METHODOLOGIES AND APPLICATION

Improved grey wolf optimization based on the two-stage search of hybrid CMA-ES

Yun-tao Zhao[1,2] · Wei-gang Li[1,2] · Ao Liu[3]

Abstract

Hybrid algorithms with different features are an important trend in algorithm improvement. In this paper, an improved grey wolf optimization based on the two-stage search of hybrid covariance matrix adaptation-evolution strategy (CMA-ES) is proposed to overcome the shortcomings of the original grey wolf optimization that easily falls into the local minima when solving complex optimization problems. First, the improved algorithm divides the whole search process into two stages. In the first stage, the improved algorithm makes full use of the global search ability of grey wolf optimization on a large scale and thoroughly explores the location of the optimal solution. In the second stage, due to CMA-ES having a strong local search capability, the three CMA-ES instances use the α wolf, β wolf and δ wolf as the starting points. In addition, these instances have different step size for parallel local exploitations. Second, in order to make full use of the global search ability of the grey wolf algorithm, the Beta distribution is used to generate as much of an initial population as possible in the non-edge region of the solution space. Third, the new algorithm improves the hunting formula of the grey wolf algorithm, which increases the diversity of the population through the interference of other individuals and reduces the use of the head wolf's guidance to the population. Finally, the new algorithm is quantitatively evaluated by fifteen standard benchmark functions, five test functions of CEC 2014 suite and two engineering design cases. The results show that the improved algorithm significantly improves the convergence, robustness and efficiency for solving complex optimization problems compared with other six well-known optimization algorithms.

Keywords Grey wolf optimization · CMA-ES · Function optimization · Hybrid algorithm · Two-stage search

1 Introduction

For many complex optimization problems, the traditional methods based on calculus (such as the Newton method and conjugate gradient method) have difficulty finding the optimal solution because of the lack of differentiability.

✉ Yun-tao Zhao
zhyt@wust.edu.cn

1 School of Information Science and Engineering, Wuhan University of Science and Technology, No. 947, Heping Road, Qingshan District, Wuhan 430081, Hubei, China

2 Engineering Research Center for Metallurgical Automation and Measurement Technology of Ministry of Education, Wuhan 430081, China

3 School of Management, Wuhan University of Science and Technology, Wuhan 430081, China

Intelligent optimization methods, such as particle swarm optimization (PSO) (Eberhart and Kennedy 2002), the differential evolution algorithm (DE) (Storn and Price 1997), the fruit fly algorithm (FOA) (Pan 2012), the multi-verse optimizer (MVO) (Mirjalili et al. 2015), the whale optimization algorithm (WOA) (Mirjalili and Lewis 2016) and other intelligent optimization methods, have significant advantages in solving complex optimization problems. These intelligent optimization algorithms are based on natural heuristics and do not have any special requirements for the differentiability, continuity or convexity of the problem. Therefore, these intelligent optimization algorithms have been widely used (Peng et al. 2013; Chen et al. 2018).

The common goal of these algorithms is to find the best quality solutions and better convergence performance. To do this, intelligent optimization algorithms should be equipped with exploration and exploitation to ensure that they find the global optimum (Mirjalili et al. 2012; Chakri

🙆 Springer

et al. 2017). Intelligent optimization algorithms can be broadly classified into three categories: constructive algorithms, improvement algorithms and hybrid algorithms (Nagano and Moccellin 2002). Although many research achievements have been achieved for the intelligent optimization algorithm, it still easily falls into the local optimal and has low precision in solving complex problems.

Considering the restrictiveness of the constructive algorithms and the improvement algorithms, recently, the research concentration has expanded to hybrid algorithms instead of a sole intelligent algorithm. The purpose of hybrid algorithms is to synthesize the advantages of different algorithms (Aydilek 2018). The no free lunch theorem has shown that all optimization algorithms have the same average performance in the mathematical sense (Wolpert and Macready 1997). Therefore, each algorithm has its scope of application. An important purpose of the hybrid algorithm is to make use of the unique features and advantages of each algorithm and to obtain the best solution in the largest problem domain. It has become evident that the hybrid of two or more intelligent algorithms is the most efficient approach and can have good application in dealing with the real-world engineering problems (Qiu et al. 2017; Chi et al. 2017; Anand and Suganthi 2018; Lin and Chiu 2018). The evolutionary computing and hybrid algorithm research expert, Raidl, believes that (Raidl 2006) choosing an appropriate hybrid algorithm is necessary to solve the most complex problems efficiently.

Grey wolf optimization (GWO), proposed by Mirjalili et al. (2014) in 2014, is a new intelligent optimization algorithm, which simulates the social hierarchy and hunting behaviors of grey wolves. In numerical optimization, the GWO algorithm (Mirjalili et al. 2014) and its variants (Saxena et al. 2018; Gupta and Deep 2018a, b, c; Gupta and Deep 2019) have been proven to be superior to the particle swarm optimization, the differential evolution algorithm and the gravitational search algorithm, etc., and has been successfully applied to solving economic dispatch problems (Kamboj et al. 2016; Pradhan et al. 2016; Venkatakrishnan et al. 2018), feature selection (Emary et al. 2015; Yamany et al. 2016; Medjahed et al. 2016; Daniel et al. 2017) and wireless sensor networks (Sujitha and Baskaran 2017). However, similar to other intelligent optimization algorithms, the GWO algorithm overemphasizes the global search. Therefore, it has lower search precision when optimizing some complex functions.

The covariance matrix adaptation-evolution strategy (CMA-ES) (Hansen 2006) is an excellent stochastic method for the continuous domain of nonlinear, non-convex functions. During the exploitation of the parameter space, CMA-ES attempts to learn a second-order model of the underlying objective function. However, the CMA-ES algorithm is a local search algorithm in essence that has a strong local search capability but is relatively weak in global search capability (Preuss 2010; Melo and Iacca 2014; Xu et al. 2017).

Along with the developmental trend of the hybrid algorithms discussed above, this paper proposes an improved grey wolf optimization based on the two-stage search of the hybrid CMA-ES (CMAGWO). The CMAGWO algorithm is divided into two stages in order to make up for the disadvantage of the slow convergence speed of GWO and the weak global search ability of the CMA-ES. In the first stage, the GWO algorithm, with a strong global search capability, conducts a large-scale search in as many possible locations of the region of optimal solution. In the second stage, three CMA-ES instances are generated for the local fine exploitation in parallel. The α wolf, β wolf and δ wolf, which are obtained in the first stage, are used as the starting point of the CMA-ES. In addition, each CMA-ES instance concurrently explores different regions of the search space and exchanges information about its status. Furthermore, in order to use the global search ability of the grey wolf algorithm, the population of GWO uses the Beta distribution in the initialization process. This allows the initial solution to be thoroughly generated in the non-edge region of the solution space. In the new algorithm, the process of wolves hunting in the original GWO is improved. The diversity of the population is increased by the interference of other individuals, and the absolute guidance of the head wolves to each individual is reduced.

The quantification evaluations of the algorithm are performed with fifteen standard benchmark functions, five test functions of CEC 2014 suite and two engineering design cases. The results show that the convergence accuracy, convergence speed and search robustness of CMAGWO have been significantly improved in solving complex problems over other approaches.

The rest of the paper is organized as follows. Section 2 briefly reviews the fundamentals of GWO and CMA-ES. The proposed CMAGWO algorithm is explained in Sect. 3. The experimental results are demonstrated in Sect. 4, comparing the performance of CMAGWO with other algorithms. Section 5 solves the two engineering optimization cases. Finally, Sect. 6 concludes the paper.

2 The original GWO and original CMA-ES

In this section, the features of the original GWO and the original CMA-ES are discussed.

2.1 The original grey wolf optimization

Grey wolves are considered apex predators, meaning that they are at the top of the food chain. They have a very strict social hierarchy. Grey wolf optimization is inspired by the grey wolf's social hierarchy and hunting behavior. Compared with other evolutionary algorithms, GWO improves the optimization efficiency of the algorithm using swarm intelligence, which is based on the behavior of grey wolves within a social hierarchy when hunting prey. To mathematically model the social hierarchy of wolves, we consider the fittest solution as the α. Consequently, the second and third best solutions are named β and δ, respectively. The rest of the candidate solutions are assumed to be ζ. In the original GWO, the hunting is guided by α, β and δ. The ζ wolves follow these three wolves.

When grey wolf optimization is used to solve the continuous problem, it is assumed that the population of grey wolves is m and the dimension of each grey wolf is n. The position x_i of wolf i in the n-dimensional space can be expressed as follows:

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{in}), \quad i = 1, \ldots, m \tag{1}$$

The position x_p of the prey is the global optimal solution. During the hunting process, grey wolves encircle the prey according to the following formula:

$$D = |\xi \cdot x_p(g) - x(g)| \tag{2}$$

In the formula, g is the current number of iterations. $x_p(g)$ indicates the position of the prey when the algorithm iterates to the g th iteration, and $x(g)$ represents the position of the wolf in the gth iteration. ξ is the coefficient:

$$\xi = 2r_1 \tag{3}$$

where r_1 is a random number in the range [0, 1].

When the grey wolves chase the prey, the positions of the grey wolves are updated according to Eq. (4).

$$x(g + 1) = x_p(g) - A \cdot D \tag{4}$$

$$A = 2a \cdot r_2 - a \tag{5}$$

where A is the convergence factor and r_2 is a random number in the range [0, 1].

According to Eq. (6), a decreases linearly from 2 to 0 with the increase in the iterations. Max_iter is the maximum number of iterations.

$$a = 2 - \frac{2g}{\text{Max\_iter}} \tag{6}$$

Grey wolves have the ability to recognize the location of prey and encircle them, but in the optimization process, the position x_p of prey is unknown. To mathematically simulate the hunting behavior of grey wolves, we suppose that the α wolf, β wolf and δ wolf have better knowledge about the potential location of prey. Therefore, we save the first three best solutions obtained so far and oblige the other search agents to update their positions according to the position of the best search agents. The hunting formula follows Eqs. (7)–(13):

$$D_\alpha = |\xi_1 \cdot x_\alpha(g) - x(g)| \tag{7}$$

$$D_\beta = |\xi_2 \cdot x_\beta(g) - x(g)| \tag{8}$$

$$D_\delta = |\xi_3 \cdot x_\delta(g) - x(g)| \tag{9}$$

$$x_1(g) = x_\alpha(g) - A_1 \cdot D_\alpha \tag{10}$$

$$x_2(g) = x_\beta(g) - A_2 \cdot D_\beta \tag{11}$$

$$x_3(g) = x_\delta(g) - A_3 \cdot D_\delta \tag{12}$$

$$x(g + 1) = \frac{x_1(g) + x_2(g) + x_3(g)}{3} \tag{13}$$

According to Eqs. (7)–(12), we can calculate the distance between the grey wolf and the positions x_α, x_β and x_δ of the α wolf, β wolf and δ wolf, respectively. According to Eq. (13), the direction of the individual moving toward the prey can be determined.

2.2 The original CMA-ES

Here, we consider a standard CMA-ES with a weighted intermediate recombination, step-size adaptation, and a combination of rank-μ update and rank-one update (Hansen 2006). At each iteration of the algorithm, the members of the new population are sampled from a multivariate normal distribution with the mean (mean $\in R^n$) and covariance ($C \in R^{n \times n}$). n is the dimension. The sampling radius is controlled by the overall standard deviation (step size) θ. Let $x(g)$ represent the individual at generation g.

The CMA-ES optimization algorithm is approaching the optimal solution by repeatedly performing sampling, selection and recombination, updating parameters and so on. The algorithm stops and outputs the result when the maximum number of iterations is reached or the precision is met. The main operators of the CMA-ES algorithm are as follows:

1. Sampling

   In the CMA-ES, the population of the new search is generated by a multivariate normal distribution. The sampling formula is as follows:

$$x(g + 1) = \text{mean}^g + \theta^g \cdot N(0, C^g) \tag{14}$$

   where $N(0, C^g)$ is a multivariate normal distribution with zero mean and the covariance matrix (C^g). g is the current number of iterations. mean^g is the center point of the population, and θ^g is the step size.

2. Selection and recombination The new mean (mean^{g+1}) of the search distribution is a weighted average of the selected points μ.

$$\text{mean}^{g+1} = \sum_{i=1}^{\mu} \omega_i x_{i;m}(g+1) \tag{15}$$

where ω_i is the weight value, $\sum_{i=1}^{\mu} \omega_i = 1$, and $\omega_1 \geq \omega_2 \geq \cdots \geq \omega_\mu > 0$. The optimal individual assigns a larger weight value. $x_{i;m}(g+1)$ is the individual (i) after the sorting of the population (m) in the g th iteration.

3. Updating parameters

   1. Adapting the covariance matrix

$$C^{g+1} = (1 - c_1 - c_\mu)C^g + c_1 p_c^{g+1}(p_c^{g+1})^T$$
$$+ c_\mu \sum_{i=1}^{\mu} \omega_i y_{i;m}(g+1)(y_{i;m}(g+1))^T \tag{16}$$

where p_c^{g+1} is the evolution path of covariance matrix (C^{g+1}) in the $g + 1$th iteration. c_1 is the learning rate for the rank-one update. c_μ is the learning rate for the rank-μ update.

$$p_c^{g+1} = (1 - c_c)p_c^g$$
$$+ \sqrt{c_c(2 - c_c)\mu_{\text{eff}}} \frac{\text{mean}^{g+1} - \text{mean}^g}{\theta^g} \tag{17}$$

$$y_{i;m}(g+1) = \frac{x_{i;m}(g+1) - \text{mean}^g}{\theta^g} \tag{18}$$

where c_c is decay rate for the evolution path (p_c^{g+1}). μ_{eff} is the variance effective selection mass, and $1 \leq \mu_{\text{eff}} \leq \mu$.

   2. Step-size adaptation

$$\theta^{g+1} = \theta^g \exp\left(\frac{c_\theta}{d_\theta} \left(\frac{\left\| p_\theta^{g+1} \right\|}{E\|N(0,I)\|} - 1 \right) \right) \tag{19}$$

where p_θ^{g+1} is the evolution path of step size (θ^{g+1}) in the $g + 1$th iteration. $E\|N(0,I)\|$ is the expectation of the Euclidean norm of $N(0,I)$. I is the unit matrix. c_θ is the decay rate of the evolutionary path p_θ^{g+1}. $d_\theta \approx 1$ is the damping parameter.

$$p_\theta^{g+1} = (1 - c_c)p_\theta^g$$
$$+ \sqrt{c_\theta(2 - c_\theta)\mu_{\text{eff}}} C^{-\sqrt{g}} \frac{\text{mean}^{g+1} - \text{mean}^g}{\theta^g} \tag{20}$$

The CMA-ES algorithm completes the optimization process through the above steps. The initial search point is given or randomly generated and then randomly generates the next-generation population centered on the initial point of a certain probability density. Then, the evolutionary strategy parameters are updated to adjust the evolutionary direction, and finally, the optimal solution is achieved.

3 Improved GWO based on the two-stage search of hybrid CMA-ES

3.1 Population initialization

The advantage of the intelligent optimization algorithm is that the group members cooperate with each other and use the guidance of a certain mechanism to approach the optimal solution from the initial position. Therefore, the initial distribution of the population affects the search efficiency of the algorithm. The method of randomly generating the initial individual based on uniform distribution has the advantage of simplicity. However, in actual cases, the optimal solution is less located on the edge of the search space. It is more desirable to generate as many individuals as possible in the non-edge region and generate fewer individuals at the edge of the solution space. The Beta distribution in the stochastic process reflects this distributional characteristic very well (Klein et al. 2016).

$$\text{Beta}(x) = \frac{x^{u-1}(1-x)^{v-1}}{B(u,v)}, 0 < x < 1 \tag{21}$$

where the denominator is the Beta function, which can be defined as follows:

$$B(u,v) = \int_0^1 t^{u-1}(1-t)^{v-1} \mathrm{d}t \tag{22}$$

when $u = v = 1.2$, the graph of the Beta distribution is as follows:
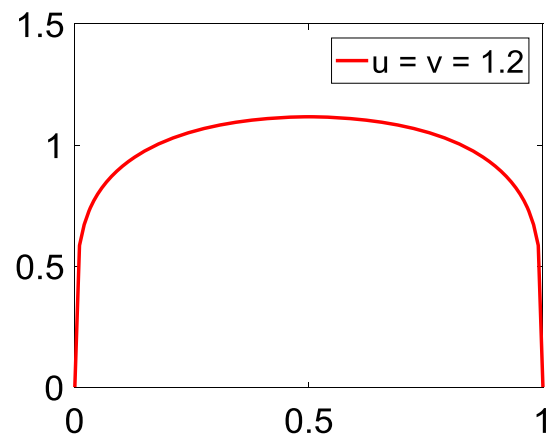

Fig. 1 The graph of the Beta distribution

Figure 1 shows that the initial individuals generated by the Beta distribution are mainly distributed in the non-edge regions. The population generated by this distribution can appear near the target area with a greater probability. Thus, the search efficiency is improved. The Beta distribution conforms to the desired initial distribution of the population.

3.2 Global guidance of the GWO for hunting

Over the course of the iterations, the α wolf, the β wolf and the δ wolf of the original GWO estimate the probable position of the prey. Each candidate solution updates its distance from the prey.

In the improved algorithm of the hybrid CMA-ES with the two-stage search, the GWO algorithm mainly plays the role of global search ability. Therefore, during the execution of the algorithm, the hunting process of the original grey wolf algorithm is improved. The overall guidance ability is increased to improve the diversity of the population by reducing the use of the head wolf's guidance to each individual.

$$D_\alpha = |\xi_1 \cdot x_\alpha(g) - x(g)| \tag{23}$$

$$D_\beta = |\xi_2 \cdot x_\beta(g) - x(g)| \tag{24}$$

$$D_\delta = |\xi_3 \cdot x_\delta(g) - x(g)| \tag{25}$$

In the hunting formula of the original GWO, the positions of wolves are guided by the positions x_α, x_β and x_δ of the α wolf, β wolf and δ wolf, respectively. In the new algorithm, the hunting formula is improved to guide the population optimization according to the distance between the grey wolves and the head wolf, thus improving the global generalization ability of the algorithm.

$$x_1(g) = a \cdot x_1'(g) - A_1 \cdot |\xi_1 \cdot x_\alpha(g) - x(g)| \tag{26}$$

$$x_2(g) = a \cdot x_2'(g) - A_2 \cdot |\xi_2 \cdot x_\beta(g) - x(g)| \tag{27}$$

$$x_3(g) = a \cdot x_3'(g) - A_3 \cdot |\xi_3 \cdot x_\delta(g) - x(g)| \tag{28}$$

$$x(g+1) = \frac{x_1(g) + x_2(g) + x_3(g)}{3} \tag{29}$$

where $x_1'(g)$, $x_2'(g)$ and $x_3'(g)$ are the grey wolf individuals, which are different from $x(g)$ in the population. As the number of iterations increases, a decreases linearly from 2 to 0. In the early stage of searching, $x_1'(g)$, $x_2'(g)$ and $x_3'(g)$ play major roles in the generation of new individuals. a gradually changes to 0 in the search process in Eq. (6). Therefore, the weights of $x_1'(g)$, $x_2'(g)$ and $x_3'(g)$ are gradually reduced.

3.3 The hybrid CMAGWO procedure

CMA-ES is suitable for strong nonlinear and non-convex problems in continuous domains and has a strong local search capability (Wang et al. 2016). By improving the initial distribution of the original grey wolf algorithm and the global guidance of the hunting process, the improved grey wolf optimization can better realize the global search.

Therefore, the CMAGWO algorithm obtained by the hybrid design in the first stage uses the improved GWO algorithm for global search, guides the entire population to optimize it and detects as many new solutions as possible within the global scope. With the algorithm running, the search performance of the improved GWO algorithm is reduced to a certain threshold value. The CMAGWO algorithm transforms from the global detection as the main target to the local refined mining search. In the local refined search stage, the head wolf's guidance ability is exploited. The positions x_α, x_β and x_δ of the three head wolves α, β and δ obtained from the improved GWO algorithm are used as the starting points of this stage. The CMAGWO algorithm generates three CMA-ES instances by performing a parallel search (instance CMA-ES-α, instance CMA-ES-β and instance CMA-ES-δ) in this stage. The starting points of the three instances of CMA-ES (instance CMA-ES-α, instance CMA-ES-β and instance CMA-ES-δ) can be described as follows:

Instance CMA-ES-α:

$$mean\_\alpha = x_\alpha \tag{30}$$

Instance CMA-ES-β:

$$mean\_\beta = x_\beta \tag{31}$$

Instance CMA-ES-δ:

$$mean\_\delta = x_\delta \tag{32}$$

The initial step size is the ratio of the Euclidean distance between the instance center and the population center to the Euclidean distance of all individual and population center.

Instance CMA-ES-α:

$$\theta\_\alpha = \frac{d|x_\alpha, \bar{x}|}{d|x_1, \bar{x}| + d|x_2, \bar{x}| + \cdots + d|x_i, \bar{x}| + \cdots + d|x_m, \bar{x}|} \tag{33}$$

Instance CMA-ES-β:

$$\theta\_\beta = \frac{d|x_\beta, \bar{x}|}{d|x_1, \bar{x}| + d|x_2, \bar{x}| + \cdots + d|x_i, \bar{x}| + \cdots + d|x_m, \bar{x}|} \tag{34}$$

Instance CMA-ES-δ:

$$\theta\_\delta = \frac{d|x_\delta, \bar{x}|}{d|x_1, \bar{x}| + d|x_2, \bar{x}| + \cdots + d|x_i, \bar{x}| + \cdots + d|x_m, \bar{x}|} \quad (35)$$

where

$$d|x_i, \bar{x}| = \sqrt{\sum_{j=1}^{n} (x_{ij} - \bar{x}_j)^2} \quad (36)$$

$$\bar{x} = (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_j, \ldots, \bar{x}_n), \quad 1 \le j \le n \quad (37)$$

$$\bar{x}_j = \frac{1}{m}(x_{1j} + x_{2j} + \cdots + x_{ij} + \cdots + x_{mj}), \quad 1 \le j \le n \quad (38)$$

The CMAGWO algorithm sorts the population of each instance and generates the instances' center points of the next generation using the weighted average of the previous μ individuals. In the second stage, the search coverage of instance CMA-ES-α, instance CMA-ES-β and instance CMA-ES-δ is as far as possible to avoid overlapping. Therefore, different scaling factors are used on the three instances: 1, 0.1 and 0.01. The CMAGWO algorithm obtains the new generation in the second stage according to Eqs. (39)–(41).

$$x\_\alpha(g+1) = mean\_\alpha^g + 1.0 * \theta\_\alpha^g \cdot N(0, C\_\alpha^g) \quad (39)$$

$$x\_\beta(g+1) = mean\_\beta^g + 0.1 * \theta\_\beta^g \cdot N(0, C\_\beta^g) \quad (40)$$

$$x\_\delta(g+1) = mean\_\delta^g + 0.01 * \theta\_\delta^g \cdot N(0, C\_\delta^g) \quad (41)$$

The value of the means (mean$\_\alpha^{g+1}$, mean$\_\beta^{g+1}$, and mean$\_\delta^{g+1}$), the covariance matrixes (C_α^{g+1}, C_β^{g+1}, and C_δ^{g+1}) and the step sizes (θ_α^{g+1}, θ_β^{g+1}, and θ_δ^{g+1}) are updated according to Eqs. (15)–(20). Each instance independently searches the solution space from the starting point in parallel. In the CMAGWO, the population uses the data exchange mechanism to exchange information and shares the excellent individuals. Individuals evolve in the different environments for the three instances and perform fine mining in their respective local areas.

3.4 Pseudo-code of the proposed algorithm

The proposed CMAGWO can be given as follows:

- *Step 1* Initializing the algorithm parameters.
  The maximum number of iterations is Max_iter. a, A, ξ, c_θ, d_θ, c_c, μ_{eff} and other parameters are generated. Let $g = 1$.
- *Step 2* In the search space, the grey wolf's m individuals are generated using the Beta distribution to form the initial population.

- *Step 3* Calculate the objective function value of each grey wolf and denote the positions as x_α, x_β, and x_δ for the best three grey wolves.
- *Step 4* Calculate the distance between the position of the grey wolf and the positions x_α, x_β and x_δ according to Eqs. (23)–(25).
- *Step 5* Update the position of each grey wolf according to Eqs. (26)–(29).
- *Step 6* When the performance of the improved GWO is reduced to a certain threshold, instance CMA-ES-α, instance CMA-ES-β and instance CMA-ES-δ are generated centered on x_α, x_β and x_δ according to Eqs. (30)–(32). The initial step size is generated by Eqs. (33)–(35). Otherwise, go to Step 3.
- *Step 7* The three CMA-ES instances are sampled according to Eqs. (39)–(41).
- *Step 8* Instance CMA-ES-α, instance CMA-ES-β and instance CMA-ES-δ perform the selection and recombination operations.
- *Step 9* Update the mean (mean^{g+1}), covariance matrix (C^{g+1}) and step size (θ^{g+1}) according to Eqs. (15)–(20).
- *Step 10* If the accuracy is met or the maximum number of iterations is reached, the algorithm ends and the optimal solution is output; otherwise, Step 11 is performed.
- *Step 11* $g = g + 1$, and return to Step 7.

3.5 Computational complexity

The computational complexity of an optimization algorithm is a key metric for evaluating the run time of an algorithm. The computational complexity of GWO and CMAGWO depends on the number of wolves m in pack, dimensions of the problem n and maximum number of iterations Max_iter. By analyzing the steps of algorithms, the computational complexity of the CMAGWO and original GWO is defined as follows according to Eqs. (42), (43):

$$O(GWO) = O(Max\_iter \times m \times n) \quad (42)$$

$$O(CMAGWO) = O(iter1 \times m \times n + 3 \times (Max\_iter - iter1) \times n) \quad (43)$$
$$= O(iter1 \times m \times n) + O((Max\_iter - iter1) \times n)$$

where iter1 represents the iteration numbers of CMAGWO in the first stage. The computational complexity of CMA-ES is $O(n)$ of each iteration (Hansen 2006).

4 Experimental verification and analysis

To verify the performance of the CMAGWO algorithm, fifteen standard benchmark functions and five test functions of CEC 2014 suite are used to test it. These functions are typical complex test functions. Moreover, the traditional methods based on calculus are difficult to obtain better results. All computational experiences for the test functions are implemented using MATLAB R2016b on a PC with an Intel core i5-2410 4.0 GHz processor and 8.0 GB memory.

The mathematical definitions and other relevant details of fifteen standard benchmark functions such as domains of variable and dimensions of the function are given in Table 1. Generally speaking, the standard benchmark functions are minimization functions and include three types: unimodal (F_1, F_7, F_8, F_9, F_{10}), multimodal (F_2, F_3, F_5, F_6, F_{15}) and fixed-dimension multimodal (F_4, F_{11}, F_{12}, F_{13}, F_{14}).

4.1 The diversity measure of GWO and CMAGWO

The exploration and exploitation are the two important diversity characteristics of population. A large value of diversity implies more exploration while low implies more exploitation. The average distance is defined as the average of distances of all individuals from the population center. This diversity measure is given in Olorunda and Engelbrecht (2008). Three standard benchmark functions (unimodal function F_1, multimodal function F_2 and fixed-dimension multimodal function F_4) are taken into consideration for the average distance to quantify the diversity measure of individuals in the GWO and CMAGWO.

In all experiments, the values of the common parameters used in each algorithm, such as the population size and the total iteration number, are chosen to be the same. For all algorithms, the population size is set as $m = 50$ and the total number of iterations is set as Max_iter = 300. To reduce the random error of the simulation, all experiments on each test function are repeated *15* runs.

A low value of the average distance shows population convergence around the population center, while a high value shows large dispersion of individuals from the population center. Figure 2 shows that the average distance of CMAGWO is large than GWO before 46th iteration. The average distance of CMAGWO is less than GWO after 46th iteration due to hybrid three CMA-ES instances. Figure 3 shows that the turning point of evolutionary iterations is 22th iteration. The turning point is 17th iteration in Fig. 4. The CMAGWO retains high diversity in early stage of the search process and proportionally decreases diversity as search progresses. It is observed that the average distance is decreased gradually over the course

of iteration, two-stage search that guarantees transition between exploration and exploitation. It is clear from the results of average distance that the exploration ability and exploitation ability of the CMAGWO based on the two-stage search of hybrid CMA-ES are better than the GWO.

4.2 Performance on the standard benchmark functions

In order to analyze the convergence speed and accuracy, the CMAGWO algorithm is compared with GWO, CMA-ES, MVO, WOA, PSOGWO (which is the hybrid of GWO and PSO) (Singh and Singh 2017), and DEGWO (which is the hybrid of GWO and DE) (Zhu et al. 2015). The algorithms and other specific parameter settings are given as follows: The GWO algorithm is the same as the parameter setting in reference (Mirjalili et al. 2014), CMA-ES refers to (Hansen 2006), MVO refers to (Mirjalili et al. 2015), WOA refers to Mirjalili and Lewis (2016), PSOGWO refers to Singh and Singh (2017), and DEGWO refers to Zhu et al. (2015). The experimental environment and the iteration numbers of algorithms are the same as above.

In practical applications, the efficiency of the algorithm is given more attention while the convergence is ensured. To compare the convergence efficiency of these algorithms, Table 2 shows the convergence algebra of each algorithm on the standard benchmark functions. (The iteration number is recorded as the convergence algebra when the MEAN value is less than 10E−3.)

The CMA-ES performs well on some functions. The results show that the CMAGWO algorithm is excellent on unimodal (F_1, F_7, F_9, F_{10}), multimodal (F_2, F_3, F_5, F_6, F_{15}) and fixed-dimension multimodal (F_4, F_{11}, F_{12}, F_{14}), converges to the global optimal solution at a faster speed, and has good optimization efficiency. Even for function F_8 and function F_{13}, the CMAGWO algorithm is better than other algorithms.

The detailed computational data of all test functions for these algorithms will be presented. The best objective function value (BEST), the worst objective function value (WORST), the mean value (MEAN) and the standard deviation (STD) of all algorithms are given. These indices are used to evaluate the accuracy of the algorithm (the ability to jump out of the local optimum) and the robustness. Tables 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 and 17 show the BEST, the WORST, the MEAN and the STD of seven algorithms on each standard benchmark function. To maintain a fair competitive environment, the stopping criterion for all the algorithms is maximum iteration. Tables 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 and 17 and Figs. 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18 and 19 show the performance results of the algorithms.

Table 1 The standard benchmark functions

| No.	Function	Dimension	Domain	Minimum
F_1	$f(x) = \sum_{i=1}^{n} i x_i^4 + \text{random}[0, 1)$	30	$[-1.28, 1.28]$	0
F_2	$f(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	$[-600, 600]$	0
F_3	$f(x) = 0.1\left\{\sin^2(3\pi x_1) + \sum_{i=1}^{n}(x_i - 1)^2[1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2[1 + \sin^2(2\pi x_n)]\right\}$ $+ \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	30	$[-50, 50]$	0
F_4	$f(x) = \sum_{i=1}^{11}\left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}\right]^2 - 0.0003$	4	$[-5, 5]$	0
F_5	$f(x) = \sum_{i=1}^{n}[x_i^2 - 10\cos(2\pi x_i) + 10]$	30	$[-5.12, 5.12]$	0
F_6	$f(x) = \frac{\pi}{n}\left\{10\sin(\pi(1 + \frac{x_1 + 1}{4})) + \sum_{i=1}^{n-1}\frac{(x_i + 1)^2}{4}\left[1 + 10\sin^2\left(\pi\left(1 + \frac{x_{i+1} + 1}{4}\right)\right)\right] + \left(\frac{x_n + 1}{4}\right)^2\right\}$ $+ \sum_{i=1}^{n} u(x_i, 10, 100, 4)$	30	$[-50, 50]$	0
F_7	$f(x) = \sum_{i=1}^{n}\lfloor x_i + 0.5 \rfloor^2$	30	$[-100, 100]$	0
F_8	$f(x) = \sum_{i=1}^{n-1}\left[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2\right]$	30	$[-30, 30]$	0
F_9	$f(x) = \max\{|x_i|, 1 \le i \le n\}$	30	$[-100, 100]$	0
F_{10}	$f(x) = \sum_{i=1}^{n}\left(\sum_{j-1}^{i} x_j\right)^2$	30	$[-100, 100]$	0
F_{11}	$f(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right) + 3.322$	6	$[0, 1]$	0
F_{12}	$f(x) = -\sum_{i=1}^{10}\left[(X - a_i)(X - a_i)^T + c_i\right]^{-1} + 10.5363$	4	$[0, 10]$	0
F_{13}	$f(x) = -\sum_{i=1}^{5}\left[(X - a_i)(X - a_i)^T + c_i\right]^{-1} + 10.1532$	4	$[0, 10]$	0
F_{14}	$f(x) = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1 + 4\pi x_2) + 0.3$	2	$[-100, 100]$	0
F_{15}	$f(x) = \left\{\left[\sum_{i=1}^{n}\sin^2(x_i)\right] - \exp\left(-\sum_{i=1}^{n} x_i^2\right)\right\} \cdot \exp\left(-\sum_{i=1}^{n}\sin^2\sqrt{|x_i|}\right)$	30	$[-10, 10]$	0
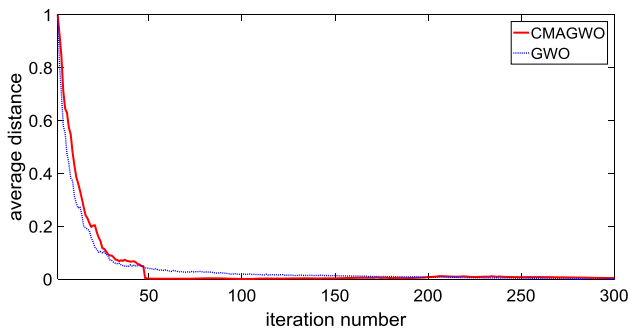
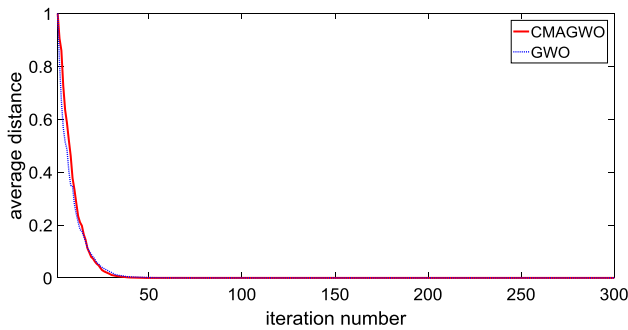**Fig. 2** The average distance for the unimodal function F_1

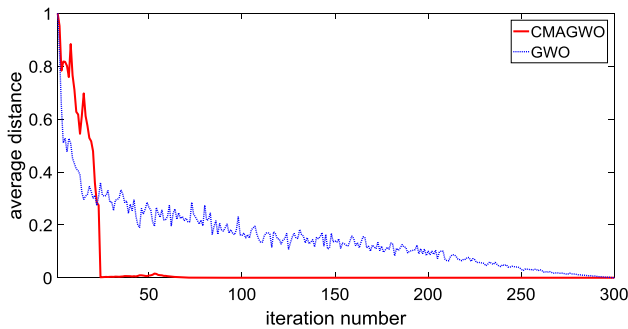**Fig. 3** The average distance for the multimodal function F_2

Fig. 4 The average distance for the fixed-dimension multimodal function F_4

From the results (Tables 3, 9, 10, 11, 12) of unimodal functions (F_1, F_7, F_8, F_9, F_{10}), it can be concluded that the CMAGWO algorithm has higher BEST values and MEAN values than the other algorithms. It is worth mentioning here that the values of the STD for unimodal functions are also competitive for CMAGWO. These low values are indicator of better solution quality obtained from independent runs. Similarly, for multimodal functions (F_2, F_3, F_5, F_6, F_{15}), fixed-dimension multimodal functions (F_4, F_{11}, F_{12}, F_{13}, F_{14}), the results of CMAGWO are competitive for these functions too. For unimodal function F_8 and fixed-dimension multimodal function F_{13}, the MEAN value of the CMAGWO algorithm is large than 10E−3, but also better than other algorithms. Therefore, the CMAGWO

Table 2 The convergence algebra on the standard benchmark functions

	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8
GWO	300	300	300	300	300	300	300	300
MVO	300	300	300	300	300	300	300	300
WOA	300	300	300	22	300	300	300	300
CMAES	300	220	195	42	300	196	213	300
DEGWO	198	300	300	36	300	300	300	300
PSOGWO	300	300	300	300	300	300	300	300
CMAGWO	52	72	125	18	71	96	119	300

	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}
GWO	251	300	300	300	300	14	12
MVO	300	300	300	300	300	299	22
WOA	300	300	300	300	300	96	13
CMAES	300	300	300	300	300	27	11
DEGWO	243	294	300	300	300	15	18
PSOGWO	300	300	300	300	300	52	13
CMAGWO	77	49	44	53	300	15	10

algorithm proposed in this paper has better robustness and accuracy on these standard benchmark functions.

To give a visualized and detailed comparison, Figs. 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18 and 19 give the convergence curves of the proposed CMAGWO algorithm, GWO, MVO, WOA, CMA-ES, PSOGWO and DEGWO on fifteen standard benchmark functions. In these figures, convergence curves are plotted corresponding to the MEAN value of the objective functions obtain in 15 runs. In these graphs, the horizontal axis represents the number of iterations and the vertical axis represents the objective function value. The initialization points of each algorithm are same.

According to the results of Figs. 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18 and 19, CMAGWO is able to provide very competitive results. It is noted that the unimodal functions (F_1, F_7, F_8, F_9 and F_{10}) are suitable for benchmarking exploitation. Therefore, these results (Figs. 5, 11, 12, 13 and 14) show the superior performance of CMAGWO based on the two-stage search in terms of exploiting the optimum. This is due to the proposed exploitation operators previously discussed. Therefore, these results evidence high exploitation capability of the CMAGWO algorithm. In contrast to the unimodal functions, multimodal functions (F_2, F_3, F_5, F_6 and F_{15}) and fixed-dimension multimodal functions (F_4, F_{11}, F_{12}, F_{13}, and F_{14}) have many local optima with the number increasing exponentially with dimension. This makes them suitable for benchmarking the exploration ability of an

Table 3 Optimal results for the function F_1

F_1	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	1.091E−03	1.966E−02	2.405E−04	2.518E−02	7.907E−05	3.275E−03	6.410E−05
WORST	4.036E−03	4.845E−02	8.885E−03	6.522E−02	2.019E−03	9.740E−03	1.787E−03
MEAN	2.493E−03	3.581E−02	2.294E−03	4.622E−02	7.055E−04	6.766E−03	6.281E−04
STD	1.172E−03	1.101E−02	2.819E−03	1.610E−02	6.843E−04	2.166E−03	5.225E−04

Table 4 Optimal results for the function F_2

F_2	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
WORST	1.984E−02	1.009E+00	5.510E−02	2.491E−06	1.478E−02	2.205E−02	0.000E+00
MEAN	1.113E−02	9.595E−01	6.888E−03	4.774E−07	1.478E−03	2.756E−03	0.000E+00
STD	9.471E−03	2.779E−02	9.321E−02	8.363E−07	4.675E−03	7.795E−03	0.000E+00

Table 5 Optimal results for the function F_3

F_3	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	2.138E−01	1.300E−01	1.992E−01	1.146E−08	1.551E−01	4.702E−04	0.000E+00
WORST	1.002E+00	4.939E−01	6.438E−01	1.321E−07	8.308E−01	1.089E−01	0.000E+00
MEAN	5.226E−01	2.461E−01	3.630E−01	6.401E−08	4.663E−01	2.103E−02	0.000E+00
STD	2.682E−01	1.277E−01	1.431E−01	5.075E−08	2.520E−01	3.861E−02	0.000E+00

Table 6 Optimal results for the function F_4

F_4	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	1.761E−05	4.053E−04	5.733E−05	7.486E−06	1.318E−04	8.300E−06	7.486E−06
WORST	2.006E−02	2.006E−02	4.440E−04	1.294E−03	9.237E−04	2.006E−02	7.486E−06
MEAN	2.625E−03	5.347E−03	2.663E−04	6.770E−04	3.944E−04	7.534E−03	7.486E−06
STD	7.047E−03	9.083E−03	1.291E−04	3.453E−04	3.367E−04	1.037E−02	0.000E+00

Table 7 Optimal results for the function F_5

F_5	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	2.310E+00	7.589E+01	0.000E+00	1.275E+01	0.000E+00	1.990E+01	0.000E+00
WORST	1.565E+01	1.581E+02	8.565E+01	5.305E+01	1.039E+00	5.583E+01	0.000E+00
MEAN	6.707E+00	1.290E+02	1.071E+01	2.274E+01	2.570E−01	3.965E+01	0.000E+00
STD	4.766E+00	3.030E+01	3.028E+01	1.281E+01	4.759E−01	1.137E+01	0.000E+00

Table 8 Optimal results for the function F_6

F_6	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	2.496E−02	5.549E−01	1.020E−02	9.635E−08	4.292E−02	1.073E−04	0.000E+00
WORST	1.025E−01	7.457E+00	9.389E−02	1.386E−06	1.972E−01	3.505E−02	0.000E+00
MEAN	7.097E−02	3.402E+00	4.198E−02	5.196E−07	9.076E−02	2.175E−02	0.000E+00
STD	2.350E−02	2.193E+00	3.023E−02	4.779E−07	5.203E−02	1.215E−02	0.000E+00

Table 9 Optimal results for the function F_7

F_7	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	7.266E−01	3.231E+00	4.983E−01	1.400E−08	2.506E−01	3.699E−04	0.000E+00
WORST	1.476E+00	5.595E+00	1.504E+00	8.242E−07	1.513E+00	2.611E−01	0.000E+00
MEAN	1.027E+00	4.130E+00	1.033E+00	1.408E−07	1.026E+00	5.166E−02	0.000E+00
STD	2.766E−01	7.409E−01	3.371E−01	2.774E−07	4.891E−01	9.865E−02	0.000E+00

Table 10 Optimal results for the function F_8

F_8	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	2.700E+01	8.127E+01	2.792E+01	2.587E+01	2.610E+01	2.632E+01	2.384E+01
WORST	2.883E+01	3.134E+03	2.879E+01	2.823E+01	2.856E+01	2.713E+01	2.431E+01
MEAN	2.772E+01	9.790E+02	2.834E+01	2.694E+01	2.701E+01	2.672E+01	2.403E+01
STD	7.714E−01	1.200E+03	3.290E−01	9.236E−01	7.400E−01	3.311E−01	1.591E−01

Table 11 Optimal results for the function F_9

F_9	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	4.063E−04	1.850E+00	1.552E+00	1.102E−02	1.453E−04	5.056E−03	7.681E−07
WORST	1.309E−03	6.899E+00	7.886E+01	4.390E−02	2.695E−03	5.800E−02	2.965E−06
MEAN	7.460E−04	3.939E+00	3.986E+01	2.631E−02	6.710E−04	2.510E−02	1.628E−06
STD	2.925E−04	1.621E+00	2.241E−01	1.321E−02	8.563E−04	1.978E−02	7.930E−07

Table 12 Optimal results for the function F_{10}

F_{10}	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	6.806E−04	5.206E+02	3.876E+04	8.338E+00	1.401E−04	4.880E−03	5.479E−08
WORST	8.351E−02	1.103E+03	1.017E+05	4.632E+01	2.504E−03	1.026E+00	2.769E−06
MEAN	2.455E−02	7.876E+02	6.573E+04	2.165E+01	1.000E−03	2.138E−01	5.486E−07
STD	2.852E−02	2.488E+02	2.238E+04	1.411E+01	7.034E−04	3.642E−01	9.051E−07

Table 13 Optimal results for the function F_{11}

F_{11}	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	2.993E−05	9.521E−06	1.225E−03	4.828E−06	2.182E−05	7.475E−05	4.828E−06
WORST	4.816E−01	1.205E−01	3.219E−01	1.189E−01	2.861E−01	2.355E−01	4.828E−06
MEAN	1.450E−01	4.482E−02	1.307E−01	1.487E−02	1.412E−01	1.636E−01	4.828E−06
STD	1.639E−01	6.184E−02	1.378E−01	4.204E−02	1.226E−01	1.304E−01	0.000E+00

Table 14 Optimal results for the function F_{12}

| F_{12}	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	2.015E−03	8.793E−04	3.556E−02	0.000E+00	5.301E−05	2.717E−03	0.000E+00
WORST	9.759E−03	8.677E+00	8.683E+00	5.408E+00	5.408E+00	3.371E−02	0.000E+00
MEAN	5.209E−03	2.775E+00	3.155E+00	2.016E+00	6.771E−01	1.822E−02	0.000E+00
STD	3.087E−03	3.940E+00	3.457E+00	2.783E+00	1.912E+00	1.183E−02	0.000E+00

Table 15 Optimal results for the function F_{13}

F_{13}	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	2.570E−03	3.805E−04	7.249E−03	3.209E−07	5.140E−04	2.915E−03	3.209E−07
WORST	5.062E+00	7.523E+00	9.271E+00	7.470E+00	6.351E+00	7.523E+00	5.098E+00
MEAN	1.270E+00	3.144E+00	2.448E+00	4.669E+00	3.344E+00	1.883E+00	1.256E+00
STD	2.340E+00	3.488E+00	3.588E+00	3.866E+00	2.799E+00	3.467E+00	2.310E+00

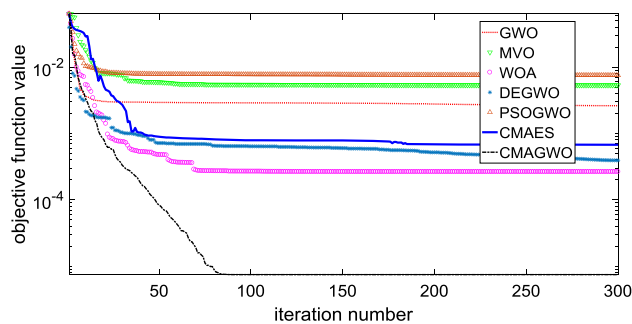**Table 16** Optimal results for the function F_{14}

| F_{14}	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	0.000E+00	1.031E−04	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
WORST	0.000E+00	1.882E−03	2.058E−06	0.000E+00	0.000E+00	0.000E+00	0.000E+00
MEAN	0.000E+00	6.267E−04	2.074E−07	0.000E+00	0.000E+00	0.000E+00	0.000E+00
STD	0.000E+00	5.244E−04	5.410E−07	0.000E+00	0.000E+00	0.000E+00	0.000E+00

Table 17 Optimal results for the function F_{15}

F_{15}	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
WORST	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
MEAN	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
STD	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
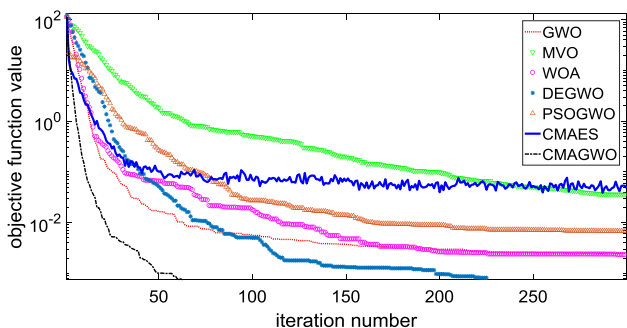
Fig. 5 The convergence curve on the function F_1
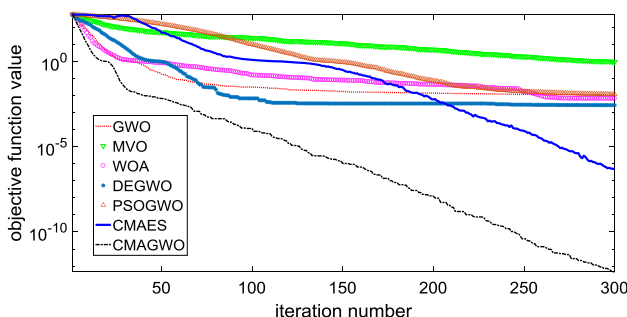
Fig. 6 The convergence curve on the function F_2
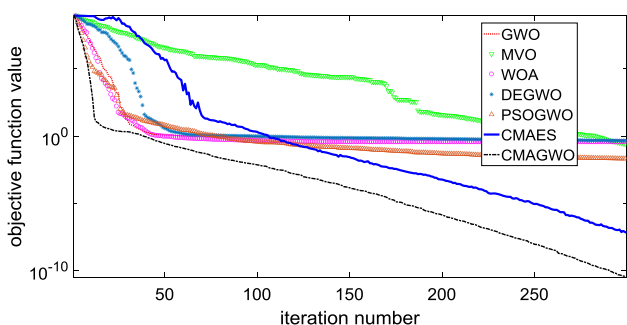
Fig. 7 The convergence curve on the function F_3

Fig. 8 The convergence curve on the function F_4

Fig. 9 The convergence curve on the function F_5
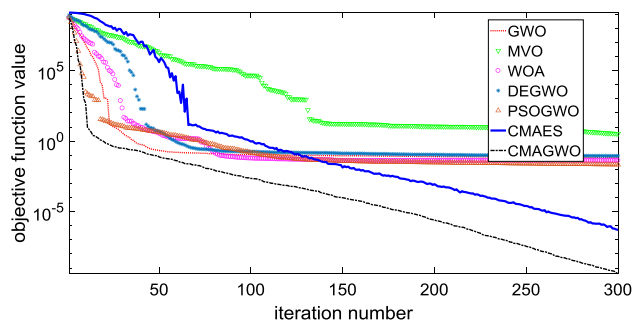
Fig. 10 The convergence curve on the function F_6

algorithm. Due to global search ability of CMAGWO with the *Beta* distribution and the improved hunting formula, the convergence graphs of fixed-dimension multimodal functions (Figs. 8, 15, 16, 17, 18) show the higher diversity in early stage of the search process and better concentration as search progresses. According to Figs. 6, 7, 9, 10 and 19, CMAGWO also provides very competitive results on the multimodal functions (F_2, F_3, F_5, F_6, F_{15}) as well. It can be

observed that the proposed algorithm CMAGWO narrates a more efficient performance compared to GWO, MVO, WOA, CMA-ES, PSOGWO and DEGWO with superior convergence speed and higher precision.
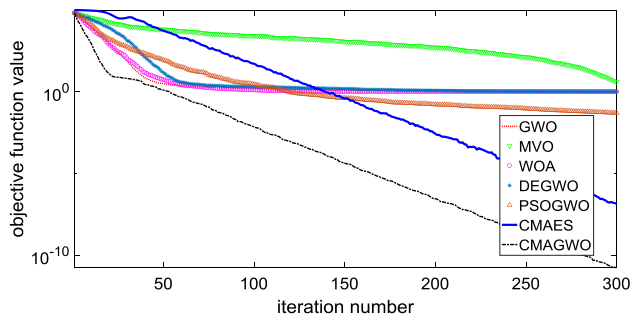
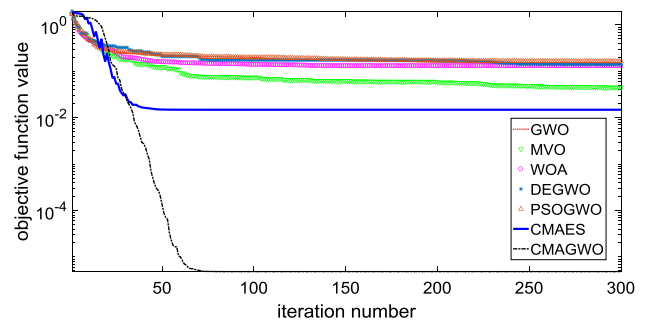**Fig. 11** The convergence curve on the function F_7

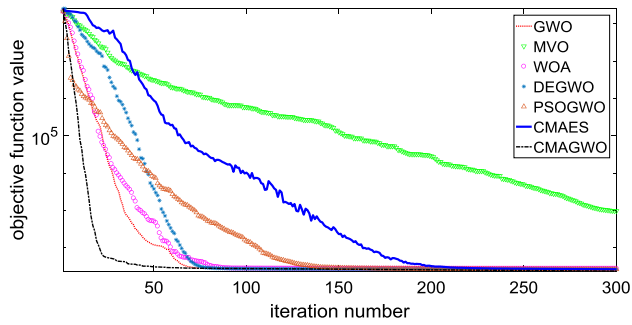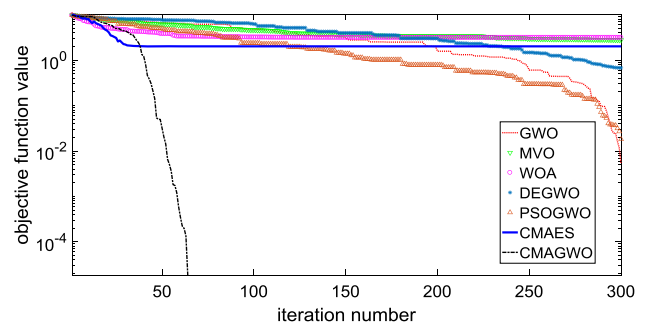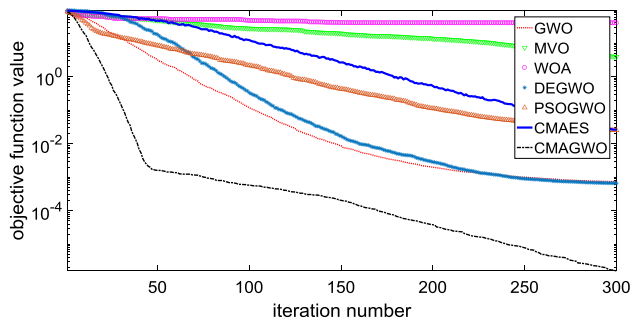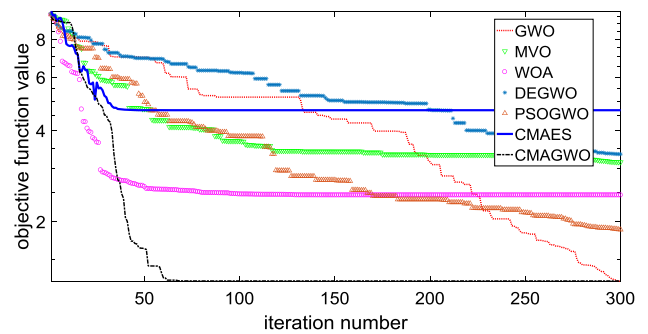**Fig. 15** The convergence curve on the function F_{11}

Fig. 12 The convergence curve on the function F_8

Fig. 16 The convergence curve on the function F_{12}

Fig. 13 The convergence curve on the function F_9

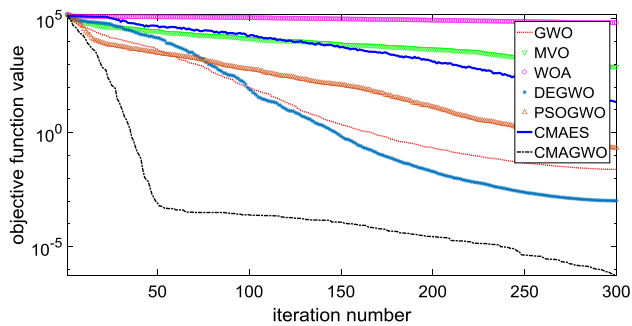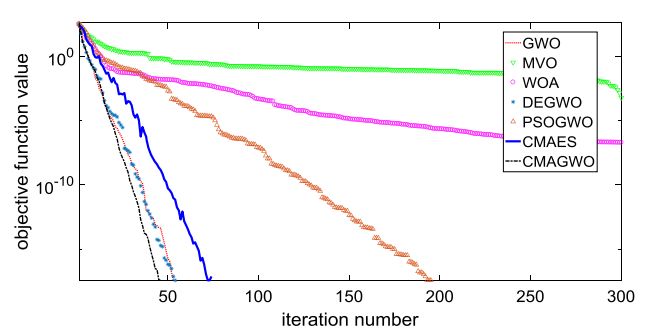**Fig. 17** The convergence curve on the function F_{13}

Fig. 14 The convergence curve on the function F_{10}

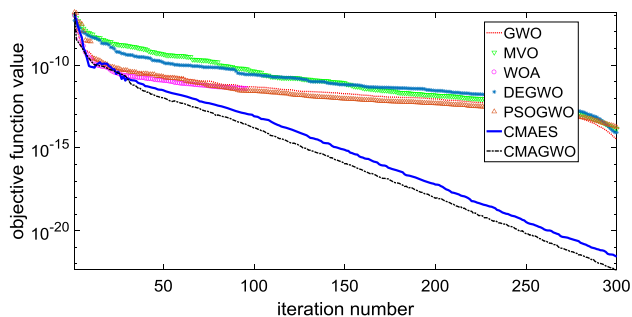**Fig. 18** The convergence curve on the function F_{14}

Fig. 19 The convergence curve on the function F_{15}

4.3 Performance on the five CEC 2014 benchmark functions

The next experimental study is designed on the five numerical optimization problems considered from CEC 2014 special session and competition on single objective real-parameter numerical optimization (Liang et al. 2013). These five test functions of CEC 2014 suite are especially equipped with various novel characteristics such as novel basic problems and functions are shifted, rotated, expanded, and combined variants of the most complicated mathematical optimization problems presented in the literature. The details of these five test functions such as the search domains and dimensions are described in Table 18. For all algorithms (GWO, MVO, WOA, CMA-ES, DEGWO, PSOGWO and CMAGWO), the population size is set as $m = 50$ and the total number of iterations is set as Max_iter = 2000. All experiments on each CEC 2014 suite are iterated over 15 independent runs. The termination criteria in all the algorithms are taken same for a fair comparison between algorithms to avoid the biasedness in performance.

From Table 18, the comparative analysis of the performance of proposed algorithms is studied. The detailed computational data of all test functions for these algorithms are presented in Tables 19, 20, 21, 22 and 23 and Figs. 20, 21, 22, 23 and 24.

The BEST, the WORST, the MEAN and the STD of all algorithms are given in Tables 19, 20, 21, 22 and 23. Inspecting the MEAN values of these functions, it is observed that MEAN values are optimal for CMAGWO as compared with the other algorithms. It is worth mentioning

here that in some cases, especially for function F_{16} Rotated High Conditioned Elliptic Function (CEC1) and F_{18} Shifted and Rotated Katsuura Function (CEC12), the CMAGWO finds the Minimum value. From the STD values obtained from experiments, it is also observed that the performance of CMAGWO is competitive. These indices evaluate the accuracy, and the robustness of the CMAGWO is better than that of other six algorithms.

Convergence curves of this experiment are shown in Fig. 20 for F_{16} Rotated High Conditioned Elliptic Function (CEC1), in Fig. 21 for F_{17} Shifted and Rotated Rastrigin's Function (CEC9), in Fig. 22 for F_{18} Shifted and Rotated Katsuura Function (CEC12), in Fig. 23 for F_{19} Hybrid Function 5 ($N = 5$) (CEC21), and in Fig. 24 for F_{20} Composition Function 5 ($N = 5$) (CEC27). The results of these CEC2014 functions strongly prove that high exploration of the CMAGWO algorithm is fruitful for avoiding local solutions. By inspecting the convergence curves of these functions, it is empirical to state that a significant improvement has been observed in convergence characteristics by the proposed the two-stage search compared to GWO, MVO, WOA, CMA-ES, PSOGWO and DEGWO.

Based on the observations reported in this section, it can be concluded that the CMAGWO algorithm provides very competitive results on these test functions. This superior capability is due to the hybrid mechanism. As mentioned above, some iterations are devoted to exploration (the first stage) and the rest to exploitation (the second stage). Moreover, by embedding Beta distribution in the initialization process and the process of wolves hunting changed, the exploration ability is accelerated. High local optima avoidance of this algorithm is finding that is inferred from above results. The five test functions of CEC 2014 suite have very difficult search spaces, so the accurate approximation of their global optima needs high exploration and exploitation combined. The results again evidence that the CMAGWO algorithm properly balances these two conflicting milestones. In the following section, the applications of proposed algorithm on real engineering problems are investigated.

Table 18 The brief description of CEC 2014 benchmark functions

No	Function	Dimension	Domain	Minimum
F_{16}	Rotated High Conditioned Elliptic Function (CEC1)	30	[− 100,100]	100
F_{17}	Shifted And Rotated Rastrigin's Function (CEC9)	30	[− 100,100]	900
F_{18}	Shifted And Rotated Katsuura Function (CEC12)	30	[− 100,100]	1200
F_{19}	Hybrid function 5 ($N = 5$) (CEC21)	30	[− 100,100]	2100
F_{20}	Composition function 5 ($N = 5$) (CEC27)	30	[− 100,100]	2700

Table 19 Optimal results for the function F_{16}

F_{16}	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	2.706E+07	8.854E+06	1.404E+08	1.000E+02	1.516E+08	1.562E+07	1.000E+02
WORST	1.169E+08	1.240E+07	1.784E+08	1.000E+02	2.077E+08	1.696E+07	1.000E+02
MEAN	7.198E+07	1.062E+07	1.594E+08	1.000E+02	1.796E+08	1.629E+07	1.000E+02
STD	6.353E+07	2.504E+06	2.686E+07	0.000E+00	3.971E+07	9.495E+05	0.000E+00

Table 20 Optimal results for the function F_{17}

| F_{17}	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	9.813E+02	9.662E+02	1.156E+03	1.091E+03	1.106E+03	1.017E+03	9.159E+02
WORST	9.929E+02	9.777E+02	1.167E+03	1.100E+03	1.119E+03	1.023E+03	9.328E+02
MEAN	9.871E+02	9.719E+02	1.162E+03	1.096E+03	1.113E+03	1.020E+03	9.244E+02
STD	8.256E+00	8.182E+00	7.329E+00	6.332E+00	9.098E+00	4.106E+00	3.960E+00

Table 21 Optimal results for the function F_{18}

F_{18}	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	1.202E+03	1.200E+03	1.201E+03	1.200E+03	1.200E+03	1.200E+03	1.200E+03
WORST	1.203E+03	1.201E+03	1.202E+03	1.200E+03	1.201E+03	1.201E+03	1.200E+03
MEAN	1.203E+03	1.201E+03	1.202E+03	1.200E+03	1.201E+03	1.201E+03	1.200E+03
STD	6.049E−01	1.378E−01	3.585E−01	0.000E+00	3.817E−01	3.233E−03	0.000E+00

Table 22 Optimal results for the function F_{19}

F_{19}	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	4.916E+05	3.258E+04	2.751E+06	2.571E+03	2.180E+06	1.549E+05	2.461E+03
WORST	1.136E+06	4.960E+04	3.986E+06	2.588E+03	2.957E+06	2.363E+05	2.525E+03
MEAN	8.140E+05	4.109E+04	2.869E+06	2.580E+03	2.569E+06	1.956E+05	2.493E+03
STD	4.560E+05	1.204E+04	1.663E+05	1.189E+01	5.497E+05	5.753E+04	1.163E+01

Table 23 Optimal results for the function F_{20}

F_{20}	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	3.430E+03	3.223E+03	3.919E+03	3.034E+03	3.546E+03	3.109E+03	3.016E+03
WORST	3.434E+03	3.466E+03	4.086E+03	3.076E+03	3.565E+03	3.117E+03	3.020E+03
MEAN	3.432E+03	3.344E+03	4.002E+03	3.055E+03	3.554E+03	3.113E+03	3.018E+03
STD	2.767E+00	1.717E+02	1.181E+02	3.011E+01	1.848E+01	5.589E+00	1.876E+00
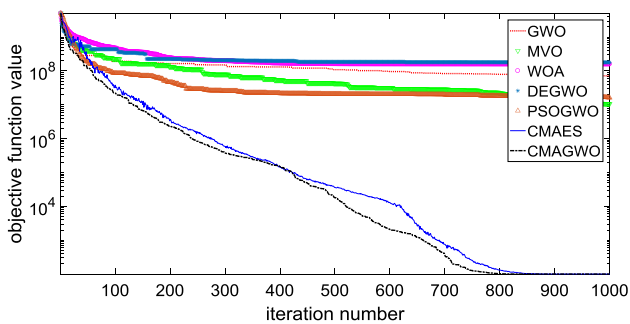
Fig. 20 The convergence curve on the function F_{16}
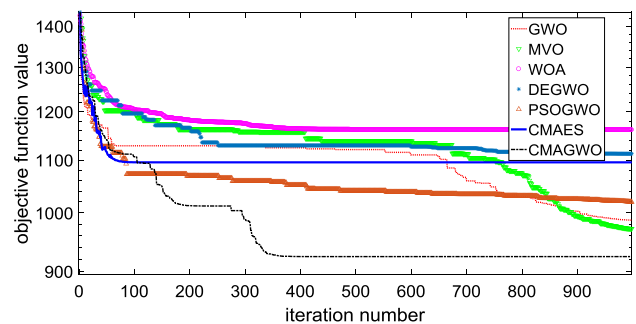
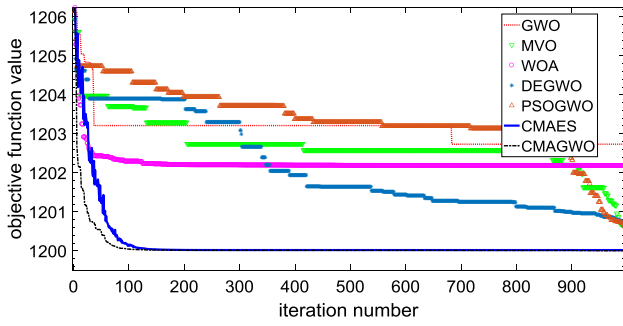**Fig. 21** The convergence curve on the function F_{17}

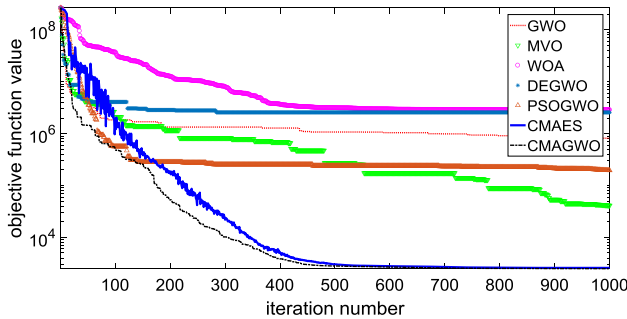**Fig. 22** The convergence curve on the function F_{18}

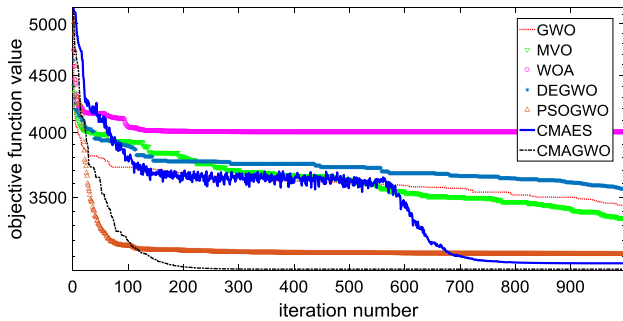**Fig. 23** The convergence curve on the function F_{19}

Fig. 24 The convergence curve on the function F_{20}

5 Application studies on two engineering cases

In this section, to validate the accuracy and effectiveness of the proposed CMAGWO, we use two engineering cases as applications.

5.1 Optimization design of linkage mechanism

The movement of the linkage mechanism (such as the trajectory of a moving point) can be described as the function derived from the relationship of the mechanism's geometric parameters. Suppose that $F\theta_i$ is the desired output angle and $S\theta$ is the actual output angle of the linkage mechanism. $F\theta_0$ is the initial angle for the linkage mechanism.

According to the mechanism's kinematics, the actual output angle of the linkage is obtained as follows:

$$S\theta = \begin{cases} \pi - \arccos\dfrac{r_i^2 - x_2^2 + x_1^2}{2r_1 x_2} - \arccos\dfrac{r_i + 24}{10r_1} & 0 < \varphi_i \le \pi \\ \pi - \arccos\dfrac{r_i^2 - x_2^2 + x_1^2}{2r_1 x_2} + \arccos\dfrac{r_i + 24}{10r_1} & \pi < \varphi_i \le 2\pi \end{cases}$$

(44)

where r_i is the connection parameter, $r_i = \sqrt{26 - 10\cos\varphi_i}$, i is the sequence number after angle division, and $i = 0, 1, 2, \ldots, 30$, φ_i is the actual position angle of the crank, $\varphi_i = \varphi_0 + \frac{i}{30} \times \frac{\pi}{2}$, and φ_0 is the initial angle for the crank.

The desired output angle of the linkage mechanism can be described as follows:

$$F\theta_i = F\theta_0 + \frac{2}{3\pi}(\varphi_i - \varphi_0)^2$$

(45)

Within the given range (in order to make the mechanism have the best transfer performance, the transmission angle of the mechanism required is the minimum value or the maximum value), the optimal design of the linkage mechanism should minimize the error between the actual output angle and the desired output angle. Therefore, the mathematical model for the optimization problem minimizes the least square error between the expected value and the actual value of the output angle of the linkage mechanism.

$$f(x) = \sum_{i=0}^{30}(F\theta_i - S\theta)^2$$

(46)

The given range of motion can be described as a constraint condition:

$$-x_1^2 - x_2^2 + \sqrt{2}x_1 x_2 + 16 \ge 0$$

(47)

$$x_1^2 + x_2^2 + \sqrt{2}x_1 x_2 - 36 \ge 0$$

(48)

where $0 \le x_1 \le 5.0$ and $0 \le x_2 \le 5.0$.

For the constraint conditions, this paper uses the penalty function method to transform the constraint problem into an unconstrained one by introducing a penalty factor.

In this work, the proposed CMAGWO algorithm and other six referred algorithms: GWO, MVO, WOA, CMA-ES, PSOGWO and DEGWO, are all employed to solve the optimization problem of the linkage mechanism. The parameters of these algorithms for this problem are set as the same as in Sect. 4. To solve this optimization problem, results are averaged over 15 independent runs. Figure 25 along with Table 24 shows the results of the CMAGWO with other six algorithms.

Figure 25 shows that compared with GWO, MVO, WOA, CMA-ES, PSOGWO and DEGWO, the CMAGWO algorithm has a faster convergence speed, which is more
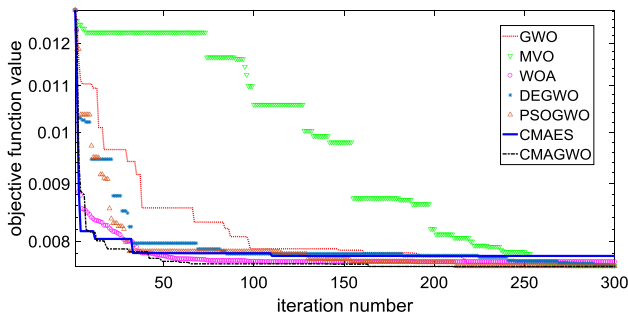
Fig. 25 The convergence curve on the optimization problem of linkage mechanism

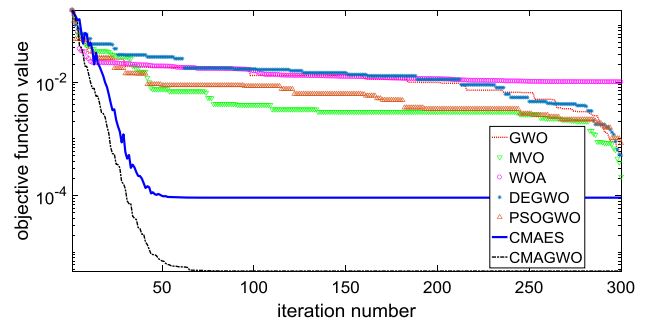**Fig. 26** The convergence curve on the position optimization of robotic arm

satisfactory with the requirements of the actual engineering efficiency. The results of MEAN value and BEST value in Table 24 show that the CMAGWO algorithm can approach the expected value of the linkage mechanism with higher precision. The CMAGWO has the smallest STD between the seven algorithms. Therefore, the CMAGWO algorithm has stronger robustness and adaptability.

5.2 Position optimization of the robotic arm

The robotic arm is composed of the rotating joints and the moving joints. Thus, the robotic arm model can be simplified to the joints that are connected end to end. The first joint is connected to the base of the robotic arm, and the ends are connected with an effector such as a manipulator, welding gun or spray gun. The end effector position that we want to achieve is finally realized by the movement of the joints of the robotic arm.

Therefore, according to the forward kinematics equation of the robotic arm, the relationship between the position of the end effector and the angle of each joint is established. To realize the end effector to complete the work task in the operation space, the actual position $PE = (PE_1, PE_2, PE_3)$ of the arm end can be calculated by the following:

$$
\begin{aligned}
PE_1 = l_1 &- l_4 * \cos(x_1) * \sin x_2 * \sin x_3 \\
&- \cos x_1 \cos x_2 \cos x_3 - l_2 * \sin x_1 - l * \sin x_1 * \sin x_4 \\
&+ \cos x_4 * \cos x_1 * sinx_2 * \sin x_3 - \cos x_1 \\
&* \cos x_2 * \cos x_3 + l_3 * \cos x_1 * \cos x_2
\end{aligned}
\tag{49}
$$

$$
\begin{aligned}
PE_2 = l_2 \cos x_1 &- l_4 * \sin x_1 * \sin x_2 * \sin x_3 \\
&- \cos x_2 \cos x_3 \sin x_1 - l * \cos x_1 * \sin x_4 \\
&- \cos x_4 * \sin x_1 * \sin x_2 * \sin x_3 \\
&- \cos x_2 * \cos x_3 * \sin x_1 + l_3 * \cos x_2 * \sin x_1
\end{aligned}
\tag{50}
$$

$$
\begin{aligned}
PE_3 = -l_4 * \cos x_2 &* \sin x_3 + \cos x_3 \sin x_2 \\
&+ l_3 * \sin x_2 - l * \cos x_4 * \cos x_2 * \sin x_3 \\
&+ \cos x_3 * \sin x_2
\end{aligned}
\tag{51}
$$

where the D–H parameters of the arm are the length of the end ($l = 0.085$) and the lengths of the common vertical line between one joint and the next joint ($l_1 = 0.175$, $l_2 = 0.082$, $l_3 = 0.38$ and $l_4 = 0.26$).

The desired position of the arm end is known. Based on the model that requires the highest accuracy to reach the target point and moves the joint angle as small as possible, the objective function is established as follows:

$$
\begin{aligned}
f(x) = &\sqrt{(PF_1 - PE_1)^2 + (PF_2 - PE_2)^2 + (PF_3 - PE_3)^2} \\
&+ 10^{-5} * ((x_1 - q0_1)^2 + 0.5 * (x_2 - q0_2)^2 \\
&+ 0.5 * (x_3 - q0_3)^2 + 0.1 * (x_4 - q0_4)^2)
\end{aligned}
\tag{52}
$$

where the initial angle of each joint of the robotic arm is $q0 = (-\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}, 0)$, and the constraint condition of each joint is $-2.62 \le x_1 \le -0.52$, $0.52 \le x_2 \le 2.62$, $-2.35 \le x_3 \le -0.79$ and $-1 \le x_4 \le 1$.

Taking the desired target position $PF = (-0.0516, -0.4006, -0.4135)$ of the arm end as an

Table 24 Optimal results for the problem of linkage mechanism

	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	7.594E−03	7.610E−03	7.593E−03	7.671E−03	7.598E−03	7.593E−03	7.592E−03
WORST	7.605E−03	7.619E−03	7.728E−03	7.864E−03	7.614E−03	7.603E−03	7.600E−03
MEAN	7.598E−03	7.614E−03	7.669E−03	7.762E−03	7.602E−03	7.598E−03	7.596E−03
STD	4.118E−06	1.339E−06	6.926E−05	9.704E−05	3.159E−06	5.262E−06	1.266E−06

Table 25 Optimal results for the end position of robotic arm

	GWO	MVO	WOA	CMAES	DEGWO	PSOGWO	CMAGWO
BEST	3.817E−04	1.705E−04	6.665E−03	2.876E−05	2.559E−04	5.709E−04	4.497E−06
WORST	6.745E−04	2.415E−04	1.192E−02	1.704E−04	6.649E−04	9.832E−04	5.030E−06
MEAN	5.281E−04	2.158E−04	1.001E−02	9.184E−05	5.142E−04	8.417E−04	4.688E−06
STD	1.464E−04	3.936E−05	2.908E−03	7.206E−05	2.223E−04	2.346E−04	2.989E−07

example, CMAGWO is used for position optimization and compared with the other algorithms.

Figure 26 shows the convergence curves of the seven algorithms for the position optimization of the robotic arm. The parameters of these algorithms for this problem are set as the same as in Sect. 4. Figure 26 shows that compared with GWO, MVO, WOA, CMA-ES, PSOGWO and DEGWO, the CMAGWO algorithm can approach the desired target position at a faster convergence speed.

The results obtained by the proposed CMAGWO are compared with those gained by other six algorithms, and the comparisons are presented in Table 25. The experimental results in Table 25 show that the CMAGWO algorithm has a higher BEST value and MEAN value in the optimization position of robotic arm. Moreover, the STD of CMAGWO is also the smallest in these algorithms. Table 25 shows that CMAGWO has stronger robustness and accuracy of robotic arm position optimization and verifies the effectiveness of the CMAGWO algorithm.

The results of the design optimization of linkage mechanism and position optimization of the robotic arm demonstrate the performance of the CMAGWO algorithm in terms of exploration, exploitation, local optima avoidance and convergence. This is again due to the two-stage search of hybrid CMA-ES. Furthermore, in order to give full global search ability of the grey wolf optimization, the initial population is thoroughly generated in the non-edge region of the solution space by the *Beta* distribution. The new algorithm improves the hunting formula of the original GWO, increases the diversity of the population through the interference of other individuals and reduces the absolute guidance of the head wolf to each individual. The results show a good balance between exploration and exploitation. This comprehensive study shows that the proposed CMAGWO algorithm has merit among the GWO, MVO, WOA, CMA-ES, PSOGWO and DEGWO.

6 Conclusion

To break through the limitations of the sole algorithm, this paper proposes an improved grey wolf optimization based on the two-stage search of hybrid CMA-ES, which is combined with the global search ability of the GWO algorithm and the strong local search ability of the CMA-ES algorithm. In the first stage, the CMAGWO algorithm mainly uses the global ability of the grey wolf algorithm to explore the entire search space. At this stage, the approximate location of the optimal solution is located as much as possible. When a certain search threshold is reached, the local parallel refinement search is switched. In the second stage, three instances are used to exploit the key areas. During the search process, data exchange and information sharing between instances are carried out. The purposes of the two stages are different, and the operators and operation strategies are different.

Finally, the improved algorithm is validated by fifteen standard benchmark functions and five test functions of CEC 2014 suite. The CMAGWO has been compared with other six popular meta heuristic algorithms. The results show that the CMAGWO algorithm proposed in this paper effectively improves the performance and overcomes the disadvantages of the GWO algorithm, which easily falls into the local optimum in solving complex functions. Moreover, the applicability of the CMAGWO has been demonstrated on two engineering design cases. Through the analysis of the optimal results and the convergence curve, it is observed that the CMAGWO is more satisfactory with the requirements of the actual engineering efficiency. Application of the proposed CMAGWO on more practical challenging problems will be investigated in future.

Compliance with ethical standards

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

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

References

Anand A, Suganthi L (2018) Hybrid GA-PSO optimization of artificial neural network for forecasting electricity demand. Energies 11(4):1–15

Aydilek İB (2018) A hybrid firefly and particle swarm optimization algorithm for computationally expensive numerical problems. Appl Soft Comput 66:232–249

Chakri A, Khelif R, Benouaret M, Yang XS (2017) New directional bat algorithm for continuous optimization problems. Expert Syst Appl 69:159–175

Chen Y, Li L, Xiao J, Yang Y, Liang J, Li T (2018) Particle swarm optimizer with crossover operation. Eng Appl Artif Intell 70:159–169

Chi R, Su Y, Zhang D, Chi X, Zhang H (2017) A hybridization of cuckoo search and particle swarm optimization for solving optimization problems. Neural Comput Appl 4:1–18

Daniel E, Anitha J, Gnanaraj J (2017) Optimum laplacian wavelet mask based medical image using hybrid cuckoo search—grey wolf optimization algorithm. Knowl Based Syst 131:58–69

Eberhart R, Kennedy J (2002) A new optimizer using particle swarm theory. In: MHS'95. Proceedings of the 6th international symposium on micro machine and human science, pp 39–43

Emary E, Zawbaa HM, Hassanien AE (2015) Binary grey wolf optimization approaches for feature selection. Neurocomputing 172(8):371–381

Gupta S, Deep K (2018a) Cauchy grey wolf optimiser for continuous optimisation problems. J Exp Theor Artif Intell 30(6):1051–1075

Gupta S, Deep K (2018b) Random walk grey wolf optimizer for constrained engineering optimization problems. Comput Intell 34(4):1025–1045

Gupta S, Deep K (2018c) An opposition-based chaotic grey wolf optimizer for global optimisation tasks. J Exp Theor Artif Intell 30:1–29

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

Hansen N (2006) The CMA evolution strategy: a comparing review. Stud Fuzziness Soft Comput 192:75–102

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

Klein CE, Segundo EHV, Mariani VC, Coelho LDS (2016) Modified social-spider optimization algorithm applied to electromagnetic optimization. IEEE Trans Magn 52(3):1–4

Liang JJ, Qu BY, Suganthan PN (2013) Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. Technical report, Zhengzhou University and Nanyang Technological University

Lin JT, Chiu C-C (2018) A hybrid particle swarm optimization with local search for stochastic resource allocation problem. J Intell Manuf 29(3):481–495

Medjahed SA, Saadi TA, Benyettou A, Ouali M (2016) Gray wolf optimizer for hyperspectral band selection. Appl Soft Comput 40:178–186

Melo VVD, Iacca G (2014) A modified covariance matrix adaptation evolution strategy with adaptive penalty function and restart for constrained optimization. Expert Syst Appl 41(16):7077–7094

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

Mirjalili SA, Hashim SZM, Sardroudi HM (2012) Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm. Appl Math Comput 218(22):11125–11137

Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69(3):46–61

Mirjalili S, Mirjalili SM, Hatamlou A (2015) Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Comput Appl 27(2):495–513

Nagano MS, Moccellin JV (2002) A high quality solution constructive heuristic for flow shop sequencing. J Oper Res Soc 53(12):1374–1379

Olorunda O, Engelbrecht AP (2008) Measuring exploration/exploitation in particle swarms using swarm diversity. In: Proceedings of the IEEE congress on evolutionary computation, Hong Kong, IEEE Congress on, pp 1128–1134

Pan WT (2012) A new fruit fly optimization algorithm: taking the financial distress model as an example. Knowl Based Syst 26(2):69–74

Peng H, Li L, Kurths J, Li S, Yang Y (2013) Topology identification of complex network via chaotic ant swarm algorithm. Math Probl Eng 3:1–5

Pradhan M, Roy PK, Pal T (2016) Grey wolf optimization applied to economic load dispatch problems. Int J Electr Power Energy Syst 83:325–334

Preuss M (2010) Niching the CMA-ES via nearest-better clustering. In: Conference companion on genetic & evolutionary computation, vol 78. ACM, pp 1711–1718

Qiu J, Xie J, Cheng F, Zhang X (2017) A hybrid social spider optimization algorithm with differential evolution for global optimization. J Univers Comput Sci 23(7):619–635

Raidl GR (2006) A unified view on hybrid metaheuristics. In: Proceedings of the 3rd international workshop on hybrid metaheuristics, Gran Canaria, Spain, pp 1–12

Saxena A, Kumar R, Das S (2018) β-Chaotic map enabled grey wolf optimizer. Appl Soft Comput 75:84–105

Singh N, Singh SB (2017) Hybrid algorithm of particle swarm optimization and grey wolf optimizer for improving convergence performance. J Appl Math 2017:1–15

Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 11(4):341–359

Sujitha J, Baskaran K (2017) Genetic grey wolf optimizer based channel estimation in wireless communication system. Wirel Pers Commun 99(2):965–984

Venkatakrishnan GR, Rengaraj R, Salivahanan S (2018) Grey wolf optimizer to real power dispatch with non-linear constraints. CMES Computer Model Eng Sci 115(1):25–45

Wang X, Haynes RD, Feng Q (2016) A multilevel coordinate search algorithm for well placement, control and joint optimization. Comput Chem Eng 95:75–96

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

Xu Z, Iizuka H, Yamamoto M (2017) Attraction basin sphere estimation approach for niching cma-es. Soft Comput 21(5):1327–1345

Yamany W, Emary E, Hassanien AE (2016) New rough set attribute reduction algorithm based on grey wolf optimization. In: The 1st international conference on advanced intelligent system and informatics, Beni Suef, pp 241–251

Zhu A, Xu C, Li Z, Wu J, Liu Z (2015) Hybridizing grey wolf optimization with differential evolution for global optimization and test scheduling for 3D stacked SoC. J Syst Eng Electron 26(2):317–328