Journal of Ambient Intelligence and Humanized Computing
https://doi.org/10.1007/s12652-020-02153-1

ORIGINAL RESEARCH

A better exploration strategy in Grey Wolf Optimizer

Jagdish Chand Bansal[1] · Shitu Singh[1]

Received: 22 April 2020 / Accepted: 25 May 2020

Abstract
The Grey Wolf Optimizer (GWO) is a recently developed population-based meta-heuristics algorithm that mimics the leadership hierarchy and hunting mechanism of grey wolves in nature. Although, GWO has shown very good results on several real-life applications but still it suffers from some issues like, the low exploration and slow convergence rate. Therefore in this paper, an improved grey wolf optimizer is proposed to modify the exploration as well as exploitation abilities of the classical GWO. This improvement is performed by using the explorative equation and opposition-based learning (OBL). The validation of the proposed modification is done on a set of 23 standard benchmark test problems using statistical, diversity and convergence analysis. The experimental results on test problems confirm that the efficiency of the proposed algorithm is better than other considered metaheuristic algorithms.

Keywords Swarm intelligence · Grey wolf optimizer · Explorative equation · Opposition-based learning (OBL) · Exploration and exploitation

1 Introduction

Over recent years, Swarm intelligence has become a prominent area in the field of nature−inspired techniques. Mostly, it is being used to solve real-world optimization problems. It is based on the collective behavior of creatures living in swarms or colonies. Swarm based optimization algorithms find a solution by the collaborative trial and error method. The well-known techniques in this category are Particle Swarm Optimization (PSO) (Kennedy 2010), Artificial Bee Colony(ABC) (Karaboga 2005), Ant Colony Optimization (ACO) (Dorigo et al. 2006), Firefly Optimization (FO) (Yang 2010), Cuckoo Search Optimization (CSO) (Yang and Deb 2009), Genetic Algorithm (GA) (Goldberg 2006), Spider Monkey Optimization (SMO) (Bansal et al. 2014), Ant lion optimizer (Mirjalili 2015a), Harris hawks optimization (HHO) (Heidari et al. 2019), Moth-flame Optimisation (MFO) (Mirjalili 2015c) and Grey Wolf Optimizer (GWO) (Mirjalili et al. 2014). GWO algorithm is comparatively a new and well-regarded optimization algorithm in the field of swarm intelligence. GWO, proposed by (Mirjalili et al. 2014) in 2014, is inspired by the leadership hierarchy and hunting mechanism of grey wolves in nature.

In the last few years, significant growth in the application of GWO is observed for solving different real-life application problems. GWO has been used for training Multi-Layer Perceptron (MLP) (Mirjalili 2015b). Emary et al. (2016) have proposed a novel binary version of GWO and used to select optimal feature subset for classification purposes. Mohanty et al. (2015) have proposed a new maximum power point tracking (MPPT) design using the GWO technique for the photovoltaic system under partial shading conditions. In 2015 (Sulaiman et al. 2015), GWO is used for solving optimal reactive power dispatch problem (ORPD). Komaki and Kayvanfar (2015) have proposed the GWO algorithm for the two-stage assembly flow shop scheduling problem with a release time of jobs which is applicable in many industrial areas such as computer manufacturing industry, fire engine assembly plant, etc. Jayabarathi et al. (2016) proposed economic dispatch using a hybrid GWO for solving economic dispatch problems that are nonlinear, non-convex and discontinuous in nature, with numerous equality and inequality constraints. El-Fergany and Hasanien (2015) proposed single and multi-objective optimal power flow using GWO and DE algorithms. GWO algorithm-based tuning of fuzzy control systems with reduced parametric sensitivity (Precup et al.

✉ Jagdish Chand Bansal
jcbansal@sau.ac.in

Shitu Singh
singh.shitu93@gmail.com

[1] South Asian University, New Delhi, India

Published online: 16 June 2020

2016). Jayakumar et al. (2016) proposed GWO for Combined Heat and Power Dispatch (CHPD) with cogeneration systems. Kamboj et al. (2016) proposed solution of non-convex economic load dispatch problem using GWO and many other applications have shown the ability of GWO in terms of exploration strength compared to other algorithms.

The aim of the present work is to increase the search mechanism of the classical GWO with the help of two important concepts known as explorative equation which has been firstly used in the exploration phase of (Heidari et al. 2019) and opposition-based learning (OBL) (Tizhoosh 2005). Here the explorative equation is introduced to increase the exploration ability of classical GWO. To handle the stagnation problem and maintaining the faster convergence, OBL is incorporated into the classical GWO. In this paper, the greedy selection approach is also engaged within the algorithm to attract the GWO toward the search domain of search space. To evaluate the performance of the proposed algorithm, a set of 23 classical benchmark test functions have been taken. The experimental results show that the improved GWO is a better optimizer.

The remaining paper is organized as follows—a brief description of the classical GWO and OBL are explained in Sect. 2. In Sect. 3, a modified version of GWO is presented. In Sect. 4, a complete set of experimental results with different analyses are given. Section 5 concludes the work.

2 Preliminaries

2.1 Grey Wolf Optimizer

The GWO algorithm was developed in 2014 by Mirjalili et al. (2014). The GWO algorithm is inspired by social and leadership behavior of grey wolves. Grey wolves, also known as Canis Lupus always live in a pack of approximately 5–12 wolves. In the pack, the Dominant hierarchy is the main feature of a grey wolf pack. To maintain the discipline within the pack, their pack is divided into four types of wolves (Mirjalili et al. 2014). The first category is a leader wolf called alpha (α) wolves which are the dominant wolf of the pack and are the decision-maker. The second category is beta (β) wolves which are subordinate to the α in their absence and works as a messenger for the α wolves. The third category wolves are delta (δ) wolves which are caretakers of the pack and protect the pack from enemies and the fourth category consists of those wolves that have permission to eat only in the end. These wolves are called omega (ω) wolves which commonly play the role of scapegoat. The ω wolves are very important part of the pack because in the absence of these wolves, the wolf pack may face internal fighting and problems. This is due to the venting of violence and frustration of all grey wolves by the omega wolves. These wolves also assist in satisfying the entire pack and maintaining the dominance structure. In some cases, ω wolves also work as babysitters.

The group hunting is another important social behavior of grey wolves in the pack. According to (Muro et al. 2011) their hunting process involves three steps-

1. Chasing and approaching the prey
2. Encircling the prey
3. Attacking the prey.

2.1.1 Mathematical model

In this subsection, we describe the mathematical models of social hierarchy, encircling prey, hunting, attacking prey and search for prey.

2.1.2 Social hierarchy

The algorithm starts with three best solutions which are assigned random positions in the search space. α, β and δ correspond to the first best, the second-best, and the third-best solutions respectively, and other solutions are considered as ω. The ω solutions iteratively improve their positions by following α, β and δ wolves.

2.1.3 Encircling prey

The encircling process of grey wolves is calculated from the following mathematical equations:

$$x^{(t+1)} = x^t_{prey} - A \times D \tag{1}$$

where,

$$A = 2 \times a \times r_1 - a \tag{2}$$

$$a = 2 - 2\left(\frac{t}{T}\right) \tag{3}$$

$$D = |C \times x^t_{prey} - x^t| \tag{4}$$

$$C = 2 \times r_2 \tag{5}$$

and x^t_{prey} is the position of the prey at tth iteration, $x^{(t+1)}$ and x^t are the positions of the wolf at $(t+1)th$ and tth iterations, respectively. It is clear from Eq. (2) that the value of vector A is in the range $[-a, a]$ which is responsible for exploration and exploitation. a is a linearly decreasing vector from 2 to 0 over iterations and calculated from Eq. (3). At the initial iteration, the value of vector A is maximum, but as iteration counter progresses the value of A decreases. It is shown in Fig. 1. C is a random vector in the range [0, 2]. r_1 and r_2 are random numbers in [0, 1]. T denotes the maximum number of iterations.

Equation (1) indicates that the wolves to decrease distances from their prey (x^t_{prey}). The distance depends on the vector A and D.

2.1.4 Hunting

For mathematically simulate the hunting behavior of grey wolves, it is assumed that the three best solutions α, β and δ wolves have better knowledge about the location of the prey (optimal solution). In this way, the updated positions of each wolf based on the positions of α, β and δ using the above equations are (Mirjalili et al. 2014):

$$x_1 = x^t_\alpha - A_1 \times D_\alpha \tag{6}$$

$$x_2 = x^t_\beta - A_2 \times D_\beta \tag{7}$$

$$x_3 = x^t_\delta - A_3 \times D_\delta \tag{8}$$

where,

$$\begin{aligned} &A_1 = 2\times a \times r'_1 - a, \quad A_2 = 2\times a\times r'_2 - a, \quad A_3 = 2\times a\times r'_3 - a \\ &D_\alpha = |C_1 \times x^t_\alpha - x^t|, \quad D_\beta = |C_2\times x^t_\beta - x^t|, \quad D_\delta = |C_3\times x^t_\delta - x^t| \\ &C_1 = 2\times r''_1, \quad C_2 = 2\times r''_2, \quad C_3 = 2\times r''_3 \\ &x^{(t+1)} = \frac{x_1 + x_2 + x_3}{3} \end{aligned} \tag{9}$$

Here, x^t_α, x^t_β and x^t_δ are the positions of α, β and δ wolf at tth iteration and x^t represents the solution in tth iteration. r'_1, r'_2, r'_3, r''_1, r''_2 and r''_3 are random vectors whose components are in the range [0, 1]. A_1, A_2 and A_3 are random vectors in the range $[-a, a]$. a is defined by equation (3). C_1, C_2 and C_3 are random vectors in the range [0, 2]. Equation (9) is the positions update equation of grey wolves and average of positions corresponding to α, β and δ wolf.

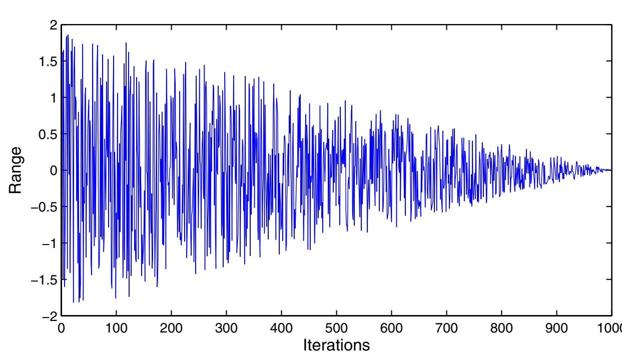


Fig. 1 Decreasing pattern for range of A

2.1.5 Attacking prey (Exploitation) and search for prey (Exploration)

In the GWO algorithm, A and C are responsible for explorative and exploitative behaviors. The A is a random value in the range $[-a, a]$. a is linearly decreasing 2 to 0 and defined by Eq. (5). When $|A| > 1$ and $C > 1$, the wolf explores the whole search space. In addition, exploitation happens when $|A| < 1$ and $C < 1$.

The pseudo code of GWO algorithm is presented in Algorithm 1.

Algorithm 1 Grey Wolf Optimizer (GWO) algorithm

```
Initialize the parameters
Initialize the n grey wolves positions say x_i (i = 1, 2, ... n)
Evaluate the fitness say f(x_i) at x_i
Select α, β, and δ wolf
initalize a, A and C
initialize t = 0
while Termination criteria is meet do
    for for each wolf do
        update the position of wolves using (9)
    end for
    update a, A and C
    update α, β, and δ wolf
    t=t+1
end while
return the α wolf
```

2.2 Opposition-based learning (OBL)

The concept of OBL is introduced in 2005 (Tizhoosh 2005) which is used by many researchers to enhance the ability to learn, search and optimize the metaheuristic algorithms. In OBL, for each solution of the population, opposite solutions are generated and then the best solutions (equal to population size) are retained so that the population size remains the same. The opposite of a real number $x \in [a, b]$ can be given by $\bar{x}$ as follows:

$$\bar{x} = a + b - x \tag{10}$$

a and b are the lower and upper bounds of the solution space in $\mathbb{R}$. In $D-$ dimensional space the opposite solution $\bar{x}$=$(\bar{x}_1, \bar{x}_2, \bar{x}_3, \dots, \bar{x}_D)$ of a vector $x = (x_1, x_2, \dots, x_D) \in \mathbb{R}^D$ can be calculated as:

$$\bar{x}_j = a_j + b_j - x_j, \quad j = 1, 2, \dots, D \tag{11}$$

Here, a_j and b_j are the lower and upper bounds of the solution space in jth dimension. Finally, in the process of optimization method, the current solution x is selected if $f(x)$ is better than $f(\bar{x})$; otherwise, $\bar{x}$ is selected.

3 Proposed method

The classical GWO algorithm has fewer number of parameters to perform the search and is easy to implement. However, the experimental analyses show that in some cases, it is prone towards the local optima due to insufficient exploration ability of grey wolves. Hence, there is scope of enhancing the explorative ability of wolves to make it a better optimizer. Therefore, we have introduced an explorative equation (Heidari et al. 2019) for the grey wolves, so that the large area of the search space can be explored. In addition to this, to maintain the convergence speed (or in fact to make better) efficiently, an OBL (Tizhoosh 2005) is used for leading wolves in each iteration.

The description of all the applied strategies is as follows:

First, an explorative equation for the wolves is introduced, which is given by:

$$x^{(t+1)} = \begin{cases} x^t_{rand} - r_1 \times |(x^t_{rand} - 2 \times r_2 \times x^t)|, & r_5 \geq 0.5. \\ (x^t_{alpha} - x^t_a) - r_3 \times (lb + r_4 \times (ub - lb)), & r_5 < 0.5. \end{cases} \tag{12}$$

where $x^{(t+1)}$ is the position of grey wolf in the $(t + 1)th$ iteration, x^t_{alpha} is the position vector of alpha wolf at tth iteration, x^t is the position vector of grey wolf in the tth iteration, r_1, r_2, r_3, r_4 and r_5 are random numbers in [0, 1], lb and ub are the lower and upper bounds of decision variables. x^t_{rand} is a randomly selected wolf at tth iteration from the population and x_a is the average position of grey wolves's position. In this explorative equation, the current new solution is generated around the random solution or best solution which helps to enhance exploration and communications among the grey wolves. The newly generated positions of wolves will pass to the next iteration or not, this will decide by the greedy apporach described below:

$$x^{(t+1)} = \begin{cases} x^{(t+1)}, f(x^{(t+1)}) \leq f(x^t) \\ x^t, \quad otherwise \end{cases} \tag{13}$$

where $x^{(t+1)}$ and x^t are the positions of wolves at $(t + 1)$th and tth iterations, respectively. $f(x^{(t+1)})$ is the fitness value at $(t + 1)$th solution and $f(x^t)$ is the fitness value at tth solution. Here it is assumed that the optimization problem under consideration is a minimization problem. If the explorative equation (12) fails to provide a better position then the equation (9) of the classical GWO is applied to update the solution x.

In addition to this, to maintain or to improve the convergence rate and to avoid stagnation, an OBL is employed for leading wolves (α, β and δ wolf). The opposite solution corresponding to leading wolves are calculated as follows:

$$\overline{\mathrm{x}}_m = (lb + ub) - x_m \tag{14}$$

where, $m = 1$ refers to α wolf, $m = 2$ refers to β wolf and $m = 3$ refers to δ wolf. $\overline{\mathrm{x}}_m$ is the opposite position of leading wolf (α, β or δ wolf) and x_m is the position of leading wolf (α, β or δ wolf) . lb and ub are the lower and upper bounds of the decision variable. After that, all the updated wolves are sorted and then, the last 3 wolves having worse fitness are replaced by the opposite wolves, which are obtained by Eq. (14). The steps of the proposed algorithm Improved

Table 1 Unimodal benchmark functions

Name	Benchmark	*dim*	Range	$f_{\min}$
Sphere function	$f1(x) = \sum_{i=1}^{n} x_i^2$	30	[− 100,100]	0
Schwefel's problems 2.22	$f2(x) = \sum_{i=1}^{n} \lvert x_i^2\rvert + \prod_{i=1}^{n} \lvert x_i\rvert$	30	[− 10,10]	0
Schwefel's problems 1.2	$f3(x) = \sum_{i=1}^{n} \left(\sum_{j-1}^{i} x_j\right)^2$	30	[− 100,100]	0
Schwefel's problems 2.21	$f4(x) = \max_i \left\{\lvert x_i\rvert, 1 \leq i \leq n\right\}$	30	[− 100,100]	0
Generalized Rosenbrock's function	$f5(x) = \sum_{i=1}^{n-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + \left(x_i - 1\right)^2\right]$	30	[− 30,30]	0
Step function	$f6(x) = \sum_{i=1}^{n} \left(\left[x_i + 0.5\right]\right)^2$	30	[− 100,100]	0
Quartic function with noise	$f7(x) = \sum_{i=1}^{n} ix_i^4 + random[0, 1)$	30	[− 1.28,1.28]	0

Table 2 Multimodal benchmark functions

Name	Benchmark	*dim*	*Range*	$f_{\min}$
Generalized Schwefel's problem 2.26	$f8(x) = \sum_{i=1}^{n} -x_i \sin\left(\sqrt{\lvert x_i \rvert}\right)$	30	[− 500,500]	− 418.9829 $\times dim$
Generalized Rastrigin's function	$f9(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos\left(2\pi x_i\right) + 10\right]$	30	[− 5.12,5.12]	0
Ackley's function	$f10(x) = -20\exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos\left(2\pi x_i\right)\right) +20 + e$	30	[- 32,32]	0
Generalized Griewank function	$f11(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	[− 600,600]	0
Generalized penalized function	$f12(x) = \frac{\pi}{n}\left\{10\sin\left(\pi y_1\right) + \sum_{i=1}^{n-1}(y_i - 1)^2\left[1 + 10\sin^2(\pi y_{i+1})\right] +(y_n - 1)^2\right\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_i+1}{4} u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	30	[− 50,50]	0
Generalized penalized function	$f13(x) = 0.1\left\{\sin^2(3\pi x_1) + \sum_{i=1}^{n}\left(x_i - 1\right)^2\left[1 + \sin^2(3\pi x_i + 1)\right] +(x_n - 1)^2\left[1 + \sin^2(2\pi x_n)\right]\right\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	30	[− 50,50]	0

GWO (IGWO) are presented in the form of pseudo code in Algorithm 2.

Algorithm 2 Improved grey wolf optimizer (IGWO) algorithm

```
Initialize the parameters
Initialize n grey wolves positions say x_i(1 = 1, 2, ... n)
Evaluate the fitness f(x_i) at x_i
Select α, β, and δ wolf
initalize a, A and C
t = 0
while Termination criteria is meet do
  for i = 1 : n do
    Update the positions of grey wolves using (12) say x_i^*
    Evaluate the fitness say f(x_i^*) at x_i^*
    if f(x_i^*) ≤ f(x_i) then
      f(x_i) = f(x_i^*)
      x_i = x_i^*
    else
      Update the positions of each grey wolf using position updated equation of GWO algorithm say (x_i^**)
      Evaluate the fitness at x_i^** say f(x_i^**)
    end if
  end for
  Update α, β, and δ wolf  of the wolf pack
  Apply OBL on α, β, and δ wolf  obtain opposite α_oppos, β_oppos, and δ_oppos of α, β, and δ respectively
  Include α_oppos, β_oppos, and δ_oppos in the current population and sort ascending to fitness from best to worst
  Remove worst 3 solutions
  Select α, β and δ wolf  in the updated popuation
  t = t + 1
end while
return the α
```

4 Experimental results and analysis

4.1 Benchmark functions and parameter settings

In this section to analyze the performance of the IGWO algorithm, we select 23 well known and classical benchmark functions which are presented in Tables 1, 2 and 3. These benchmark functions are denoted by $f1, f2...f23$. In these tables, unimodal ($f1$–$f7$), multimodal ($f8 - f13$) and fixed dimensional multimodal ($f14$–$f23$) benchmark functions are presented respectively. *dim* denotes the dimension of the problems, *Range* refers the range of the decision variables and f_{min} refers optimal value of the problem. These benchmark functions have been utilized by many researchers to evalute their algorithms (Mirjalili 2015a; Heidari et al. 2019; Mirjalili 2015c; Mirjalili and Lewis 2016; Bansal et al. 2018) and represent wide variety of problems. Numerical results on these problems using IGWO are obtained and compared with recent variants of GWO apart from the classical GWO and other popular swarm intelligence based state−of-the−art optimization algorithms. Recent variants of GWO are Random Walk Grey Wolf Optimizer(RW-GWO) (Gupta and Deep 2019), Modified Grey Wolf Optimizer(MGWO) (Mittal et al. 2016), Opposition based Grey Wolf Optimizer(OGWO) (Pradhan et al. 2017), and swarm intelligence based state−of-the−art optimization algorithms are the GWO (Mirjalili et al. 2014), ABC (Karaboga 2005), BBO (Simon 2008) and Covariance Matrix Adaption Evolution Strategy (CMA-ES) (Hansen 2006a) and other well established algorithms used for comparison. For all algorithms, we set the same swarm size and same maximum number of iterations for a fair comparison. The swarm size and the maximum number of iterations are set to 50 and 1000 respectively. Each algorithm is run 30 times independently and the results recorded.

4.1.1 Comparison of IGWO algorithm with classical GWO and its variants

In this subsection, the numerical results obtained by the classical GWO, RW-GWO, MGWO, OGWO and the proposed IGWO algorithms are recorded and presented in Tables 4, 5 and 6. In these tables, "Mean" denotes the average of best values (*alpha* score), "Best" refers to the *alpha* score, "Worst" means the function value at worst solution, "Median" represents the median value of *alpha* scores and "SD" denotes the standard deviation of alpha values recorded over 30 independent runs. The best results for each test function are highlighted with boldface in Tables 4, 5 and 6.

4.1.2 Comparison in terms of exploitation and exploration

In stochastic meta-heuristic algorithms, the exploitation ability and convergence rate can be determine by unimodal benchmark functions. In these functions, only one optima is present which is known as global optima. From Table 4 it is clear that IGWO algorithm obtains optimal value and close to optimal value. Hence it can be concluded that the IGWO algorithm is better than RW-GWO, M-GWO, OGWO and GWO in terms of unimodal problems ($f1-f7$).

Multimodal benchmark functions (scalable and non-scalable multimodal functions) are suitable to evaluate the exploration ability of a meta-heuristic algorithm. These benchmark functions have more than one local optima that make them hard to be tackled. The results on these problems given in Table 5 also demonstrate that the search agents in the IGWO are able to explore the promising domains of the search space. In the problems $f8, f9, f11, f14-f18, f20-f23$ the IGWO has achieved optimal value (alpha score) while

Table 3 Fixed-dimensional multimodal benchmark functions

Benchmark	Name	*dim*	Range	$f_{\min}$
Shekel's Foxholes function	$f14(x) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j+\sum_{i=1}^{2}(x_i-a_{ij})^6}\right)^{-1}$	2	$[-65, 65]$	0.998
Kowalik's function	$f15(x) = \sum_{i=1}^{11}\left[a_i - \frac{x_1(b_i^2+b_ix_2)}{b_i^2+b_ix_3+x_4}\right]^2$	4	$[-5, 5]$	0.00030
Six-hump camel-back function	$f16(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	$[-5, 5]$	− 1.0316
Branin function	$f17(x) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos x_1 + 10$	2	$[-5, 5]$	0.398
Goldstein-Price function	$f18(x) = \left[1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)\right]$ $\left[30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)\right]$	2	$[-2, 2]$	3
Hartman's family function	$f19(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{3} a_{ij}(x_j - p_{ij})^2\right)$	3	$[1, 3]$	− 3.86
Hartman's family function	$f20(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right)$	6	$[0, 1]$	−3.32
Shekel's family function	$f21(x) = -\sum_{i=1}^{5}\left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$	4	$[0, 10]$	− 10.1532
Shekel's family function	$f22(x) = -\sum_{i=1}^{7}\left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$	4	$[0, 10]$	− 10.4028
Shekel's family function	$f23(x) = -\sum_{i=1}^{10}\left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$	4	$[0.10]$	− 10.5363

the OGWO has obtained the optimal value (α score) for $f9$,$f11$,$f14$–$f18$,$f20$–$f23$ and also RW-GWO, MGWO and GWO have obtained optimal value (alpha score) for $f14$–$f18$,$f20$–$f23$. But in the problems $f8$–$f14$, $f17$, $f19$, and $f21$–$f23$, the IGWO perform better in terms of mean, best, worst, median and standard deviation and in $f16$, $f18$ and $f20$, the IGWO is better in terms of mean, best, worst, median. In $f16$ the classical GWO performed better in terms of mean, best, worst, median and standard deviation and in $f18$, OGWO is better. Thus, the overall comparison of numerical results based on mean, best, worst, median and standard deviation of the alpha values in 30 independent runs verifies that the proposed IGWO is a better optimizer than the classical GWO, RW-GWO, MGWO and OGWO. More statistical analysis of these results is reported in Sect. 4.1.5.

4.1.3 Diversity analysis

In this sub-section, the exploration and exploitation abilities of the proposed IGWO are measured. Obviously this can be measured in terms of loss of diversity during the search process. In order to check, how the swarm diversity varies with iterations in the IGWO search process. In intermediate iterations, the difference between every pair of the solutions is calculated maximum and minimum differences for a few representative functions are recorded in Table 7. The diversity curves for maximum distance and minimum distance

Table 4 Mean, best, worst , median and standard deviation (SD) of alpha values obtained in 30 runs on unimodal test problems by the variants of GWO and IGWO

Function	Algorithms	Mean	Best	Worst	Median	SD
$f1$	RW-GWO	1.36E−74	1.05E−76	1.36E−73	3.75E−75	2.83E−74
	MGWO	1.42E−99	1.20E−102	1.59E−98	2.19E−100	3.01E−99
	OGWO	1.31E−156	1.39E−165	3.59E−155	3.38E−161	6.56E−156
	GWO	8.83E−77	1.19E−79	2.00E−75	8.46E−78	3.63E−76
	IGWO	**0**	**0**	**0**	**0**	**0**
$f2$	RW-GWO	3.25E−43	1.96E−44	1.44E−42	1.70E−43	3.82E−43
	MGWO	4.76E−57	1.66E−58	2.85E−56	1.48E−57	7.45E−57
	OGWO	5.90E−93	6.84E−96	5.96E−92	8.66E−94	1.25E−92
	GWO	5.45E−45	1.16E−45	1.98E−44	4.76E−45	4.39E−45
	IGWO	**0**	**0**	**0**	**0**	**0**
$f3$	RW-GWO	3.04E−12	1.07E−15	3.08E−11	4.28E−13	7.72E−12
	MGWO	3.27E−20	4.70E−27	6.83E−19	2.30E−23	1.30E−19
	OGWO	9.39E−42	4.98E−57	2.39E−40	1.07E−46	4.37E−41
	GWO	7.54E−18	1.73E−24	1.61E−16	1.79E−19	2.93E−17
	IGWO	**0**	**0**	**0**	**0**	**0**
$f4$	RW-GWO	7.65E−14	1.15E−15	3.51E−13	5.10E−14	8.14E−14
	MGWO	6.32E−23	7.48E−25	1.26E−21	6.88E−24	2.28E−22
	OGWO	3.34E−58	2.60E−61	3.41E−57	2.52E−59	7.69E−58
	GWO	1.83E−16	6.12E−18	1.77E−15	6.76E−17	3.49E−16
	IGWO	**0**	**0**	**0**	**0**	**0**
$f5$	RW-GWO	2.59E+01	2.50E+01	2.71E+01	2.61E+01	5.05E−01
	MGWO	2.65E+01	2.53E+01	2.79E+01	2.62E+01	6.77E−01
	OGWO	2.73E+01	2.72E+01	2.87E+01	2.72E+01	3.16E−01
	GWO	2.63E+01	2.51E+01	2.79E+01	2.62E+01	7.17E−01
	IGWO	**2.38E−02**	**8.20E−03**	**6.63E−02**	**2.15E−02**	**1.35E−02**
$f6$	RW-GWO	2.84E−02	3.09E−06	2.50E−01	7.70E−06	7.20E−02
	MGWO	4.12E−01	1.62E−05	1.01E+00	4.48E−01	2.77E−01
	OGWO	4.50E−01	3.69E−06	1.00E+00	5.00E−01	2.40E−01
	GWO	2.97E−01	4.55E−06	7.53E−01	2.51E−01	2.45E−01
	IGWO	**7.35E−06**	**3.02E−06**	**1.68E−05**	**5.88E−06**	**4.11E−06**
$f7$	RW-GWO	1.01E−03	4.08E−04	1.89E−03	9.93E−04	4.07E−04
	MGWO	4.53E−04	1.02E−04	1.20E−03	3.30E−04	3.05E−04
	OGWO	1.02E−02	2.80E−04	3.53E−02	7.89E−03	8.63E−03
	GWO	6.24E−04	1.98E−04	1.17E−03	6.03E−04	2.41E−04
	IGWO	**1.15E−04**	**7.54E−06**	**3.85E−04**	**1.20E−04**	**8.38E−05**

Best results are highlighted in bold

Table 5 Mean, best, worst, median and standard deviation (SD) of alpha values obtained in 30 runs on multimodal test problems by the variants of GWO and IGWO

Function	Algorithms	Mean	Best	Worst	Median	SD
*f*8	RW-GWO	− 8.82E+03	-1.00E+04	− 7.59E+03	− 8.82E+03	5.32E+02
	MGWO	− 5.53E+03	− 8.28E+03	− 3.15E+03	− 6.04E+03	1.57E+03
	OGWO	− 7.11E+02	− 8.38E+02	− 4.19E+02	− 8.38E+02	1.83E+02
	GWO	− 6.47E+03	− 8.53E+03	− 3.76E+03	− 6.62E+03	1.02E+03
	IGWO	**-1.26E+04**	**− 1.26E+04**	**− 1.26E+04**	**− 1.26E+04**	**2.06E−01**
*f*9	RW-GWO	9.36E+00	**0**	2.17E+01	8.42E+00	5.81E+00
	MGWO	1.98E+00	**0**	4.21E+01	**0**	7.79E+00
	OGWO	**0**	**0**	**0**	**0**	**0**
	GWO	4.30E+00	**0**	1.36E+01	2.84E−14	5.06E+00
	IGWO	**0**	**0**	**0**	**0**	**0**
*f*10	RW-GWO	7.99E−15	7.99E−15	7.99E−15	7.99E−15	**0**
	MGWO	6.34E−15	4.44E−15	7.99E−15	7.99E−15	1.80E−15
	OGWO	**8.88E−16**	**8.88E−16**	**8.88E−16**	**8.88E−16**	**0**
	GWO	7.99E−15	7.99E−15	7.99E−15	7.99E−15	**0**
	IGWO	**8.88E−16**	**8.88E−16**	**8.88E−16**	**8.88E−16**	**0**
*f*11	RW-GWO	4.20E−03	**0**	3.08E−02	**0**	7.99E−03
	MGWO	2.99E−04	**0**	9.00E−03	**0**	1.60E−03
	OGWO	**0**	**0**	**0**	**0**	0
	GWO	3.16E−03	**0**	2.31E−02	**0**	6.23E−03
	IGWO	**0**	**0**	**0**	**0**	**0**
*f*12	RW-GWO	3.89E−03	5.60E−07	1.43E−02	3.04E−03	4.24E−03
	MGWO	2.03E−02	2.34E−06	4.73E−02	1.98E−02	1.03E−02
	OGWO	8.20E−02	5.89E−02	1.24E−01	7.85E−02	1.22E−02
	GWO	2.17E−02	6.26E−03	6.20E−02	1.96E−02	1.35E−02
	IGWO	**9.57E−07**	**4.68E−07**	**1.71E−06**	**9.08E−07**	**3.07E−07**
*f*13	RW-GWO	4.17E−02	7.38E−06	1.83E−01	2.78E−05	5.68E−02
	MGWO	2.92E−01	3.03E−05	6.26E−01	2.98E−01	1.90E−01
	OGWO	4.51E−01	1.11E−01	9.25E−01	4.17E−01	2.23E−01
	GWO	2.97E−01	8.51E−06	8.01E−01	2.47E−01	1.98E−01
	IGWO	**1.40E−05**	**7.05E−06**	**2.51E−05**	**1.24E−05**	**5.10E−06**

Best results are highlighted in bold

are shown in Figs. 2 and 3. In these figures, the iterations of an algorithm are shown on *x-axis* and the distance between search agents is shown on the *y-axis*. Figure 2a–f denotes maximum distance graph for $f1, f4, f8, f10, f15$ and $f22$ and the diversity curves for minimum distance are shown in Fig. 3. Figure 3a–f represents minimum distance graph for $f1, f4, f8, f10, f15$ and $f22$. The diversity curves for unimodal problems show that the whole population in the IGWO algorithm converges as the maximum and minimum distance between the wolves is close to zero. Also, the distance in the IGWO algorithm is less than that of GWO, which shows a better convergence rate of the proposed algorithm than the convergence rate of GWO. For the scalable multimodal problems and fixed dimensional multimodal problems, the maximum diversity curves show that the proposed algorithm allows more exploration than the GWO while the minimum diversity curves show that the wolves in the proposed algorithm start to converge better than GWO.

4.1.4 Convergence analysis

To investigate the convergence speed of the proposed algorithm IGWO is compared with GWO and other GWO variants like RW-GWO, MGWO, and OGWO under 6 selected classical benchmark functions as shown in Fig. 4. The selected functions for convergence curves are $f1, f4, f8, f10$, $f15$ and $f22$. In these functions, the first 2 are unimodal, and the remaining are multimodal and fixed dimensional multimodal functions. The convergence curves are plotted between iterations and the best value (alpha score) in the

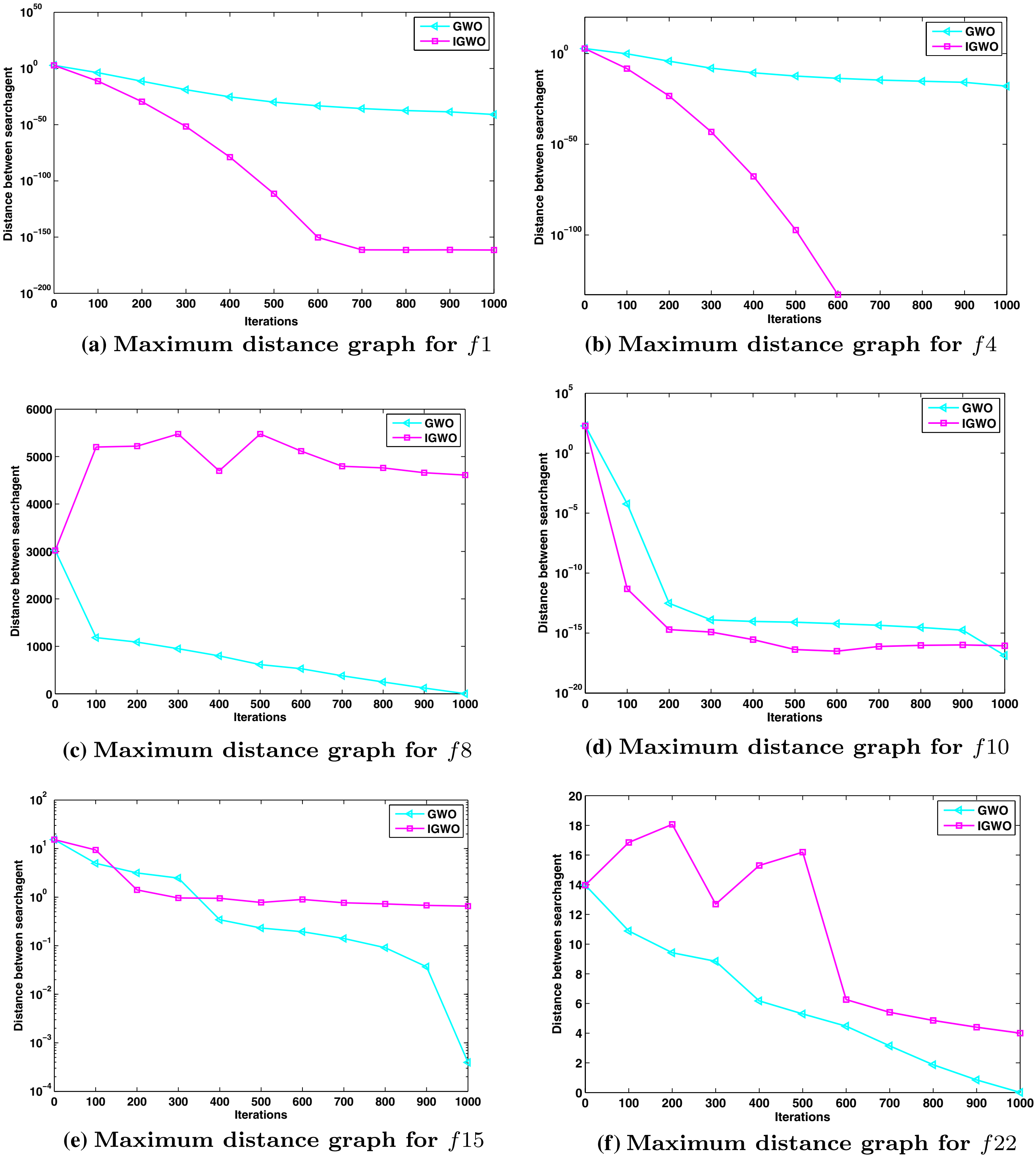
Table 6 Mean, best, worst, median and standard deviation (SD) of alpha values obtained in 30 runs on fixed dimensional test problems by the variants of GWO and IGWO

Function	Algorithms	Mean	Best	Worst	Median	SD
*f*14	RW-GWO	**9.98E−01**	**9.98E−01**	**9.98E−01**	**9.98E−01**	8.60E−12
	MGWO	3.67E+00	**9.98E−01**	1.08E+01	**9.98E−01**	4.05E+00
	OGWO	5.12E+00	2.98E+00	1.27E+01	2.98E+00	3.62E+00
	GWO	3.80E+00	**9.98E−01**	1.27E+01	**9.98E−01**	4.45E+00
	IGWO	**9.98E−01**	**9.98E−01**	**9.98E−01**	**9.98E−01**	**3.94E−12**
*f*15	RW-GWO	1.03E−03	**3.07E−04**	2.04E−02	**3.07E−04**	3.66E−03
	MGWO	3.10E−03	**3.07E−04**	2.04E−02	3.08E−04	6.90E−03
	OGWO	**3.86E−04**	**3.07E−04**	**1.22E−03**	3.08E−04	**1.72E−04**
	GWO	3.04E−03	**3.07E−04**	2.04E−02	**3.07E−04**	6.91E−03
	IGWO	1.10E−03	**3.07E−04**	2.04E−02	**3.07E−04**	3.70E−03
*f*16	RW-GWO	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	1.23E−09
	MGWO	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	5.66E−09
	OGWO	− 7.19E−01	− 1.03E+00	− 1.55E−09	− 1.01E+00	4.18E−01
	GWO	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	**1.88E−09**
	IGWO	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	3.53E−05
*f*17	RW-GWO	**3.98E−01**	**3.98E−01**	**3.98E−01**	**3.98E−01**	5.08E−08
	MGWO	**3.98E−01**	**3.98E−01**	**3.98E−01**	**3.98E−01**	1.67E−07
	OGWO	**3.98E−01**	**3.98E−01**	**3.98E−01**	**3.98E−01**	1.29E−07
	GWO	**3.98E−01**	**3.98E−01**	**3.98E−01**	**3.98E−01**	3.40E−08
	IGWO	**3.98E−01**	**3.98E−01**	**3.98E−01**	**3.98E−01**	**1.94E−08**
*f*18	RW-GWO	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	9.16E−07
	MGWO	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	2.62E−06
	OGWO	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	**1.37E−07**
	GWO	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	3.37E−06
	IGWO	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	3.02E−06
*f*19	RW-GWO	**− 3.00E−01**	**− 3.00E−01**	**− 3.00E−01**	**− 3.00E−01**	**2.26E−16**
	MGWO	**− 3.01E−01**	**− 3.01E−01**	**− 3.01E−01**	**− 3.01E−01**	**2.26E−16**
	OGWO	**− 3.00E−01**	**− 3.00E−01**	**− 3.00E−01**	**− 3.00E−01**	**2.26E−16**
	GWO	**− 3.00E−01**	**− 3.00E−01**	**− 3.00E−01**	**− 3.00E−01**	**2.26E−16**
	IGWO	**− 3.01E−01**	**− 3.01E−01**	**− 3.01E−01**	**− 3.01E−01**	**2.26E−16**
*f*20	RW-GWO	− 3.25E+00	**− 3.32E+00**	− 3.20E+00	− 3.20E+00	**5.91E−02**
	MGWO	− 3.22E+00	**− 3.32E+00**	− 3.13E+00	− 3.20E+00	6.95E−02
	OGWO	− 3.16E+00	**− 3.32E+00**	− 3.09E+00	− 3.14E+00	9.27E−02
	GWO	− 3.23E+00	**− 3.32E+00**	− 3.13E+00	− 3.20E+00	6.78E−02
	IGWO	**− 3.26E+00**	**− 3.32E+00**	**− 3.14E+00**	**− 3.26E+00**	7.05E−02
*f*21	RW-GWO	-9.31E+00	**− 1.02E+01**	− 5.06E+00	**− 1.02E+01**	1.92E+00
	MGWO	-9.81E+00	**− 1.02E+01**	− 5.06E+00	**− 1.02E+01**	1.29E+00
	OGWO	-9.81E+00	**− 1.02E+01**	− 5.06E+00	**− 1.02E+01**	1.29E+00
	GWO	-9.98E+00	**− 1.02E+01**	− 5.10E+00	**− 1.02E+01**	9.22E−01
	IGWO	**− 1.02E+01**	**− 1.02E+01**	**− 1.02E+01**	**− 1.02E+01**	**3.67E−05**
*f*22	RW-GWO	− 1.02E+01	**− 1.04E+01**	− 5.09E+00	**− 1.04E+01**	9.70E−01
	MGWO	− 1.02E+01	**− 1.04E+01**	− 5.09E+00	**− 1.04E+01**	9.70E−01
	OGWO	− 1.00E+01	**− 1.04E+01**	− 5.09E+00	**− 1.04E+01**	1.35E+00
	GWO	− 1.02E+01	**− 1.04E+01**	− 5.09E+00	**− 1.04E+01**	9.70E−01
	IGWO	**− 1.04E+01**	**− 1.04E+01**	**− 1.04E+01**	**− 1.04E+01**	**5.13E−05**
*f*23	RW-GWO	**− 1.05E+01**	**− 1.05E+01**	**− 1.05E+01**	**− 1.05E+01**	6.82E−05
	MGWO	**− 1.05E+01**	**− 1.05E+01**	**− 1.05E+01**	**− 1.05E+01**	1.79E−04
	OGWO	− 1.05E+01	**− 1.05E+01**	**− 1.05E+01**	**− 1.05E+01**	6.01E−05
	GWO	− 1.04E+01	**− 1.05E+01**	− 5.13E+00	**− 1.05E+01**	9.87E−01
	IGWO	**− 1.05E+01**	**− 1.05E+01**	**− 1.05E+01**	**− 1.05E+01**	**5.04E−05**

Best results are highlighted in bold

Table 7 Iteration wise maximum and minimum difference between solutions

	f1				f4			
	GWO		IGWO		GWO		IGWO	
Iteration	Min	Max	Min	Max	Min	Max	Min	Max
0	2.68E+02	5.71E+02	2.68E+02	5.71E+02	2.93E+02	5.82E+02	2.93E+02	5.82E+02
100	4.34E−05	1.73E−04	1.38E−13	5.85E−12	4.17E−02	1.66E−01	8.18E−11	2.98E−09
200	4.69E−13	1.78E−12	7.92E−32	4.00E−30	1.62E−06	6.31E−06	4.72E−26	3.24E−24
300	3.89E−21	1.14E−20	6.28E−54	6.20E−53	7.04E−10	1.93E−09	4.08E−46	7.20E−44
400	1.19E−27	3.44E−27	5.98E−82	1.52E−80	2.36E−12	7.36E−12	1.46E−68	7.92E−68
500	4.02E−32	1.17E−31	2.06E−114	7.16E−113	4.33E−14	1.27E−13	1.16E−99	4.23E−98
600	1.93E−35	5.35E−35	2.12E−152	1.69E−151	1.31E−15	3.43E−15	4.96E−135	2.88E−134
700	8.63E−38	2.13E−37	0	3.14E−162	1.95E−16	4.60E−16	0	0
800	1.67E−39	3.99E−39	0	3.14E−162	4.10E−17	9.61E−17	0	0
900	9.68E−41	2.24E−40	0	3.85E−162	1.06E−17	2.45E−17	0	0
1000	5.26E−43	1.16E−42	0	3.14E−162	8.09E−20	1.98E−19	0	0
	f8				f10			
	GWO		IGWO		GWO		IGWO	
Iteration	Min	Max	Min	Max	Min	Max	Min	Max
0	1.51E+03	2.99E+03	1.51E+03	2.99E+03	9.50E+01	1.85E+02	9.50E+01	1.85E+02
100	4.63E+02	1.34E+03	7.53E+01	5.18E+03	2.52E−05	1.05E−04	4.36E−13	9.14E−12
200	3.36E+02	1.12E+03	2.01E+01	5.48E+03	1.79E−13	5.72E−13	1.43E−16	5.99E−15
300	3.12E+02	8.91E+02	0	5.26E+03	2.89E−15	1.09E−14	2.23E−17	1.09E−15
400	2.63E+02	8.07E+02	5.65E+01	5.01E+03	2.86E−15	7.67E−15	1.18E−17	7.33E−16
500	1.93E+02	5.51E+02	0	5.14E+03	2.31E−15	7.16E−15	5.88E−18	4.89E−16
600	1.59E+02	4.98E+02	2.78E+01	4.77E+03	1.92E−15	4.90E−15	2.45E−18	7.43E−16
700	7.77E+01	3.89E+02	1.69E+02	4.73E+03	1.32E−15	4.32E−15	1.75E−18	1.68E−15
800	7.27E+01	2.44E+02	1.55E+02	4.69E+03	8.93E−16	2.65E−15	1.25E−18	3.40E−15
900	4.00E+01	1.18E+02	3.68E+01	4.67E+03	3.72E−16	1.18E−15	1.80E−19	1.32E−15
1000	3.63E−01	1.09E+00	7.59E−01	4.61E+03	4.31E−18	1.24E−17	4.08E−21	1.07E−15
	f15				f22			
	GWO		IGWO		GWO		IGWO	
Iteration	Min	Max	Min	Max	Min	Max	Min	Max
0	6.76E−01	1.48E+01	6.76E−01	1.48E+01	6.69E−01	1.43E+01	6.69E−01	1.43E+01
100	9.46E−02	5.37E+00	1.10E−01	1.16E+01	6.22E−01	1.09E+01	0	1.82E+01
200	3.28E−02	2.57E+00	6.05E−02	5.52E+00	5.98E−01	9.03E+00	0	1.67E+01
300	3.57E−02	1.33E+00	2.53E−02	1.16E+00	4.50E−01	7.46E+00	0	1.51E+01
400	1.85E−02	2.97E−01	1.48E−02	1.03E+00	2.68E−01	5.61E+00	3.80E−02	7.30E+00
500	1.63E−02	2.04E−01	1.10E−02	7.76E−01	3.73E−01	6.09E+00	2.83E−01	6.39E+00
600	8.70E−03	1.83E−01	5.10E−03	7.67E−01	1.83E−01	4.20E+00	2.74E−01	5.67E+00
700	4.70E−03	1.19E−01	5.40E−03	6.73E−01	1.74E−01	2.80E+00	1.69E−01	5.05E+00
800	5.00E−03	9.51E−02	3.90E−03	6.89E−01	1.42E−01	2.28E+00	1.33E−01	4.74E+00
900	2.80E−03	3.59E−02	8.72E−04	6.57E−01	6.73E−02	1.10E+00	6.32E−02	4.49E+00
1000	2.60E−05	3.37E−04	2.46E−05	6.57E−01	4.40E−04	8.40E−03	6.68E−04	4.00E+00



(a) Maximum distance graph for $f1$

(b) Maximum distance graph for $f4$

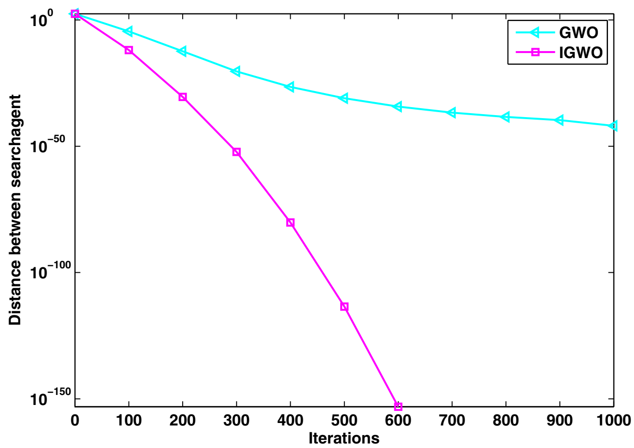
(c) Maximum distance graph for $f8$

(d) Maximum distance graph for $f10$

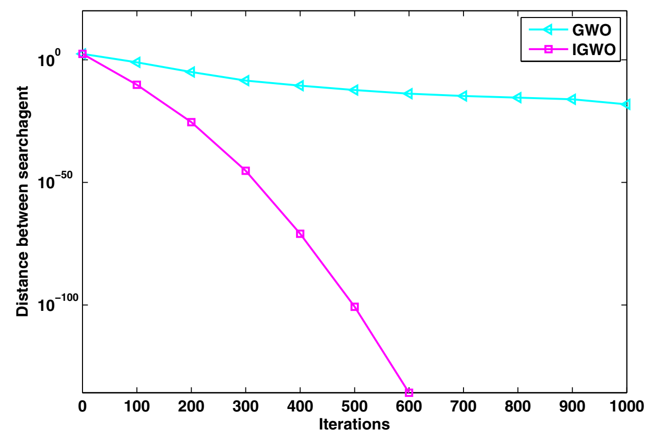
(e) Maximum distance graph for $f15$

(f) Maximum distance graph for $f22$

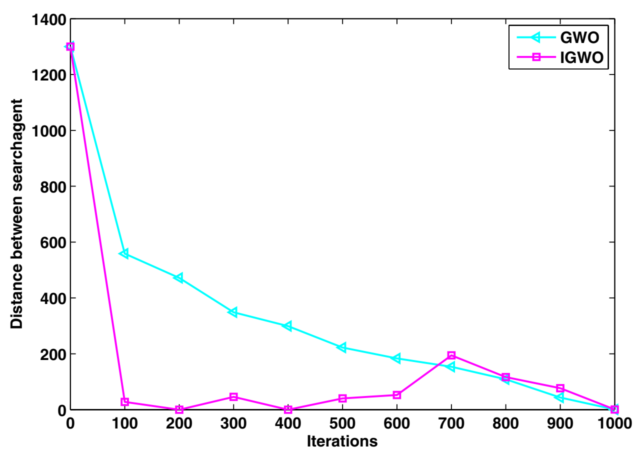
Fig. 2 Diversity curves for maximum distance



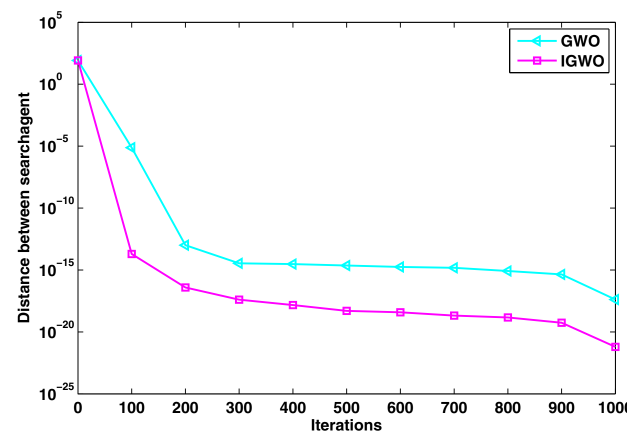
(a) Minimum distance graph for $f1$



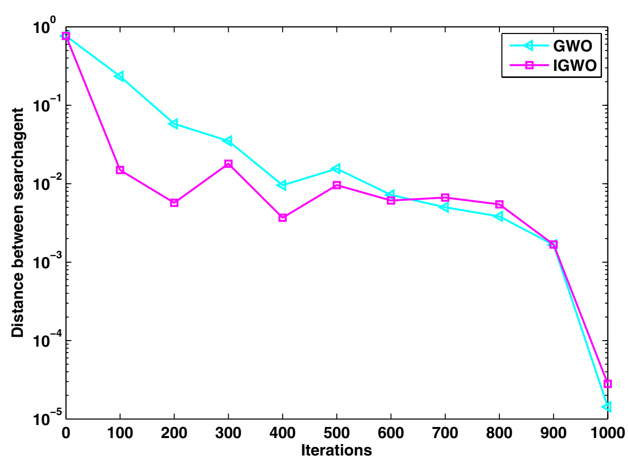
(b) Minimum distance graph for $f4$



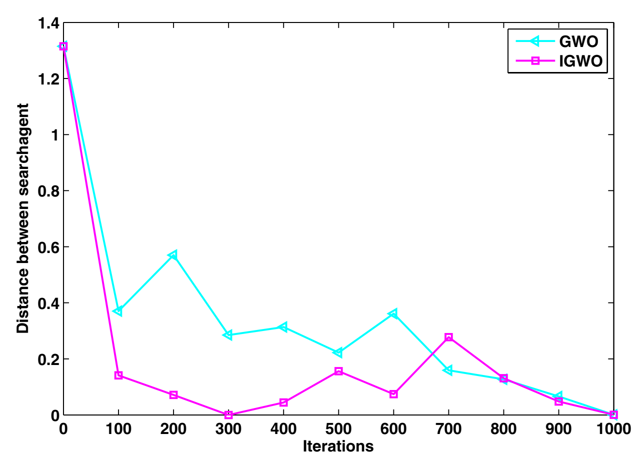
(c) Minimum distance graph for $f8$



(d) Minimum distance graph for $f10$

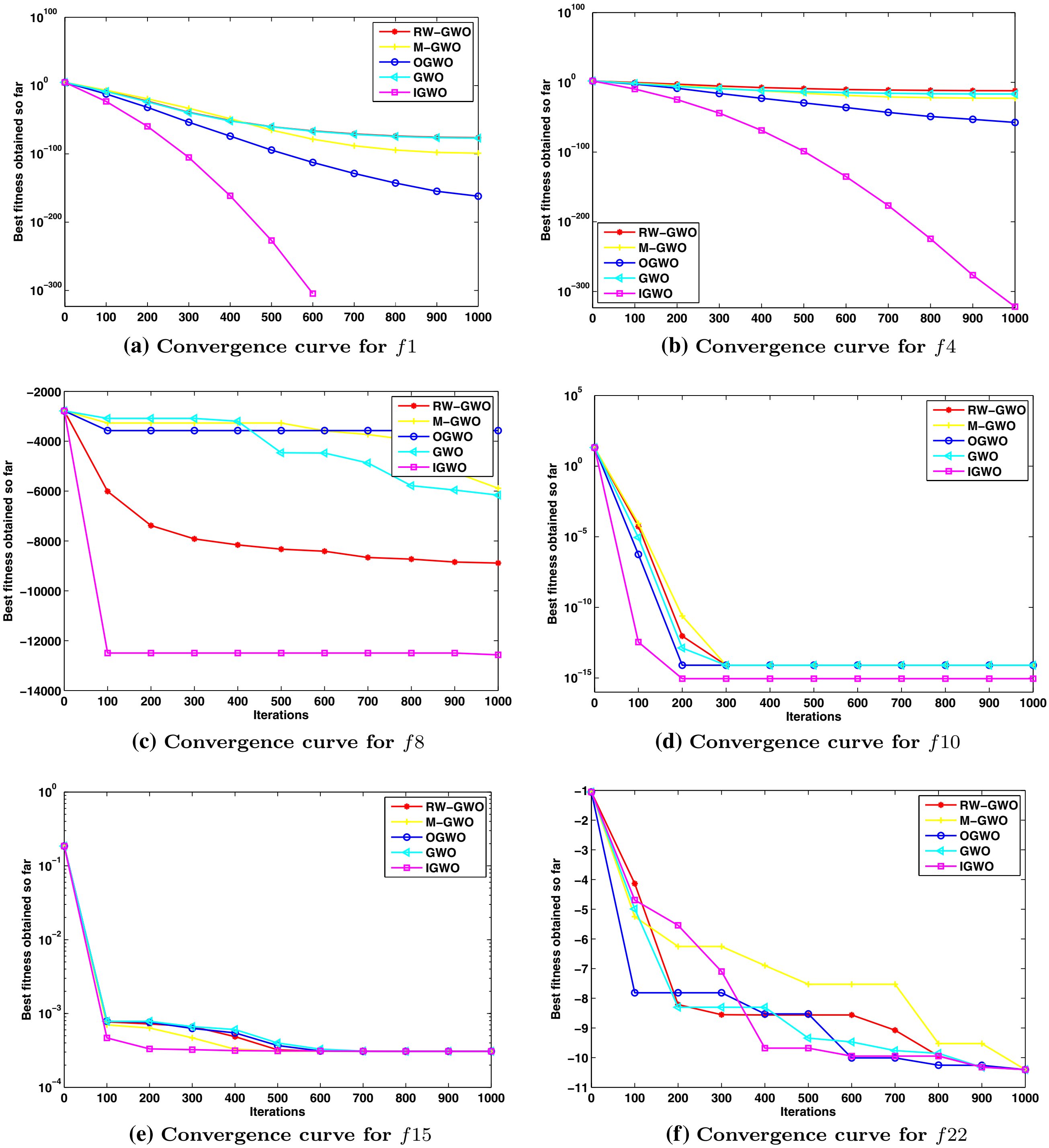


(e) Minimum distance graph for $f15$



(f) Minimum distance graph for $f22$

Fig. 3 Diversity curves for minimum distance



(a) Convergence curve for $f1$

(b) Convergence curve for $f4$

(c) Convergence curve for $f8$

(d) Convergence curve for $f10$

(e) Convergence curve for $f15$

(f) Convergence curve for $f22$

Fig. 4 Convergence curves

Table 8 p-values obtained by the Friedman test with IGWO

TP	p value	TP	p-value	TP	p value
$f1$	2.20E−16	$f9$	2.20E−16	$f17$	1
$f2$	2.20E−16	$f10$	2.20E−16	$f18$	1
$f3$	2.20E−16	$f11$	2.20E−16	$f19$	1
$f4$	2.20E−16	$f12$	2.20E−16	$f20$	1.30E−07
$f5$	2.20E−16	$f13$	2.20E−16	$f21$	2.19E−10
$f6$	2.20E−16	$f14$	2.46E−14	$f22$	8.19E−12
$f7$	2.20E−16	$f15$	9.83E−05	$f23$	1.88E−09
$f8$	2.20E−16	$f16$	2.32E−13		

Table 9 p values obtained by the Bonferroni test with IGWO

TP	RW-GWO	MGWO	OGWO	GWO
$f1$	2.00E−16	2.00E−16	2.00E−16	2.00E−16
$f2$	2.00E−16	2.00E−16	2.00E−16	2.00E−16
$f3$	2.00E−16	2.00E−16	2.00E−16	2.00E−16
$f4$	2.00E−16	2.00E−16	2.00E−16	2.00E−16
$f5$	2.00E−16	2.00E−16	2.00E−16	2.00E−16
$f6$	4.70E−03	2.20E−16	2.20E−16	2.20E−16
$f7$	2.00E−16	2.00E−16	2.00E−16	2.00E−16
$f8$	2.00E−16	2.00E−16	2.00E−16	2.00E−16
$f9$	2.00E−16	7.80E−03	1	2.20E−16
$f10$	2.20E−16	2.20E−16	1	2.20E−16
$f11$	6.30E−16	1	1	1.30E−04
$f12$	7.20E−10	2.20E−16	2.20E−16	2.20E−16
$f13$	4.30E−05	2.20E−16	2.20E−16	2.20E−16
$f14$	1	2.20E−16	2.20E−16	2.20E−16
$f15$	7.84E−02	4.30E−04	2.30E−14	1
$f16$	4.40E−06	4.40E−06	2.20E−16	4.40E−06
$f17$	1	1	1	1
$f18$	1	1	1	1
$f19$	1	1	1	1
$f20$	1	8.20E−03	2.20E−16	2.80E−02
$f21$	2.20E−16	2.20E−16	2.20E−16	1.50E−13
$f22$	4.00E−14	2.20E−16	7.20E−10	4.00E−12
$f23$	2.90E−05	2.20E−16	2.70E−09	8.50E−07

intermediate iterations. In the curves, the iterations of an algorithm are shown on the horizontal axis and the objective function value is shown on the vertical axis. Figure 4a–f are convergence curve for $f1$, $f4$, $f8$, $f10$, $f15$ and $f22$. From the convergence curves, it can be easily observed that in terms of convergence rate, the IGWO algorithm perform well as compared to RW-GWO, MGWO, OGWO and GWO algorithms.

4.1.5 Statistical analysis

In this sub-section, the Friedman test is used to compare the performance of all considered algorithms simultaneously which is a two-stage method (the statistical Friedman test and then a post-hoc test). A two-stage method is used to check whether the results obtained by the considered algorithms are significantly different from each other or not. This non-parametric statistical test is performed pairwise at 1% level of significance with the null hypothesis, 'There is no significant difference between the results obtained by the considered pair'. After using the Friedman test, we need post-hoc statistical analysis. Some post-hoc statistical test is: Bonferroni, Holm- Bonferroni, Hochberg, Hommel, Benjamin-Hochberg (BH), and Fdr. In this paper, for pairwise comparisons, we also reported the adjusted p values achieved by the Bonferroni procedure. The Friedman test and the Bonferroni procedure are implemented in R (pohlert-2014pairwise; ripley2001r). The p values obtained by Friedman test and Bonferroni procedure are presented in Table 8 and Table 9 respectively. In these tables, 1 denotes that the IGWO algorithm is significantly same as the GWO, RW-GWO, MGWO, and OGWO. From the statistical conclusions, it can be seen that the proposed IGWO is significantly outperforming the classical GWO, RW-GWO, MGWO, and OGWO.

4.2 Comparison with well-known metaheuristic algorithms

In this section, the performance of the proposed IGWO algorithm is compared with some other popular state−of-the−art meta-heuristic algorithms. Same benchmark set (Tables 1, 2 and 3) is used. ABC (Karaboga 2005), BBO (Simon 2008) and CMA-ES (Hansen 2006b) are compared with IGWO. The parameter settings of algorithms are the same as used in their original papers. For a fair comparison, the same population size and same the maximum number of iterations are kept same. 30 runs are considered, The comparison results between IGWO and metaheuristic algorithms are shown in the Tables 10, 11 and 12. In these tables, mean, best value, worst value, median and standard devitation are presented.

In Table 10, IGWO is able to provide better results for $f1$–$f5$, and $f7$ except for $f6$. In $f6$, CMA-ES has better results compared to ABC, BBO, and IGWO. In Table 11

Table 10 Mean, best, worst, median and standard deviation (SD) of alpha values obtained in 30 runs on unimodal test problems by the considered metaheuristic algorithms and IGWO

Function	Algorithms	Mean	Best	Worst	Median	SD
f1	ABC	1.69E−11	3.53E−12	9.03E−11	1.22E−11	1.72E−11
	BBO	5.83E+00	2.55E+00	1.24E+01	5.31E+00	2.46E+00
	CMA-ES	2.36E−54	2.54E−56	3.42E−53	6.66E−55	6.38E−54
	IGWO	**0**	**0**	**0**	**0**	**0**
f2	ABC	7.87E−07	2.04E−07	1.47E−06	7.85E−07	2.80E−07
	BBO	8.82E−01	6.40E−01	1.29E+00	8.67E−01	1.63E−01
	CMA-ES	2.06E−25	2.98E−26	7.85E−25	1.32E−25	1.85E−25
	IGWO	**0**	**0**	**0**	**0**	**0**
f3	ABC	1.24E+04	8.63E+03	1.56E+04	1.23E+04	2.03E+03
	BBO	9.02E+03	3.73E+03	1.37E+04	8.93E+03	2.11E+03
	CMA-ES	6.52E−44	5.45E−46	3.69E−43	2.85E−44	8.81E−44
	IGWO	**0**	**0**	**0**	**0**	**0**
f4	ABC	2.11E+01	1.40E+01	2.80E+01	2.12E+01	3.77E+00
	BBO	6.52E+00	4.71E+00	8.31E+00	6.46E+00	1.04E+00
	CMA-ES	2.44E−20	1.99E−21	2.10E−19	1.31E−20	3.88E−20
	IGWO	**0**	**0**	**0**	**0**	**0**
f5	ABC	1.56E+00	5.83E−02	6.28E+00	7.65E−01	1.74E+00
	BBO	4.11E+02	1.62E+02	2.23E+03	3.22E+02	3.90E+02
	CMA-ES	5.52E+00	3.72E+00	7.89E+00	5.30E+00	8.58E−01
	IGWO	**2.38E−02**	**8.20E−03**	**6.63E−02**	**2.15E−02**	**1.35E−02**
f6	ABC	1.26E−11	1.54E−12	4.14E−11	1.04E−11	9.59E−12
	BBO	6.18E+00	1.38E+00	1.07E+01	6.09E+00	2.50E+00
	CMA-ES	**1.18E−30**	**4.53E−31**	**2.50E−30**	**1.18E−30**	**4.34E−31**
	IGWO	7.35E−06	3.02E−06	1.68E−05	5.88E−06	4.11E−06
f7	ABC	1.11E−01	6.53E−02	1.69E−01	1.13E−01	2.62E−02
	BBO	2.27E−02	8.40E−03	4.28E−02	2.21E−02	8.10E−03
	CMA-ES	5.31E−02	1.75E−02	1.12E−01	5.05E−02	2.32E−02
	IGWO	**1.15E−04**	**7.54E−06**	**3.85E−04**	**1.20E−04**	**8.38E−05**

Best results are highlighted in bold

for $f8$–$f10$, the proposed IGWO algorithm is better in terms of mean, best, worst, median and standard deviation while in the problems $f12$ and $f13$, CMA-ES found better as compare to the IGWO, ABC, and BBO. For $f11$, the IGWO and the CMA-ES perform better than ABC and BBO algorithm. For functions $f14$–$f23$, the IGWO outperforms for $f19$ and $f23$. For $f14$, $f16$, $f17$, $f18$, $f21$ and $f22$, the IGWO is the second best compared to the ABC, BBO and CMA-ES algorithm. For $f14$, $f17$, $f19$, $f21$, and $f22$, ABC perform better than BBO, CMA-ES and IGWO. For $f15$, $f16$ and $f18$, the CMA-ES provides better results than ABC, BBO and IGWO. Furthermore, the convergence speed comparison of proposed IGWO and other metaheuristic algorithms can be seen through convergence curves. The convergence curves are plotted in Fig. 5 by considering the best value on the vertical axis and iterations on horizontal axis. Figure 5a–f are convergence curves for $f1$, $f4$, $f8$, $f10$, $f15$ and $f22$. It can be observed from the convergence curves that IGWO has the fastest convergence rate as compared to ABC, BBO,

Table 11 Mean, best, worst, median and standard deviation (SD) of alpha values obtained in 30 runs on multimodal test problems by the considered metaheuristic algorithms and IGWO

Function	Algorithms	Mean	Best	Worst	Median	SD
f8	ABC	− 1.22E+04	− 1.26E+04	− 1.19E+04	− 1.22E+04	1.33E+02
	BBO	**− 1.26E+04**	**− 1.26E+04**	− 1.25E+04	**− 1.26E+04**	5.22E+00
	CMA-ES	− 1.18E+02	− 1.18E+02	− 1.18E+02	− 1.18E+02	**4.66E−14**
	IGWO	**− 1.26E+04**	**− 1.26E+04**	**− 1.26E+04**	**− 1.26E+04**	2.06E−01
f9	ABC	3.43E−01	2.64E−08	1.99E+00	1.16E−04	5.57E−01
	BBO	2.51E+00	1.18E+00	4.76E+00	2.55E+00	9.01E−01
	CMA-ES	1.48E+01	7.96E+00	1.99E+01	1.39E+01	2.86E+00
	IGWO	**0**	**0**	**0**	**0**	**0**
f10	ABC	1.53E−05	2.05E−06	6.26E−05	1.08E−05	1.27E−05
	BBO	1.25E+00	8.49E−01	2.10E+00	1.18E+00	2.92E−01
	CMA-ES	6.10E−15	4.44E−15	7.99E−15	4.44E−15	1.80E−15
	IGWO	**8.88E−16**	**8.88E−16**	**8.88E−16**	**8.88E−16**	**0**
f11	ABC	1.50E−03	6.17E−11	1.48E−02	2.18E−08	3.68E−03
	BBO	1.06E+00	1.00E+00	1.14E+00	1.05E+00	3.04E−02
	CMA-ES	**0**	**0**	**0**	**0**	**0**
	IGWO	**0**	**0**	**0**	**0**	**0**
f12	ABC	3.90E−12	3.67E−14	8.84E−11	5.45E−13	1.60E−11
	BBO	4.49E−02	6.70E−03	1.41E−01	2.30E−02	4.23E−02
	CMA-ES	**5.51E−31**	**2.63E−31**	**9.53E−31**	**5.57E−31**	**1.63E−31**
	IGWO	9.57E−07	4.68E−07	1.71E−06	9.08E−07	3.07E−07
f13	ABC	9.14E−12	6.62E−13	3.82E−11	6.25E−12	8.52E−12
	BBO	2.75E−01	1.29E−01	5.50E−01	2.56E−01	9.34E−02
	CMA-ES	**5.12E−30**	**1.20E−30**	**1.22E−29**	**4.07E−30**	**2.80E−30**
	IGWO	1.40E−05	7.05E−06	2.51E−05	1.24E−05	5.10E−06

Best results are highlighted in bold

and CMA-ES. Additionally, to confirm that the better results which are obtained through the proposed IGWO are not just by chance, a non-parametric Friedman test is used. The statistical conclusions which are drawn by applying the Friedman test and Bonferroni procedure between metaheuristic algorithms and proposed IGWO are presented in Tables 13 and 14. From the statistical analyses, it can be concluded that the proposed IGWO significantly outperforms algorithms ABC, BBO and CMA-ES.

5 Conclusion

In the present paper, an improved version of the classical GWO has been proposed which is based on inclusion of two strategies. First is the explorative equation and second is the Opposition-Based Learning. The explorative equation has helped to enhanced the exploration capability of GWO. The OBL has helped to prevent the GWO stagnation and increased the convergence speed. The proposed IGWO has been evaluated on 23 well-known benchmark problems. The obtained results are compared with other latest variants of GWO and other popular meta-heuristics. Statistical analysis has been carried out and it is found that IGWO is a better optimizer which has better exploration capability while maintaining the high convergence speed. In the future work, the proposed IGWO can be analyzed on other benchmark test problems and used to solve several real-life applications problems.

Table 12 Mean, best, worst, median and standard deviation (SD) of alpha values obtained in 30 runs on fixed dimension test problems by the considered metaheuristic algorithms and IGWO

Function	Algorithms	Mean	Best	Worst	Median	STD
f14	ABC	**9.98E−01**	**9.98E−01**	**9.98E−01**	**9.98E−01**	**2.51E−16**
	BBO	1.00E+00	**9.98E−01**	1.02E+00	**9.98E−01**	4.70E−03
	CMA-ES	1.27E+01	1.27E+01	1.27E+01	1.27E+01	1.51E−13
	IGWO	**9.98E−01**	**9.98E−01**	**9.98E−01**	**9.98E−01**	3.94E−12
f15	ABC	6.46E−04	3.61E−04	8.46E−04	6.58E−04	1.27E−04
	BBO	5.70E−03	7.83E−04	2.20E−02	3.10E−03	7.30E−03
	CMA-ES	**3.29E−04**	**3.07E−04**	**9.62E−04**	**3.07E−04**	**1.20E−04**
	IGWO	1.10E−03	**3.07E−04**	2.04E−02	**3.07E−04**	3.70E−03
f16	ABC	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	**4.97E−16**
	BBO	**− 1.03E+00**	**− 1.03E+00**	− 1.02E+00	**− 1.03E+00**	2.70E−03
	CMA-ES	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	**4.97E−16**
	IGWO	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	**− 1.03E+00**	3.53E−05
f17	ABC	**3.98E−01**	**3.98E−01**	**3.98E−01**	**3.98E−01**	**0**
	BBO	3.99E−01	**3.98E−01**	4.07E−01	3.99E−01	2.10E−03
	CMA-ES	**3.98E−01**	**3.98E−01**	**3.98E−01**	**3.98E−01**	5.63E−45
	IGWO	**3.98E−01**	**3.98E−01**	**3.98E−01**	**3.98E−01**	1.94E−08
f18	ABC	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	5.77E−04
	BBO	6.65E+00	3.00E+00	3.06E+01	3.02E+00	9.40E+00
	CMA-ES	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	**7.37E−15**
	IGWO	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	3.02E−06
f19	ABC	**− 3.00E−01**	**− 3.00E−01**	**− 3.00E−01**	**− 3.00E−01**	**2.26E−16**
	BBO	− 2.84E−01	− 2.96E−01	− 2.60E−01	− 2.88E−01	9.60E−03
	CMA-ES	− 1.86E+00	− 3.35E−01	− 3.86E+00	− 3.35E−01	1.78E+00
	IGWO	**− 3.01E−01**	**− 3.01E−01**	**− 3.01E−01**	**− 3.01E−01**	**2.26E−16**
f20	ABC	**− 3.32E+00**	**− 3.32E+00**	**− 3.32E+00**	**− 3.32E+00**	**1.73E−15**
	BBO	− 3.28E+00	− 3.32E+00	− 3.20E+00	− 3.32E+00	5.83E−02
	CMA-ES	− 1.00E+00	− 3.32E+00	− 7.06E−02	− 1.66E−01	1.42E+00
	IGWO	− 3.26E+00	− 3.32E+00	− 3.14E+00	− 3.26E+00	7.05E−02
f21	ABC	**− 1.02E+01**	**− 1.02E+01**	**− 1.02E+01**	**− 1.02E+01**	**2.06E−05**
	BBO	− 5.36E+00	− 1.01E+01	− 2.62E+00	− 3.86E+00	3.27E+00
	CMA-ES	− 5.06E+00	− 5.06E+00	− 5.06E+00	− 5.06E+00	1.26E−15
	IGWO	**− 1.02E+01**	**− 1.02E+01**	**− 1.02E+01**	**− 1.02E+01**	3.67E−05
f22	ABC	**− 1.04E+01**	**− 1.04E+01**	**− 1.04E+01**	**− 1.04E+01**	**2.19E−09**
	BBO	− 7.32E+00	− 1.04E+01	− 2.74E+00	− 1.03E+01	3.54E+00
	CMA-ES	− 5.09E+00	− 5.09E+00	− 5.09E+00	− 5.09E+00	4.01E−15
	IGWO	**− 1.04E+01**	**− 1.04E+01**	**− 1.04E+01**	**− 1.04E+01**	1.74E−08
f23	ABC	**− 1.05E+01**	**− 1.05E+01**	**− 1.05E+01**	**− 1.05E+01**	3.01E−04
	BBO	− 5.50E+00	**− 1.05E+01**	− 1.86E+00	− 3.83E+00	3.61E+00
	CMA-ES	− 5.13E+00	− 5.13E+00	− 5.13E+00	− 5.13E+00	**4.47E−15**
	IGWO	**− 1.05E+01**	**− 1.05E+01**	**− 1.05E+01**	**− 1.05E+01**	5.04E−05

Best results are highlighted in bold

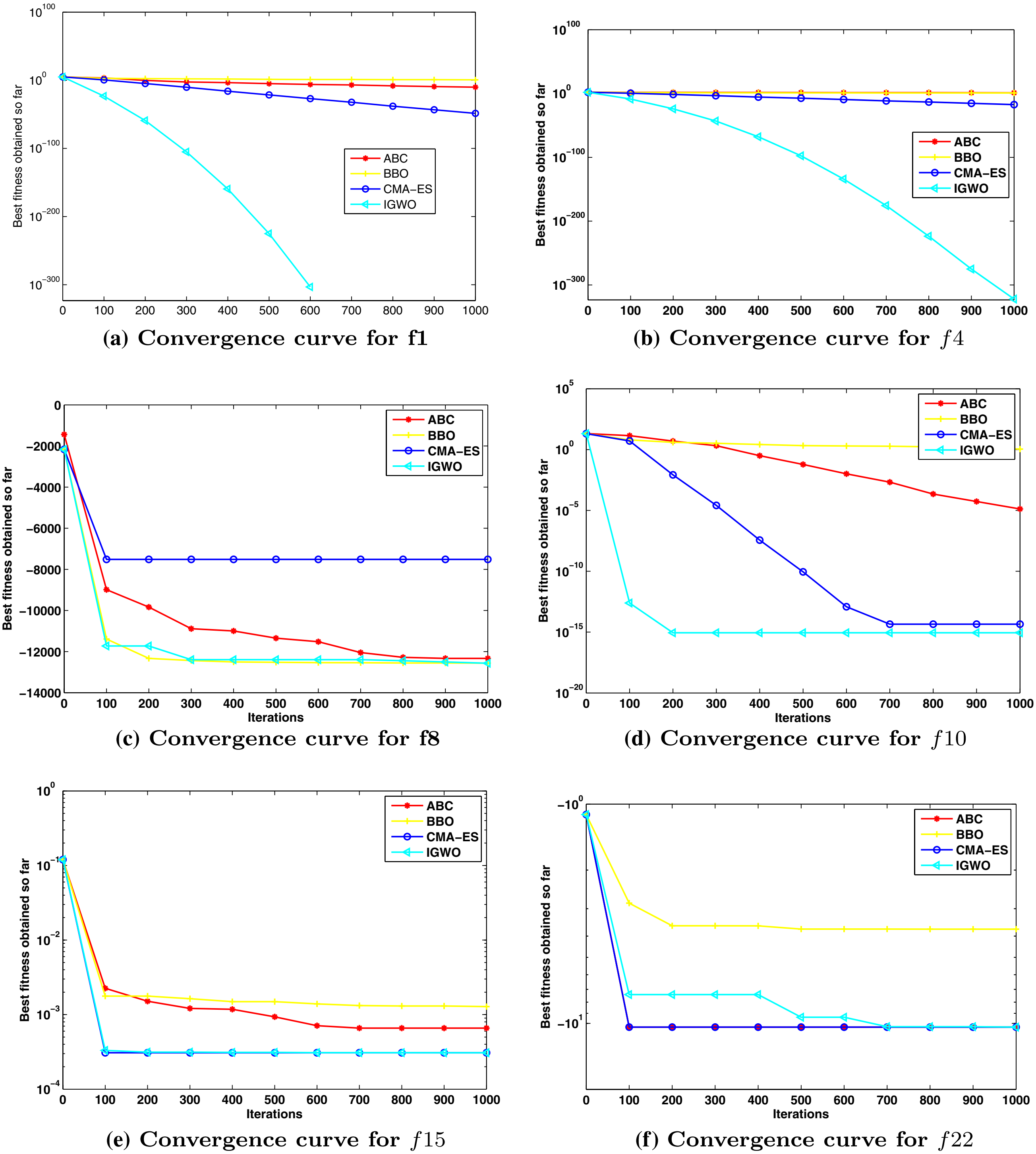


Fig. 5 Convergence curves

Table 13 p-values obtained by the Friedman test with IGWO

TP	p-value	TP	p-value	TP	p-value
*f*1	2.20E−16	*f*9	2.20E−16	*f*17	4.01E−09
*f*2	2.20E−16	*f*10	2.20E−16	*f*18	2.20E−16
*f*3	2.20E−16	*f*11	2.20E−16	*f*19	2.20E−16
*f*4	2.20E−16	*f*12	2.20E−16	*f*20	1.06E−08
*f*5	2.20E−16	*f*13	2.20E−16	*f*21	2.20E−16
*f*6	2.20E−16	*f*14	2.20E−16	*f*22	1.87E−14
*f*7	2.20E−16	*f*15	3.29E−16	*f*23	3.07E−14
*f*8	2.20E−16	*f*16	0.1116		

Table 14 p-values obtained by the Bonferroni test with IGWO

TP	ABC	BBO	CMA-ES	
*f*1	2.00E−16	2.00E−16	2.00E−16	
*f*2	2.00E−16	2.00E−16	2.00E−16	
*f*3	2.00E−16	2.00E−16	2.00E−16	
*f*4	2.00E−16	2.00E−16	2.00E−16	
*f*5	2.00E−16	2.00E−16	2.00E−16	
*f*6	2.00E−16	2.00E−16	2.00E−16	–
*f*7	2.00E−16	2.00E−16	2.00E−16	
*f*8	2.00E−16	0.0033	2.00E−16	
*f*9	2.00E−16	2.00E−16	2.00E−16	
*f*10	2.20E−16	2.20E−16	2.20E−16	
*f*11	2.00E−16	2.00E−16	1	
*f*12	2.00E−16	2.00E−16	2.00E−16	
*f*13	2.00E−16	2.00E−16	2.00E−16	
*f*14	1	1.30E−06	2.00E−16	
*f*15	2.00E−16	2.00E−16	5.30E−05	
*f*16	1	2.00E−06	1	
*f*17	1	2.00E−06	1	
*f*18	1	2.00E−06	1	
*f*19	1	2.00E−16	2.00E−16	
*f*20	7.60E−15	0.046	2.00E−16	
*f*21	1	2.00E−16	2.00E−16	
*f*22	1	2.00E−16	2.00E−16	
*f*23	1	2.00E−16	2.00E−16	

Funding The author Shitu Singh acknowledges the funding from South Asian University New Delhi, India to carry out this research.

Compliance with ethical standards

Conflict of interest Author Shitu Singh declares that she has no conflict of interest. Author Jagdish Chand Bansal declares that he has no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

Bansal JC, Joshi SK, Nagar AK (2018) Fitness varying gravitational constant in GSA. Appl Intell 48(10):3446–3461

Bansal JC, Sharma H, Jadon SS, Clerc M (2014) Spider monkey optimization algorithm for numerical optimization. Memetic Comput 6(1):31–47

Dorigo M, Gambardella LM, Birattari M, Martinoli A, Poli R, Stützle T (2006) Ant colony optimization and swarm intelligence. In: 5th international workshop, ants 2006, Brussels, Belgium, 2006, proceedings (VOL. 4150). Springer

El-Fergany AA, Hasanien HM (2015) Single and multi-objective optimal power flow using grey wolf optimizer and differential evolution algorithms. Electric Power Components Syst 43(13):1548–1559

Emary E, Zawbaa HM, Hassanien AE (2016) Binary grey wolf optimization approaches for feature selection. Neurocomputing 172:371–381

Goldberg DE (2006) Genetic algorithms. Pearson Education India, Chennai

Gupta S, Deep K (2019) A novel random walk grey wolf optimizer. Swarm Evolution Comput 44:101–112

Hansen N (2006b) Towards a new evolutionary computation. Stud Fuzziness Soft Comput 192:75–102

Hansen N (2006a) The cma evolution strategy: a comparing review. In: Towards a new evolutionary computation (pp. 75–102). Springer

Heidari AA, Mirjalili S, Faris H, Aljarah I, Mafarja M, Chen H (2019) Harris hawks optimization: algorithm and applications. Future Generat Comput Syst 97:849–872

Jayabarathi T, Raghunathan T, Adarsh B, Suganthan PN (2016) Economic dispatch using hybrid grey wolf optimizer. Energy 111:630–641

Jayakumar N, Subramanian S, Ganesan S, Elanchezhian E (2016) Grey wolf optimization for combined heat and power dispatch with cogeneration systems. Int J Electr Power Energy Syst 74:252–264

Kamboj VK, Bath S, Dhillon J (2016) Solution of non-convex economic load dispatch problem using grey wolf optimizer. Neural Comput Appl 27(5):1301–1316

Karaboga D (2005) An idea based on honey bee swarm for numerical optimization (Tech. Rep.). In: Technical report-tr06, Erciyes university, engineering faculty, computer

Kennedy J (2010) Particle swarm optimization. Encycl Mach Learn 2010:760–766

Komaki G, Kayvanfar V (2015) Grey wolf optimizer algorithm for the two-stage assembly flow shop scheduling problem with release time. J Comput Sci 8:109–120

Mirjalili S (2015a) The ant lion optimizer. Adv Eng Softw 83:80–98

Mirjalili S (2015b) How effective is the grey wolf optimizer in training multi-layer perceptrons. Appl Intell 43:1150–161

Mirjalili S (2015c) Moth-flame optimization algorithm: A novel naturE−inspired heuristic paradigm. Knowl-Based Syst 89:228–249

Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67

Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61

Mittal N, Singh U, Sohi BS (2016) Modified grey wolf optimizer for global engineering optimization. Appl Comput Intell Soft Comput 2016:8

Mohanty S, Subudhi B, Ray PK (2015) A new mppt design using grey wolf optimization technique for photovoltaic system under partial shading conditions. IEEE Trans Sustain Energy 7(1):181–188

Muro C, Escobedo R, Spector L, Coppinger R (2011) Wolf-pack (canis lupus) hunting strategies emerge from simple rules in computational simulations. Behav Process 88((3)):192–197

Pradhan M, Roy PK, Pal T (2017) Oppositional based grey wolf optimization algorithm for economic dispatch problem of power system. Ain Shams Eng J 9:2015–2025

Precup R-E, David R-C, Petriu EM (2016) Grey wolf optimizer algorithm-based tuning of fuzzy control systems with reduced parametric sensitivity. IEEE Trans Industr Electron 64(1):527–534

Simon D (2008) Biogeography-based optimization. IEEE Trans Evol Comput 12((6)):702–713

Sulaiman MH, Mustaffa Z, Mohamed MR, Aliman O (2015) Using the gray wolf optimizer for solving optimal reactive power dispatch problem. Appl Soft Comput 32:286–292

Tizhoosh HR (2005) Opposition-based learning: a new scheme for machine intelligence. In: International conference on computational intelligence for modelling, control and automation and international conference on intelligent agents, web technologies and internet commerce (cimca-iawtic'06) (vol 1, pp 695–701)

Yang X-S (2010) Firefly algorithm, levy flights and global optimization. In: Research and development in intelligent systems xxvi (pp 209–218). Springer

Yang X-S, Deb, S (2009) Cuckoo search via lévy flights. In: 2009 world congress on nature & biologically inspired computing (nabic) (pp 210–214)