Mixed Game Pigeon-Inspired Optimization for Unmanned Aircraft System Swarm Formation

Haibin Duan[1(✉)], Bingda Tong[1], Yin Wang[2], and Chen Wei[1]

[1] School of Automation Science and Electrical Engineering, Beihang University, Beijing 100083, China
hbduan@buaa.edu.cn

[2] College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

Abstract. This paper proposes a novel mixed game pigeon-inspired optimization (MGPIO) algorithm for unmanned aircraft system (UAS) swarm formation control. The outer loop controller based on artificial potential field method is designed to transform the UAS swarm formation into abstract movements in the potential field. The inner loop controller based on PIO is designed to solve the optimal UAS position. A novel pigeon-inspired optimization integrated with mixed game theory is proposed to enhance its capacity and convergence speed to solve complex problem while reducing the computational load. This method maintains the capability of the PIO to diversify the pigeons' exploration in the solution space. Moreover, the proposed method improves the quality of the pigeons based on the situation. A series of simulation experiments are conducted compared with basic PIO and Particle Swarm Optimization (PSO) approach. The experimental results verify the feasibility and effectiveness of the proposed method.

Keywords: Pigeon-inspired optimization · Mixed game theory · Unmanned aircraft system · Swarm formation

1 Introduction

Unmanned Aircraft System (UAS) has demonstrated repeatedly major potential for diverse applications in military, civilian and public domains [1]. UAS swarm formation control has strong coupling and nonlinearity and no direct mapping relationship between the performance index and the model parameters, the selection of the control input of the close formation model is a key problem. The swarm intelligence optimization algorithm has no special requirements for solving these problems, hence it has obvious advantages in controlling unmanned vehicles, robot path planning and UAS swarm formation.

Pigeon-inspired optimization (PIO) is a novel optimization algorithm, which presented in 2014 [2]. Unlike other swarm-based algorithms such as Particle Swarm Optimization (PSO) and Differential Evolution (DE), PIO uses the special homing ability of pigeons that they combine the sun, the earth's magnetic field and landmarks

Y. Tan et al. (Eds.): ICSI 2019, LNCS 11655, pp. 429–438, 2019.
https://doi.org/10.1007/978-3-030-26369-0_40

to find their destination. However, the basic PIO algorithm is easy to fall into the local optimal solution. Xu et al. proposed a modified method based on PIO to avoid falling into the local optimal value and increase the population diversity by introducing the adjacent-disturbances and integrated-dispatching strategies [3]. Duan and Wang employed PIO approach in the training process of the Echo state network (ESN) to obtain desired parameters [4]. Zhang et al. proposed a novel predator-prey pigeon-inspired optimization (PPPIO) to solve the UAV three-dimension path planning problem in dynamic environment [5]. In this paper, a novel pigeon-inspired optimization integrated with mixed game theory (MGPIO) is proposed to solve the problem for swarm formation of the UAS. PIO is aimed at pigeons' navigation behavior, by simulating its characteristics, to find the global optimal solution.

2 Design of Outer Loop Controller Based on Artificial Potential Field Method

Consider a UAS consisting of n drones in a 3-dimensional Euclidean space, each drone is considered as a particle, then the kinetic model of each drone is described as follows:

$$\dot{P}^i = v^i, m^i \dot{v}^i = u^i - k^i v^i, i = 1, \ldots, n \tag{1}$$

where $\dot{P}^i \in \mathbb{R}^3$ indicates the position vector of drone i, $\dot{v}^i \in \mathbb{R}^3$ indicates the speed vector of drone i, $m^i > 0$ indicates the mass of drone i. $\dot{u}^i \in \mathbb{R}^3$ is the control input value and the $-k^i v^i$ is the speed damping term.

In order to achieve the desired speed of the entire UAS and maintain constant distance between drones, it is necessary to control the speed of UAS to make it consistent and tend to expect speed. At the same time, it is necessary to control the distance between the drones so that the total potential energy is minimized. In summary, the control input u^i of drone i can be described as:

$$u^i = \alpha^i + \beta^i + \gamma^i + k^i v^i \tag{2}$$

where α^i represents the component generated by the artificial potential function in the UAS swarm, it comes from Eq. (3). β^i represents the component which drone i converges with its neighboring drones. γ^i represents the component of drone i speed tending to the desired speed, which depends on the input signal of the leader drone.

The potential function between drone i and its adjacent drone j is:

$$V^{ij}\left(\left\|P^{ij}\right\|\right) = \ln\left\|P^{ij}\right\|^2 + \frac{R_{desire}^2}{\|p^{ij}\|^2} \tag{3}$$

where $P^{ij} = P^i - P^j$ indicates the relative position vector between drone i and drone j. R_{desire} indicates the desired distance between the drone i and drone j [6] in the UAS.

The control input u^i of wingman i includes three dimensions. The first two dimension $u^i_{1,2}$ are the control input in the horizontal direction and the third dimension

u_3^i is in the vertical direction. It is assumed that all drones in the UAS can receive leader's input signal (leader's speed state), the control input $u_{1,2}^i$ can be defined as follows:

$$u_{1,2}^i = -K_p \sum \nabla_{\|P_{1,2}^{ij}\|} V^{ij} - K_v \sum \left(v_{1,2}^i - v_{1,2}^j\right) - m^i \left(v_{1,2}^i - v_{1,2}^1\right) + k^i v_{1,2}^i \tag{4}$$

where $K_v > 0$ indicates the speed feedback gain factor and $K_p > 0$ indicates the artificial potential field gain factor to control the priority of speed's consistence and the formation.

The control input in the vertical direction u_3^i is defined as follows:

$$u_3^i = -K_h\left(P_3^i - P_3^j\right) - K_v \sum \left(v_3^i - v_3^j\right) - m^i\left(v_3^i - v_3^1\right) + k^i v_3^i \tag{5}$$

where K_h indicates the altitude feedback gain factor to control the altitude of the formation.

3 Design of Inner Loop Controller Based on Pigeon-Inspired Optimization and Mixed Game Theory

3.1 Pigeon-Inspired Optimization

Pigeons have special navigation capabilities. Pigeons use the sun, the Earth's magnetic field and landmarks to find paths, and use different navigation tools at different stages of the itinerary. When they start flying, the pigeons rely more on navigation tool like compass. In the middle of the itinerary, the navigation tool can be switched to the landmark, this moment the individual pigeons will re-evaluate the route they have experienced and make corrections.

Based on the special behavior of the pigeons during the itinerary, pigeon-inspired optimization uses two different operator models to mimic the different navigation tools in different stages of the pigeon flight.

Map and Compass Operator

The rules in the map and compass operator are defined with the position X_i and the velocity V_i of pigeon i, and the positions and velocities in a D-dimension search space are updated in each iteration. The new position X_i and velocity V_i of pigeon i at the t-th iteration can be calculated as follows:

$$V_i(t) = V_i(t-1)e^{-Rt} + r_1\left(X_g - X_i(t-1)\right) \tag{6}$$

$$X_i(t) = X_i(t-1) + V_i(t) \tag{7}$$

where R is the map and compass factor, r_1 is a random number, and X_g is the current global best position, which can be obtained by comparing the positions among all the pigeons.

As shown in Fig. 1, the best position of the pigeons is developed by using map and compass operator. By comparing the pigeons' positions, the pigeon on the right is the best pigeon. Each pigeon can adjust its flying direction according to (6), which is expressed by the thick arrows. The thin arrows are its former flying direction. The vector sum of these two arrows is its next flying direction.

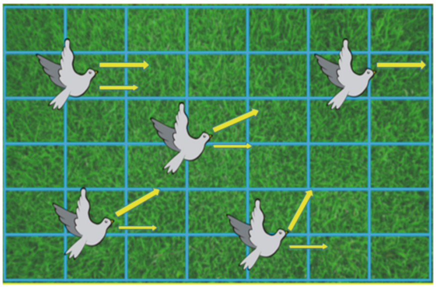

Fig. 1. Map and compass operator model of PIO

Landmark Operator

In the landmark operator, half of pigeons is decreased by N_p in every generation. However, the pigeons are still far from the destination, and they are unfamiliar with the landmarks. Let X_c be the center of some pigeons' position at the t-th iteration, and suppose every pigeon can fly straight to the destination. The position updating rule for pigeon i at t-th iteration can be given by:

$$N_p(t) = \frac{N_p(t-1)}{2} \tag{8}$$

$$X_c(t) = \frac{\sum_{N_p} X_i(t) f(X_i(t))}{\sum_{N_p} f(X_i(t))} \tag{9}$$

$$X_i(t) = X_i(t-1) + r_2(X_c(t) - X_i(t-1)) \tag{10}$$

where r_2 is a random number and f is the quality of the pigeon individual. For maximum problems, $f = f(x)$, for minimum problems, $f = \frac{1}{f(x) + \varepsilon}$, where ε is a constant and $f(x)$ is the cost function.

As shown in Fig. 2, the center of these pigeons is their final destination. Half of the pigeons (pigeons out of the circle) will follow the pigeon, which are close to their destination. The pigeons, which are close to their destination (pigeons in the circle), will fly to their destination very quickly.

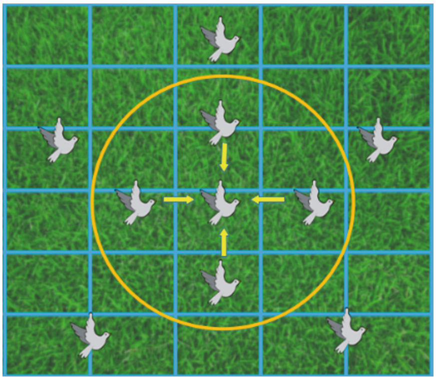

Fig. 2. Landmark operator model in PIO

3.2 Mixed Game Theory

Mixed Strategy Nash Equilibrium

A mixed strategy is a strategy consisting of possible moves and a probability distribution (collection of weights) which corresponds to how frequently each move is to be played. A player would only use a mixed strategy when he is indifferent between several pure strategies, and when keeping the opponent guessing is desirable - that is, when the opponent can benefit from knowing the next move [7].

If each player in an n-player game has a finite number of pure strategies, then there exists at least one equilibrium in (possibly) mixed strategies. If there are no pure strategy equilibria, there must be a unique mixed strategy Nash equilibrium. However, it is possible for pure strategy and mixed strategy Nash equilibria to coexist [8].

Playing the Field

The concept of an 'unbeatable strategy' or an 'evolutionarily stable strategy' is extended to cases in which the payoff to an individual adopting particular strategy depends, not on the strategy adopted by one or a series of individual opponents, but on some average property of the population as a whole, or some section of the population [9, 10].

3.3 Pigeon-Inspired Optimization Integrated with Mixed Game Theory

In the mixed game theory, players can choose different strategies with some kind of probability rather than pure strategies. The basic PIO model is improved by combining mixed game theory (MGPIO) to increase the diversity of the population and improve the feasibility and accuracy of solving the problem of UAS swarm formation.

The velocity and position of pigeon i will be updated as follows:

$$V_i(t) = V_i(t-1)e^{-Rt} + s \cdot r_1 \left(X_g - X_i(t-1)\right) + (1-s) \cdot r_2 (X_c - X_i(t-1)) \quad (11)$$

$$X_i(t) = X_i(t-1) + V_i(t) \quad (12)$$

where $s = 1\, or\, 0$, which indicates the pigeon's available strategies (following the best pigeon or following the center of the pigeons' position). r_1 and r_2 are random numbers between (0, 1). The probability matrix of the pigeons is defined as:

$$\Pi = \left(\frac{p_1}{p_1 + p_2} \quad \frac{p_2}{p_1 + p_2} \right) \tag{13}$$

$$p_1 = Q_1 \cdot f\left(X_g\right) \tag{14}$$

$$p_2 = Q_2 \cdot f(X_c) \tag{15}$$

where $Q_i, i \in \{1, 2\}$ represent the ratio of strategy i at last iteration, f denotes the fitness value of the position X_g or X_c.

Table 1. Procedure of MGPIO

Step 1	Set parameters and initialize the pigeons' position and velocity
Step 2	Calculate each pigeon's fitness value. Determine the best pigeon's position X_g and center of the pigeons' positions X_c
Step 3	According to Eqs. (13)–(15), fill the probability matrix and decide the strategy in t-th iteration
Step 4	Update positions and velocity. Determine the current optimum solution
Step 5	If $Nc < Nc_{max}$, go to Step 2. Otherwise output the best found solution

Table 1 shows the procedure of the proposed MGPIO.

3.4 Computation Complexity of MGPIO

From the mathematical description of the MGPIO algorithm, the computation complexity of the algorithm can be calculated as follows: Time complexity of the map and compass operator or the landmark operator on one generation is $O\left(DN_p\right)$ because the MGPIO algorithm need to use (11) (12) to update every dimensionality of every pigeon. Since the number of iterations is N_c, we can sum them up and find out the computation complexity of the algorithm which is $O\left(DN_pN_c\right)$.

4 Implementation of UAS Swarm Formation Control

4.1 Process of UAS Swarm Formation

The specific process of UAS swarm formation based on MGPIO is as Table 2.

In summary, the basic idea of the inner and outer loop control method to solve the UAS swarm formation control is: The outer loop controller takes the current cluster

Table 2. Process of UAS swarm formation

Step 1	The current leader drone control input is given and get the status output
Step 2	Use the artificial potential function (3) and get the expected position
Step 3	Use the inner loop controller based on MGPIO and get the control input of the wing-man drones in the UAS and the status output
Step 4	Go to step 1 until the termination condition is reached

state as the controller input, and its output is the expected state at the next moment. At the same, it also provides an optimization target for the inner loop controller. The purpose of improving the MGPIO is to find the optimal control input, so that the difference between actual state and the expected state of the next moment is as small as possible. In the case that the outer loop controller continuously provides the expected state, the inner loop controller continuously solves the corresponding input, and so on, to solve the problems of UAS swarm formation control.

4.2 Comparative Experimental Results

In order to evaluate the performance of our proposed MGPIO algorithm and the effectiveness of UAS swarm formation, a series of experiments compared with basic PIO algorithm and PSO algorithm are conducted in MATLAB R2018a programming environment on a PC with 2.50 GHz CPU.

Assume that there are 6 drones in the swarm, including 1 leader and 5 wingmen. Figures 3, 4 and 5 shows the simulation results when using the MGPIO algorithm.

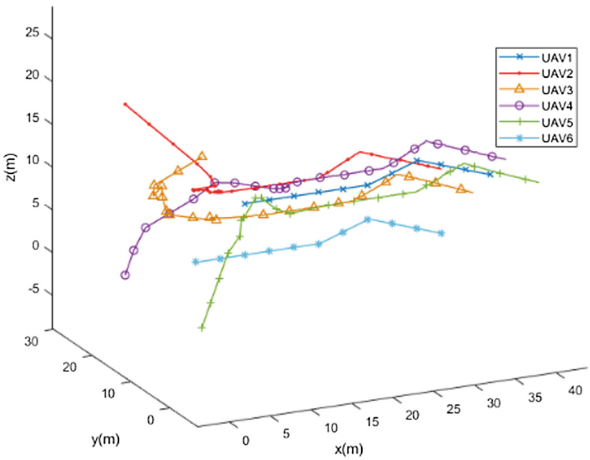

Fig. 3. Simulation result in a 3-D view

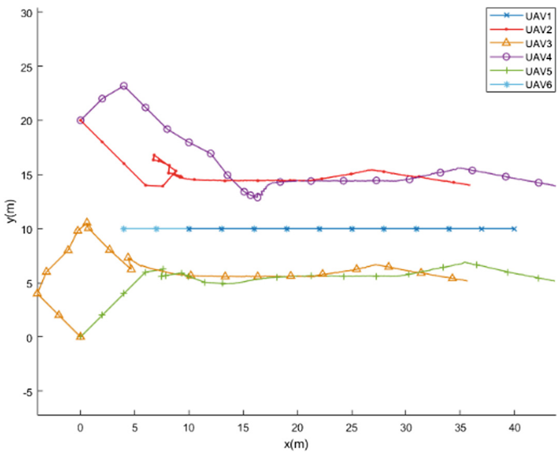

Fig. 4. Simulation result in a top-down view

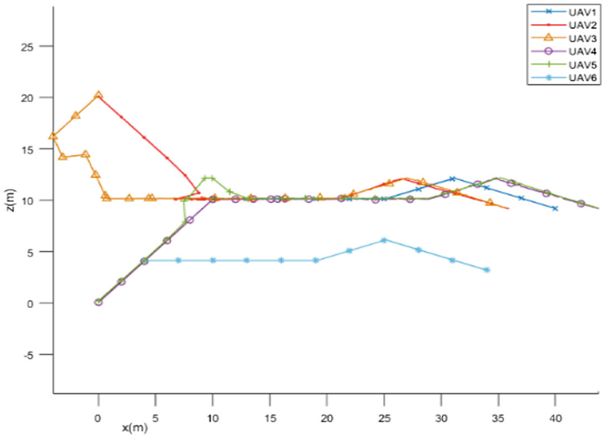

Fig. 5. Simulation result in a side view

From Figs. 3, 4 and 5, MGPIO algorithm could form a stable formation. This is because the MGPIO algorithm have faster convergence speed and it is better at avoiding local minimum. Simulation and comparison experiments verify the feasibility and effectiveness of the proposed method. The comparative evolutionary curves of MGPIO with basic PIO, PSO in artificial potential function (3) is showed in Fig. 6. From evolution curves of three algorithms, it shows MGPIO converged faster than basic PIO and PSO algorithm and the final result of MGPIO is better than the other two algorithms.

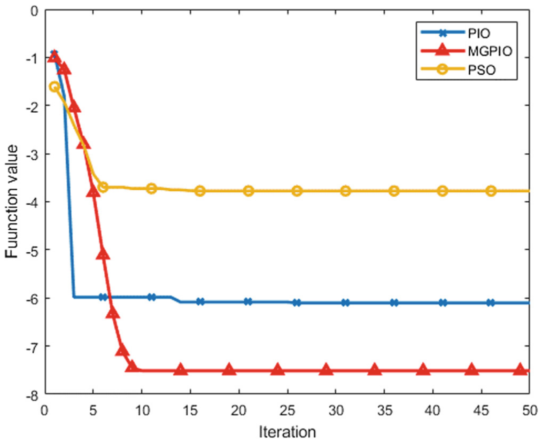

Fig. 6. Comparative evolutionary curves of MGPIO, PIO and PSO

5 Conclusions

The UAS swarm formation is a challenging technical problem. This paper uses the inner and outer loop control to design a UAS swarm formation controller. The outer loop controller selects the artificial potential field function and the mixed game pigeon-inspired optimization algorithm is introduced as a parameter regulator for the inner loop controller. At the same time, the simulation and comparison experiments verify the feasibility and effectiveness of the proposed method, and verify the effectiveness of the MGPIO algorithm by comparing the effects of the UAS swarm form under different inner loop controllers.

References

1. Tzafestas, S., Dalamagkidis, K., Piegl, L., Valavanis, K.: On Integrating Unmanned Aircraft Systems into the National Airspace System. ISCA 2009, vol. 36. Springer, Dordrecht (2009). https://doi.org/10.1007/978-1-4020-8672-4
2. Duan, H., Qiao, P.: Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning. Int. J. Intell. Comput. Cybern. **7**, 24–37 (2014)
3. Xu, X., Deng, Y.: UAV power component-DC brushless motor design with merging adjacent-disturbances and integrated-dispatching pigeon-inspired optimization. IEEE Trans. Magn. **54**, 1–7 (2018)
4. Duan, H., Wang, X.: Echo state networks with orthogonal pigeon-inspired optimization for image restoration. IEEE Trans. Neural Netw. Learn. Syst. **27**, 2413–2425 (2016)
5. Zhang, B., Duan, H.: Three-dimensional path planning for uninhabited combat aerial vehicle based on predator-prey pigeon-inspired optimization in dynamic environment. IEEE/ACM Trans. Comput. Biol. Bioinf. **14**, 97–107 (2017)
6. Duan, H., Qiu, H., Fan, Y.: Unmanned aerial vehicle close formation cooperative control based on predatory escaping pigeon-inspired optimization. Scientia Sinica Technologica **45**, 559–572 (2015)

7. Harsanyi, J.: Games with randomly disturbed payoffs: a new rationale for mixed-strategy equilibrium points. Int. J. Game Theory **2**, 1–23 (1973)
8. Ungureanu, V.: Sets of Nash equilibria in polymatrix mixed-strategy games. Pareto-Nash-Stackelberg Game and Control Theory. SIST, vol. 89, pp. 57–81. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-75151-1_3
9. Sandholm, W.: Evolutionary game theory. In: Computational Complexity, pp. 1000–1029 (2012)
10. Smith, J.: Evolution and the Theory of Games. Cambridge University Press, Cambridge (1982)