INTERNATIONAL JOURNAL OF ADVANCED STUDIES IN COMPUTER SCIENCE AND ENGINEERING
IJASCSE VOLUME 5 ISSUE 11, 2016
11/30/2016

Towards Adaptive Hybrid Optimization Technique

Marwah Alian
Princess Sumaya University For Technology
marwah2001@yahoo.com

Amjad Hudaibline
Princess Sumaya University For Technology
a.hudaib@psut.edu.jo

Dima Suleiman
Princess Sumaya University For Technology
Dimah_1999@yahoo.com

***Abstract*— Differential Evolution (DE) Continuous optimization is one of the most important activities in heuristic optimization. Enhancement on algorithms can be achieved using hybrid model by combining different search strategies. The proposed algorithm combines both LSHADE with mutation of Invasive Weed Optimization (IWO) on parameter F. the algorithm try to take the search process away from stagnation point using IWO distribution. The proposed algorithm tested using benchmark functions of Congress on Evolutionary Computation (CEC) 2014 and it gives competitive results.**

Keywords- Optimization; Differential evolution; Invasive Weed Optimizer.

I. INTRODUCTION

Due to population based approach for Evolutionary algorithms, they provide advantage over classical optimization methods. They maintain possible solutions as a population, and they are processed in each generation, and if the multiple solutions can be preserved over all these generations, then at the end of the algorithm there will be multiple good solutions, instead of only the best solution.[9]

Researchers propose many algorithms and compare it on several benchmarks functions [15][18], with different performance depending on the problems. It is supposed that a trial for combining different search strategies to be desirable to obtain the best performance of each of these techniques. Y.Zhou et al. [11] present DEIWO which is a hybrid algorithm for solving nonlinear equation systems based on an invasive weed optimization (IWO) algorithm and the differential evolution (DE) algorithm. Where, the mutation and crossover of DE is used with IWO technique to give better results in search space.

Selecting an appropriate algorithm to solve a continuous optimization problem is a critical task. Since an algorithm can be configured to perform in a proper manner in a given range of problems and considering their dimensionality, it can behave in a degrade way as this dimensionality increases. [6]

However, Subhrajit et al. [9] propose a hybrid two-stage optimization technique that combines Invasive Weed Optimization (IWO), with modified Group Search Optimizer (GSO). In this technique IWO is run for 80% of the total amount of function evaluations while GSO is invoked in each sub-region discovered with IWO, and it runs for 20% of amount of function evaluation. They modify both IWO and GSO to be used in the multimodal problems used in their work.

The proposed approach belongs to hybrid techniques such that, a tuning in the mutation factor F in LSHADE is used according to IWO varying distribution of seeds. Where In IWO, weeds represent feasible solutions and the set of all weeds represent population [8]. A finite number of weeds are being dispread over the search area by normally distributed random numbers with a varying variance. This variance is used with mutation factor F in LSHADE. The proposed algorithm is tested using benchmark functions for CEC2014.

This research is organized as follows; section 2 is a review of related work in the field of optimization techniques especially differential evolution and hybrid techniques. Section 3 is a representation for the proposed algorithm, and section 4 experiment and results are discussed. Finally conclusion is in section 5.

II. RELATED WORK

One approach for handling optimization problems is Differential Evolution (DE) which is simple and effective.Recent researches have been focusing on search efficiency and optimization performance.[7]

Differential Evolution (DE) which is proposed by R. Storn and K. Price is a parallel direct search method that utilizes NP parameter vectors. DE main idea is a scheme for generating trial parameter vectors by adding a weighted difference vector between two population individuals to a third individual. If the resulting vector has a lower objective function value than a predetermined population individual, the predecessor vector will be replaced in next generation.[1]

Several variants of DE have been proposed to achieve better optimization performance and to develop selfadaptive mechanisms [4][5][15]. For example, J. Zhang and A. C. Sanderson propose a better algorithm than other classic or adaptive DE algorithms and call it JADE. In this algorithm a new mutation strategy "DE/current-to-pbest" is implemented and an optional external archive is used with adaptive update for control parameters. [2]

R. Tanabe and A. Fukunaga aim to improve upon the robustness of JADE, by proposing SHADE, which differs from JADE in the mechanism of parameter adaptation that is based on a historical record of successful parameter settings. SHADE maintains a diverse set of parameters to control parameter adaptation as search progresses by storing the mean values of crossover and mutation parameters (CR, F) for each generation while JADE uses a single pair (CR, F) in the process of parameter adaptation. [10]

In 2014 R. Tanabe and A. Fukunaga extends Success-History based Adaptive DE (SHADE) algorithm with Linear Population Size Reduction (LPSR), which decreases the population size in a continuous manner according to a linear function. This algorithm is mentioned as Success-History based Adaptive DE with Linear Population Size Reduction (LSHADE) [12]. While Awad et al. propose new Differential Evolution algorithm as an enhancement of JADE algorithm. They introduce a new technique that uses a memory-based structure of previous successful settings. This algorithm tries to adapt all the parameters of DE algorithm; first adaptation is for the population size is used in order to find the most suitable size that guide the search in each optimization iteration. The second adaptation is for the control parameters F, CR that adapted by storing parameters used with successful individuals. [17]

In 2013, Y.Zhou et al. present Differential Evolution Invasive Weed Optimization algorithm (DEIWO). It is a hybrid algorithm for solving nonlinear equation systems based on an invasive weed optimization (IWO) algorithm and the differential evolution (DE) algorithm. Where, the global exploration of invasive weed optimization algorithm provides an efficient search area for differential evolution and the heuristic search capability of differential evolution algorithm provides a reliable guide for invasive weed optimization. Their results show that the proposed algorithm is an efficient algorithm for solving nonlinear systems of equations.[11]

Muthana [13] utilize the Invasive Weeds optimization to produce the array radiation pattern that is near to the desired objective that represent side lobe level (SLL) suppression and null placement. IWO method is used as an adaptive beam former that makes a uniform linear antenna array lead the main lobe towards the Direction of Arrival (DoA).

In an optimization framework like Evolutionary algorithms (EA), a proper number of function evaluations is required. It is required to locate an optimal solution for computationally real-world optimization problems. Surrogate models are integrated with EA as a technique to solve complex multimodal problems within a limited number of function evaluations [16].

Miruna et al. propose a Diversity Controlled Parameter adapted Differential Evolution with Local Search algorithm (DCPaDE-LS) which is an improved algorithm integrated with two dynamic surrogate models and two variants. The two variants are, Surrogate Assisted Parameter adapted Differential Evolution with Artificial Neural Networks and Response Surface Methodology. In this work they show that the variants are able to reduce the number of function evaluation without loss in success rate for all the functions [16].

III. THE PROPOSED ALGORITHM

In this research, a hybrid algorithm is proposed. This algorithm combines the benefit of two heuristic search methods; LSHADE and IWO distribution. Where LSHADE uses a success-history based adaptation which is a novel mechanism for parameter adaptation based on a historical memory of successful parameter settings that were found during the previous runs also it uses a simple deterministic population resizing method which continuously reduces the population size, those properties improves the performance of LSHADE over other DE family [12].

Figure 1 represents the main steps for the proposed algorithm, it starts by initialization of a population, then LSHADE starts to produce several generations with new solutions. When the solutions are repeated for 10 generations then IWO distribution is used with the mutation factor F and it helps the search to move a new promising region.

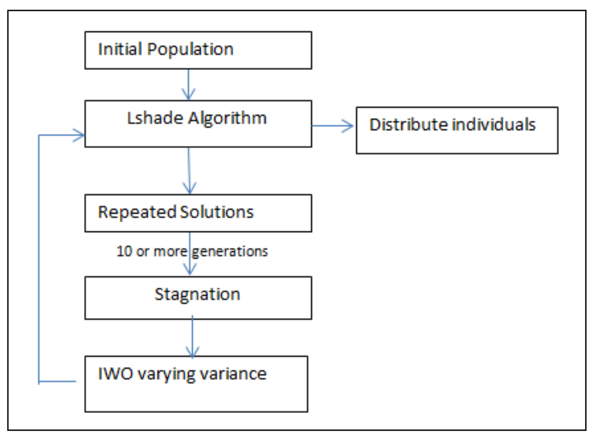

Figure 1. Main steps for the proposed Algorithm

The pseudo code for this algorithm is shown in Figure 2 where the first step in the proposed algorithm is to initialize the population $P = (x_{1,G}, ..., x_{N,G})$ randomly then for each parameter vector $x_{i,G}$ in generation G a trial vector $u_{i,G}$ is generated according to current-to-pbest/1/bin mutation strategy using equation (1) as in JADE[3] , and the mutation factor F is randomly selected between 0 and σ. After that, Binomial crossover is used as in DE [1] then a selection process is going to take palace by selecting individual $x_{i,G+1}$ for next generation. The selected individual is the one with highest fitness, the success mutation factor F and the crossover rate Cr are going to be used for next generation. We check if the difference between solutions of generation G_i and next generation G_{i+1} less or equal to 1×10-6 then we compute standard deviation σ using IWO variance distribution using equation (2). If the algorithm did not enter stagnation then we compute σ as in LSHADE. An archive is used for the unselected individuals that have less fitness value and the historical memory is used for saving F and CR that were used with the success individuals. Then linear reduction strategy for population size is used as in LSHADE.

Initialize population P = ($x_{1,G}$, ..., $x_{N,G}$) randomly;
Set all values in parameters F,CR in the memory to 0.5;
While (*termination criteria not met)* do
  Initialize the success set of CR and F to ∅,
  ri randomly Selected from [1,H];
  compute F, CR for generation G randomly;
  Generate trial vector $u_{i,G}$ according to current-to-pbest/1/bin as in JADE;
  Mutate the individual using DE mutation
  compute Fi randomly between 0, σ
  Use Binomial crossover as in DE
  Select individual $x_{i,G+1}$ for next generation which has highest fitness
  Use success F and Cr for next generation
  if difference between solutions of G_i and G_{i+1} less or equal to 1×10-6 then
  compute standard deviation σ using IWO
    $\sigma = ((iter_{max} - iter)^n / (iter_{max})^n)(\sigma_{init} - \sigma_{final}) + \sigma_{final}$
  Else compute sigma as in LSHADE
  If the archive size exceed |A| delete randomly selected individuals from the archive.
  Update memory for CR and F
  Apply Linear population reduction strategy as in LSHADE
END While

Figure 2. The proposed algorithm

From Invasive weed optimization (IWO), we benefit from the way that weeds randomly distributed over the search space using standard deviation calculation by normally distributed random numbers. In LSHADE algorithm the mutation process is very sensitive to the adaptation of the mutation factor (F). The mutant vector is generated as in equation (1) according to the mutation of JADE [3]:

$$v_{i,G} = x_{i,G} + Fi \cdot (x_{pbest,G} - x_{i,G}) + Fi \cdot (x_{r1,G} - x_{r2,G}) \quad (1)$$

For each parameter vector $x_{i,G}$, the mutant vector $v_{i,G}$ is generated according to equation (1), and xpbest is randomly selected as one of the top 100 p% individuals in the current population with $p \in (0, 1]$, and Fi is the mutation factor. The indices r1, r2 are distinct random numbers selected from [1,N] as well as i, where N is the number of individuals in the population.

In order to benefit from the IWO method and to improve LSHADE, we try to check when stagnation occurs in LSHADE. Stagnation means that the solutions reach local optimum and stuck there. It can be detected for a number of generations that the solution is not changed, In this case we can alternate the way of calculation of F parameter with varying variance σ used in IWO [2] and computed as in equation (2):

$$\sigma_{iter} = [(iter_{max} - iter)/ (iter_{max})^n]\times(\sigma_{init} - \sigma_{final}) + \sigma_{final} \quad (2)$$

Where σ_{init} is σ initial value for standard deviation, and σfinal is σ_{final} value, $iter_{max}$ is the maximum number of iterations, σiter is the Standard deviation at the present time and n is the nonlinear modulation [2].
As the mutant vector changes its position according to the new calculation of mutant factor (F) then it will move the search process towards global optimum.

IV. EXPEREMENT AND RESULTS

Two variants are handled in our experiment; first one is the experiment of using IWO distribution in the mutation of LSHADE through all runs. The second one, is to alternate from LSHADE to IWO if a stagnation problem occurs. If the solution of a number of sequence generations is the same or it is not changed then IWO distribution is used as an adaptation of parameter F.

The error value that is used in this experiment is 1e-7 in order to decide if there is a visual change between generations. The algorithm has been run on a PC with a 2.1 GHz processor, 8G RAM, and windows XP. The set of real world set instances introduced in CEC2014 are used in the experiment where the details of these problems are presented in [14].

A total of 30 optimization functions have been considered for this experiment. The results reported for this work are the average of 50 independent run for each function, two dimensions have been tested: D=50 and D=100 with a maximum number of function evaluations equals to 10,000*dimension. Table I and Table II contain the average error, for each function with dimension size 50 and 100 of the proposed algorithm with its two variants.

TABLE I. EXPERIMENT RESULTS FOR VARIANT ONE

Fun#	Variant1 D=50	Difference	P-Value	Variant1 D=100	Difference	P-Value
F1	3.44507680E+03	-2.768E+03	0.000	2.72096824E+05	-1.209E+05	0.000
F2	0.00000000E+00	0.000E+00	0.000	3.75582320E-07	-3.756E-07	0.000
F3	0.00000000E+00	0.000E+00	0.000	4.63678455E-10	-4.637E-10	0.315
F4	4.26102597E+01	-2.653E+00	0.771	1.54808857E+02	7.434E+00	0.213
F5	2.03844771E+01	-1.297E-01	0.000	2.06718514E+01	-1.205E-01	0.000
F6	4.93680994E-01	-1.982E-01	0.118	4.93680994E-01	9.284E+00	0.000
F7	0.00000000E+00	0.000E+00	0.000	0.00000000E+00	1.450E-04	0.320
F8	0.00000000E+00	0.000E+00	0.000	2.90922056E-03	-1.586E-03	0.000
F9	2.04875230E+01	-8.861E+00	0.000	7.80102567E+01	-3.971E+01	0.000
F10	1.12728946E-02	3.341E-02	0.000	7.08450616E+00	1.248E+01	0.000
F11	4.10050541E+03	-8.462E+02	0.000	1.30317423E+04	-2.239E+03	0.000
F12	3.52713235E-01	-1.289E-01	0.000	3.96194283E-01	1.922E-02	0.021
F13	2.04634472E-01	-4.208E-02	0.000	2.84308306E-01	-4.607E-02	0.000
F14	6.51352387E+00	-6.200E+00	0.000	3.23087438E-01	-1.148E-01	0.000
F15	2.93941004E-01	4.754E+00	0.000	2.01921805E+01	-4.594E+00	0.000
F16	1.72298643E+01	-2.907E-01	0.000	3.97791393E+01	-5.047E-01	0.000
F17	1.54667519E+03	-2.947E+00	0.974	4.50796034E+03	-5.524E+01	0.685
F18	9.83620158E+01	4.580E-01	0.872	2.17095288E+02	3.621E+00	0.224
F19	8.86603681E+00	-4.757E-01	0.191	9.71617970E+01	5.703E+01	0.000
F20	1.44821501E+01	-1.901E+00	0.036	1.77082861E+02	-3.316E+01	0.002
F21	4.82045396E+02	5.771E+00	0.837	2.22968534E+03	1.476E+01	0.887
F22	2.40360611E+02	-1.267E+02	0.000	1.47468904E+03	-4.522E+02	0.000
F23	3.44004501E+02	0.000E+00	1.000	3.48234959E+02	0.000E+00	1.000
F24	2.75248317E+02	-2.564E-01	0.156	3.92454873E+02	2.013E+00	0.000
F25	2.05171084E+02	1.664E-01	0.004	2.00000000E+02	0.000E+00	1.000
F26	1.00207406E+02	-3.498E-02	0.000	2.00000000E+02	0.000E+00	1.000
F27	3.29163602E+02	2.062E+00	0.731	4.20638521E+02	-2.433E+01	0.000
F28	1.10382820E+03	-9.778E-01	0.880	2.20512647E3,	3.470E+01	0.003
F29	8.06137658E+02	2.069E+01	0.009	2.20179485E+03	-2.505E+01	0.038
F30	8.91565789E+03	-1.809E+02	0.085	8.10346906E+03	3.653E+02	0.051

TABLE II. EXPERIMENT RESULTS FOR VARIANT TWO

Fun#	Variant2 D=50	Difference	P-Value	Variant2 D=100	Difference	P-Value
F1	6.85762694E+02	-8.801E+00	0.974	1.75045683E+05	-3.984E+03	0.865
F2	0.00000000E+00	0.000E+00	0.000	0.00000000E+00	-1.750E+05	0.000
F3	0.00000000E+00	0.000E+00	0.000	0.00000000E+00	0.000E+00	0.000
F4	5.31649026E+01	-1.321E+01	0.159	1.64947194E+02	1.622E+02	0.000
F5	2.02380498E+01	1.675E-02	0.003	2.05250182E+01	-1.444E+02	0.000
F6	2.24466989E-01	7.101E-02	0.487	2.24466989E-1	-1.102E+01	0.000

F7	0.00000000E+00	0.000E+00	0.000	1.45020399E-04	-8.983E+00	0.000
F8	0.00000000E+00	0.000E+00	0.000	2.08239268E-04	1.178E-03	0.000
F9	2.68713303E+01	-1.524E+01	0.000	6.74792944E+01	3.830E+01	0.000
F10	2.02293289E-02	2.445E-02	0.000	2.57620239E+00	-4.792E+01	0.000
F11	3.29157870E+03	-3.732E+01	0.541	1.08357687E+04	1.079E+04	0.000
F12	1.97403139E-01	2.638E-02	0.000	3.96194283E-01	-1.084E+04	0.000
F13	2.08314463E-01	-4.576E-02	0.000	2.80242686E-01	-1.580E-01	0.000
F14	2.83820013E-01	2.971E-02	0.000	3.26326747E-01	-7.196E-02	0.000
F15	5.46045854E+00	-4.122E-01	0.000	1.69327724E+01	1.527E+01	0.000
F16	1.72307254E+01	-2.916E-01	0.000	3.94337331E+01	2.234E+01	0.000
F17	1.46936565E+03	7.436E+01	0.408	4.49573379E+03	4.413E+03	0.000
F18	9.71617970E+01	-1.363E+00	0.641	2.17961766E+02	-4.275E+03	0.000
F19	9.14610369E+00	-7.557E-01	0.045	9.71617970E+01	-1.215E+02	0.000
F20	1.28016073E+01	-2.203E-01	0.789	1.42359799E+02	4.676E+01	0.000
F21	4.95571051E+02	-7.755E+00	0.782	2.24823709E+03	2.102E+03	0.000
F22	1.59321137E+02	-4.568E+01	0.005	1.10273247E+03	-1.226E+03	0.000
F23	3.44004501E+02	0.000E+00	1.000	3.48230000E+02	-7.545E+02	0.000
F24	2.75101558E+02	-1.096E-01	0.547	3.93338052E+02	4.624E+01	0.000
F25	2.05330294E+02	7.146E-03	0.916	2.00000000E+02	-1.933E+02	0.000
F26	1.00190960E+02	-5.143E-02	0.992	2.00000000E+02	0.000E+00	1.000
F27	3.24230510E+02	-2.871E+00	0.546	4.12064145E+02	1.877E+02	0.000
F28	1.11753261E+03	1.273E+01	0.051	2.20681036E+03	1.821E+03	0.000
F29	7.92872865E+02	7.430E+00	0.322	7.90171731E+02	-1.440E+03	0.000
F30	8.97862293E+03	-1.180E+02	0.075	7.87150954E+03	6.035E+03	0.000

For the purposes of analysis, t-test is used after running the two variants of the proposed algorithm on benchmark functions. The second variant shows significantly better results than LSHADE on dimension 50 for functions: F5, F6, F10, F12, F14, F17, F25, F28, and F29, while the results of F1, F2, F3, F4, F7, F8, F11, F17, F18 F20, F21, F23, F26, F27, and F30 are similar to LSHADE. However, the first variant gives similarity to LSHADE on dimension 50 for functions F2, F3, F4, F6, F7, F8, F17, F18, F19, F21, F23, F24, F27, F28, and F30 while it gives better results for F10, F15, F25, and F29.

On dimension 100, the second variant shows significantly better results for function F4, F8, F9, F11, F15, F16, F17, F20, F24, F27, F28, and F30. Also, it has similar results for functions F1, F3, and F29. While the first variant, on dimension 100 gives similarity to the results of LSHADE in F2, F3, F7, F25, and F26 but it produces better results for F10, F24, and F28. So, the overall results that are achieved by the second variant are better than the first variant.

Figure 3 demonstrates the results of the two variants and the difference between them on dimension 50. The first variant when using IWO distribution all the time and the second when alternating between IWO distribution and the LSHADE adaptation for F parameter, as shown in the figure variant two gives better results than variant one.

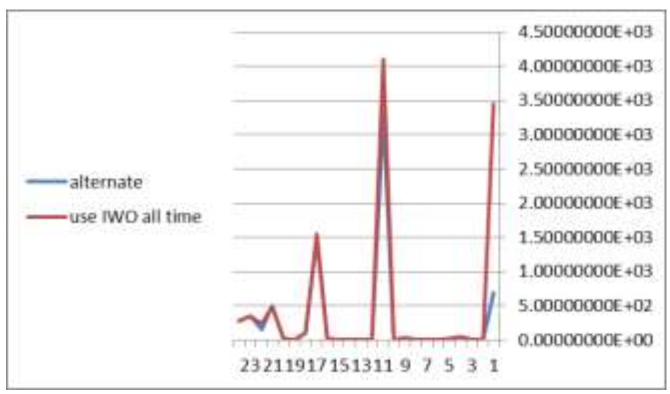

Figure 3. The results of the two variants with D=50.

V. Conclusion

In this paper, a new hybrid algorithm for solving nonlinear equation systems is presented based on the distribution of IWO used with LSHADE algorithm. Two variants are obtained after running the proposed approach on 30 benchmark functions from the CEC2014 Real-Parameter Single Objective Optimization benchmark suite. The results obtained by the proposed approach are optimistic. In optimization process, IWO is useful to bring the search out of stagnation region which happens when the solutions of several generations are centered on local optimum. This approach will guide the search process to global optimum.

REFERENCES

[1] 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.

[2] A.R. Mehrabian,C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization", Ecological informatics, 1-2006, pp.355-366, science direct, Elsevier 2006

[3] J. Zhang and A. C. Sanderson, "JADE: Adaptive Differential Evolution With Optional External Archive," IEEE Tran. Evolutionary Computation, vol. 13, no. 5, 2009, pp. 945–958.

[4] A.K. Qin, V.L. Huang, P.N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization", IEEE Trans. Evol.Comput. 13, pp.398–417, 2009.

[5] R. Mallipeddi, P.N. Suganthan, "Differential evolution algorithm with ensemble of parameters and mutation and crossover strategies", in: B.K. Panigrahi, S. Das, P.N. Suganthan, S.S. Dash (Eds.), Swarm, Evolutionary, and Memetic Computing, pp. 71–78, 2010.

[6] Antonio LaTorre , Santiago Muelas, Jose´-Marı´a Pen˜a "A MOS-based dynamic memetic differential evolution algorithm for continuous optimization: a scalability test", Soft Computing - A Fusion of Foundations, Methodologies and Applications, v.15 n.11, springer, November 2011, pp.2187-2199.

[7] S. Das and P. N. Suganthan, "Differential Evolution: A Survey of the State-of-the-Art," IEEE Transaction, Evolutionary Computation, vol. 15, no. 1, 2011, pp. 4–31.

[8] Y.-l. Li, J. Zhang, "A new differential evolution algorithm with dynamic population partition and local restart", Proceedings of the 13th Annual Genetic and Evolutionary Computation Conference, pp. 1085–1092, 2011.

[9] Subhrajit Roy, Sk. Minhazul Islam, Swagatam Das, Saurav Ghosh, Multimodal optimization by artificial weed colonies enhanced with localized group search optimizers, Applied Soft Computing 13, pp. 27–46, 2013.

[10] R. Tanabe and A. Fukunaga, "Success-History Based Parameter Adaptation for Differential Evolution," in IEEE CEC, 2013, pp. 71–78.

[11] Yongquan Zhou,Qifang Luo, and Huan Chen, "A Novel Differential Evolution Invasive Weed Optimization Algorithm for Solving Nonlinear Equations Systems", Journal of Applied Mathematics, Hindawi Publishing Corporation, Volume 2013, Article ID 757391, 18 pages.

[12] R. Tanabe and A. S. Fukunaga, "Improving the Search Performance of SHADE Using Linear Population Size Reduction", IEEE Congress on Evolutionary Computation (CEC), 2014.

[13] Muthana Khallil Ibrahim, "Linear Antenna Array Optimization by Using Evolutionary Algorithms", Journal of Babylon University,Engineering Sciences, No.(2), Vol.(22), 2014.

[14] J. J. Liang1 , B. Y. Qu2 , P. N. Suganthan, "Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization", IEEE CEC2014 Competetion, 2014.

[15] R.A. Sarker, S.M. Elsayed, T. Ray, Differential evolution with dynamic parameters selection for optimization problems, IEEE Trans. Evol.Comput. 18 pp. 689–707, 2014.

[16] Miruna Joe, Amali S. , and Baskar S.," Surrogate assisted-hybrid differential evolution algorithm using diversity control", Expert Systems, Vol. 32, No. 4, 2015, pp. 531-546.

[17] Awad, N. ; Ali, M.Z. ; Reynolds, R.G.A, "differential evolution algorithm with success-based parameter adaptation for CEC2015 learning-based optimization", IEEE Congress on Evolutionary Computation (CEC),pp1099-1105, 2015.

[18] Xianbing Meng,X. Z. Gao, Yu Liu, Novel Hybrid Bat Algorithm with DifferentialEvolution Strategy for Constrained Optimization,International Journal of Hybrid Information Technology,Vol.8, No.1, pp. 383- 396,2015.