REGULAR RESEARCH PAPER

CrossMark

Close formation flight of swarm unmanned aerial vehicles via metric-distance brain storm optimization

Haibin Duan[1] · Daifeng Zhang[2] · Yuhui Shi[3] · Yimin Deng[1]

Abstract

Close formation flight of swarm unmanned aerial vehicles (UAVs) has drawn much attention from scholars due to its significant importance in many aspects. In this paper, we focus on an advanced controller design for swarm UAV close formation based on a novel bio-inspired algorithm, i.e., metric-distance brain storm optimization (MDBSO). The proposed method utilizes the brain storm optimization (BSO) which has been extensively adopted in complicated systems with great performances and modifies its basic operators to formulate the formation flight controller design. The original clustering operator in BSO is replaced by a fresh clustering method based on metric distances, while the individual updating operator utilizes Lévy distribution to extend search steps to fit into the metric searching regions. Then the proposed algorithm is applied to optimize the benchmark controller in swarm UAV close formation to enhance the tracking performances under complicated circumstances. Simulation results demonstrate that our approach is more superior in stable configuration of swarm UAV close formations by comparing with several generic methods.

Keywords Close formation flight · Brain storm optimization · Metric distance · Clustering operation

1 Introduction

It's well known that swarm unmanned aerial vehicles (UAVs) can leverage the capabilities and performances in complex missions via cooperation and formations among different

✉ Haibin Duan
hbduan@buaa.edu.cn

Daifeng Zhang
zdfskh@buaa.edu.cn

Yuhui Shi
shiyh@sustc.edu.cn

Yimin Deng
ymdeng@buaa.edu.cn

1 State Key Laboratory of Virtual Reality Technology and Systems, School of Automation Science and Electrical Engineering, Beihang University, Beijing 100083, People's Republic of China

2 School of Automation Science and Electrical Engineering, Science and Technology on Aircraft Control Laboratory, Beihang University, Beijing 100083, People's Republic of China

3 Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, People's Republic of China

platforms to maximize the team benefits [1]. Sometimes flying in close formation like geese or pigeon flocks may even minify the fuel cost of the whole team and enhance the abilities of self-protection through confusing the enemies once the swarm falls into a trap. Recent studies have shown that birds in close formation are endowed with the capability to perceive the up-wash vortex generated by leaders and prone to reduce their drag forces to expand the flight range. The trajectory observation data of bird flocks corroborated that following birds could experience 11.4–14.5% energy savings when flying in the up-wash vortex of preceding leaders and forming tight formation [2]. Enlightened by this spectacular natural phenomenon, many researchers have concentrated on the control and realization of close formation of aircrafts [3–8]. In [3], real flight tests have been conducted to analyze the benefits of close formation on Dryden F/A-18 platforms under certain conditions. In terms of flight control, Proud et al. designed an autopilot for the following aircrafts to maintain the geometry of close formation where the additional up-wash coupling aerodynamics of wing followers was considered [4]. This controller was based on feedback errors and proportional control scheme which has been considered as a classical templet due to its simple structure. After that, other advanced methods are continuously presented for UAV

close formation flight including the robust techniques [7] and artificial potential functions [8] etc. The main objectives are either improving the position tracking precision of following UAV to gather a perfect up-wash effect or keeping stability of formation configuration for safe flight.

Recently swarm intelligence (SI) algorithms especially the bio-inspired optimizations have caught sustained attentions from enormous researchers in various fields and industries. Typical instances include Particle Swarm Optimization (PSO) [9], Pigeon-inspired Optimization (PIO) [10–12] and Differential Evolution (DE) [13] etc. The diversity of swarm intelligence optimizers has reinforced the freedom of choice for the users in different research domains and thus extended applications in multicolored industries. During this period, a novel bio-inspired algorithm named Brain Storm Optimization (BSO) [14–21] which is first proposed by Shi has been widely used to cope with multiple research problems including complex multi-objective engineering problems. Theoretical analyses such as the convergence proof [15], parameter investigation [16] and population diversity [17] have also been developed recently. However, there are still spaces for BSO to be improved and various modified versions spring up in succession. For example, Qiu et al. improved the original BSO via a brand-new clustering standard for receding horizon control of multiple UAV formation flight [18].

In terms of the BSO structure, the clustering operation is absolutely essential for the whole process and issues on increasing the clustering efficiency are always discussed in recent literatures. Cheng et al. [21] demonstrated that the cluster information in BSO reveals the distribution of solutions, which could be utilized to reveal the landscapes and other proprieties of optimization problems. However the normal clustering operators in BSO and most modified versions are always based on the topological-distance framework [22–26] which indicates that the individuals always interact with neighbors in compact and dense clusters. The traditional topological distance-based clustering method may easily lead to prematurity when all clusters swarm into a local optimal point and thus impact the convergence to global optima. On the contrary, the metric-distance neighborhood [23–26], which stands for the neighborhood with fixed radius and fixed space domain, may have the ability to overcome the imperfections of topological-distance clustering operations and to improve the BSO performances accordingly.

The main attempts of this paper include two aspects: (1) Metric-distance rules are adopted to replace the original topological-distance rules in general BSO clustering strategy and a new improved BSO algorithm is derived. In accordance with the new cluster operations, the step sizes which are subject to Gaussian distribution during individual generations are replaced by the Lévy distribution steps to extend the search lengths in metric-distance subspaces. (2) The proposed metric-distance BSO (MDBSO) is applied

to configure the swarm UAVs' close formation and to optimize the parameters of the baseline formation controller to obtain a great precision and stability for formation flight. The rest of paper is organized as follows. Section 2 introduces the swarm UAVs' model and a simple proportional-integral (PI) strategy for basic control in close formation flight. In Sect. 3, the general BSO are introduced and the MDBSO is described in detail. In Sect. 4, the MDBSO-based formation controller is implemented by optimizing the control parameters via MDBSO, and comparative simulations with some homogenous methods are provided and discussed accordingly. Finally Sect. 5 will give a conclusion to this paper.

2 Modeling of swarm UAVs in close formation flight

In this section, a classical model [4] is introduced to formulate the swarm UAV close formation in leader–follower form (to be consistent with the leader-follower concept, we will use follower UAV instead of following UAV, and leader UAV instead of leading UAV in the remaining of this paper). Both the follower and leader aircrafts are modeled using Flight Control System (FCS) structure whereas additional aerodynamics are also involved in follower model due to the up-wash effects caused by the leader. Then a basic PI controller is given to hold the formation while the corresponding essential parameters are stressed in the end.

2.1 Classical UAV model with FCS for leader

Without loss of generality, it is assumed that FCS is equipped on each aircraft that involves the airspeed-hold, heading-hold and height-hold autopilots [4]. Hence the motion and dynamics are easier to depict without considering complicated aero forces. The coordinate system is given in Fig. 1 where a global North-East-Down (NED) coordinate and a follower-bound relative coordinate are both involved. Taking the global frame and FCS into consideration, the classical UAV model of the leader aircraft can be described by

$$
\begin{aligned}
\dot{x}'_L &= V_L \cos \psi_L \\
\dot{y}'_L &= V_L \sin \psi_L \\
\dot{V}_L &= -\frac{1}{\tau_V} V_L + \frac{1}{\tau_V} V_{Lc} \\
\dot{\psi}_L &= -\frac{1}{\tau_\psi} \psi_L + \frac{1}{\tau_\psi} \psi_{Lc} \\
\ddot{z}'_L &= -\left(\frac{1}{\tau_a} + \frac{1}{\tau_b} \right) \dot{z}'_L - \frac{1}{\tau_a \tau_b} z'_L + \frac{1}{\tau_a \tau_b} z'_{Lc}
\end{aligned} \tag{1}
$$

where x'_L, y'_L, z'_L illustrate the global positions in NED frame and V_L, ψ_L stand for the airspeed value and orientation in
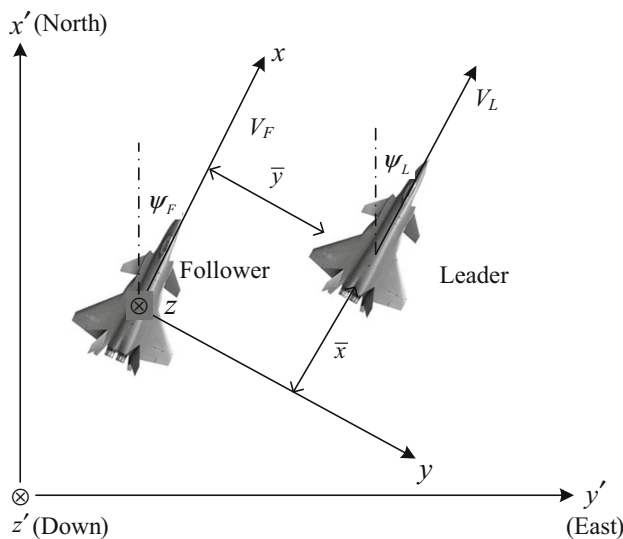
Fig. 1 Close formation geometry and coordinates with leader-follower configuration

global coordinate. The commands of airspeed-hold, heading-hold and height-hold autopilots are given by $V_{Lc}, \psi_{Lc}, z_{Lc}$. Note that the direction of z'_L is reversed with regard to the height that is $z'_{Lc} = -h_{Lc}$. The constant factors $\tau_V, \tau_\psi, \tau_a$ and τ_b denote the corresponding time delays in each autopilot. Since the up-wash effects don't affect the leader, the general model above is used to formulate the leader aircraft but not the followers.

2.2 Follower model with coupling up-wash effects

It is well known that the follower aircrafts are successively influenced by the aerodynamic interactions generated by the up-wash and side-wash airflow from the leader aircraft during the close V-shape formation flight [2–6]. Hence the motion dynamics of followers are changed by additional aerodynamic forces which shall be described by the relative geometries between leader and follower. Thus in this paper, the follower aerodynamic model is deduced on the basis of a follower-bound relative coordinate as shown in Fig. 1.

The relative frame attached to the follower is a rotating right-handed frame in which the forward axis always points to the airspeed. The vector $[x, y, z]$ denotes the relative positions from the leader to the follower in terms of the relative coordinate, and $[\bar{x}, \bar{y}, \bar{z}]$ represents the expected relative distances accordingly. The relative distances can be obtained through rotation from the global frame. According to literature [4], when flying in specific geometries, the follower can exploit the leader's vortex to reduce the drags and benefit from less fuel consumption. On the other hand, it's broadly believed that the individuals in close formation can obtain an optimal spacing when $\bar{x} = 2b, \ \bar{y} = \pi/4 \cdot b, \ \bar{z} = 0$, where

b is the wingspan of the corresponding leader aircraft [4–6]. Then on the condition of optimal formation geometry, the motion dynamics can be depicted as

$$\dot{x} = -\frac{\bar{y}}{\tau_\psi} \cdot \psi_F - V_F + V_L \cos(\psi_L - \psi_F) + \frac{\bar{y}}{\tau_\psi} \cdot \psi_{Fc}$$
$$+ \bar{y} \cdot \frac{qS}{mV_F} \cdot (\Delta C_{Y_y} \cdot y + \Delta C_{Y_z} \cdot z)$$
$$\dot{y} = \frac{\bar{x}}{\tau_\psi} \cdot \psi_F + V_L \sin(\psi_L - \psi_F) - \frac{\bar{x}}{\tau_\psi} \cdot \psi_{Fc}$$
$$- \bar{x} \cdot \frac{qS}{mV_F} \cdot (\Delta C_{Y_y} \cdot y + \Delta C_{Y_z} \cdot z)$$
$$\dot{z} = \zeta$$
$$\dot{V}_F = -\frac{1}{\tau_V} \cdot V_F + \frac{1}{\tau_V} \cdot V_{Fc} + \frac{qS}{m} \cdot \Delta C_{D_z} \cdot z$$
$$\dot{\psi}_F = -\frac{1}{\tau_\psi} \cdot \psi_F + \frac{1}{\tau_\psi} \cdot \psi_{Fc} + \frac{qS}{mV_F} \cdot (\Delta C_{Y_y} \cdot y + \Delta C_{Y_z} \cdot z)$$
$$\dot{\zeta} = -\left(\frac{1}{\tau_a} + \frac{1}{\tau_b}\right) \cdot \zeta - \frac{1}{\tau_a \tau_b} \cdot z + \frac{1}{\tau_a \tau_b} \cdot z_c + \frac{qS}{m} \cdot \Delta C_{L_y} \cdot y$$

$$(2)$$

where m, q and S denote the corresponding mass, dynamic pressure and wing area of the follower aircraft and ζ is an intermediate variable in vertical control. Due to the additional effects from the up-wash induced speed of the trailing vortex of the leader aircraft, additional forces are generated by adding the coupling forces which can be calculated by the current location and auxiliary stability derivatives $\Delta C_{Y_y}, \ \Delta C_{Y_z}, \ \Delta C_{D_z}, \ \Delta C_{L_y}$ [4]. The F-16 model in close formation is adopted in this paper and detail values of its configuration are given in Table 1.

2.3 PI controller and adjustable parameters for formation flight

In the previous research [4], a simple PI controller with a few state feedbacks was proposed to make up a stable multi-UAV close formation. As is illustrated in Eq. (3), in order to overcome the coupling effects caused by leader aircraft, a linear mixer was used to decouple the dynamic equations by the feedback compensation of homogenous states in the same channels. For example, the lateral position y mainly influences the orientation channel compared with other channels, so it's reasonable to implement both y and heading error feedbacks to the heading controller. Obviously the tracking responses are fully determined by the adjustable PI gains $k_x, \ k_y, \ k_V$ etc. Thus how to design a perfect strategy to optimize those parameters is an essential problem in this paper. This will be discussed in Sect. 4 where an optimal controller will be proposed.

$$e_x = k_x(\bar{x} - x) + k_V(V_L - V_F)$$
$$e_y = k_y(\bar{y} - y) + k_\psi(\psi_L - \psi_F)$$

Table 1 F-16 aircraft characteristic values in desired close formation

Wing area	S	300 ft^2	Altitude	h	45,000 ft
Wingspan	b	30 ft	Dynamic pressure	q	155.8 lb/ft^2
Mass	m	25,000 lb	Stability derivatives	ΔC_{Y_y}	0.0033
Velocity time delay	τ_V	5 s	Stability derivatives	ΔC_{Y_z}	-0.0011
Heading time delay	τ_ψ	0.75 s	Stability derivatives	ΔC_{D_z}	-0.000782
Altitude time delay	τ_a	0.3075 s	Stability derivatives	ΔC_{L_y}	-0.0077
Altitude time delay	τ_b	3.85 s	Expected forward space	$\bar{x}$	60 ft
Airspeed	V	825 ft/s	Expected lateral space	$\bar{y}$	23.6 ft

$$e_z = \bar{z} - z$$

$$V_{Fc} = k_{x_P} e_x + k_{x_I} \int_0^t e_x \mathrm{d}t$$

$$\psi_{Fc} = k_{y_P} e_y + k_{y_I} \int_0^t e_y \mathrm{d}t$$

$$z_c = k_{z_P} e_z + k_{z_I} \int_0^t e_z \mathrm{d}t \tag{3}$$

3 Metric-distance brain storm optimization

Known as an important swarm intelligence algorithm, BSO has been widely used to solve complex and multi-objective problems in both theoretical and practical situations [20]. However the searching ability of BSO for global solutions is somewhat limited because of its prematurity after a few iterations. In this section, a modified BSO is proposed with fresh clustering schemes to improve the global convergence accuracy, and the advanced method will also be applied to obtain an optimal controller in the next section.

3.1 General BSO algorithm

BSO is intuitively enlightened by the brainstorming process of human beings in which a group of persons consistently share and generate manifold ideas to integrate and produce a best idea for a difficult and challenging problem [14–20]. Motivated by the general rules of brainstorming process, three basic operations are developed to constitute the BSO including the population initialization, the individual clustering, and the individual generation [14].

Similar to most SI algorithms, BSO begins with the random initialization of individuals and the uniform distribution is selected to generate the swarm evenly located in the search space. Then the clustering operator divides the whole population into several clusters in terms of the individual locations to reinforce the distinction of population divergence. In the general BSO, k-means clustering method [14] is applied, where clusters are identified by the principle of proximity defined by the Euclidean distances accordingly. The cluster center is evaluated as the individual with the best fitness value in

each cluster. In order to extend the population diversity, a randomly selected center may be replaced by a new individual randomly generated in the whole search space with a constant probability p_{5a}. The individual generation is a paramount step in population evolution where one or two clusters are selected to evolve and generate new individuals. Whether one or two clusters are evolved is determined with a probability p_{6b}. Moreover during new individual generation, two sources shall be chosen to create new individuals either from cluster centers or from random individuals inside the clusters in both situations. The selection for the generation sources in one cluster evolution is determined with the probability p_{6biii} while the sources are chosen by the probability p_{6c} in two cluster situation. The probability p_{6bi} determines which cluster will be evolved in one cluster evolution and it is proportional to the number of individuals in this cluster. If one cluster is selected, the rule for the new individual generation is

$$X_{\text{new}} = X_{\text{selected}} + \xi N(\mu, \sigma) \tag{4}$$

where $N(\mu, \sigma)$ denotes the Gaussian distribution with expectation μ and variance σ. The step length ξ is a coefficient that weights the contribution of the random value which satisfies

$$\xi = \text{logsig}\left(\frac{0.5 N_{\text{cmax}} - N_{\text{c}}}{20}\right) \cdot rand \tag{5}$$

where N_{cmax}, N_{c} and $rand$ represent the maximum index of iterations, index of current iteration and uniformly random number, whereas logsig() is a logarithmic sigmoid transfer function. On the other hand, if two clusters are chosen, the candidate individual changes to

$$X_{\text{selected}} = \omega_1 X_1 + \omega_2 X_2 \tag{6}$$

where ω_1 and ω_2 are constant weights for the impacts of two cluster vectors. Finally evaluating the new individual and the existing individual, the better one will be reserved. If all the existing individuals are updated, then continue to next iteration until the maximum iteration index is arrived.
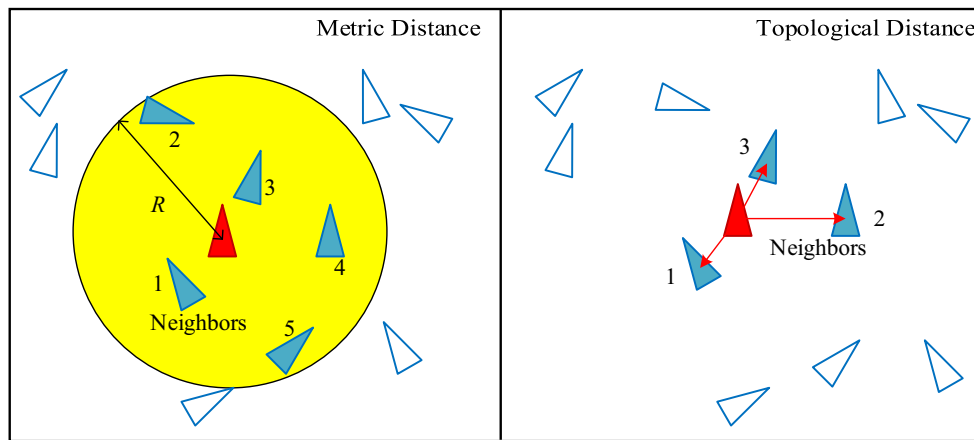
Fig. 2 Neighborhood defined by the metric distance and topological distance

3.2 Metric distance versus topological distance

As shown in Fig. 2, there are two common definitions of distances for the neighborhood in the natural flocking model where individuals frequently interact with their neighbors to reach consensus of the whole flock. One is metric distance and the other is topological distance [22–26]. A typical instance of the metric definition is the classical Reynolds's Boid model [24]. It stipulates a constant range of neighborhood for an individual and the communications are only related to the neighbors. Metric distance tends to attract the populations into a few solid flocks and finally aggregate to consensus by interacting between flocks [24]. As for the topological distance, it is determined by a fixed number of nearest neighbors around an individual, and the range and shape of the neighborhood is changeable with time [22–24]. The topological distances are regularly observed in bird flocks and other animal groups, and are often used to formulate the communication structures of swarm robots.

Figure 2 illustrates the differences between the metric distance and the topological distance with respect to the neighborhood definition. As shown in Fig. 2, the neighborhood of a red individual defined by the metric distance is composed of 5 blue individuals in a fixed yellow circle region surrounding the corresponding red one with a radius of R. However, in the same layout, if we define the number of neighbors in topological distance neighborhood as 3, the 3 nearest blue individuals around the red one become the neighbors and the corresponding area is declined. Hence when we hope to maintain the search scope and avoid the premature in the later iteration, the metric distance neighborhood may provide wider search area than that of topological distance.

Since the general BSO uses the k-means algorithm to generate the clusters which is similar to a topological distance based approach, the individuals are prone to aggregate with high density. However when the clusters swarm into a local optima, it's difficult to jump out and the solutions may fall into premature. In this paper, the traditional clustering method will be instead replaced with the metric distance based operation. In order to balance the convergence efficiency in the metric distance regions, Lévy distribution is utilized to update the individuals.

3.3 Metric-distance BSO

Unlike the k-means clustering in the general BSO, MDBSO divides the search space into several subspaces which share an equal and fixed space radius. Then the population is randomly initialized and the clusters in the MDBSO are defined as the individuals located at the subspaces. Note that the number of the clusters is equal to that of subspaces. Then the modified clustering method is presented by the following rules.

$$X_i \in C_j, \quad \forall i = 1, \ldots N, \ j = 1, \ldots M \text{ and } X_i^{d_*} \in (s_j, s_{j+1}]$$

$$s_1 = L^{d_*}, \ s_{M+1} = U^{d_*}, \ s_{j+1} = s_j + \frac{U^{d_*} - L^{d_*}}{M} \qquad (7)$$

where C_j represents the jth cluster and M denote the scope of clusters and subspaces. N stands for the scope of population. d_* is the criterion dimension randomly chosen in each iteration, which is limited in the space dimension D. L^{d*} and U^{d*} stand for the lower bound and upper bound of the whole search space at criterion dimension d_*, and s_j represents the boundary between adjacent subspaces at criterion dimension d_*. As is illustrated in Eq. (7), the clusters are classified into several continuous zones with the same span $\left(U^{d_*} - L^{d_*} \right) /M$ (metric distance). Note that in each loop, the criterion dimension is changed randomly to ensure the homogeneity of search dimension and the diversity of clusters. Meanwhile to guarantee the duration of the algorithm, the number of individuals in each cluster P should be kept at least one.

During the individual generation process, the center is selected as the individual with the best fitness value in each cluster, and the random mutation operation of the cluster centers also occurs with probability p_{5a} which is consistent with the general BSO. However during the one cluster evolution, the cluster selection probability p_{6bi} is replaced by

$$p_{6bi}^j = \begin{cases} \dfrac{\frac{1}{f_{center}^j}}{\sum_{k=1}^{M} \frac{1}{f_{center}^k}}, & \text{for minimization problem} \\[4mm] \dfrac{f_{center}^j}{\sum_{k=1}^{M} f_{center}^k}, & \text{for maximization problem} \end{cases} \quad (8)$$

where f_{center}^j denotes the fitness value of the jth cluster center and p_{6bi}^j stands for the selection probability of the jth cluster. Since the search region is divided uniformly with metric distances, the average search span is larger than that of topological distances as iteration ascends.

In order to balance the search span and convergence speed in metric-distance zones, the heavy-tail Lévy distribution [12, 27] is used to update the step sizes in individual generation operations instead of the Gaussian distribution in the general BSO. As is depicted in the following probabilistic model [12], the element *step* is subject to Lévy distribution which is a nonlinear composition of two elements subject to the Gaussian distribution.

$$step = \frac{\lambda}{|\theta|^{1/\beta}}, \ \lambda \sim N\left(0, \sigma_\lambda^2\right), \ \theta \sim N\left(0, \sigma_\theta^2\right)$$

$$\sigma_\lambda = \left(\frac{\Gamma(1+\beta) \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \beta \cdot 2^{(\beta-1)/2}} \right)^{1/\beta}, \ \sigma_\theta = 1 \quad (9)$$

where σ_λ and σ_θ are variances of corresponding Gaussian distributions, while $\Gamma()$ represents the gamma function. β is a characterized parameter revealing the fluctuation of Lévy distribution. A random sample test has been conducted as shown in Fig. 3 where samples subject to 3 types of Gaussian distributions with different variances and 3 types of Lévy distributions with various parameter β have been utilized. The scale of samples in all random models is taken as 1000. Then respectively the scatter diagram and probability density plot about these sample results are given in Fig. 3a, b. Besides the variance of all the samples are also listed in Table 2. Apparently the Lévy distributions are heavy-tail and share more violent fluctuations than the Gaussian distributions and therefore have higher probability to produce wider search spans. This principle helps to enhance the balance between the exploitation and the exploration of MDBSO when searching in large-scale regions.

Here the Lévy distribution is adopted to generate new individuals in metric-distance search regions and we substitute
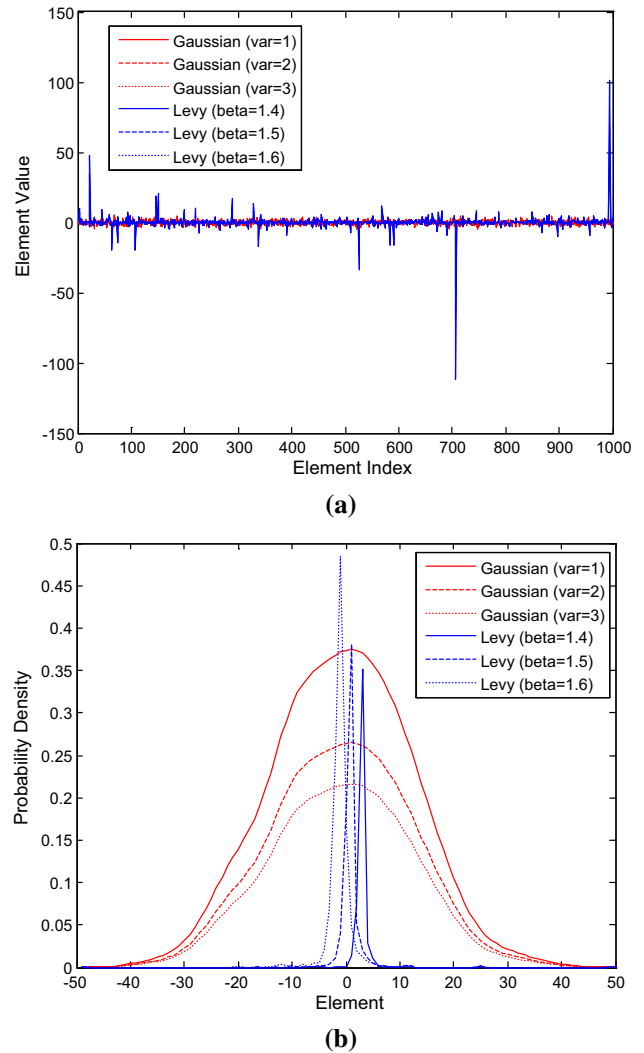
(a)

(b)

Fig. 3 Typical results for the random sample test. **a** Scatter diagram for sample test with different distributions. **b** Probability density for sample test with different distributions

the original method in Eq. (4) in general BSO by the following method.

$$X_{new} = X_{selected} + step \quad (10)$$

$$X_i = \begin{cases} X_{new}, & f(X_{new}) < f(X_i) \\ X_i, & \text{otherwise} \end{cases}, \quad X_i \in C_j \quad (11)$$

where $X_{selected}$ is the candidate individual generated by the cluster evolution and *step* is a step size subject to Lévy distribution. $f()$ denotes the fitness function. The individual generation also abides by the greedy mechanism as shown in Eq. (11). Note that the new individual X_{new} is not necessary to stay within the cluster subspace C_j that the current individual X_i belongs to and the members in each cluster are changeable with iteration ascending accordingly. Therefore to ensure each cluster is not empty during evolution, at least

Table 2 Sample variance list of different distributions

	Type of distributions					
	$N(0, 1)$	$N(0, \sqrt{2})$	$N(0, \sqrt{3})$	$Levy(\beta = 1.4)$	$Levy(\beta = 1.5)$	$Levy(\beta = 1.6)$
Variance	0.9784	1.9568	2.9352	32.9793	15.4708	7.8461

one individual should be reserved in a single metric-distance cluster. As a result, the detailed procedure of MDBSO is listed as the following steps and Fig. 4 provides a visualized flowsheet to summarize the implementation process.

Step 1 Initialize the parameters of MDBSO, including search space dimension and boundaries $\{D, U, L\}$, maximum iteration N_{cmax}, individual population scope N, cluster scope M, fitness function $f()$, series of constant probabilities and other relative parameters β and P etc. Generate the initial population randomly and uniformly in the whole search space.

Step 2 Metric-distance clustering strategy. Select a criterion dimension d_* randomly and divide the whole search space into M subspaces according to Eq. (7). Then rank each individual's location and classify into a corresponding cluster using Eq. (7). Select the cluster center as the best individual within the corresponding subspace.

Step 3 With probability p_{5a}, randomly select a cluster center and replace it with a new idea generated randomly and uniformly in the whole search space.

Step 4 Individual generation. With probability p_{6b}, randomly confirm one or two clusters to participate in the new individual generation.

Step 5 If one cluster is selected, randomly choose a cluster with probability p_{6bi} calculated according to the Eq. (8). With probability p_{6biii}, pick up the cluster center as the selected individual and generate new individual according to the Eqs. (9) and (10). Otherwise, randomly choose an individual within this cluster and run the generation operator as according to the Eqs. (9) and (10). Then go to Step 7.

Step 6 If two clusters are selected, randomly and uniformly choose two clusters, and with probability p_{6c}, combine the two relative cluster centers using the Eq. (6) as the selected individual and generate new individual according to the Eqs. (9) and (10). Otherwise, randomly choose two individuals within two clusters accordingly and run the above generation operator. Then go to next step.

Step 7 If the newly generated individual is superior to the existing individual and at least P members are reserved in the existing cluster, the current individual is replaced by the new individual using the Eq. (11). Otherwise, reserve the current individual.

Step 8 If all individuals have been updated, go to Step 9; otherwise go to Step 4.

Step 9 If the current iteration N_c arrives at the predefined maximum iteration N_{cmax}, terminate and output the global best individual as the global optimal solution; otherwise go to Step 2.

3.4 Complexity analysis

The computational complexity of MDBSO is easily obtained from the cluster and individual generation processes. Since the cluster centers and members are changed after new individual generation, we shall reclassify the population and reevaluate the cluster centers before next individual updating, which is similar with the basic calculation of BSO. In this paper the polling scheme is used to update the clusters, and therefore the complexity of MDBSO in each iteration cycle is $O(N \cdot M)$. On the other hand, assume the calculation cost of fitness function is T_f, the fitness values for the population are also required to recalculate after each iteration. As the complexity of the individual updating operator is also $O(N \cdot M)$, we can summarize that the time complexity of MDBSO is $O(N_{cmax}NM + N_{cmax}NT_f)$. As the time cost grows linearly with the complexity of the fitness function, when the objective model is complicated, the real-time performance may not be satisfied and the algorithm should be adopted offline.

4 MDBSO-based optimal controller design and simulation results

In this section, we apply our proposed MDBSO to the formulation of an optimal control law for a swarm UAV close formation. The optimal controller is based on the basic control approach presented in Sect. 2.3 and the main task for MDBSO is to optimize the crucial parameters of the basic controller to obtain precise tracking and strong stability. Besides, comparative experiments with several homogenous methods are conducted to verify the advantages of MDBSO.

4.1 Optimal controller for close formation flight based on MDBSO

As is mentioned above, the adjustable parameters within the Eq. (3) mainly contribute to the performances of a basic controller. Hence it's possible and necessary to seek out at least
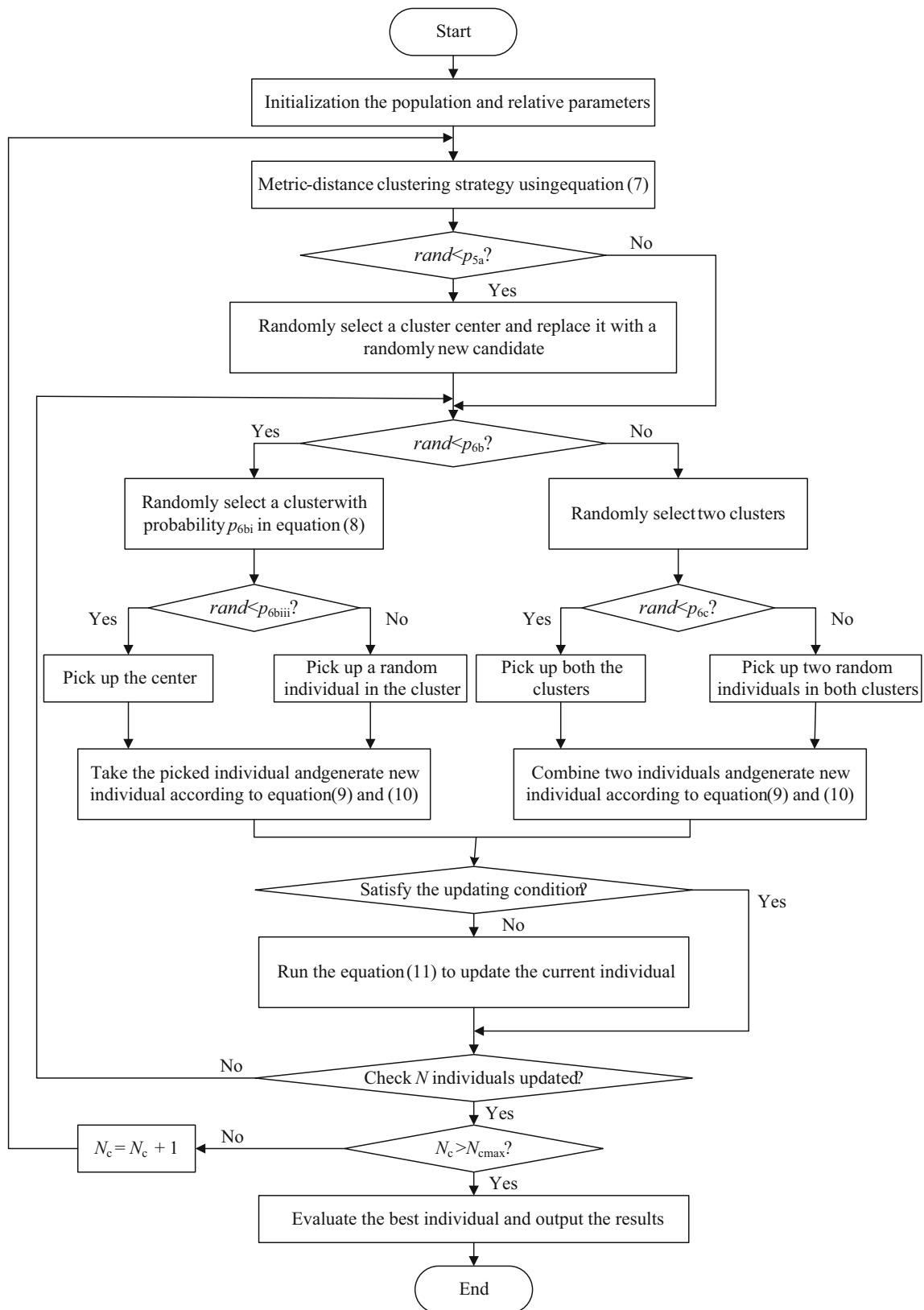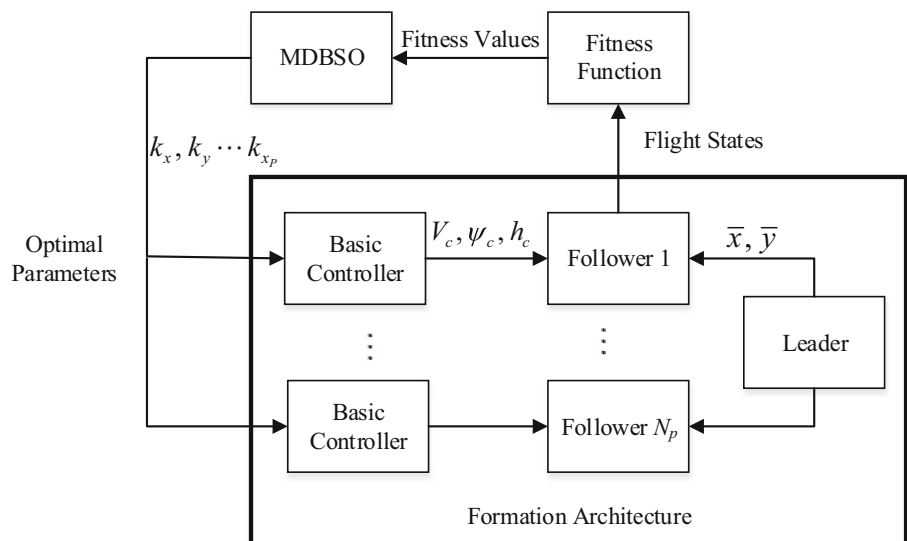
Fig. 4 Flowchart of MDBSO implementation

Fig. 5 Framework of close formation with MDBSO-based optimal controller

a group of satisfied parameters for stable and reliable swarm UAV formation flight. The controller design can turn into an optimization problem when we define a comprehensive fitness function and the novel MDBSO algorithm could be used to find the optimal parameter solutions and thus formulate the optimal controller in close formation. Note that in addition to the PI gains that should be considered, the feedback coefficients k_x, k_y etc. in mixers are also required to optimize.

The framework of the whole system with the optimal controller is shown in Fig. 5 which is composed of two subsystems. The first is the architecture for close formation including the formation model and basic controller equipped at each UAV. The second is the optimal processing module with MDBSO. The fitness function and optimal parameters establish the links between these two subsystems. Therein the fitness function determines the criterion to the parameter optimization and enhancement effectiveness on the basic controller. Here we choose the following error-integral criterion as the fitness function [12] in MDBSO.

$$f = \left(\frac{1}{N_p} \sum_{i=1}^{N_p} \int_0^t \left(e_{x_i}^2 + e_{y_i}^2 + e_{z_i}^2 + e_{V_i}^2 + e_{\psi_i}^2 \right) dt \right)^{1/2} \tag{12}$$

where N_p denotes the total number of follower aircrafts. This criterion is clearly a minimization problem that is composed of the followers' tracking errors of NED positions, airspeeds and headings relative to the formation of the leader. Therefore the objective of MDBSO is to find a group of solutions that minimize the above fitness function. Randomly generate a group of initial parameters of basic controller as the initial solutions and input into the formation architecture, the flight states and fitness value will be simultaneously updated. Con-

ducting the MDBSO processes, the solutions will be evolved until the termination conditions are satisfied. Then the whole system will obtain a group of optimal parameter solutions and likewise an optimized controller.

As for the parameter selection of MDBSO, the manipulation of the probabilities p_{5a}, p_{6b}, p_{6biii}, and p_{6c} are similar with the general BSO, which could be referred to [16]. The criterion dimension d_* is uniformly distributed within $[1, D]$ and selected randomly after each iteration. The minimum number of individuals in a cluster P should be guaranteed at least one, however through multiple experiments, we find a range of $[2, 5]$ is preferable for the convergence precision of MDBSO. On the other hand, according to the random sample test in Sect. 3.3, the characterized parameter $\beta = 1.5$ of Lévy distribution is more reasonable since the corresponding variance is moderate that could balance the expansion of search spans and the fluctuation of search steps.

4.2 Comparative experiments and analysis

In this section, numerical simulations are carried out to verify the advantages of MDBSO-based controller. Comparisons with the general BSO as well as PSO and DE are also implemented throughout experiments. In these simulations, four follower F-16 aircrafts are involved to constitute a V-shape close formation with the expected spaces listed in Table 1. The leader dynamics comply with the following guidance law and the initial states of each aircraft in global frame are given in Table 3.

$$\begin{cases} V_{Lc}(t) = 825 \text{ ft/s} \\ \psi_{Lc}(t) = 30 \cdot \frac{t}{T_{\max}} \text{ deg} \\ h_{Lc}(t) = 45,000 \text{ ft} \end{cases} \tag{13}$$

Table 3 Initial global states of swarm UAVs

	Leader	Follower1	Follower2	Follower3	Follower4
North Pos. (ft)	0	-80	-80	-160	-160
East Pos. (ft)	0	-40	40	-80	80
Altitude (ft)	45, 000	42,000	42,000	42,000	42,000
Airspeed (ft/s)	820	800	800	800	800
Heading (deg)	0	-10	10	-10	10
$\bar{x}$ (ft)	0	60	60	120	120
$\bar{y}$ (ft)	0	23.562	-23.562	47.124	-47.124

Table 4 Configurations for BSO and MDBSO in comparative experiments

Common coefficients							
N	180	D	10	M	6	N_{cmax}	30

BSO and MDBSO parameters									
p_{5a}	0.2	p_{6b}	0.8	p_{6biii}	0.4	p_{6c}	0.5	P	3
ω_1	0.5	ω_2	0.5	μ	1	σ	0	β	1.5

Parameter boundaries of basic controller									
k_x	$[-10, 0]$	k_y	$[-10, 0]$	k_V	$[0, 10]$	k_ψ	$[0, 10]$	k_{x_P}	$[0, 10]$
k_{x_I}	$[0, 1]$	k_{y_P}	$[0, 10]$	k_{y_I}	$[0,1]$	k_{z_P}	$[0, 10]$	k_{z_I}	$[0, 1]$

Table 4 provides a list of configuration coefficients in BSO and MDBSO. The termination time of once flight simulation is $T_{\text{max}} = 10$ s while the sampling period of swarm UAV close formation model is 0.01 s. Moreover the homogenous methods including PSO and DE are also configured that is given in Table 5. On the other hand, in order to verify the robustness and stability of an optimal controller, wind disturbances in the form of the Eq. (14) are also involved which mainly affect the airspeed channel. Two types of wind are considered that are wind turbulence and gust. Wind turbulence emerges with probability $P_T = 0.5$ and performs as cosine fluctuations with maximal magnitude $V_{\text{max}} = 40$ ft/s, while the wind gust V_{gust} is a standard Gaussian white noise. Simulation results are shown in Figs. 6, 7 and 8. The average fitness evolution curves of MDBSO and other methods are displayed in Fig. 6. State tracking curves in forward and lateral channels with MDBSO are displayed in Fig. 7. Figure 8 illustrates a 3D scene graph where 4 followers and 1 leader constitute a V-shape close formation with the guidance of a MDBSO-based optimal controller.

$$V_W (t) = \begin{cases} \frac{V_{\text{max}}}{2} \left(1 - \cos\left(\frac{\pi t}{T_{\text{max}}}\right)\right), & rand\,(t) < P_T \\ 5 \cdot V_{\text{gust}}\,(t), & \text{else} \end{cases} \quad (14)$$

According to Fig. 6, it's demonstrated that the proposed MDBSO contributes to more accurate fitness solutions and faster convergence speed than general BSO and other homogenous algorithms. It is assumed that the reasons for this superior search ability of MDBSO come from two

Table 5 Configurations for PSO and DE in comparative experiments

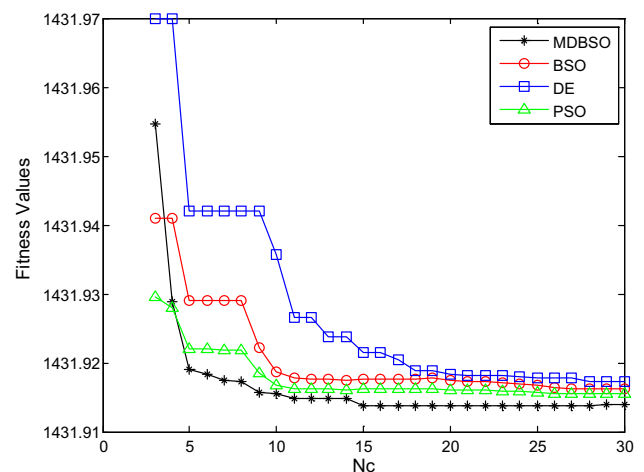| Coefficient	Description	Value
ω	Inertia factor	0.6
c_1	Self-best factor	2
c_2	Global-best factor	2
P_{cr}	Crossover probability	0.7
F	Mutation factor	0.5

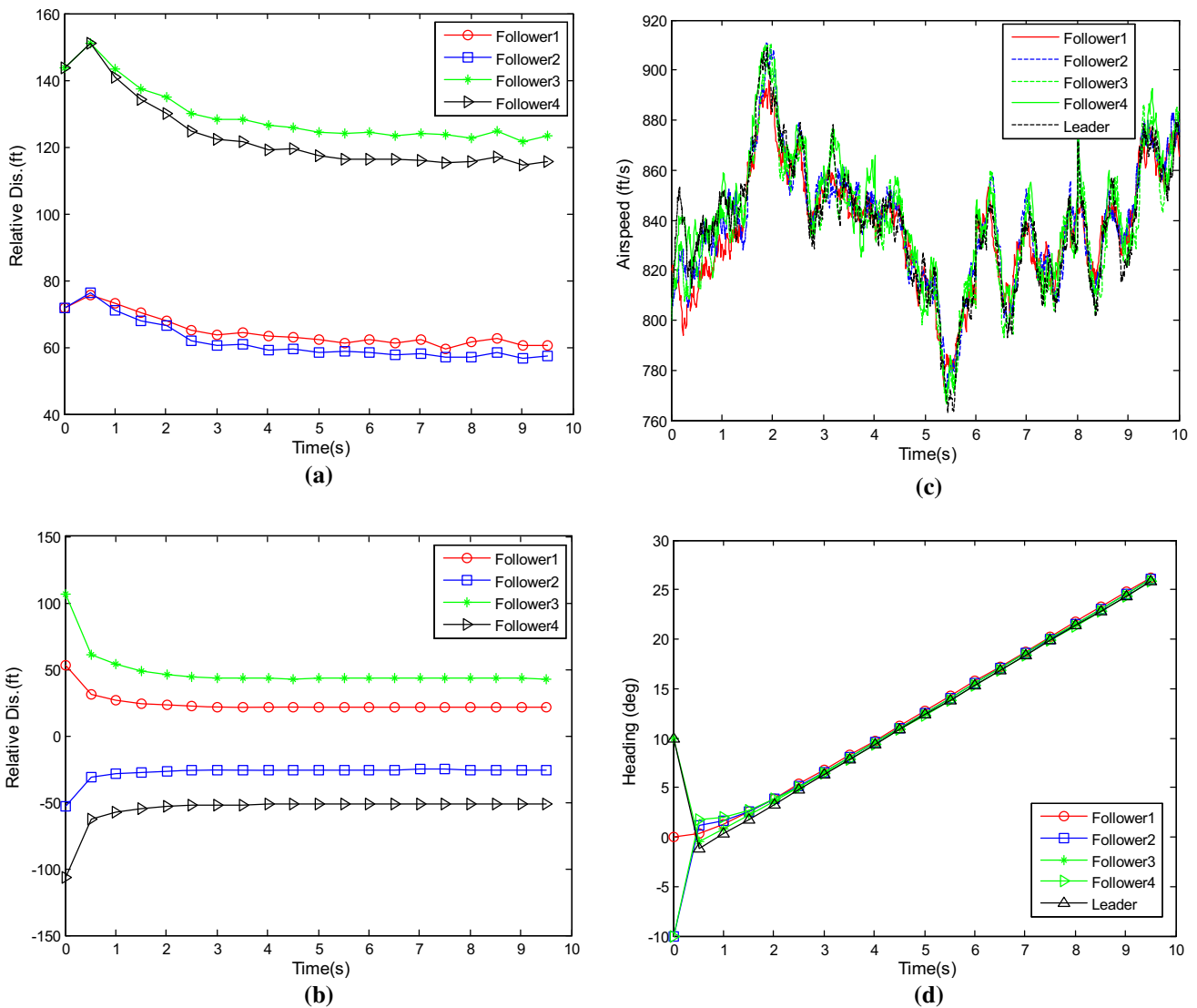**Fig. 6** Average evolution fitness curves of multiple optimizations

Fig. 7 State evolutions in forward and lateral channels with MDBSO-based controller. **a** Tracking trajectories of longitudinal relative distances of multiple following UAVs. **b** Tracking trajectories of lateral relative distances of multiple following UAVs. **c** Airspeed fluctuations of multiple UAVs with wind disturbances. **d** Heading variations of multiple UAVs

aspects. One is the wider average search spans of multiple clusters in metric distance operations which increase the probability to explore more unknown spaces and therefore improve the probability to obtain the global optima. Moreover after a few iterations, clusters in different metric-distance subspaces still enable a high frequency of population exchanges, instead of the weak interactions in topological clusters that general BSO behaves. This mechanism enhances the diversity of individuals in the later period of algorithm and thus improves the large scale search ability. The other is the Lévy distribution search step operator which could generate large scale step sizes and enable fast investigations in wider metric-distance areas. Since the variances of Lévy distribution are higher than that of other typical distribu-

tions, MDBSO could have higher probabilities to generate large-length search steps and enhance the balance between the convergence speed and precision in the metric-distance subspaces, and therefore enhance the accuracy and search speed simultaneously.

What's more, the results in Fig. 7 indicate that although subject to strong wind turbulences and gusts, the MDBSO-based optimal controller still keeps the whole formation robust and solid. Table 6 exhibits the optimal parameters that MDBSO generated. From Fig. 7a, b, it is obvious that the tracking precision of expected distances in both directions is satisfactorily achieved, and we also obtain a fast and stable tracing of the leader heading according to Fig. 7d. In terms of Fig. 7c, although wind disturbances strongly influences
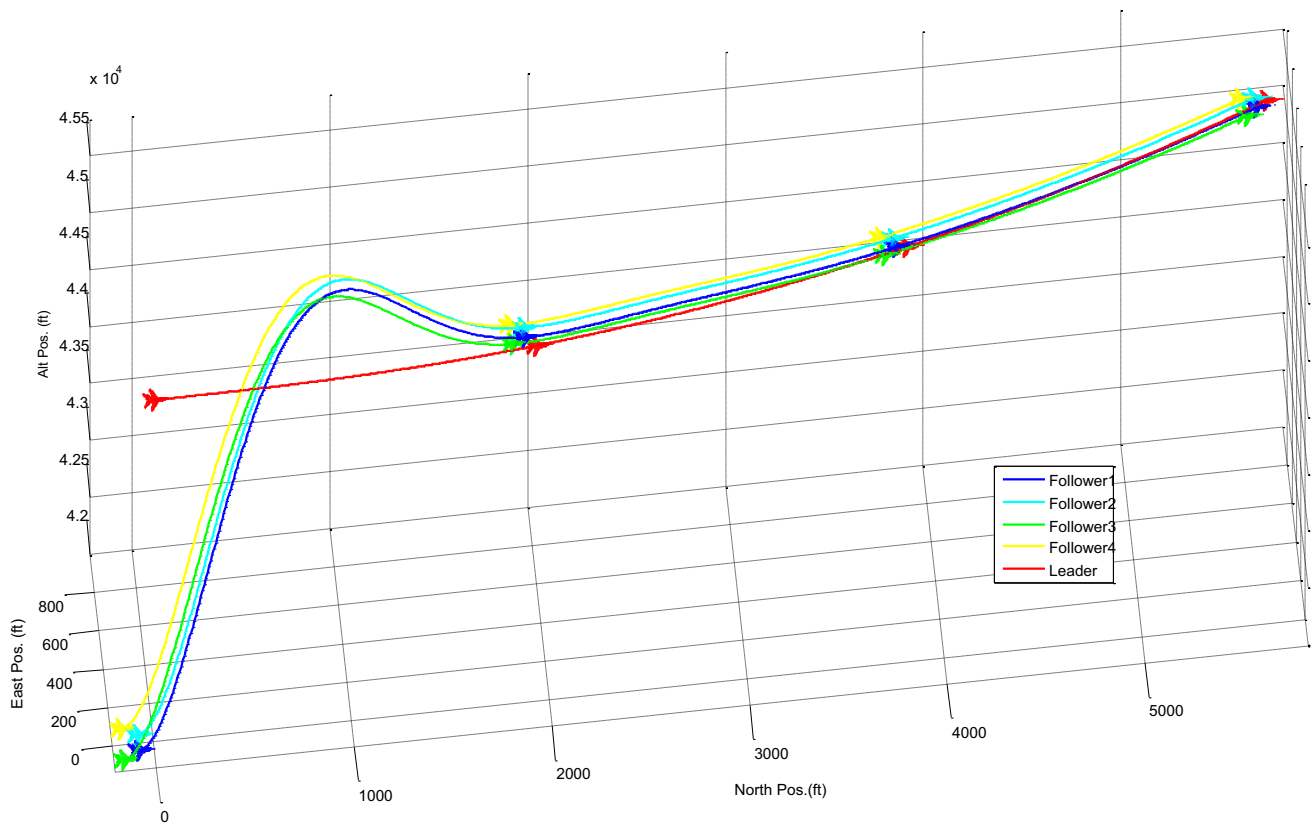
Fig. 8 3D trajectories of V-shape close formation

Table 6 Optimal parameters for formation control via MDBSO

k_x	k_y	k_V	k_ψ	k_{x_P}
-9.7451	-0.6900	9.5771	5.9970	7.0000
k_{x_I}	k_{y_p}	k_{y_I}	k_{z_P}	k_{z_I}
0.5468	9.8000	0.6400	9.7689	0.0300

the airspeed of the leader UAV and increase the difficulty of stable retention, the follower UAVs still keep pace with the leader with slight fluctuations. Hence the robustness and strong stability of the proposed controller are verified. Finally Fig. 8 presents a vivid and distinct simulation spectacle where all swarm UAVs are able to keep up with a stable V-shape close formation under strong unknown circumstances.

5 Conclusions

In this paper, a novel metric distance based brain storm optimization algorithm has been developed and applied for the stable control of swarm UAV close formation flight. With reference to a previous research, a classical feedback controller has been utilized and the concentration in this paper

is focused on the optimal strategy for parameter adjustment via the proposed method.

Although BSO has been widely used in solving almost all kinds of scientific puzzles, there are still some limitations intuitively. Since the clustering benchmark of a general BSO is based on topological distance-based dividing approach, which however declines the search ability in large-scale areas in later iteration period, a metric-distance clustering mechanism has been used to improve the BSO's performances. In MDBSO, the clusters are partitioned in terms of fixed search subspaces and the average search span for each cluster is wider than that of a general BSO. Meanwhile the heavy-tail Lévy distribution has been utilized to generate new individuals for the sake of accelerating the search speed and balancing the equilibrium between exploitations and explorations in the metric-distance subspaces. Comparative simulation results have illustrated that MDBSO is capable of generating optimal parameters of the basic controller and the MDBSO-based optimal controller is adequate to stable and accurate formation tracking even under rough situations. In the future, the novel algorithm is expected to extend the application in multiple disciplines and the concept of the metric-distance search regions is welcomed to transplant into more advanced intelligent algorithms.

References

1. Karimoddini A, Hai L, Chen BM, Lee TH (2013) A bumpless hybrid supervisory control algorithm for the formation of unmanned helicopters. Mechatronics 23(6):677–688
2. Weimerskirch H, Martin J, Clerquin Y et al (2001) Energy saving in flight formation. Nature 413(6857):697–698
3. Ray R, Cobleigh B, Vachon MJ, John CS (2002) Flight test techniques used to evaluate performance benefits during formation flight. In: AIAA atmospheric flight mechanics conference and exhibit, Monterey, California, 5–8 August, https://doi.org/10.2514/6.2002-4492
4. Proud AW, Pachter M, D'Azzo JJ (1999) Close formation flight control. In: AIAA guidance, navigation, and control conference and exhibit, Portland, 9-11 August, pp 1231–1246
5. Wagner MG, Jacques LD, Blake W, Pachter M (2002) Flight test results of close formation flight for fuel savings. In: AIAA atmospheric flight mechanics conference and exhibit, Monterey, California, 5–8 August, https://doi.org/10.2514/6.2002-4490
6. Zhang Q, Liu Hugh HT (2016) Robust design of close formation flight control via uncertainty and disturbance estimator. In: AIAA guidance, navigation, and control conference, San Diego, California, 4–8 January, https://doi.org/10.2514/6.2016-2102
7. Johnson Y, Dasgupta S (2014) Robust controller design and performance of forward-velocity dynamics of UAVs in close formation flight. In: IEEE international conference on advances in green energy, Trivandrum, Kerala, India, 17–18 December, pp 124–131
8. Palacios L, Ceriotti M, Radice G (2015) Close proximity formation flying via linear quadratic tracking controller and artificial potential function. Adv Space Res 56(10):2167–2176
9. Lalwani S, Kumar R, Gupta N (2015) A novel two-level particle swarm optimization approach for efficient multiple sequence alignment. Memet Comput 7(2):119–133
10. Xin L, Xian N (2017) Biological object recognition approach using space variant resolution and pigeon-inspired optimization for UAV. Sci China Technol Sci 60(10):1577–1584
11. Pei J, Su Y, Zhang D (2017) Fuzzy energy management strategy for parallel HEV based on pigeon-inspired optimization algorithm. Sci China Technol Sci 60(3):425–433
12. Dou R, Duan H (2017) Lévy flight based pigeon-inspired optimization for control parameters optimization in automatic carrier landing system. Aerosp Sci Technol 61:11–20
13. Mininno E, Neri F (2010) A memetic differential evolution approach in noisy optimization. Memet Comput 2:111–135
14. Shi Y (2011) Brain storm optimization algorithm. In: International conference on swarm intelligence, Chongqing, China, 12–15 June, pp 303–309
15. Zhou Z, Duan H, Shi Y (2016) Convergence analysis of brain storm optimization algorithm. In: IEEE congress on evolutionary computation, Vancouver, Canada, 24–29 July, pp 3747–3752
16. Zhan Z, Chen W, Lin Y et al. (2013) Parameter investigation in brain storm optimization, In: IEEE symposium on swarm intelligence, Singapore, 16–19 April, pp 103–110
17. Cheng S, Shi Y, Qin Q et al. (2014) Maintaining population diversity in brain storm optimization algorithm, In: IEEE congress on evolutionary computation, Beijing, China, 6–11 July, pp 3230–3237
18. Qiu H, Duan H (2014) Receding horizon control for multiple UAV formation flight based on modified brain storm optimization. Nonlinear Dyn 78(3):1973–1988
19. Sun C, Duan H, Shi Y (2013) Optimal satellite formation reconfiguration based on closed-loop brain storm optimization. IEEE Comput Intell Mag 8(4):39–51
20. Cheng S, Qin Q, Chen J, Shi Y (2016) Brain storm optimization algorithm: a review. Artif Intell Rev 46:445–458
21. Cheng S, Shi S, Qin Q, Gao S (2013) Solution clustering analysis in brain storm optimization algorithm. In: IEEE symposium on swarm intelligence, Singapore, 16–19 April, pp 111–118
22. Ballerini M, Cabibbo N, Candelier R et al (2008) Interaction ruling animal collective behavior depends on topological rather than metric distance: evidence from a field study. Proc Natl Acad Sci USA 105(4):1232–1237
23. Niizato T, Gunji Y (2011) Metric-topological interaction model of collective behavior. Ecol Model 222:3041–3049
24. Sakamoto Y, Takahashi T (2014) Metric and topological neighborhoods in flocking models. In: International conference on bioinspired information and communications technologies, Boston, 1–3 December, pp 118–121
25. Niizato T, Murakami H, Gunji Y (2014) Emergence of the scale-invariant proportion in a flock from the metric-topological interaction. Biosystems 119:62–68
26. Shang Y, Bouffanais R (2014) Consensus reaching in swarms ruled by a hybrid metric-topological distance. Eur Phys J B 87(12):294
27. Eliazar I (2014) A geometric theory for Levy distributions. Ann Phys 347:261–286