Indonesian Journal of Electrical Engineering and Computer Science
Vol. 30, No. 1, April 2023, pp. 577~585
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v30.i1.pp577-585

Improved grey wolf optimizer for multiple unmanned aerial vehicles task allocation

Yu Wang[1], Qifang Luo[1,2], Yongquan Zhou[1,2]
[1]College of Information Science and Engineering, Guangxi University for Nationalities, Nanning, China
[2]Guangxi Key Laboratories of Hybrid Computation and IC Design Analysis, Nanning, China

Article Info

Article history:

Received Oct 1, 2022
Revised Nov 13, 2022
Accepted Nov 23, 2022

Keywords:

Congestion control strategy
Global best search strategy
Grey wolf optimizer
Improve grey wolf optimization
Metaheuristic optimization
Multiple UAVs task allocation

ABSTRACT

Grey wolf optimizer (GWO) is a metaheuristic optimization algorithm proposed in 2014, which has already been applied in many fields. However, there are still two problems in GWO: i) during the optimization process, there are three leading wolves to lead the population for search, resulting in poor population diversity and ii) because of its position updated equation which not only brings strong convergence ability but also makes it easily fall into local optimal. In this paper, to overcome this, the following contributions were made: i) an improved GWO (IGWO) with two strategies was proposed to solve the above problems and ii) for verifying the effectiveness of IGWO, it was applied in solving multiple UAVs task allocation problems. The experimental results show that IGWO can solve this problem well and suit for large-scale complex examples.

Corresponding Author:

Yongquan Zhou
College of Information Science and Engineering, Guangxi University for Nationalities
530006 Nanning, Guangxi, China
Email: yongquanzhou@126.com

1. INTRODUCTION

As a kind of high mobility, low cost aircraft, unmanned aerial vehicle (UAV) has been widely used in civil and military fields. In the military fields, UAV has used in the suppression of enemy air defense (SEAD) missions. However, with the increasing requirements of complex environments and situations, individual UAV can hardly reach the ideal result. Thus, heterogeneous UAVs with different operational capabilities are needed. For effectively finishing the mission, team cooperation of multiple UAVs is extraordinarily significant. In essence, the task allocation problem is a complex and NP-hard combinatorial optimization problem [1], in which computation cost is exponentially increased proportionally to the number of variables. In general, there are two methods for solving complex combinatorial optimization problems: traditional methods and intelligent optimization methods.

As for traditional ones, dynamic programming, and mixed integer linear methods, have been used in solving this problem. Optimal solutions can be obtained through traditional methods in small-scale problems. However, in large-scale problems, it is difficult to get the optimal solution in a reasonable time. The intelligent optimization methods mainly refer to a series of newly emerging algorithms called a meta-heuristic algorithm, which is inspired by the nature concepts like animal behaviors, physical phenomena, and so on. Different from the traditional methods, the metaheuristic algorithm cannot always obtain the exact optimal solution since its randomness. Instead, what is worked out by the algorithm is an approximate optimal solution in a reasonable time [2]. Several meta-heuristic algorithms have already been used to solve the task allocation problems in SEAD, like particle swarm optimization (PSO) [3], genetic algorithm (GA) [4], and anti colony optimization (ACO) [5]. Since it's free from gaining substantial gradient information, those algorithms are more efficient than traditional methods.

According to the no free lunch (NFL) theorem, different metaheuristic algorithms suited for different optimization problems [6]. Many meta-heuristic algorithms with novel search mechanisms have been proposed. They all seek a balance between exploration and exploitation in the search process [7]. The exploration process is more inclined to search the whole search space, which aims at discovering the region where the optimal solution may exist [8]. Grey wolf optimizer (GWO) is one of the meta-heuristic optimizations which was proposed by Seyedali Mirjalili in 2014 [9]. Due of its simple implementation, flexible use, and fast convergence, GWO has been widely used in various optimization problems like feature selection [10], structural damage identification [11], forecasting electric loads [12], path planning [13], and so on. Like most meta heuristic algorithms, GWO also seeks a balance between exploration and exploitation. However, during the update process, all search agents move toward a globally optimal solution, which allows for rapid convergence of GWO but leads to poor population diversity and easy to falls into local optimal when dealing with large-scale optimization problems [14], [15]. To remedy the defect of GWO, in this article, an improved GWO (IGWO) with a congestion control strategy based on population control and a global best search strategy based on random search was proposed. Then the IGWO was applied to solving the multiple UAVs task allocation. The experimental results proved that the IGWO proposed in this paper has more advantages on the large-scale multiple UAVs task allocation problem.

2. MODEL

2.1. Multiple UAVs task allocation model

2.1.1. Basic model definition

Tables 1 and 2 lists the parameter settings of the problem. The parameters are defined based on the works in [16]-[18]. There are N_vUAVs in the heterogeneous UAV system $U_j, j = \{1,2,\dots N_v\}$ and $N_t$targets in the $T_i.i = \{1,2,\dots\text{Nt}\}$ with a two-dimension position $L_i = (x_i, y_i)$. Each target contains three types of tasks ($k = 1,2,3$ for reconnaissance, attack, and verification) that need to be performed sequentially. t_k indicates the performing time of the k-type task. The difference between UAVs is mainly characterized by the equipment for performing the different tasks, which is indicated $A^U_{j,k},\ k = 1,2,3$, by the j UAV's ability to perform reconnaissance, attack, and verification tasks. Correspondingly, the i target's demand of the ability for performing the tasks is indicated as $A^T_{i,k},\ k = 1,2,3$. For simplicity, the value of the ability belongs to $[0,1]$ only in the value of the UAVs ability is larger than the target demanding ability, then tasks can be performed.

Table 1. Attributes of targets and tasks

Model	Attribute	Parameter
Target, T_i	Number of the targets	N_t
	Target location	$L_i = (x_i,\ y_i)$
	Demanding ability	$A^T_{i,k},\ k \in K$
Task, $M^{T_i}_k$	Number of the tasks	N_k
	Task type	$K = \{1,2,3\}$
	Performing time	$t_k, k \in K$

Table 2. Attributes of UAVs

Model	Attribute	Parameter
UAV, U_j	Number of UAVs	N_U
	Velocity	V_U
	Executive ability	$A^U_{j,k}\ k \in K$

2.1.2. Mathematical models

Based on the above considerations, the mathematical model is shown as follows:

$$min\left(\max_{i\in T,k=3}(t_{ik})\right) + \sum_{u_{ik}\in U,i\in T} T_{O_{u_{ik}},i} + T_{PT_{u_{ik}},i} \quad (1)$$

$$S.T.\ \sum_{u=1}^{Nu} x^u_{i,k} = 1, \quad i \in T, k = 2 \quad (2)$$

$$A^u_{u_{i,k},k} \cdot x^u_{i,k} \geq A^T_{i,k}, \quad k = 1,2,3 \quad (3)$$

$$\sum_{i=1}^{Nt} x_{i,k}^{u} \leq 1, \quad k = 1,2,3 \tag{4}$$

$$t_{i,k} = \begin{cases} max\left(T_{O_{u_{i,k}},i}, t_{i,k-1}\right) + t_k & PT_{u_i,k} = 0 \\ max\left(t_{PT_{u_{ik}},PTS_{u_{ik}}} + T_{PT_{u_{ik}},i}, t_{i,k-1}\right) + t_k & PT_{u_i,k} \neq 0 \end{cases} \tag{5}$$

In (1) i is the sequence number of the target and k is the task type. $k = 1,2,3$indicate respectively reconnaissance, attack, and verification tasks. $t_{i,k}$ is the completion time of the i target and the k task. $u_{i,k}$is the sequence number of UAV that performs the i target and k task. $PT_{U_{i,k}}$is the sequence number of previous targets of the UAV $u_{i,k}$. $T_{0u_{i,k},i}$ is the time of the UAV $u_{i,k}$ flight from the airport to ^{i} the target. $T_{PT_{Ui,k},i}$ is the time of the UAV $u_{i,k}$ flight from the previous target to the i target. In (2) and (3), $x_{i,k}^{u}$ is a binary decision variable. When $x_{i,k}^{u}$=1, it presents the UAV u performing the i target and the j task, otherwise $x_{i,k}^{u}$=0. In (2) constrains UAVs can only attack once, and (3) guaranteed each task for each target is performed only once by one UAV. In (4), $A_{u_{i,k},k}^{u}$is the ability of UAV $u_{i,k}$ to perform k type task and $A_{i,k}^{T}$is the demanding ability of the i target. It's noted that to calculate the latest finish time of the task, every task's time for each target needs to be calculated by (5). $PT_{u_{i,k}} = 0$indicates UAV has no previous target. $PTS_{U_{i,k}}$is the previous task type of the target i.

2.2. The original GWO

2.2.1. The inspiration

This paper is based on the GWO optimizer which is inspired by the grey wolf packs hunting behavior and their strict social hierarchy.The social hierarchy is shown in Figure 1. α,β, and δ are the leader classes. ω is the subordinate class. During the hunting, the ω wolf will follow the navigation of the leaders. Hunting behavior can be divided into three main processes: searching, encircling, and attacking. In the searching process, the packs will search for the prey in the territory. When finding prey, the α wolf will direct the other wolves to encircle and harass the prey to consume its endurance. When the prey is exhausted, the packs will give the prey the last attack.

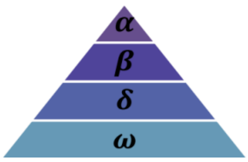

Figure 1. Hierarchy of grey wolf

2.2.2. Mathematical model

The (6)-(9) are used to update the position of the wolves for the next generation

$$r_1 \in [0,1],\ r_2 \in [0,1],\ a = 2\left(1 - \frac{t}{T}\right) \tag{6}$$

$$A = 2ar_1 - a,\ C = 2r_2 \tag{7}$$

$$D = \left|CX_p(t) - X(t)\right| \tag{8}$$

$$X(t+1) = X_p(t) - A \cdot D \tag{9}$$

In (6), r_1 and r_2 are two different n-dimension random vectors between 0 and 1 and a is the tuning parameter for exploration and exploitation which is decreased linearly from 2 to 0 over iterations. A and C are the adjusted vector that can generate disturbance to imitated uncertainties. The GWO assumes that the leader wolves have more information about the prey what is means they are closer to the prey than the ω wolves. Under the guidance of the leader wolves, other wolves approach the prey continuously until catch it. This process can present in:

$$D_\alpha = |CX_\alpha(t) - X(t)|,\ D_\beta = \left|CX_\beta(t) - X(t)\right|,\ D_\delta = |CX_\delta(t) - X(t)| \tag{10}$$

$$X_1 = X_\alpha(t) - A_\alpha \cdot D_\alpha,\ X_2 = X_\beta(t) - A_\beta \cdot D_\beta,\ X_3 = X_\delta(t) - A_\delta \cdot D_\delta \tag{11}$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3} \tag{12}$$

3. METHOD

So far, there are many studies to enhance the performance of GWO, mainly including three ways: adjustment strategy for parameters, integrating operators, and combining GWO with other metaheuristic algorithms. In terms of adjusting parameters, Meidani *et al.* [19] proposed AGWO to enhance the performance by modifying the parameter. Jitkongchuen et al. [20] introduce the weighted parameters for three leader wolves for control the influence of each leader to improve the ability to escape the local optimal. In terms of integrating operators, Heidari and Pahlavani [21] introduce the Lévy flight strategy into GWO to enhance exploration, Gupta and Deep [22] added the random walk strategy to GWO to avoid the local optimal. As for combining with other existing meta-heuristic algorithms, Singh and Singh [23] combined GWO with PSO and proposed a hybrid meta-heuristic algorithm named HGWOPSO. Tawhid and Ali [24] combined GWO with GA and achieved good results. In this paper, we introduced two strategies to enhance the performance of GWO and the details are described below.

3.1. Congestion control strategy

For avoiding premature convergence in the early stage, agents need to explore the search space as much as possible. Keeping the wolves at a distance is a viable approach. It's the same in the natural situation, when the wolves encircle the prey, they will keep their distance from each other to avoid hurting themselves [25]. By the inspiration of this, for keeping the distance between the leader wolves and other wolves, a threshold is settled. When the distance is small than the threshold, the position will be reset randomly in the search space. The threshold is as follows:

$$w = -\left(\frac{w_i}{1+\left(\frac{w_i}{w_f}-1\right)\times e^{-(r\times t)}}\right) + \left(w_i + w_f\right) \tag{13}$$

$$r = 0.01 \times \left(log_{10} w_i - log_{10} w_f\right) \tag{14}$$

$$w_i = 0.05 \times (ub - lb) \tag{15}$$

$$X(t) = \begin{cases} X(t) & d \le w \\ reset & d > w \end{cases} \tag{16}$$

$$d = \left|X(t) - X_p(t)\right| \tag{17}$$

where w is the threshold to control the distances. In (13). w_i is the initial value of w and it depends on the upper and lower bounds. w_f is the final value of w, which is related to the accuracy of the problem. $r$is the step length, and t is the iteration. To describe the distance between individuals, Euclidean distance d is used as a measure. The position resets when the distance d is less than w.

3.2. Global best search strategy

During the search process, the ω wolves' position updating mainly depends on the guidance of three leading wolves, which will make the newly selected leading wolves very likely to be near the position of the previous leading wolves, which can make the algorithm converge quickly, but it also limits the exploration ability of the algorithm and makes the algorithm easy to fall into a locally optimal solution. To overcome this drawback, an update phase is introduced to the leading wolves. At this stage, the leading wolves will perform a random search, and if it finds a more optimal position, it will update the position otherwise it will unchanged. The updated formula is as follows:

$$x_i = x_i + (2u-1)(ub_i - lb_i), \quad u \in [0,1], i = 1,2,\cdots, dim \tag{18}$$

in (18), x_i is the i component of leading wolves. u is a random value in [0,1], ub_i and lb_i are the upper and lower boundaries.

4. RESULTS AND DISCUSSION

4.1. Experimental setup

The experiment was implemented in MATLAB R2019a. Experiments were performed on a PC with a 3.00 GHz, Intel(R) Core(TM) i7-9700 CPU. Four meta-heuristic algorithms, PSO [26], GWO [9], ACO [27], and DE [28], were tested on two different scale examples to compare with the proposed IGWO. Each algorithm was run independently 10 times, and the optimal solution, the worst solution, the variance, the mean value, and the running time of the results were taken as evaluation indexes. The parameter Settings of each algorithm are shown in Table 3. The specific parameters of three examples can get in https://github.com/sameleer/UAVS.

Table 3. Parameters setting for experiments

Algorithm	Parameter	Value
PSO	Acceleration constants (c_1, c_2)	[1.5,2.0]
	Inertia weights (ω)	[1,0.99]
DE	Crossover probability (p_c)	0.8
	Differential weight	0.5
GWO	a	a was linearly decreased from 2 to 0
ACO	Pheromone Exponential Weight (α)	1
	Evaporation Rate（rho）	0.1
IGWO	a	a was linearly decreased from 2 to 0

4.2. Result analyze

Example 1 is a small-scale example, including 5 mission targets and 8 UAVs. The experimental results are shown in Table 4 and Figures 2 and 3. It can be seen from Table 4, in the best situation, the gap between the algorithms is not very large. However, in the worst case, GWO and PSO will fail to obtain a feasible solution. This is because it is trapped in the local optimal solution in the search process. However, the improved IGWO does not fall into the local optimal solution, and gives a feasible solution even in the worst case, which indicates that our improved strategy is effective, and it enhances the ability of the original GWO to jump out of the local optimal solution. DE has the best performance, which is ahead of other algorithms in terms of best, worst, mean, and time. The running time of IGWO is longer than other algorithms because of its higher computational complexity. The specific allocation scheme can be seen from Figure 3. Where Figures 3(a)-(e) represent the solution results of IGWO, GWO, PSO, ACO and DE, respectively. The vertical coordinatesrepresents the number of the UAV, the horizontal axis represents the time, and the most reasonable allocation scheme is given by DE. Example 2 is a large-scale example, including 30 mission targets and 50 UAVs. The experimental results are shown in Table 5, Figures 4 and 5. As can be seen from the results, as the scale of the problem increases, the difficulty of solving it also increases. Except for IGWO, other algorithms do not give feasible solutions. Experimental results show that IGWO has distinct advantage.

Table 4. The result of example 1

Algorithm	Best	Worst	Mean	Std	Time(S)
IGWO	88.5193	99.1645	93.6945	3.8762	7.745
GWO	105.5721	1105.9565*	216.4815	312.5892	5.352
PSO	105.598	1100.8524*	214.9853	311.4252	5.085
ACO	97.9968	105.9746	102.4901	**2.8058**	6.72
DE	**87.7699**	**96.7998**	**91.6986**	3.2435	**4.871**

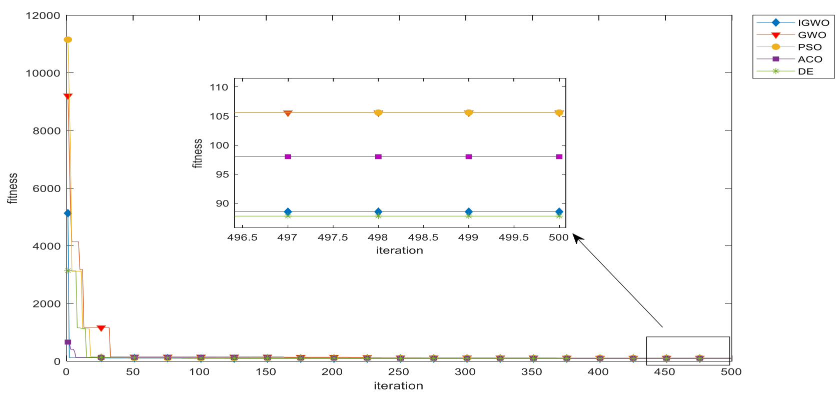

Figure 2. The convergence curve of example 1

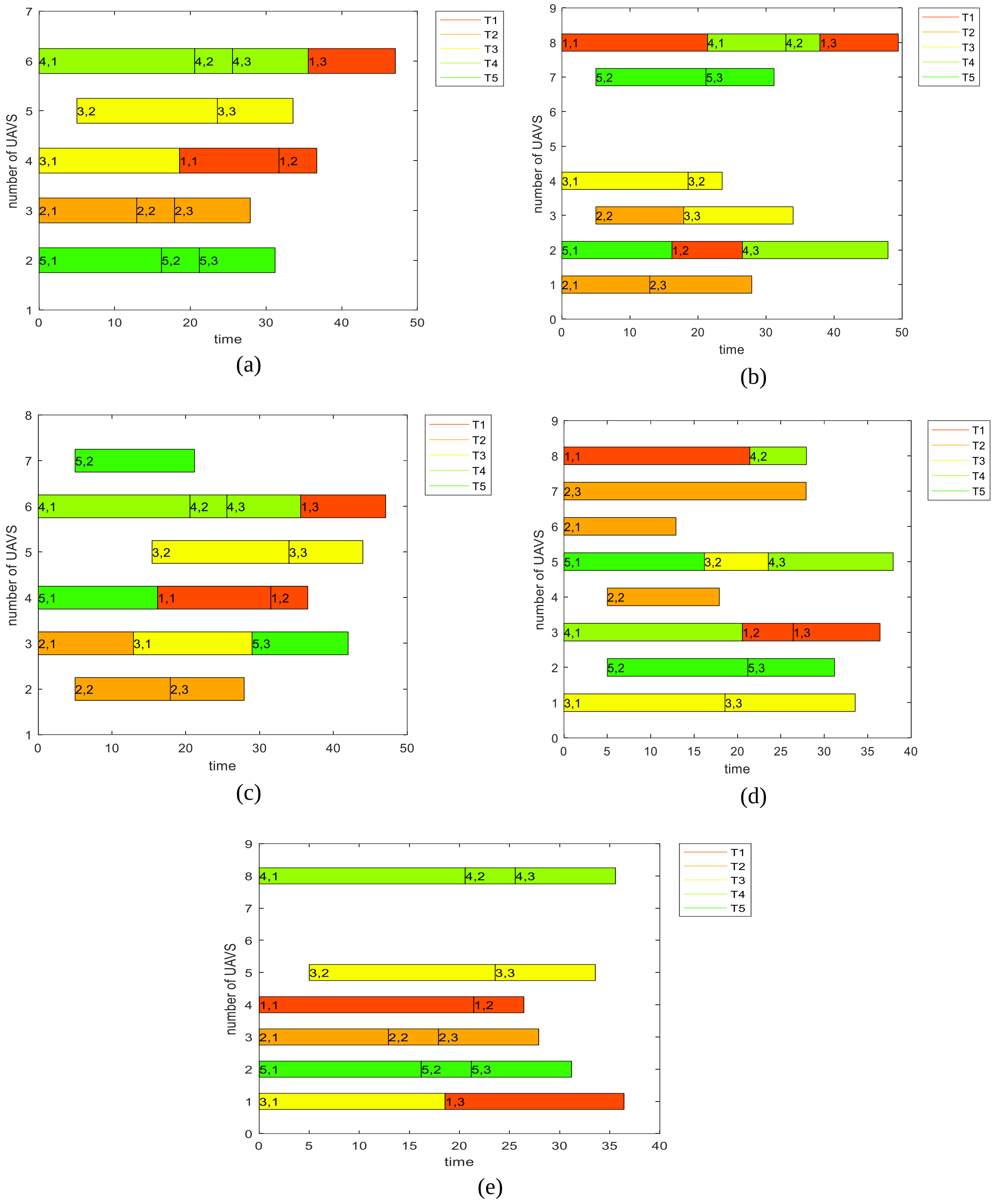

Figure 3. The best result of example 1: (a) IGWO, (b) GWO, (c) PSO, (d) ACO, and (e) DE

Table 5. The result of example 2

Algorithm	Best	Worst	Mean	Std	Time
IGWO	**3468.4639**	**4632.3932**	**3840.5823**	**367.3512**	59.886
GWO	48156.1822*	146032.6772*	98867.2849	36823.326	22.568
PSO	19179.4908*	163847.6439*	41248.0035	43582.1682	**21.189**
ACO	10317.7007*	12512.2719*	11529.5041	712.6525	34.463
DE	37765.628*	50173.7063*	42871.0893	3746.2556	22.029

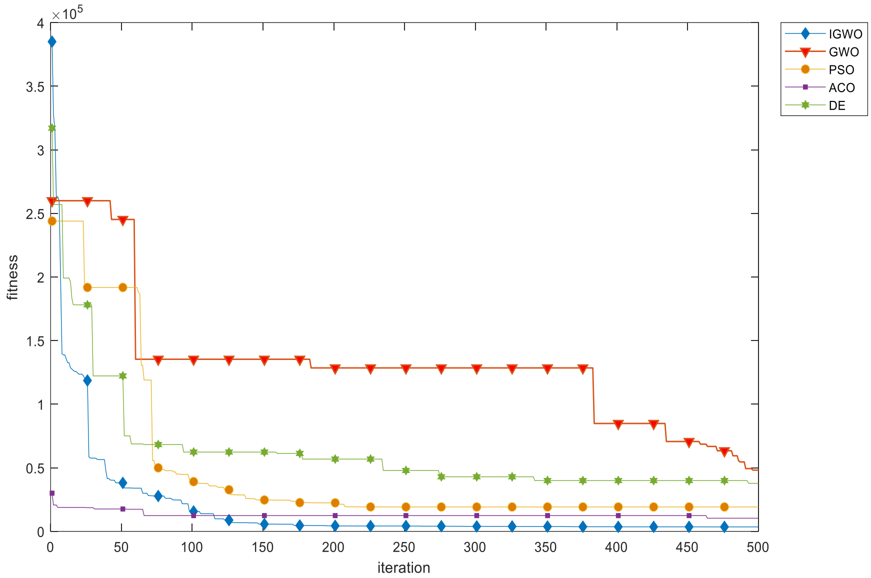

Figure 4. The convergence curve of the example2

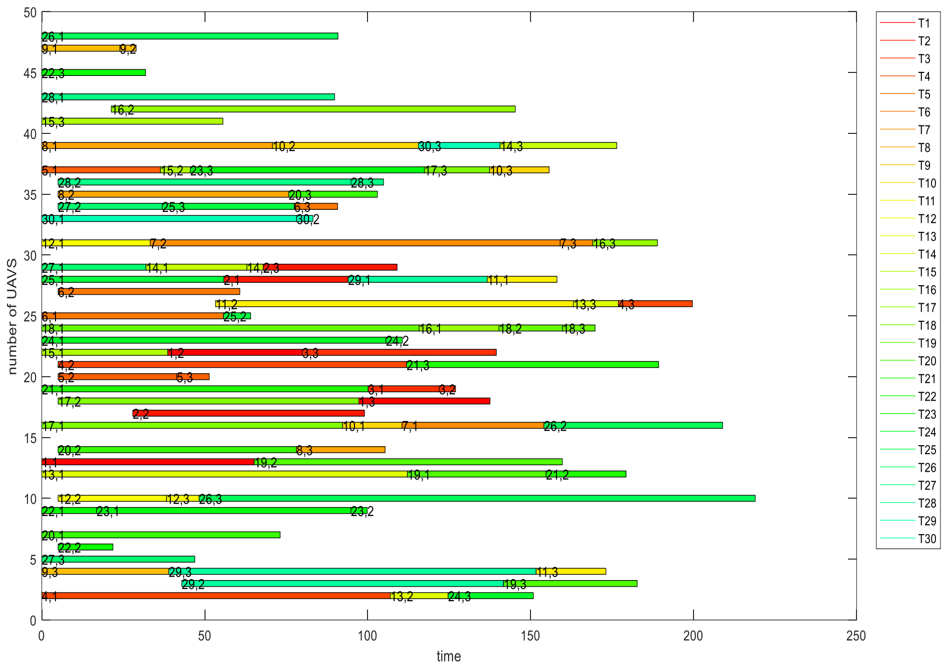

Figure 5. The best result of IGWO in example 2

5. CONCLUSION

In this paper, the congestion control strategy and the global best search strategy are introduced into GWO to remedy the poor population diversity and premature convergence. Then, the proposed IGWO was applied to solve the multiple UAVs task allocation problem. The following conclusions can be drawn from experiments on three examples of different scales: i) the introduced congestion control strategy ensured the diversity of the population by resetting the positions of individuals too close to the leader wolves, ii) the introduced global optimal search strategy improves the exploration performance of the algorithm by adding a random search phase of the leading wolves and enhances the ability of the algorithm to jump out of the local optimal solution, and iii) IGWO performs well in small-scale examples and shows certain advantages in large-scale examples, which indicates that IGWO is more suitable for large-scale optimization problems. However, the IGWO still has shortcomings. Due to the added search process of the leader wolves, the computational complexity increased. This makes it take more time compared to other algorithms. So the future work we will consider reducing its computational complexity to make it solve large-scale optimization problems faster.

ACKNOWLEDGMENT

This work was supported by the National Science Foundation of China under Grant No. 62066005, and the Project of the Guangxi Science and Technology under Grant No. AD21196006.

REFERENCES

[1] N. Buckman, H. L. Choi, and J. P. How, "Partial replanning for decentralized dynamic task allocation," In *AIAA Scitech* 2019 Forum, Jan. 2019, doi: 10.2514/6.2019-0915.

[2] X. S. Yang, "Mathematical analysis of nature-inspired algorithms," in *Nature-Inspired Algorithms and Applied Optimization, Studies in Computational Intelligence,* vol. 1, 2018, pp. 1-25, doi: 10.1007/978-3-319-67669-2_1.

[3] R. P. Zhang, Y. X. Feng, and Y. K.Yang, "Hybrid particle swarm algorithm for multi-UAV cooperative task allocation," *Acta Aeronautica et Astronautica Sinica,* Sep. 2021, doi:10.7527/S1000-6893.2 021.26011.

[4] Y. Eun and H. Bang, "Cooperative task assignment/path planning of multiple unmanned aerial vehicles using genetic algorithm," *Journal of Aircraft,* vol. 46, no. 1, pp. 338–343, May. 2012, doi: 10.2514/1.38510.

[5] L. Chen, W.-L. Liu, and J. Zhong, "An efficient multi-objective ant colony optimization for task allocation of heterogeneous unmanned aerial vehicles," *Journal of Computational Science,* vol. 58, 2022, doi: 10.1016/j.jocs.2021.101545.

[6] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE transactions on evolutionary computation*, vol. 1, no. 1, pp. 67-82, Apr.1997, doi: 10.1109/4235.585893.

[7] A. Kumar, M. Nadeem, and H. Banka, "Nature inspired optimization algorithms: a comprehensive overview," *Evolving Systems*, 2022, doi: 10.1007/s12530-022-09432-6.

[8] X. S. Yang, S. Deb, and S. Fong,"Metaheuristic algorithms: optimal balance of intensification and diversification," *Applied Mathematics & Information Sciences*, vol. 8, no. 3, pp. 977-983, May. 2014, doi: 10.12785/amis/080306.

[9] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in engineering software*, vol. 69, pp. 46-61, Mar. 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[10] Preeti and K. Deep, "A random walk Grey wolf optimizer based on dispersion factor for feature selection on chronic disease prediction," *Expert Systems with Applications,* vol. 206, 2022, doi: 10.1016/j.eswa.2022.117864.

[11] T. Sang-To, H. Le-Minh, S. Mirjalili, M. A. Wahab, and T. Cuong-Le, "A new movement strategy of grey wolf optimizer for optimization problems and structural damage identification," *Advances in Engineering Software,* vol. 173, 2022, doi: 10.1016/j.advengsoft.2022.103276.

[12] Z. Zhang and W.C. Hong, "Application of variational mode decomposition and chaotic grey wolf optimizer with support vector regression for forecasting electric loads," *Knowledge-Based Systems*, vol. 228, 2021, doi: 10.1016/j.knosys.2021.107297.

[13] J.X. Lv et al., "A new hybrid algorithm based on golden eagle optimizer and grey wolf optimizer for 3D path planning of multiple UAVs in power inspection," *Neural Computing and Applications,* vol. 34, no. 14, pp. 11911-11936, 2022, doi: 10.1007/s00521-022-07080-0.

[14] M. H. Nadimi-Shahraki, S. Taghian, and S. Mirjalili, "An improved grey wolf optimizer for solving engineering problems," *Expert Systems with Applications,* vol. 166, 2021, doi: 10.1016/j.eswa.2020.113917.

[15] H. Faris, I. Aljarah, M. A. Al-Betar, and S. Mirjalili, "Grey wolf optimizer: a review of recent variants and applications," *Neural computing and applications*, vol. 30, no. 2, pp. 413-435. Nov. 2017, doi: 10.1007/s00521-017-3272-5.

[16] H. Y. Zhang, L. Wang, X. Zhang, Y. Ding, and C. lv, "Multi-UAV cooperative mission planning considering subsystem execution capability," *Systems Engineering and Electronics*, pp. 1-14, Apr. 2022.

[17] H. X. Chen, Y. Nan, and Y. Yang,"Multi-UAV reconnaissance task assignment for heterogeneous targets based on modified symbiotic organisms search algorithm," *Sensors,* vol. 19, no. 3, pp. 734, Feb. 2019, doi: 10.3390/s19030734.

[18] Q. T. Han, "An application of improved PSO algorithm in cooperative search task allocation," 2021 *IEEE International Conference on Power Electronics, Computer Applications (ICPECA),* 2021, pp. 580-583, doi: 10.1109/ICPECA51329.2021.9362515.

[19] K. Meidani, A. Hemmasian, S. Mirjalili, and A. B. Farimani, "Adaptive grey wolf optimizer," *Neural Computing and Applications,* vol. 34, no. 10, pp. 7711-7731, 2022, doi: 10.1007/s00521-021-06885-9.

[20] D. Jitkongchuen, W. Sukpongthai and A. Thammano, "Weighted distance grey wolf optimization with immigration operation for global optimization problems," in *2017 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD),* 2017, pp. 5-9, doi: 10.1109/SNPD.2017.8022652.

[21] A. A. Heidari and P. Pahlavani,"An efficient modified grey wolf optimizer with Lévy flight for optimization tasks,"*Applied Soft Computing,* vol. 60, pp. 115-134, Nov. 2017, doi: 10.1016/j.asoc.2017.06.044.

[22] S. Gupta and K. Deep, "A novel random walk grey wolf optimize," *Swarm and evolutionary computation*, vol. 44, pp. 101-112, Feb. 2019, doi: 10.1016/j.swevo.2018.01.001.

[23] N. Singh and S. B. Singh, "Hybrid algorithm of particle swarm optimization and grey wolf optimizer for improving convergence performance," *Journal of Applied Mathematics,* vol. 2017, Nov. 2017, Art. no. 2030489, doi: 10.1155/2017/2030489.

[24] M. A. Tawhid, and A. F. Ali, "A hybrid grey wolf optimizer and genetic algorithm for minimizing potential energy function," *Memetic Computing*, vol. 9, no. 4, pp. 347-359, May. 2017, doi: 10.1007/s12293-017-0234-5.

[25] C. Muro, R. Escobedo, L. Spector, and R. P. Coppinger,"Wolf-pack (Canis lupus) hunting strategies emerge from simple rules in computational simulations," *Behavioural processes*, vol. 88, no. 3, pp. 192-197, Nov. 2011, doi: 10.1016/j.beproc.2011.09.006.

[26] R. Poli, J. Kennedy, and J. Blackwell, "Particle swarm optimization," *Swarm intelligence,* vol. 1, no. 1, pp. 33-57, Aug. 2007, doi: 10.1007/s11721-007-0002-0.

[27] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," in *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 28-39, Nov. 2006, doi: 10.1109/MCI.2006.329691.

[28] R. Storn and K. Price, “Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces,” *Journal of Global Optimization*, vol. 11, pp. 341–359, 1997.

BIOGRAPHIES OF AUTHORS

Yu Wang was born in 1995 Shanxi, ChangZhi, China. 2013-2017 was studied in Changzhou Institute of Technology, major in Mechanical and Electronic Engineering. 2020-now was studied in GuangxiUniversity for Nationalities the Master Degree of Engineering. His research areas include swarm intelligence optimization algorithms, neural networks, and deep learning. He can be contacted at email: yuwang042@163.com.

Qifang Luo Prof. received her BS degree from Guangxi University of School of Computer and Electronis Information, Guangxi, China, in 1993. He is currently research interest is in computation intelligence, neural networks. He can be contacted at email: l.qf@163.com.

Yongquan Zhou received Ph.D. & Prof. He received the MS degree in computer science from Lanzhou University, Lanzhou, China, in 1993 and the Ph.D. degree in computation intelligence from the Xiandian University, Xi'an, China, in 2006. He is currently a professor in Guangxi University for Nationalities. His research interests include computation intelligence, neural networks, and intelligence information processing et al. He has published 3 books, and more than 250 research papers in journals. He can be contacted at email: yongquanzhou@126.com.