

# Collaborative Strategy for Grey Wolf Optimization Algorithm

Esra F. Alzaghoul<sup>1</sup> & Sandi N. Fakhouri<sup>1</sup>

<sup>1</sup> King Abdullah II School of Information Technology, The University of Jordan, Amman, Jordan

Correspondence: Esra Alzaghoul, King Abdullah II School of Information Technology, The University of Jordan, Amman, Jordan. E-mail: e.zaghoul@ju.edu.jo

Received: September 2, 2017

Accepted: March 14, 2018

Online Published: June 22, 2018

doi:10.5539/mas.v12n7p73

URL: <https://doi.org/10.5539/mas.v12n7p73>

## Abstract

Grey wolf Optimizer (GWO) is one of the well known meta-heuristic algorithm for determining the minimum value among a set of values. In this paper, we proposed a novel optimization algorithm called collaborative strategy for grey wolf optimizer (CSGWO). This algorithm enhances the behaviour of GWO that enhances the search feature to search for more points in the search space, whereas more groups will search for the global minimal points. The algorithm has been tested on 23 well-known benchmark functions and the results are verified by comparing them with state of the art algorithms: Polar particle swarm optimizer, sine cosine Algorithm (SCA), multi-verse optimizer (MVO), supernova optimizer as well as particle swarm optimizer (PSO). The results show that the proposed algorithm enhanced GWO behaviour for reaching the best solution and showed competitive results that outperformed the compared meta-heuristics over the tested benchmarked functions.

**Keywords:** Grey wolf optimizer (GWO), Particle Swarm optimization, meta-heuristic, optimization, Real-world optimization problem

## 1. Introduction

A common mathematical problem in all engineering disciplines is optimization which is finding the best solutions. In nature, optimization algorithms can be deterministic or stochastic. Existing methods for solving optimization problems require huge computational costs, to solve the problem such as stochastic optimization. A better method to solve this optimization problem is using meta-heuristics based on iterative improvement, or a population of solutions (both in evolutionary and swarm algorithms). Meta-heuristic algorithms are used in solving difficult optimization problems in many fields including engineering design, economics, Internet routing and cloud computing: problems such as data migration and resource scheduling. As resources are limited and cost money, an optimal feasible solution is a crucial requirement. Real-world optimization problems vary from nonlinear to multimodal, under different complex constraints. However, finding an optimal solution or even sub-optimal solutions is challenging. Many meta-heuristic algorithms have been proposed. Heuristic approach used in solving complex optimization tasks, especially where traditional methods fail. Heuristics, imitate the strategy of nature, they use a lot of random solutions, which classify them as a special class of randomized algorithms. Meta-heuristic algorithms include: a) choosing the correct representation of the problem; b) assessing the quality of the solution using the fitness function; c) defining operators in such a way as to obtain a new set solutions.

Swarm-based meta-heuristic algorithms have been considered as powerful optimization techniques that are inspired by animals' social behavior. Particle swarm optimization (PSO) has been known as one of the most popular optimizers and is widely used in optimization of different applications. PSO was first proposed by J. Kennedy and R. C. Eberhart et al in 1995 developed an evolutionary computing technology, derived from a simplified social model of the simulation. Where "swarm" comes from the particle swarm in line with M. M. Millonas's five basic principles of group intelligence in the development of models applied to artificial life. "Particle" is a compromise because it is necessary to describe the members of the group as no quality, no volume, and to describe its speed and acceleration status. Particle Swarm Optimization, (PSO) method of numerical optimization, for the use of which you do not need to know the exact gradient of the optimized function. This method belongs to the group algorithms of swarm intelligence, which describe collective behavior of a decentralized self-organizing system.

Grey Wolf Optimizer (GWO) inspired by grey wolves (Mirjalili et al., 2014). The GWO algorithm mimics the leadership hierarchy and hunting mechanism of grey wolves in nature. Four types of grey wolves such as alpha, beta, delta, and omega are employed for simulating the leadership hierarchy. In addition, the three main steps of

hunting, searching for prey, encircling prey, and attacking prey, are implemented.

Since the particles in the GWO move toward their best position and the best position of the neighborhood or group history, the rapid convergence effect of how the wolf population is formed. The local extreme, premature convergence. At the same time, GWO performance also depends on the algorithm parameters. In order to overcome the above shortcomings, researchers have put forward various enhancement measures. Similar to any meta-heuristic the GWO enhancements fields into four categories: initialization, neighborhood topology, parameter selection and hybrid strategy, figure 1 shows the categories of the two enhancements

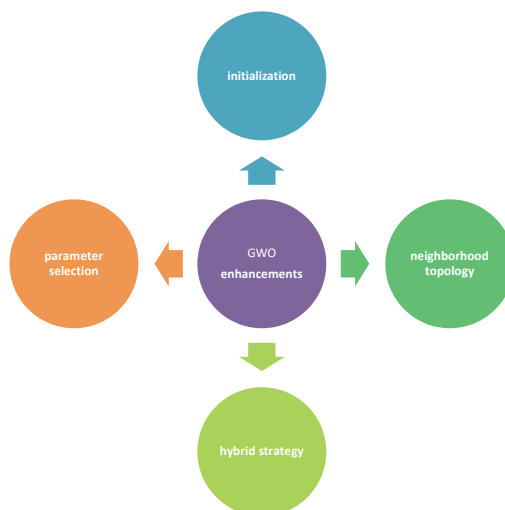


Figure 1. GWO enhancement research fields

In this paper, we propose an enhanced meta-heuristic algorithm that modifies the behavior of GWO algorithm by enhancing the neighborhood topology and movement strategy to find the optimal solution in the search space; by applying a collaborative strategy for Grey wolf Optimization Algorithm (CSGWO). Applying the collaborative strategy in optimization has been widely used to solve many real-life problems and applications and this inspired us to apply the collaborative strategy behavior during the movement of GWO to enhance its behavior in finding the local and global minimum.

The rest of this paper is organized as follows: section two presents the related work, while section three presents the CSGWO strategy, including the inspiration, pseudocode and the algorithm explanation. Section four presents the experimental results of applying the algorithm on CEC2005 benchmark functions and compares it with state of the art algorithms. Finally section five presents, findings and conclusion.

## 2. Related Work

According to Handl & Meyer (2007), algorithms can be categorized into two groups: the first group of algorithms is inspired by nature and animals' behavior such as ants' methods in building colonies and where researchers followed the same strategy. The second group of algorithms uses techniques such as clustering-methods and data mining. Researchers were inspired by observing wasp nests (Campos et al, 2000) and bee colonies in coming up with a gravitational search algorithm which was named the Bee Colony algorithm by Pham et al., (2005). This algorithm depends on two kinds of searching for explorative purposes: local and global, and is done through agents with two different types, as can be seen in the bee colony. Bee colonies also inspired the artificial bee colony algorithm by Karaboga in 2005, which is somehow different from the previous one; it consists of employed, onlooker and scout bees: three different kinds of agents which are trying to find food in the surrounding area using varying methods.

The PSO idea was first introduced by Kennedy & Eberhart (1995) then improved by them in the same year (Eberhart & Kennedy, 1995). Selvan, S. Easter et al. (2003), proposed a modified PSO algorithm that was implemented in Proportional Integral Derivative (PID) controller tuning by incorporating some special features in the conventional PSO technique; its usefulness was tested in a common control application involving PID controller tuning. The results obtained for different plant transfer functions confirmed the theoretical predictions. Yu, S. et al. (2012), proposed a new hybrid PSO (HPSO) to solve the problem that PSO often easily falls into local

optima; the results have shown that HPSO has a faster convergence rate on those simple unimodal functions and superior global search ability on those multimodal functions compared to other PSO. Binh, (2013) proposed a new hybrid particle swarm optimization algorithm for solving Multi-Area Economic Dispatch Problem (MAEDP), the results show that their proposed approaches are stable and quite effective with MAEDP. Khanesar et al. (2007), proposed a new interpretation for the velocity of binary PSO related to the change rate particles' bits. In addition, they were able to solve the problem of choosing a  $w$  value (the inertia term) for older versions of binary PSO. Gimmmler et al. (2006) proposed a Hybrid Particle Swarm Optimization algorithm (HPSO) that combines a Particle Swarm Optimization (PSO) algorithm with the Nelder-Mead-Simplex-method (NMS) and with Powell's Direction-Set-Method (PDS). Their experimental results indicate that PSO performance can be improved by using iterative improvement algorithms.

Mirjalili et al., (2014) proposed a new meta-heuristic called Grey Wolf Optimizer (GWO) inspired by grey wolves (*Canis lupus*). The GWO algorithm mimics the leadership hierarchy and hunting mechanism of grey wolves in nature. Mirjalili, 2016 proposed a novel population-based optimization algorithm called Sine Cosine Algorithm (SCA) for solving optimization problems. The SCA created multiple initial random candidate solutions and required them to fluctuate outwards or towards the best solution using a mathematical model based on. Mirjalili et al. 2016 proposed a novel nature-inspired algorithm called Multi-Verse Optimizer (MVO), based on three concepts in cosmology: white hole, black hole, and wormhole. The mathematical models of these three concepts are developed to perform exploration, exploitation, and local search, respectively.

Hudaib and Fakhouri, (2018) proposed a novel optimization algorithm called supernova optimizer (SO) inspired by the supernova phenomena in nature. SO mimics supernova natural phenomena aiming to improve the three main features of optimization; exploration, exploitation, and local minima avoidance.

Al-Sayyed et al., 2017 proposed novel optimization algorithm called POLARPSO that enhances the behavior of PSO and avoids the local minima problem by using a polar function to search for more points in the search space

### 3. Collaborative Strategy Grey Wolf Optimizer (CSGWO)

In this section, we first describe the standard GWO algorithm and its pseudocode then we describe the proposed collaborative strategy grey wolf optimizer (CSGWO). The original Grey Wolf Optimizer was first described by Mirjalili (2014). Grey wolves are considered as apex predators, meaning that they are at the top of the food chain. Grey wolves mostly prefer to live in a pack. The group size is 5–12 on average. Of particular interest is that they have a very strict social dominant hierarchy. The leaders are a male and female, called alphas. The alpha is mostly responsible for making decisions about hunting, sleeping place, time to wake, and so on. The alpha's decisions are dictated to the pack. However, some kind of democratic behaviour has also been observed, in which an alpha follows the other wolves in the pack. In gatherings, the entire pack acknowledges the alpha by holding their tails down. The alpha wolf is also called the dominant wolf since his/her orders should be followed by the pack. The pseudocode of GWO is shown in figure 2.

```

Initialize the grey wolf population  $X_i$  ( $i = 1, 2, \dots, n$ )
Initialize  $a$ ,  $A$ , and  $C$ 
Calculate the fitness of each search agent
 $X_\alpha$  = the best search agent
 $X_\beta$  = the second best search agent
 $X_\delta$  = the third best search agent
while ( $t < \text{Max number of iterations}$ )
  For each search agent Update the position of the current search agent
End For ;
Update  $a$ ,  $A$ , and  $C$ 
Calculate the fitness of all search agents
Update  $X_\alpha$ ,  $X_\beta$ , and  $X_\delta$ 
 $t = t + 1$ 
end while
return  $X_\alpha$ 

```

Figure 2. GWO pseudocode

CSGWO introduces a new parameters to improve its convergence performance ( $X_\delta$ ,  $X_\epsilon$ , and  $X_\zeta$ ). The strategy changes the structure as well as the implemental and computational advantages of the basic GWO. The algorithm

consists of two groups of grey wolves each has its own best position and at each iteration every group search independently with the other group then both groups inform the other group about the best point and best position found the pseudocode of the algorithms is shown in figure 3. The algorithm consists of a two phases; the first is the initialization phase and the second is the iteration phase.

In the initialization phase, two groups are randomly generated so as to uniformly cover the search space. Each agent in the group maintaining its own fitness value and best position. In the second phase the iteration phase start the two groups exchange information and record two leaders best position value who lead the entire wolf groups algorithmic steps and Pseudo code The Collaborative strategy GWO algorithm is described briefly with the following steps: Step 1: Initialize the GWO parameters such as search agents (Gs), design variable size (Gd), vectors a, A, C and maximum number of iteration (itermax).

$$A = 2a \cdot \text{rand}_1 - a \quad \text{eq.1}$$

$$C = 2 \cdot \text{rand}_2 \quad \text{eq.2}$$

The values of “a” are linearly decreased from 2 to 0 over the course of iterations.

Step 2: Generate two wolves groups randomly based on size. Mathematically, these wolves can be expressed as:

Group 1					Group 2				
G11	G12	G13	.....	G1n	G11	G12	G13	.....	G1n
G21	G22	G23	.....	G2n	G21	G22	G23	.....	G2n
G31	G32	G13	.....	G3n	G31	G32	G13	.....	G3n
Gn1	Gn2	Gn3	.....	Gnn	Gn1	Gn2	Gn3	.....	Gnn

Step 3: Estimate the fitness value of each hunt agent using Equations 3, 4

$$D = C \cdot G_p(t) - G(t) \quad \text{eq.3}$$

$$G(t+1) = G_p(t) - A \cdot D \quad \text{eq.4}$$

Step 4: Identify the best hunt agent (G), the second best hunt agent (G) and the third best hunt agent (G) using Equations 5 to 16

$$D_a = C_1 \cdot G_a - G \quad \text{eq.5}$$

$$D_a = C_1 \cdot G_a - G \quad \text{eq.6}$$

$$D_a = C_1 \cdot G_a - G \quad \text{eq.7}$$

$$G_1 = G_a - A_1 \cdot (D_a) \quad \text{eq.8}$$

$$G_1 = G_a - A_1 \cdot (D_a) \quad \text{eq.9}$$

$$G_1 = G_a - A_1 \cdot (D_a) \quad \text{eq.10}$$

$$D\delta = C_1 \cdot G\delta - G \quad \text{eq.11}$$

$$\vec{D\varepsilon} = \vec{C_1} \cdot \vec{G\varepsilon} - \vec{G} \dots\dots\dots \text{eq.12}$$

$$\vec{D\zeta} = \vec{C_1} \cdot \vec{G\zeta} - \vec{G} \dots\dots\dots \text{eq.13}$$

$$\vec{G_1} = \vec{G\delta} - \vec{A_1} \cdot (\vec{D\delta}) \dots\dots\dots \text{eq.14}$$

$$\vec{G_1} = \vec{G\varepsilon} - \vec{A_1} \cdot (\vec{D\varepsilon}) \dots\dots\dots \text{eq.15}$$

$$\vec{G_1} = \vec{G\zeta} - \vec{A_1} \cdot (\vec{D\zeta}) \dots\dots\dots \text{eq.16}$$

Step 5: Renew the location of the current hunt agent using Equation 17 to 19

$$\vec{G_x(t+1)} = \frac{\vec{G_1} + \vec{G_2} + \vec{G_3}}{3} \dots\dots\dots \text{eq.17}$$

$$\vec{G_y(t+1)} = \frac{\vec{G_1} + \vec{G_2} + \vec{G_3}}{3} \dots\dots\dots \text{eq.18}$$

$$\vec{G(t-1)} = \vec{G_x(t-1)} \cdot \text{rand}() + \vec{G_y(t-1)} \cdot \text{rand}() \dots\dots\dots \text{eq.19}$$

The Collaborative strategy behaviour improves the GWO search ability at each iteration by employing multi-group search ability. This improvement include more points and polar direction to the particle movement which increased the possibility to find the global minima and avoid the local minima problem. CSGWO increases the ability of exploitation and exploration by exploring more points in the search space. Collaborative strategy includes multiple groups that search in the target search space in different dimensions, the search groups collaborate, and each group optimize the components of its solution vector. The pseudo code for CSGWO is shown in Figure 3.

```

Initialize the grey wolf population Xi (i = 1, 2,..., n)
  Initialize a, A, and C
Calculate the fitness of each search agent
  Xα=the best search agent
  Xβ=the second best search agent
  Xδ=the third best search agent
  while (t < Max number of iterations)
    for each search agent Update the position of the current search agent by equation (3.*)
  for each search agent Update the position of the current search agent by equation (3.*)
  for each search agent Update the position of the current search agent by equation (3.*)
end for
Update a, A, and C
Calculate the fitness of all search agents
Update Xα, Xβ, and Xδ

```

```

Update Xδ, Xε, and Xζ
t=t+1
end while
return Xα

```

Figure 3. Collaborative strategy Grey wolf optimizer (CSGWO) Pseudocode

#### 4. Experimental Results and Evaluation of CSGWO

This section presents an empirical evaluation of CSGWO. We evaluated the performance of CSGWO on the CEC2005 Special Session on Real-Parameter Optimization benchmark suite and compare CSGWO to state-of-the-art GWO, SCA, MVO and the well-known PSO optimization algorithm. The benchmark set consists of 20 test functions table 1. The mathematical formulations of the Functions from F1 to F5 which represent Unimodal Functions are shown in table 2. F6 to F12 represent Multimodal Functions shown in table 3. F13 to F14 are Expanded Functions shown in table 4. Finally, F15 to F20 shown in table 5 represent Hybrid Composition Functions combining multiple test problems into a complex landscape.

Table 1. Classification of Benchmark Functions

<b>Unimodal Functions (1-5)</b>	<b>Unimodal Functions</b>	F1: Shifted Sphere Function
		F2: Shifted Schwefel's Problem 1.2
		F3: Shifted Rotated High Conditioned Elliptic Function
		F4: Shifted Schwefel's Problem 1.2 with Noise in Fitness
		F5: Schwefel's Problem 2.6 with Global Optimum on Bounds
<b>Multimodal Functions (6-23)</b>	<b>Basic Functions</b>	F6: Shifted Rosenbrock's Function
		F7: Shifted Rotated Griewank's Function without Bounds
		F8: Shifted Rotated Ackley's Function with Global Optimum on Bounds
		F9: Shifted Rastrigin's Function
		F10: Shifted Rotated Rastrigin's Function
	<b>Expanded Functions</b>	F11: Shifted Rotated Weierstrass Function
		F12: Schwefel's Problem 2.13
		F13: Expanded Extended Griewank's plus Rosenbrock's Function (F8F2)
		F14: Shifted Rotated Expanded Scaffer's F6
		F15: Hybrid Composition Function
	<b>Hybrid Composition Functions</b>	F16: Rotated Hybrid Composition space Function
		F17: Rotated Hybrid Composition Function with Noise in Fitness
		F18: Rotated Hybrid Composition Function
		F19: Rotated Hybrid Composition Function with a Narrow Basin for the Global Optimum
		F20: Rotated Hybrid Composition Function with the Global Optimum on the Bounds

Table 2. Unimodal Functions mathematical formulation

Function	Formula
1.	$f1(x) = \sum_{i=1}^n x_i^2$
2.	$f2(x) = \sum_{i=1}^n  x_i  + \prod_{i=1}^n  x_i $
3.	$f3(x) = \sum_{i=1}^n \left( \sum_{j=1}^n x_j \right)^2$

---


$$4. \quad f4(x) = \max i\{|x_i|, 1 \leq i \leq n\}$$

$$5. \quad f5(x) = \sum_{i=1}^{n=1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$$


---

Table 3. Multimodal Basic Functions

---


$$6. \quad f6(x) = \sum_{i=1}^n (x_i + 0.5)^2$$


---

$$7. \quad f7(x) = \sum_{i=1}^n ix_i^4 + \text{random}(0,1)$$


---

$$8. \quad f8(x) = \sum_{i=1}^n -x_i \sin \sqrt{|x_i|} * \sum_{i=1}^n ix_i^4 * \text{random}(0,1) *$$


---

$$9. \quad f9(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$$


---

$$10. \quad f10(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{x=i}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e$$


---

$$11. \quad f11(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$


---

$$f12(x) = \frac{\pi}{n} \left\{ 10 \sin(\pi y_1) + \sum_{i=1}^{n=1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 \right\} + \sum_{i=1}^n u(x_i, 10, 100, 4)$$

$$12. \quad y_i = 1 + \frac{x_i + 1}{4}$$

$$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$$


---

Table 4. Multimodal expanded Functions

---


$$13. \quad f13(x) = 0.1 \left\{ \sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] \right\} + \sum_{i=1}^n u(x_i, 5, 100, 4)$$


---

$$14. \quad f14(x) = -\sum_{i=1}^n \sin(x_i) \cdot \left( \sin\left(\frac{ix_i^2}{\pi}\right) \right)^{2m}, m=10$$


---

Table 5. Hybrid Composition Functions

$f15(x) = \sum_{i=1}^{11} \left[ a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 x_4} \right]$
$f16(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$
$F17(X) = \left( x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6 \right)^2 + 10 \left( 1 - \frac{1}{8\pi} \right) \cos X_1 + 10$
$f18(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] x [30 + (2x_1 - 3x_2)^2x (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$
$f19(x) = - \sum_{i=1}^4 C_i \exp \left( - \sum_{j=1}^3 a_{ij} (x_j - p_{ij})^2 \right)$
$f20(x) = - \sum_{i=1}^4 C_i \exp \left( - \sum_{j=1}^6 a_{ij} (x_j - p_{ij})^2 \right)$

### Experiment Parameters

We performed our evaluation on the 23 CEC2005 benchmark functions. The number of iterations performed was 1000. The number of runs per problem was 30, and the average performance and standard deviation of these runs were calculated for evaluation purposes. Table 6 describes the Experimental parameters of the functions.

Table 6. Parameters for Experimented Functions

Function	Upper bound	Lower bound	Dimension	F Min
<b>F1</b>	-100	100	30	0
<b>F2</b>	-10	10	30	0
<b>F3</b>	-100	100	30	0
<b>F4</b>	-100	100	30	0
<b>F5</b>	-30	30	30	0
<b>F6</b>	-100	100	30	0
<b>F7</b>	-1.28	1.28	30	0
<b>F8</b>	-500	500	30	-418.9829*5
<b>F9</b>	-5.12	5.12	30	0
<b>F10</b>	-32	32	30	0
<b>F11</b>	-600	600	30	0
<b>F12</b>	-50	50	30	0
<b>F13</b>	-50	50	30	0
<b>F14</b>	-65.536	65.536	2	1
<b>F15</b>	-5	5	4	0.00030
<b>F16</b>	-5	5	2	1.0316
<b>F17</b>	[-5,0]	[10,15]	2	0.398
<b>F18</b>	-2	2	2	3
<b>F19</b>	0	1	3	-3.86
<b>F20</b>	0	1	6	-3.32
<b>F21</b>	0	10	4	-10.1532
<b>F22</b>	0	10	4	-10.4028
<b>F20</b>	0	10	4	-10.5363

We compared the results recorded of **CSGWO** with four functions; namely: PSO, GWO, SCA, and MVO, on the



benchmark problems set. The results of the MEAN, STD, MIN, and MAX values over 30 Experiments and the overall results on all 23 functions are shown in Tables 7 to 9.

Table 7. MEAN, STD, MIN, and MAX values over 30 Experiments for **CSGWO** and **GWO**

	CSGWO				GWO			
	mean	std	min	max	MEAN	STD	MIN	MAX
<b>F1</b>	0	0	0	0	7.81E-94	1.57E-93	1.64E-96	6.49E-93
<b>F2</b>	0	0	0	0	1.33E-53	2.83E-53	7.84E-55	1.6E-52
<b>F3</b>	0	0	0	0	1.5E-29	8.27E-29	1.89E-36	4.6E-28
<b>F4</b>	0	0	0	0	2.5E-24	1.96E-24	1.1E-25	7.26E-24
<b>F5</b>	28.86249	0.076894	28.71994	28.94166	25.89532	0.805491	24.28707	27.10698
<b>F6</b>	6.661193	0.316552	5.5816	7.000133	0.088737	0.15214	4.15E-06	0.501891
<b>F7</b>	3.92E-05	5E-05	3.79E-06	0.000256	0.000164	8.09E-05	3.52E-05	0.000366
<b>F8</b>	-2195.18	533.343	-3708.88	-1460.98	-6646.72	665.4916	-7899	-5230.34
<b>F9</b>	0	0	0	0	0	0	0	0
<b>F10</b>	4.32E-15	6.49E-16	8.88E-16	4.44E-15	9.03E-15	2.09E-15	7.99E-15	1.51E-14
<b>F11</b>	0	0	0	0	0.000981	0.003112	0	0.012634
<b>F12</b>	1.009224	0.156587	0.563144	1.456688	0.012597	0.013294	1.96E-07	0.06579
<b>F13</b>	2.935852	0.093118	2.551796	2.99184	0.112092	0.098292	3.45E-06	0.412478
<b>F14</b>	10.65079	3.429023	2.982105	12.67051	1.254017	0.676136	0.998004	2.982105
<b>F15</b>	0.000973	0.000692	0.000408	0.003918	0.001014	0.003598	0.000307	0.020363
<b>F16</b>	-1.01593	0.01386	-1.0316	-1	-1.03163	1.27E-09	-1.03163	-1.03163
<b>F17</b>	0.428369	0.04732	0.398473	0.587096	0.397887	2.2E-07	0.397887	0.397888
<b>F18</b>	4.789313	22.03944	3	12.03419	3	3.62E-07	3	3.000002
<b>F19</b>	-3.82304	0.040577	-3.85402	-3.6657	-3.86174	0.002643	-3.86278	-3.8549
<b>F20</b>	-2.1043	0.459438	-3.01145	-1.27737	-3.23261	0.077225	-3.32199	-3.02424

Table 8. MEAN, STD, MIN, MAX values over 30 experiments for **PSO**, **SCA**

	PSO				SCA			
	MEAN	STD	MIN	MAX	MEAN	STD	MIN	MAX
<b>F1</b>	8.41E-18	1.08E-17	1.79E-19	3.75E-17	4.0634E-06	8.65E-08	223.3377	1.881502
<b>F2</b>	1.42E-08	4.75E-08	8.37E-11	2.6E-07	8.83042E-06	2.67E-07	248.8871	1.913471
<b>F3</b>	1.572928	0.856245	0.426768	3.660063	3.95422E-11	1.42E-11	0.506488	0.273875
<b>F4</b>	0.153126	0.057302	0.063917	0.283493	3.49693E-05	1.46E-06	865.3114	8.817002
<b>F5</b>	52.88656	27.94947	13.31383	103.6437	4.0634E-06	8.65E-08	223.3377	1.881502
<b>F6</b>	1.25E-17	3.55E-17	3.01E-20	1.95E-16	8.83042E-06	2.67E-07	248.8871	1.913471
<b>F7</b>	0.023079	0.01037	0.00663	0.046117	3.95422E-11	1.42E-11	0.506488	0.273875
<b>F8</b>	-6983.88	645.3402	-8522.6	-5521.17	3.49693E-05	1.46E-06	865.3114	8.817002
<b>F9</b>	27.32976	6.088181	16.9143	40.79327	4.0634E-06	8.65E-08	223.3377	1.881502
<b>F10</b>	2.23E-09	2E-09	2.65E-10	6.87E-09	8.83042E-06	2.67E-07	248.8871	1.913471
<b>F11</b>	0.010427	0.009673	0	0.032045	3.95422E-11	1.42E-11	0.506488	0.273875
<b>F12</b>	9.41E-20	2.23E-19	9.16E-22	1.12E-18	3.49693E-05	1.46E-06	865.3114	8.817002
<b>F13</b>	0.000366	0.002006	2.84E-20	0.010987	4.0634E-06	8.65E-08	223.3377	1.881502
<b>F14</b>	1.031138	0.181484	0.998004	1.992031	8.83042E-06	2.67E-07	248.8871	1.913471
<b>F15</b>	0.000571	0.000309	0.000307	0.001081	3.95422E-11	1.42E-11	0.506488	0.273875
<b>F16</b>	-1.03163	6.78E-16	-1.03163	-1.03163	3.49693E-05	1.46E-06	865.3114	8.817002
<b>F17</b>	0.397887	0	0.397887	0.397887	4.0634E-06	8.65E-08	223.3377	1.881502
<b>F18</b>	3	2.18E-15	3	3	8.83042E-06	2.67E-07	248.8871	1.913471
<b>F19</b>	-3.86278	2.71E-15	-3.86278	-3.86278	3.95422E-11	1.42E-11	0.506488	0.273875
<b>F20</b>	-3.26651	0.060328	-3.322	-3.2031	3.49693E-05	1.46E-06	865.3114	8.817002

Table 9. MEAN, STD, MIN, MAX values over 30 experiments for MVO

	MVO			
	MEAN	STD	MIN	MAX
<b>F1</b>	0.06987	0.01721	0.033928	0.103761
<b>F2</b>	0.163783	0.031259	0.105083	0.237172
<b>F3</b>	3.734672	1.437699	1.190944	6.932981
<b>F4</b>	0.299265	0.122061	0.114662	0.61259
<b>F5</b>	110.0559	342.5613	24.32757	1909.894
<b>F6</b>	0.070938	0.020428	0.031562	0.116613
<b>F7</b>	0.005609	0.002286	0.002963	0.012472
<b>F8</b>	-8140.38	840.3777	-9523.94	-5916.09
<b>F9</b>	97.01212	23.66112	44.79188	130.3767
<b>F10</b>	0.289247	0.511412	0.055902	2.131228
<b>F11</b>	0.249762	0.065966	0.140284	0.393325
<b>F12</b>	0.673869	0.819071	0.000176	2.700481
<b>F13</b>	0.012404	0.009557	0.003528	0.053325
<b>F14</b>	0.998004	1.52E-12	0.998004	0.998004
<b>F15</b>	0.00114	0.003642	0.000308	0.020363
<b>F16</b>	-1.03163	1.55E-08	-1.03163	-1.03163
<b>F17</b>	0.397887	2.3E-08	0.397887	0.397887
<b>F18</b>	3	1.84E-07	3	3.000001
<b>F19</b>	-3.86278	3.39E-08	-3.86278	-3.86278
<b>F20</b>	-3.26252	0.060492	-3.322	-3.20295

To evaluate the effectiveness of any meta-heuristic algorithm in finding global minima, three major comparison features should be focused on: Exploitation Feature, Exploration Feature and Avoiding Local Minima. The chosen benchmark functions test set has functions that cover these three types, regarding the Exploitation Feature. It can be seen from the results in Table 3 that the CSGWO algorithm has shown very competitive results in functions F1, F2, F3, F4 and F7 with all other algorithms (GWO, SCA, PSO, and MFO). Since the unimodal functions are suitable for benchmarking exploitation, therefore, the results illustrated that CSGWO is better than the compared algorithms in terms of optimum exploitation.

Regarding the Exploration Feature, since the multimodal functions F8 to F20 have many local optima with the number increasing exponentially with dimension, then these functions are suitable to test the exploration behavior of the algorithm. The results of Table 12 to 17 of the compared algorithms showed very strong results for the CSGWO over other algorithms for the basic and the expanded multimodal benchmark functions and excellent results for the Hybrid Composition benchmark functions. Thus it can be inferred that CSGWO has a very good exploration feature.

Regarding Avoidance Local Minima, the Hybrid Composition benchmark functions F15 to F20 are classified as very challenging for any meta-heuristic algorithm because they have a large number of local optima values that have to be avoided. A good algorithm has to avoid them all to reach the global minima, the results showed that the CSGWO exhibits a very competitive result in local minima avoidance, and exploitation features in the Hybrid Composition benchmark functions.

We compared the algorithms and the obtained results for the unimodal benchmark functions we applied student t test to see if the results are significant and the p values of the student T test over 30 runs the results for the unimodal benchmark functions F1 to F5 are shown in Table 10, whereas the significant results has been shaded where  $p < 0.05$ . multimodal expanded benchmark functions F8 to F12 t test are shown in Table 11, on multimodal Basic benchmark functions F13 to F14 displayed in Table 12, Hybrid Composition benchmark functions F15 to F20 student t test results seen in table 13. Student t test has been applied to have a fair comparison over the 30 runs in order to draw a statistically meaningful conclusion. This statistical test must be done due to the stochastic nature of meta-heuristics.

Table 10. p values of the student T test over 30 runs (The significant results has been shaded where  $p < 0.05$ ) for the unimodal benchmark functions F1 to F5

	GWO	SCA	PSO	MVO
1	<b><u>0.009492</u></b>	0.055118	<b><u>0.001542</u></b>	<b><u>9.11E-20</u></b>
2	<b><u>0.014502</u></b>	<b><u>0.001247</u></b>	<b><u>0.006904</u></b>	<b><u>7.64E-23</u></b>
3	0.321825	<b><u>0.013772</u></b>	<b><u>1.67E-12</u></b>	<b><u>1.3E-14</u></b>
4	<b><u>7.32E-08</u></b>	<b><u>1.6E-05</u></b>	<b><u>5.23E-14</u></b>	<b><u>5.62E-14</u></b>
5	<b><u>6.45E-20</u></b>	<b><u>2.66E-10</u></b>	<b><u>0.000883</u></b>	0.204482

The results of tables 10 and table 11 shows that the proposed algorithm is able to provide very competitive results on the unimodal test functions. The p values also prove that the competitive results are significant in the majority of the results. This testifies that the proposed algorithm has high exploitation ability. Also, the results were almost significant for all of the functions from F1 to F5 for the 30 experiment run.

Table 11. p values of the student T test over 30 runs (The significant results has been shaded where  $p < 0.05$ ) for multimodal Basic benchmark functions F6 to F12

	GWO	SCA	PSO	MVO
6	<b><u>3.75E-39</u></b>	<b><u>5.29E-28</u></b>	<b><u>3.72E-40</u></b>	<b><u>7.04E-40</u></b>
7	<b><u>3.4E-08</u></b>	<b><u>6.81E-08</u></b>	<b><u>4.43E-18</u></b>	<b><u>6.94E-14</u></b>
8	<b><u>3.49E-24</u></b>	<b><u>4.71E-19</u></b>	<b><u>1.14E-22</u></b>	<b><u>1.05E-24</u></b>
9	<b><u>8.89E-19</u></b>	<b><u>0.025619</u></b>	<b><u>5.36E-19</u></b>	<b><u>6.95E-20</u></b>
10	<b><u>4.81E-11</u></b>	<b><u>8.93E-05</u></b>	<b><u>1.2E-06</u></b>	<b><u>0.004302</u></b>
11	0.089517	0.059868	<b><u>1.15E-06</u></b>	<b><u>6.14E-19</u></b>
12	<b><u>4.27E-25</u></b>	<b><u>2.08E-15</u></b>	<b><u>2.22E-25</u></b>	<b><u>0.046129</u></b>

Table 12. p values of the student T test over 30 runs (The significant results has been shaded where  $p < 0.05$ ) the Multimodal expanded Functions F13 to F14

	GWO	SCA	PSO	MVO
13	<b><u>1.21E-39</u></b>	<b><u>3.55E-26</u></b>	<b><u>3.06E-45</u></b>	<b><u>3.28E-45</u></b>
14	<b><u>4.59E-15</u></b>	<b><u>1.68E-15</u></b>	<b><u>3.65E-15</u></b>	<b><u>1.65E-15</u></b>

Table 13. p values of the student T test runs (The significant results has been shaded where  $p < 0.05$ ) for the Hybrid Composition Functions F15 to F20

	GWO	SCA	PSO	MVO
15	0.926088	<b><u>0.031044</u></b>	<b><u>0.003283</u></b>	0.808711
16	<b><u>9.08E-07</u></b>	<b><u>9.12E-07</u></b>	<b><u>9.08E-07</u></b>	<b><u>9.08E-07</u></b>
17	<b><u>0.001415</u></b>	<b><u>0.00147</u></b>	<b><u>0.001415</u></b>	<b><u>0.001415</u></b>
18	0.160926	0.160926	0.160926	0.160926
19	<b><u>1.64E-05</u></b>	<b><u>7.67E-05</u></b>	<b><u>9.22E-06</u></b>	<b><u>9.22E-06</u></b>
20	<b><u>1.98E-13</u></b>	<b><u>1.32E-12</u></b>	<b><u>7.67E-14</u></b>	<b><u>2.83E-14</u></b>

After discussing the exploitation feature of CSGWO in the unimodal test functions, we are going to discuss the exploration feature of CSGWO. To discuss this we will refer to the results obtained from multi-model test function Table 7 to 9. And the significant of the results has been testified using student t test over the tested functions The results showed that the proposed algorithm is able to provide a very good exploration behavior. The p values suggest that the significant of the CSGWO algorithm is results in most factions. This indicates that that CSGWO results are competitive in exploration feature as shown in table 12.

The results also showed that the CSGWO has a good local optima avoidance. In fact, stagnation in local solutions can be resolved by promoting exploration. local optima avoidance of CSGWO is also competitive as seen in the results of the hybrid Composite test functions that provide a balance between exploration and exploitation features, due to the difficulty of this set of test functions.

#### 4.1 Stability of the CSGWO

The standard deviation values for all functions from F1 to F20 shown in Table 18. except for F8 are all near the zero and this means that CSGWO is stable for the 30 experiments that were performed, the reason for the high standard deviation value of F8 is due to the very large values in the search space; see Fig. 4.

Table 18. The standard deviation vales functions F1 to F20

std		std	
<b>F1</b>	0	<b>F11</b>	0
<b>F2</b>	0	<b>F12</b>	6.49E-16
<b>F3</b>	0	<b>F13</b>	0
<b>F4</b>	0	<b>F14</b>	0.156587
<b>F5</b>	0.076894	<b>F15</b>	0.093118
<b>F6</b>	0.316552	<b>F16</b>	3.429023
<b>F7</b>	5E-05	<b>F17</b>	0.000692
<b>F8</b>	533.343	<b>F18</b>	0.01386
<b>F9</b>	0	<b>F19</b>	0.04732
<b>F10</b>	0	<b>F20</b>	22.03944

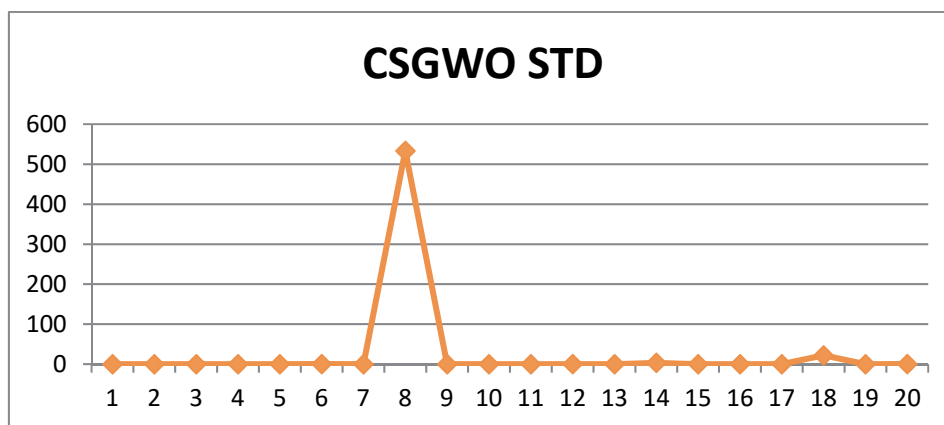
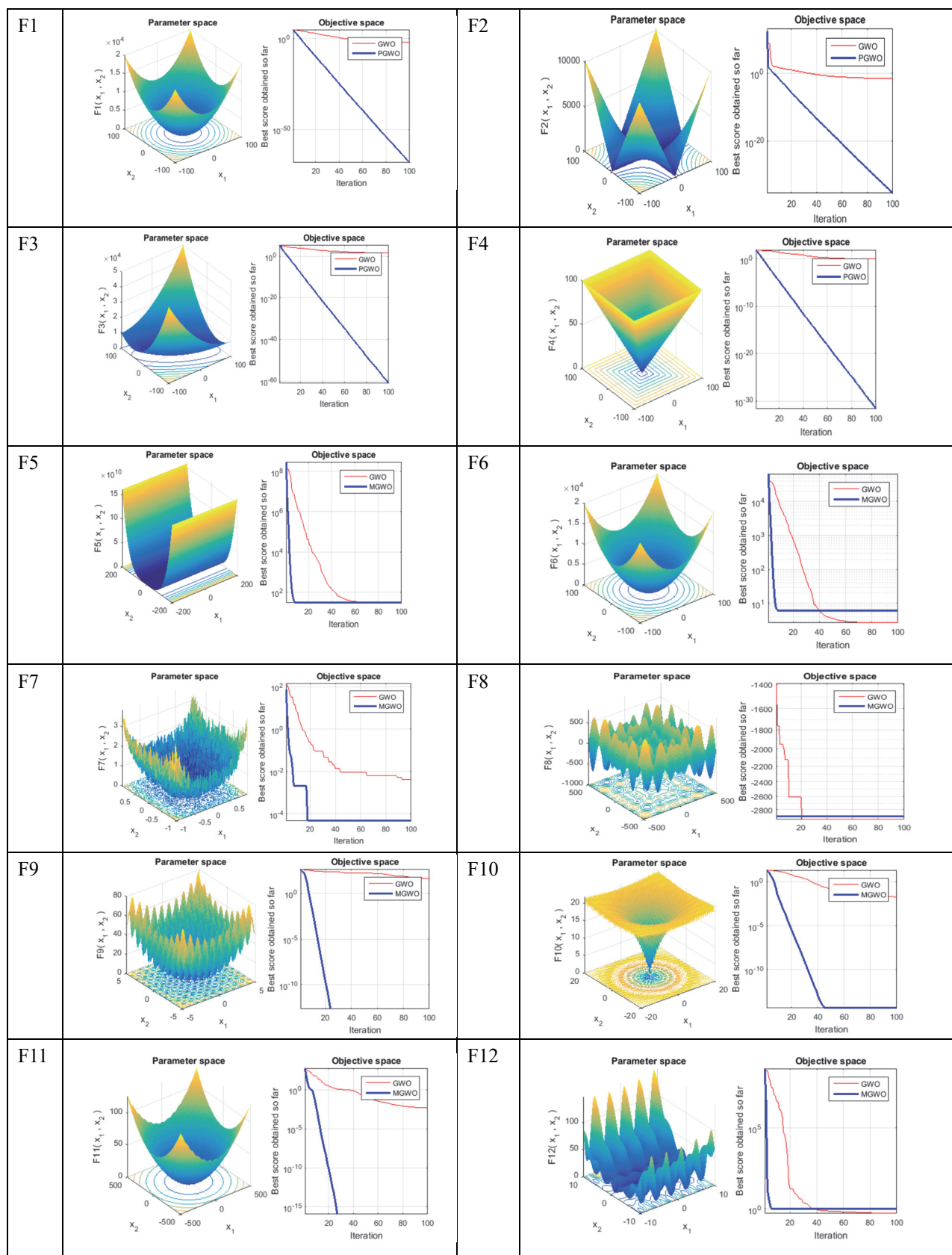


Figure 4. Plot of the Standard Deviation Vales for all Functions: F1 to F20

#### Convergence Curve Analysis

To test the speed of the CSGWO in finding the optimal solution of the benchmark functions we drew the converge curve of CSGWO compared to PSO, GWO, SCA and MVO. As shown in table 14. It can be seen from the table that the CSGWO is very fast in finding the global minima for major of the tested 20 functions when compared to all of PSO, GWO, SCA and MVO.

Table 20. Converge curve for the benchmark functions over CSGWO compared to PASO, GWO, SCA and MVO



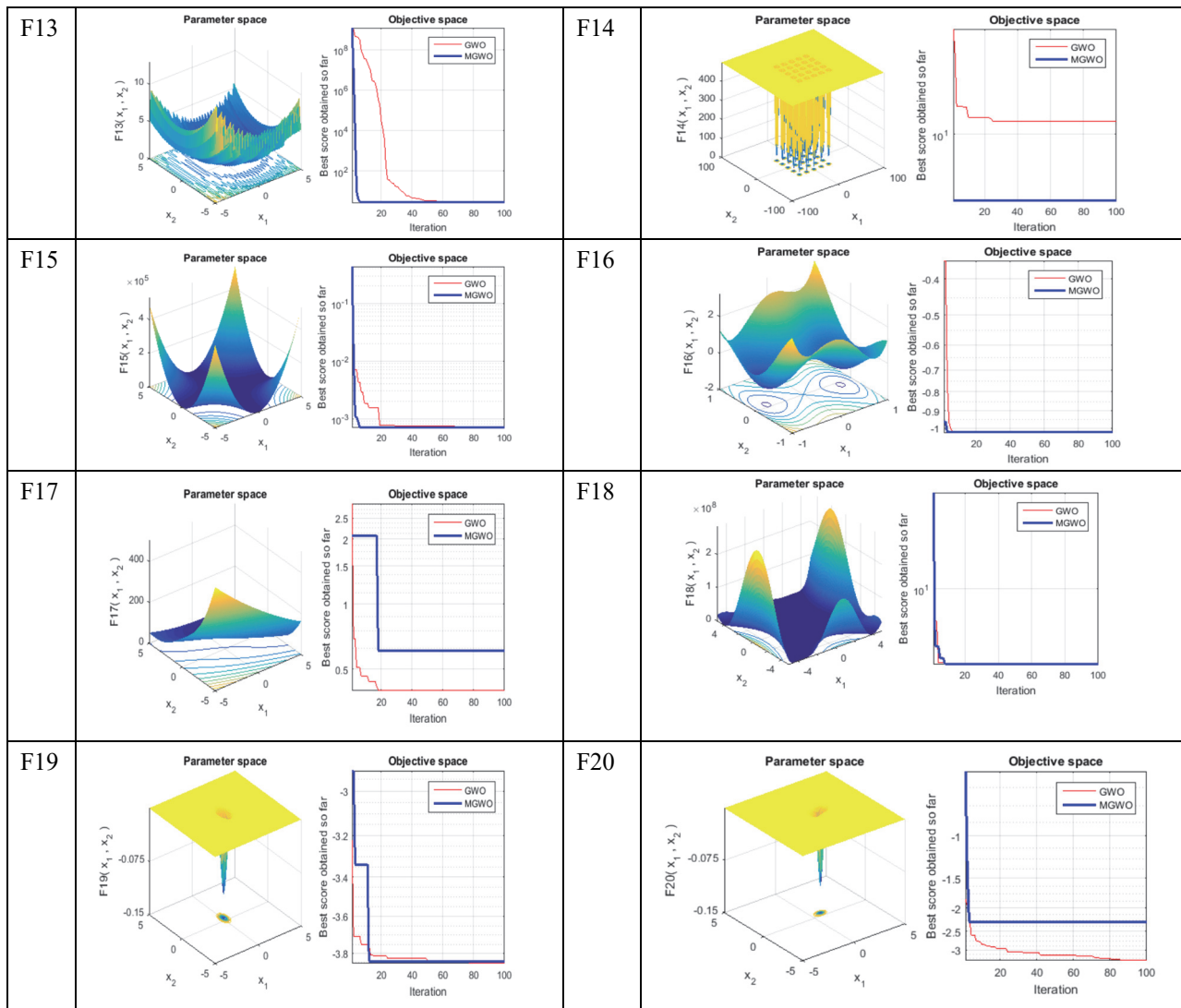


Figure 4. Convergence curve of supernova algorithm in comparison with GWO, SCA, MVO, MFO, PLOARPSO, WOA, and PSO

It can be noticed from figure 4 that CSGWO outperformed all of the six optimization algorithms (GWO, SCA, MVO, MFO, WOA, and PSO) and it was the fastest to find the best solution as shown in the Convergence curve chart for each iteration for the bench mark functions from F1 to F5, which represent Unimodal Functions (Shifted Sphere Function, Shifted Schwefel's Problem 1.2, Shifted Rotated High Conditioned Elliptic Function, Shifted Schwefel's Problem 1.2 with Noise in Fitness, Schwefel's Problem 2.6 with Global Optimum on Bounds) and F7 which is a Multimodal Basic Functions (Shifted Rotated Griewank's Function without Bounds). This testifies that the proposed algorithm has a high exploitation ability. It can be also observed that CSGWO optimizer succeed to be the fastest and most effective optimizer for all of the compared algorithms (GWO, SCA, MVO, MFO, WOA, and PSO) except the POLARPSO algorithm when it is tested for the bench mark functions F9- F13 which is from the Multimodal Functions set for Shifted Rastrigin's Function, Shifted Rotated Rastrigin's Function, Shifted Rotated Weierstrass Function respectively. The analysis of results also shows that supernova optimizer showed a competitive and effective performance in finding the best solution that is almost similar to the compared algorithms for the functions F14, F15, F16 which represent more complex problems from the set of Multimodal Expanded Functions and Hybrid Composition Functions that represent the functions of (Shifted Rotated Expanded Scaffer's, Rotated Hybrid Composition Function with the Global Optimum on the Bounds, Rotated Hybrid Composition Function with High Condition Number Matrix, Non-Continuous Rotated Hybrid Composition Function)

## 5. Conclusion

In this work, we proposed an enhanced optimization technique that is based on the well know PSO algorithm. We

called the proposed algorithm CSGWO. We tested our proposed algorithm on the CEC2005 benchmark set problems of 20 test functions classified in four categories: unimodal, multi-modal (basic and expanded) and the new hybrid composite functions. We compared the results recorded for CSGWO with four state-of-the-art meta-heuristic optimization algorithms: PSO, GWO, SCA, and MVO. The results showed that CSGWO showed a competitive result in finding the optimum solution and it outperformed the other meta-heuristic algorithms in term of speed of finding the solutions as illustrated in the coverage curve analysis.

## References

- "Introduction to the Polar Coordinate System." Boundless Algebra Boundless, 14 Oct. 2016. Retrieved 24 Jan. 2017 from <https://www.boundless.com/algebra/textbooks/boundless-algebra-textbook/complex-numbers-and-polar-coordinates-344/the-polar-coordinate-system-417/introduction-to-the-polar-coordinate-system-418-17463/>
- Abbass, H. A. (2001). MBO: Marriage in honey bees optimization-A haplometrosis polygynous swarming approach. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on* (Vol. 1, pp. 207-214). IEEE.
- Amjad, A. H., & Hussam, N. (2018). Fakhouri, Supernova Optimizer: A Novel Natural Inspired Meta-Heuristic, *Modern Applied Science*, 12(1). Published by Canadian Center of Science and Education.
- Askarzadeh, A., & Rezazadeh, A. (2013). A new heuristic optimization algorithm for modeling of proton exchange membrane fuel cell: bird mating optimizer. *International Journal of Energy Research*, 37(10), 1196-1204.
- Binh, H. T. T. (2013). Hybrid particle swarm optimization for Solving multi-area economic dispatch Problem. *International Journal on Soft Computing*, 4(2), 17.
- Boyer, C. B. (1949). Newton as an originator of polar coordinates. *The American Mathematical Monthly*, 56(2), 73-78.
- Brown, R. G., Gleason, A. M., & Brown, M. A. (1992). *Advanced mathematics: precalculus with discrete mathematics and data analysis*. Houghton Mifflin Company.
- Campos, M., Bonabeau, E., Theraulaz, G., & Deneubourg, J. L. (2000). Dynamic scheduling and division of labor in social insects. *Adaptive Behavior*, 8(2), 83-95.
- Clerc, M. (2012). Standard particle swarm optimisation.
- Coolidge, J. L. (1952). The origin of polar coordinates. *The American Mathematical Monthly*, 59(2), 78-85.
- Dorigo, M., Birattari, M., & Stutzle, T. (2006). Ant colony optimization. *IEEE computational intelligence magazine*, 1(4), 28-39.
- Eberhart, R., & Kennedy, J. (1995, October). A new optimizer using particle swarm theory. In *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on* (pp. 39-43). IEEE.
- Gandomi, A. H., & Alavi, A. H. (2012). Krill herd: a new bio-inspired optimization algorithm. *Communications in Nonlinear Science and Numerical Simulation*, 17(12), 4831-4845.
- Gimmler, J., Stützle, T., & Exner, T. E. (2006, September). Hybrid particle swarm optimization: An examination of the influence of iterative improvement algorithms on performance. In *International Workshop on Ant Colony Optimization and Swarm Intelligence* (pp. 436-443). Springer Berlin Heidelberg.
- Goldberg, E. D. (1989). *Genetic Algorithms in Search, Optimisation and Machine Learning*, Addison Wesley PC.
- Handl, J., & Meyer, B. (2007). Ant-based and swarm-based clustering. *Swarm Intelligence*, 1(2), 95-113.
- Karaboga, D. (2005). An idea based on honey bee swarm for numerical optimization (Vol. 200). Technical report-tr06, Erciyes university, engineering faculty, computer engineering department.
- Kennedy, J., & Eberhart, R. C. (1995). Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks*, Piscataway, NJ. pp. 1942-1948.
- Khanesar, M. A., Teshnehlab, M., & Shoorehdeli, M. A. (2007, June). A novel binary particle swarm optimization. In *Control & Automation, 2007. MED'07. Mediterranean Conference on* (pp. 1-6). IEEE.
- Li, X. L. (2003). A new intelligent optimization-artificial fish swarm algorithm. Doctor thesis, Zhejiang University of Zhejiang, China.
- Lu, X., & Zhou, Y. (2008, September). A novel global convergence algorithm: bee collecting pollen algorithm. In *International Conference on Intelligent Computing* (pp. 518-525). Springer Berlin Heidelberg.

- Mirjalili, S. (2016). SCA: a sine cosine algorithm for solving optimization problems. *Knowledge-Based Systems*, 96, 120-133.
- Mirjalili, S., Mirjalili, S. M., & Hatamlou, A. (2016). Multi-verse optimizer: a nature-inspired algorithm for global optimization. *Neural Computing and Applications*, 27(2), 495-513.
- Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, 69, 46-61.
- Mucherino, A., & Seref, O. (2007, November). Monkey search: A novel metaheuristic search for global optimization. In O. Seref, O. E. Kundakcioglu, & P. Pardalos (Eds.), *AIP conference proceedings* (Vol. 953, No. 1, pp. 162-173). AIP.
- Pan, W. T. (2012). A new fruit fly optimization algorithm: taking the financial distress model as an example. *Knowledge-Based Systems*, 26, 69-74.
- Pham, D. T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S., & Zaidi, M. (2005). The bees algorithm. Technical note. Manufacturing Engineering Centre, Cardiff University, UK, 1-57.
- Pinto, P. C., Runkler, T. A., & Sousa, J. M. (2007, April). Wasp swarm algorithm for dynamic MAX-SAT problems. In *International Conference on Adaptive and Natural Computing Algorithms* (pp. 350-357). Springer Berlin Heidelberg.
- Rizik, M. H., Al-Sayyed, H. N., Fakhouri, Ali, R., & Colin, P. (2017). Polar Particle Swarm Algorithm for Solving Cloud Data Migration Optimization Problem. *Modern Applied Science*, 11(8). ISSN 1913-1844 E-ISSN 1913-1852. Published by Canadian Center of Science and Education.
- Roth, M. (2005). Termite: A swarm intelligent routing algorithm for mobile wireless ad-hoc networks.
- Selvan, S. E., Subramanian, S., & Solomon, S. T. (2003, April). Novel technique for PID tuning by particle swarm optimization. In *Proc. 7th Annu. Swarm Users/Researchers Conf. (SwarmFest 2003)*.
- Shiqin, Y., Jianjun, J., & Guangxing, Y. (2009, May). A dolphin partner optimization. In *Intelligent Systems, 2009. GCIS'09. WRI Global Congress on* (Vol. 1, pp. 124-128). IEEE.
- Yang, X. S. (2010). Firefly algorithm, stochastic test functions and design optimisation. *International Journal of Bio-Inspired Computation*, 2(2), 78-84.
- Yang, X. S. (2011, May). Metaheuristic optimization: Algorithm analysis and open problems. In *International Symposium on Experimental Algorithms* (pp. 21-32). Springer Berlin Heidelberg.
- Yang, X. S., & Deb, S. (2009, December). Cuckoo search via Lévy flights. In *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on* (pp. 210-214). IEEE.
- Yu, S., Wu, Z., Wang, H., Chen, Z., & Zhong, H. (2012). A HYBRID PARTICLE SWARM OPTIMIZATION ALGORITHM BASED ON SPACE TRANSFORMATION SEARCH AND A MODIFIED VELOCITY MODEL. *International Journal of Numerical Analysis & Modeling*, 9(2).
- Yushui, G., & Jiaheng, Y. (2015, August). Cloud Data Migration Method Based on PSO Algorithm. In *Distributed Computing and Applications for Business Engineering and Science (DCABES), 2015 14th International Symposium on* (pp. 143-146). IEEE.
- Yushui, G., & Jiaheng, Y. (2015, August). Cloud Data Migration Method Based on PSO Algorithm. In *2015 14th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)* (pp. 143-146). IEEE.
- Yushui, G., Jiaheng, Y., & Tao, S. (2016). Cloud Data Migration Method Based on ABC Algorithm. *International Journal of Database Theory and Application*, 9(5), 141-148.

## Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).