ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/328018222

Testing Optimization Algorithms Easier with a New Test Function for Single-Optimization Problems and Validating Evolutionary Algorithms (2305-0543)

Article · October 2018

CITATIONS
2

READS
706

1 author:

Seyed Muhammad Hossein Mousavi
Independent Scientist

135 PUBLICATIONS **162** CITATIONS

SEE PROFILE

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

Project New support vector finder system, based on evolutionary clustering and pentagon calculations View project

Project Iranin Kinect Face Data Base (IKFDB) View project

All content following this page was uploaded by Seyed Muhammad Hossein Mousavi on 02 October 2018.

The user has requested enhancement of the downloaded file.

Vol. 8(30), Oct. 2018, PP. 4009-4017

Contents list available at IJMEC

International Journal of Mechatronics, Electrical and Computer Technology (IJMEC)

Journal Homepage: www.aeuso.org

PISSN: 2411-6173 - EISSN: 2305-0543

Testing Optimization Algorithms Easier with a New Test Function for Single-Optimization Problems and Validating Evolutionary Algorithms

Seyed Muhammad Hossein Mousavi

Department of Computer Engineering, Bu Ali Sina University, Hamadan, Iran

Phone Number: +98-021-33057186

***Corresponding Author's E-mail:** mosavi.a.i.buali@gmail.com*

Abstract

Selecting the best element, based on some prior knowledge and inputs with considering some criterion with the goal of having lowest cost value is called optimization which has lots of application in mathematics and related sciences. There are lots of optimization and evolutionary algorithms, which could be assessed somehow for the power of functionality purposes. Functions that take input of these algorithms and returned numerical results based on their performance is called optimization test function. These test functions could assess convergence speed, accuracy, robustness and their total functionality of optimization and evolutionary algorithms. This paper presents a robust and new test function for single-optimization algorithms and especially for testing the performance of evolutionary algorithms. Single-objective optimization algorithms are the underlying basis to more sophisticated algorithms such: multi-objective optimization algorithms, niching, constrained optimization algorithms, etc. For validation purposes, some of the evolutionary algorithms are employed which acquired results is compared with other benchmark test functions results on the same evolutionary algorithms and the same condition. The results shows that proposed test function can perform as good as other benchmark test functions and converges the evolutionary algorithms in global maxima with low number of iterations and high accuracy. This test function is Pyramid function, according to its 3-D model.

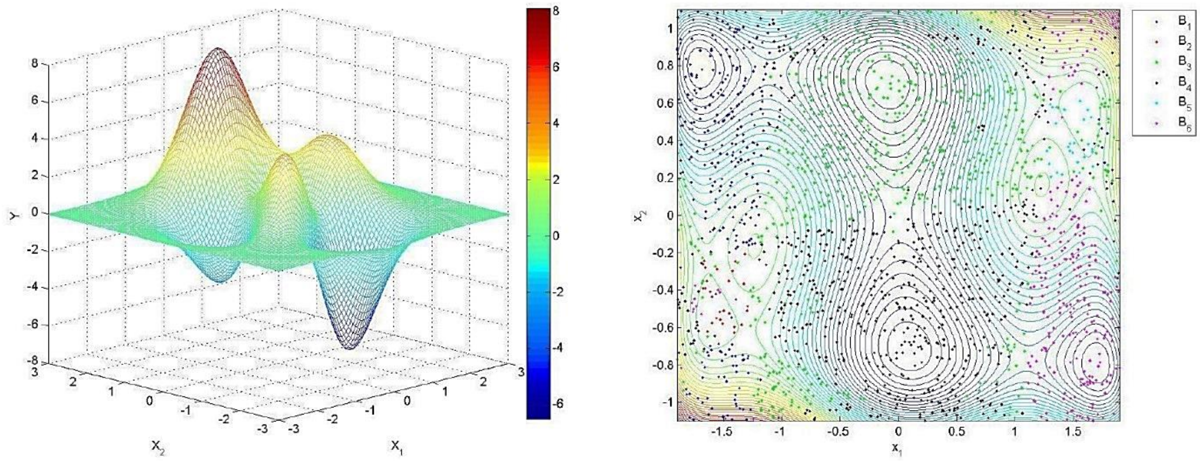
Keywords: *Optimization, Evolutionary Algorithms, Single-Optimization Test Function, Convergence Speed, Global Maxima, Pyramid Function*

1. Introduction

The main goal of all optimization problems is finding the best value of x out of set (x) based on total index set of F= [f1, f2… fn]. This index is called 'objective function' [1]. The test function must have different characteristics in order to be able of being utilized properly for a variety of algorithms. The reliability, functionality and validation of optimization algorithms are all depended on choosing a proper test function for that algorithm. All test functions must be Unbiased and Diverse. For instance, with performing it by an evolutionary algorithm [18] [19], in every performance we will achieve different results [2]. In order to solve an optimization problem, two questions should be answered. First, which aspects of function landscape makes optimization problems complicated? And second, which type of prior data is more suitable for detecting a special kind of function landscape? For answering these questions, test function is classified based on features such: Separability, Valleys, Basins, Modality and Dimensionality [3]. Number of ambiguous peaks in function landscape is related to modality of function. If along processing, algorithm confronts one of these peaks, it is possible for algorithm to be trapped in one of these ambiguous peaks. A relatively steep decline surrounding a

Article History:
Received Date: Jul. 24, 2018
Accepted Date: Sep. 03, 2018
Available Online: 01. Oct, 2018

large area is called a Basin. Optimization algorithm can be easily attracted to these fields. A Valley or time period happens when a narrow area of limited changes is occupied by an area of steep descent [4]. Such Basin, optimization algorithms are attracted to these areas. The Separability is a measure of difficulty of different benchmark functions. In general, separable functions are relatively easy to solve, when compared with their inspirable counterpart, because each variable of a function is independent of the other variables. It is due to this fact that every variable of function is independent of other variables. Generally, difficulty level of a problem increases by increasing of its dimensions. Also, if the number of parameters increases, search space will become bigger exponentially. For more data on this you can refer to [2]. Figure 1 shows 3-model of a test function with emphasize on Valleys and Basins. Optimization and evolutionary algorithms has application in planning [21], decision making [22], statistical pattern recognition [23], machine learning [24], chemistry [25], business [26], management [27], engineering [28] and more.



(a) (b)

Figure 1: (a). Illustrates the landscape as a three dimensional surface composed of ridges, valleys, plateaus and basins, (b). Contour plot which colors representing one of six observed basins [5]

2. Prior related researches

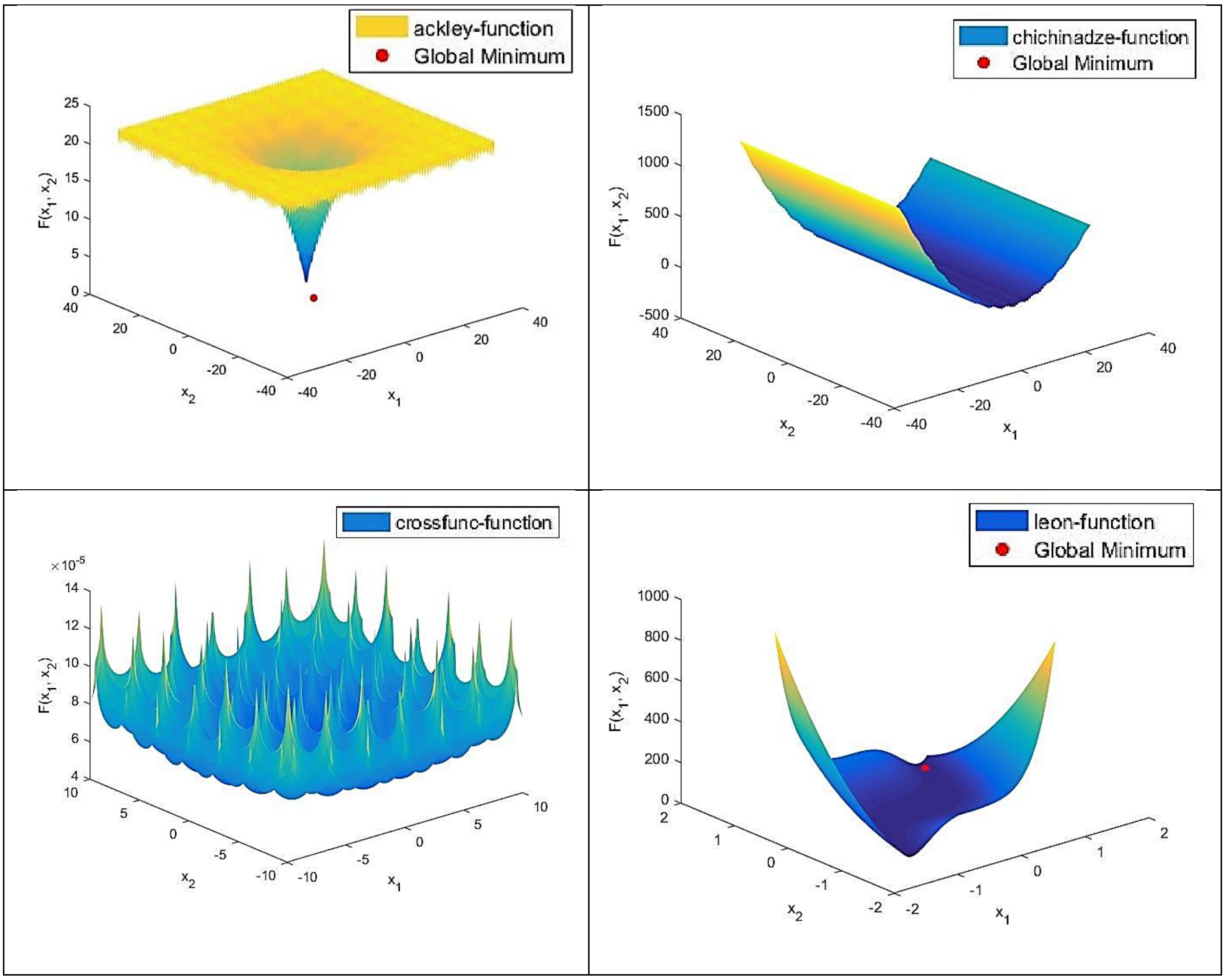
There are more than 200 test function for optimization which we use 8 of those for comparing purposes. Comparing test functions which we use, are among of most famous ones which are called benchmark test functions. All of these functions are for single-optimization problems, as our proposed one is one of them. Ackley [6], Rastrigin [6], Schaffer [6], Leon [7], Cross [7], Zettle [7], Chichinadze [7], and Penholder [7] are used test functions for comparing purposes. Table 1 represents these functions' formulas, global minimums and search domains. Table 2 presents the 3-D models of mentioned test functions.

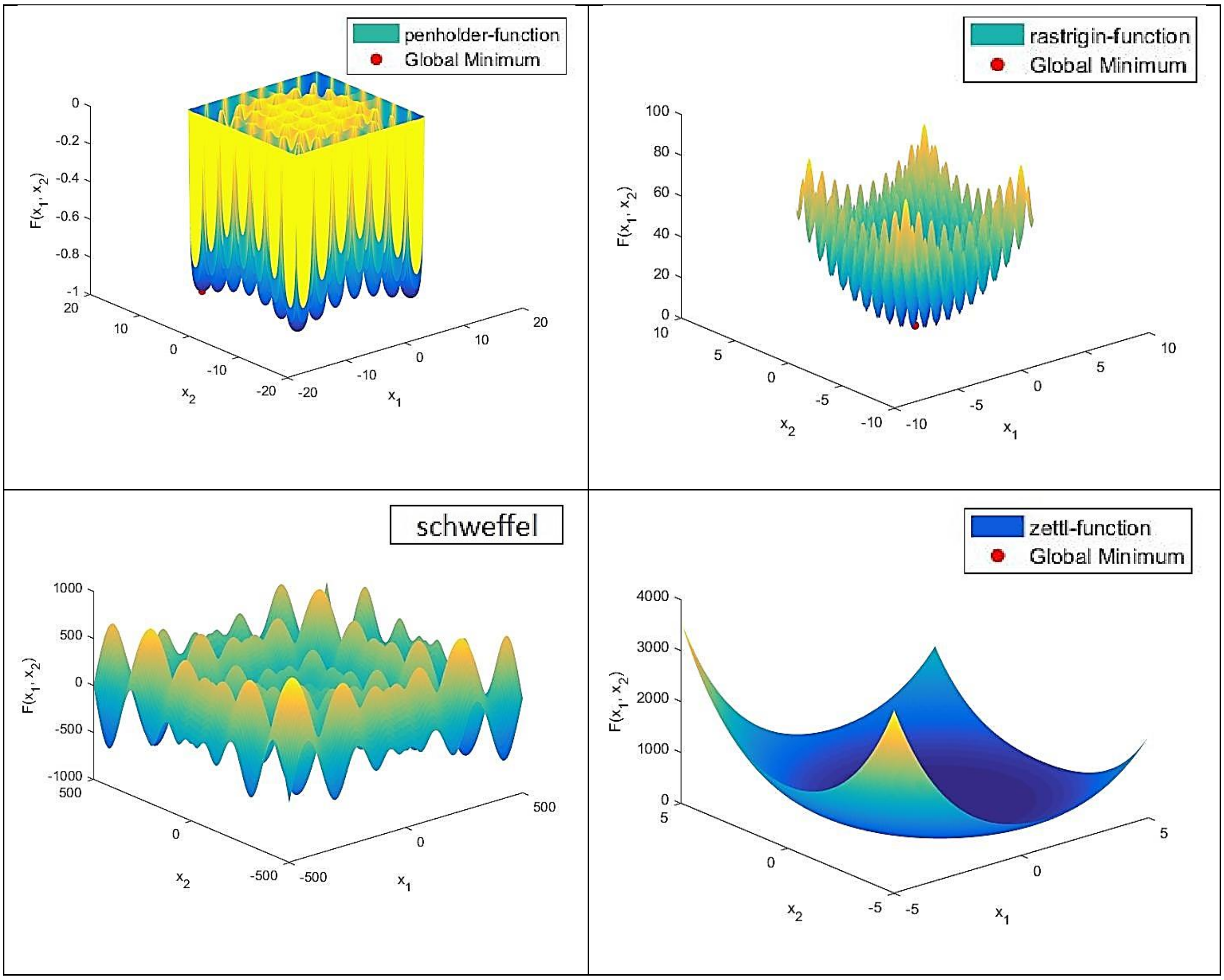
Table 1: Used benchmark single-optimization test functions properties

Function	**Formula**	**Global Minimum**	**Search Domain**
Ackley (1)	$f(x) = An + \sum_{i=1}^{n}\left[x_i^2 - A\cos(2\pi x_i)\right]$ Where: $A = 10$	$f(0,0) = 0$	$-5.12 \le x, y \le 5.12$
Rastrigin (2)	$f(x,y) = -20\exp\left(-0.5\sqrt{-0.2(x^2+y^2)}\right) - \exp\left(0.5(\cos(2\pi y))\right)$	$f(0,0) = 0$	$-5 \le x, y \le 5$

Schaffer (3)	$f(x) = 0.5 + \frac{\sin^2[\sqrt{x_1^2 + x_2^2}] - 0.5}{[l + 0.001(x_1^2 + x_2^2)]^2}$	f(0,1.2)=0.2	$x_1, x_2 \in [-100,100]$
Leon (4)	$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$	$f(1,1) = 0$	$x_1, x_2 \in [-1.2,1.2]$
Cross (5)	$f(x) = [\lvert \sin(x_1) \sin(x_2) e^{\lvert 100 - [(x_1^2+x_2^2)^{0.5}]/\pi \rvert} \rvert + 1]^{-0.1}$	$f(x^*) = 0$	$x_i \in [-10,10]$
Zettle (6)	$f(x) = (x_1^2 + x_2^2 - 2x_1)^2 + 0.25x_1$	$f(-0.0299,0) = -0.003791$	$x_1, x_2 \in [-5,5]$
Chichinadze (7)	$f(x) = x_1^2 - 12x_1 + 11 + 10\cos(\pi x_1/2) + 8\sin(5\pi x_1) - (1/5)^{0.5} e^{-0.5(x_2-0.5)^2}$	$f(5.90133, 0.5) = -43.3159$	$x_1, x_2 \in [-30,30]$
Penholder (8)	$f(X) = -\exp[\lvert 1 - [(x_1^2 + x_2^2)^{0.5}/\pi] \rvert$	$f(x^*) = -0.96354$	$x_{isubseteq}[-11,11)$

Table 2: 3-D models of used benchmark single-optimization test functions





3. Proposed test function

Proposed single-objective test function's equation, global minimum and search domain is shown in (9). Also Figure 2 represents 3-model of proposed test function in different views. As it is clear, this function looks like a pyramid, so it is called pyramid test function. Table III shows implementation code for the pyramid function (using Matlab software). Figure 3 presents the performance of proposed pyramid test function using Genetic Algorithm in 50 iteration. As it is clear in this figure, algorithm could be converged in -61.1611 point, which is promising. Note that in this test, convergence lower bound did not considered on zero and it could go down even in –n as number of iteration let it.

$$Pyramid\ function$$

$$= \partial(sin(p\ .*\ 2)) * 12 + cot(\sqrt{(exp^{\pi})})\ *\ \pi.*\ (\frac{\sqrt{0.1}}{\pi} * ((\frac{\sqrt{sqrt}(\pi+12)}{20})) -\ 20$$

$$*\ exp^{(\,-1/73\, *\, exp\, \sqrt{\left(\frac{12}{length(p.*313)}\right)})}\ *\ exp^{(\,\partial(p\ .\hat{}\ 5))\hat{}5)} + (\frac{\left(\frac{2}{5}\right)}{\sum(p/3\ ./\ 3)})$$

$$-\ exp^{\frac{\left(\frac{2}{length(p*999)}\right)}{\prod(sin(12*pi.\hat{}1.2\ .*\ p/1.25))}})))$$

F (0, 0) =0 || -40.313≤variables≤40.313 || p=max (-40.313, min (40.313, p)) (9)

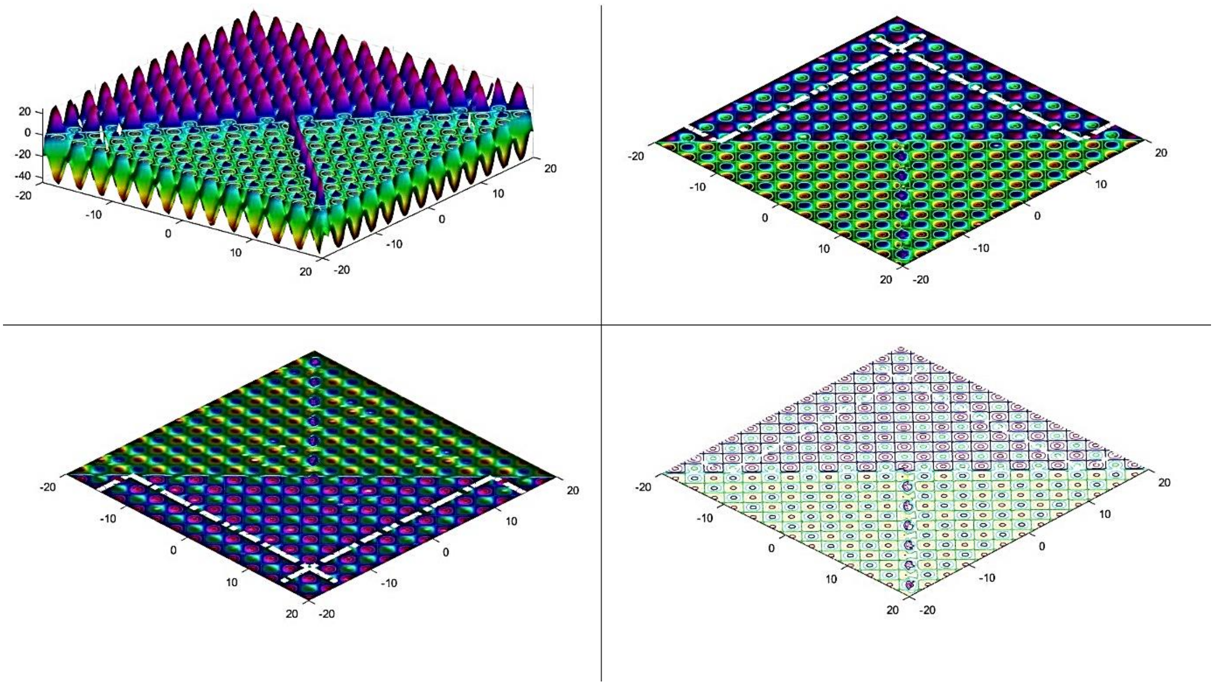


Figure 2: 3-D model of proposed test function in different views

Table 3: Implementation code for the pyramid function (using Matlab software)

```
function scores = Pyramid fcn(pop)
scores = zeros(size(pop,1),1);
   for i = 1:size(pop,1)
    p = pop(i,:);
     p = max(-40.313,min(40.313,p));
      scores(i) =  diff(sin(p .* 2))*12+cot(sqrt(exp(pi))) * pi.*(sqrt(0.1))/ pi*((sqrt(pi+12)./20))...
      -  20* exp( -1/73 * exp ( sqrt( (12/length(p.*313))* exp( diff(p .^ 5))^5)...
     +((2/5) /sum(p/3 ./ 3))- exp((2/length(p*999)) / prod(sin(12*pi.^1.2 .* p/1.25)))));
end
```

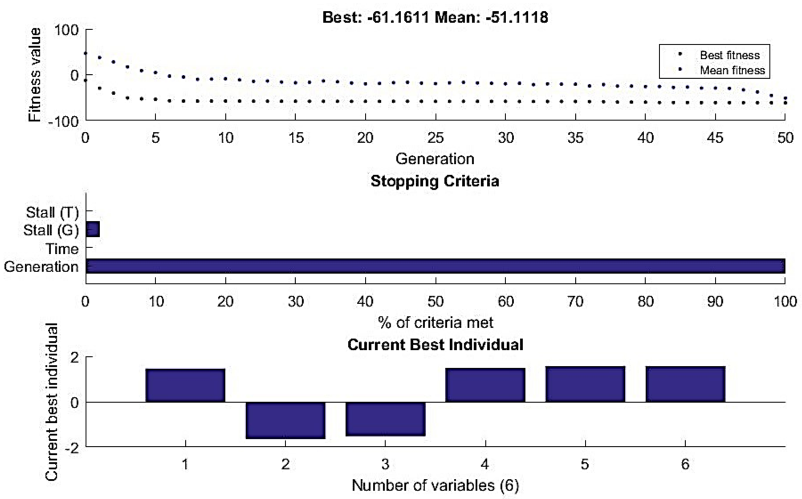


Figure 3: Performance of proposed pyramid test function using Genetic Algorithm in 50 iteration

4. Validation and results

These test functions also called artificial landscape, which proposed test function or artificial landscape will compare with mentioned 8 other famous and benchmark single-objective optimization test functions or artificial landscape. Functions are Ackley [6], Rastrigin [6], Schaffer [6], Leon [7], Cross

[7], Zettle [7], Chichinadze [7], and Penholder [7] and explained in section 2. Also validation process performs using 6 evolutionary optimization algorithms which are Genetic Algorithm [8], Particle Swarm Optimization (PSO) [9], Artifactual Bee Colony algorithm (ABC) [10], Ant Colony Optimization (ACO) [11] Differential Evolution [12] [13] Bat Algorithm (BA) [14] in same condition and best cost average along with respective standard deviation will be calculated. All the objective functions considered as cost (even GA). It means as acquired value be closer to zero and in lowest number of iteration, that function have better performance. Figure 4 indicate the performance of benchmark test functions along with proposed one using ABC algorithm. Table 4 represents the parameters for all optimization and evolutionary algorithms which are going to be tested with 8 + proposed functions in same conditions. And finally Table 5 shows final result of comparing and validation using 8 + 1 test functions on 6 benchmark and famous evolutionary algorithms in same condition (based on parameter's values in Table 4). Figure 4 presents the performance of Bat Algorithm on proposed Pyramid test function in last 20 iterations of run. As it is clear in Figure 4, all the bats are going toward the center of the function which is our global maxima or best point. As number of iteration on this run is just 100, it is not completely converged in last iteration, and if it increased for example to 300, all the bats must be in the center of the function. Figure 5 shows the performance of the test functions (including ours), with ABC algorithm during 300 iterations in similar condition. As it is clear, our Pyramid test functions is converged faster than others using this algorithms in 71 iteration number. Also worst performance is belong to Schaffer function, which converged at 282 iterations.

Table 4: Parameters for all optimization and evolutionary algorithms for validating test functions (100 generations)

PARAMETERS	GA	PSO	ABC	ACO	DE	BA
Number of Decision Variables	800	800	800	800	800	800
Size of Decision Variables Matrix	[1,800]	[1,800]	[1,800]	[1,800]	[1,800]	[1,800]
Lower Bound of Variables	-5	-5	-5	-5	-5	-5
Upper Bound of Variables	10	10	10	10	10	10
Maximum Number of Iterations	700	1000	250	350	400	250
Population Size	100	40	35	20	20	40
Crossover Percentage	0.6	-	-	-	0.2	-
Number of Offspring's (Parents)	75	-	-	-	-	-
Mutation Percentage	0.3	0.2	0.1	0.2	0.2	0.1
Number of Mutants	45	-	-	-	-	-
Mutation Rate	0.2	0.3	0.2	0.3	0.1	0.2
Selection Pressure	7	-	-	-	-	-
Inertia Weight	-	1	-	-	-	-
Inertia Weight Damping Ratio	-	0.98	-	-	-	-
Personal Learning Coefficient	-	1	-	-	-	-
Global Learning Coefficient	-	2	-	-	-	-
Lower Bound of Scaling Factor	-	-		-	0.3	
Upper Bound of Scaling Factor	-	-		-	0.8	
Sample Size	-	-	-	55	-	-
Intensification Factor (Selection Pressure)	-	-	-	0.3	-	-
Deviation-Distance Ratio	-	-	-	1	-	-
Swarm size	-	-	20	-	-	-
Number of onlookers	-	-	50%	-	-	-
Number of employed bees	-	-	50%	-	-	-
Number of scouts	-	-	1	-	-	-
Loudness	-	-	-	-	-	0.7
Pulse rate	-	-	-	-	-	0.5
Frequency minimum	-	-	-	-	-	0
Frequency maximum	-	-	-	-	-	2

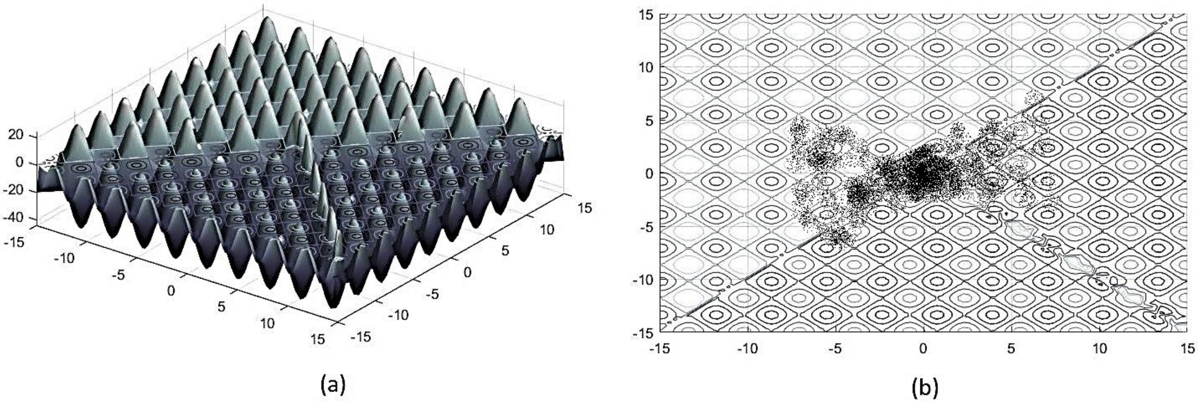


Figure 4: Proposed Pyramid function (a) and some bats in last 20 iterations of run using Bat Algorithm

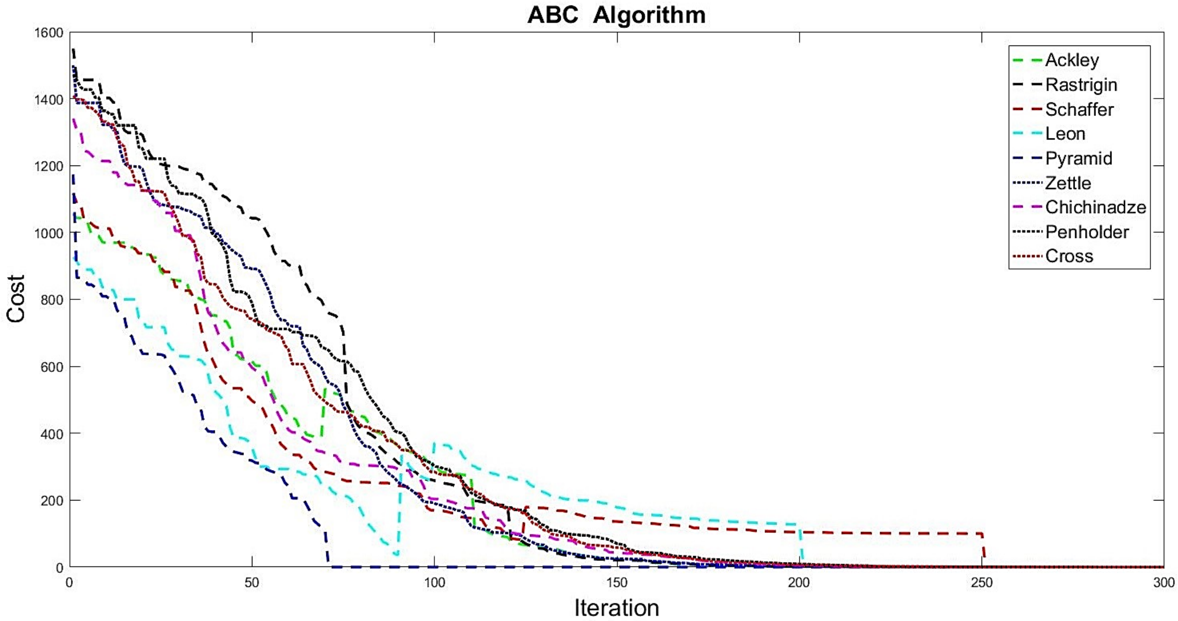


Figure 5: The performance of test functions using ABC algorithm

Table 5: Final result of comparing and validation using 8 + 1 test functions on 6 famous evolutionary algorithms in same condition (based on parameter's values in Table 4)

	GA	**PSO**	**ABC**	**ACO**	**DE**	**BA**
	Avg-STD	Avg-STD	Avg-STD	Avg-STD	Avg-STD	Avg-STD
Ackley	0.0220±0.0202	0.0689±0.0451	0.0524±0.0133	0.0443±0.0471	0.0207±0.009	0.0036±0.0078
Rastrigin	0.0330±0.0255	0.1008±0.0750	0.0900±0.0899	0.0657±0.0500	0.0312±0.0068	0.0152±0.0100
Schaffer	0.0539±0.0700	0.2020±0.1151	0.0980±0.0698	0.0912±0.0480	0.0688±0.0102	0.0291±0.0202
Leon	0.0740±0.0641	0.2520±0.1900	0.1121±0.0887	0.1516±0.0208	0.0836±0.0256	0.0404±0.0248
Zettle	0.0051±0.0011	0.0077±0.0021	0.0211±0.0050	0.0126±0.0044	0.0002±0.0002	0.0013±0.0020
Chichinadze	0.0113±0.0200	0.0251±0.0188	0.0442±0.0230	0.0313±0.0348	0.0082±0.0037	0.0020±0.0008
pyramid	0.0023±0.0001	0.0057±0.0025	0.0186±0.0055	0.0096±0.0030	0.0000±0.0027	0.0008±0.0001
Penholder	0.0200±0.0108	0.0787±0.0458	0.0933±0.0436	0.0663±0.0301	0.0104±0.0192	0.0042±0.0019

CONCLUSION

In this paper a new test function was modeled for single-objective optimization problems. This function has features such: multimodal, scalable, non-separable, differentiable and continuous. These features make the function compatible with the majority of optimization problems. The main aim of presenting this function is testing and validating evolutionary algorithms. Performance of proposed Pyramid test function in compare to other benchmark functions validates using some famous evolutionary algorithms in same condition and system returned satisfactory and promising results. Functions seems a little bit more complicated than other functions but in run time, no sign of delay has been detected. It is possible to change this function to multi-objective [21] format with a little bit of change too. Also function could be tested with other algorithms such as: Imperialist Competitive Algorithm ICA [15], Cuckoo search [16], Harmony search [17] and more.

References

[1] R.L. Haupt and S.E. Haupt. Practical genetic algorithms. John Wiley & Sons, 2004.

[2] M. Jamil and Xin-She Yang. “A literature survey of benchmark functions for global optimisation problems”, International Journal of Mathematical Modelling and Numerical Optimisation 4.2 (2013): 150-194.

[3] P.H. Winston, “Artificial Intelligence, 3rd ed.”, Addison-Wesley, 1992.

[4] C.-J. Chung and Robert G. Reynolds. "CAEP: an evolution-based tool for real-valued function optimization using cultural algorithms”, International Journal on Artificial Intelligence Tools 7.03 (1998): 239-291.

[5] Muñoz, Mario A., et al. "Algorithm selection for black-box continuous optimization problems: A survey on methods and challenges." Information Sciences 317 (2015): 224-245.

[6] Molga, Marcin, and Czesław Smutnicki. "Test functions for optimization needs." Test functions for optimization needs 101 (2005).

[7] Mishra, Sudhanshu K. "Some new test functions for global optimization and performance of repulsive particle swarm method." (2006).

[8] Mitchell, Melanie (1996). “An Introduction to Genetic Algorithms”. Cambridge, MA: MIT Press. ISBN 9780585030944.

[9] J. Kennedy and R. Eberhart, “Particle swarm optimization. «Proc.» IEEE International Conf. on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, 1995 (in press).

[10] Karaboga, Dervis. An idea based on honey bee swarm for numerical optimization. Vol. 200. Technical report-tr06, Erciyes university, engineering faculty, computer engineering department, 2005.

[11] Dorigo, Marco, and Christian Blum. "Ant colony optimization theory: A survey." Theoretical computer science 344.2 (2005): 243-278.

[12] Storn, Rainer, and Kenneth Price. "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces." Journal of global optimization 11.4 (1997): 341-359.

[13] Storn, Rainer. "On the usage of differential evolution for function optimization." Fuzzy Information Processing Society, 1996. NAFIPS., 1996 Biennial Conference of the North American. IEEE, 1996.

[14] Yang, Xin-She. "A new metaheuristic bat-inspired algorithm." Nature inspired cooperative strategies for optimization (NICSO 2010). Springer, Berlin, Heidelberg, 2010. 65-74.

[15] Atashpaz-Gargari, Esmaeil, and Caro Lucas. "Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition." Evolutionary computation, 2007. CEC 2007. IEEE Congress on. IEEE, 2007.

[16] Yang, Xin-She, and Suash Deb. "Cuckoo search via Lévy flights." Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on. IEEE, 2009.

[17] Geem, Zong Woo, Joong Hoon Kim, and Gobichettipalayam Vasudevan Loganathan. "A new heuristic optimization algorithm: harmony search." simulation 76.2 (2001): 60-68.

[18] Bäck, Thomas, David B. Fogel, and Zbigniew Michalewicz. Handbook of evolutionary computation. CRC Press, 1997.

[19] Back, Thomas. Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms. Oxford university press, 1996.

[20] Coello, Carlos A. Coello, Gary B. Lamont, and David A. Van Veldhuizen. Evolutionary algorithms for solving multi-objective problems. Vol. 5. New York: Springer, 2007.

[21] Moselhi, Osama, and Adel Alshibani. "Crew optimization in planning and control of earthmoving operations using spatial technologies." Journal of Information Technology in Construction (ITcon) 12.7 (2007): 121-137.

[22] Murty, Katta G. Optimization for decision making. Springer, 2009.

[23] Smith, Brandye M., and Paul J. Gemperline. "Wavelength selection and optimization of pattern recognition methods using the genetic algorithm." Analytica Chimica Acta 423.2 (2000): 167-177.

[24] Sra, Suvrit, Sebastian Nowozin, and Stephen J. Wright, eds. Optimization for machine learning. Mit Press, 2012.

[25] Leardi, Riccardo. "Genetic algorithms in chemometrics and chemistry: a review." Journal of chemometrics 15.7 (2001): 559-569.

[26] Vergidis, K., Ashutosh Tiwari, and Basim Majeed. "Business process improvement using multi-objective optimisation." BT Technology Journal 24.2 (2006): 229-235.

[27] Biethahn, Jörg, and Volker Nissen, eds. Evolutionary algorithms in management applications. Springer Science & Business Media, 2012.

[28] Dasgupta, Dipankar, and Zbigniew Michalewicz, eds. Evolutionary algorithms in engineering applications. Springer Science & Business Media, 2013.

Authors

Seyed Muhammad Hussain Mousavi received his MS.c degree from Bu-Ali Sina University Hamadan-Iran. He received his MS.c degree in Artifactual Intelligence in 2017. His research interests are Evolutionary algorithms, Pattern recognition, Image Processing, Fuzzy Logic, Human computer interactions, Classifications and clustering, Artificial intelligence, RGB_D data, Expert systems, Kinect, Data mining, Facial expression recognition, Face recognition, Age estimation and Gender recognition.

Email: mosavi.a.i.buali@gmail.com

Phone Number: +98-09332892726

View publication stats