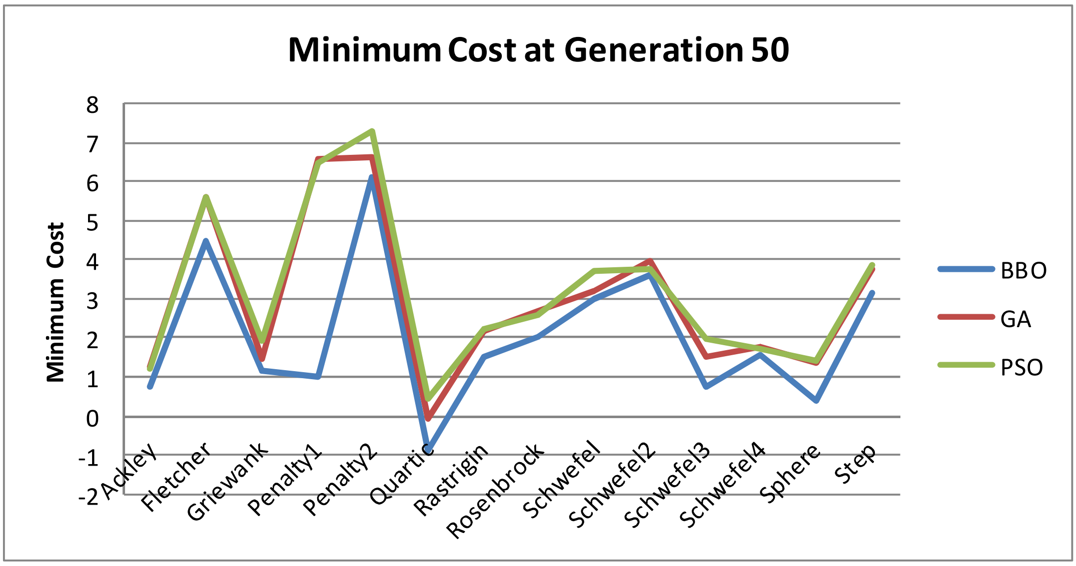

Fig. 7: Minimum Cost at Generation 50 for Each Benchmark Functions

Table 4: Number of Generations to Reach the Minimum Cost

Benchmark Functions	BBO	PSO	GA
Ackley	48	10	34
Fletcher	50	23	49
Griewank	50	11	47
Penalty1	50	43	48
Penalty 2	50	38	50
Quartic	50	20	46
Rastrigin	48	40	35
Rosenbrock	50	33	48
Schwefel	49	22	41
Schwefel 2	50	2	50
Schwefel 3	47	1	44
Schwefel 4	37	9	17
Sphere	50	45	49
Step	50	43	50

The stopping criteria for each experiment is at generation 50. This means that if the method can reach generation of 50 with the lowest minimum cost, the method can be claimed as a good optimization method. However, if the method cannot reach generation 50, most probably it has been trapped to local minima. From Table 4, we see that BBO performed the best on the nine benchmark functions with the lowest minimum cost at generation 50. GA was the second most effective on performing the best on three benchmark function which is Penalty2, Schwefel2 and Step. Though, for the three benchmark function, BBO still can manage to have the lowest minimum cost compared to GA (refer to Fig. 7). On the other hand, PSO and GA sometimes are trapping in a local minimum before reaching to the 50 generations. The local minima can make the convergence becomes slower and produce poor optimization results. Table 4 shows that PSO find their minimum cost at early generation but then been trapped with local minima till generation 50. Similar to PSO, GA is also trapped in a local minima, however, GA is a bit late from trapping in a local minima compared to PSO. While the BBO is still running to find the minimum cost till the end of the generation even though at one point BBO is also been trapped in a local minima but only for a while. For Schwefel4 function, the BBO, PSO and GA are not performed very well. This is because the function is multimodal and causes these optimization algorithms easily being trapped in a local minimum. The search process is even harder because the Schwefel4 is also separable function. The process must be able to avoid the local minima as far as possible to the global optimum. running to find the minimum cost till the end of the generation eventhough at one point BBO also is been trapped with local minima but only for a while. For Schwefel4 function, the BBO, PSO and GA are not performed very well. This is because the function is multimodal and make all three optimization algorithms are easily been trapped to local minima. The search process is even harder because the Schwefel4 is also separable function. The process must be able to avoid the local minima as far as possible to the global optimum.

6 Conclusion

The heuristic algorithms have been applied by many researches to solve the optimization problems. Thus, the advantages for these algorithms are [32]:

1. They are robust and can adapt solutions with changing conditions and environment.
2. They can be applied in solving complex multimodal problems.
3. They may incorporate mechanisms to avoid getting trapped in local optima.
4. They are not problem-scientific algorithm.
5. These algorithms are able to find promising regions in a reasonable time due to exploration and exploitation ability.
6. They can be easily employed in parallel processing.

As seen in this study, the complexity of the real-world optimization problem may be inspired from the findings of the benchmark functions. The BBO can give a new pattern or a new challenge among the population-based algorithms for optimization problems. The BBO does not group with the habitat that having identical characteristics, where PSO are grouped into their similar characteristics. The usage of fitness-based selection for each solution feature in each offspring shows that BBO can be applied to many high-dimension problems with multiple local optima.

ACKNOWLEDGEMENTS

This work is supported by Universiti Teknologi Malaysia (UTM) under Research University Grant (VOT Q.J130000.2628.08J38). The authors would like to thank Soft Computing Research Group (SCRG) for their support and incisive comments in making this study a success.

References

[1] Black, P.E., *Algorithms and theory of computation handbook*, 1999, CRC Press LLC.

[2] F. Rothlauf, *Design of Modern Heuristics*, Natural Computing Series, DOI 10.1007/978-3-540-72962-4 2, © Springer-Verlag Berlin Heidelberg 2011.

[3] Yang, X.-S., *Nature-inspired metaheuristic algorithms*. 2010: Luniver Press.

[4] Kokash, N., *An introduction to heuristic algorithms*. Department of Informatics and Telecommunications, 2005.

[5] Padberg, M. and G. Rinaldi, *A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems*. SIAM review, 1991. 33(1): p. 60-100.

[6] Lin, S. and B.W. Kernighan*, An effective heuristic algorithm for the traveling-salesman problem*. Operations research, 1973. 21(2): p. 498-516.

[7] Martin, A.D. and K.M. Quinn, *Dynamic ideal point estimation via Markov chain Monte Carlo for the US Supreme Court*, 1953–1999. Political Analysis, 2002. 10(2): p. 134-153.

[8] Kirkpatrick, S., D.G. Jr., and M.P. Vecchi, *Optimization by simmulated annealing*. science, 1983. 220(4598): p. 671-680.

[9] Glover, F. and M. Laguna*, Tabu search*. Vol. 1. 1998: Springer.

[10] Eiben, A.E. and J.E. Smith*, Introduction*. 2003: Springer.

[11] R. Eberhart, Y.S., and J. Kennedy, Swarm intelligence. Morgan Kaufmann. 2001.

[12] Holland, J.H., *Adaption in natural and artificial systems.* 1975.

[13] Simon, D., *Biogeography-based optimization.* Evolutionary Computation, IEEE Transactions on, 2008. **12**(6): p. 702-713.

[14] Muhlenbein, H. and D. Schlierkamp-Voosen, *Predictive models for the breeder genetic algorithm i. continuous parameter optimization*. Evolutionary computation, 1993. 1(1): p. 25-49.

[15] Back, T., D.B. Fogel, and Z. Michalewicz, *Handbook of evolutionary computation.* 1997: IOP Publishing Ltd.

[16] Parsopoulos, K.E. and M.N. Vrahatis, *Recent approaches to global optimization problems through particle swarm optimization*. Natural computing, 2002. 1(2-3): p. 235-306.

[17] Clerc, M., *Particle Swarm Optimization*. Vol. 93. 2010: John Wiley & Sons.

[18] Zhang, Q. and C. Wang. *Using genetic algorithm to optimize artificial neural network: A case study on earthquake prediction*. in *Genetic and Evolutionary Computing, 2008. WGEC'08. Second International Conference on*. 2008. IEEE.

[19] Shi, Y. (2004). *Particle Swarm Optimization. IEEE Neural Network Society:* 8-13.

[20] Ferguson, D. (2004). *Particle Swarm.* University of Victoria, Canada.

[21] MacArthur, R.H. and E.O. Wilson, *The theory of island biogeography*.

[22] del Valle, Y., et al., *Particle swarm optimization: basic concepts, variants and applications in power systems.* Evolutionary Computation, IEEE Transactions on, 2008. 12(2): p. 171-195.

[23] Domingo. *Benchmark Problems*. 2005 2005-07-11 [cited 2013 November 15]; Available from: http://www.cs.cmu.edu/afs/cs/project/jair/pub/volume24/ortizboyer05a-html/node6.html

[24] Molga, M. and C. Smutnicki, *Test functions for optimization needs*. Test functions for optimization needs, 2005.

[25] Smith, A.E. and D.W. Coit, *Penalty functions*. Handbook of Evolutionary Computation, 1995. 97(1): p. C5.

[26] Kenneth, A., De Jong. *An analysis of the behavior of a class of genetic adaptive systems*, 1975, PhD thesis, Dept. of Computer and Comm. Sciences, Univ. of Michigan, Ann Arbor, MI, 1975. Univ. Microfilms.

[27] Fletcher, R. and M.J. Powell, *A rapidly convergent descent method for minimization.* The Computer Journal, 1963. 6(2): p. 163-168.

[28] Ackley, D.H., *An empirical study of bit vector function optimization*. Genetic algorithms and simulated annealing, 1987. 1: p. 170-204.

[29] Goldberg, D.E., *Genetic algorithms in search, optimization, and machine learning.* 1989.

[30] Gustafson, S. and E.K. Burke, *The speciating island model: An alternative parallel evolutionary algorithm.* Journal of Parallel and Distributed Computing, 2006. **66**(8): p. 1025-1036.

[31] Hamed, A. and H. Nuzly, *Particle swarm optimization for neural network learning enhancement*, 2006, Universiti Teknologi Malaysia, Faculty of Computer Science and Information System.

[32] Beheshti, Z. and S.M.H. Shamsuddin, *A Review of Population-based Meta-Heuristic Algorithm*. Int. J. Advance. Soft Comput. Appl, 2013. 5(1).