DSSO-Directional Shrinking Search Optimization

D P Tripathi¹, U R Jena²
¹Research Scholar, Department of Electronics & Telecommunication Engineering. 
Veer Surendra Sai University of Technology 
Burla-768018, India. 
e-mail: dpt.taru@gmail.com

²Professor, Department of Electronics & Telecommunication Engineering. 
Veer Surendra Sai University of Technology. 
Burla-768018, India. 
e-mail: urjena@rediffmail.com

Abstract: Researchers are continuously trying to develop new heuristic search algorithms in the field of optimization. In this paper, a novel optimization technique has been proposed that tries to maximize the effectiveness of exploration and exploitation by the algorithms in the search of global optimum point. In this proposed algorithm, the points in the search regions are directing towards the global optimum point in the same time the search region is also starts reducing exponentially. A stochastic process has been added in the end of the algorithm to provide better exploitation of the search region. The performance of the algorithm has been compared with some well known optimization techniques and the derived empirical results on solving standard benchmark functions clearly signifies the high performance of the proposed algorithm.

Keywords: Heuristic search Algorithm, Stochastic process, Optimization techniques and Benchmark Functions

1. INTRODUCTION

With the advances in engineering and technology, complexity and sizes of the problems are also taking a toll. Researchers are continuously looking for different methodology for successfully solving these problems. Many numbers of problems are so much complex (NP-hard problems [1]) that traditional algorithms cannot able to solve. Hence researchers are continuously trying hard to develop different algorithms for solving global numerical optimization problems in better ways. Floudas and Gounaris [2] have illustrated recent advances in solving global optimization problems. In last two decades many algorithms have been developed by researchers emulating the behavior of nature. Some of the popular
algorithms derived by emulating nature are Genetic Algorithm (GA) \cite{3}-\cite{8}, Particle Swarm optimization (PSO) \cite{9}-\cite{12}, Differential Evolution algorithm (DE)\cite{13}-\cite{17}, Firefly algorithm (FFA) \cite{18}-\cite{20}, Gravitational search algorithm (GSA) \cite{21}, and lots more. These nature inspired algorithms have their own advantages and disadvantages while finding the global optimum solution in a search space.

All optimization algorithms have to perform two distinct operations every time, when they are finding global optimum solution in problem domain i.e. Exploration and Exploitation. The process of exploration shows how exhaustive search can be done in the search space while exploitation refers to the ability of the algorithm to find best solution in the search space. When algorithm perform exhaustive search in the search space then it consumes more time hence computational time increased. On the other hand if exploitation is more pronounced in a algorithm then chances of trapping in local optimum solution is more. Hence to make a balance between these two criteria’s becomes very essential. Performance of an algorithm can be judged by its accuracy for finding global optimum solution in a less computational time.

In this paper we proposed a stochastic iterative technique of optimization termed as Directional Shrinking Search Optimization (DSSO) algorithm. Here the algorithm starts constricting the size of the search space gradually during exploration process and finally gets a high probable region for exploiting the optimum solution. This makes the algorithm to take less time computationally.

2. PROPOSED ALGORITHM

In most of the optimization problems with the increase in the dimension the search region grows exponentially. Hence classical techniques are not able to provide a practical solution. In this proposed technique, at any instant all randomly generated solution points are made to move towards the best solution observed at that instant. Directional movement of points towards best solution point is controlled by a factor calculated from the unit vectors between respective points and the best solution points. During iteration all points switf towards the best solution point gradually. To facilitate high exploration capability, a stochastic move has been induced to every point through levy flight. After wards the fitness values of the points in the search space indicates how well points are converging towards the better solution. To reduce the computational requirement of the algorithm, the maximum and minimum dimension value of the search points are calculated. These values will act as boundary and shrink the search region keeping maximum possibility of obtaining best solution in that region. In every iteration, the same process of directional movement of points with stochastic effect and shrinking of search region will take place. This process not only facilitates better exploration but also exploiting the best solution in the search space. Hence global optimum solution is found in less computational time.

The algorithm for the proposed optimization technique is given below:

1. Given the objective function $f(x)$ where $x = (x_1, x_2, x_3, ..., x_d)$ \hspace{1cm} (1)
2. Generate N number of points with dimension d in search space randomly. The ith point in the system is define by $x_i = (x_{i1}, x_{i2}, x_{i3}, ..., x_{id})$, where $i$ varies from 1 to N. \hspace{1cm} (2)
3. Compute the fitness value $f(x_i)$ for all $i$ varies from 1 to N
4. Compute the minimum fitness value as $f_{min}$ and its corresponding position value in the population of points. Iteration begins: Till the stopping criteria does not satisfy
6. Calculate the distance between the minima point and other points in the search space
\[ R_{dist,i} = \|x_{min,i} - x_i\|; \text{where}\\min \text{varies from 1 to } N \]
(3)
7. Calculate the position vector directing from the corresponding point towards the minima
\[ \text{PositionVector}_i = [x_{min} - x_i]; \text{where}\\min \text{varies from 1 to } N \]
(4)
8. Calculating the Unit vector directing towards the minima from the corresponding point
\[ \text{UnitVector}_i = \left(\frac{\text{PositionVector}_i}{R_{dist,i}}\right) \]
(5)
9. Then calculate the Magic factor \( f_{act_i} \) given by
\[ f_{act_i} = \sum_{j=1}^{dim} \text{UnitVector}_{i,j} \times \sigma; \text{where}\ \sigma = 1.5 \]
(6)
10. Update the position of points in the population
\[ x_i = f_{act_i} \times x_i + rand \times \text{minima} - x_i; \text{where}\\min \text{varies from 1 to } N \]
(7)
11. Shrinking search region process begins; Maximum and Minimum search region can be explored using equation
\[ \alpha = 0.2; \% \text{Provide a variance to dimensions of the search region} \]
\[ \text{Maximum}_{\text{dim}} = \max\text{value(from updated point of population)} + \alpha \]
\[ \text{Minimum}_{\text{dim}} = \min\text{value(from updated point of population)} - \alpha \]
(8)
During every iteration the points of the population are shifting towards the minima point, the maximum and minimum value of the point from the population of points decides the probable search region.
12. Again calculate the fitness value for all points in the population and update the minimum fitness value using greedy technique.
13. Limiting the population within the new search region by using above boundary condition.
14. Then to provide better exploitation of the minima point in the search region points are stochastically generated around every point using golden expression
\[ \text{goldenexpression} = \sin(rand \times 1/\text{iteration}^2) \]
(10)
The stochastically generated points are define using the expression
\[ x_i = x_i \times \text{goldenexpression}; \text{where}\\min \text{varies from 1 to } N \]
(11)
15. Iteration ends and loop moves towards step-5 and process continue the stopping criteria satisfy.

3. EXPERIMENTAL RESULTS AND DISCUSSION

Our proposed Directional shrinking search optimization (DSSO) algorithm performance is evaluated using some standard benchmark functions discussed in literature. The algorithm was coded in MATLAB (R2012a) and computational experiments were conducted in a Windows 8.1 PC with an Intel(R) core(TM) i3-4000M with 2.4GHz CPU and 4 GB RAM. The benchmark functions used for performance evaluation are shown in details in Table-1.
### Table-I Benchmark Functions Details

| Function                          | Mathematical Representation | Range          | Optima |
|----------------------------------|----------------------------|----------------|--------|
| Sphere Model                     | $f_1(X) = \sum_{i=1}^{\text{dim}} x_i^2$ | $(-100,100)^{\text{dim}}$ | 0      |
| Schwefel’s Problem 2.22          | $f_2(X) = \sum_{i=1}^{\text{dim}} |x_i| + \prod_{i=1}^{\text{dim}} |x_i|$ | $(-10,10)^{\text{dim}}$ | 0      |
| Schwefel’s Problem 1.2           | $f_3(X) = \sum_{i=1}^{\text{dim}} \left( \sum_{j=1}^{i} x_j \right)^2$ | $(-100,100)^{\text{dim}}$ | 0      |
| Schwefel’s Problem 2.21          | $f_4(X) = \max_i \{|x_i|, 1 \leq i \leq d\}$ | $(-100,100)^{\text{dim}}$ | 0      |
| Generalized Rastrigin’s Function | $f_5(X) = \sum_{i=1}^{\text{dim}} [x_i^2 - 10 \cos(2\pi x_i) + 10]$ | $(-5.12, 5.12)^{\text{dim}}$ | 0      |
| Ackley’s Function                | $f_6(X) = -20 \exp \left( -0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{\text{dim}} x_i^2} \right) - \exp \left( \frac{1}{n} \sum_{i=1}^{\text{dim}} \cos(2\pi x_i) \right)$ | $(-32, 32)^{\text{dim}}$ | 0      |
| Generalized Griewank Function    | $f_7(X) = \frac{1}{4000} \sum_{i=1}^{\text{dim}} x_i^2 - \prod_{i=1}^{\text{dim}} \cos \left( \frac{x_i}{\sqrt{i}} \right) + 1$ | $(-600, 600)^{\text{dim}}$ | 0      |

The seven benchmark functions taken are combination of unimodal and multimodal characteristics. The number of point population taken is 50 and each population point is having dimension 30. For each function the algorithm is run for 30 numbers of times. The comparative result shown in Table-II clearly reveals that proposed DSSO algorithm converges to the global optima point for the given test functions.
Table-II Comparative results for RGA,PSO,GSA and DSSO

| Function No. | Measures       | RGA [21]       | PSO [21]       | GSA [21]       | DSSO       |
|-------------|----------------|----------------|----------------|----------------|------------|
| 1           | Average Best   | 23.13          | 1.8 x 10^{-3}  | 7.3 x 10^{-11} | 0          |
|             | Mean Best      | 23.45          | 5.0 x 10^{-2}  | 2.1 x 10^{-10} | 0          |
| 2           | Average Best   | 1.07           | 2.0            | 4.03 x 10^{-5} | 0          |
|             | Mean Best      | 1.07           | 2.0            | 6.9 x 10^{-5}  | 0          |
| 3           | Average Best   | 5.6 x 10^{+3}  | 4.1 x 10^{+3}  | 0.16 x 10^{+3} | 0          |
|             | Mean Best      | 5.6 x 10^{+3}  | 2.9 x 10^{+3}  | 0.16 x 10^{+3} | 0          |
| 4           | Average Best   | 0.06           | 0.04           | 0.018          | 0          |
|             | Mean Best      | 0.56           | 1.04           | 0.533          | 0          |
| 5           | Average Best   | 5.90           | 55.1           | 15.32          | 0          |
|             | Mean Best      | 5.92           | 72.8           | 15.32          | 0          |
| 6           | Average Best   | 2.13           | 9.0 x 10^{-3}  | 6.9 x 10^{-6}  | 8.88 x 10^{-16} |
|             | Mean Best      | 2.15           | 0.02           | 1.1 x 10^{-5}  | 8.88 x 10^{-16} |
| 7           | Average Best   | 1.16           | 0.01           | 0.29           | 0          |
|             | Mean Best      | 1.16           | 0.055          | 0.29           | 0          |

Moreover the rate of convergence of these benchmark functions shown in figures below clearly depicts that the computation time taken by DSSO is very small. It has been observed that within 50 to 100 iterations, all the functions converge to global optimum point in the search space.

Fig-1- Performance curve for F1  
Fig-2-Performance curve for F2
4. CONCLUSION

Recent years have witnessed the development of many heuristic optimization algorithms. Most of these algorithms are inspired from nature, physics etc. The common point among all is the drifting of points using a mathematical expression emulating the some natural phenomena towards global optimum. Here the drifting of point is done through directional vector and in the mean time the reduction of search region improves the exploration and exploitation capability of the DSSO algorithm. Since according the free lunch theorem [22], all optimization problems cannot be converged properly by any particular algorithm. Hence researchers may utilized this algorithm for converging more number of complex optimization functions and hybrid the algorithm with some more optimization algorithm which may lead us to more fruitful results in future application of this algorithm.
REFERENCES

[1]. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP Completeness, W.H. Freeman and Co, New York (1979)

[2]. C. A. Floudas, C. E. Gounaris, A review of recent advances in global optimization, Journal of Global Optimization, springer, DOI 10.1007/s10898-008-9332-8. July 2008

[3]. D. Beasley, D. Bull, R. Martin, An Overview of Genetic Algorithms. Part 2, University of Cardiff, Cardiff, 1993.

[4]. D.B. Fogel, What is evolutionary computation, in: IEEE Spectrum, 2000, pp.26–32.

[5]. D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learn-ing, Addison Wesley, Reading, 1989.

[6]. D. Beasley, D. Bull, R. Martin, An Overview of Genetic Algorithms. Part 1, University of Cardiff, Cardiff, 1993.

[7]. J.H. Holland, Genetic algorithms, Sci. Am. 267 (1992) 66–72.

[8]. M. Mitchell, An Introduction to Genetic Algorithms, MIT Press, Cambridge, MA, 1996.

[9]. J. Kennedy, R.C. Eberhart, Particle swarm optimization, in: Proceedings ofIEEE International Conference on Neural Networks, Piscataway, 1995, pp.1942–1948.

[10]. J. Kennedy, R.C. Eberhart, A Discrete Binary Version of the Particle Swarm Algo-rithm, 1997, pp. 4104–4108.

[11]. O.M. Nezami, A. Bahrampour, P. Jamshidlou, Dynamic Diversity Enhance-ment in Particle Swarm Optimization (DDEPSO) algorithm for pre-venting from premature convergence, Procedia Comput. Sci. 24 (2013)54–65.

[12]. M. Clerc, J. Kennedy, The particle swarm: explosion, stability and conver-gence in multi-dimensional complex space, IEEE Trans. Evol. Comput. 20 (2002)1671–1676.

[13]. P. Bergey, C. Ragsdale, Modified differential evolution: a greedy random strategy for genetic recombination, Omega 33 (2005) 255–265.

[14]. A. Salman, A. Engelbrecht, M. Omran, Empirical analysis of self-adaptive differential evolution, Eur. J. Oper. Res. 183 (2007) 785–804.

[15]. J. Vesterstrom, R. Thomsen, A comparative study of differential evolution parti-cle swarm optimization and evolutionary algorithms on numerical benchmark problems, in: IEEE Congress on Evolutionary Computation (CEC’ 2004), Piscat-away, NJ, 2004, pp. 1980–1987.

[16]. K. Price, R. Storn, Differential evolution – a simple and efficient heuristicfor global optimization over continuous spaces, J. Global Optim. 11 (1997)341–359.

[17]. K. Price, R. Storn, Differential Evolution a Practical Approach to Global Optimiza-tion, Springer Natural Computing Series, 2005.

[18]. X.-S. Yang, Firefly algorithm, stochastic test functions and design optimisation,Int. J. Bio-Inspir. Comput. 2 (2010) 78–84.

[19]. Fister, I., Fister Jr., I., Yang, X.-S., Brest, J.: A comprehensive review of firefly algorithms. Swarm and Evolutionary Computation (2013). Available via ScienceDirect. http://www. science direct.com/science/article/pii/S2210650213000461.

[20]. Fister, I., Yang, X.-S., Brest, J., Fister Jr, I.: Memetic self-adaptive firefly algorithm. In: Yang, X.-S., Xiao, R.Z.C., Gandomi, A.H., Karamanoglu, M. (eds.) Swarm Intelligence and Bio-Inspired Computation: Theory and Applications, pp. 73–102. Elsevier, Amsterdam (2013)
[21]. EsmatRashedi, Hossein Nezamabadi-pour, SaeidSaryazdi, GSA: A Gravitational Search Algorithm, Information Sciences 179 (2009) 2232–2248

[22]. D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Trans. Evol. Comput. 1 (1997) 67–82.

D. P. Tripathi obtained bachelor’s degree in Electronics & Telecommunication Engineering in the year 2002 from Fakir Mohan University, Balasore, India and completed his M.Tech. in the year 2009 in Communication System Engineering from Veer Surendra Sai University of Technology, Burla, Odisha, India. Teaching experience of more than 13 years and currently pursuing research in the field of Computer Vision, Image Compression and Soft Computing.

U. R. Jena received B.Sc. Engg. in 1983 from Sambalpur University, Odisha, India, completed M.Tech. in 1997 from I.I.T. Kharagpur, India and Ph.D. in 2003 from Jadavpur university, Kolkata, India. Currently working as Professor in the Department of Electronics and Telecommunication Engineering, Veer Surendra Sai University of Technology, Burla, Odisha, India. His research interest includes Computer Vision, Image Processing and Soft Computing. He is having an experience of teaching more than 27 years and also having numbers of publications in both national and international journals and conferences. He is a life time Member of ISTE.