Decision Science Letters 5 (2016) 361–380

Contents lists available at GrowingScience

Decision Science Letters

homepage: www.GrowingScience.com/dsl

Improved symbiotic organisms search algorithm for solving unconstrained function optimization

Sukanta Nama[a*], Apu Kumar Saha[a] and Sima Ghosh[b]

[a]*Department of Mathematics, National Institute of Technology Agartala, Tripura-799046, India.*
[b]*Department of Civil Engineering, National Institute of Technology Agartala, Tripura-799046, India*

C H R O N I C L E

Article history:
Received October 25, 2015
Received in revised format:
February 12, 2016
Accepted February 22, 2016
Available online
Februray 22 2016

Keywords:
Symbiotic organisms search
Population based algorithm
Unconstrained optimization
Random weighted difference vector
Random weighed reflection

A B S T R A C T

Recently, Symbiotic Organisms Search (SOS) algorithm is being used for solving complex problems of optimization. This paper proposes an Improved Symbiotic Organisms Search (I-SOS) algorithm for solving different complex unconstrained global optimization problems. In the improved algorithm, a random weighted reflective parameter and predation phase are suggested to enhance the performance of the algorithm. The performances of this algorithm are compared with the other state-of-the-art algorithms. The parametric study of the common control parameter has also been performed.

1. Introduction

In the real life scenario, the optimization problems are not as simple that they can be solved by deterministic search techniques. There may have more than one global optimal solution and one may be interested to find many global solutions for various reasons depending on the need of the problem. In addition, some functions may have discontinuities, and thus the derivative information is not easy to obtain for those functions. This may pose a strong challenge to many traditional methods of optimization. To overcome this difficulty, efficient optimization algorithms are needed which can deal with this kind of challenges. There are many optimization algorithms which can be classified in many ways, depending on the focus and characteristics. Some of the well-known algorithms found by the literature survey are Genetic Algorithm (GA) (Holland, 1975), Differential Evolution (DE) (Storn & Price, 1997), Particle Swarm Optimization (PSO) (Kennedy & Eberhart, 1995; Shi & Eberhart 1998), Artificial Bee Colony (ABC) (Dorigo et al., 1991), Harmony Search (HS) (Geem et al., 2001), Ant

* Corresponding author.
E-mail address: sukanta1122@gmail.com (S. Nama)

doi: 10.5267/j.dsl.2016.2.004

Colony Optimization (ACO) (Blum, 2005), Teaching Learning Based Optimization (TLBO) (Rao et al. 2011), Water Cycle Algorithm (WCA) (Eskandar et al., 2012) etc. GA is a search and optimization techniques that mimics the natural law of evolution and chromosomal processing in genetics. DE is another evolutionary optimization algorithm imitating the Darwin theory of evolution. PSO is based on the behavior of bird flocking or fish schooling. ABC uses the foraging behavior of a honey bee, ACO implemented based on the searching behavior of ant from the source to the destination, and HS is related to the improvisation process of a musician. Some of the applications of these algorithms are given in references (Chander et al., 2011; Wang et al., 2012; Canelas et al., 2013; Hecker et al., 2014; Ghasemi et al., 2014; Li & He, 2014; Baykasoğlu et al., 2014; Nama et al., 2015; Rao, 2016; Bolañosa et al., 2015; Mohammadia et al., 2015; Bhunia et al., 2015; Hosseini et al., 2014; Aulady, 2013; Nama et al., 2016; Rao & Patel, 2014; Barati et al., 2016; Eshraghi, 2016; Rout et al., 2016; Mir & Rezaeian, 2016; Abido, 2016; Gen et al., 2016; Aickelin & Dowsland, 2014; Hecker et al., 2013).

Recently, a new metaheuristic optimization algorithm Symbiotic Organisms Search (SOS), proposed by Min-Yuan and Doddy (2014), is based on the interactions relationship between two organisms in ecosystems. Some of the application of this algorithm has been found in literature which are given in references (Prasad and Mukherjee 2015; Abdullahi and Ngadi 2016; Cheng et al. 2015; Kavousi-Fard et al. 2015; Eki et al. 2015; Verma et al. 2015). In ecosystem predation is a biological interaction where a predator (an organism that is hunting) feeds on its prey (the organism that is attacked). Predators may or may not kill their prey prior to feeding on them, but the act of predation often results in the death of its prey and the ultimate absorption of the prey's tissue through consumption.

Therefore, in the present work, the authors suggest the predation phase to improve the performance of the algorithm and introduce the random weighted reflection vector to enhance the search ability of the SOS algorithm influenced by Satapathy and Naik (2012), which used the random weighted difference vector to improve the performance of TLBO.

The remaining portion of this paper is organized as follows: Section 2 presented the reviews of the original concept of SOS. The new improve SOS (ISOS) is presented in Section 3. Section 4 and 5 presented the result and discussion on solving unconstrained and real world optimization problem. Finally, section 6 summarizes the conclusion of the paper on the whole study.

2. The SOS algorithm

Symbiotic Organisms Search (SOS) is a comparatively new algorithm which simulates the interactive behaviour of the organisms in nature. The mostly common symbiotic relations between the organisms in ecosystem are mutualism, commensalism, and parasitism. Mutualism is a symbiosis relationship in which both organisms benefit. Commensalism is symbiosis in which one organism benefits and the other is not harmed or helped. If in a interaction one organism benefits and other organism harmed, the relation is called Parasitism. Based on the principle of biological interaction in ecosystem mutualism phase, commensalism phase, and parasitism phase are developed.

In SOS, the initial population called the ecosystem and in ecosystem a group of organisms is generated randomly within the search space. The candidate solution of the problem is the organism of the ecosystem and the fitness value of each organism, reflects the degree of adaptation to the desired objective. In SOS, the new solution is generated by executing the mutualism phase, commensalism phase, and parasitism phase. The descriptions of these three phases are given below.

Mutualism phase: In this phase, an organism X_i to interact an organism X_j which select randomly from the ecosystem. Both the organisms try to increases of their mutual survival advantage in the ecosystem. The new candidate solutions X_i^{new} and X_j^{new} for the organism X_i and X_j, is calculated by Eqn. (1) and Eqn. (2).

$$X_i^{new} = X_i + rand(0,1) \times (X_{best} - Mutual_{Vector} \times BF1) \quad (1)$$

$$X_j^{new} = X_j + rand(0,1) \times (X_{best} - Murual_{Vector} \times BF2) \quad (2)$$

where

$$Mutual\_Vector = \frac{X_i + X_j}{2} \quad (3)$$

Here, the value of benefit factors BF1 and BF2 is either 1 or 2 which represent level of benefit to each organisms, i.e., whether an organism is benefitted partially or fully from the interaction. The ‘‘Mutual_Vector’’ signifies the relationship characteristic between organism X_i and X_j. And Xbest represents an organism with best objective function value in the ecosystem.

Commensalism phase: Since in this phase X_i attempts to increase the benefits from X_j, new candidate solution X_i^{new} is calculated by the commensal symbiosis relationship between organisms X_i and X_j, according to the Eq. (4).

$$X_i^{new} = X_i + rand(-1,1) * \left(X_{best} - X_j\right) \quad (4)$$

Parasitism phase: In SOS, organism X_i is given a role similar to the anopheles mosquito through the creation of an artificial parasite called ‘‘Parasite_Vector’’. By modifying the randomly selected dimensions of organism X_i, Parasite_Vector is created. Organism X_j is chosen randomly from the ecosystem and serves as a host to the parasite vector. Parasite_Vector attempts to replace X_j in the ecosystem. Both organisms are then evaluated to measure their fitness. If Parasite_Vector has a better fitness value, it will kill organism X_j and assume its position in the ecosystem. If the fitness value of X_j is better, X_j will have immunity from the parasite and the Parasite_ Vector will no longer be able to live in that ecosystem.
The implementation steps of SOS algorithm are as follows:

Step 1: Initialized each organism with uniform random generation and evaluate the objective function value in the ecosystem.

Step2: Generation

Step 2.1: Evaluate new organisms by mutualism phase and update in the ecosystem which are calculated by Eqn.(1) and (2).

Step 2.2: Evaluate new organisms by commensalism phase and update in the ecosystem which are calculated by using Eqn. (4).

Step 2.3: Evaluate the new organisms by parasitism phase (given in section 2) and update them in the ecosystem.

Step 3: Check the termination criterion, if it is fails, go to Step 2; otherwise the best objective function value is considered as the required solution.

3. I-SOS

In this section, the proposed Predation phase and Random weighed reflection vector are discussed briefly.

3.1 Predation phase

In ecology, predation is a biological interaction where a predator (an organism that's searching) feeds on its prey (the organism that is attacked). Predators may or would possibly not kill their prey previous to feeding on them, but the act of predation regularly outcome in the death of its prey and the eventual absorption of the prey's tissue by means of consumption. Predation and parasitism, in both cases, one organism is harmed and the other is benefited. On the other hand, predators almost always kill and eat their prey, while not all parasites kill their hosts. In this phase, one predator generated by Organism X_i, called the Predation_Vector. The Predation_Vector is generated by Eq. (5).

$$Predation\_Vector = X_i + rand(0,1) * (X_i^{max} - X_i^{min}) \tag{5}$$

where X_i^{max} and X_i^{min} are the maximum and minimum value of dimension of organism X_i. Then the worst organisms in the ecosystem are replaced by the Predation_Vector. In this work the predator size (i.e. the number of predator in the ecosystem) is chosen as 4.

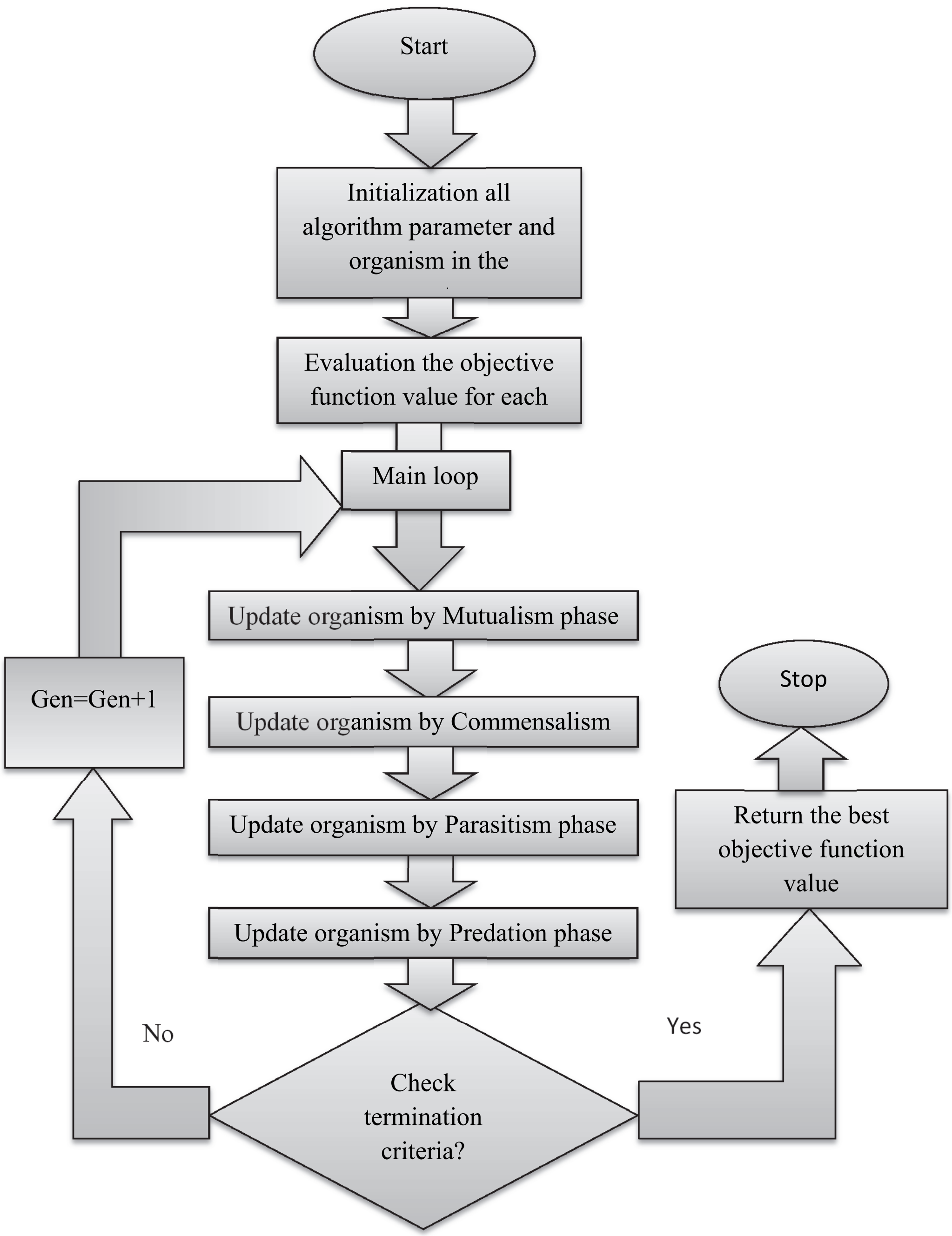

Fig. 1. Pseudocode of the proposed algorithm

3.2 Random weighted reflection vector

Satapathy and Naik (2012) used the random weighted differential vector (Satapathy and Naik, 2012) in a random manner in the range (0.5, 1) by using the Eqn. (6)

$$RWDV = 0.5 \times (1 + \text{rand }(1, D)), \tag{6}$$

where rand (0, 1) is a uniformly distributed random number within the range [0, 1]. Therefore, the mean value of this weighted differential scale factor is 0.75. In this work, the authors suggested the random weighted reflection vector (RWRV) to enhance the search ability of the algorithm which is calculated by Eq. (7).

$$RWRV = 1 - 0.5 \times (1 + rand(1, D)) \tag{7}$$

So the new sets of Mutualism phase & Commensalism phase which are formulated by Eq. (1), Eq. (2), and Eq. (4) can be modified by using the following equation:

$$X_i^{new} = X_i + RWRV * (X_{best} - Mutual\_Vector * BF1) \tag{8}$$
$$X_j^{new} = X_j + RWRV * (X_{best} - Murual\_Vector * BF2) \tag{9}$$
$$\text{and } X_i^{new} = X_i + RWRV * \left(X_{best} - X_j\right) \tag{10}$$

If the organism X_i^{New} violates the boundary constraint, violating organism is reflected back from the violated boundary using the following rule:

$$X_i^{new} = \begin{cases} 0.5 * \text{rand}(0,1) * (X_i^{new} - \text{ub}) & if\ X_i^{new} > ub \\ 0.5 * \text{rand}(0,1) * (\text{ub} + X_i^{new}) & if\ X_i^{new} < lb \end{cases} \tag{11}$$

The Pseudo code of the I-SOS algorithm for solving benchmark functions are shown in Fig.1 and the algorithm steps can be summarized in the following way:

Step 1: Randomly initialize the common control parameter i.e. Eco-size (number of organisms in the ecosystem); function evaluation; the ecosystem organisms and evaluate the fitness of each organism.

Step 2: Calculate the new organisms and update by mutualism phase using Eq. (8) and Eq. (9) and repair the infeasible organisms of the ecosystem to be feasible using En. (11).

Step 3: Calculate the new organisms and update by commensalism phase using Eq. (4) and repair the infeasible organisms of the ecosystem to be feasible using Eq. (10).

Step 4: Calculate the new organisms and update by parasitism phase which is given in Section 2.

Step 5: Update organisms by predation phase in Eq. (5) and repair the infeasible organisms of the ecosystem to the organisms to be feasible using Eq. (11).

Step 6: Replace the worst organism by Predation_Vector.

Step 7: If the stopping criterion is not satisfied go to Step 2, until the best fitness value is obtained.

Table 1
Benchmark functions for Experiment-1(D: Dimension, Fmin: Global optimum value)

Function	Formulation of Objective Function	Search Space	Fmin
F1. Sphere	$f(x)=\sum_{i=1}^{D} x_i^2$	[-100,100]	0
F2. Schwefel2.22	$f(x)=\sum_{i=1}^{D}\lvert x_i\rvert+\prod_{i=1}^{D}\lvert x_i\rvert$	[-10, 10]	0
F3. Schwefel1.2	$f(x)=\sum_{i=1}^{D}\sum_{j=1}^{i} x_i^2$	[-100, 100]	0
F4. Schwefel2.21	$f(x)=\max\{\lvert x_i\rvert, 1\le i\le D\}$	[-100, 100]	0
F5. Rosenbrock	$f(x)=\sum_{i=1}^{D}[100(x_{i+1}-x_i^2)^2+(x_i-1)^2]$	[-2.048,2.048]	0
F6. Step	$f(x)=\sum_{i=1}^{D}(x_i+0.5)^2$	[-100, 100]	0
F7. Quartic	$f(x)=\sum_{i=1}^{D} i x_i^4+random(0,1)$	[-1.28, 1.28]	0
F8. Rastrigin	$f(x)=10D+\sum_{i=1}^{D}[x_i^2-10\cos(2\pi x_i)]$	[-5.12,5.12]	0
F9. Ackley	$f(x)=20+e-20e^{\frac{1}{D}(\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^2})}-e^{\frac{1}{D}(\sum\cos(2\pi x_i))}$	[-32.768,32.768]	0
F10. Griewank	$f(x)=\sum_{i=1}^{D}\frac{x_i^2}{4000}-\prod_{i=1}^{D}\cos(\frac{x_i}{\sqrt{i}})-1$	[-600, 600]	0
F11. Penalized1	$f(x)=\frac{\pi}{D}\left\{\begin{array}{l}10\sin^2(\pi y_i)+\sum_{i=1}^{D-1}(y_i-1)^2[1+10\sin^2(3\pi y_{i+1})\\+(y_D-1)^2]\end{array}\right\}+\sum_{i=1}^{D}u(x_i,10,100,4)$ where $y_i=1+\frac{1}{4}(x_i+1), u(x_i,a,k,m)=\begin{cases}k(x_i-a)^m & x_i>a\\ 0 & -a<x_i<a\\ k(-x_i-a)^m & x_i<a\end{cases}$	[-50, 50]	0
F12. Penalized2	$f(x)=0.1\left\{\begin{array}{l}10\sin^2(\pi x_i)+\sum_{i=1}^{D-1}(x_i-1)^2[1+10\sin^2(3\pi x_{i+1})\\+(x_D-1)^2[1+\sin^2(2\pi x_D)]]\end{array}\right\}+\sum_{i=1}^{D}u(x_i,5,100,4)$	[-50, 50]	0

4. Experimental Result and Discussion

Three experiments are considered for the investigation of the validity of the proposed algorithms. All the benchmark function for these experiments is given in Table 1, Table 2 and Table 3 respectively.

4.1. Effect of the Eco-size

Experiment l: The effects of Eco-size (number of organism in the ecosystem) of the algorithm are discussed in this section by taking different values of Eco-size. Here the values of Eco-size are considered as 10, 20, 30, 40, and 50 respectively. Twenty five independent runs are carried out for each eco-size in each problem. The best, mean and standard deviation of the function error values $f(\vec{x}) - f(\overrightarrow{x^*})$ among 25 runs are recorded on each problem, where $\vec{x}$ the best solution is found by the algorithm in a run and $\overrightarrow{x^*}$ is the global optimum of the test problem. A run is successful if its function error value is not larger than the target error accuracy level ε, which is set to 10^{-8} (Suganthan et al. 2005). We record the number of successful runs and report the success rate, the percentage of the successful runs among 25 runs.

Table 2

Benchmark functions for Experiment-ll (D: Dimension, Fmin: Global optimum value)

Function	Formulation of Objective Function	D	Search Space	Fmin
F1. Sphere	$f(x) = \sum_{i=1}^{D} x_i^2$	30	[-100,100]	0
F2. Rastrigin	$f(x) = 10D + \sum_{i=1}^{D} [x_i^2 - 10\cos(2\pi x_i)]$	30	[-5.12,5.12]	0
F3. Six Hump Camel back	$f(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	[-5, 5]	-1.031628
F4. Step	$f(x) = \sum_{i=1}^{D} (x_i + 0.5)^2$	30	[-100, 100]	0
F5. Rosenbrock	$f(x) = \sum_{i=1}^{D} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	[-2.048,2.048]	0
F6. Ackley	$f(x) = 20 + e - 20e^{\frac{1}{D}(\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2})} - e^{\frac{1}{D}(\sum \cos(2\pi x_i))}$	30	[-32.768,32.768]	0
F7. Griewank	$f(x) = \sum_{i=1}^{D} \frac{x_i^2}{4000} - \prod_{i=1}^{D} \cos(\frac{x_i}{\sqrt{i}}) - 1$	30	[-600, 600]	0
F8. Salomon	$f(x) = 1 - \cos(2\pi\sqrt{\sum_{i=1}^{D} x_i^2}) + 0.1\sqrt{\sum_{i=1}^{D} x_i^2}$	30	[-100, 100]	0
F9. Quartic	$f(x) = \sum_{i=1}^{D} ix_i^4 + random(0,1)$	30	[-1.28, 1.28]	0
F10. Alpine	$f(x) = \sum_{i=1}^{D} \lvert x_i \sin(x_i) + 0.1x_i \rvert$	30	[-10, 10]	0
F11. Branin	$f(x) = (x_2^2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi})\cos x_1 + 1$	2	[-5, 5]	0
F12. Easom		2	[-100, 100]	0
F13. Goldstein and Price	$f(x) = [1 + (x_1 + x_2 + 1)^2(10 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2^2)(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	2	[-2, 2]	3
F14. Shubert	$f(x) = (\sum_{i=1}^{5} i\cos((i+1)x_1 + i)) \times (\sum i\cos((i+1)x_2 + i))$	2	[-10, 10]	-186.7309
F15. Hartmann	$f(x) = -\sum_{i=1}^{4} \alpha_i \exp(-\sum_{j=1}^{3} a_{ij}(x_j - b_{ij})^2)$	3	[0, 1]	-3.86278

Table 2
Benchmark functions for Experiment-ll (D: Dimension, Fmin: Global optimum value) (Continued)

F16. Levy	$f(x)=\sin^2(\pi x_1)+\sum_{i=1}^{D-1}(x_i-1)^2[1+10\sin^2(\pi x_i+1)]$ $+(x_D-1)^2[1+\sin^2(2\pi x_D)]$ where $x_i=1+\frac{1}{4}(x_i-1)$, i=1, 2,……….D	30	[-10, 10]	0
F17. Michalewicz	$f(x)=-\sum_{i=1}^{D}\sin(x_i)\sin^{2m}(\frac{ix_i^2}{\pi})$, m=10.	10	[0, π]	-9.66015
F18. Schaffers	$0.5+\frac{\sin^2(x_1^2+x_2^2)-0.5}{[1+0.001(x_1^2+x_2^2)]^2}$	2	[-100, 100]	0
F19. Schwefel1.2	$f(x)=\sum_{i=1}^{D}\sum_{j=1}^{i}x_i^2$	30	[-100, 100]	0
F20. Kowalik	$f(x)=\sum_{i=1}^{11}[a_i-\frac{x_1(b_i^2+b_ix_2)}{b_i^2-b_ix_3-x_4}]^2$	4	[-5, 5]	0.0003075
F21. Shekel5	$f(x)=-\sum_{i=1}^{5}[(x-a_i)(x-a_i)^T+c_i]^{-1}$	4	[0, 10]	-10.1532
F22. Penalized1	$f(x)=\frac{\pi}{D}\left\{\begin{array}{l}10\sin^2(\pi y_i)+\sum_{i=1}^{D-1}(y_i-1)^2[1+10\sin^2(3\pi y_{i+1})\\+(y_D-1)^2]\end{array}\right\}$ $+\sum_{i=1}^{D}u(x_i,10,100,4)$ Where $y_i=1+\frac{1}{4}(x_i+1)$, $u(x_i,a,k,m)=\begin{cases}k(x_i-a)^m & x_i>a\\ 0 & -a<x_i<a\\ k(-x_i-a)^m & x_i<a\end{cases}$	30	[-50, 50]	0
F23. Penalized2	$f(x)=0.1\left\{\begin{array}{l}10\sin^2(\pi x_i)+\sum_{i=1}^{D-1}(x_i-1)^2[1+10\sin^2(3\pi x_{i+1})\\+(x_D-1)^2[1+\sin^2(2\pi x_D)]]\end{array}\right\}$ $+\sum_{i=1}^{D}u(x_i,5,100,4)$	30	[-50, 50]	0
F24. Schwefel2.21	$f(x)=\max\{\lvert x_i\rvert,1\le i\le D\}$	30	[-100, 100]	0
F25. Schwefel2.22	$f(x)=\sum_{i=1}^{D}\lvert x_i\rvert+\prod_{i=1}^{D}\lvert x_i\rvert$	30	[-10, 10]	0

The effect of Eco-size, justified on 12 different benchmark functions which are given in Table 1. Table 4, 5 and 6 show the effect of Eco-size of these 12 benchmark functions for dimension (D) 3, 5, and 7 respectively with D*10000 function evaluations.

Table 3
Benchmark functions for Experiment-lll (D: Dimension, Fmin: Global optimum value)

Function	Formulation of Objective Function	D	Search Space	Fmin
F1. Sphere	$f(x)=\sum_{i=1}^{D} x_i^{2}$	50	[-100, 50]	0
F2. Rosenbrock	$f(x)=\sum_{i=1}^{D}[100(x_{i+1}-x_i^{2})^2+(x_i-1)^2]$	50	[-2.048, 2.048]	0
F3. Ackley	$f(x)=20+e-20e^{\frac{1}{D}(\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^{2}})}-e^{\frac{1}{D}(\sum\cos(2\pi x_i))}$	50	[-32.768, 16]	0
F4. Griewank	$f(x)=\sum_{i=1}^{D}\frac{x_i^{2}}{4000}-\prod_{i=1}^{D}\cos(\frac{x_i}{\sqrt{i}})-1$	50	[-600, 200]	0
F5. Rastrigin	$f(x)=10D+\sum_{i=1}^{D}[x_i^{2}-10\cos(2\pi x_i)]$	50	[-0.5, 0.2]	0
F6. Rastrigin_noncont	$f(x)=\sum_{i=1}^{D}[y_i^{2}-10\cos(2\pi y_i)+10]$ where $y_i=\begin{cases} x_i & \lvert x_i\rvert<\frac{1}{2} \\ \frac{rand(2x_i)}{2} & \lvert x_i\rvert\geq\frac{1}{2}\end{cases}$	50	[-5.12, 2]	0
F7. Schwefel	$f(x)=418:9829\text{n}-\sum x_i\sin(\sqrt{\lvert x_i\rvert})$	50	[-600, 600]	0
F8. Dixon and Price function.	$f(x)=\sum_{i=1}^{D}i(2x^{2}{}_{i+1}-x_i)^2+(x_1-1)^2$	50	[-10, 10]	0
F9. Levy function	$f(x)=\sin^2(\pi x_1)+\sum_{i=1}^{D-1}(x_i-1)^2[1+10\sin^2(\pi x_i+1)]$ $+(x_D-1)^2[1+\sin^2(2\pi x_D)]$ $x_i=1+\frac{1}{4}(x_i-1)$, i=1, 2,………..D	50	[-10, 10]	0
F10. Sum Squares function	$f(x)=\sum_{i=1}^{D}ix_i^{2}$	50	[-10, 10]	0
F11. Zakharov	$f(x)=\sum_{i=1}^{D}x_i^{2}+(\sum_{i=1}^{D}\frac{ix_i}{2})^2+(\sum_{i=1}^{D}\frac{ix_i}{2})^2$	50	[-5.12, 5.12]	0

In these tables, the boldface represents the best result found after reaching the maximum number of function evaluation. From Table 4, it is seen that strategy with Eco-size 10 produce the best result for function F1, F8, F10, F11, and F12; strategy with Eco-size 20 produce the best result for function F5 and F9 but for function F2 strategy with Eco-size 20 and 30 produce the identical result; for function F6 strategy with Eco-size 40 produce the best result; for function F4 and F7 strategy with Eco-size 50 produce the best result but for function F3 produce the identical result and hence there is no effect of Eco-size for this function.

Table 4

Effect of the eco-size of 12 different benchmark function (given in Table 1) with dimension 3 and function evaluation D×10000

Function		eco-size =10	eco-size =20	eco-size =30	eco-size =40	eco-size =50
F1	Best	7.34E-73	7.79E-62	2.20E-03	2.42E-56	1.37E-42
	Mean	**1.09E-57**	2.61E-48	6.77E-37	3.84E-31	3.02E-26
	SD	4.44E-57	1.27E-47	2.99E-36	1.33E-30	1.23E-25
F2	Best	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Mean	3.98E-02	**0.00E+00**	**0.00E+00**	1.28E-14	2.30E-12
	SD	1.99E-01	0.00E+00	0.00E+00	6.32E-14	1.15E-11
F3	Best	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Mean	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	SD	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F4	Best	8.04E-09	3.44E-11	1.55E-10	4.76E-10	4.06E-10
	Mean	7.51E-03	1.91E-04	1.70E-02	2.83E-03	**8.48E-05**
	SD	2.27E-02	6.19E-04	8.45E-02	1.21E-02	1.61E-04
F5	Best	8.88E-16	8.88E-16	8.88E-16	8.88E-16	8.88E-16
	Mean	2.31E-15	**8.88E-16**	1.03E-15	1.31E-15	1.05E-13
	SD	1.78E-15	0.00E+00	1.03E-15	1.18E-15	4.80E-13
F6	Best	0.00E+00	5.41E-05	0.00E+00	0.00E+00	0.00E+00
	Mean	1.57E-02	1.61E-02	1.27E-02	**7.51E-03**	1.03E-02
	SD	1.03E-02	1.29E-02	1.01E-02	7.32E-03	8.48E-03
F7	Best	2.79E-05	2.29E-05	4.54E-05	1.61E-05	4.80E-05
	Mean	3.64E-04	3.71E-04	4.04E-04	4.02E-04	**3.47E-04**
	SD	3.75E-04	2.60E-04	2.08E-04	2.91E-04	2.44E-04
F8	Best	7.27E-71	3.06E-63	2.80E-61	2.16E-51	1.60E-56
	Mean	**1.19E-57**	4.37E-49	3.48E-39	1.55E-31	4.36E-27
	SD	5.91E-57	1.74E-48	7.71E-39	5.36E-31	1.41E-26
F9	Best	5.08E-26	1.49E-26	5.56E-25	1.65E-24	3.05E-23
	Mean	2.83E-06	**3.48E-21**	1.80E-17	1.72E-13	9.75E-17
	SD	1.32E-05	1.05E-20	8.55E-17	8.35E-13	2.25E-16
F10	Best	1.48E-23	6.84E-28	4.71E-25	6.24E-23	6.02E-25
	Mean	**4.78E-09**	4.39E-04	1.29E-08	4.74E-04	2.50E-08
	SD	2.38E-08	2.20E-03	6.47E-08	2.20E-03	1.25E-07
F11	Best	3.38E-34	2.63E-32	3.35E-28	4.31E-26	1.46E-26
	Mean	**9.33E-29**	5.15E-24	3.53E-19	1.81E-15	1.02E-13
	SD	9.33E-29	1.32E-23	1.03E-18	6.44E-15	3.10E-13
F12	Best	1.09E-35	6.36E-33	8.37E-30	3.64E-27	2.34E-30
	Mean	**1.89E-29**	3.08E-25	2.46E-20	8.86E-19	3.65E-14
	SD	6.73E-29	8.30E-25	5.80E-20	2.52E-18	1.71E-13

Table 5

Effect of the eco-size of 12 different benchmark function given in Table 1) with dimension 5 and function evaluation D×10000

Function		eco-size =10	eco-size =20	eco-size =30	eco-size =40	eco-size =50
F1	Best	2.01E-99	1.11E-87	8.59E-71	3.23E-54	1.09E-63
	Mean	**4.77E-87**	1.02E-71	7.49E-55	5.90E-44	7.90E-36
	SD	4.77E-87	3.60E-71	2.82E-54	2.62E-43	3.44E-35
F2	Best	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Mean	1.99E-01	1.99E-01	**1.32E-01**	1.68E-01	1.44E-01
	SD	4.97E-01	4.98E-01	3.66E-01	4.88E-01	3.84E-01
F3	Best	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Mean	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	SD	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F4	Best	5.66E-10	1.01E-08	1.13E-08	1.43E-07	2.38E-07
	Mean	7.52E-01	3.99E-01	4.27E-01	2.18E-01	**1.00E-01**
	SD	1.12E+00	7.17E-01	9.09E-01	4.33E-01	2.16E-01
F5	Best	8.88E-16	8.88E-16	8.88E-16	8.88E-16	8.88E-16
	Mean	4.30E-15	2.59E-15	**2.31E-15**	**2.31E-15**	2.45E-15
	SD	7.11E-16	1.81E-15	1.78E-15	1.78E-15	1.80E-15
F6	Best	0.00E+00	0.00E+00	9.81E-04	2.33E-14	7.46E-03
	Mean	5.94E-02	4.79E-02	5.44E-02	4.39E-02	**3.86E-02**
	SD	4.48E-02	3.69E-02	4.54E-02	3.22E-02	2.77E-02
F7	Best	5.89E-05	6.02E-05	3.28E-05	4.85E-06	4.22E-05
	Mean	2.71E-04	**2.42E-04**	3.61E-04	3.49E-04	3.98E-04
	SD	1.88E-04	1.34E-04	2.24E-04	2.54E-04	2.39E-04
F8	Best	1.98E-99	7.46E-82	4.40E-70	1.39E-60	5.87E-53
	Mean	**2.88E-86**	7.87E-72	2.28E-53	1.65E-43	3.46E-37
	SD	1.41E-85	2.76E-71	8.78E-53	5.12E-43	1.19E-36
F9	Best	4.48E-02	1.95E-31	5.59E-29	1.35E-23	5.01E-22
	Mean	2.66E-05	**1.53E-22**	3.35E-18	3.06E-19	5.61E-09
	SD	8.77E-05	4.57E-22	1.62E-17	6.53E-19	2.80E-08
F10	Best	8.77E-05	1.86E-31	4.78E-27	2.24E-25	1.13E-23
	Mean	8.77E-05	5.78E-04	1.34E-05	4.12E-06	**2.13E-07**
	SD	2.20E-03	2.83E-03	6.60E-05	1.48E-05	1.06E-06
F11	Best	6.19E-46	1.29E-40	3.32E-34	8.04E-27	3.21E-25
	Mean	**1.65E-39**	1.29E-33	8.86E-26	6.92E-21	5.20E-18
	SD	6.00E-39	1.29E-33	3.83E-25	2.08E-20	1.49E-17
F12	Best	6.79E-48	3.37E-41	2.02E-37	3.15E-35	1.67E-29
	Mean	**3.59E-44**	2.63E-36	5.25E-28	3.74E-23	4.33E-20
	SD	7.44E-44	9.25E-36	1.34E-27	8.68E-23	1.20E-19

From Table 5, it is seen that strategy with Eco-size 10 produce the best result for function F1, F8, F11 and F12 but for function F3 produce the identical result and hence there is no effect of Eco-size for this function; for function F7 and F9 strategy with Eco-size 20 produce the best result; for function F2 strategy with Eco-size 30 produce the best result but for function F5 produce the identical result with the strategy of Eco-size 40; for function F6 and F10 strategy with Ecosize 50 produce the best result.

Table 6

Effect of the population size of 12 different benchmark function (given in Table 1) with dimension 7 and function evaluation D×10000

Function		eco-size =10	eco-size =20	eco-size =30	eco-size =40	eco-size =50
F1	Best	1.88E-128	3.29E-106	1.53E-81	4.30E-68	4.30E-68
	Mean	**5.13E-119**	3.52E-95	9.91E-72	2.28E-56	2.28E-56
	SD	2.45E-118	1.45E-94	3.15E-71	6.36E-56	6.36E-56
F2	Best	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Mean	8.88E-01	**7.55E-01**	1.10E+00	9.62E-01	9.62E-01
	SD	7.28E-01	9.01E-01	1.07E+00	1.36E+00	1.36E+00
F3	Best	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Mean	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	SD	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F4	Best	1.98E-05	1.40E-11	1.21E-08	3.82E-10	3.82E-10
	Mean	2.47E+00	1.17E+00	6.62E-01	**5.98E-01**	**5.98E-01**
	SD	2.41E+00	1.54E+00	1.13E+00	9.58E-01	9.58E-01
F5	Best	8.88E-16	8.88E-16	8.88E-16	8.88E-16	8.88E-16
	Mean	4.30E-15	**4.01E-15**	**4.01E-15**	4.16E-15	4.16E-15
	SD	7.11E-16	1.18E-15	1.18E-15	9.84E-16	9.84E-16
F6	Best	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Mean	7.40E-02	**4.47E-02**	5.79E-02	5.32E-02	5.32E-02
	SD	5.86E-02	5.85E-02	6.90E-02	4.01E-02	4.01E-02
F7	Best	5.03E-05	3.89E-05	4.14E-05	1.29E-05	1.29E-05
	Mean	4.11E-04	2.93E-04	**2.53E-04**	3.07E-04	3.07E-04
	SD	2.95E-04	1.97E-04	1.57E-04	1.76E-04	1.76E-04
F8	Best	1.68E-129	1.50E-103	2.91E-81	9.61E-71	9.61E-71
	Mean	**6.98E-118**	2.38E-92	1.96E-70	1.35E-56	1.35E-56
	SD	2.41E-117	1.18E-91	6.96E-70	4.73E-56	4.73E-56
F9	Best	7.04E-08	6.94E-32	1.51E-29	1.07E-24	1.07E-24
	Mean	2.22E-02	9.48E-10	5.81E-08	**6.91E-19**	**6.91E-19**
	SD	1.10E-01	4.20E-09	2.90E-07	3.22E-18	3.22E-18
F10	Best	1.70E-07	4.24E-28	1.92E-29	7.62E-25	7.62E-25
	Mean	3.50E-03	3.94E-03	**1.48E-03**	1.85E-03	1.85E-03
	SD	6.64E-03	5.36E-03	3.67E-03	4.02E-03	4.02E-03
F11	Best	1.07E-57	1.40E-48	1.96E-37	2.79E-33	2.79E-33
	Mean	**4.58E-54**	1.47E-43	1.37E-32	1.52E-26	1.52E-26
	SD	7.76E-54	4.37E-43	3.42E-32	3.38E-26	3.38E-26
F12	Best	2.96E-63	2.73E-52	8.98E-43	3.29E-35	3.29E-35
	Mean	**9.72E-59**	4.53E-47	1.08E-35	6.69E-29	6.69E-29
	SD	2.79E-58	1.20E-46	4.05E-35	2.82E-28	2.82E-28

Table 7

No of fitness function evolution required to obtain the accuracy level of 12 benchmark function (given in Table 1) with dimension 3 and function evaluation D×10000

Function		eco-size =10	eco-size =20	eco-size =30	eco-size =40	eco-size =50
F1	Mean(SD)	2522.00(1051.73)	**2439.20(1078.59)**	2823.60(642.05)	3547.20(1001.25)	4178.00(1517.54)
	SR	100	100	100	100	100
F2	Mean(SD)	5821.20(1660.50)	**4224.80(1665.03)**	6020.40(2226.19)	7438.40(3336.20)	8898.00(3200.67)
	SR	100	100	100	100	100
F3	Mean(SD)	2717.20(1187.19)	**2439.20(899.56)**	3150.00(1032.28)	3892.80(1587.90)	4354.00(1144.73)
	SR	100	100	100	100	100
F4	Mean(SD)	6102.80(1934.62)	**5716.00(1951.96)**	7806.00(3040.53)	9012.80(3434.70)	10738.00(4065.33)
	SR	100	100	100	100	100
F5	Mean(SD)	26170.00(0.00)	26260.00(1810.19)	**19890.00(5643.40)**	26920.00(0.00)	NaN(NaN)
	SR	4	8	16	4	0
F6	Mean(SD)	**390.80(135.21)**	698.40(266.45)	903.60(238.80)	1012.80(312.99)	1218.00(280.95)
	SR	100	100	100	100	100
F7	Mean(SD)	NaN(NaN)	NaN(NaN)	NaN(NaN)	NaN(NaN)	NaN(NaN)
	SR	0	0	0	0	0
F8	Mean(SD)	9552.40(5621.27)	8512.80(5751.46)	**6327.60(3340.14)**	9723.48(5765.02)	11586.00(8262.11)
	SR	100	100	100	92	100
F9	Mean(SD)	6330.00(2183.82)	**5997.60(2399.95)**	6913.20(2649.73)	7323.20(2881.90)	10946.00(4632.18)
	SR	100	100	100	100	100
F10	Mean(SD)	4690.00(2998.13)	7220.00(0.00)	**21390.00(6000.00)**	14280.00(3846.66)	16450.00(16970.56)
	SR	8	4	12	8	8
F11	Mean(SD)	10613.33(4624.86)	8916.00(3381.20)	**8113.20(3005.37)**	8769.60(2395.59)	11570.00(3059.41)
	SR	98	100	100	100	100
F12	Mean(SD)	16287.65(7546.98)	**10649.57(4429.54)**	11080.91(4752.03)	12450.43(6217.74)	12902.17(6305.33)
	SR	63	92	88	92	92

Here "Mean (SD)" represents the statistical result of fitness evolution requires, "SR" represents the success rate.

From Table 6 it is observed that for function F1, F8, F11 and F12 strategy with Eco size 10 produce the best result but for function F3 produce the identical result; for function F2 and F6 strategy with Eco size 20 produce the best result but for function F5 produce the identical result with the strategy of Eco-size 30; for function F7 and F10 strategy with Eco size 30 produce the best result; for function F4 and F9 strategy with Eco size 40 produce the identical result with the strategy of Eco-size 50. Table 7, 8, and 9 present the number of fitness evolution and number of successful run to obtain the accuracy level ε (1e-08) (Suganthan et al., 2005). In this table the boldface represents the minimum function evaluation require to obtain the accuracy level ε. From Table 7, it is observed that strategy with Eco-size 10 requires minimum function evaluation on one function; strategy with Eco-size 20 requires minimum function evaluation on six functions; strategy with Eco-size 30 requires minimum function evaluation on four functions.

Table 8

No of fitness function evolution required to obtain the accuracy level of 12 benchmark function (given in Table 1) with dimension 5 and function evaluation D×10000

Function		eco-size =10	eco-size =20	eco-size =30	eco-size =40	eco-size =50
F1	Mean(SD)	3862.80(1186.74)	**3623.20(1455.56)**	4014.00(1528.92)	4968.00(1425.11)	6834.00(2431.68)
	SR	100	100	100	100	100
F2	Mean(SD)	7686.80(1421.08)	**7661.60(2447.65)**	8756.40(2603.27)	11022.40(4441.47)	12354.00(3548.30)
	SR	100	100	100	100	100
F3	Mean(SD)	**3805.20(1419.31)**	3898.40(1264.30)	4762.80(1736.92)	5979.20(1840.77)	7234.00(2641.92)
	SR	100	100	100	100	100
F4	Mean(SD)	**8518.80(1462.40)**	9965.60(2381.12)	10897.20(3752.45)	13467.20(4238.72)	17346.00(4948.10)
	SR	100	100	100	100	100
F5	Mean(SD)	NaN(NaN)	NaN(NaN)	NaN(NaN)	48360.00(0.00)	NaN(NaN)
	SR	0	0	0	4	0
F6	Mean(SD)	**762.00(291.20)**	992.80(252.64)	1258.80(262.18)	1460.80(334.09)	1778.00(443.02)
	SR	100	100	100	100	100
F7	Mean(SD)	NaN(NaN)	NaN(NaN)	NaN(NaN)	NaN(NaN)	NaN(NaN)
	SR	0	0	0	0	0
F8	Mean(SD)	31403.68(12525.88)	18120.00(12457.95)	21360.00(13274.08)	**22356.52(10878.81)**	24872.22(9959.42)
	SR	76	64	64	92	72
F9	Mean(SD)	**7338.00(1500.93)**	8436.00(2401.78)	10292.40(2965.63)	11726.40(3745.64)	12674.00(5321.75)
	SR	100	100	100	100	100
F10	Mean(SD)	NaN(NaN)	**9460.00(6561.95)**	NaN(NaN)	46440.00(0.00)	NaN(NaN)
	SR	0	8	0	4	0
F11	Mean(SD)	34858.00(10518.19)	20320.80(9181.70)	**17818.80(5048.72)**	18241.60(4423.76)	18578.00(4215.40)
	SR	20	100	100	100	100
F12	Mean(SD)	39530.00(15388.10)	22740.00(10010.99)	18246.00(6770.74)	**19971.43(7133.30)**	19965.79(6683.42)
	SR	12	80	80	84	76

Here "Mean (SD)" represents the statistical result of fitness evolution requires, "SR" represents the success rate.

Table 9

No of fitness function evolution required to obtain the accuracy level of 12 benchmark function (given in Table 1) with dimension 7 and function evaluation D×10000

Function		eco-size =10	eco-size =20	eco-size =30	eco-size =40	eco-size =50
F1	Mean(SD)	**4000.40(1127.43)**	4205.60(1215.61)	5578.80(1272.45)	6465.60(1409.82)	8450.00(2175.62)
	SR	100	100	100	100	100
F2	Mean(SD)	**8128.40(1460.62)**	8800.80(2623.92)	11684.40(3152.20)	14145.60(3709.93)	16562.00(4835.31)
	SR	100	100	100	100	100
F3	Mean(SD)	5117.20(962.41)	**4359.20(1215.30)**	5809.20(1350.41)	7374.40(2096.18)	9010.00(2581.99)
	SR	100	100	100	100	100
F4	Mean(SD)	**9347.60(1544.11)**	10176.80(2807.27)	12778.80(3672.16)	16424.00(3550.17)	18706.00(4872.89)
	SR	100	100	100	100	100
F5	Mean(SD)	NaN(NaN)	69140.00(0.00)	**35370.00(19622.72)**	NaN(NaN)	62450.00(0.00)
	SR	0	4	16	0	4
F6	Mean(SD)	**1203.60(905.37)**	1210.40(333.32)	1633.20(458.10)	1960.00(261.28)	2226.00(490.99)
	SR	100	100	100	100	100
F7	Mean(SD)	NaN(NaN)	NaN(NaN)	NaN(NaN)	NaN(NaN)	NaN(NaN)
	SR	0	0	0	0	0
F8	Mean(SD)	NaN(NaN)	36886.67(17673.30)	39870.00(19055.20)	28648.00(13182.90)	**34480.77(17877.70)**
	SR	0	48	52	40	52
F9	Mean(SD)	48146.00(18517.62)	**10106.40(1699.87)**	10762.80(2876.26)	13992.00(3448.98)	16402.00(5234.91)
	SR	40	100	100	100	100
F10	Mean(SD)	**8416.40(1065.06)**	10900.00(3846.66)	42150.00(7976.16)	NaN(NaN)	55450.00(4808.33)
	SR	12	8	8	0	8
F11	Mean(SD)	22396.67(10914.47)	29960.00(14173.16)	31508.40(12096.75)	**22900.80(4723.91)**	27858.00(6895.53)
	SR	24	96	96	100	100
F12	Mean(SD)	68250.00(0.00)	47406.67(15981.06)	**26506.36(9976.17)**	30226.67(11162.15)	31413.64(13413.09)
	SR	4	48	88	84	88

Here "Mean (SD)" represents the statistical result of fitness evolution requires, "SR" represents the success rate.

From Table 8, it is observed that strategy with Eco-size 10 requires minimum function evaluation on four functions; strategy with Eco-size 20 requires minimum function evaluation on two functions; strategy with Eco-size 30 requires minimum function evaluation on one function; strategy with Eco-size 40 requires minimum function evaluation on two functions.

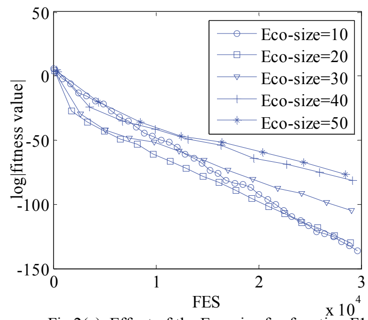

Fig.2(a). Effect of the Eco-size for function F1

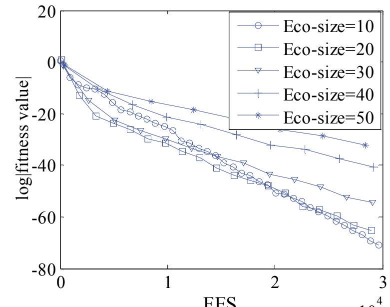

Fig.2(b). effect of Eco-size for function F2

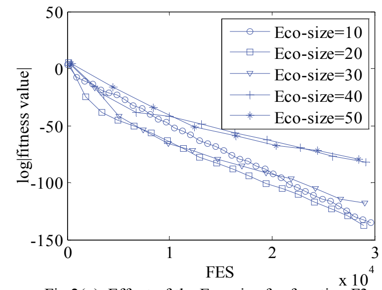

Fig.2(c). Effect of the Eco-size for function F3

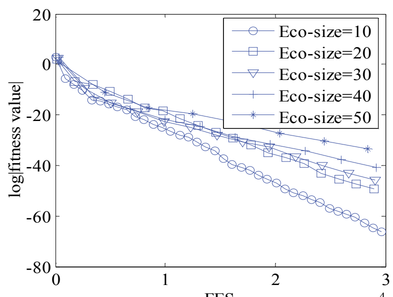

Fig.2(d). Effect of Eco-size for function F4

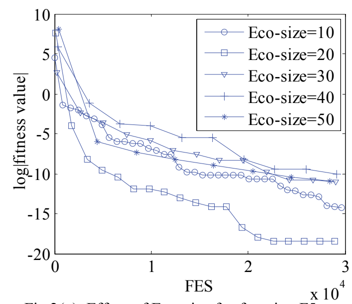

Fig.2(e). Effect of Eco-size for function F5

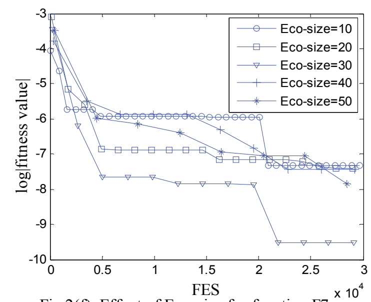

Fig.2(f). Effect of Eco-size for function F7

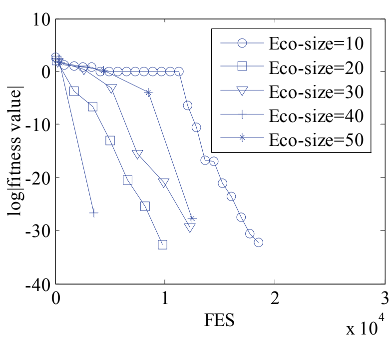

Fig.2(g). Effect of Eco-size for function F9

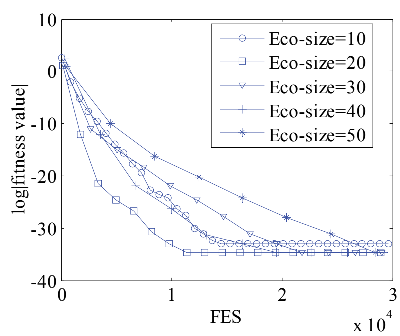

Fig.2(h). Effect of Eco-size for function F10

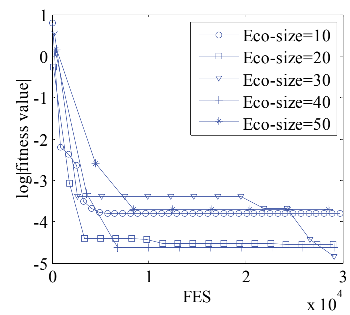

Fig.2(i). Effect of Eco-size for function F11

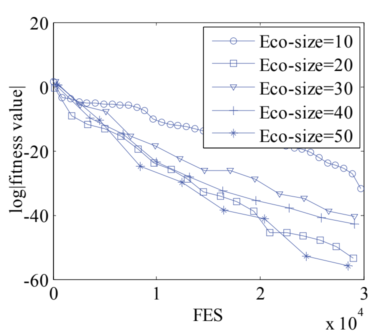

Fig.2(j). Effect of Eco-size for function F11

Fig. 2. Effect of Eco-size of ten benchmark function which are given in Table 1 with dimension 3 and 30000 function evaluation (FES)

From Table 9, it is observed that strategy with Eco-size 10 requires minimum function evaluation on five functions; strategy with Eco-size 20 requires minimum function evaluation on two functions; strategy with Eco-size 30 requires minimum function evaluation on two functions; strategy with Eco-size 40 requires minimum function evaluation on one function; strategy with Eco-size 50 requires minimum function evaluation on one function. Fig.2 (a)-Fig.2 (j) shows the convergence graph of 8 benchmark functions with dimension 3. From this figure it is seen that strategy with Eco-size 10 converges faster on one function (Fig.2. (d)); strategy with Ecosize 20 converges faster on five functions (Fig.2. (a), Fig.2. (b), Fig.2. (c), Fig.2. (e), Fig.2. (h)); strategy with Eco-size 30 converges faster on one function(Fig.2. (f)); strategy with Eco-size 40 converges faster on two functions (Fig.2. (g), Fig.2. (i)); strategy with Eco-size 50 converges faster on one function (Fig.2. (j)).

4.2 Comparison with other algorithms

Experiment-ll: In this experiment performance of the results of I-SOS algorithm compared to others PSO, DE and SOS algorithm. For this experiment the details of the parameter setting of all algorithms are given in the Table 10, the details of the function formulation are reported in Table 2, and the entire algorithms run 30 times.

Table 10

Parameter setting for Experiment-ll (MAX_FEs: maximum function evaluation)

DE	PSO	SOS	I-SOS
F = 0.5;CR = 0.9; MAX_FEs = 50000; Population size=50; Run=30 times	C1=C2=2; wmax=0.9; wmin=0.4; Population size=50; w(i)=wmax-(wmax-wmin)*it/max_it; MAX_FEs = 50000; Run=30 times	Eco-size=50; MAX_FEs = 50000; Run=30 times	Eco-size=50; MAX_FEs = 50000; Run=30 times

The performances results of this experiment are reported in Table11. The bold of the Table 11 are represented the best performance of the algorithm. From this table it is observed that I-SOS gives best value on eight problems, identical result on ten problems and the inferior on other problems. Based on the above discussion it is conclude that the proposed I-SOS better than other algorithms.

Table 11

Performance Comparison of I-SOS of 25 benchmark function (given in Table 2)

Function		**DE**	**PSO**	**SOS**	**I-SOS**
F1	Best	3.214e-012	3.2607e-009	2.9455e-041	7.75e-066
	Mean	2.1731e-011	7.05e-008	7.9881e-040	**1.76e-056**
	SD	3.4304e-011	1.0422e-007	9.5062e-040	6.12e-056
F2	Best	83.0589	18.9042	0	0.00e+000
	Mean	162.3172	31.0095	**0**	1.07e+001
	SD	23.2552	10.3863	0	1.60e+001
F3	Best	-1.0316	-1.0316	-1.0316	-1.0316
	Mean	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**
	SD	6.7752e-016	6.3208e-016	6.3208e-016	3.55e-009
F4	Best	0	0	0	0.00e+000
	Mean	0.033333	**0**	**0**	**0.00e+000**
	SD	0.18257	0	0	0.00e+000
F5	Best	23.7345	23.6004	22.2768	3.28e-004
	Mean	25.7495	24.5653	22.9007	**9.61e-001**
	SD	0.73654	0.99983	0.26148	4.53e+000
F6	Best	4.0813e-007	1.2854e-005	4.4409e-015	4.44e-015
	Mean	1.3849e-006	5.8766e-005	**4.4409e-015**	4.68e-015
	SD	1.1691e-006	2.8047e-005	0	9.01e-016
F7	Best	4.1092e-012	1.1155e-008	0	0.00e+000
	Mean	0.0027898	0.01124	**0**	**0.00e+000**
	SD	0.006045	0.013354	0	0.00e+000

Table 11
Performance Comparison of I-SOS of 25 benchmark function (given in Table 2) (Continued)

Function		DE	PSO	SOS	I-SOS
F8	Best	0.19987	0.19987	0.099873	0.099873
	Mean	0.25884	0.27995	**0.099873**	**0.099873**
	SD	0.049034	0.040531	2.9947e-008	4.98e-002
F9	Best	0.0068033	0.009478	0.00069807	9.52e-005
	Mean	0.014463	0.016506	0.0017759	**3.68e-004**
	SD	0.0051564	0.0048712	0.0006258	2.63e-004
F10	Best	1.6298e-005	2.1402e-007	5.1197e-022	6.12e-035
	Mean	0.0045973	2.2402e-006	3.6525e-021	**5.50e-022**
	SD	0.005478	1.9682e-006	2.8203e-021	3.01e-021
F11	Best	0.39789	0.39789	0.39789	0.39789
	Mean	**0.39789**	**0.39789**	**0.39789**	**0.39789**
	SD	0	0	0	0.00e+000
F12	Best	-1	-1	-1	-1.00e+000
	Mean	**-1**	**-1**	**-1**	**-1.00e+000**
	SD	0	0	0	0.00e+000
F13	Best	3	3	3	3.00e+000
	Mean	**3**	**3**	**3**	**3.00e+000**
	SD	2.531e-015	5.3444e-016	1.708e-015	1.18e-015
F14	Best	-186.7309	-186.7309	-186.7309	-186.7309
	Mean	**-186.7309**	**-186.7309**	**-186.7305**	**-186.7309**
	SD	3.4891e-009	3.4204e-014	0.00084567	1.59e-002
F15	Best	-3.8628	-3.8628	-3.8628	-3.8628
	Mean	**-3.8628**	**-3.8628**	**-3.8628**	**-3.8628**
	SD	3.1618e-015	3.1618e-015	3.1618e-015	3.16e-015
F16	Best	1.1067e-013	1.5872e-009	2.8218e-008	9.46e-006
	Mean	0.054385	0.71453	0.22979	**6.89e-004**
	SD	0.14808	0.74783	0.11915	8.09e-004
F17	Best	-9.6602	-9.5427	-9.5296	-9.63e+000
	Mean	-8.9527	-8.9887	-9.1651	**-9.24e+000**
	SD	0.48428	0.45012	0.23738	2.41e-001
F18	Best	0	0	3.3862e-015	0.00e+000
	Mean	**0**	0.0012955	1.329e-006	7.13e-003
	SD	0	0.0033592	5.4257e-006	4.37e-003
F19	Best	4.5021	0.89406	5.5408e-011	4.57e-033
	Mean	27.7742	2.3884	2.3875e-009	**1.76e-026**
	SD	26.9685	1.0705	4.5843e-009	5.15e-026
F20	Best	0.00030749	0.00030749	0.00030749	3.07e-004
	Mean	0.00085002	0.00050397	0.00031011	**3.07e-004**
	SD	0.0002561	0.0004502	7.7699e-006	7.31e-018
F21	Best	-10.1532	-10.1532	-10.1532	-10.1532
	Mean	**-10.1532**	-5.5552	**-10.1532**	**-10.1532**
	SD	7.2269e-015	3.2222	9.2123e-007	3.14e-006
F22	Best	1.6689e-012	6.6181e-010	8.5597e-013	3.54e-007
	Mean	0.0003645	0.30743	**3.7727e-010**	4.30e-003
	SD	0.0019964	0.40339	6.9244e-010	2.34e-002
F23	Best	5.9035e-013	5.3616e-011	2.8314e-012	2.59e-005
	Mean	**2.3836e-011**	0.00036625	0.0021975	6.85e-003
	SD	4.6132e-011	0.002006	0.00447	9.34e-003
F24	Best	1.7075e-193	0	1.0881e-159	9.87e-163
	Mean	**5.6919e-188**	5.9594e-054	4.2619e-149	3.45e-153
	SD	0	1.4478e-053	1.7671e-148	1.08e-152
F25	Best	**1.1052e-193**	4.8283e-064	2.5408e-158	1.17e-163
	Mean	3.7392e-186	2.4588e-053	1.2298e-149	5.59e-152
	SD	0	1.257e-052	5.1479e-149	3.06e-151

Experiment-lll: For this experiment a set of benchmark function which are given in Table 3 is taken and the experimental result are compared to CoDE (Wang et al. 2011) and EPSDE (Mallipeddi et al. 2010). The algorithms were run 25 times with Eco-size 50 and 200,000 function evolution.

Table 12
Comparison of the Statistical result of I-SOS with other DE variants for 11 benchmark function which are given in Table 3 (eco-size=50, Fitness evolution (FE)=200000, D=50, Run=25)

Function		CoDE	EPSDE	I-SOS
F1	Best	2.05E-12	1.99E-75	2.57E-120
	Mean	1.01E-11	6.23E-72	**7.44E-109**
	Std.	7.09E-12	1.53E-71	2.19E-108
F2	Best	3.56E+01	3.64E+00	2.03E-03
	Mean	3.67E+01	9.60E+00	**1.47E-01**
	Std.	1.22E+00	3.03E+00	2.31E-01
F3	Best	3.94E-07	7.11E-15	3.55E-15
	Mean	6.31E-07	3.52E-02	**4.69E-15**
	Std.	2.29E-07	1.76E-01	1.69E-15
F4	Best	2.52E-12	0.00E+00	0.00E+00
	Mean	1.65E-11	8.88E-04	**0.00E+00**
	Std.	2.16E-11	2.45E-03	0.00E+00
F5	Best	5.33E-15	0.00E+00	0.00E+00
	Mean	3.75E-14	1.59E-01	**0.00E+00**
	Std.	2.66E-14	3.72E-01	0.00E+00
F6	Best	4.54E+01	2.16E-11	0.00E+00
	Mean	5.04E+01	1.19E+01	**9.77E+00**
	Std.	3.61E+00	1.30E+01	1.48E+01
F7	Best	2.10E+01	1.82E-11	1.74E+03
	Mean	7.45E+02	**4.74E+00**	4.65E+03
	Std.	6.03E+02	2.37E+01	1.51E+03
F8	Best	6.67E-01	6.67E-01	7.60E-04
	Mean	6.67E-01	6.67E-01	**3.46E-02**
	Std.	9.16E-10	2.80E-16	1.32E-01
F9	Best	4.08E-13	1.50E-32	2.80E-04
	Mean	**1.85E-12**	1.74E-01	6.43E-03
	Std.	1.20E-12	4.22E-01	6.49E-03
F10	Best	7.20E-13	7.93E-77	8.03E-119
	Mean	3.13E-12	2.51E-72	**1.29E-109**
	Std.	2.07E-12	9.54E-72	4.35E-109
F11	Best	1.44E-05	6.12E-11	1.86E-22
	Mean	5.91E-05	7.37E+01	**1.10E-19**
	Std.	4.90E-05	1.65E+02	2.57E-19

The performance results are reported in Table 12. Here the bold value represented the best performance of the algorithms. From this table it is observed that the proposed I-SOS algorithms give us good performed on nine functions out of eleven functions.

5. Real world application

The proposed I-SOS is applied to two real world problems. The problems are taken from (Li et al. 2011). The formulation of these problems is given below:

RWP.1.Gas transmission compressor design problem:

$$Min\ f(x) = 8.61 \times 10^5 \times x_1^{\frac{1}{2}} \times x_2 \times x_3^{\frac{-2}{3}} \times (x_2^2 - 1)^{-\frac{1}{2}} + 3.69 \times 10^4 \times x_3 + 7.72 \times 10^8 \times x_1^{-1} \times x_2^{0.219} - 765.43 \times 10^6 \times x_1^{-1}$$

s.t 10 ≤ x1 ≤ 55, 1.1 ≤ x2 ≤ 2, 10 ≤ x3 ≤ 40;

RWP.2. Optimal capacity of gas production facilities:

$$Min\ f(x) = 61.8 + 5.72 \times x_1 \times 0.2623 \times \left[(40 - x_1) \times \ln\left(\frac{x_2}{200}\right)\right]^{-0.85} + 0.087 \times (40 - x_1) \times \ln\left(\frac{x_2}{200}\right) + 700.23 \times x_2^{-0.75}$$

S.t $x2 \geq 17.5$, $x2 \geq 200$, $17.5 \leq x1 \leq 40$, $300 \leq x2 \leq 600$;

Table 13

RWP.1. Comparison performance of I-SOS with DE, GSA, DE-GSA and DFO to solve Gas Transmission Compressor Design

Item	DE	GSA	DE-GSA	Beightler and Phillips	I-SOS
X_1	52.3966	53.0547	53.5080	55	**53.4467**
X_2	1.1875	1.1919	1.1901	1.195	**1.1901**
X_3	24.6697	24.5070	24.7624	25.026	**24.7186**
f(X)	2.96443E+06	2.96449E+06	2.96437E+06	2.96455E+06	**2.96438e+006**
RWP.2. Comparison performance of I-SOS with DE, GSA, DE-GSA and DFO to solve Optimal Capacity of Gas production facilities.					
X_1	17.5	17.5	17.5	17.5	**17.5**
X_2	600	600	600	465	**599.9999**
f(X)	169.844	169.844	169.844	173.76	**169.8437**

The experimental results of this problem are given in Table 13. In Table 13, results except I-SOS are taken from (Li et al. 2011). The bold face represents the best result than other algorithms. From this table it is observed that the performance results show of I-SOS algorithm possesses superior with other algorithms.

6. Conclusion

In SOS method, proposed by Min-Yuan and Prayogo, a new phase called predation phase is suggested to enhance the performance of the algorithm. Also a random weighted reflection vector is suggested to enhance the search ability of the algorithm. The modified algorithm thus obtained, called improved SOS (I-SOS) Algorithm, is presented in this paper to solve global numerical optimization problems. For the validity of this algorithm, the performance of the proposed algorithm is testified on a set of benchmark functions and reported in Table 1, 2 and 3. The effect of the common control parameters i.e. Eco-size and number of function evaluation are investigated by varying these two parameters. Also the performance results are compared with other basic algorithms and state of the art DE variants. The performance of proposed I-SOS is also applied on two real world problems. From the above discussion of the performance results it is conclude that I-SOS algorithm is superior to the other algorithms. Future research may be carried out by employing I-SOS in solving complex real problems even for constrained and multi-objective optimization problems.

Acknowledgement

The authors would like to thank the anonymous referees for constructive comments on earlier version of this paper.

References

Abdullahi, M., & Ngadi, M. A. (2016). Symbiotic Organism Search optimization based task scheduling in cloud computing environment. *Future Generation Computer Systems*, 56, 640-650.

Abido, M.A. (2009) Multiobjective particle swarm optimization for environmental/economic dispatch problem. *Electric Power Systems Research*, *79*, 1105–1113

Aickelin, U., & Dowsland, K. A. (2014). An indirect genetic algorithm for a nurse scheduling problem. *Computers & Operations Research*, *31*(5), 761-778.

Aulady, M. (2013). A hybrid symbiotic organisms search-quantum neural network for predicting high performance concrete compressive strength. Master's Thesis, http://pc01.lib.ntust.edu.tw/ETD-db/ETD-search/view_etd?URN=etd-0705114-091239.

Barati, M.A., Mohammadi, M., & Naderi, B. (2016). Multi-period fuzzy mean-semi variance portfolio selection problem with transaction cost and minimum transaction lots using genetic algorithm. *International Journal of Industrial Engineering Computations*, *7*(2), 217–22

Baykasoğlu, A., Hamzadayi, A., & Köse, S. Y. (2014). Testing the performance of teaching–learning based optimization (TLBO) algorithm on combinatorial problems: Flow shop and job shop scheduling cases.*Information Sciences*, *276*, 204-218.

Bhunia, A., Pal, P., & Chattopadhyay, S. (2015). A hybrid of genetic algorithm and Fletcher-Reeves for bound constrained optimization problems.*Decision Science Letters*, *4*(2), 125-136.

Blum, C. (2005). Ant colony optimization: Introduction and recent trends. *Physics of Life Reviews*, *2* (4), 353-373.

Bolaños, R., Echeverry, M., & Escobar, J. (2015). A multiobjective non-dominated sorting genetic algorithm (NSGA-II) for the Multiple Traveling Salesman Problem. *Decision Science Letters*, *4*(4), 559-568.

Canelas, A., Neves, R., & Horta, N. (2013). A SAX-GA approach to evolve investment strategies on financial markets based on pattern discovery techniques. *Expert Systems with Applications*, *40*(5), 1579-1590.

Chander, A., Chatterjee, A., & Siarry, P. (2011). A new social and momentum component adaptive PSO algorithm for image segmentation.*Expert Systems with Applications*, *38*(5), 4998-5004.

Cheng, M. Y., & Prayogo, D. (2014). Symbiotic Organisms Search: A new metaheuristic optimization algorithm. *Computers & Structures*, *139*, 98-112.

Cheng, M.-Y., Prayogo, D., & Tran, D.-H. (2015). Optimizing multiple-resources leveling in multiple projects using discrete symbiotic organisms search. *Journal of Computing in Civil Engineering*, DOI: 10.1061/(ASCE)CP.1943-5487.0000512

Dorigo, M., Maniezzo, V., & Colorni, A. (1991).Positive feedback as a search strategy. *Technical Report 91-016, Dipartimento di Elettronica, Politecnico di Milano, IT*.

Eki, R., Vincent, F. Y., Budi, S., & Redi, A. P. (2015). Symbiotic Organism Search (SOS) for Solving the Capacitated Vehicle Routing Problem. *World Academy of Science, Engineering and Technology, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, *9*(5), 850-854.

Eshraghi, A. (2016). A new approach for solving resource constrained project scheduling problems using differential evolution algorithm. *International Journal of Industrial Engineering Computations*, *7*(2), 205-216.

Eskandar, H., Sadollah, A., Bahreininejad, A., & Hamdi, M. (2012). Water cycle algorithm–A novel metaheuristic optimization method for solving constrained engineering optimization problems. *Computers & Structures*,*110*, 151-166.

Geem, Z. W., Kim, J. H., & Loganathan, G. V. (2001). A new heuristic optimization algorithm: harmony search. *Simulation*, *76*(2), 60-68.

Gen, M., Tsujimura, Y., & Kubota, E. (1994, October). Solving job-shop scheduling problems by genetic algorithm. In *Systems, Man, and Cybernetics, 1994. Humans, Information and Technology., 1994 IEEE International Conference on* (Vol. 2, pp. 1577-1582). IEEE.

Ghasemi, M., Ghavidel, S., Ghanbarian, M. M., Massrur, H. R., & Gharibzadeh, M. (2014). Application of imperialist competitive algorithm with its modified techniques for multi-objective optimal power flow problem: a comparative study. *Information Sciences*, *281*, 225-247.

Hecker, F. T., Hussein, W. B., Paquet-Durand, O., Hussein, M. A., & Becker, T. (2013). A case study on using evolutionary algorithms to optimize bakery production planning. *Expert Systems with Applications*, *40*(17), 6837-6847.

Hecker, F. T., Stanke, M., Becker, T., & Hitzmann, B. (2014). Application of a modified GA, ACO and a random search procedure to solve the production scheduling of a case study bakery. *Expert Systems with Applications*,*41*(13), 5882-5891.

Holland, J. H. (1975). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. U Michigan Press.

Hosseini, M., Sadeghzade, M., & Nourmandi-Pour, R. (2014). An efficient approach based on differential evolution algorithm for data clustering.*Decision Science Letters*, *3*(3), 319-324.

Kavousi-Fard, A., Rostami, M. A., & Niknam, T. (2015). Reliability-oriented reconfiguration of vehicle-to-grid networks. *Industrial Informatics, IEEE Transactions on*, *11*(3), 682-691.

Kennedy, J., Eberhart, R., (1995). Particle swam optimization, *in: Proceeding of the IEEE International Conference on Neural Network, Piscataway*, *4*, 1942–1948.

Li, X., Yin, M., & Ma, Z. (2011). Hybrid differential evolution and gravitation search algorithm for unconstrained optimization. *Int. J. Phys. Sci*, *6*(25), 5961-5981.

Li, X.L., & He, X.D. (2014) A hybrid particle swarm optimization method for structure learning of probabilistic relational models. *Information Sciences, 28*3, 258–266

Mallipeddi, R., Suganthan, P. N., Pan, Q. K., & Tasgetiren, M. F. (2011). Differential evolution algorithm with ensemble of parameters and mutation strategies. *Applied Soft Computing*, *11*(2), 1679-1696.

Mir, M.S.S., & Rezaeian, J., (2016), A robust hybrid approach based on particle swarm optimization and genetic algorithm to minimize the total machine load on unrelated parallel machines, *Applied Soft Computing*, *41*, 488-504

Mohammadi, S., Rahmani, M., & Azadi, M. (2015). Optimization of continuous ranked probability score using PSO. *Decision Science Letters*,*4*(3), 373-378.

Nama, S., Saha, A., & Ghosh, S. (2016). A new ensemble algorithm of differential evolution and backtracking search optimization algorithm with adaptive control parameter for function optimization. *International Journal of Industrial Engineering Computations*, *7*(2), 323-338.

Nama, S., Saha, A. K., & Ghosh, S. (2015). Parameters Optimization of Geotechnical Problem Using Different Optimization Algorithm. *Geotechnical and Geological Engineering*, *33*(5), 1235-1253.

Orouji, M. (2016). Theory of constraints: A state-of-art review. *Accounting*,*2*(1), 45-52.

Prasad, D., & Mukherjee, V. (2015). A novel symbiotic organisms search algorithm for optimal power flow of power system with FACTS devices. *Engineering Science and Technology, an International Journal*, *19*(1), 79-89.

Rao, R & Patel, V. (2014). A multi-objective improved teaching-learning based optimization algorithm for unconstrained and constrained optimization problems. *International Journal of Industrial Engineering Computations*, *5*(1), 1-22.

Rao, R. (2016). Review of applications of TLBO algorithm and a tutorial for beginners to solve the unconstrained and constrained optimization problems.*Decision Science Letters*, *5*(1), 1-30.

Rao, R. V., Savsani, V. J., & Vakharia, D. P. (2011). Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. *Computer-Aided Design*, *43*(3), 303-315.

Rout, N. K., Das, D. P., & Panda, G. (2016) Particle swarm optimization based nonlinear active noise control under saturation nonlinearity. *Applied Soft Computing*, *41*, 275-289

Satapathy, S., & Naik, A. (2013). Improved teaching learning based optimization for global function optimization. *Decision Science Letters*, *2*(1), 23-34.

Shi, Y., & Eberhart, R. (1998). A modified particle swarm optimizer. *In: Evolutionary computation proceedings. IEEE World Congress on Computational Intelligence.* Pp. 69 – 73, doi:10.1109/ICEC.1998.699146.

Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, *11*(4), 341-359.

Suganthan, P. N., Hansen, N., Liang, J. J., Deb, K., Chen, Y. P., Auger, A., & Tiwari, S. (2005). Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. *KanGAL report*,*2005005*, 2005.

Verma, S., Saha, S., & Mukherjee, V. (2015). A novel symbiotic organisms search algorithm for congestion management in deregulated environment. *Journal of Experimental & Theoretical Artificial Intelligence*, DOI: 10.1080/0952813X.2015.1116141

Wang, Y., Cai, Z., & Zhang, Q. (2011). Differential evolution with composite trial vector generation strategies and control parameters. *IEEE Transactions on Evolutionary Computation, 15*(1), 55–66.

Wang, Y., Zhou, J., Zhou, C., Wang, Y., Qin, H., & Lu, Y. (2012). An improved self-adaptive PSO technique for short-term hydrothermal scheduling. *Expert Systems with Applications*, *39*(3), 2288-2295.