

## OPMODE: Opposition based particle swarm optimization instilled with differential evolution



Qamar Abbas<sup>1,\*</sup>, Jamil Ahmad<sup>2</sup>, Hajira Jabeen<sup>1</sup>

<sup>1</sup>Computer Department, Iqra University, Islamabad, 44000, Pakistan

<sup>2</sup>Computer Department, Abasyn University, Islamabad, 44000, Pakistan

### ARTICLE INFO

#### Article history:

Received 21 October 2016

Received in revised form

17 April 2017

Accepted 18 May 2017

#### Keywords:

Opposition

Particle swarm optimization

Differential evolution

Initialization

### ABSTRACT

Particle Swarm Optimization (PSO) is a very powerful global optimization technique. Differential Evolution (DE) is another fast and emerging algorithm of evolutionary computing. PSODE is hybrid of PSO and DE that incorporates diversity in the PSO algorithm. In this research a new opposition based version of PSODE (OPMODE) is proposed that incorporates some more diversity by employing the opposition based learning in the PSODE algorithm. Some standard benchmark functions are used to access the performance of the OPMODE algorithm. The proposed version is then compared with the PSO, OPSO, and PSODE algorithm. The research result shows that the new version OPMODE has significance performance.

© 2017 The Authors. Published by IASE. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

### 1. Introduction

The theory of evolutionary algorithms is borrowed from the Darwin's theory of evolution that describes the survival of fittest through natural selection and the fitness improvement of individual species. The idea of successful survival in the evolutionary computing is inherited from reproduction and fitness of natural evolution process (Palit and Popovic, 2005). Evolutionary computing is an optimization paradigm based on mechanisms of evolution such as biological genetics and natural selection (Eberhart et al., 2001). The key objective of the optimization models is to get the best possible choice among the candidate solutions (Engelbrecht, 2005). Best choice refers to an acceptable or satisfactory solution that can be absolute best or any of the best solution over a set of candidate solutions. The advantage of evolutionary computing over other types of numerical methods is their ability to escape from local minima (Engelbrecht, 2005; Haupt and Haupt, 2004), however, EC algorithms do not guarantee to find the exact global optima (Engelbrecht, 2005; 2007).

Particle Swarm Optimization (PSO) is one of the well-known population based stochastic evolutionary algorithm that is proposed by Kennedy and Eberhart (1995). PSO has been applied to solve a

variety of optimization problems like other popular stochastic search techniques Genetic algorithms, Differential Evolution, Simulated Annealing etc. (Engelbrecht, 2005; 2007; Price et al., 2005). PSO has shown itself to have a good performance in solving many test as well as real life optimization problems.

Differential Evolution (DE) is one of the famous population based stochastic evolutionary algorithm that is proposed by Storn and Price (1995). It is an easy, a powerful and one of the simple evolutionary algorithms that are widely used in global optimization. The small number of control parameters associated with DE algorithm makes it suitable for global optimization problems. The major advantage of DE algorithm over other evolutionary algorithms is that the diverse nature of control parameters and mutation strategies of DE algorithm increases the probability of finding optima for function optimization problems and other optimization problems (Wang et al., 2014; Yildiz, 2013; Ali et al., 2005; Engelbrecht, 2007; Storn and Price 1995; 1997; Zamee et al., 2016). The varying nature of DE algorithm parameters enables it to escape from local optima problem (Pant et al., 2009; Wang et al., 2015; Li and Yin, 2016). The single objective as well as multi-objective versions of DE algorithm is successfully applied to many real life problems (Adeyemo et al., 2010). EC algorithms can be improved by enhancing their exploration search ability and incorporate some diversity in these algorithms that can improve the global optima finding ability.

In the rest of this paper, section 2 contains the background of PSO and DE, section 3 describes the

\* Corresponding Author.

Email Address: [qamar.bhk@gmail.com](mailto:qamar.bhk@gmail.com) (Q. Abbas)

<https://doi.org/10.21833/ijaas.2017.07.008>

2313-626X/© 2017 The Authors. Published by IASE.

This is an open access article under the CC BY-NC-ND license

(<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

proposed technique, benchmark functions used in this research are given in section 4, the results and convergence are discussed in section 5, section 6 discusses the conclusion of this research work.

## 2. Background

### 2.1. Particle swarm optimization

PSO algorithm is inspired from behaviour of different species due to their social and cooperative nature (Engelbrecht, 2007). PSO consists of population members of solutions termed as particles. The particles in PSO are moved throughout the search space in search of optimal solution. The particles in PSO evolve their position by using their flying direction termed as velocity. The particles of PSO keep track of their position and velocity. Personal best is referred to as a best position so for a particle has reached and gbest maintains the record of best of best particle in the swarm. Every particle in PSO has two attributes namely, position and velocity. Position shows that position of any particle in the n-dimensional space and velocity shows the step size that the PSO particle will use to update its position. The velocity of any  $i^{th}$  particle in D-dimensional space is represented by  $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ . The direction of  $i^{th}$  particle is calculated by using the following Eq. 1.

$$V_{id}^{k+1} = wV_{id}^k + c_1r_1(P_{id} - x_{id}) + c_2r_2(P_{gd} - x_{id}) \quad (1)$$

In D-dimensional search space the position of the  $i^{th}$  particle is represented by  $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$  for D dimensional space.

The position of each particle is updated by using the following Eq. 2.

$$X_{id}^{k+1} = X_{id}^k + V_{id}^{k+1} \quad (2)$$

In Eqs 1 and 2  $P_{id}$  is the particle  $i$  personal best position,  $P_{gd}$  is the global best position,  $x_{id}$  is the current position of a particle,  $c_1$  and  $c_2$  are social factors,  $r_1$  and  $r_2$  are uniform random numbers in the range(0, 1), inertia weight is denoted by  $w$ .

### 2.2. Differential evolution

DE is one of the well-known stochastic population based algorithm where n dimensional search space is used to randomly initialize the potential solutions. All potential solutions are equally likely to be selected as parent in DE algorithm. The candidate solutions evolve themselves by using a specified objective function to locate the optima by exploring the search space overtime (Yao et al., 1999). The amplified difference vector is added in random, best or current population member that depends on the mutation strategy being used (De Oliveira and Saramago, 2007).

To generate a mutation vector in DE algorithm, different vectors used in the mutation strategy are

selected from the existing population. DE algorithm candidate solutions are represented as an N-dimensional vector of the population NP. The randomly initialized population members are supposed to be scattered over the entire search space.  $X_{i,G}$  represents  $i^{th}$  population member where  $i = 1, 2, 3, \dots, NP$  at  $G^{th}$  generation in DE algorithm. The population of DE algorithm is evolved with the help of selection, mutation and crossover operators.

For each target vector  $X_{i,G}$  a mutation vector  $v_{i,G}$  is created by using the Eq. 3.

$$V_{i,G+1} = x_{r1,G} + F(x_{r2,G} - x_{r3,G}) \quad (3)$$

Such that the indices  $r1, r2, r3 \in \{1, 2, 3, \dots, NP\}$  and  $r1 \neq r2 \neq r3 \neq i$ . Eq. 4 is used to generate a trial vector  $u_{i,G+1}$ .

$$u_{i,G+1} = \begin{cases} v_{i,j,G} & \text{if } (randj(0,1) \leq CR \text{ or } j = j_{rand}) \\ x_{i,j,G} & \text{otherwise} \end{cases} \quad (4)$$

Here  $j = 1, 2, 3, \dots, D$  is dimension and  $randi[1, D]$  will return an integer in the given range.

After creating a trial vectors  $u_{i,G+1}$  its fitness value  $f(u_{i,G+1})$  is evaluated using given fitness function that is then compared with the fitness value of the target vector  $f(x_{i,G})$  and the vector having best fitness is moved to the next generation of DE algorithm. The selection is done based on the greedy approach by using Eq. 5.

$$x_{i,G+1} = \begin{cases} u_{i,G+1} & \text{if } (f(u_{i,G+1}) < f(x_{i,G})) \\ x_{i,G} & \text{otherwise} \end{cases} \quad (5)$$

Fitness is calculated against a given fitness function.

### 2.3. Related works in hybrid PSO and DE

Number of researchers has done their research work on the hybrid of PSO algorithm and DE algorithm. The detail of the research work on the hybrid of PSO and DE on various problems and applications is discussed in this section. Research work done by other researcher's shows the importance of the research work about the hybrid of PSO and DE algorithms. Abdullah et al. (2011) have discussed the limitation of trapping into local optima problem. They have incorporated DE operators in PSO algorithm to improve the local best capability of the PSO algorithm in solving complex optimization problems. The numerical analysis on the benchmark functions shows that the proposed technique PSODE outperforms both PSO and DE algorithms. Social and cognitive experience evolution in PSO using DE algorithm is considered in a research work of (Epitropakis et al., 2011). They have used DE in PSO algorithm to enhance the convergence of PSO and to efficiently guide the evolution process in PSO. The hybrid variant shows the promising results over a test suit of multimodal benchmark functions. Fu et al. (2010) have introduced two hybrid techniques of

PSO based on the DE operators and local search. The performance of both proposed schemes is better than standard PSO for a suit of high dimensional multimodal functions. Crossover and mutation operators of DE are employed in the PSO to enhance the convergence capability of the PSO algorithms and a simple local search is used to search of optimum value in the neighborhood of the current optimum value. [Fu et al. \(2011\)](#) have used a hybrid of PSO and DE (PSODE) approach for image segmentation. Research results of PSODE and compared with Otsu method of image segmentation and Gaussian Mixture Model (GMM) of image segmentation. Experimental result shows the PSODE is efficient as well as effective as compared to Otsu and GMM image segmentation methods. [Hu and Wu \(2010\)](#) have proposed Diploid Hybrid PSO with DE (DHPD) for open vehicle routing problem (OVRP) based on diploid genetic theory for. OVRP is used to define the fleet of the vehicle. In OVRP each root is treated as the Hamiltonian path. OVRP is used by the logistic companies, newspaper home delivery, local bus etc. a suit of 14 benchmark problems are available online at: [www.branchandcut.org/VRP/data/](http://www.branchandcut.org/VRP/data/) where the test size in the range of 55 to 199 customer data is used. The result shows that DHPD have better performance. [Khamsawang et al. \(2010\)](#) have applied hybrid PSO-DE have considered generator constraints to solve the economic dispatch problem. They have used DE mutation operators of DE algorithm in PSO algorithm to improve the diversity exploration of the PSO algorithm. For the hybrid PSODE the authors have implemented four different scenarios of the mutation of the DE algorithm. The research results of PSODE are compared with the PSO, GA and some other evolutionary algorithms. The results of PSODE are better in terms of mean generation cost and better convergence to reach to minimum generation cost. [Kim and Lee \(2009\)](#) have integrated PSO algorithm with the DE algorithm and named it Hybrid PSO (HPSO). HPSO employs the crossover and mutation operators of DE algorithms. They have also used Q-learning reinforcement learning method in the HPSO. A suit of 5 benchmark multimodal functions is used along with the piston oil optimization problems. The research result shows that the performance of the HPSO refines the optimization performance as compared to PSO, DE or GA algorithm. [Liu et al. \(2010\)](#) have used the hybrid of DE and PSO algorithm for engineering optimization and constrained optimization problems. They have integrated the DE crossover and mutation in PSO that may help PSO to jump out of stagnation. The result shows that the hybrid version improves the performance and speeds up the convergence of the PSO algorithm. The authors have used a suit of 11 benchmark functions and 5 engineering optimization problems to measure the performance of the hybrid version. [Niknam et al. \(2011\)](#) have introduced the hybrid fuzzy adaptive PSO algorithm and DE algorithm for the distribution of feeder reconfiguration. Fuzzy adaptive PSODE (FAPSODE) combines the features of PSO and DE

algorithms. Fuzzy adaptive algorithms have two parts first part is the fuzzy adaptive binary PSO which determines the status of the tie switches to be open or close. The second part of the fuzzy PSO determines sectionalizing of switch numbers. The research results shows that FAPSODE-DE have better performance as compared to PSO or DE algorithm. [Niu and Li \(2008a\)](#) have used the hybrid of PSO and DE algorithm and named it PSODE. They have used PSODE for global optimization problems. Four benchmark functions are used to compare the performance of PSODE algorithm with PSO and DE algorithm. Research results shows that the PSODE have better performance as compared to PSO and DE algorithms. [Niu and Li \(2008b\)](#) have discussed the design of the T-S fuzzy model in their research work. Their proposed model is based on the hybrid PSODE algorithm. A PSODE use the concept of two populations where on population is evolved using DE algorithm and the other population is evolved using PSO algorithm. Non-linear inertia weight and the mutation mechanisms are presented in the PSODE algorithm. PSO, DE and PSODE are applied to a fuzzy identifier design which shows that PSODE have promising performance. [Parassuram et al. \(2011\)](#) have considered the hybridization of PSO and DE algorithm named hybrid PSO for valve point effected economic dispatch problem. Valve point effect in thermal is used to get the accurate fuel cost of the. The hybrid PSO integrated selection and mutation evolutionary operators with PSO. The hybrid PSO is compared with other evolutionary algorithms shows that hybrid PSO having fast convergence speed and robustness to solve the hard optimization problems. [Saber and Rahman \(2011\)](#) have used the hybrid of PSO and DE with name Particle Swarm Differential Evolution Optimization (PSDEO). They have used PSDEO for economic dispatch problem (ELD) in their research work. This hybrid version balances the local and global search abilities of PSO and DE algorithms. PSDEO have minimum cost as compared to PSO, DE, GA and some other evolutionary techniques when applied to the ELD benchmark problem. [Sedki and Ouazar \(2012\)](#) have used hybrid of PSO and DE (PSODE) to design optimal solution of water distribution problem. The three benchmark problems for water distribution the two-loop network, the New York Tunnels network and the Hanoi network are considered in their research work. The research results shows that the results of the PSODE are better than the results of the PSO, GA, DE and some other evolutionary algorithms. [Voglis et al. \(2012\)](#) have introduced hybrid PSODE and Memetic search (MEMPSODE) for a set of benchmark functions. The research result shows that the performance of the MEMPSODE has better results when compared with the PSO and DE algorithm. [Wan et al. \(2011\)](#) have hybridized Gaussian PSO and DE algorithm to solve three constrained engineering optimization problems. Differential evolution mutation and regeneration strategy is employed in the GPSO algorithm (DGPSO).

The computational execute time of DGPSO is greatly cut down as compared to GPSO and DE algorithm.

Wang et al (2010) have used hybrid PSODE with three populations based on multiple differential evolutionary modes. The three modes of DE used in this research are the rand mode, best mode and current-best mode. The research result shows that all three modes of DE improve the performance of the PSODE algorithm for a test suit of 4 benchmark functions.

Wang et al. (2010) have used hybrid of PSO and DE algorithm for reactive power optimization application. To validate the effectiveness of PSODE algorithm its results are compared with DE and PSO algorithm. PSODE outperforms PSO and DE algorithm for power loss reduction and number of iterations required to achieve the desired power.

Xu et al. (2012) have introduced hybrid PSODE algorithm with least square support vector regression for the modeling of ammonia conversion rate in the ammonia synthesis. In their research work characterizes relationship between operational variables and ammonia conversion rate, is established using least squares support vector regression (LSSVR). The research result shows the good global convergence reliability and feasibility of the proposed algorithm.

Xu and Gu (2009) have hybridized PSO and DE with prior crossover. In their research work they have introduced novel version of hybrid of PSO and DE (PSODE) by introducing an extra population for prior crossover operation.

To prior crossover is used to incorporate some diversity in the population that increases the probability to reach to the global optimal point. Research results are generated using a suit of 5 benchmark functions that shows that prior crossover based PSODE outperforms PSO and DE algorithms.

### 3. Proposed version

The proposed algorithm is the infusion of the opposition based particle swarm optimization and differential evolution algorithm (OPSODE) shown in Fig. 1. Different researchers work on the hybrid of PSO and DE that is discussed in section 2 of this paper.

In this research an effort is made to improve the performance of the hybrid of PSO and DE algorithm by integrating the concept of the opposition based population initialization scheme. This opposition based scheme in the hybrid of PSO and DE will incorporate the diversity in the population to find the optimum value over the continuous range of search problems.

Tizhoosh (2005) has introduced the opposition-based learning (OBL) scheme in the machine learning. Opposition-based learning scheme uses the concept of opposite points and opposite numbers. OBL generates the opposite number  $\bar{x}$  for any

number  $x \in R$  over the specified interval  $[a, b]$  by using the following Eq. 6.

$$\bar{x} = a + b - x \quad (6)$$

### 4. Test functions

A test suite of 15 benchmark test functions is used to in this research work. All test functions are the common test functions picked up from the literature (Ali et al., 2005; Brest et al., 2006; Xu and Li, 2007; Ali et al., 2009a, 2009b, 2009c; Abbas et al., 2015). These test functions are given in Table 1 along with the other details like the name of the function, search space and the optimum value.

```

1. Population Random initialization  $P(N_p)$ 
2. Fitness calculation of initial population
3. Calculate the opposite of initial population  $OP(N_p)$  using Eq. 3
4. Fitness calculation of opposite population
5. Select  $N$  fittest individuals from  $\{P(N_p) \cup OP(N_p)\}$ 
6. While termination condition is not true
7. For each population member  $P$  calculate fitness
8. If  $(fitness(P) < fitness(P_{best}))$ 
9. Copy particle  $P$  in  $P_{best}$ 
10. Set best of all  $P_{best}$  as  $G_{best}$ 
11. end
12. Update Particle position and velocity of all particles according to Eqs.1 and 2
13. For each population member create offspring by
14. Perform Mutation using Eq. 3
15. Perform Crossover using Eq. 4
16. Calculate the fitness of Offspring's
17. Replace Offspring with parent if Offspring is better than Parent
18. end

```

Fig. 1: Pseudocode of proposed OPSODE Algorithm

### 5. Experimental settings, results and convergence graphs

The parameters used by the PSO and DE algorithm to obtain the results are given as: the value of the  $r_1$  and  $r_2$  is taken from (0, 1) and  $c_1=c_2=1.49$  (Poli et al., 2007).

The range (0.4, 0.9) is used for the value of inertia weight (Poli et al., 2007); Population size  $N_p = 30$  is used as a size of population, The small as well as big dimensions are used during the experimentation such as 2, 4, 10, 20 and 30; Crossover rate and mutation probability have values  $CR=0.9$  and  $F= 0.5$  respectively (Brest et al., 2008; Rahnamayan et al., 2008).

Average fitness experimental results are obtained by taking 100 runs and 2000 iterations of DE algorithm. The results obtained against this parameter setting are reported in Table 2.

The convergence graphs PSO, OPSO, PSODE and OPSODE are generated that are shown in Fig. 2. The graphs in Fig. 2 are generated for some selected functions and selected dimensions.

To assess the convergence performance of proposed algorithm, the graph showing the convergence is plotted in Fig. 2 for sample functions for various dimensions.

**Table 1:** List of benchmark functions

Function	Function Name	Mathematical Equation	Search Space	Dimension
$f_1(x)$	"Sphere model"	$\sum_{i=0}^n x_i^2$	$x_i \in [-5.12, 5.12]$	nD
$f_2(x)$	"Axis parallel hyperellipsoid"	$\sum_{i=0}^n i \cdot x_i^2$	$x_i \in [-5.12, 5.12]$	nD
$f_3(x)$	"Griewangk's function"	$\sum_{i=1}^n \left( \frac{x_i^2}{4000} - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) \right) + 1$	$x_i \in [-600, 600]$	nD
$f_4(x)$	"Sum of different power"	$\sum_{i=1}^n  x_i ^{(i+1)}$	$x_i \in [-1, 1]$	nD
$f_5(x)$	"Ackley's path function"	$-20 \exp \left[ -0.2 \sqrt{\frac{\sum_{i=1}^n x_i^2}{n}} \right] - \exp \left[ \frac{\sum_{i=1}^n \cos(2\pi x_i)}{n} \right] + 20 + e$	$x_i \in [-32, 32]$	nD
$f_6(x)$	"Schwefel's problem 2.22"	$\sum_{i=1}^n  x_i  + \prod_{i=1}^n  x_i $	$x_i \in [-10, 10]$	nD
$f_7(x)$	"Quartic function, i.e., noise"	$\sum_{i=1}^n i x_i^4 + \text{random}[0,1]$	$x_i \in [-1.28, 1.28]$	nD
$f_8(x)$	"De Jong's function 4 (no noise)"	$\sum_{i=0}^n i x_i^4$	$x_i \in [-1.28, 1.28]$	nD
$f_9(x)$	"Cigar"	$x_1^2 + 100000 \sum_{i=1}^n x_i^2$	$x_i \in [-10, 10]$	nD
$f_{10}(x)$	"Function-15"	$\sum_{i=1}^{n-1} [0.2x_i^2 + 0.1x_i^2 \sin(2x_i)]$	$x_i \in [-10, 10]$	nD
$f_{11}(x)$	"Miele and Cantell Problem"	$(\exp(x_1) - x_2)^4 + 100(x_2 - x_3)^6 + (\tan(x_3 - x_4))^4 + x_1^8$	$x_i \in [-1, 1]$	4D
$f_{12}(x)$	"Powell's Quadratic Problem"	$(x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$	$x_i \in [-10, 10]$	4D
$f_{13}(x)$	"Backer and Lago problem"	$( x_1  - 5)^2 + ( x_2  - 5)^2$	$x_i \in [-10, 10]$	2D
$f_{14}(x)$	"Eggcrate"	$x_1^2 + x_2^2 + 25(\sin^2 x_1 + \sin^2 x_2)$	$(x_1, x_2) \in [-2\pi, 2\pi]$	2D
$f_{15}(x)$	"Matyas function"	$0.26(x_1^2 + x_2^2) - 0.48(x_1 x_2)$	$x_i \in [-10, 10]$	2D

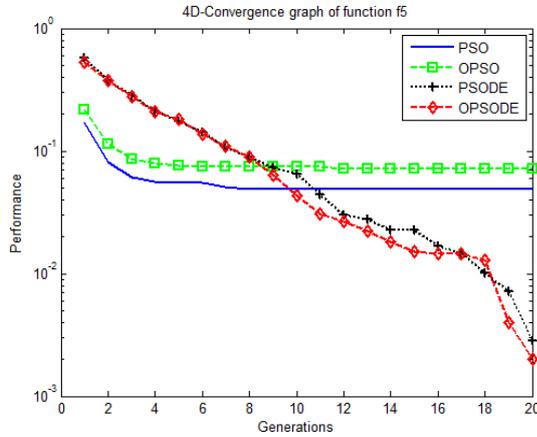
**Table 2:** Performance results of PSO, PSODE & OPSODE for suit of functions optimization

Function	DIM	PSO	OPSO	PSODE	OPSO	Function	DIM	PSO	OPSO	PSODE	OPSO		
		Mean±S.D	Mean±S.D	Mean±S.D	Mean±S.D			Mean±S.D	Mean±S.D	Mean±S.D	Mean±S.D		
$f_1$	2	5.67E-249	9.50E-249	1.46E-264	1.11E-262	$f_7$	2	1.09E-04	1.13E-04	1.51E-04	1.48E-04		
		±	±	±0.00E+00	±			±1.66E-04	±1.32E-04	±1.26E-04	±1.19E-04		
	0.00E+00	0.00E+00	0.00E+00	0.00E+00	4		4	2.80E-04	3.13E-04	7.09E-04	7.72E-04		
	1.56E-214	2.58E-215	7.44E-195	2.56E-196				±3.03E-04	±3.20E-04	±4.04E-04	±4.68E-04		
	±0.00E+00	±0.00E+00	±0.00E+00	±0.00E+00	10		10	2.00E-03	1.87E-03	5.61E-03	6.01E-03		
	3.39E-140	1.16E-139	4.11E-106	3.07E-108				±1.73E-139	±8.12E-139	±3.28E-105	±1.89E-107	±1.63E-03	±1.85E-03
	6.05E-62	1.63E-60	5.35E-60	1.76E-60	20		20	1.15E-02	9.72E-03	1.62E-02	1.72E-02		
	±6.01E-61	±1.62E-59	±2.86E-59	±6.91E-60				±7.45E-03	±6.16E-03	±6.18E-03	±6.92E-03	±7.45E-03	±6.16E-03
	1.31E+00	2.62E-01	1.58E-42	2.05E-42	30		30	3.21E-02	2.82E-02	2.85E-02	2.68E-02		
	±5.71E+00	±2.61E+00	±3.99E-42	±5.82E-42				±1.90E-02	±1.61E-02	±8.43E-03	±1.08E-02	±1.90E-02	±1.61E-02
7.15E-283	1.14E-279	2.50E-313	4.09E-316	2	2	0.00E+00	0.00E+00	0.00E+00	0.00E+00				
±0.00E+00	±0.00E+00	±0.00E+00	±0.00E+00			±0.00E+00							
4.49E-230	1.86E-230	7.91E-237	5.31E-237	4	4	0.00E+00	0.00E+00	0.00E+00	0.00E+00				
±0.00E+00	±0.00E+00	±0.00E+00	±0.00E+00			±0.00E+00							
4.68E-149	2.79E-149	2.70E-147	6.51E-147	10	10	3.33E-272	4.56E-276	2.34E-267	9.95E-270				
±3.94E-148	±1.88E-148	±2.23E-146	±4.39E-146			±0.00E+00							
8.65E+00	6.55E+00	2.46E-72	2.62E-01	20	20	2.01E-134	4.44E-134	5.04E-133	1.56E-131				
2.57E+01	1.87E+01	2.44E-71	2.61E+00			±1.25E-133	±2.38E-133	±2.41E-132	±1.52E-130	±1.25E-133	±2.38E-133	±2.41E-132	±1.52E-130
4.22E+01	2.52E+01	2.36E+00	3.15E+00	30	30	4.03E-01	1.07E-01	1.20E-65	2.41E-64				
7.44E+01	5.93E+01	9.87E+00	1.19E+01			±1.54E+00	±7.52E-01	±7.47E-65	±2.35E-63	±1.54E+00	±7.52E-01	±7.47E-65	±2.35E-63
1.13E-03	1.33E-03	7.40E-05	4.44E-04	2	2	9.22E-243	1.03E-243	3.66E-260	5.22E-259				
±2.71E-03	±2.84E-03	±7.36E-04	±1.76E-03			±0.00E+00							
2.14E-02	1.89E-02	4.18E-02	4.22E-02	4	4	2.26E-209	3.64E-209	6.66E-189	5.78E-189				
±1.92E-02	±1.58E-02	±2.33E-02	±2.21E-02			±0.00E+00							
9.17E-02	8.76E-02	1.07E-01	1.15E-01	10	10	5.28E-124	3.19E-135	5.10E+05	1.00E+04				
±5.83E-02	±4.54E-02	±5.70E-02	±7.64E-02			±5.25E-123	±1.29E-134	±5.00E+05	±9.95E+04	±5.25E-123	±1.29E-134	±5.00E+05	±9.95E+04
9.31E-01	4.32E-02	1.91E-02	2.35E-02	20	20	1.00E+06	1.00E+06	1.00E+06	1.00E+06				
±8.99E+00	±6.99E-02	±2.14E-02	±3.04E-02			±0.00E+00							
1.90E+00	1.01E+00	7.93E-03	8.27E-03	30	30	1.00E+06	1.00E+06	1.00E+06	1.00E+06				
±1.26E+01	±8.97E+00	±9.39E-03	±1.05E-02			±0.00E+00							
1.58E-135	2.72E-137	3.16E-155	9.67E-155	2	2	3.55E-281	5.32E-281	0.00E+00	0.00E+00				
±1.56E-134	±1.05E-136	±1.94E-154	±8.52E-154			±0.00E+00							
1.82E-131	7.65E-132	8.40E-144	6.13E-144	4	4	2.47E-230	1.92E-230	9.38E-238	8.85E-239				
±9.11E-	±6.66E-	±4.28E-143	±4.29E-			±0.00E+00							

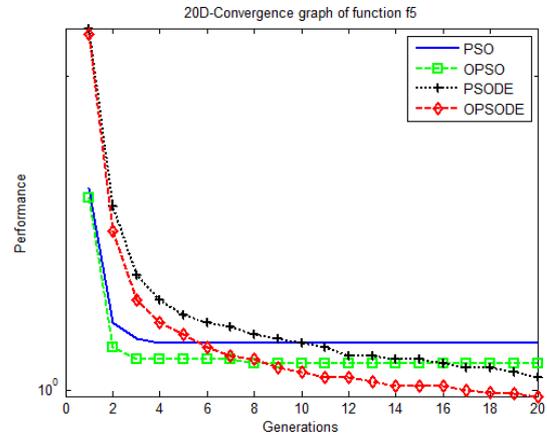
	131	131		143		6.05E-151	1.75E-149	4.41E-148	9.86E-148
10	1.91E-116 ±1.65E-115	1.05E-111 ±1.04E-110	2.54E-121 ±2.53E-120	8.23E-125 ±4.46E-124		±2.17E-150	±1.40E-148	±3.10E-147	±5.55E-147
20	4.22E-35 ±4.20E-34	9.43E-41 ±9.38E-40	6.96E-76 ±6.67E-75	6.01E-76 ±4.01E-75		4.41E-67 ±4.38E-66	4.05E-69 ±3.43E-68	1.84E-75 ±1.21E-74	2.47E-70 ±2.46E-69
30	5.44E-23 ±5.41E-22	1.07E-19 ±1.07E-18	1.52E-13 ±1.44E-12	6.02E-12 ±4.15E-11		3.63E-01 ±2.54E+00	3.49E-09 ±1.91E-08	3.28E-14 ±2.59E-13	4.36E-15 ±2.69E-14
2	-1.44E-16 ±0.00E+00	-1.44E-16 ±0.00E+00	-1.44E-16 ±0.00E+00	-1.44E-16 ±0.00E+00	<i>f11</i>	3.55E-281 ±0.00E+00	5.32E-281 ±0.00E+00	0.00E+00 ±0.00E+00	0.00E+00 ±0.00E+00
4	4.89E-02 ±9.07E-02	7.23E-02 ±1.18E-01	2.84E-03 ±2.14E-02	1.99E-03 ±1.98E-02	<i>f12</i>	2.47E-230 ±0.00E+00	1.92E-230 ±0.00E+00	9.38E-238 ±0.00E+00	8.85E-239 ±0.00E+00
<i>f5</i>	5.38E-01 ±1.87E-01	5.62E-01 ±1.89E-01	4.38E-01 ±1.87E-01	4.32E-01 ±1.82E-01	<i>f13</i>	6.05E-151 ±2.17E-150	1.75E-149 ±1.40E-148	4.41E-148 ±3.10E-147	9.86E-148 ±5.55E-147
20	1.11E+00 ±3.98E-01	1.06E+00 ±2.99E-01	1.03E+00 ±2.85E-01	9.87E-01 ±2.54E-01	<i>f14</i>	4.41E-67 ±4.38E-66	4.05E-69 ±3.43E-68	1.84E-75 ±1.21E-74	2.47E-70 ±2.46E-69
30	1.77E+00 ±5.58E-01	1.59E+00 ±3.59E-01	1.38E+00 ±3.47E-01	1.33E+00 ±2.95E-01	<i>f15</i>	3.63E-01 ±2.54E+00	3.49E-09 ±1.91E-08	3.28E-14 ±2.59E-13	4.36E-15 ±2.69E-14
2	4.89E-125 ±1.88E-124	4.32E-125 ±2.20E-124	2.70E-132 ±2.08E-131	1.96E-132 ±1.04E-131					
4	2.81E-109 ±1.22E-108	7.90E-109 ±5.79E-108	5.62E-98 ±3.53E-97	1.41E-98 ±4.29E-98					
<i>f6</i>	2.85E-55 ±2.84E-54	3.78E-55 ±3.76E-54	3.13E-55 ±1.78E-54	5.80E-56 ±1.91E-55					
10	4.06E-03 ±2.11E-02	8.18E-04 ±5.81E-03	1.67E-31 ±1.07E-30	9.81E-32 ±2.27E-31					
20	2.42E-01 ±1.05E+00	3.94E-01 ±1.44E+00	3.29E-22 ±7.11E-22	1.19E-21 ±3.60E-21					
30									

The x-axis is the number of iterations and the y-axis shows the performance (average fitness value). Research results are generated using a test suit of 15 benchmark functions reported in Table 1. The average fitness value and the standard deviation of test suit of functions are reported in Table 2 of this paper. The results are generated using different dimensions 2D, 4D, 10D, 20D and 30D to measure the performance of proposed version with the

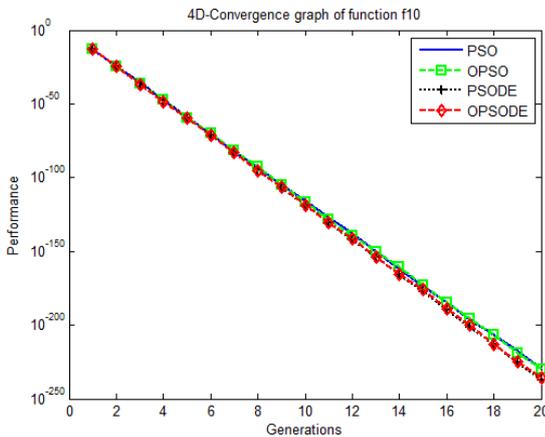
existing ones. The functions  $f_{11}$ ,  $f_{12}$  are the 4 dimensional functions; functions  $f_{13}$ - $f_{15}$  are 2 dimensional functions and remaining are n-dimensional functions. The results are generated for a suit of test functions given in Table 1. The values that are reported as bold faces in Table 2 are the best values vales among the comparing algorithms. The research result shows that the proposed OPSODE has significant performance.



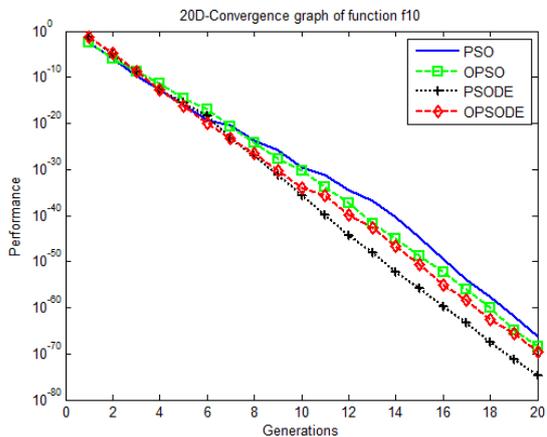
(a)



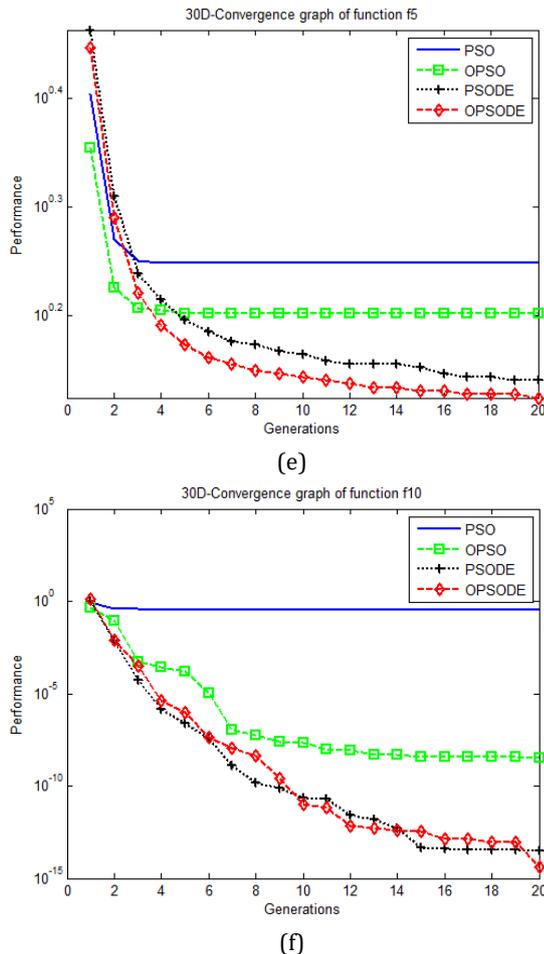
(c)



(b)



(d)



**Fig. 2:** (a-f) Convergence graphs of functions  $f_5$ ,  $f_{10}$  for 4, 20, 30 dimensions

## 6. Conclusion

In this research a new variation of the PSODE is proposed and implemented on a test suit of benchmark functions. Diversity enhances the global search capability of heuristic algorithms. Diversity is incorporated by utilizing the concept of opposition based concept.

The algorithms are run 100 times with fix number of iterations and then average fitness is reported in the result section. Most of the functions are n-dimensional functions, however, some small dimensional functions ( $f_{13}$ ,  $f_{15}$ ) with 2 dimensions and ( $f_{11}$ ,  $f_{12}$ ) with 4 dimensions are used to generate the experimental results. Results are generated with parameter setting given in the result section for various dimensions (2, 4, 10, 20, 30) for the functions in Table 1.

The research results are reported in Table 2 showing better results as bold faces. It is obvious from the experimental results that the overall performance of OPSODE is better than the performance of PSO, OPSO, PSODE and OPSODE.

## Conflict of interest statement

The authors declare that there is no conflict of interests regarding the publication of this paper.

## References

- Abbas Q, Ahmad J, and Jabeen H (2015). A novel tournament selection based differential evolution variant for continuous optimization problems. *Mathematical Problems in Engineering*. <http://dx.doi.org/10.1155/2015/205709>
- Abdullah A, Deris S, Hashim SZM, Mohamad MS, and Arjunan SNV (2011). An improved local best searching in particle swarm optimization using differential evolution. In the 11<sup>th</sup> International Conference on Hybrid Intelligent Systems, IEEE, Melacca, Malaysia: 115-120. <https://doi.org/10.1109/HIS.2011.6122090>
- Adeyemo J, Bux F, and Otieno F (2010). Differential evolution algorithm for crop planning: Single and multi-objective optimization model. *International Journal of the Physical Sciences*, 5(10): 1592-1599.
- Ali M, Pant M, and Abraham A (2009a). A modified differential evolution algorithm and its application to engineering problems. *International Conference of Soft Computing and Pattern Recognition*, IEEE: 196-201. <https://doi.org/10.1109/SoCPaR.2009.48>
- Ali M, Pant M, and Abraham A (2009b). Simplex differential evolution. *Acta Polytechnica Hungarica*, 6(5): 95-115.
- Ali M, Pant M, and Singh VP (2009c). An improved differential evolution algorithm for real parameter optimization problems. *International Journal of Recent Trends in Engineering*, 1(5): 63-65.
- Ali MM, Khompatraporn C, and Zabinsky ZB (2005). A numerical evaluation of several stochastic algorithms on selected continuous global optimization test problems. *Journal of Global Optimization*, 31(4): 635-672.
- Brest J, Greiner S, Boskovic B, Mernik M, and Zumer V (2006). Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. *IEEE Transaction on Evolutionary Computing*, 10(6): 646-657.
- Brest J, Zamuda A, Boskovic B, Maucec MS, and Zumer V (2008). High-dimensional real-parameter optimization using self-adaptive differential evolution algorithm with population size reduction. In the IEEE Conference on Evolutionary Computation, IEEE, Hong Kong, China: 2032-2039. <https://doi.org/10.1109/CEC.2008.4631067>
- De Oliveira, GTS and Saramago SFP (2008). A contribution to the study about differential evolution. *Ciência and Engenharia*, 16(1/2): 1-8.
- Eberhart RC, Shi Y, and Kennedy J (2001). *Swarm intelligence*. Morgan Kaufmann Division of Academic Press, San Francisco, USA.
- Engelbrecht A (2005). *Fundamentals of computational swarm intelligence*. John Wiley and Sons, West Sussex, England.
- Engelbrecht AP (2007). *Computational intelligence an introduction*. 2<sup>nd</sup> Edition, John Wiley and Sons, West Sussex, England.
- Epitropakis MG, Tasoulis DK, Pavlidis NG, Plagianakos VP, and Vrahatis MN (2011). Enhancing differential evolution utilizing proximity-based mutation operators. *IEEE Transactions on Evolutionary Computation*, 15(1): 99-119.
- Fu W, Johnston M, and Zhang M (2010). Hybrid particle swarm optimisation algorithms based on differential evolution and local search. In the Australasian Joint Conference on Artificial Intelligence: 313-322. Springer, Berlin, Heidelberg, Germany. [https://doi.org/3\\_2-17432-642-3-978/10.1007](https://doi.org/3_2-17432-642-3-978/10.1007)
- Fu W, Johnston M, and Zhang M (2011). A hybrid particle swarm optimisation with differential evolution approach to image segmentation. In the European Conference on the Applications of Evolutionary Computation, Springer, Berlin, Heidelberg, Germany: 173-182. [https://doi.org/10.1007/978-3-642-20525-5\\_18](https://doi.org/10.1007/978-3-642-20525-5_18)

- Haupt RL and Haupt SE (2004). Practical genetic algorithms. 2<sup>nd</sup> Edition, John Wiley and Sons, New Jersey, USA.
- Hu F and Wu F (2010). Diploid hybrid particle swarm optimization with differential evolution for open vehicle routing problem. In the 8<sup>th</sup> World Congress on Intelligent Control and Automation, IEEE, Jinan, China: 2692-2697. <https://doi.org/10.1109/WCICA.2010.5554989>
- Kennedy J and Eberhart RC (1995). Particle swarm optimization. In the IEEE International Conference on Neural Networks, IEEE, Perth, Australia: 1942-1948. <https://doi.org/10.1109/ICNN.1995.488968>
- Khamsawang S, Wannakarn P, and Jiriwibhakorn S (2010). Hybrid PSO-DE for solving the economic dispatch problem with generator constraints. In the 2<sup>nd</sup> International Conference on Computer and Automation Engineering (ICCAE), IEEE, Singapore, Singapore: 135-139. <https://doi.org/10.1109/ICCAE.2010.5451501>
- Kim P and Lee J (2009). An integrated method of particle swarm optimization and differential evolution. Journal of Mechanical Science and Technology, 23(2): 426-434.
- Li X and Yin M (2016). Modified differential evolution with self-adaptive parameters method. Journal of Combinatorial Optimization, 31(2): 546-576.
- Liu H, Cai Z, and Wang Y (2010). Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization. Applied Soft Computing, 10: 629-640.
- Niknam T, Farsani EA, Nayeripour M, and Firouzi BB (2011). Hybrid fuzzy adaptive particle swarm optimization and differential evolution algorithm for distribution feeder reconfiguration. Electric Power Components and Systems, 39(2): 158-175.
- Niu B and Li L (2008a). A novel PSO-DE-based hybrid algorithm for global optimization. In the 4<sup>th</sup> international conference on Intelligent Computing: Advanced Intelligent Computing Theories and Applications - with Aspects of Artificial Intelligence (ICIC '08), Springer-Verlag Berlin, Heidelberg, Shanghai, China: 156 - 163. [https://doi.org/10.1007/978-3-540-85984-0\\_20](https://doi.org/10.1007/978-3-540-85984-0_20)
- Niu B and Li L (2008b). Design of T-S fuzzy model based on PSODE algorithm. In the 4<sup>th</sup> International Conference on Intelligent Computing, Springer, Berlin, Heidelberg: 384-390. [https://doi.org/10.1007/978-3-540-85984-0\\_47](https://doi.org/10.1007/978-3-540-85984-0_47)
- Palit AK and Popovic D (2005). Computational intelligence in time series forecasting: Theory and Engineering Applications Advances in Industrial Control. Springer-Verlag, USA.
- Pant M, Thangaraj R, and Singh VP (2009). A new differential evolution algorithm for solving global optimization problems. International Conference on Advanced Computer Control, IEEE, Singapore: 388-392. <https://doi.org/10.1109/ICACC.2009.102>
- Parassuram A, Deepa SN, and Karthick M (2011). A hybrid technique using particle swarm optimization and differential evolution to solve economic dispatch problem with valve-point effect. International Conference on Recent Advancements in Electrical, Electronics and Control Engineering, IEEE, Sivakasi, India: 51-56. <https://doi.org/10.1109/ICONRAEECE.2011.6129744>
- Poli R, Kennedy J, and Blackwell T (2007). Particle swarm optimization An overview. Swarm Intelligence, 1(1): 33-57.
- Price K, Storn RM, and Lampinen JA (2005). Differential evolution: A Practical approach to global optimization. 1<sup>st</sup> Edition, Springer, New York, USA.
- Rahnamayan S, Tizhoosh HR, and Salama MM. (2008). Opposition-based differential evolution. IEEE Transactions on Evolutionary Computation, 12(1): 64-79.
- Saber AY and Rahman DMF, (2011). Economic load dispatch using particle swarm differential evolution optimization. In the IEEE Power and Energy Society General Meeting, IEEE, Detroit, USA: 1-8. <https://doi.org/10.1109/PES.2011.6039891>
- Sedki A and Ouazar D (2012). Hybrid particle swarm optimization and differential evolution for optimal design of water distribution systems. Advanced Engineering Informatics, 26: 582-591.
- Storn R and Price K (1995). Differential evolution—A simple and efficient adaptive scheme for global optimization over continuous spaces. TR-95-012, International Computer Science Institute, Berkeley, USA.
- Storn R and Price K (1997). Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization, 11(4): 341-359.
- Tizhoosh H (2005). Opposition-based learning: A new scheme for machine intelligence. In the International Conference on Computational Intelligence for Modelling, Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, IEEE, Vienna, Austria: 695-701. <https://doi.org/10.1109/CIMCA.2005.1631345>
- Voglis C, Piperagkas GS, Parsopoulos KE, Papageorgiou DG, and Lagaris IE (2012). MEMPSODE: comparing particle swarm optimization and differential evolution within a hybrid memetic global optimization framework. In the 14<sup>th</sup> Annual Conference Companion on Genetic and Evolutionary Computation, Philadelphia, USA: 253-260. <https://doi.org/10.1145/2330784.2330821>
- Wan C, Wang J, Yang G, and Zhang X (2011). Gaussian particle swarm optimization with differential evolution mutation. In the International Conference in Swarm Intelligence, Springer Berlin Heidelberg, Chongqing, China: 439-446. [https://doi.org/10.1007/978-3-642-21515-5\\_52](https://doi.org/10.1007/978-3-642-21515-5_52)
- Wang GG, Gandomi AH, Alavi AH, and Hao GS (2014). Hybrid krill herd algorithm with differential evolution for global numerical optimization. Neural Computing and Applications, 25(2): 297-308.
- Wang SL, Ng TF, Jamil NA, Samuri SM, Mailok R, and Rahmatullah B (2015). Self-adapting approach in parameter tuning for differential evolution. In the Conference on Technologies and Applications of Artificial Intelligence (TAAI), IEEE, Tainan, Taiwan: 113-119. <https://doi.org/10.1109/TAAI.2015.7407109>
- Wang X, Yang Q, and Zhao Y (2010). Research on hybrid PSODE with triple populations based on multiple differential evolutionary models. In the International Conference on Electrical and Control Engineering (ICECE), IEEE, Wuhan, China: 1692-1696. <https://doi.org/10.1109/iCECE.2010.1418>
- Xu W and Gu X (2009). A hybrid particle swarm optimization approach with prior crossover differential evolution. In the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation (GEC '09), Shanghai, China: 671-678. <https://doi.org/10.1145/1543834.1543926>
- Xu W, Wang R, Zhang L, and Gu X (2012). A multi-population cultural algorithm with adaptive diversity preservation and its application in ammonia synthesis process. Neural Computing and Applications, 21(6): 1129-1140.
- Xu X and Li Y (2007). Comparison between particle swarm optimization, differential evolution and multi-parents crossover. In the International Conference on Computational Intelligence and Security, IEEE, Harbin, China: 124-127. <https://doi.org/10.1109/CIS.2007.37>
- Yao X, Liu Y, and Lin G (1999). Evolutionary Programming made faster. IEEE Transaction on Evolutionary Computation, 3(2): 82-102.
- Yildiz AR (2013). A new hybrid differential evolution algorithm for the selection of optimal machining parameters in milling operations. Applied Soft Computing, 13(3): 1561-1566.
- Zamee MA, Islam KK, Ahmed AA, and Zafreen KR (2016). Differential evolution algorithm based load frequency control

in a two-area conventional and renewable energy based nonlinear power system. In the 4<sup>th</sup> International Conference on the Development in the in Renewable Energy Technology,

IEEE, Dhaka, Bangladesh: 1-6. <https://doi.org/10.1109/ICDRET.2016.7421476>.