International Journal of Engineering & Technology, 7 (2.6) (2018) 88-92

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

Hypercube optimization based global solution in numerical benchmark and sensor localization

S.R. Sujatha [1]*, M. Siddappa [2]

[1] *Research Scholar, Sri Siddhartha Academy of Higher Education [SSAHE], Tumkur, India*
[2] *Professor and Head, Dept. of Computer Science & Egg. Sri Siddhartha Institute of Technology, Tumkur, India*
**Corresponding author E-mail: siddappa.p@gmail.com*

Abstract

An original learning algorithm for solving global numerical optimization problems is proposed. The proposed algorithm is strong stochastic search method which is based on evaluation and optimization of a hypercube and is called the hypercube optimization (HO) algorithm. The hypercube optimization algorithm includes the initialization and evaluation process, and searching space process. The designed HO algorithm is tested on specific benchmark functions. The comparative performance analysis have made against with other approaches of dynamic weight particle swarm optimization and self-adaptive differential evolution algorithm. Convergence characteristics of self-adaptive differential evolution algorithm has deliver the much better functional value in compare to dynamic weight based particle swarm optimization.

Keywords: *Self adaptive differential evolution, Dynamic weight PSO.*

1. Introduction

Finding the globally optimal solutions is one of the basic problem in numerical optimization techniques for high-dimensional functions. Finding the optimum values is also one of the goal in the area of optimization. Optimum values are obtained from learning parameters of the function given in the well-defined domains. The learning algorithms are mainly classified into two categories, derivative based learning algorithm and derivative free learning algorithm. Derivative based learning algorithm based on derivatives of the cost functions and derivative free learning algorithm do not use the derivatives of the cost functions. To obtain the solution of different optimization problems various learning methods have been used .For finding global optimal solutions of the nonlinear problems, derivative based learning techniques are not a proper techniques. These problems are characterized by various parameters and many iterations and arithmetic operations are needed for evaluations of these functions .In practical applications, evaluation of these function is often very expensive and large number of function evaluations might not be very feasible for the proposed scheme is called hypercube optimization algorithm[HO]. This algorithm has been widely used to designing hypercube optimization of the objective function for multivariate systems and selecting the best elements. Best elements are determined during parameter learning process.

2. Related work

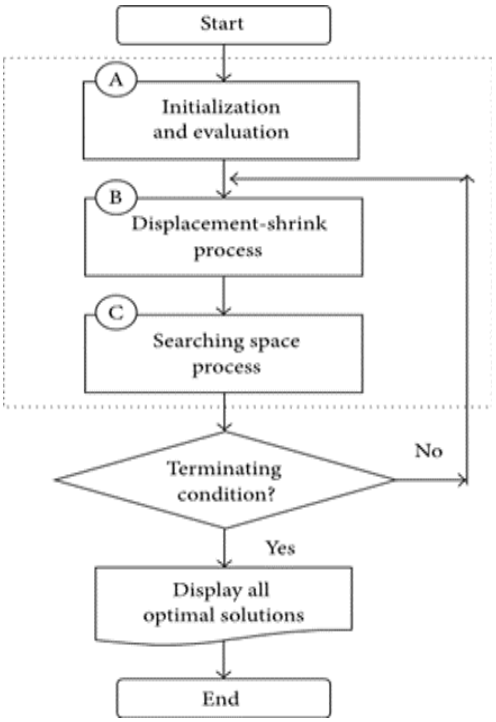
To find global optimal solution, several learning algorithms have been applied. Monte-Carlo method [1], Vegas algorithm [2], and Cat algorithm [3] are widely used to find the solution for different optimization problems. Genetic algorithm(GA) has been proposed in [4,5],[7],particle swarm optimization algorithm proposed in [8], and the non-evolutionary approaches like simulated annealing in [9], tabusearch [10], ant-colony optimization algorithm proposed in [11], and artificial bee colony algorithm in [12]. The combination of the approaches with swarm intelligence methods is widely used to resolve different practical problems of engineering and science [13–18]. Self-adaptive differential evolution as been proposed in [24,25].

2.1. Hypercube Optimization Algorithm

The HO algorithm is an evolutionary algorithm. This algorithm takes motivation from the behavior of a dove searching new areas for food in natural life. . The functional distances between each of two solutions are determined in each iteration. This distance are mainly used to find the next new search area. This is performed using the displacement-shrink process. The HO algorithm working concepts based on derivative-free learning method to estimate the set of points. Once after estimate the set of point shifts and contracts are based on the average between previous best points and new best points inside the hypercube. The contraction is greater when the movement is smaller to accelerate the convergence. At the end of iterations this operation will be informed as an optimal solution.

2.2. Flowchart of the HO algorithm

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



This algorithm mainly consisting of three steps.

The steps are as follows

Step A is initialization and evaluation process.

Step B is displacement-shrink process.

Step C is searching space processes.

Step A: (initialization and evaluation process) this algorithm initially begins with the hypercube generation and set the size of matrices and variables inside the hypercube. Here the hypercube is represented by the center and size (radii). The different points with uniform distribution are arbitrarily produced inside the hypercube. It gates through the main loop, by which merging to the global minimum is sought, and it completes when any of the termination conditions is satisfied.

Step B: (displacement-shrink process). This process is mainly used to find the new best point. This process is designed by computing the average of the previous best point and current best point. The average between both values is taken as a conventional measure to avoid excessive oscillations in the search.

Step C: (searching space process).This process is used limits the movements of solutions within the well-defined interval. A new hypercube is initialized in each iteration and repeats the complete process.

At each iteration, the newly created hypercube alters and contracts its sizes till the optimum points are situated. The set of points in the HO algorithm are not moving according to the exact rule nor does the method record them, excluding the best points. This allows a fast collection of a new best sector and a strong search in it. Thus the HO algorithm is always for global search not for the local search. This attitude allows this algorithm to move quickly to globally best points, not for the local searches.

3. Test Functions

The proposed HO algorithm is verified on three continues test functions which are widely used in all applications like Rosenbrock function, Rastrigin function, Ackley path function [18–22]. The test functions are more applicable for the experimental evaluations of methods used in global optimization problems.

3.1. Rosenbrock Valley Function

This function is scalable, continuous, naturally no convex, non-separable, and unimodal. This function is framed as follows:

$$F_1 = \sum_{I=1}^{M-1} [100(Y_{I+1} - Y_I)^2 + (Y_1 - 1)^2] \quad (1)$$

Where L ≥ 2 is a number of dimensions and Yi = (Y1, Y2… YL) is L dimensional row vector. The test zone is typically assessed in the limits of −2.048 ≤ Y i ≤ 2.048, i = (1, L). Global minimum f(Y) = 0 is obtainable for Yi = (1, 1).

3.2. Rastrigin Function

Rastrigin function is scalable, continuous, and divisible and is extremely multimodal global optimization function.

This function is framed as follows:

$$f_2 = \sum_{i=1}^{M} \left[Y_i^2 - 10cos(2\pi x Y_i) + 10\right] \quad (2)$$

Where L is a number of dimensions and Yi = (Y1, Y2… YL) is L dimensional row vector. The test zone is typically assessed in the limits of −5.12 ≤ Yi ≤ 5.12, i = (1… L). Global minimum f(Y) = 0 is obtainable for Yi = (0, 0).

3.3. Ackley Path Function

Ackley path function is scalable, continuous, and noseparable and is an extensively multimodal function.

This function is framed as follows:

$$f_3 = exp\left(-0.2\sqrt{\frac{1}{M}\sum_{I=1}^{M} Y_i^2}\right) - exp\left(\frac{1}{M}\sum_{I=1}^{M} cos(2\pi Y_i)\right) + 20 + e \quad (3)$$

Where L is a number of dimensions and Yi = (Y1, Y2… YL) is L dimensional row vector. The test zone is typically assessed in the limits of −32 ≤ Yi ≤ 32, i = (1… L). Global minimum f(Y) = 0 is obtainable for Yi = (0, 0).

3.4. Dynamic weight PSO

DWPSO is a population based stochastic optimization method. In discrete search area this algorithm is an effective and efficient global optimizer.

Algorithm: Step1: Selection of initial configuration the centroid of the anchor nodes within the transmission range of the target node is a good initial estimate of the solution. The value of gbest is initialized with the position of the centroid.

$$(x_c, y_c) = \left(\frac{1}{M}\sum_{i=1}^{M} x_i, \frac{1}{M}\sum_{i=1}^{M} y_i,\right) \quad (4)$$

Step 2: The group of' K particles is initialized with the random positions using polar coordinates, with the centroid as the origin. The pbest is initialized with this value of particle positions. Also the initial velocity of all particles is set to zero.

Step 3: For each particle, evaluate the objective function. If I" particle's current objective function value is better than that of pbest, update pbest with the current location of the particle.

Step 4: Update gbest with the particle with best objective function value in the particle population.

Step 5: For each particle, update the velocity and position according to the Equations (5) and (6) respectively.

Vid(k)=wVid(k-1)+C1R1(Pid(t)+Xid(t)) C2R2(Pgd(t)- Xid(t)) (5)

Xid (t+1) =Xid (t) +Vid (t+1) (6)

Step 6: Steps 3-5 are repeated till either the threshold values of objective function or the maximum number of iterations are achieved.

3.5. Self-adaptive differential evolution

DE is a one of the simple evolutionary method that generates new candidate solutions by merging the parent and some other entities of the same population. A candidate substitutes the parent only if it has better fitness. This technique uses mutation operation as a search mechanism and selection operation to direct the search toward the prospective regions in the search space. Self-adapting control parameter mechanism exist which take control of variation on crossover constant CR, weighting factor F. The capacities jrand, jϵ {1, 2, 3, and 4} signifies uniform arbitrary standards in the range [0, 1].To adjust the control parameters F and CR, τ1 and τ2 probabilities are used respectively. 0.1, 0.1, 0.1, 0.9 are fixed values for Constants τ1, τ2, Fl, Fu respectively. The control parameter values are found earlier the mutation process is completed.

Self- Adaptive Differential Evaluation Algorithm
1: choice Np individuals $\vec{x}_{p,q}$ randomly
2: For m=1 to Np let $f_p = f(\vec{x}_{p,,q})$
3: If Np is even then let $l = N_p/2$ else $l = (N_p + 1)/2$
4: while (convergence criterion not yet met) do steps 5 to 15
5: For p=1 to Np do steps 6, 7
6: For J=i+1 to Np do steps 7
7: If $f_{j>}f_p$ then swap $\vec{x}_{p,q}, \vec{x}_{j,q}$ and swap f_j, f_p
8: For p=1 to l do steps 9 to 14
9: select three different random indexes $r_0, r_1\ and\ r_2$ between 1 to Np≠ $r_0 \neq r_1 \neq r_2$.
10: Let $\vec{V}_{p.q} = \vec{X}_{r0,q} + F(\vec{X}_{r1,q} - \vec{X}_{r2,q})$
11: For j=1 to n do steps 12 to 13
12: select randomly r_j variable $(0 \leq r_j < 1)$ and j_{rand} index $(1 \leq j_{rand} \leq l)$
13: If $r_j \leq C_r$ or $j = j_{rand}$ then $u_{j,p,q} = V_{j,p,q}$ else $u_{j,p,q} = X_{j,p,q}$
14: If $f(\vec{U}_{p,q}) \leq f_p$ then $\vec{X}_{p,q+1} = \vec{U}_{j,q}$; $f_p = f(\vec{U}_{p,q})$ else $\vec{X}_{p,q+1} = \vec{X}_{p,q}$
15: For i=p+1 to Np let $\vec{X}_{p,q+1} = p_{i,q}$

4. Experimental analysis

Numerical Bench Mark

Three different numerical benchmark problem as shown in Table1, have taken to get the relative benefits between DWPSO and SADDE. These functions have to minimize i.e. achieve the mini-mum value and involved number of dimension is 10.In all case of DWPSO and SADDE, there is population size of 50 have taken and 500 iterations have allowed. For each function 10 independent trails have given and mean value and their standard deviation along with best in terms of minimum value of function have presented in Table2 .Convergence characteristics for all the cases have also shown in Fig.1 to Fig.6.It is clear that SADDE has deliver the much better function value in compare to DWPSO, not only that there is nearly same convergence characteristic appeared for all 10 trails while lots of disperse appeared in case of DWPSO. This can be observed with low value of standard deviation in case of SADDE and high value in case of DWPSO.

Table 1: Numerical bench mark functions to achieve the minimum value

Function M	Range	fmin
$f_1 = \sum_{i=1}^{M-1} [100(Y_{i+1} - Y_i)^2 + (Y_i - 1)^2]$	10 (-30,30)M	0
$f_2 = \sum_{I=1}^{M} [Y_i^2 - 10cos(2\pi Y_i) + 10]$	10(5.12,5.12) M	0
$f_3 = exp\left(-0.2\sqrt{\frac{1}{N}\sum_{i=1}^{M} Y_i^2}\right) - exp\left(\frac{1}{N}\sum_{i=1}^{M} cos(2\pi Y_i)\right) + 20 + e$	10 (-32,32) M	0

Table 2: Performance analysis for all the function in 10 independent trials

	DWPSO Mean Std.Dev Best	SADDE Mean Std.Dev Best
F1(Generalized Rosenbrock's Function)	358.19 475.88 4.86	3.11 1.27 0.57
F2(Generaize Rastrigin's Function)	22.7962 12.1493 5.9698	4.12e-009 7.22e-009 2.82e-011
F3(Ackley's Function)	0.87 0.93 1.04e -004	1.04e-009 4.71e-010 4.07e-010

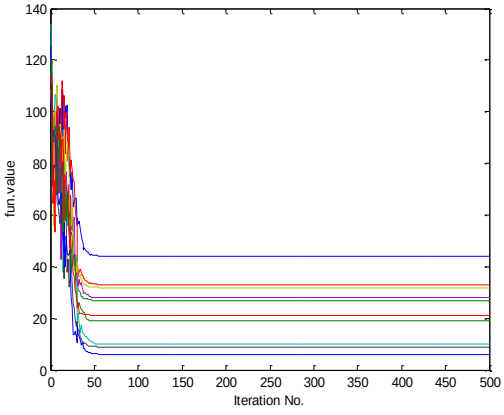


Fig. 1: Convergence characteristics for 10 independent trails for Generalized Rosenbrock'S function in DWPSO

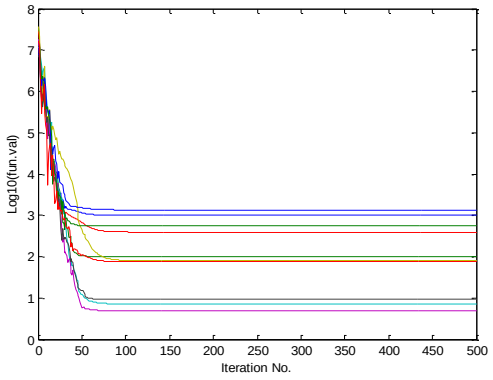


Fig. 2: Convergence characteristics for 10 independent trails for Generalized Rosenbrock'S function in SADDE

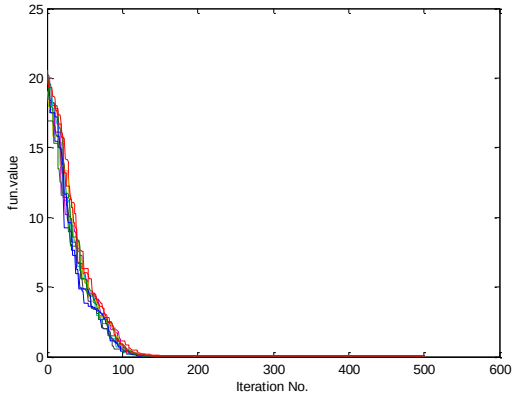


Fig. 5: Convergence characteristics for 10 independent trails for Ackley's function in DWPSO

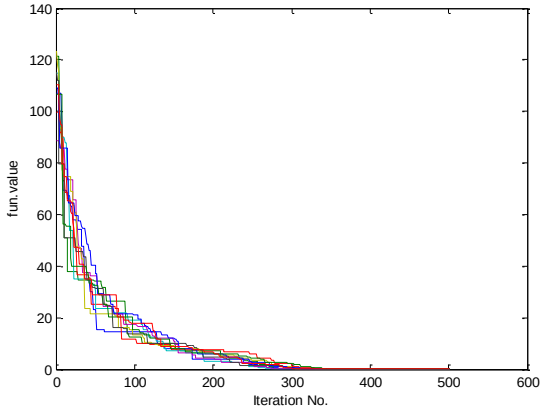


Fig. 3: Convergence characteristics for 10 independent trails for Generalized Rastrigin's function in DWPSO

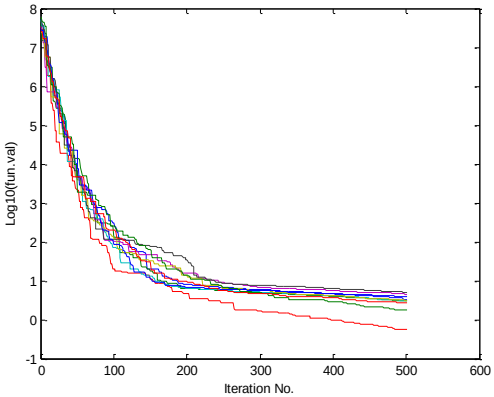


Fig. 6: Convergence characteristics for 10 independent trails for Ackley's function in SADDE

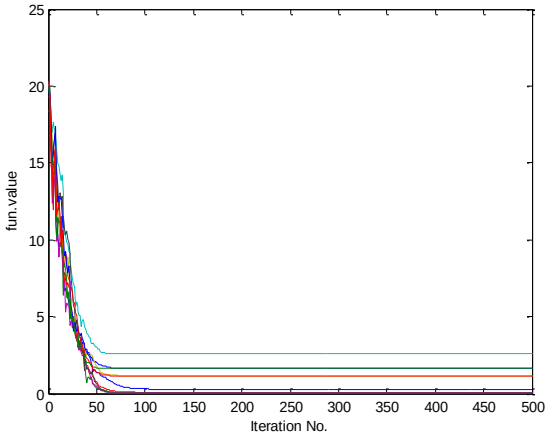


Fig. 4: Convergence characteristics for 10 independent trails for Generalized Rastrigin's function in SADDE

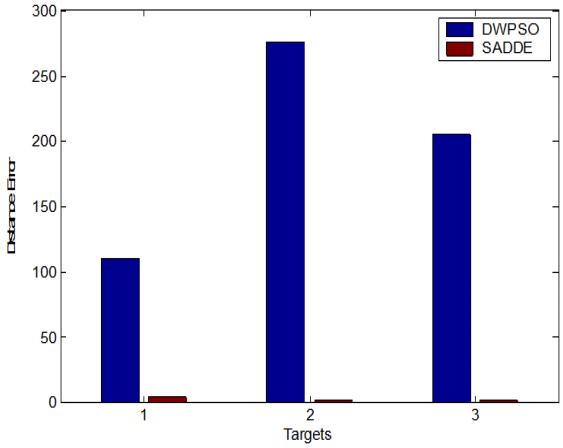


Fig. 7: Distance error comparison in Localization of three targets between DWPSO & SADDE

5. Conclusion

The hypercube optimization algorithm has yielded in general quite better results, sometimes reaching the better solution faster than well-established algorithms. The usage of multi objective evolutionary algorithms allows us to find global optimal solutions and avoid local optimum problem. Whole experiment setup has designed in MATLAB environment. For each target position estimation 100 iterations have allowed and error as estimated .It is clear that there is very less error occurred in estimation of position by SADDE in compare to DWPSO. There is more fluctuation is also observed in performance of DWPSO which indicate trends of improvements is not very regular in DWPSO while in case of SADDE, almost there is decreasing tendency with iterations. In

our future work, we will focus on solving some real-world problems with the SADDE. We will also tend to modify it for discrete problems

References

[1] G. S. Fishman, Monte Carlo: Concepts, Algorithms, and Applications, *Springer*, New York, NY, USA, 1995.

[2] G. P. Lepage, "A new algorithm for adaptive multidi-mensional integration," *Journal of Computational Physics*, vol. 27, no. 2, pp. 192–203, 1978. View at Publisher • View at Google Scholar • View at Scopus.

[3] D. J. Weiss and G. G. Kingsbury, "Application of com-puterized adaptive testing to educational problems," *Journal of Educational Measurement*, vol. 21, pp. 361–375, 1984. View at Google Scholar.

[4] D. E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley Publishing, 1989.

[5] J. H. Holland, "Adaptation in Natural and Artificial Systems, University of Michigan Press", Ann Arbor, Mich, USA, 1975. View at MathSciNet.

[6] K. Deb, A. Anand, and D. Joshi, "A computationally efficient evolutionary algorithm for real-parameter optimization," Evolutionary Computation, Vol. 10, No. 4, pp. 371–395, 2002. View at Publisher • View at Google Scholar • View at Scopus.

[7] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, Vol. 11, no. 4, pp. 341–359, 1997. View at Publisher • View at Google Scholar • View at MathSciNet.

[8] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in Proceedings of the *IEEE International Conference on Neural Networks*, Vol. 4, pp. 1942–1948, IEEE, Perth, Australia, December 1995. View at Publisher • View at Google Scholar • View at Scopus.

[9] S. Kirkpatrick, J. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," Science, Vol. 220, No. 4598, pp. 671–680, 1983. View at Publisher • View at Google Scholar • View at MathSciNet.

[10] F. Glover and M. Laguna, Tabu Search, Kluwer Academic Publishers, Boston, Mass, USA, 1997.

[11] D. Parabola, "Artificial bee colony algorithm," Scholar-pedia, Vol. 5, No. 3, article 6915, 2010. View at Publisher • View at Google Scholar.

[12] R. H. Abiyev and M. Menekay, "Fuzzy portfolio selec-tion using genetic algorithm," Soft Computing, Vol. 11, No. 12, pp. 1157–1163, 2007. View at Publisher • View at Google Scholar • View at Scopus.

[13] R. H. Abiyev, "Fuzzy wavelet neural network for pre-diction of electricity consumption", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*: AIEDAM, Vol. 23, No. 2, pp. 109–118, 2009. View at Pub-lisher • View at Google Scholar • View at Scopus .

[14] T. Vidal, T. G. Crainic, M. Gendreau, N. Lahr chi, and W. Rei, "A hybrid genetic algorithm for multidepot and periodic vehicle routing problems," Operations Research, Vol. 60, No. 3, pp. 611–624, 2012. View at Publisher • View at Google Scholar • View at MathSciNet • View at Scopus.

[15] X. Yang, C.-B. Hu, K.-X. Peng, and C.-N. Tong, "Load distribution of evolutionary algorithm for complex-process optimization based on differential evolutionary strategy in hot rolling process," *Mathematical Problems in Engineering*, Vol. 2013, Article ID 675381, 8 pages, 2013. View at Publisher • View at Google Scholar • View at Scopus.

[16] L. Zhang, M. Zhang, W. Yang, and D. Dong, "Golden ratio genetic algorithm based approach for modeling and analysis of the capacity expansion of urban road traffic network," Computational Intelligence and Neuroscience, Vol. 2015, Article ID 512715, 9 pages, 2015. View at Publisher • View at Google Scholar

[17] J.-T. Tsai, J.-H. Chou, and W.-H. Ho, "Improved quantum-inspired evolutionary algorithm for engineering design optimization," Mathematical Problems in Engineering, Vol. 2012, Article ID 836597, 27 pages, 2012. View at Publisher • View at Google Scholar • View at MathSciNet.

[18] C. Grosan and A. Abraham, "A novel global optimization technique for high dimensional functions," International Journal of Intelligent Systems, Vol. 24, No. 4, pp. 421–440, 2009. View at Publisher • View at Google Scholar • View at Scopus.

[19] Z. Yang, K. Tang, and X. Yao, "Differential evolution for high-dimensional function optimization," in Proceedings of the *IEEE Congress on Evolutionary Computation (CEC '07)*, pp. 3523–3530, IEEE, Singapore, September 2007. View at Publisher • View at Google Scholar • View at Scopus.

[20] X. You, "Differential evolution with a new mutation operator for solving high dimensional continuous optimization problems," *Journal of Computational Information Systems*, Vol. 6, No. 9, pp. 3033–3039, 2010. View at Google Scholar • View at Scopus.

[21] J. Brest, A. Zamuda, B. Bošković, M. S. Maučec, and V. Žumer, "High-dimensional real-parameter optimization using self-adaptive differential evolution algorithm with population size reduction," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '08)*, pp. 2032–2039, Hong Kong, June 2008. View at Publisher • View at Google Scholar • View at Scopus.

[22] P. Korosec and J. Silc, "High-dimensional real-parameter optimization using the differential ant-stigmergy algorithm," *International Journal of Intelligent Computing and Cybernetics*, Vol. 2, No. 1, pp. 34–51, 2009. View at Publisher • View at Google Scholar • View at MathSciNet • View at Scopus.

[23] Rastko R. Selmic, Vir V. Phoha, Abdul Serwadda, "Localization and Tracking in WSNs", *Wireless Sensor Networks*, pp. 155-177, Date: 03 November 2016.

[24] Sujatha S R and, M.Siddappa, "Node Localization Method for Wireless Sensor Networks Based on Hybrid Optimization of Particle Swarm Optimization and Differential Evolution", IOSR *Journal of Computer Engineering (IOSR-JCE)* e-ISSN: 2278-0661,p-ISSN: 2278-8727, Vol. 19, Issue 2, Ver. III (Mar.-Apr. 2017), PP 07-12 www.iosrjournals.org.

[25] Sujatha S R and M.Siddappa, "Self-Adaptive Differential Evolution based Localization of Sensors in Wireless Sensor Network" *International Journal of Applied Engineering Research* ISSN 0973-4562 Volume 13, Number 1 (2018) pp. 86-90 © Research India Publications. http://www.ripublication.com- View at Scopus.