The Bat Algorithm with Dynamic Niche Radius for Multimodal Optimization

Takuya Iwase

The University of Electro-Communications  
Tokyo, Japan

[tanu\_iwa@cas.lab.uec.ac.jp](mailto:tanu_iwa@cas.lab.uec.ac.jp)

Hiroyuki Sato

The University of Electro-Communications  
Tokyo, Japan

[h.sato@uec.ac.jp](mailto:h.sato@uec.ac.jp)

Ryo Takano

The University of Electro-Communications  
Tokyo, Japan

[takano@cas.lab.uec.ac.jp](mailto:takano@cas.lab.uec.ac.jp)

Keiki Takadama

The University of Electro-Communications  
Tokyo, Japan

[keiki@inf.uec.ac.jp](mailto:keiki@inf.uec.ac.jp)

Fumito Uwano

The University of Electro-Communications  
Tokyo, Japan

[uwano@cas.lab.uec.ac.jp](mailto:uwano@cas.lab.uec.ac.jp)

**ABSTRACT**

In this paper, we proposed Bat Algorithm extending with Dynamic Niche Radius (DNRBA) which enables solutions to locate multiple local and global optima for solving multimodal optimization problems. This proposed algorithm is designed Bat Algorithm (BA) dealing with the exploration and the exploitation search with Niche Radius which is calculated by the fitness landscape and the number of local and global optima to avoid converging solutions at the same optimum. Although the Niche Radius is an effective niching method for locating solutions at the peaks in the fitness landscape, it is not applicable for uneven multimodal functions and easily fails to keep multiple optima. To overcome this problem, we introduce a dynamic niche sharing scheme which is able to adjust the distance of the niche radius in the search process dynamically for the BA. In the experiment, we employ several multimodal functions and compare DNRBA with the conventional BA to evaluate the performance of DNRBA.

**CCS Concepts**

• **Mathematics of computing➝Bio-inspired optimization.**

**Keywords**

Bat Algorithm, Multimodal optimization, Swarm Intelligence, Niching method

# INTRODUCTION

There are many studies on evolutionary algorithms (EAs) solving the real-world optimization problems which are mostly multimodal and complex optimization problems. These problems have not only single global optimum but also many local optima, hence EAs are required to find multiple optima which might be changed as the environment changes.

To tackle multimodal optimization problems, various niching methods have been proposed. Thomsen proposed the DE [8] extends with a crowding scheme (CDE) [9] to replace the high-quality solutions by the most similar candidate solutions. Li proposed the DE with Speciation (SDE) [4] to keep a solution away from the nearest neighbor solution when the distance of both solutions is less than a threshold. However, these niching methods are not still enough to find multiple local optima because both of them do not consider searching globally though they consider the solution movement according to the Euclidean distance between the nearest neighbor solutions. For solving this problem, this paper focuses on the Bat Algorithm (BA) [5] that has the characteristic of echolocation which can predict the distance between bats and the target (*i.e., object/food source*). This algorithm enables bats to estimate the distance between their location and the target even in situations such as the target being surrounded by obstacles and the absence of light. Bats can adjust their velocity which is controlled by their loudness, pulse emission rate and frequency, toward the target. While the iteration search step continues until bats reach the target in the evolutionary process, bats will stop searching the target within its perceptible distance. This research employs a BA which copes with exploitation and exploration search, and extends it with niche radius for multimodal optimization. The niche radius is a threshold distance calculated by the fitness landscape and the number of its peaks.

This paper is organized as follows. After this section, the mechanism of the BA and the proposed algorithm DNRBA are explained in Sections 2 and 3. Section 4 describes the multimodal functions as the test-bed problem in the experiment. Section 5 shows the results while Section 6 discusses them. Finally, this paper concludes in Section 7.

# BAT ALGORITHM

As mentioned in Section 1, the BA is a metaheuristic algorithm based on the bat behavior with echolocation which are the loudness and the pulse emission rate of the reflect wave controlling the balance between the exploitation and the exploration search. As a bat approaches a better solution than its current solution, the BA turns the loudness up and the pulse rate down. The bats behavior is updated by the following three solution search phases: (i) the bat flies to the target (*i.e.*, the bat which finds the best solution) with the flight speed controlled by frequency ; (ii) the bat flies around the target as a local search; and (iii) the bat flies randomly in the search space as a global search.

First, in the exploitation phase (i), all bats change their locations with their velocities toward the global best solution. For this calculation, the frequency, velocity , and location , of the bat are calculated as follows:

In detail, the new solution is updated by adding the new the velocity which is derived from the previous velocity , the distance between the global best location and the previous location , and frequency which range is where and . is uniform random value from 0 to 1. Next, in the local search phase (ii), the new solution is generated around the global best solution as follows:

where is uniform random value within . In Eq.(6), is the averaged loudness of all bats. Finally, in the global search phase (iii), is generated randomly in the search space as follows:

where and describe the upper and lower bounds of the search space, and is a dimensional uniform random value within .

When a bat finds a better solution than the current one, the loudness and pulse emission rate are updated as follows:

Note that the loudness is initialized as and the pulse rate is initialized as a uniform random value within or a number close around zero. The parameters and are the symbolized damping coefficients. The pseudo code of the BA is given in the Algorithm 1 and its brief summary is described below.

* STEP1: Population initialization of bats (line 1 to 3)  
  The population of bats (i=1, 2, ..., N), the loudness , the pulse rate are initialized as the initial values. The frequency is initialized by Eq.(1)
* STEP2: New solution updates (line 6)  
  The new solutions is calculated by Eqs. (2) and (3).
* STEP3: New solution generation around the global best solution (line 7 to 9)  
  A new solution is generated around by Eq. (4) when the pulse emission rate is lower than a random value.
* STEP4: Random new solution generation (line 10)  
  A new solution is generated randomly by Eq. (5).
* STEP5: Solutions update (line 11 to 14)  
  When , the best solution is selected from , , and by Eqs.(6) and (7)
* STEP6: Return to STEP2

# SHARING SCHEME

The first sharing mechanism was proposed by Holland [3] to spread individuals widely in multimodal optimization. The concept of a sharing scheme is to reduce the fitness value of similar individuals and classify these individuals within the population.

Algorithm 1 Bat Algorithm

**Require**: Objective Function

1: Initialize Population (i=1,2,..., N) and

2: Define frequency at location [Eq.(1)]

3: Initialize pulse rates , and loudness

4: **while** ( Max number of iterations) **do**

5: **for** i=1 to N **do**

6: Generate a new solution and velocity [Eqs.(2),(3)]

7: **if** rand **then**

8: Generate a new solution around a global best solution [Eq.(4)]

9: **end if**

10: Generate a new solution randomly [Eq.(5)]

11: **if** (rand & **then**

12: Accept the new solution, and update pulse rate & loudness [Eqs. (6)(7)]

13: **end if**

14: Evaluate all bats and select a best solution in the current solutions

15: **end for**

16: **end while**

## Niche Radius

The *niche radius* is the distance calculated by the upper and lower bounds of the search space. The equation is defined by:

where are the upper and lower bounds of the dimension in the search space, and is the number of peaks of the fitness landscape.

## Fitness Sharing

SAMPLE: Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

*Conference’10*, Month 1–2, 2010, City, State, Country.

Copyright 2010 ACM 1-58113-000-0/00/0010 …$15.00.

DOI: http://dx.doi.org/10.1145/12345.67890

*Fitness Sharing* is derived from the concept of a *crowding scheme* [2] replacing a new individual by nearby similar individual in the population. The most widely used *sharing function* is given as follows:

where is the distance between individuals *i,j*, and is the niche radius defined above in Eq.(9) as the threshold. is the coefficient parameter, basically set to 1. By the *sharing function*, the *niche count* which represents the density of nearby similar individuals, is defined by:

Subsequently, the *shared fitness* is given by:

where is the raw fitness value of the individual and is the niche count. The *shared fitness* indicates the fitness of the individual considering the density of similar individuals.

## Dynamic Niche Sharing

In order to cut off the redundancy of *the shared fitness*, *dynamic niche sharing* is proposed by Miller in [6]. This scheme enables to identify the *q* peaks of the fitness landscape and classify all individuals into several groups in the same domain with the radius dynamically.

where is the *j*-th niche radius and is the *niche count* defined in Eq.(11) as mentioned above. The *shared fitness* is calculated as follows:

# PROPOSED ALGORITHM

## Bat Algorithm with Dynamic Niche Radius

In our algorithm, we provide a new *dynamic niche radius* to classify all individuals into several groups by the density of some of them in the same domain with niche radius to avoid overlapping the same peak in the fitness landscape. The new *dynamic niche radius* is updated each iteration step as follows:

By this equation, the movement of solutions is given by

where is the best solution in the domain of the niche radius and indicates the Euclidean distance between the nearest neighbor solutions.

In the local search phase, the new solution is generated around the best solution in the domain of the radius as follows:

where are the same values as in the BA.

In the global search phase, the new solution is generated randomly in all domains of the dynamic niche radius as follows:

## Algorithm Description

The pseudo codes of the Dynamic Niche Radius and DNRBA are given in Algorithm 2 and 3. Its brief summary is described below.

Algorithm 2 Dynamic Niche Radius

**Require**: Current Population and

1: **for** i=1 to N **do**

2: **for** j=1 to N **do**

3: Calculate between individual

4: **if** **then**

5: [Eq.(10)]

6: **else**

7:  [Eq.(10)]

8: **end if**

9: **end for**

10: [Eq.(11)]

11: **end for**

12: **for** i=1 to N **do**

13: **if**  **then**

14: [Eq.15]

15: **else**

16:  [Eq.15]

17: **end if**

18: **end for**

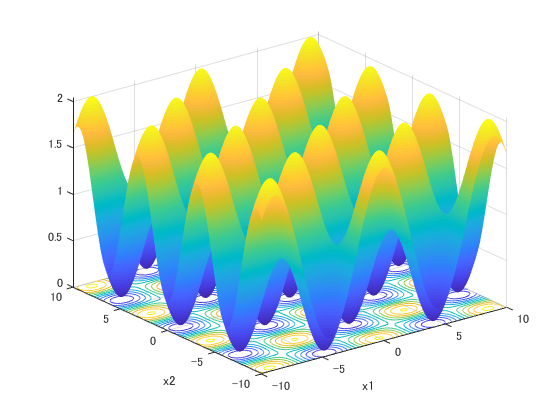
19: **return** Dynamic Niche Radius

* STEP1: Population initialization of bats (line 1 to 3)  
  The population of bats , the loudness , the pulse rate are initialized as the initial values. The frequency  initialized by Eq.(1).
* STEP2: Calculate dynamic niche radius (line 5)
* STEP3: New solution updates (line 7)  
  The new solutions are calculated by Eqs. (16) and (17)
* STEP4: New solution generation around the best solution (line 8 to 10)  
  A new solution is generated around by Eq. (18) when the pulse emission rate is lower than a random value.
* STEP5: Random new solution generation (line 11)  
  A new solution is generated randomly by Eq. (19).
* STEP6: Solutions update (line 15 to 16)  
  When rand < , the best solution is selected from , , and by Eqs.(6) and (7).
* STEP7: Return to STEP2

# EXPERIMENTAL PROCEDURE

To measure the number of local optima and the convergence speed, we compare the performance of the BA with the DNRBA. In this section, two multimodal functions which Griewank function has both global and local optima and Shubert function has only global optima, are employed for minimization [1][7].

Algorithm 3 Bat Algorithm with Dynamic Niche Radius (DNRBA)



1. Fitness landscape

**Require**: Objective Function

1: Initialize Population (i=1,2,..., N) and

2: Define frequency at location [Eq.(1)]

3: Initialize pulse rates , and loudness

4: **while** ( Max number of iterations) **do**

5: Calculate dynamic niche radius (Algorithm 2)

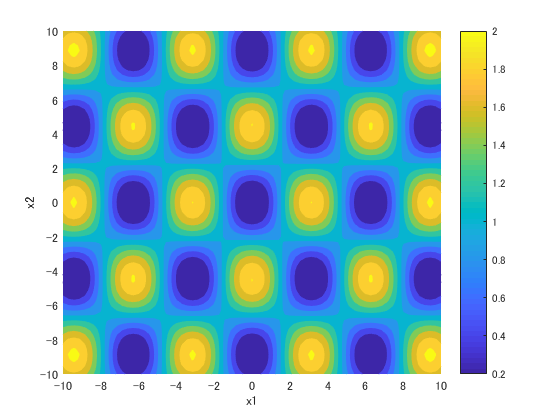
6: **for** i=1 to N **do**

7: Generate a new solution and velocity [Eqs.(16),(17)]

8: **if** rand **then**

9: Generate a new solution around the best solution in the radius [Eq.(18)]

10: **end if**



1. Contour plot

11: Generate a new solution randomly in the radius [Eq.(19)]

12: **if** (rand & **then**

13: Accept the new solution, and update pulse rate & loudness [Eqs. (6)(7)]

14: **end if**

15: Evaluate all bats and select a best solution in the current solutions

16: **end for**

17: **end while**

## Multimodal Test Functions

Figure : Griewank Function

**: Griewank Function**

This function is described as follows as shown in Fig. 1.

The fitness value of the global optima is . This function has 17 optima (1 global optimum and 16 local optima) in the range between [-10, 10].

**: Shubert Function**

This function is described as follows as shown in Fig. 2.

where *D* is the number of dimensions and the fitness value of the global optima is -187.731. This function has global optima in the range of the search space with . Fig. 2 shows an example of the Shubert 2D function which has 18 global optima.

## Experimental Measurement

### Peak Ratio

This experiment employs Peak Ratio (PR) [9] as the evaluation criterion in the CEC (*IEEE Congress on Evolutionary Computation*) 2013 competition [10]. The PR value measures the ratio of the found global and local optima in the total number of true peaks and it is calculated as follows:

where *MR* indicates the maximum run, *FPs* indicates the number of peaks found by the optimization algorithm. *TP* indicates the number of all known peaks of the function. We define that the peak is found when the Euclidean distance between the all known peaks and the nearest solution calculated by the optimization algorithm is less than the thresholds {1.0E-1, 1.0E-2}.

### Peak Accuracy

To measure how close solutions are close to the peaks (global and local optima), we employ Peak Accuracy (PA) [9] calculated as follows:

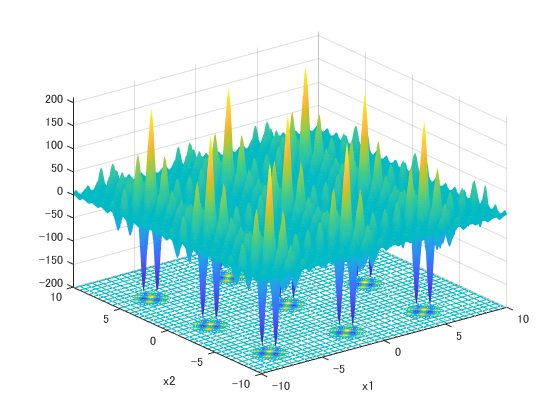
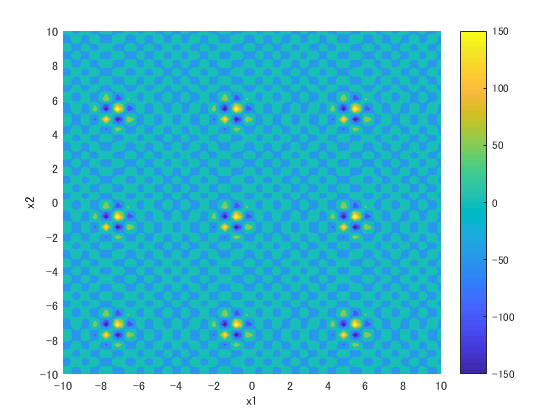
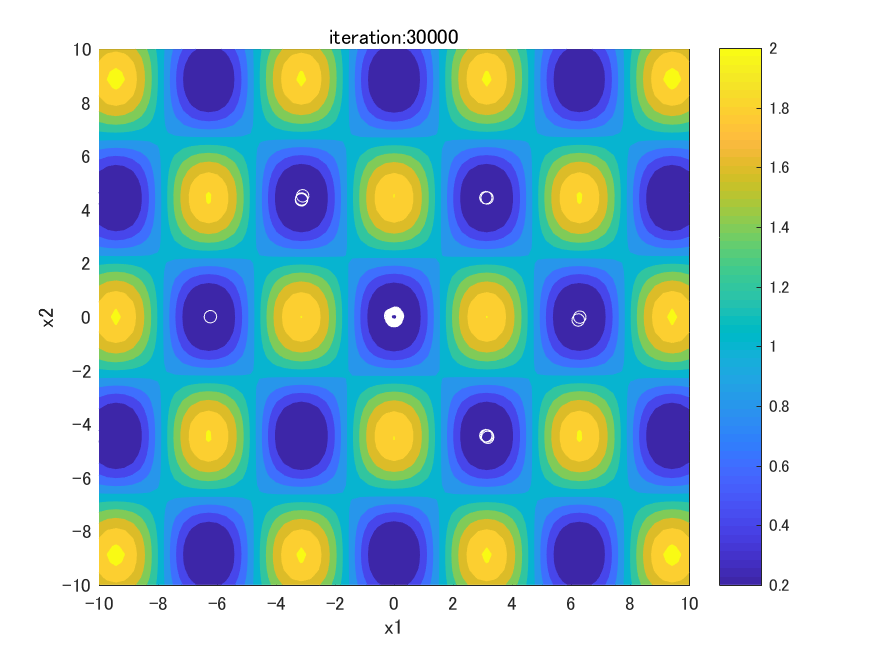
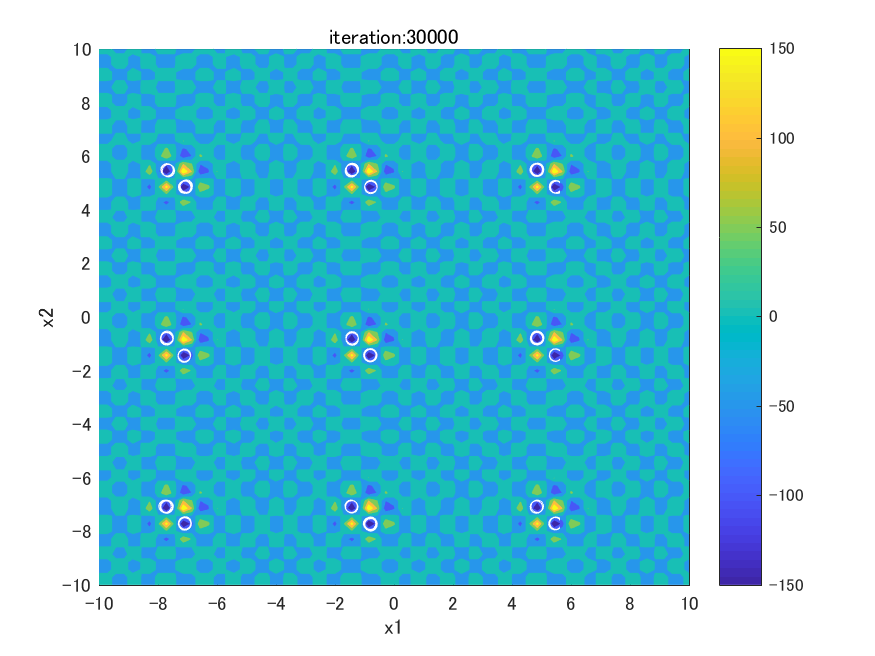
1. Fitness landscape
2. Contour plot

Figure : Shubert Function

where and denote the each known peak and the nearest neighbor solution. As the closest distance between both of them is short, the value of PA is close to 0.

## Experimental Parameters

We employ the parameters as follows: frequency , loudness , parse rate with . The population size . This experiments are conducted with 30 runs with different random seeds and 30000 evaluations as the termination condition for each run.

# RESULTS AND ANALYSIS

To test the effect of the DNRBA mechanism, this section investigates the peak ratio (PR) and the peak accuracy (PA) of each benchmark test function. Table 1 and 2 show the results that the PR and the PA values of two algorithms based on the settings of averaged over 30 individual runs at the final iteration. Fig. 3 and 4 show that the solutions are distributed at the final iteration for all functions.

Figure BA

## Peak Ratio

The case of =1.0E-1 from Table 1, the PR values of DNRBA were greatly higher than BA in . It can be seen that DNRBA found almost all global and local optima, as shown in Fig. 4(a) Although Fig. 4(b) appears to locate solutions at all peaks, the values of DNRBA were less than BA in , because has an extremely sharp and complex landscape. Moreover, the exploitation of BA outperformed that of DNRBA so that  is for finding only global optima (\textit{not searching local optima}). Table 2 shows the results of the case of =1.0E-2. The value of DNRBA is the same as =1.0E-1 in . However, DNRBA remarkably decreased from 0.4241 to 0.0426 in . It can be seen that the exploitation of DNRBA is not efficient to reach the peaks precisely.

## Peak Accuracy

DNRBA outperformed BA for from Table 1 and 2 in =1.0E-1, 1.0E-2 to find global and local optima. In contrast, the PA value of BA was better than DNRBA in . BA enables to locate all solutions at all peaks from Fig. 3(b). As well as Fig. 4(b), although DNRBA seems to locate at several global optima (several solutions are located at no peaks), the value of DNRBA is lower than BA.

# CONCLUSION

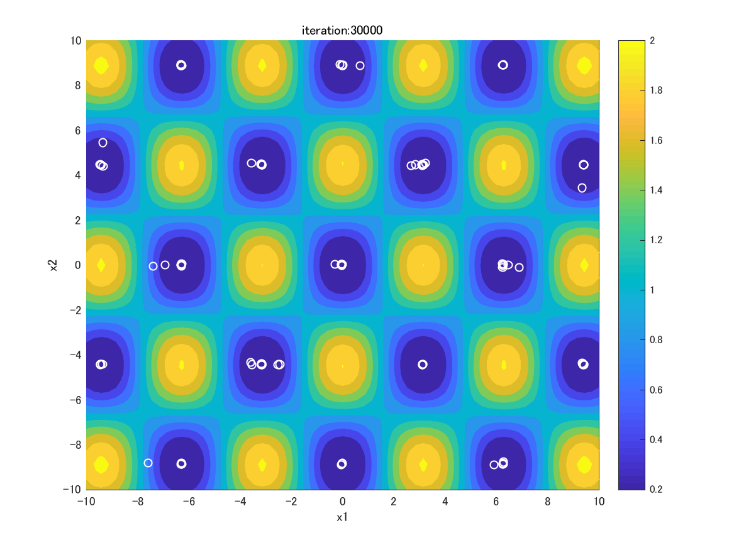
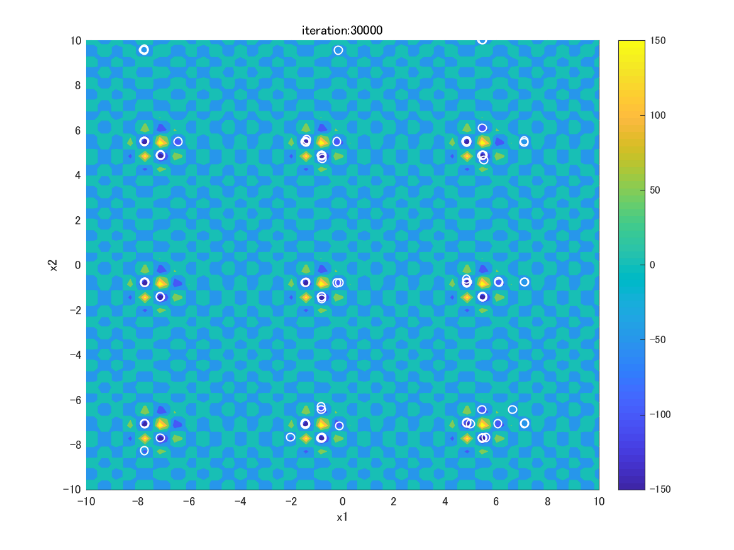
This paper proposed BA extended with a Dynamic Niche Radius which enables the solutions to avoid overlapping into the same peak, and to locate multiple global optima in several multimodal functions. To evaluate the performance of DNRBA, this algorithm

Table Peak Ratio and Peak Accuracy of BA and DNRBA (averaged over 30 runs)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | BA | | DNRBA | |
| Function | PA  (Mean and SD) | PR  (Mean and SD) | PA  (Mean and SD) | PR  (Mean and SD) |
|  | 0.2725 0.0598 | 0.2416 0.0149 | **0.9373** 0.1176 | **0.0094** 0.0155 |
|  | **0.4889** 0.1819 | **1.8160** 0.4990 | 0.4241 0.388 | 2.4123 0.6780 |

Table Peak Ratio and Peak Accuracy of BA and DNRBA (averaged over 30 runs)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | BA | | DNRBA | |
| Function | PA  (Mean and SD) | PR  (Mean and SD) | PA  (Mean and SD) | PR  (Mean and SD) |
|  | 0.2725 0.0608 | 0.2416 0.0151 | **0.9373** 0.1176 | **0.0094** 0.0155 |
|  | **0.0556** 0.0619 | **1.8160** 0.4990 | 0.0426 0.0477 | 2.4123 0.6780 |



were compared with BA. The results show that NRBA outperformed at not only global optima but also local optima in the fitness landscape, because the spatial distribution mechanism in DNRBA copes with the trade-off between the exploration and the exploitation. In contrast, BA is still better than NRBA regarding the exploitation due to converge to a single global best solution. In future work, we will compare DNRBA with current state-of-the-art algorithms and apply it for dynamic optimization problems.

# REFERENCES

1. R.Stoean, et al. C. Stoean, M. Preuss: Multimodal optimization by means of a topological species conservation. *IEEE Transactions on Evolutionary Computation* 14, 6, 842-864, (2010).
2. K.A. DeJong: Analysis of the behavior of a class of genetic adaptive system, *Ph.D. dissertation, Dept. Compu*t. Sci. (1975).
3. J.H. Holland: Adaptation in natural and artificial systems. *Ann Arbor: University of Michigan Press*. (1975).
4. X.Li: Efficient Differential Evolution using Speciation for Multimodal Function Optimization. *In Proceedings of the Conference on genetic and evolutionary computation (GECCO’05)*, 873-880,(2005).
5. X. S. Yang: A Metaheuristic Bat-Inspired Algorithm, in: Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Springer, Berlin, 284, 65-74, (2010)
6. B. Miller: Genetic algorithms with dynamic niche sharing for multimodal function optimization. *In: Proceedings of the 1996 IEEE International Conference on Evolutionary Computation (ICEC’96)*, (1996).
7. M. Molga nad C. Smutnicki. Retrieved June 2013. Test functions for optimization needs, (2005). DOI=<http://www.zsd.ict.pwr.wroc.pl/files/docs/functions.pdf>.
8. K. Price R. Storn: Differential Evolution – A simple and efficient adaptive scheme for global optimization over continuous spaces. *Journal of global optimization* 11, 4, 341-359, (1997).

Figure DNRBA

1. R. Thomsen. 2004. Multimodal Optimization using Crowding-based Differential Evolution. *In the IEEE Congress on Evolutionary Computation (CEC2004)* 2, 4, 1382-1389, (19-23 June 2004).
2. A. Engelbrecht X. Li and M. G. Epitropakis: Benchmark Functions for CEC’2013 Special Session and Competition on Niching Methods for Multimodal Function Optimization. *Evol. Comput*. (2013).

Columns on Last Page Should Be Made As Close As Possible to Equal Length

Authors’ background

|  |  |  |  |
| --- | --- | --- | --- |
| Your Name | Title\* | Research Field | Personal website |
|  |  |  |  |
|  |  |  |  |

\*This form helps us to understand your paper better, the form itself will not be published.

\*Title can be chosen from: master student, Phd candidate, assistant professor, lecture, senior lecture, associate professor, full professor