Aerospace Science and Technology 32 (2014) 153–161

Contents lists available at ScienceDirect

Aerospace Science and Technology

www.elsevier.com/locate/aescte

Chaotic predator–prey biogeography-based optimization approach for UCAV path planning

Weiren Zhu, Haibin Duan *

Science and Technology on Aircraft Control Laboratory, School of Automation Science and Electrical Engineering, Beijing University of Aeronautics and Astronautics, Beijing 100191, PR China

ARTICLE INFO

Article history:
Received 12 March 2013
Received in revised form 26 October 2013
Accepted 11 November 2013
Available online 27 November 2013

Keywords:
Uninhabited Combat Air Vehicle (UCAV)
Path planning
Biogeography-Based Optimization (BBO)
Chaos theory
Predator–prey

ABSTRACT

This paper proposes a novel Chaotic Predator–Prey Biogeography-Based Optimization (CPPBBO) approach for solving the path planning problems of Uninhabited Combat Air Vehicle (UCAV). To generate optimal or near-optimal flight path, path planning is a key part of UCAV assignment planning system. The planned path can ensure UCAV avoid hostile threats and safely reach an intended target with minimum fuel cost. An improved biogeography-based optimization algorithm is presented for solving the optimization problem in the path planning process. Biogeography-Based Optimization (BBO) is a new bio-inspired optimization algorithm. This algorithm searches for global optimum mainly through two steps: migration and mutation. To enhance the global convergence of the BBO algorithm, the chaos theory and the concept of predator–prey are adopted to get new search mechanism. The comparative simulation results are given to show that our proposed CPPBBO algorithm is more efficient than basic BBO, CBBO and PPBBO in solving the UCAV path planning problems.

1. Introduction

Military operations are turning to more complex and advance automation technologies for minimum risk and maximum efficiency, a critical piece to this strategy is Uninhabited Combat Air Vehicles (UCAVs), owing to their potential to perform dangerous, repetitive tasks in remote and hazardous environment [7]. Path planning for UCAVs is intended to creating a flight plan to guide the UCAV from its initial position to the pre-arranged destination under specific constraints, it is one of the essential parts of the UCAV research. Series of algorithms have been proposed to solve this complicated optimization problem, for example, Eva Besada-Portas presented a path planner for multiple Uninhabited Combat Air Vehicles (UCAVs) based on evolutionary algorithms (EAs) for realistic scenarios [3]. In Ref. [5], Duan et al. presented an Intelligent Water Drops (IWD) optimization algorithm for solving the single UCAV path planning problems in various combating environments. In Ref. [14], a new vibrational genetic algorithm was developed enhanced with a Voronoi diagram. In Ref. [8], a new variant of PSO, named θ-QPSO algorithm was proposed and has shown its high performance in solving path planning problem for UAV in different known and static threat environments.

Biogeography-Based Optimization (BBO), which is a new evolutionary optimization algorithm based on the science of biogeography for global optimization, was originally presented by Simon [16] in 2008. BBO has some features in common with other population-based optimization algorithm, such as the ability to share information between candidate solutions. However, the BBO also has certain features that differ from other population-based optimization methods. One of the characteristics of BBO is that it maintains solutions from one iteration to the next and improved the solutions by migration [17]. As a popular and competitive optimization approach, BBO has been applied to certain problems. What's more, several variations of BBO have been proposed to improve the optimization performance of the basic BBO and to solve a number of constrained optimization problems. Wenyin Gong et al. extended the original BBO and proposed a Real-Coded BBO (RCBBO) approach for the global optimization problems in the continuous domain, and the mutation operator is integrated into RCBBO. In this way, the diversity of the population can be improved, and the exploration ability of RCBBO can be enhanced [9]. Seyed Habib A. Rahmati et al. developed the migration operators of BBO for searching a solution area of Flexible Job Shop Scheduling (FJSP) problem and introduced BBO algorithm to scheduling area [15]. Ling Wang et al. proposed a hybridized BBO with DE, and a simplex search operator was used to design a hybrid algorithm called SSBBODE, which combines the migration and mutation mechanism to enhance the exploration and the exploitation ability [18]. Xiangtao Li et al. introduced multi-parent crossover to the standard BBO and developed a Multi-Operator Biogeography-Based Optimization (MOBBO) method, which cannot only satisfy a balance of exploration and exploitation but also improve the diversity of population [12].

* Corresponding author. Tel.: +86 10 82317318.
E-mail address: hbduan@buaa.edu.cn (H. Duan).

1270-9638/$ – see front matter
http://dx.doi.org/10.1016/j.ast.2013.11.003

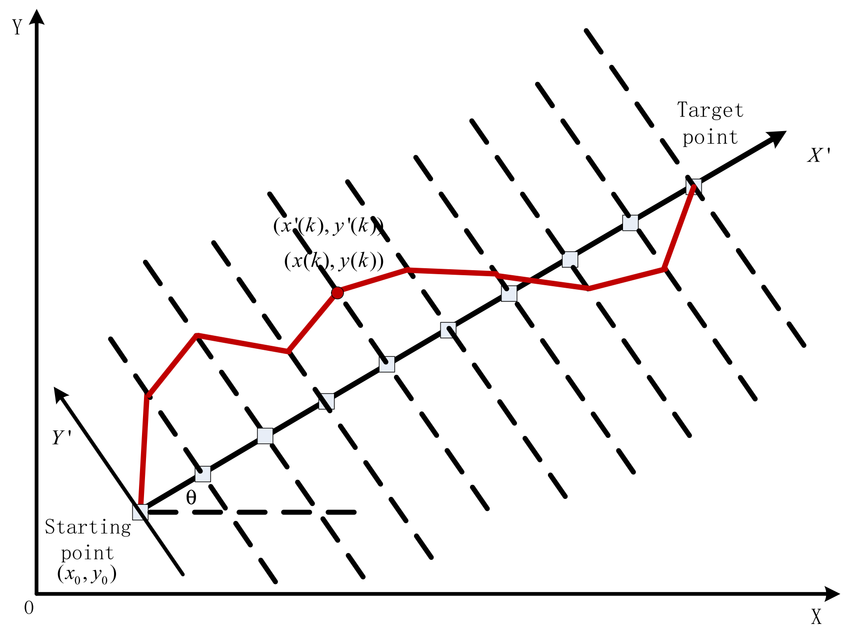

Fig. 1. Transformation of coordinates system.

In this paper, we propose a chaotic predator–prey biogeography-based optimization (CPPBBO) method, integrating the chaos theory and the concept of predator–prey into the classical BBO, to solve the UCAV path planning problem. First, we propose a new migration operation based on the chaos theory to generate new offspring to find the global optimal solution. Second, we include the concept of predator–prey in BBO algorithm in order to improve its capability of finding satisfactory solutions and increasing the diversity of the population. Simulation results and comparisons demonstrate the effectiveness of the proposed algorithm.

The rest of this paper is organized as follows. Section 2 introduces the threat resource and objective function in UCAV path planning. Section 3 describes the principle of basic BBO algorithm, while Section 4 specifies implementation procedure of our proposed chaotic predator–prey BBO algorithm. Finally, in Section 5, series of comparative experiments with basic BBO, CBBO and PPBBO are conducted. Our concluding remarks are contained in the final section.

2. Problem formulation

2.1. Path elements model in UCAV path planning

Modeling of the threat sources is the key task in UCAV optimal path planning. In an abstract term, path planning involves creating a plan to guide a point-like object from its initial position to a destination waypoint [1]. In our model, the path planning process is initialized by determining the start point as S and the target point as T, as is shown in Fig. 1. There are some threatening areas in the task region, such as missiles, radars, and artillery, which are all presented in the form of a circle, inside of which will be vulnerable to the threat with a certain probability proportional to the distance away from the threat center, while out of which will not be attacked. The flight task is to generate an optimal path between S and T considering all these threatening areas and the fuel cost.

Firstly, we connect point S and point T, then divide segment ST into $(D+1)$ equal portions. Draw a vertical line of ST at each segment point, this set of lines can be denoted as $L_1, L_2, \dots, L_k, \dots, L_D$. Take a discrete point at each line, engendering a discrete points collection $C = \{S, (x(1), y(1)), (x(2), y(2)), \dots, (x(k), y(k)), \dots, (x(D), y(D)), T\}$, and connect them in sequence to form a flight path. In this way, the path planning problem is turning into optimizing the coordinate series to achieve a superior fitness value of the objective function.

To accelerate the search speed of the algorithm, we can let line ST be the x axis and take the coordinate transformation on each discrete point $(x(k), y(k))$ according to formula (1), where θ is the angle that the original x axis counterclockwise rotate to parallel segment ST, while (x_s, y_s) represents the coordinates in the original coordinate system.

$$\begin{bmatrix} x'(k) \\ y'(k) \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x(k) - x_s \\ y(k) - y_s \end{bmatrix} \tag{1}$$

As shown in Fig. 1, x coordinate of each point can be obtained by a simple formula $x'(k) = \frac{|ST|}{D+1} \cdot k$, therefore the points collection C can be simplified to $C' = \{(0,0), (x'(1), y'(1)), (x'(2), y'(2)), \dots, (x'(k), y'(k)), \dots, (x'(D), y'(D)), (|ST|, 0)\}$, we also normalized $x'(k) = \frac{k}{D+1}$ when computing, and then multiplied it with $|ST|$ after optimization, which can greatly reduce the computational cost (see Fig. 2).

2.2. The performance evaluation function of route optimization

The performance indicators of the UCAV route mainly include the threat cost J_t and the fuel cost J_f, which can be evaluated as follow [11,2]:

$$J_t = \int_0^L w_t \, dl \tag{2}$$

$$J_f = \int_0^L w_f \, dl \tag{3}$$

where w_t and w_f are variables closely related with the current path and changing along with 'l', respectively presenting the threat cost and fuel cost of each line segment on the path, while L is the total length of the generated path.

In order to simplify the calculations, more efficient approximation to the exact solution is adopted. In this work, threat cost of each edge connecting two discrete points was calculated at five points along it, as is shown in Fig. 3.

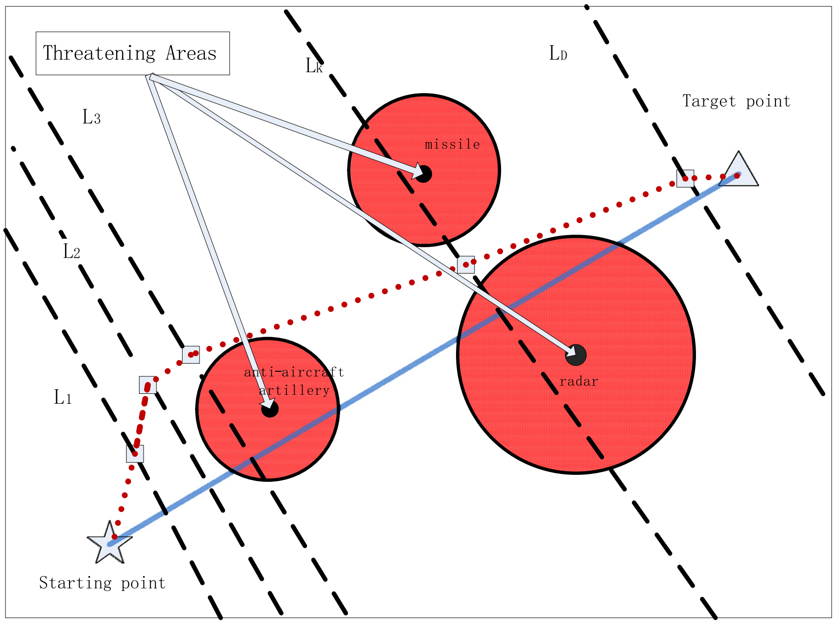

Fig. 2. Typical UCAV battle field model.

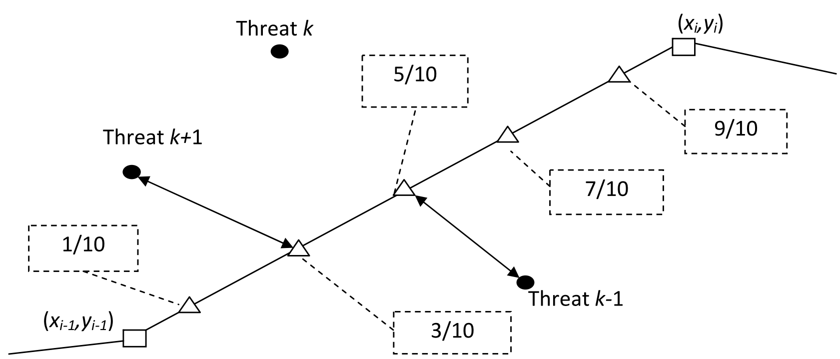

Fig. 3. Computation of threat cost.

If the ith edge is within the effect range, the threat cost associated with this threat is given by the expression [19].

$$w_{t,L_i} = \frac{L_i}{5} \cdot \sum_{k=1}^{N_t} t_k \cdot \left(\frac{1}{d_{0.1,i,k}^4} + \frac{1}{d_{0.3,i,k}^4} + \frac{1}{d_{0.5,i,k}^4} + \frac{1}{d_{0.7,i,k}^4} + \frac{1}{d_{0.9,i,k}^4} \right) \tag{4}$$

where N_t is the number of threatening areas, L_i is the ith sub-path length, $d_{0.1,i,k}$ is the distance from the 1/10 point on the ith edge to the kth threat, and t_k is the threat level of kth threat.

Assuming that the speed of UCAV is a constant, and then the fuel cost of the path J_f can be considered equal to L, the total length of path.

As J_t and J_f have different dimensions, both of them were normalized between zero and one before optimization. The total cost for traveling along the trajectory comes from a weighted sum of the threat and fuel costs, as is defined in Eq. (5),

$$J = kJ_t + (1-k)J_f \tag{5}$$

where k is a variable determining the relative emphasis of the various cost components with respect to the overall cost function, which gives the designer certain flexibility to dispose relations between the threat exposition degree and the fuel consumption. The value of k is normalized between zero to one (0.5 in our algorithm). When k is more approaching 0, a shorter path is needed to be planned, and less attention is paid to the collision avoidance. When k is more approaching 1 it requires avoiding the threat as far as possible on the cost of sacrificing the trajectory length. The optimized path is founded only when function J reaches its minimal value.

2.3. Restriction conditions

Considering UCAV flying under real air combat environment, the various restriction conditions of UCAV must be taken into account to obtain an appropriate flight path. The restrictions we mainly consider are as follow:

(1) The maximum yawing angle $\psi_{\max}$. Supposed that the coordinate of the UCAV's current location is (x_1, y_1), then the next position (x_2, y_2) must satisfy

$$\left| \arctan\left(\frac{y_2 - y_1}{x_2 - x_1} \right) \right| \leqslant \psi_{\max} \tag{6}$$

(2) The maximum length of flight path $L_{\max}$. As a UCAV must reach the destination before it run out of all fuel, the total length of the flight path L must satisfy

$$L \leqslant L_{\max} \tag{7}$$

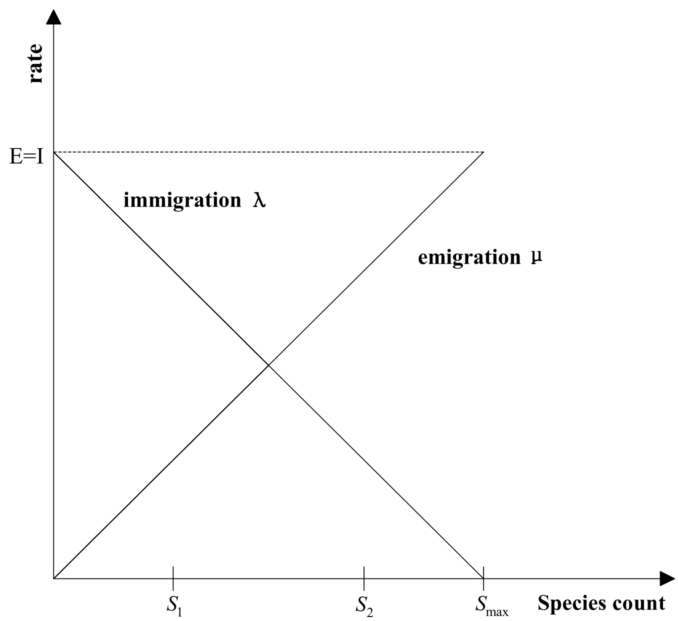

Fig. 4. Species model of a habitat.

So the path planning model of UCAV can be expressed as the minimization problem of Eq. (5) under the restriction conditions (6)–(7).

3. Principles of the basic BBO algorithm

BBO is a novel population-based optimization technique for solving global optimization problems. It is based on the concept of biogeography, which deals with the distribution of species over space and time. In the science of biogeography, a habitat is an ecological area that is inhabited by a particular plant or animal species and is geographically isolated from other habitats. Each habitat is considered as an individual and has its measure of goodness for living which is known as the habitat suitability index (HSI). Habitats that are well suited as residences for biological species are said to have a high HSI while habitats that are not good have a low HSI. The HSI of a habitat depends upon a number of features of the habitat, such as rainfall, diversity of vegetation, temperature, land area, etc. Each of these features that characterize habitability is known as suitability index variables (SIVs). The dynamics of the movement of the species among different habitats is mainly governed by two parameters called immigration rate (λ) and emigration rate (μ) and these two parameters depends upon the number of species in the habitats. From immigration graph of Fig. 4, the equation for immigration rate λ_k and emigration rate μ_k for k number of species can be written as

$$\lambda_k = I\left(1 - \frac{k}{n}\right) \tag{8}$$

$$\mu_k = E\left(\frac{k}{n}\right) \tag{9}$$

where E and I are the maximum emigration rate and maximum immigration rate, and $n = S_{\max}$ is the maximum number of species in a habitat.

We assume that each habitat has an identical species curve, with $E = I$ for normalization and simplicity, then combining (8) and (9) gives

$$\lambda_k + \mu_k = E \tag{10}$$

Mathematically, the concept of emigration and immigration can be represented by a probabilistic model. Now consider the probability P_s that the habitat contains exactly S species. P_s changes from time t to time $(t + \Delta t)$ as follows:

$$\begin{aligned} P_s(t + \Delta t) &= P_s(t)(1 - \lambda_s \Delta t - \mu_s \Delta t) \\ &\quad + P_{s-1}\lambda_{s-1}\Delta t + P_{s+1}\mu_{s+1}\Delta t \end{aligned} \tag{11}$$

where λ_s and μ_s are the immigration and emigration rates when there are S species in the habitat. This equation holds because in order to have S species at time $(t + \Delta t)$, one of the following conditions must hold:

(1) There were S species at time t, and no immigration or emigration occurred between t and $(t + \Delta t)$.
(2) There were $(S - 1)$ species at time t, and one specie immigrated.
(3) There were $(S + 1)$ species at time t, and one specie emigrated.

If time Δt is small enough so that the probability of more than one immigration or emigration can be ignored, then taking the limit of formula (11) as $\Delta t \to 0$ gives the following equation

$$\dot{P}_s = \begin{cases} -(\lambda_s + \mu_s)P_s + \mu_{s+1}P_{s+1}, & S = 0 \\ -(\lambda_s + \mu_s)P_s + \lambda_{s-1}P_{s-1} + \mu_{s+1}P_{s+1}, & \\ \quad 1 \leqslant S \leqslant S_{\max} - 1 & \\ -(\lambda_s + \mu_s)P_s + \lambda_{s-1}P_{s-1}, & S = S_{\max} \end{cases} \tag{12}$$

BBO concept is mainly based on the two mechanisms. These are migration and mutation.

3.1. Migration

In BBO algorithm, the candidate solutions for a problem are considered as habitats. Each solution is associated with a fitness value which is analogous to the HSI of a habitat. High-HSI habitat represents a good solution and low-HSI habitat represents an inferior solution. The emigration and immigration rates of each solution are used to probabilistically share features between habitats. (The shared features do not disappear from the high-HSI solutions.) Therefore, the migration operation, which is about emigration and immigration, are used to improve the solutions. For a D-dimensional optimization problem, a habitat is a $1 \times D$ array. The population consists of $NP = n$ parameters vector $X_i, i = 1, 2, \ldots, n$. One option for implementing the migration operator is shown in Fig. 5.

A procedure of elitism is introduced to preserve some of the best habitats in the previous generation. These habitats are kept from the migration operation by setting their immigration rate λ to zero. This procedure, known as elitism operation, helps us to prevent best solutions from being corrupted due to the process of immigration.

for $i=1$ to NP
  Select X_i with probability $\propto \lambda_i$
  If X_i is selected **then**
    for $j=1$ to NP **do**
      Select X_j with probability $\propto \mu_j$
      if X_j is selected **then**
        Randomly select a SIV from X_j
        Replace a random SIV in X_i with selected SIV of X_j
      end if
    end for
  end if
end for

Fig. 5. Algorithm for migration process of the BBO.

```
for i = 1 to NP
  for j = 1 to D do
    Use λ_i and μ_i to compute the probability P_i using (10)
    Select a SIV H_i(j) with probability based on probability m_s
      if H_i(j) is selected then
        Replace H_i(j) with a randomly generated SIV
      end if
  end for
end for
```

Fig. 6. Algorithm for mutation process of the BBO.

3.2. Mutation

It is well known that due to some natural calamities or other events, HIS of a natural habitat might change suddenly from its equilibrium value. In BBO, this event is represented by the mutation of SIV. The species count probabilities are used to determine mutation rates. The probabilities of each species count are determined by the differential equation in (12). Mutation rate of each set of solution can be calculated in terms of species count probability using the following equation:

$$m(s) = m_{\max}\left(\frac{1 - P_s}{P_{\max}}\right) \tag{13}$$

where $m(s)$ is the mutation rate for habitat is contains S species, $m_{\max}$ is the maximum mutation rate, $P_{\max}$ is the maximum probability. The algorithm for mutation process of BBO is shown in Fig. 6.

This mutation scheme makes both high and low HIS solutions likely to mutate, which gives them a chance for improving. The procedure of mutation tends to increase diversity among the population.

4. Description of the proposed method

4.1. Chaos theory

Chaos theory is epitomized by the so-called "butterfly effect" detailed by Lorenz [13]. Chaos is a general nonlinear but deterministic phenomenon in nature [10]. In this paper, a chaos variable is introduced as a disturbance through the traditional method in order to optimize the search process.

The method of chaos optimization uses the logistic model:

$$x_{n+1} = f(x_n) = \mu x_n(1 - x_n) \tag{14}$$

where $x_n \in [0, 1]$.

μ is called logistic parameter. When μ equals to 4, the iterations produce values for a pseudo-random distribution. A very small difference in the initial value of x would give rise to large difference in its long-time behavior, which is the basic characteristic of chaos. We use this feature to avoid local convergence. Its randomness ensures the capability conducting a large-scale search and help to overcome the limitation of local best solutions. Therefore, after each generation of mutation, we can conduct the chaotic search in the neighborhood of the current optimal parameters by listing a certain number of new generated parameters through chaotic process. In this way, we make use of the ergodicity and irregularity of the chaotic variable to help the algorithm to jump out of the local optimum as well as finding the optimal parameters. The experimental results in Section 5 show the efficiency of our proposed method.

4.2. Predator–prey concept

Predatory behavior is one of the most common phenomena in nature, and many optimization algorithms are inspired by the predator–prey strategy from ecology [6,4]. In nature, predators hunt prey to guarantee their own survival, while the preys need to be able to run away from predators. On the other hand, predators help to control the prey population while creating pressure in the prey population. In this model, an individual in predator population or prey population represents a solution, each prey in the population can expand or get killed by predators based on its fitness value, and a predator always tries to kill preys with least fitness in its neighborhood, which represents removing bad solutions in the population. In this paper, the concept of predator–prey is used to increase the diversity of the population, the predators are modeled based on the worst solutions as Eq. (15) demonstrates:

$$P_{predator} = P_{worst} + \rho(1 - t/t_{\max}) \tag{15}$$

where $P_{predator}$ is the predator (a possible solution), P_{worst} is the worst solution in the population, t is the current iteration, while $t_{\max}$ is the maximum number of iterations and ρ is the hunting rate. To model the interactions between predator and prey, Eq. (16) is also used and this provides the solutions to maintain a distance from the predator

$$\begin{cases} P_{k+1} = P_k + \rho e^{-|d|}, & d > 0 \\ P_{k+1} = P_k - \rho e^{-|d|}, & d < 0 \end{cases} \tag{16}$$

where d is the distance between the solution and the predator, and k is the current iteration.

4.3. Proposed chaotic predator–prey BBO (CPPBBO)

Due to the flexibility, versatility and robustness in solving optimization problems, BBO algorithm has already aroused intense interest. However, there still exist some flaws on this algorithm, such as the large number of iterations to reach the global optimal solution and the tendency to converge to local best solutions. In order to overcome these flaws of the classical BBO algorithm, CPPBBO, which integrates BBO with chaotic variable and the concept of predator–prey, was proposed in our work. After the mutation of each generation, conduct the chaotic search and the predator–prey behavior in order to choose better solutions into next generation. In this way, our proposed algorithm takes the advantage of the characteristics of the chaotic variable and the predator–prey concept to make the individuals of sub generations distributed ergodically in the defined space and thus to avoid from the premature of the individuals, as well as to increase the speed of finding the optimal solution.

The implementation procedure of our proposed BBO approach to UCAV path planning can be described as follows:

Step 1. According to the environmental modeling in Section 2, initialize the detailed information about the path planning task, as well as the threatened information including the coordinates of threat centers, threat radii and threat levels. In order to simplify the calculation, conduct the coordinate transformation on discrete points related with the task according to formula (1).

Step 2. Initialize the BBO parameters, such as the maximum species count $S_{\max}$ and the maximum migration rates E and I (see Fig. 4), the maximum mutation rate $m_{\max}$, and an elitism parameter Keep. Initialize random population of D-dimensional parameters $C' = \{y'(1), y'(2), \ldots, y'(k), \ldots, y'(D)\}$ within the bound of

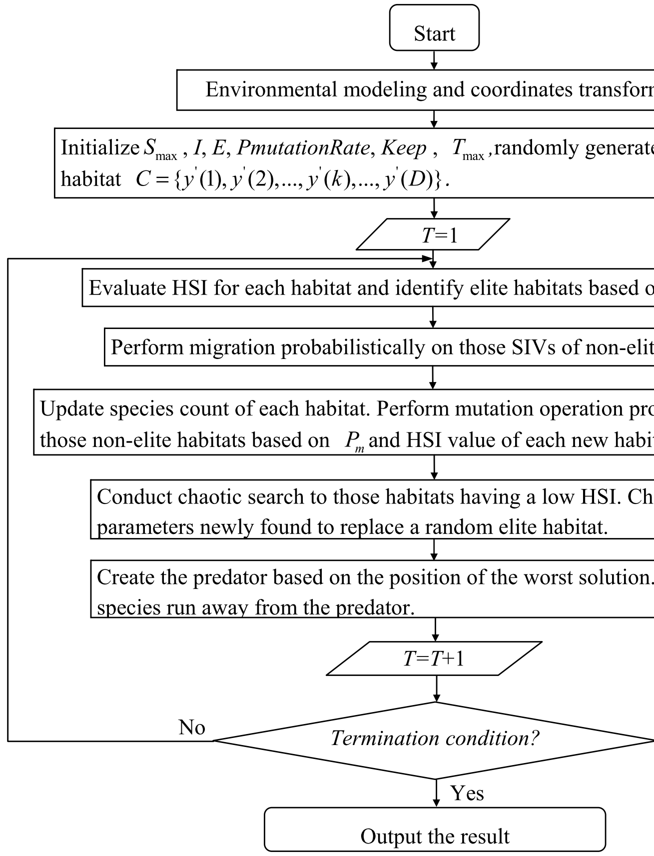

Fig. 7. Procedure of our proposed method.

the battlefield, which represent the y coordinates of each discrete point as we discussed in Section 2, while the corresponding x coordinates could be easily obtained by the formula $x'(k) = \frac{|ST|}{D+1} \cdot k$. Each group of parameters can engender a path that leading the UAV from the starting point S to the target point T, and the goal is to find the optimal combination of parameters that can provide relatively satisfactory performance. Initialize a set of habitats, each habitat corresponding to a potential solution to the given problem. Initialize the starting iteration $T = 1$.

Step 3. According to the parameters of the generated habitats, calculate the cost of each path formed by relative parameters based on formulas (2), (3), (4), (5). The smaller the cost value is, the better performance the path maintains. Based on the values of HSI, elite habitats are identified.

Step 4. Use immigration and emigration to modify each non-elite habitat probabilistically as described in Section 3.1 and recomputed HSI of each modified habitat. Feasibility of a solution is verified when each SIV satisfies equality and inequality constraints of generator as mentioned in the specific problem.

Step 5. Update the species count probability of each habitat. Then, perform mutation operation on the habitats as discussed in Section 3.2 and recomputed each HSI value of new habitats.

Step 6. Conduct the chaotic search to those habitats having a low HSI based on formula (14) after transforming the parameters ranges into $(0, 1)$. Among the engendered series of solutions, select the best one and use it to replace a random elite habitat.

Step 7. Model the predators based on the worst solution as formula (15) demonstrates. Then, use formula (16) to provide the other solutions to maintain a distance from the predator.

Step 8. If 1) the path is feasible (all the constraints fulfilled), safe (out of all the threat areas), and short enough, or (2) T_{max} is reached, output the optimal parameters and optimal cost value. Otherwise, go to Step 3.

The detailed procedure can also be shown with Fig. 7.

5. Comparative experimental results

In order to evaluate the performance of the proposed CPPBBO algorithm in this work, series of experiments are conducted in a Matlab2012a programming environment on a PC with 1.2 GHz CPU. Coordinates of the starting point are set as $(10, 15)$, and the target point as $(80, 75)$. The UCAV fly from starting point to target point. In the flight course, there exist 8 circular threat areas. Center of the threat areas and corresponding threat radii and threat levels are shown in Table 1.

Table 1
Information of threatening areas.

Threat center	(59, 52)	(55, 80)	(27, 58)	(24, 33)	(12, 48)	(70, 65)	(70, 34)	(30, 70)
Threat radius	10	9	9	9	12	7	12	10
Threat level	9	7	3	12	1	5	13	2

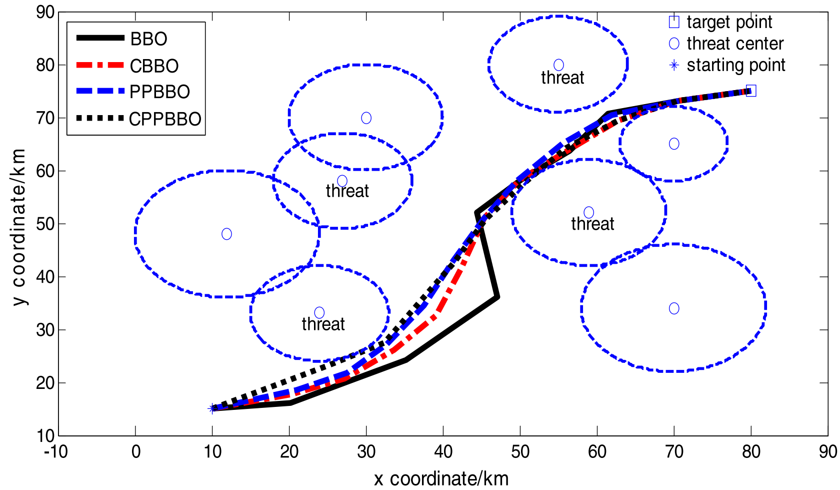

Fig. 8. Comparative path planning results, $D = 10$.

The initial parameters of both classical BBO algorithm and CPPBBO were adjusted as:

- Population size: $S_{max} = 60$;
- Maximum possible immigration rate: $I = 1$;
- Maximum possible emigration rate: $E = 1$;
- Maximum mutation rate: PmutationRate = 0.03;
- Elitism parameter: Keep = 5;
- Cycle counter: $T_{max} = 100$.

Respectively assume D as 10 and 15 to carry our experiments, the results of which are shown in Figs. 9, 10, 11. In our work, in order to show the improvement of exploration ability with the chaos theory and the predator–prey concept, we also described the performances of Chaotic BBO (CBBO) and Predator–Prey BBO (PPBBO).

When $D = 10$, the experimental results of classical BBO and our proposed BBO algorithm have differences due to the calculating complexity as shown in Fig. 8 and Fig. 9. When the value of D is increased to 15, we can see the superiority of our proposed method over the classical BBO algorithm more clearly in the comparative experimental results shown in Fig. 10 and Fig. 11.

For comparison, we also used A* algorithm for this path planning problem. The optimization result by A* algorithm is as in Fig. 12.

From Fig. 12, path planning result optimized by A* algorithm may not satisfy all the constraints, while this is not a problem for our proposed method as BBO algorithm can easily handle constraints.

To further prove the performance of our proposed method against classical BBO algorithm, we run the program for 100 times to obtain the mean minimum and the best minimum of our generated best path of each algorithm when $D = 10$. Results comparison between four algorithms is shown in Table 2.

CPPBBO performs the best in terms of both average performance and the best performance, apparently showing that our improved method can find the feasible and optimal path for UCAV more stable than classical BBO algorithm, and can effectively solve the path planning problem of UCAV in complex combat field environment.

From evolution curves of four algorithms, we see CBBO sometimes converged faster than CPPBBO, which actually shows the effect of predator–prey concept because it aims to increase the diversity of the population and prevent the population getting into local minimum solution, thus it reduce the convergence speed in some degree, however, the final result of CPPBBO is better than

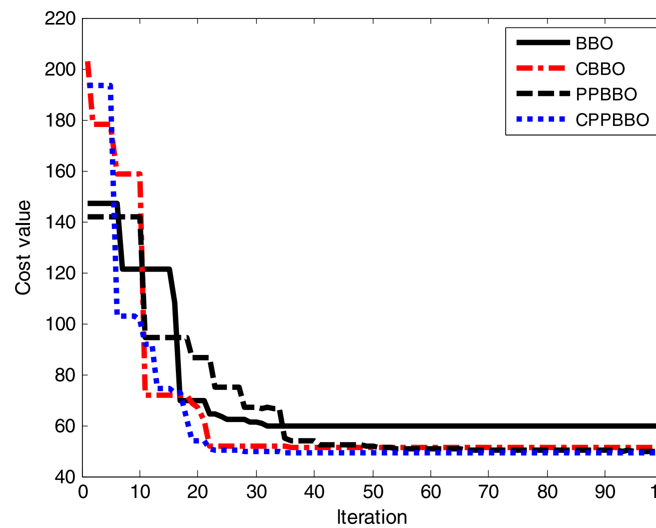

Fig. 9. Evolution curves of four algorithms, $D = 10$.

CBBO as predator–prey concept really help the optimization process, especially in the last phase.

We have also made some analysis about parameter of the proposed algorithm:

(1) Population size S_{max}: S_{max} is an important parameter in BBO algorithm, increase of S_{max} would enlarge the search space and result in a diversity of solutions. However, a too large S_{max} ($S_{max} = 200$) may make the population converge slowly and require a large amount of computational effort. On the other hand, a too small S_{max} ($S_{max} = 15$) would lead to local best solution. After a number of simulations under different conditions ($S_{max} = 15, 60, 100, 200$), we finally decide $S_{max} = 60$ as it is suitable for the path planning problem in this paper.

(2) Max possible im/emmigration rate: We assume that each habitat has an identical species curve, with $E = I$ for normalization and simplicity, which means the immigration rate and emigration rate of habitats ranges from 0 to 1.

(3) Maximum mutation rate m_{max}: Mutation operation tends to increase diversity among the population, it makes low HIS solutions likely to mutate, which gives them a chance of improving. However, a too high mutation rate may cause an excess of search and is not beneficial to the improvement of solutions. After numbers of simulations, we chose m_{max} to be 0.03, which is suitable for this problem.

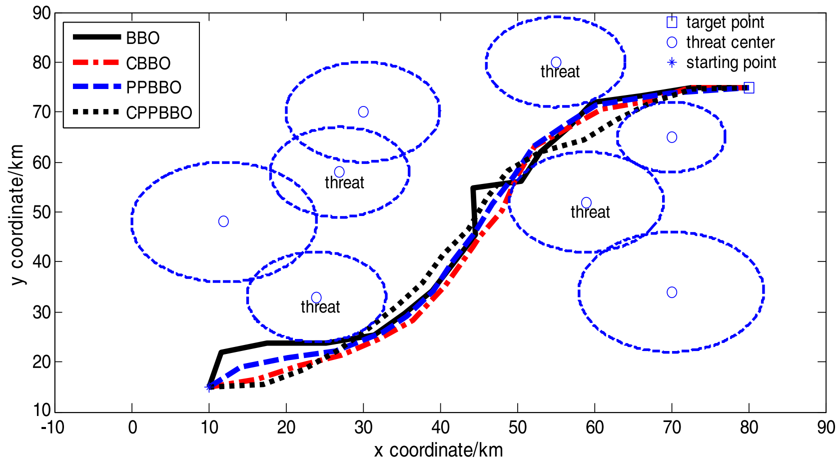

Fig. 10. Comparative path planning results, $D = 15$.

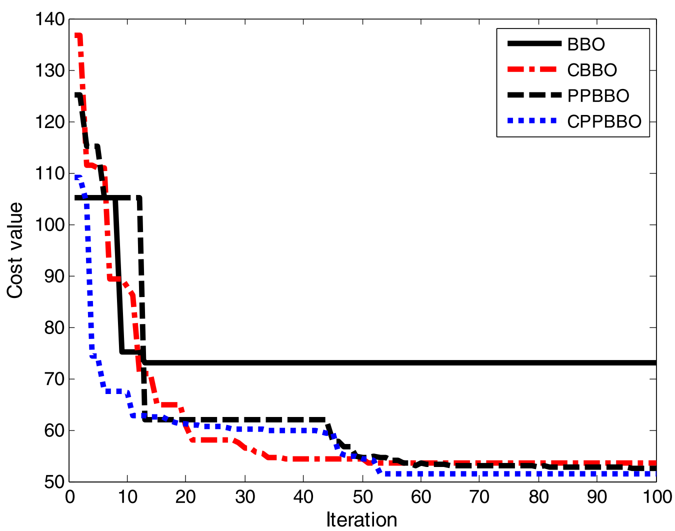

Fig. 11. Evolution curves of four algorithms, $D = 15$.

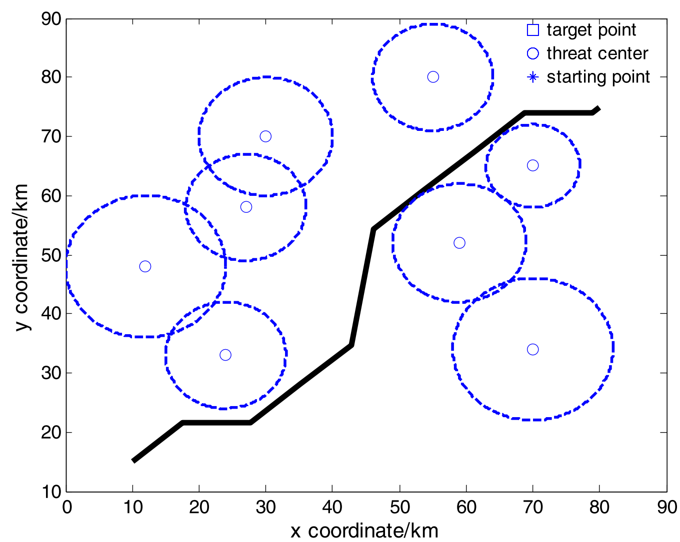

Fig. 12. Path planning result by A* algorithm.

From the above experimental results, we can clearly see that using classical BBO algorithm could possibly lead to a path that does not satisfy the requirements, especially when the optimized dimension increases. Therefore, we make use of the ergodicity of chaotic variable or the concept of predator–prey to help the classical BBO algorithm to jump out of the local best and obtain a favorable path.

6. Conclusions

In this paper, a novel CPPBBO approach for solving the UCAV path planning problem in complex combat field environment was proposed. The generated path can ensure the maximum safety with the minimum fuel cost of UCAV. Series of comparative simulation results were given to show that our proposed CPPBBO algorithm is more efficient than basic BBO, CBBO and PPBBO in solving the UCAV path planning problems.

Our future work will focus on the exact application of our proposed CPPBBO method in UCAV path re-planning and formation, which are two challenging issues for UCAV.

Acknowledgements

This work was partially supported by Natural Science Foundation of China (NSFC) under grants #61333004, #61273054, and #60975072, National Key Basic Research Program of China (973 Project) under grants #2014CB046401 and #2013CB035503, Top-Notch Young Talents Program of China, National Magnetic Confinement Fusion Research Program of China under grant #2012GB102006, and Aeronautical Foundation of China under grant #20135851042.

Table 2
Results comparison between four algorithms.

	BBO	CBBO	PPBBO	CPPBBO
Mean minimum	66.4467	54.9258	56.4431	52.8848
Best minimum	53.3449	51.5333	49.8546	49.7885

References

[1] M.A. Al-Jarrah, M.M. Hasan, HILS setup of dynamic flight path planning in 3D environment with flexible mission planning using ground station, Journal of the Franklin Institute 348 (1) (2011) 45–65.
[2] E. Alison, R. Arthur, Path-planning with avoidance using nonlinear branch-and-bound optimization, in: AIAA Guidance, Navigation and Control Conference and Exhibit, 20–23 August 2007, Hilton Head, South Carolina, in: AIAA, 2007, 6793.
[3] E. Besada-Portas, L. de la Torre, J.M. de la Cruz, B. de Andrés-Toro, Evolutionary trajectory planner for multiple UAVs in realistic scenarios, IEEE Transaction on Robotics 26 (4) (2010) 619–634.
[4] M.A. Costa e Silva, L.S. Coelho, L. Lebensztajn, Multiobjective biogeography-based optimization based on predator–prey approach, IEEE Transactions on Magnetics 48 (2) (2012) 951–954.

[5] H.B. Duan, S.Q. Liu, J. Wu, Novel intelligent water drops optimization approach to single UCAV smooth trajectory planning, Aerospace Science and Technology 13 (8) (2009) 442–449.
[6] H.B. Duan, Y.X. Yu, Z.Y. Zhao, Parameters identification of UCAV flight control system based on predator–prey particle swarm optimization, Science China Information Sciences 56 (1) (2013) 012202(12).
[7] J.L. Foo, J. Knutzon, V. Kalivarapu, J. Oliver, E. Winer, Path planning of unmanned aerial vehicles using B-splines and particle swarm optimization, Journal of Aerospace Computing, Information, and Communication 6 (2008) 271–290.
[8] Y.G. Fu, M.Y. Ding, C.P. Zhou, Phase angle-encoded and quantum-behaved particle swarm optimization applied to three-dimensional route planning for UAV, IEEE Transaction on Systems, Man, and Cybernetics-Part A: Systems and Humans 42 (2) (2012) 511–526.
[9] W. Gong, Z.H. Cai, C.X. Ling, H. Li, A real-coded biogeography-based optimization with mutation, Applied Mathematics and Computation 216 (9) (2010) 2749–2758.
[10] H.M. Jiang, C.K. Kwong, Z.Q. Chen, Y.C. Ysim, Chaos particle swarm optimization and T–S fuzzy modeling approaches to constrained predictive control, Expert System with Applications 39 (1) (2012) 194–201.
[11] J. Karimi, S.H. Pourakdoust, Optimal maneuver-based motion planning over terrain and threats using a dynamic hybrid PSO algorithm, Aerospace Science and Technology 26 (1) (2013) 60–71.
[12] X.T. Li, M.H. Yin, Multi-operator based biogeography based optimization with mutation for global numerical optimization, Computers & Mathematics with Applications 64 (9) (2012) 2833–2844.
[13] E.N. Lorenz, Deterministic nonperiodic flow, Journal of the Atmospheric Sciences 20 (2) (1963) 130–141.
[14] Y.V. Pehlivanoglu, A new vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV, Aerospace Science and Technology 16 (1) (2012) 47–55.
[15] S.H.A. Rahmati, M. Zandieh, A new biogeography-based optimization (BBO) algorithm for the flexible job shop scheduling problem, The International Journal of Advanced Manufacturing Technology 58 (9–12) (2012) 1115–1129.
[16] D. Simon, Biogeography-based optimization, IEEE Transactions on Evolutionary Computation 12 (6) (2008) 702–713.
[17] D. Simon, M. Ergezer, D. Dawei, R. Rarick, Markov models for biogeography-based optimization, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 41 (1) (2011) 299–306.
[18] L. Wang, Y. Xu, An effective hybrid biogeography-based optimization algorithm for parameter estimation of chaotic systems, Expert System with Applications 38 (12) (2011) 15103–15109.
[19] C.F. Xu, H.B. Duan, F. Liu, Chaotic artificial bee colony approach to uninhabited combat air vehicle (UCAV) path, Aerospace Science and Technology 14 (8) (2010) 535–541.