Soft Computing (2023) 27:4669–4684
https://doi.org/10.1007/s00500-022-07593-9

An information entropy-based grey wolf optimizer

Kunshan Yao[1] · Jun Sun[1] · Chen Chen[2] · Yan Cao[1] · Min Xu[1] · Xin Zhou[1] · Ningqiu Tang[1] · Yan Tian[1]

Accepted: 11 October 2022 / Published online: 22 October 2022

Abstract

In this study, an entropy-based grey wolf optimizer (IEGWO) algorithm is proposed for solving global optimization problems. This improvement is proposed to alleviate the lack of population diversity, the imbalance between exploitation and exploration, and the premature convergence of grey wolf optimizer algorithm and consists of three aspects: Firstly, we proposed an information entropy-based population generation strategy to optimize the distribution of initial grey wolf pack. Secondly, a modified dynamic position update equation based on information entropy is introduced to maintain the population diversity in the process of iteration, thus avoiding premature convergence. Thirdly, a nonlinear convergence strategy is proposed to balance the exploration and exploitation. The performance of the proposed IEGWO algorithm is assessed on the CEC2014 and CEC2017 test suites and compared with other meta-heuristic algorithms. Furthermore, two engineering design problems and one real-world problem are also solved using the IEGWO algorithm. The experimental and statistical results indicate that the IEGWO algorithm has better solution accuracy and robustness than the compared algorithms in solving global optimization problems.

Keywords Global optimization · Grey wolf optimizer · Information entropy · Meta-heuristic

1 Introduction

Meta-heuristic algorithms play an important role in solving complex problems in different applications due to their simple structure and easy implementation (Manna et al. 2021; Kumar et al. 2021), such as chip design (Venkataraman et al. 2020), diseases diagnosis (Arjenaki et al. 2015), production inventory problem (Das et al. 2021; Manna and Bhunia 2022), feature selection (Hu et al. 2021), and path planning (Qu et al. 2020). Many nature-inspired algorithms have been proposed by simulating the swarm intelligence behavior of various biological systems in nature, such as particle swarm optimization (PSO) (Eberhart and Kennedy 1995), fruit fly optimization algorithm (FOA) (Wang et al. 2013), whale optimization algorithm (WOA) (Mirjalili and Lewis 2016), butterfly optimization algorithm (BOA) (Arora and Singh 2019), conscious neighborhood-based crow search algorithm (CCSA) (Zamani et al. 2019), sparrow search algorithm (SSA) (Xue and Shen 2020), slime mold algorithm (SMA) (Li et al. 2020), etc.

Grey wolf optimizer (GWO) (Mirjalili et al. 2014) is a relatively novel meta-heuristic inspired by the hunting behavior of grey wolf pack and is the only swarm intelligence algorithm based on leadership hierarchy (Luo and Zhao 2019). Due to its excellent optimization performance, GWO has been increasingly focused and successfully applied to many practical engineering problems. Zhang et al. (2016) used GWO to solve the unmanned combat aerial vehicle two-dimension path planning problem. Hadavandi et al. (2018) hybridized GWO and neural network to achieve the prediction of yarns tensile strength. Samuel et al. (2020) combined empirical method and GWO to optimize the extraction process of biodiesel from waste oil. Sundaramurthy and Jayavel (2020) hybridized GWO and PSO with C4.5 approach to realize the prediction of rheumatoid arthritis. Kalemci et al. (2020) designed a reinforced concrete cantilever retaining wall using GWO algorithm. Karasu and Saraç (2020) classified the power quality disturbances by combining GWO and k-Nearest

✉ Jun Sun
sun2000jun@sina.com

[1] School of Electrical and Information Engineering of Jiangsu University, Zhenjiang 212000, China

[2] School of Economics and Management of Jiangsu University of Science and Technology, Zhenjiang 212000, China

Neighbor (KNN). Saxena et al. (2020) utilized GWO to solve the problem of harmonic estimation in power networks. Zareie et al. (2020) developed a GWO-based method to identify the influential users in viral marketing and achieved the best results so far. Naserbegi and Aghaie (2021) applied GWO to optimize the exergy of nuclear-solar dual proposed power plant.

However, the “no free lunch” theorem (Wolpert and Macready 1997) has logically demonstrated that no metaheuristic can perfectly handle all optimization problems. For example, GWO is prone to get stuck at local optimum when dealing with complex multimodal problems (Long et al. 2018a). Therefore, several variants of GWO have been developed. Rodríguez et al. (2017) introduced a fuzzy operator in GWO algorithm to optimize the leadership hierarchy of grey wolves. Long et al. (2017) added a modulation index into GWO algorithm to balance the exploration and exploitation. Tawhid and Ali (2017) embedded genetic mutation operator in GWO to avoid premature convergence. Gupta and Deep (2020) optimized the search mechanism of GWO algorithm by applying crossover and greedy selection. Adhikary and Acharyya (2022) introduced random walk with student’s t distribution into GWO to balance the exploration and exploitation. Mohakud and Dash (2022) proposed a neighborhood based searching strategy that integrates individual haunting strategies and global haunting strategies to balance the exploration and exploitation of GWO.

The above researches indicate that increasing population diversity and balancing exploration and exploitation are an effective approach to improve the performance of GWO. For this purpose, this paper proposed a modified GWO called information entropy-based GWO (IEGWO). In the IEGWO, an initial population generation method based on information entropy is proposed to optimize the distribution of initial grey wolf pack, a dynamic position update mechanism based on information entropy is introduced to maintain the population diversity in the process of iteration, and a nonlinear convergence factor strategy is proposed to balance the exploration and exploitation. The proposed IEGWO algorithm is tested on 10 well-known benchmark functions to analyze the influence of information entropy and nonlinear strategy. The performance of IEGWO is tested and compared with other GWO variants on CEC2014 (Liang et al. 2013) and CEC2017 (Awad et al. 2016) problems. In this paper, the IEGWO is also applied to solve two engineering design problems and to optimize the model parameters in the field of hyperspectral imaging.

The rest of this paper is structured as follows. The next section introduces the original GWO. In Sect. 3 we present the proposed IEGWO in detail. In Sect. 4, the IEGWO is tested by 10 well-known benchmark functions, CEC2014 and CEC2017. In Sect. 5, results on two engineering optimization problems and one practical problem are presented. The last section concludes this paper and provides some ideas for future study.

2 Grey wolf optimizer

The GWO algorithm (Mirjalili et al. 2014) mimics the social leadership and hunting behavior of the grey wolf pack. In this algorithm, the wolves are divided into two parts, one is dominant wolf pack, and the other wolves are named as ω. The dominant wolf pack includes α, β and δ wolves, representing the optimal solution, suboptimal solution, and the third optimal solution, respectively.

To mimic the hunting behavior, the following motion equations are used:

$$\vec{D} = |\vec{C} \cdot \vec{X}_P(t) - \vec{X}(t)| \tag{1}$$

$$\vec{X}(t+1) = \vec{X}_P(t) - \vec{A} \cdot \vec{D} \tag{2}$$

where t represents the current iteration; $\vec{X}_P$ and $\vec{X}$ are the position vectors of the prey and a grey wolf, respectively; $\vec{A}$ and $\vec{C}$ denote the coefficient vectors, which can be calculated as follows:

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \tag{3}$$

$$\vec{C} = 2 \cdot \vec{r}_2 \tag{4}$$

where $\vec{r}_1$ and $\vec{r}_2$ denote random vectors between 0 and 1; $\vec{a}$ is the convergence factor linearly decreased from 2 to 0 as follows:

$$\vec{a}(t) = 2 - \frac{2t}{m} \tag{5}$$

where m indicates the maximum number of iterations.

Figure 1 shows the movement mechanism of GWO, where the red, yellow, purple and blue circles represent α, β, δ and ω wolves, respectively. In this process, the ω

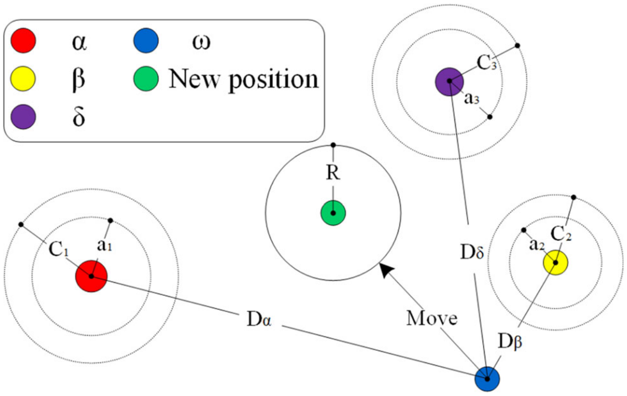


Fig. 1 Evolution of position in GWO

wolves move to the center of α, β and δ wolves by applying the following formulas:

$$\vec{D}_\alpha = |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}|, \vec{D}_\beta = |\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}|, \vec{D}_\delta = |\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}| \tag{6}$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot \vec{D}_\alpha, \vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot \vec{D}_\beta, \vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \cdot \vec{D}_\delta \tag{7}$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{8}$$

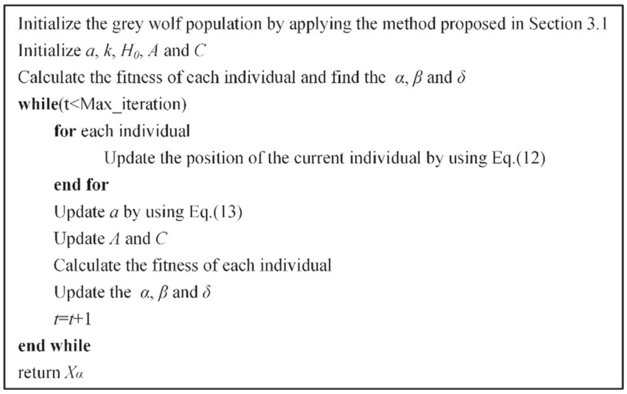
where $\vec{A}_1$, $\vec{A}_2$ and $\vec{A}_3$ are the same as $\vec{A}$, $\vec{C}_1$, $\vec{C}_2$ and $\vec{C}_3$ are the same as $\vec{C}$. Figure 2 presents the pseudocode of GWO algorithm.

3 Information entropy-based grey wolf optimizer

The theorem of entropy was originally proposed by Shannon (1948) to describe the complexity of spatial energy distribution, which plays a fundamental and important role in the field of modern information theory. The formula of information entropy is as follows (Feng et al. 2022):

$$H(p_1, p_2, \ldots p_n) = -\sum_{i=1}^{n} p_i(x) \log_2 p_i(x) \tag{9}$$

where n is the number of information records in a system and $p_i(x)$ is the probability of the ith record. For meta-heuristic algorithms, the information entropy can also be used to reflect the diversity of population. Inspired by this idea, the information entropy is introduced into GWO to increase the population diversity in the process of generating initial population and updating position. Besides, a nonlinear convergence factor $\vec{a}$ is proposed to balance the exploration and exploitation.



```
Initialize the grey wolf population by applying the method proposed in Section 3.1
Initialize $a$, $k$, $H_0$, $A$ and $C$
Calculate the fitness of each individual and find the $\alpha$, $\beta$ and $\delta$
while(t<Max_iteration)
    for each individual
        Update the position of the current individual by using Eq.(12)
    end for
    Update $a$ by using Eq.(13)
    Update $A$ and $C$
    Calculate the fitness of each individual
    Update the $\alpha$, $\beta$ and $\delta$
    $t$=$t$+1
end while
return $X_\alpha$
```

Fig. 2 Pseudocode of GWO algorithm

3.1 Initial population generation based on information entropy

In the original GWO, the initial population is generated by random sampling. However, in practical application, it is inevitable that some grey wolves may be overly concentrated in a local area, resulting in a restricted search range. To optimize the distribution of initial population in the solution space, an information entropy-based sampling method was proposed. Suppose that there is a population with group size of N and dimension of D, the entropy H_j of the jth dimension can be defined by:

$$H_j = \sum_{i=1}^{N-1} \frac{1}{N-1} \sum_{k=i+1}^{N} \frac{-P_{ik}\log_2 P_{ik} - (1-P_{ik})\log_2(1-P_{ik})}{N-i}$$
$$P_{ik} = 1 - \frac{|x_j^i - x_j^k|}{2(ub_j - lb_j)} \tag{10}$$

where ub_j and lb_j denote the upper and lower limits of the jth dimension, respectively. The value of P_{ik} describes the similarity probability of the jth dimension between the ith and the kth individuals. The entropy value of the whole population H is defined by:

$$H = \frac{1}{D}\sum_{j=1}^{D} H_j \tag{11}$$

The process of generating initial population is as follows:

Step1: Set a critical value H_0 for entropy (for example, $H_0 = 0.25$).
Step2: Generate the first individual by random sampling method in the solution space.
Step3: Generate a new individual by random sampling. If the entropy value of the population $H > H_0$, the new individual will be retained; otherwise, a new individual will be regenerated until $H > H_0$.
Step4: Repeat Step 3 until enough individuals are generated.

3.2 Dynamic position update equation

In the original GWO algorithm, all the ω wolves constantly move to the center of α, β and δ wolves in the process of iteration. As a consequence, GWO is prone to fall into local optimum due to the over learning from the dominant wolf pack and the rapid reduction in population diversity. To maintain the population diversity during the process of updating position and weaken the influence of dominant wolf pack, we modified the position-updating equation as follows:

$$\vec{X}(t+1) = w_1\vec{X}_1 + w_2\vec{X}_2 + w_3\vec{X}_3$$
$$w_1 = \frac{P_{\omega\alpha}}{P_{\omega\alpha} + P_{\omega\beta} + P_{\omega\delta}},$$
$$w_2 = \frac{P_{\omega\beta}}{P_{\omega\alpha} + P_{\omega\beta} + P_{\omega\delta}}, w_1 = \frac{P_{\omega\delta}}{P_{\omega\alpha} + P_{\omega\beta} + P_{\omega\delta}} \tag{12}$$

where $P_{\omega\alpha}$, $P_{\omega\beta}$ and $P_{\omega\delta}$ are the same of P_{ik} in Eq. (10). It can be seen from Eq. (12) that each w wolf updates its position according to its similarity to the dominant wolf pack to avoid over-concentration of wolves. It should be noted that the weights of w_1, w_2 and w_3 are constantly changing in the process of iteration. Thus, it can bring more information and maintain the diversity of population.

3.3 Nonlinear convergence factor $\vec{a}$

From the original paper of GWO, the grey wolves attack toward prey when the value $|A| < 1$ and diverge from prey when the value $|A| > 1$. Therefore, the exploration and exploitation are balanced by the linear convergence factor $\vec{a}$ using Eq. (5), and the ratio is 1:1. However, the real search process is highly complex, and the linear strategy is difficult to adapt to it. Based on the above consideration, we proposed a nonlinear convergence factor $\vec{a}$ as follows:

$$\vec{a} = \begin{cases} 2 - 2k + 2k[-P_1\log_2 P_1 - (1-P_1)\log_2(1-P_1)], P_1 = \frac{1}{2}\left(1 + \frac{t}{km}\right), & 0 < t < km \\ 2 - 2k, & t = km \\ (2-2k)[1 - (-P_2\log_2 P_2 - (1-P_2)\log_2(1-P_2))], P_2 = \frac{1}{2}\left(\frac{t-km}{m-km}\right), & km < t < m \end{cases} \tag{13}$$

where t denotes the current iteration, m indicates the maximum number of iterations and k is the nonlinear modulation parameter in [0, 1]. The iterative curves of convergence factor $\vec{a}$ with different values of k are shown in Fig. 3.

Compared with the original GWO algorithm, the value of $\vec{a}$ is increased in the early stage to explore more unknown regions and then reduced to narrow the search range in the later stage. In other words, the modification is to enhance the global exploration in the early stage and the local exploitation in the later stage, and the parameter k is used to control the proportion of exploration and exploitation. The ratio of iterations used for exploration and exploitation with different values of k is shown in Table 1.

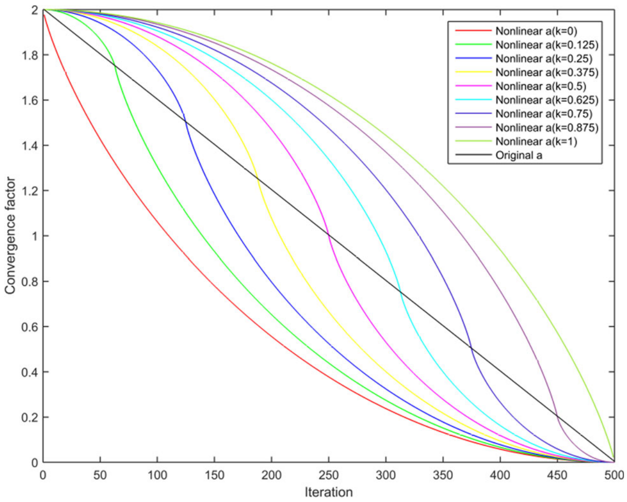


Fig. 3 Iterative curves of nonlinear convergence factor $\vec{a}$ with different values of k

The pseudocode of the proposed IEGWO is presented in Fig. 4. The influence of parameter k to the performance of IEGWO will be discussed in the parameter setting experiment.

4 Results and discussion

4.1 Comparison with GWO

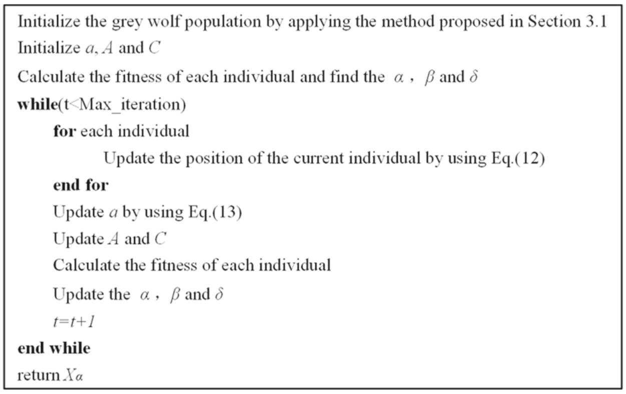
4.1.1 Benchmark functions and parameter settings

In this section, the impact of information entropy and nonlinear convergence factor on the performance of IEGWO is analyzed by 10 well-known benchmark functions, including 5 unimodal benchmark functions and 5 multimodal benchmark functions. These functions are listed in Table 2, where $f_{\min}$ denotes the theoretical optimal value in search range. The 2-D versions of these functions are shown in Fig. 5.

In all experiments, the maximum number of iterations, the population size, and the dimension of test functions

Table 1 Ratio of exploration and exploitation with different values of k

k	0 (%)	0.125 (%)	0.25 (%)	0.375 (%)	0.5 (%)	0.625 (%)	0.75 (%)	0.875 (%)	1 (%)
Exploration	22.2	27.8	34.2	41.4	50	58.6	65.8	72.2	77.8
Exploitation	77.8	72.2	65.8	58.6	50	41.4	34.2	27.8	22.2



```
Initialize the grey wolf population by applying the method proposed in Section 3.1
Initialize $a, A$ and $C$
Calculate the fitness of each individual and find the $\alpha$ , $\beta$ and $\delta$
while(t<Max_iteration)
    for each individual
        Update the position of the current individual by using Eq.(12)
    end for
    Update $a$ by using Eq.(13)
    Update $A$ and $C$
    Calculate the fitness of each individual
    Update the $\alpha$ , $\beta$ and $\delta$
    $t=t+1$
end while
return $X\alpha$
```

Fig. 4 Pseudocode of IEGWO algorithm

were fixed as 500, 30, and 30, respectively. Each algorithm was executed 30 times independently. All programs were coded in MATLAB 2016b and ran on a computer with CPU of AMD Ryzen7 5800U (4.4 GHz) under Windows 10 system.

4.1.2 The impact of information entropy and nonlinear convergence factor

To investigate the impact of information entropy and nonlinear convergence factor, the GWO, IEGWO with original linear convergence factor (IEGWO-L), and IEGWO with different k values were executed on the 10 benchmark functions, and the results are shown in Table 3.

Table 2 Test functions

Function type	Function formula	Range	$f_{\min}$
Unimodal functions	$F_1(x) = \sum_{i=1}^{n} x_i^2$	[− 100, 100]	0
	$F_2(x) = \sum_{i=1}^{n} \lvert x_i\rvert + \prod_{i=1}^{n} \lvert x_j\rvert$	[− 10, 10]	0
	$F_3(x) = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j\right)^2$	[− 100, 100]	0
	$F_4(x) = \max_i\{\lvert x_i\rvert, 1 \le i \le n\}$	[− 100, 100]	0
	$F_5(x) = \sum_{i=1}^{n} ix_i^4 + \text{random }[0, 1]$	[− 1.28, 1.28]	0
Multimodal functions	$F_6(x) = [x_i^2 - 10\cos(2\pi x_i) + 10]$	[− 5.12, 5.12]	0
	$F_7(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e$	[− 32, 32]	0
	$F_8(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[− 600, 600]	0
	$F_9(x) = \frac{\pi}{n}\left\{10\sin(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)\left[1 + 10\sin^2(\pi y_{y+1})\right] + (y_n - 1)^2\right\}$ $+ \sum_{i=1}^{n} u(x_i, 5, 100, 4)$ $\sum_{i=1}^{n} u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, x_i > a \\ 0, -a < x_i < a \\ k(-x_i - a)^m,\ x_i < -a \end{cases}$	[− 50, 50]	0
	$F_{10}(x) = 0.1\left\{\sum_{i=1}^{n}\begin{array}{c} \sin^2(3\pi x_1) + \\ (x_i - 1)^2\left[1 + \sin^2(3\pi x_i + 1)\right] \\ +(x_n - 1)^2\left[1 + \sin^2(2\pi x_n)\right] \end{array}\right\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	[− 50, 50]	0

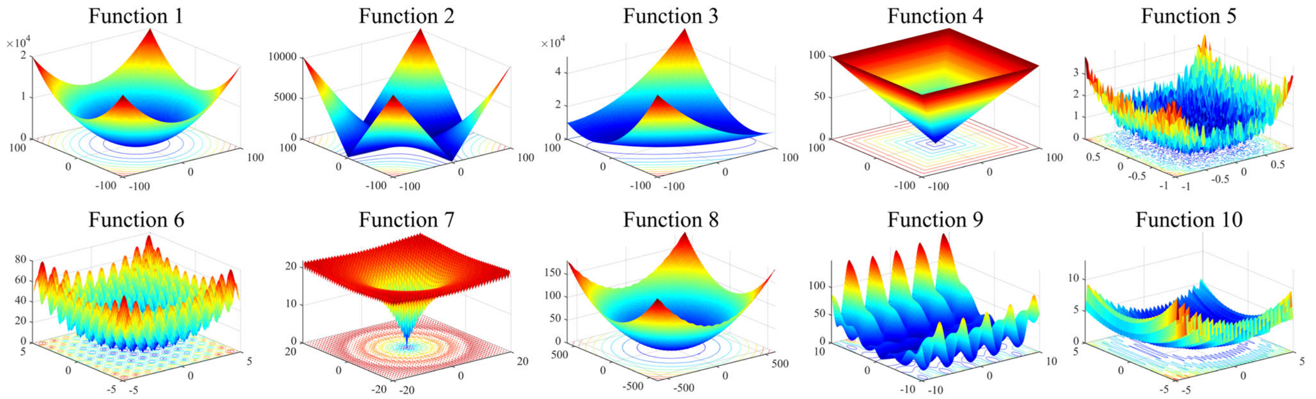


Fig. 5 2-D version of test functions

The unimodal functions (F1–F5) can be used to evaluate the exploitation performance of algorithms, while the multimodal functions (F6–F10) are utilized to examine the exploration strength and the ability of avoiding local optimum of algorithms. As seen from Table 3, the optimization performance of IEGWO-L outperforms that of GWO on all benchmark functions. Although the improvement in solution accuracy is small on the unimodal functions, it is significant on the multimodal functions, which is because the increase in population diversity improves the ability of the algorithm to jump out of the local optimum. The reduction in standard deviation (Std Dev) indicates that the introduction of information entropy can effectively improve the robustness of GWO. With the increase in k, the performance of IEGWO is evidently improved, and when the value of k is greater than 0.75, the IEGWO outperforms GWO and IEGWO-L obviously. Therefore, for unimodal functions, more exploration helps to improve the optimization accuracy and robustness, while for multimodal functions, the optimization performance of IEGWO increases and then decreases as the proportion of exploration increases, and when the value of k is 0.875, the error of the objective function values reaches the lowest, indicating that the exploration and exploitation are well balanced.

4.1.3 Wilcoxon test

In order to statistically assess the performance of GWO and IEGWO, Wilcoxon test at a 5% significance is utilized. The results are recorded in Table 4, where the following rating criteria are applied.

1. A—IEGWO has better performance and p-value ≤ 0.05.
2. B—The performance of the two algorithms is comparable and p-value > 0.05.
3. C—IEGWO has poor performance and p-value ≤ 0.05.

From Tables 1 and 4, it can be seen that when the exploration ratio is greater than 65.8%, the performance of IEGWO is significantly better than that of GWO. When the exploration ratio reaches 72.2%, all the ratings of IEGWO are A, indicating that more proportion of exploration can effectively improve the optimization performance of IEGWO. However, with the increasing proportion of exploration, the performance of IEGWO on multimodal functions decreases due to the scarcity of exploitation. Therefore, 0.875 is an appropriate value for the parameter k, which provides a good balance between exploitation and exploration.

4.1.4 Computational complexity and running time

The computational complexity is an important index to evaluate the running time of an optimization algorithm, which can be defined based on the structure of the algorithm (Gupta and Deep 2019). The major computational cost of GWO and IEGWO is in the while loop, both equal to $O\,(N \times D \times T)$, where N is the population size, D is the dimension of the problem, and T is the maximum number of iterations. Figure 6 shows the total time consumed by GWO and IEGWO to run 30 times on each function. As shown in Fig. 6, the running time of IEGWO is approximately 6% longer than that of GWO, which is mainly due to the fact that some grey wolf individuals need to be regenerated to meet the critical value of information entropy in the process of generating initial population.

4.1.5 Convergence analysis

The convergence curves of GWO and IEGWO (k = 0.875) are shown in Fig. 7. Due to the expansion of the search range in the early stage, the error in objective function

Table 3 Experimental results of GWO and IEGWO

Function	Algorithm	Average	Std Dev	Minimum	Maximum
F1	GWO	1.01E−27	1.75E−27	1.57E−29	6.78E−27
	IEGWO-L	2.29E−28	2.21E−28	1.06E−29	7.21E−28
	IEGWO (k = 0.000)	2.34E−13	1.70E−13	2.28E−14	6.53E−13
	IEGWO (k = 0.125)	9.87E−16	1.15E−15	5.35E−17	6.46E−15
	IEGWO (k = 0.250)	1.40E−18	1.66E−18	6.22E−20	7.29E−18
	IEGWO (k = 0.375)	2.53E−22	4.36E−22	3.95E−24	2.21E−21
	IEGWO (k = 0.500)	7.81E−26	2.30E−25	1.29E−27	1.26E−24
	IEGWO (k = 0.625)	7.67E−30	1.44E−29	4.64E−31	7.36E−29
	IEGWO (k = 0.750)	5.97E−33	9.80E−33	8.99E−35	4.42E−32
	IEGWO (k = 0.875)	3.29E−36	8.27E−36	9.40E−39	4.32E−35
	IEGWO (k = 1.000)	7.91E−40	1.27E−39	4.63E−42	5.95E−39
F2	GWO	1.32E−16	1.17E−16	1.02E−17	4.70E−16
	IEGWO-L	8.13E−17	4.94E−17	9.98E−18	1.70E−16
	IEGWO (k = 0.000)	2.52E−08	1.20E−08	7.45E−09	6.20E−08
	IEGWO (k = 0.125)	1.26E−09	5.55E−10	5.21E−10	2.62E−09
	IEGWO (k = 0.250)	2.13E−11	1.09E−11	3.94E−12	4.57E−11
	IEGWO (k = 0.375)	1.35E−13	8.30E−14	2.96E−14	4.13E−13
	IEGWO (k = 0.500)	8.50E−16	6.33E−16	1.23E−16	2.73E−15
	IEGWO (k = 0.625)	5.95E−18	6.02E−18	6.12E−19	2.50E−17
	IEGWO (k = 0.750)	6.59E−20	5.28E−20	8.33E−21	2.08E−19
	IEGWO (k = 0.875)	5.65E−22	5.61E−22	1.23E−22	2.54E−21
	IEGWO (k = 1.000)	5.70E−24	6.02E−24	3.66E−25	2.41E−23
F3	GWO	1.62E−05	4.71E−05	2.86E−08	2.40E−04
	IEGWO-L	2.74E−06	6.94E−06	1.02E−08	2.24E−05
	IEGWO (k = 0.000)	3.10E−02	4.16E−02	9.14E−04	1.59E−02
	IEGWO (k = 0.125)	1.64E−02	4.91E−02	7.16E−05	2.70E−01
	IEGWO (k = 0.250)	1.11E−03	1.40E−03	4.32E−06	4.21E−03
	IEGWO (k = 0.375)	2.74E−04	5.33E−04	4.68E−07	2.08E−03
	IEGWO (k = 0.500)	1.12E−04	3.04E−04	2.86E−09	1.63E−03
	IEGWO (k = 0.625)	4.17E−05	1.37E−04	8.78E−10	5.51E−04
	IEGWO (k = 0.750)	1.14E−06	2.35E−06	3.32E−10	1.12E−05
	IEGWO (k = 0.875)	2.52E−07	8.18E−07	1.68E−11	3.77E−06
	IEGWO (k = 1.000)	5.34E−08	1.63E−07	5.57E−12	7.29E−07
F4	GWO	7.43E−07	7.27E−07	6.23E−08	3.09E−06
	IEGWO-L	3.17E−07	2.31E−07	3.41E−08	9.18E−07
	IEGWO (k = 0.000)	4.17E−03	2.26E−03	1.38E−03	1.08E−02
	IEGWO (k = 0.125)	7.51E−04	6.70E−04	2.36E−04	2.85E−03
	IEGWO (k = 0.250)	5.63E−05	3.51E−05	1.04E−05	1.45E−04
	IEGWO (k = 0.375)	8.64E−06	1.31E−05	9.32E−07	6.87E−05
	IEGWO (k = 0.500)	6.27E−07	8.72E−07	4.45E−08	4.55E−06
	IEGWO (k = 0.625)	6.85E−08	8.92E−08	2.50E−09	4.05E−07
	IEGWO (k = 0.750)	5.14E−09	4.20E−09	9.98E−10	2.16E−08
	IEGWO (k = 0.875)	7.33E−10	8.05E−10	1.95E−11	2.93E−09
	IEGWO (k = 1.000)	1.29E−10	1.55E−10	4.73E−12	7.67E−10

Table 3 (continued)

Function	Algorithm	Average	Std Dev	Minimum	Maximum
F5	GWO	2.12E−03	1.21E−03	7.16E−04	6.38E−03
	IEGWO-L	1.95E−03	8.65E−04	5.50E−04	3.31E−03
	IEGWO (k = 0.000)	5.05E−03	2.04E−03	1.60E−03	9.82E−03
	IEGWO (k = 0.125)	3.45E−03	1.60E−03	6.19E−04	8.32E−03
	IEGWO (k = 0.250)	2.41E−03	1.19E−03	2.26E−04	4.86E−03
	IEGWO (k = 0.375)	2.26E−03	1.21E−03	1.73E−04	4.95E−03
	IEGWO (k = 0.500)	1.58E−03	9.78E−04	3.27E−04	4.04E−03
	IEGWO (k = 0.625)	1.65E−03	9.78E−04	3.97E−04	3.50E−03
	IEGWO (k = 0.750)	1.42E−03	8.23E−04	5.12E−04	4.36E−03
	IEGWO (k = 0.875)	1.24E−03	6.94E−04	4.14E−04	2.99E−03
	IEGWO (k = 1.000)	1.20E−03	6.69E−04	2.49E−04	2.73E−03
F6	GWO	2.3658	3.9833	5.68E−14	12.1458
	IEGWO-L	1.3073	2.9159	5.68E−14	8.5567
	IEGWO (k = 0.000)	14.6802	6.1002	5.1678	29.5166
	IEGWO (k = 0.125)	3.6039	3.2711	1.14E−12	15.5130
	IEGWO (k = 0.250)	2.6884	4.6974	4.55E−13	20.4066
	IEGWO (k = 0.375)	2.1049	3.3490	5.68E−14	10.5640
	IEGWO (k = 0.500)	0.6802	2.4129	0	12.2120
	IEGWO (k = 0.625)	0.6679	1.8368	0	6.6453
	IEGWO (k = 0.750)	0.0340	0.1864	0	1.0210
	IEGWO (k = 0.875)	1.33E−14	2.86E−14	0	1.14E−13
	IEGWO (k = 1.000)	0.0758	0.4152	0	2.2743
F7	GWO	1.02E−13	1.29E−14	7.90E−14	1.29E−13
	IEGWO-L	8.94E−14	9.83E−15	7.19E−14	1.00E−13
	IEGWO (k = 0.000)	1.09E−07	4.97E−08	3.91E−08	2.35E−07
	IEGWO (k = 0.125)	8.03E−09	4.16E−09	3.15E−09	2.08E−08
	IEGWO (k = 0.250)	1.81E−10	1.88E−10	4.13E−11	1.01E−09
	IEGWO (k = 0.375)	3.66E−12	3.27E−12	4.66E−13	1.42E−11
	IEGWO (k = 0.500)	1.57E−13	5.03E−14	1.00E−13	2.92E−13
	IEGWO (k = 0.625)	4.14E−14	1.06E−14	2.93E−14	7.55E−14
	IEGWO (k = 0.750)	2.53E−14	5.65E−15	1.51E−14	4.00E−14
	IEGWO (k = 0.875)	1.52E−14	2.87E−15	7.99E−15	2.22E−14
	IEGWO (k = 1.000)	1.70E−14	3.46E−15	1.51E−14	2.93E−14
F8	GWO	4.51E−03	7.07E−03	0	1.73E−02
	IEGWO-L	1.22E−03	3.86E−03	0	1.22E−02
	IEGWO (k = 0.000)	8.16E−03	1.01E−02	1.63E−13	3.11E−02
	IEGWO (k = 0.125)	4.07E−03	9.01E−03	5.44E−15	3.55E−02
	IEGWO (k = 0.250)	2.45E−03	6.47E−03	1.11E−16	2.31E−02
	IEGWO (k = 0.375)	3.96E−03	8.38E−03	0	2.63E−02
	IEGWO (k = 0.500)	3.15E−03	8.77E−03	0	3.31E−02
	IEGWO (k = 0.625)	1.23E−03	4.70E−03	0	2.02E−02
	IEGWO (k = 0.750)	1.07E−03	4.09E−03	0	1.74E−02
	IEGWO (k = 0.875)	3.18E−04	1.74E−03	0	9.53E−03
	IEGWO (k = 1.000)	9.32E−04	3.61E−03	0	1.66E−02

Table 3 (continued)

Function	Algorithm	Average	Std Dev	Minimum	Maximum
F9	GWO	0.0466	0.0201	0.0194	0.090
	IEGWO-L	0.0396	0.0171	0.0183	0.076
	IEGWO (k = 0.000)	0.0981	0.1049	0.0133	0.5874
	IEGWO (k = 0.125)	0.0362	0.0171	0.0131	0.0729
	IEGWO (k = 0.250)	0.0504	0.0998	0.0127	0.5695
	IEGWO (k = 0.375)	0.0419	0.0256	0.0062	0.1130
	IEGWO (k = 0.500)	0.0386	0.0198	0.0195	0.1062
	IEGWO (k = 0.625)	0.0370	0.0202	0.0111	0.0944
	IEGWO (k = 0.750)	0.0360	0.0152	0.0125	0.0734
	IEGWO (k = 0.875)	0.0358	0.0210	0.00626	0.0842
	IEGWO (k = 1.000)	0.0389	0.0189	0.0132	0.1080
F10	GWO	0.6107	0.2920	0.1837	1.3925
	IEGWO-L	0.5236	0.1963	0.1114	0.7720
	IEGWO (k = 0.000)	1.1242	0.2621	0.4016	1.5961
	IEGWO (k = 0.125)	0.5528	0.2192	0.1000	1.0598
	IEGWO (k = 0.250)	0.4917	0.2106	0.2013	1.1431
	IEGWO (k = 0.375)	0.5043	0.2155	0.0970	1.0649
	IEGWO (k = 0.500)	0.4920	0.2461	0.0923	1.0324
	IEGWO (k = 0.625)	0.4893	0.1746	0.1982	0.9455
	IEGWO (k = 0.750)	0.4814	0.2090	0.1004	0.8997
	IEGWO (k = 0.875)	0.4594	0.1841	0.1005	0.7576
	IEGWO (k = 1.000)	0.4736	0.1423	0.2019	0.8379

values of IEGWO is larger than that of GWO in some functions (F1–F4, F7). However, the expansion also increases the possibility of finding better solutions by exploring more unknown regions, so the error reduction rate of IEGWO is accelerated in the middle stage. In the late stage of optimization, IEGWO has higher solution accuracy due to the enhancement of local exploitation. From Fig. 7, it can be concluded that the proposed strategies effectively improve the performance of GWO.

4.1.6 Comparison with other modified GWO algorithms

The previous section shows that IEGWO has better performance than GWO in terms of exploitation and exploration. In this section, the IEGWO is compared with E-GWO (Duarte et al. 2020), IGWO (Long et al. 2018a), EEGWO (Long et al. 2018b), and FH-GWO (Rodríguez et al. 2017) on 30 dimensional CEC2014 and CEC2017 problems based on the average error in objective function value. The parameter setting of the three algorithms is the same as reported in their original papers. The comparison is presented in Tables 5 and 6. For clarity, the best solutions are marked in boldface. Tables 5 and 6 show that the proposed IEGWO has the smallest error in 21 of the 30 functions on CEC 2014 and 18 of the 29 functions on CEC 2017. The outcomes obtained by applying the Wilcoxon test are also recorded in the same tables. Whether for the unimodal, multimodal, and hybrid benchmark problems, the IEGWO provides better results compared to other GWO variants in most of the tested functions. Therefore, it can be concluded that the performance of IEGWO is significantly better than that of the other modified GWO algorithms.

5 Real applications of IEGWO

The proposed IEGWO algorithm is applied to two classical engineering applications: tension/compression spring design and pressure vessel design, which are often used as constrained optimization benchmark problems (Mirjalili et al. 2014). Besides, the IEGWO is also employed to optimize the model parameters in the field of hyperspectral imaging.

5.1 Tension/compression spring design

The aim of the problem is to minimize the weight of a spring with constraints such as minimum deflection, shear stress, and surge frequency. The design variables include

Table 4 Results of Wilcoxon test

Function	Rating								
	k = 0.000	*k* = 0.125	*k* = 0.250	*k* = 0.375	*k* = 0.500	*k* = 0.625	*k* = 0.750	*k* = 0.875	*k* = 1.000
F1	C	C	C	C	C	A	A	A	A
F2	C	C	C	C	C	A	A	A	A
F3	C	C	C	C	C	B	A	A	A
F4	C	C	C	C	B	A	A	A	A
F5	C	C	C	C	B	B	A	A	A
F6	C	C	C	B	A	A	A	A	A
F7	C	C	C	C	C	A	A	A	A
F8	C	C	C	B	A	A	A	A	A
F9	C	B	C	B	B	A	A	A	B
F10	C	B	B	B	B	B	B	A	A

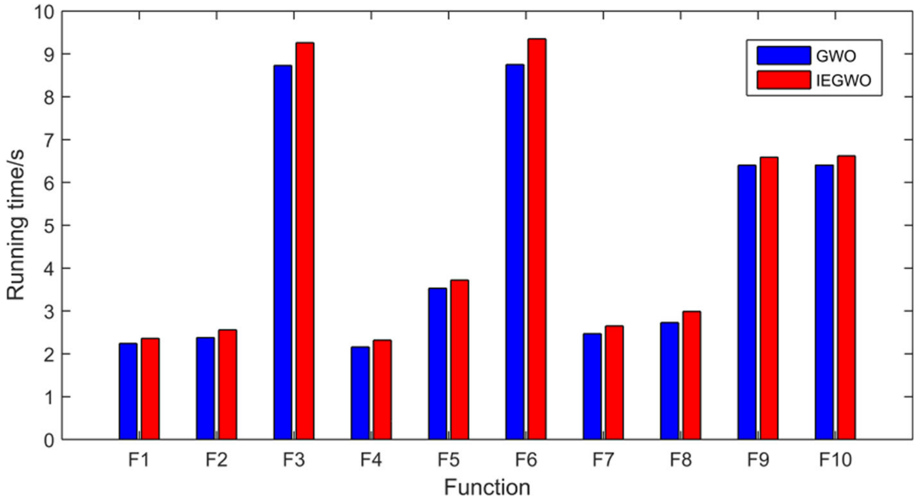


Fig. 6 Running time of GWO and IEGWO

wire diameter $d(x_1)$, the mean coil diameter $D(x_2)$, and the number of active coils $N(x_3)$. The mathematical formulation of this problem is stated as follows:

$$\text{Minimize} \qquad f(x) = x_1^2 x_2 (x_3 + 2)$$

$$\text{s.t.} \quad \begin{aligned} g_1(x) &= \frac{x_2^3 x_3}{71785 x_1^4} + 1 \leq 0 \\ g_2(x) &= \frac{4x_2^2\, x_1 x_2}{12566(x_1^3 x_2\, x_1^4)} + \frac{1}{5108 x_1^2} - 1 \leq 0 \\ g_3(x) &= \frac{140.45 x_1}{x_2^2 x_3} + 1 \leq 0 \\ g_4(x) &= \frac{x_1 + x_2}{1.5} - 1 \leq 0 \end{aligned}$$

Variable range

$$0.05 \leq x_1 \leq 2,\ 0.25 \leq x_2 \leq 1.3,\ 2 \leq x_3 \leq 15$$

The GA (Coello and Montes 2002), CPSO (He and Ling 2007), SMA (Li et al. 2020), FH-GWO (Rodríguez et al. 2017), IGWO (Long et al. 2018a), EEGWO (Long et al. 2018b), E-GWO (Duarte et al. 2020), and the proposed IEGWO are applied to solve this problem. The population size and maximum iterations of all algorithms are set to 20 and 1500. The results obtained by conducting 30 runs are shown in Table 7. In the same table, the outcomes of the statistical results obtained by using Wilcoxon test are also recorded. The table verified the better performance of IEGWO compared to other meta-heuristic algorithms.

5.2 Pressure vessel design

The goal of this problem is to minimize the cost including material, forming and welding of a vessel as shown in Fig. 8. The design variables include shell thickness $Ts(x_1)$, head thickness $T_h(x_2)$, inner radius $R(x_3)$, and shell length $L(x_4)$. The mathematical formulation of this problem is as follows:

Fig. 7 Convergence curves of GWO and IEGWO

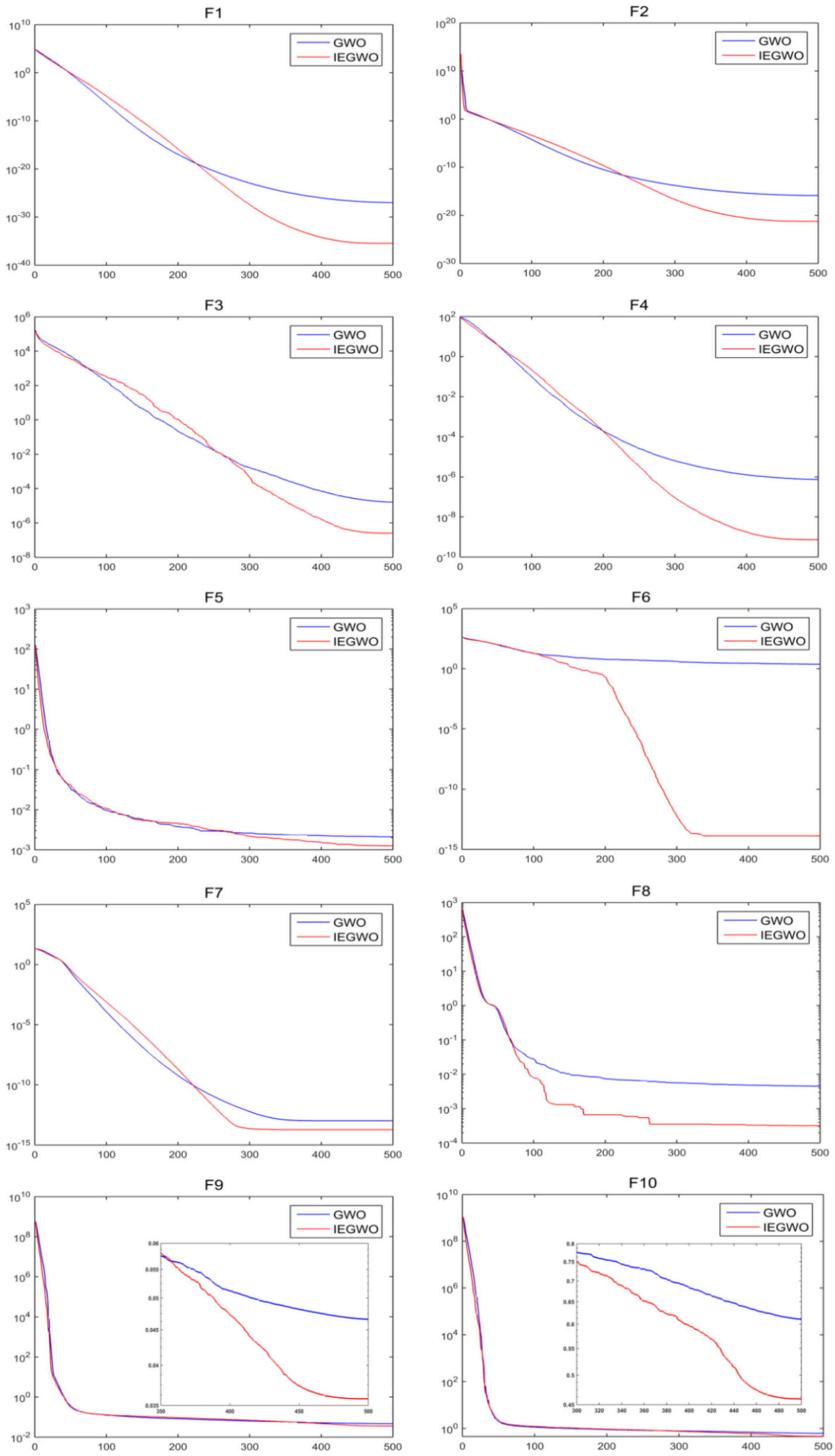


Table 5 Comparison of average error in objective function values on CEC 2014

Problem	E-GWO	IGWO	EEGWO	FH-GWO	IEGWO
F1	4.35E+08	2.33E+08	5.18E+08	7.85E+07	**6.89E+07**
F2	5.32E+09	1.18E+10	7.01E+09	4.33E+09	**3.41E+09**
F3	**2.15E+04**	5.30E+04	2.50E+04	4.05E+04	3.94E+04
F4	4.02E+02	7.87E+02	2.59E+03	5.21E+02	**3.12E+02**
F5	2.10E+01	2.10E+01	2.19E+01	2.10E+01	**2.09E+01**
F6	2.89E+01	2.71E+01	**1.18E+01**	1.69E+01	1.53E+01
F7	1.78E+02	1.28E+02	2.11E+02	8.31E+01	**3.00E+01**
F8	2.16E+02	2.21E+02	**1.58E+02**	1.02E+03	9.35E+02
F9	1.96E+02	3.25E+02	1.38E+02	1.25E+02	**1.11E+02**
F10	**1.83E+03**	4.88E+03	1.85E+03	4.52E+03	2.53E+03
F11	**3.23E+03**	5.91E+03	3.41E+03	4.25E+03	3.31E+03
F12	2.77E+00	2.65E+00	3.28E+00	2.43E+00	**2.18E+00**
F13	1.59E+00	2.67E+00	4.97E+00	8.51E-01	**5.42E−01**
F14	3.98E+01	4.23E+01	3.73E+01	1.01E+01	**6.64E+00**
F15	7.54E+03	2.91E+03	8.02E+03	8.91E+02	**4.43E+02**
F16	2.26E+01	1.28E+01	4.14E+01	1.88E+01	**1.15E+01**
F17	6.90E+06	9.56E+06	8.98E+05	**6.29E+05**	2.52E+06
F18	3.25E+06	4.14E+07	2.77E+06	**2.01E+06**	1.09E+07
F19	6.58E+01	1.02E+02	5.89E+01	5.59E+01	**5.28E+01**
F20	6.54E+04	3.85E+04	8.57E+05	2.25E+04	**1.96E+04**
F21	4.35E+06	2.31E+06	5.75E+06	1.03E+06	**8.20E+05**
F22	7.65E+02	6.87E+02	5.21E+02	4.98E+02	**4.11E+02**
F23	4.21E+02	3.84E+02	2.31E+02	1.86E+02	**1.37E+02**
F24	2.26E+02	2.00E+02	2.00E+02	2.00E+02	**2.00E+02**
F25	2.56E+02	2.13E+02	2.10E+02	2.09E+02	**2.09E+02**
F26	2.05E+02	1.80E+02	1.79E+02	1.75E+02	**1.70E+02**
F27	3.44E+03	9.68E+02	3.91E+02	**3.52E+02**	6.58E+02
F28	1.89E+03	4.09E+03	2.13E+02	**2.00E+02**	1.46E+03
F29	1.27E+07	3.49E+07	1.02E+06	7.49E+05	**3.46E+05**
F30	5.67E+05	4.35E+05	3.78E+05	8.71E+04	**6.76E+04**
A/B/C	21/5/4	24/6/0	20/5/5	19/6/4	

Minimize

$$f(x) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$$

$$\text{s.t.} \quad \begin{aligned} g_1(x) &= -x_1 + 0.0193x_3 \le 0 \\ g_2(x) &= -x_2 + 0.00954x_3 \le 0 \\ g_3(x) &= -\pi x_3^2x_4 - \tfrac{4}{3}\pi x_3^3 + 1296000 \le 0 \\ g_4(x) &= x_4 - 240 \le 0 \end{aligned}$$

$$\text{Variable range} \quad \begin{aligned} 0 \le x_1,\ x_2 \le 99 \\ 10 \le x_3,\ x_4 \le 200 \end{aligned}$$

The proposed IEGWO, GA (Coello and Montes 2002), CPSO (He and Ling 2007), SMA (Li et al. 2020), FH-GWO (Rodríguez et al. 2017), IGWO (Long et al. 2018a), EEGWO (Long et al. 2018b), and E-GWO (Duarte et al. 2020) are applied to solve this problem. The population size and maximum iterations of all algorithms are set to 20 and 2000. The results obtained by conducting 30 runs are shown in Table 8. In the same table, the outcomes of the statistical results obtained by using Wilcoxon test are also recorded. Except for SMA, the performance of IEGWO is significantly better than that of the other algorithms. However, the optimal solution found by IEGWO is better than the best known solution so far.

5.3 Optimization of model parameters in the field of hyperspectral imaging

In recent years, many researchers have applied hyperspectral imaging (HSI) technique in the nondestructive testing of food and agricultural products and developed various methods to improve the performance of models (Yao et al. 2022; Suktanarak and Teerachaichayut 2017). Among them, parameter optimization is a commonly used

Table 6 Comparison of average error in objective function values on CEC 2017

Problem	E-GWO	IGWO	EEGWO	FH-GWO	IEGWO
F1	1.25E+10	9.21E+09	1.42E+10	2.88E+09	**2.42E+09**
F3	5.10E+04	5.55E+04	1.41E+04	4.00E+04	**3.72E+03**
F4	**1.72E+02**	8.30E+02	1.44E+03	1.81E+02	1.97E+02
F5	2.16E+02	2.25E+02	1.44E+02	**1.19E+02**	1.26E+02
F6	6.24E+01	4.84E+01	6.96E+01	1.31E+01	**1.15E+01**
F7	3.29E+02	3.91E+02	**1.55E+02**	2.05E+02	1.72E+02
F8	2.34E+02	2.11E+02	1.28E+02	9.76E+01	**9.56E+01**
F9	3.68E+03	4.23E+03	1.44E+03	**1.17E+03**	1.29E+03
F10	3.65E+03	6.67E+03	**2.87E+03**	4.50E+03	4.05E+03
F11	6.38E+03	2.51E+03	9.91E+03	1.10E+03	**1.08E+03**
F12	6.59E+08	4.78E+08	1.07E+09	9.81E+07	**7.70E+07**
F13	6.65E+07	1.41E+08	5.59E+07	2.11E+07	**1.66E+07**
F14	2.88E+06	7.72E+05	4.02E+06	2.54E+05	**1.58E+05**
F15	9.65E+05	1.34E+06	8.87E+05	1.20E+06	**6.88E+05**
F16	8.59E+02	1.51E+03	9.92E+02	**7.03E+02**	9.59E+02
F17	3.68E+02	5.18E+02	3.58E+02	3.12E+02	**3.04E+02**
F18	3.36E+06	2.22E+06	8.65E+08	8.55E+05	**7.60E+05**
F19	2.87E+06	6.52E+06	5.09E+07	2.05E+06	**1.50E+06**
F20	4.58E+02	6.16E+02	4.09E+02	4.89E+02	**4.09E+02**
F21	**2.78E+02**	4.83E+02	3.57E+02	3.31E+02	2.94E+02
F22	1.79E+03	3.89E+03	**1.45E+03**	2.21E+03	1.85E+03
F23	5.97E+02	6.02E+02	5.89E+02	4.91E+02	**4.73E+02**
F24	9.54E+02	8.90E+02	6.24E+02	**4.52E+02**	5.80E+02
F25	8.65E+02	7.11E+02	1.01E+03	5.27E+02	**4.97E+02**
F26	3.69E+03	4.15E+03	**1.98E+03**	2.45E+03	2.26E+03
F27	7.34E+02	7.27E+02	7.09E+02	6.59E+02	**5.68E+02**
F28	**2.36E+02**	8.68E+02	1.06E+03	6.68E+02	6.44E+02
F29	9.98E+02	1.59E+03	1.01E+03	9.68E+02	**9.17E+02**
F30	1.48E+07	2.86E+07	9.67E+07	9.98E+06	**8.44E+06**
A/B/C	21/6/2	29/0/0	21/5/3	18/7/4	

Table 7 Comparison results of IEGWO and other algorithms for spring design problem

Algorithm	Optimum decision vector			Optimum value	Rating
	d	*D*	*N*		
GA	0.050008	0.317611	14.013288	0.0127192	A
CPSO	0.052971	0.388339	9.650182	0.0126953	A
SMA	0.051853	0.306716	11.061309	0.0126663	A
FH-GWO	0.050966	0.339565	12.370964	0.0126756	A
IGWO	0.052787	0.383698	9.866711	0.0126874	A
EEGWO	0.052189	0.368867	10.610792	0.0126698	A
E-GWO	0.052811	0.384300	9.837956	0.0126879	A
IEGWO	0.051608	0.354771	11.404317	0.0126656	

and effective way to improve the accuracy and robustness of the models.

Support vector regression (SVR) is a classical machine learning algorithm, which has been widely used to solve regression prediction problems (Xu et al. 2016). The performance of SVR is mainly influenced by three parameters, namely penalty factor (c), kernel parameter (g), and insensitive loss function parameter (ε). In this section, the SVR model is used to establish a prediction model for Haugh unit (HU) values of eggs based on HSI

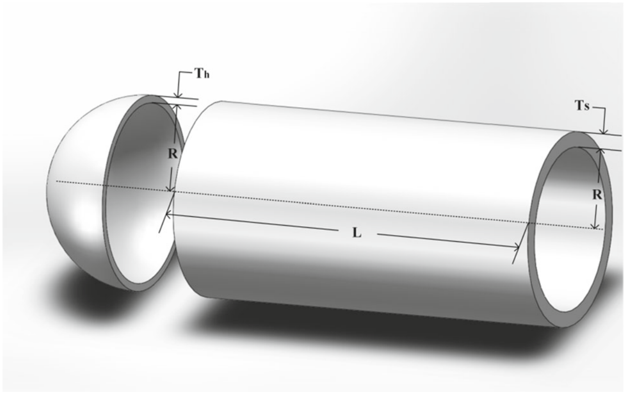


Fig. 8 Pressure vessel design problem

technique, and the proposed IEGWO is used to optimize the three parameters.

Figure 9 shows the reflectance spectral curves of 330 egg samples with different HU values in the wavelength range of 479–986 nm. The spectral reflectance is taken as the input of SVR and the HU value is taken as the output. As an example, the 330 egg samples are randomly divided into training set and test set according to the ratio of 3:1 to evaluate the performance of GA, PSO, GWO, and IEGWO.

In the process of parameter optimization, the fivefold cross-validation root mean square error (RMSECV) of training set is taken as the fitness function. The optimization range of parameters c, g, and ε are set to $[10^{-4}, 10^{3}]$, $[10^{-4}, 10^{3}]$ and $[10^{-4}, 10^{0}]$. The population size and the maximum iteration are set to 20 and 50.

Figure 10 shows the optimal fitness curves of IEGWO-SVR, GWO-SVR, PSO-SVR and GA-SVR models for 30 runs. As shown in Fig. 9, the IEGWO-SVR model has higher accuracy and faster convergence speed than the other three models.

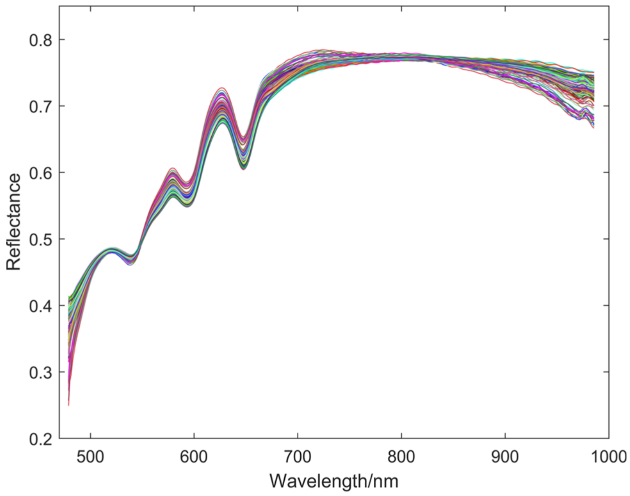


Fig. 9 Spectral curves of egg samples

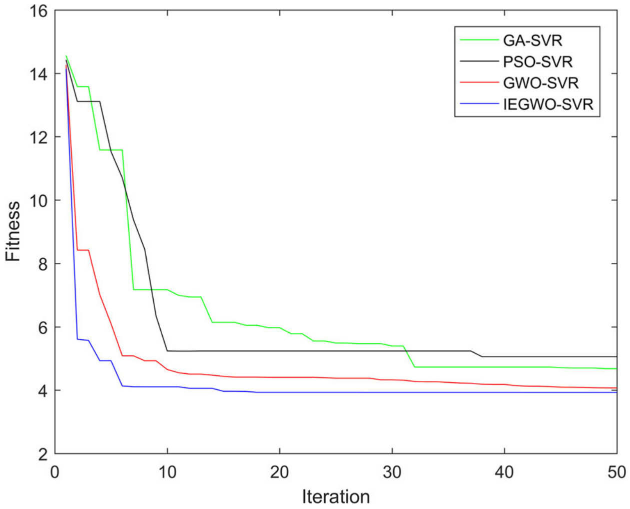


Fig. 10 Optimal fitness curves of GA-SVR, PSO-SVR, GWO-SVR, and IEGWO-SVR models

Table 8 Comparison results of IEGWO and other algorithms for pressure vessel design problem

Algorithm	Optimum decision vector				Optimum value	Rating
	TS	*TH*	*R*	*L*		
GA	0.8095	0.4361	41.9219	178.9994	6059.6894	A
CPSO	0.8052	0.4173	41.7064	181.6231	5996.2464	A
SMA	0.7828	0.3876	40.5602	196.6873	5895.3249	B
FH-GWO	0.7841	0.3881	45.6260	195.7812	5897.2074	A
IGWO	0.8113	0.4048	42.0324	177.4574	5956.7418	A
EEGWO	0.7911	0.4113	40.9830	190.9931	5969.7524	A
E-GWO	0.8195	0.4097	42.4577	172.2368	5974.8471	A
IEGWO	0.7783	0.3849	40.3261	199.9919	5886.3407	

6 Conclusion

This study proposes a modified version of GWO called information entropy-based GWO (IEGWO). In the IEGWO, an initial population generation method and a dynamic position update equation based on information entropy are proposed to maintain the population diversity. In addition, a nonlinear convergence strategy is applied to balance the exploration and exploitation. The IEGWO is tested on 10 well-known benchmark functions, CEC2014 and CEC2017, and compared with the other meta-heuristic algorithms. Two engineering design problems and one real-world parameter optimization problem in the field of hyperspectral imaging are also solved using IEGWO. The experimental results show that the IEGWO has better robustness and solution accuracy than the other compared algorithms.

Although the IEGWO is efficient, it also has some drawbacks that should be addressed. One is that the number of parameters is larger than the original GWO algorithm, and another is that the current study does not have automatic adjustment for the parameter k. In the future wok, we tend to study an adaptive parameter k to achieve the optimal optimization effect. In addition, we are going to investigate how to extend the IGWO algorithm to handle multi-objective optimization and combinatorial optimization problems.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s00500-022-07593-9.

Acknowledgements This work is partially supported by Postgraduate Research & Practice Innovation Program of Jiangsu Province (KYCX20_3033). Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD-2018-87). Six Talent Peaks Project in Jiangsu Province (ZBZZ-019).

Funding The authors have not disclosed any funding.

Data availability The data used to support the findings of this study are available from the corresponding author upon request.

Declarations

Conflict of interest The author declares that there is no conflict of interest in presenting this manuscript.

References

Adhikary J, Acharyya S (2022) Randomized balanced grey wolf optimizer (RBGWO) for solving real life optimization problems. Appl Soft Comput 117:108429

Arjenaki HG, Nadimi-Shahraki MH, Nourafza N (2015) A low cost model for diagnosing coronary artery disease based on effective features. Int J Electron Commun Comput Eng 6(1):93–97

Arora S, Singh S (2019) Butterfly optimization algorithm: a novel approach for global optimization. Soft Comput 23(3):715–734

Awad NH, Ali MZ, Liang JJ, Qu BY, Suganthan PN (2016) Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective real-parameter numerical optimization. Technical Report

Coello C, Montes EM (2002) Constraint-handling in genetic algorithms through the use of dominance-based tournament selection. Adv Eng Inform 16(3):193–203

Das SC, Manna AK, Rahman MS, Shaikh AA, Bhunia AK (2021) An inventory model for non-instantaneous deteriorating items with preservation technology and multiple credit periods-based trade credit financing via particle swarm optimization. Soft Comput 25(7):5365–5384

Duarte D, de Moura Oliveira PB et al (2020) Entropy based grey wolf optimizer. Springer International Publishing, Cham

Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. In: MHS'95. Proceedings of the sixth international symposium on micro machine and human science

Feng H, Grifoll M, Yang Z, Zheng P (2022) Collision risk assessment for ships'routeing waters: an information entropy approach with automatic identification system (AIS) data. Ocean Coast Manag 224:106184

Gupta S, Deep K (2019) An opposition-based chaotic grey wolf optimizer for global optimisation tasks. J Exp Theor Artif Intell 31(5):751–779

Gupta S, Deep K (2020) A memory-based grey wolf optimizer for global optimization tasks. Appl Soft Comput 93:106367

Hadavandi E, Mostafay S, Soltani P (2018) A grey wolf optimizer-based neural network coupled with response surface method for modeling the strength of siro-spun yarn in spinning mills. Appl Soft Comput 72:1–13

He Q, Ling W (2007) An effective co-evolutionary particle swarm optimization for constrained engineering design problems. Eng Appl Artif Intell 20(1):89–99

Hu X, Zhang S, Li M, Deng JD (2021) Multimodal particle swarm optimization for feature selection. Appl Soft Comput 113:107887

Kalemci EN, Kizler SB, Dede T, Angn Z (2020) Design of reinforced concrete cantilever retaining wall using grey wolf optimization algorithm. Structures 23:245–253

Karasu S, Saraç Z (2020) Classification of power quality disturbances by 2D-Riesz Transform, multi-objective grey wolf optimizer and machine learning methods. Digit. Signal Process 101:102711

Kumar N, Manna AK, Shaikh AA, Bhunia AK (2021) Application of hybrid binary tournament-based quantum-behaved particle swarm optimization on an imperfect production inventory problem. Soft Comput 25(16):11245–11267

Li S, Chen H, Wang M, Heidari AA, Mirjalili S (2020) Slime mould algorithm: a new method for stochastic optimization. Future Gener Comput Syst 111:300–323

Liang J, Qu B, Suganthan, P (2013) Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. In: Computational intelligence laboratory, Zhengzhou

Long W, Liang X, Cai S, Jiao J, Zhang W (2017) A modified augmented lagrangian with improved grey wolf optimization to constrained optimization problems. Neural Comput Appl 28(1):421–438

Long W, Jiao J, Liang X, Tang M (2018a) Inspired grey wolf optimizer for solving large-scale function optimization problems. Appl Math Model 60:112–126

Long W, Jiao J, Liang X, Tang M (2018b) An exploration-enhanced grey wolf optimizer to solve high-dimensional numerical optimization. Eng Appl Artif Intell 68:63–80

Luo K, Zhao Q (2019) A binary grey wolf optimizer for the multidimensional knapsack problem. Appl Soft Comput 83:105645

Manna AK, Bhunia AK (2022) Investigation of green production inventory problem with selling price and green level sensitive interval-valued demand via different metaheuristic algorithms. Soft Comput. https://doi.org/10.1007/s00500-022-06856-9

Manna AK, Akhtar M, Shaikh AA, Bhunia AK (2021) Optimization of a deteriorated two-warehouse inventory problem with all-unit discount and shortages via tournament differential evolution. Appl Soft Comput 107:107388

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

Mohakud R, Dash R (2022) Skin cancer image segmentation utilizing a novel EN-GWO based hyper-parameter optimized FCEDN. J King Saud Univ Comput Inf Sci. https://doi.org/10.1016/j.jksuci.2021.12.018

Naserbegi A, Aghaie M (2021) Exergy optimization of nuclear-solar dual proposed power plant based on GWO algorithm. Prog Nucl Energy 140:103925

Qu C, Gai W, Zhong M, Zhang J (2020) A novel reinforcement learning based grey wolf optimizer algorithm for unmanned aerial vehicles (UAVs) path planning. Appl Soft Comput 89:106099

Rodríguez L, Castillo O, Soria J, Melin P, Valdez F, Gonzalez CI, Martinez GE, Soto J (2017) A fuzzy hierarchical operator in the grey wolf optimizer algorithm. Appl Soft Comput 57:315–328

Samuel OD, Okwu MO, Oyejide OJ, Taghinezhad E, Afzal A, Kaveh M (2020) Optimizing biodiesel production from abundant waste oils through empirical method and grey wolf optimizer. Fuel 281:118701

Saxena A, Kumar R, Mirjalili S (2020) A harmonic estimator design with evolutionary operators equipped grey wolf optimizer. Expert Syst Appl 145:113125

Shannon CE (1948) The mathematical theory of communication. Bell Syst Tech J 27(3):373–423

Suktanarak S, Teerachaichayut S (2017) Non-destructive quality assessment of hens' eggs using hyperspectral images. J Food Eng 215:97–103

Sundaramurthy S, Jayavel P (2020) A hybrid grey wolf optimization and particle swarm optimization with C4.5 approach for prediction of rheumatoid arthritis. Appl Soft Comput 94:106500

Tawhid MA, Ali AF (2017) A hybrid grey wolf optimizer and genetic algorithm for minimizing potential energy function. Memet Comput 9(4):347–359

Venkataraman NL, Kumar R, Shakeel PM (2020) Ant lion optimized bufferless routing in the design of low power application specific network on chip. Circuits Syst Signal Process 39(2):961–976

Wang L, Zheng X, Wang S (2013) A novel binary fruit fly optimization algorithm for solving the multidimensional knapsack problem. Knowl-Based Syst 48:17–23

Wolpert D, Macready W (1997) No free lunch theorems for optimization. IEEE Trans Evol Comput 1(1):67–82

Xu J, Riccioli C, Sun D (2016) Development of an alternative technique for rapid and accurate determination of fish caloric density based on hyperspectral imaging. J Food Eng 190:185–194

Xue J, Shen B (2020) A novel swarm intelligence optimization approach: sparrow search algorithm. Syst Sci Control Eng 8(1):22–34

Yao K, Sun J, Chen C, Xu M, Zhou X, Cao Y, Tian Y (2022) Non-destructive detection of egg qualities based on hyperspectral imaging. J Food Eng 325:111024

Zamani H, Nadimi-Shahraki MH, Gandomi AH (2019) CCSA: conscious neighborhood-based crow search algorithm for solving global optimization problems. Appl Soft Comput 85:105583

Zareie A, Sheikhahmadi A, Jalili M (2020) Identification of influential users in social network using gray wolf optimization algorithm. Expert Syst Appl 142:112971

Zhang S, Zhou Y, Li Z, Pan W (2016) Grey wolf optimizer for unmanned combat aerial vehicle path planning. Adv Eng Softw 99:121–136

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.