Multi-target Collaborative Combat Decision-Making by Improved Particle Swarm Optimizer

Ding Yongfei[1], *Yang Liuqing*[2*], *Hou Jianyong*[3], *Jin Guting*[3], *Zhen Ziyang*[2]

1. Science and Technology on Avionics Integration Laboratory, Shanghai 200233, P. R. China;
2. College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, P. R. China;
3. China National Aeronautical Radio Electronics Research Institute, Shanghai 200233, P. R. China

(Received 25 October 2017; revised 8 January 2018; accepted 18 January 2018)

Abstract: A decision-making problem of missile-target assignment with a novel particle swarm optimization algorithm is proposed when it comes to a multiple target collaborative combat situation. The threat function is established to describe air combat situation. Optimization function is used to find an optimal missile-target assignment. An improved particle swarm optimization algorithm is utilized to figure out the optimization function with less parameters, which is based on the adaptive random learning approach. According to the coordinated attack tactics, there are some adjustments to the assignment. Simulation example results show that it is an effective algorithm to handle with the decision-making problem of the missile-target assignment (MTA) in air combat.

Key words: collaborative combat; multi-target decision-making; improved particle swarm optimization (IPSO)

CLC number: V249 **Document code**: A **Article ID**: 1005-1120(2018)01-0181-07

0 Introduction

Modern fighters have the ability to attack multiple targets and carry long range air-to-air missiles. Beyond visual range (BVR) air combat has been the mainstream with the development of modern fighters, where fighters are required to exchange information and attack multiple targets cooperatively[1-2]. To complete cooperative multiple target attack (CMTA), decision-making (DM) is necessary for fighters to allot targets and missiles according to the shared information[3-4]. Thus, the missile-target assignment (MTA) problem is the main part of DM when it comes to CMTA.

There are many algorithms applied to DM problem in CMTA, such as particle swarm optimization (PSO), genetic algorithm (GA) and ant colony optimization (ACO)[5-7]. A heuristic algorithm is introduced to adaptive genetic algorithm in Ref. [8] and improves local search capability. Adaptive pseudo-parallel genetic algorithm is also considered to deal with air combat DM problem beyond visual range[9]. However, GA is not a real-time algorithm and may not work sometimes. Some intelligent algorithms are also used to solve DM problems[10-12]. In Ref. [13], fuzzy neural network is applied to assign missiles according to the threat of enemy fighters and the bomb load of our fighters. However, it is hard to obtain practical and complex air situation data for neural network training. Considering the uncertain information in the MTA problem, grey system theory is introduced in DM problem[14].

In this paper, an improved particle swarm optimizer (IPSO) is deduced to handle with the DM problem for CMTA in the air combat. The IPSO algorithm has stronger global searching capability by designing a new velocity learning strategy.

1 DM Problem in CMTA

1.1 Air combat situation

Air combat decision-making is based on the

* Corresponding author, E-mail address: ylqnuaa@163.com.

How to cite this article: Ding Yongfei, Yang Liuqing, Hou Jianyong, et al. Multi-target collaborative combat decision-making by improved particle swarm optimizer[J]. Trans. Nanjing Univ. Aero. Astro., 2018,35(1):181-187.
http://dx.doi.org/10.16356/j.1005-1120.2018.01.181

air combat situation. To establish the model of air combat situation, it is assumed that there are M our fighters which are marked in blue and N enemy fighters which are marked in red. Denote our fighter set $B=\{B_i, i=1,2,\cdots,M\}$ and enemy fighter set $R=\{R_j, j=1,2,3,\cdots,N\}$. In an air combat, the situation between our fighters and enemy fighters can be illustrated in Fig. 1, where LOS is the line of sight and D_{ij} the distance between B_i and R_j. x_{Bi} and V_{Bi} are the position and velocity of B_i, respectively. ε_{ij} is the bore of sight (BOS) angle of R_j to B_i. x_{Rj} and V_{Rj} are the position and velocity of R_j, respectively. ε_{ji} is the BOS angle of B_i to R_j.

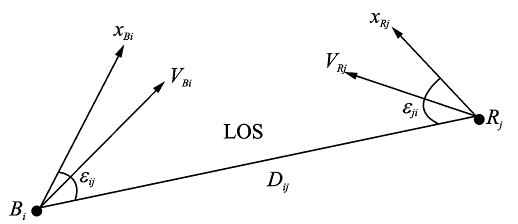

Fig. 1 The situation between B_i and R_j

Distance, BOS angle and velocity are taken into consideration as threat factors when constructing the threat function[15]. The threat function is described as a composite of all its threat factors, namely

$$th_{ij}=\omega_1 th_{ij}^{D_{ij}} th_{ij}^{\varepsilon_{ij}}+\omega_2 th_{ij}^{V_{Bi}} \tag{1}$$

where $th_{ij}^{D_{ij}}$ is the distance threat factor, $th_{ij}^{\varepsilon_{ij}}$ the BOS angle threat factor, $th_{ij}^{V_{Bi}}$ the velocity threat factor, and ω_1, ω_2 are non-negative weight coefficients and satisfy

$$\omega_1+\omega_2=1 \tag{2}$$

Moreover, the value range of all the threat factor functions is $[0,1]$. Thus, there is $th_{ij}\in[0,1]$.

The distance threat factor can be constructed as

$$th_{ij}^{D_{ij}}=\begin{cases}1 & D_{ij}\leqslant R_{aB}\\ 1-\dfrac{D_{ij}-R_{aB}}{T_{rB}-R_{aB}} & R_{aB}<D_{ij}<T_{rB}\\ 0 & D_{ij}>T_{rB}\end{cases} \tag{3}$$

where R_{aB} is the maximum effective striking distance of missiles carried by our fighters and T_{rB} the maximum radar tracking distance of our fighters. In other words, the distance threat function is proportional to $\dfrac{R_{aB}}{T_{rB}}$ and inversely proportional to $\dfrac{D_{ij}}{T_{rB}}$. However, if our fighter cannot track the enemy fighter, the distance threat is 0 and if enemy fighter is within the striking range, the distance threat is 1.

The BOS angle threat factor can be constructed as

$$th_{ij}^{\varepsilon_{ij}}=\mathrm{e}^{-\lambda_1(\pi\varepsilon_{ij}/180)^{\lambda_2}} \tag{4}$$

where λ_1, λ_2 are the positive constants. Better attack angle results in better attack effect.

The velocity threat function can be constructed as

$$th_{ij}^{V_{Bi}}=\begin{cases}1 & V_{R_j}<0.5V_{B_i}\\ 1.5-\dfrac{V_{R_j}}{V_{B_i}} & 0.5V_{B_i}\leqslant V_{R_j}\leqslant 1.4V_{B_i}\\ 0.1 & V_{R_j}>1.4V_{B_i}\end{cases} \tag{5}$$

1.2 MTA model

Multi-fighter cooperative attack problem is aimed at optimizing target assignment for missiles carried by our fighters. According to the threat function known, multiple target assignment develops a proposal where there are more attack success and less fighter casualties.

Assume that our fighter B_i carries L_i missiles to attack enemy fighter targets. Thus, there are $Z=\sum_{i=1}^{M}L_i$ missiles our fighter carried. The missile number Z satisfies

$$N\leqslant Z\leqslant 2N \tag{6}$$

Denote our Z missiles set $G=\{G_r, r=1,2,\cdots,Z\}$. The rth missile in set B corresponds to the hth missile of the B_ith blue fighter. The B_ith blue fighter carries L_i missiles. The rth missile can be defined as

$$r=\sum_{f=1}^{i-1}L_f+hh=1,2,\cdots,L_i;i=1,2,\cdots,M \tag{7}$$

Missiles in set B consist of Boolean function, X_{rj} states whether the rth missile attacks the jth enemy fighter.

$$X_{rj}=\begin{cases}1 & \text{the } r\text{th missile attacks } R_j\\ 0 & \text{the } r\text{th missile doesn't attack } R_j\end{cases} \tag{8}$$

th_{rj} is the threat value of the missile r carried by our fighter B_i to the enemy fighter R_j. Thus, the survival probability of R_j after attacked is $1-th_{rj}$. In other words, the expected remaining threat of R_j to our fighter B_i is $th_{ji} \cdot \prod_{r=1}^{Z}(1-th_{rj})X_{rj}$.

What we want is to find an optimal missile assignment proposal π to make remaining threat function minimum. The optimization function is described as

$$E(\pi)=\arg\min_{\pi\in\Omega}\sum_{j=1}^{N}\sum_{i=1}^{M}\left\{th_{ji}\cdot\left[\prod_{r=1}^{Z}(1-th_{rj})X_{rj}\right]\right\} \tag{9}$$

1.3 Analysis on coordinated attack tactics

When our fighters attack enemy fighter targets, assignment rules need to be determined for our fighters. The assignment rules work so that our fighters get more benefit when attacking.

It is supposed that each missile of our fighters can attack only one enemy fighter target. One enemy fighter is attacked by two missiles at most. It is essential to declare constraints on X_{rj}

$$\begin{aligned}&\sum_{j=1}^{n}X_{rj}=1 \quad r=1,2,\cdots,Z\\&\sum_{r=1}^{Z}X_{rj}\leqslant 2 \quad j=1,2,\cdots,N\end{aligned} \tag{10}$$

According to priority attack principle, optimization function is shown as

$$\mathrm{ASM}(i,j)=th_{ij}\times\sum_{i=1}^{M}th_{ji} \tag{11}$$

where $\sum_{i=1}^{M}th_{ji}$ is the sum of threat of R_j to B_i and th_{ij} is the threat of B_i to R_j. It also can be seen as the assigned value of B_i to R_j. The larger the assigned value is, the more chance R_j would have to be attacked by B_i. If $\sum_{i=1}^{M}th_{ji}$ is of large value, it is essential to assign two missiles to attack the target. The larger assigned value ASM(i,j) means better attack effect.

Denote missile pair in the situation where two different missiles Z_r and Z_l attack the same target $R_i(r,l,j)$. The assigned value difference of (r,l,j) is defined as

$$\mathrm{DIF}(r,l,j)=|\mathrm{ASM}(r,j)-\mathrm{ASM}(l,j)| \tag{12}$$

There is optimal attack effect when one of the assigned value is much larger than the other.

Then, the MTA problem is to find a solution π to minimize the equation above and accord with coordinated attack tactics.

2 Improved Particle Swarm Optimization

In the PSO algorithm, each particle is treated as a potential solution in D-dimensional space. The position of the ith particle is represented by a D-dimensional vector $\boldsymbol{X}_i=(x_{i1},x_{i2},\cdots,x_{iD})$, and the velocity of the ith particle can also be represented by a D-dimensional vector $\boldsymbol{V}_i=(v_{i1},v_{i2},\cdots,v_{iD})$.

In the PSO algorithm, the updating formulae of the velocity and the position of each particle are given by

$$\begin{aligned}v_{id}^{k+1}&=v_{id}^{k}+c_1\cdot\mathrm{rand}_1^k\cdot(p_{id}^k-x_{id}^k)+c_2\cdot\mathrm{rand}_2^k\cdot(p_{gd}^k-x_{id}^k)\\x_{id}^{k+1}&=x_{id}^k+v_{id}^{k+1}\end{aligned} \tag{13}$$

where k is a pseudo-time increment and represents iterations; $P_i=(p_{i1},p_{i2},\cdots,p_{iD})$ is the local optimal position of the ith particle; $P_g=(p_{g1},p_{g2},\cdots,p_{gD})$ represents the global optimal position in the swarm, here g is the index of the best particle among all the particles in the population; c_1 and c_2 are called the cognitive and the social coefficients, respectively; rand_1 and rand_2 are two random numbers in range $[0,1]$.

Based on the PSO algorithm above, an improved PSO (IPSO) is presented, in which a new learning strategy is introduced in the particle velocity update equation, described as

$$v_{jd}^{k+1}=\chi\left[\left(1-\frac{j}{n}\right)\cdot\mathrm{rand}_1^k\cdot v_{id}^k+\left(1+\frac{j}{n}\right)\cdot\mathrm{rand}_2^k\cdot(p_{bd}^k-x_{jd}^k)\right] \tag{14}$$

where rand_1 and rand_2 are the random numbers in range $[0,1]$. χ is the constriction coefficient; $P_b=[p_{b1},\cdots,p_{bD}]$ the particle position with better performance which is selected randomly; j the arrangement number according to the performance, here the smaller j corresponds to the better performance of the jth particle; n the whole number of the particles in the population.

The constriction coefficient χ is introduced to ensure the PSO algorithm to converge. $\left(1-\frac{j}{n}\right)$ dynamically adjusts the particle velocity for focusing the PSO into a local search. $\left(1+\frac{j}{n}\right)$ makes the particle position tend to a particle position with better performance.

IPSO algorithm with fewer parameters not only keeps the diversity of the velocities but also does not alleviate the certainty of directing to the destination. The particles with better performance will increase their inertia movements, which expands the searching space and improves the searching speed. The particles with worse performance will increase their learning steps, which reduces the differences among the population and improves the whole performance of the population.

Thus, the IPSO algorithm flow can be described in Fig. 2.

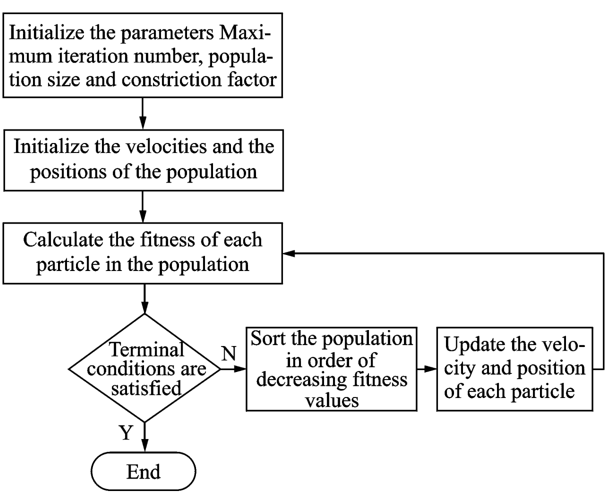

Fig. 2 IPSO algorithm flow

3 Realization of IPSO for Multi-target Collaborative Combat Decision-Making

Every possible optimal solution is seen as a particle in PSO. The adaptive value of particle needs to be calculated in every position. It is reasonable for the adaptive value to be defined as objective optimization function to get the updating velocity and direction for every particle. Based on the MTA model established above, a set of missile-target assignment is dealt with as partial swarm after updating. m optimal MTA proposals correspond to m particles in the particle swarm. Every particle is in the searching space of Z dimension. The position vector of the kth particle in the current iteration is defined as

$$\boldsymbol{\pi}_k=(c_{k1}\quad c_{k2}\quad \cdots \quad c_{kr}\quad \cdots \quad c_{kZ}) \tag{15}$$

where $k=1,2,\cdots,m$, Z the sum of missiles, and c_{kr} the position of the kth particle in the rth dimension. c_{kr} belongs to $[1\quad N\_\text{red}]$ and N_red is the sum of enemy fighter target.

The velocity of the kth particle is given by

$$\boldsymbol{V}_k=(v_{k1}\quad v_{k2}\quad \cdots \quad v_{kr}\quad \cdots \quad v_{kZ}) \tag{16}$$

where v_{kr} satisfies $v_{kr}\in[-1+N\quad N-1]$.

If the kth particle has the best fitness in the current iteration, it is defined as the local optimal solution and noted as

$$\boldsymbol{P}_k=(p_{k1}\quad p_{k2}\quad \cdots \quad p_{kr}\quad \cdots \quad p_{kZ}) \tag{17}$$

If all of the particle have the best fitness in the current iteration, it is defined as the global optimal solution and noted as

$$\boldsymbol{P}_g=(p_{g1}\quad p_{g2}\quad \cdots \quad p_{gr}\quad \cdots \quad p_{gZ}) \tag{18}$$

The updating formulae of the velocity and the position of each particle based on IPSO are given by

$$v_{kr}(t+1)=\chi\left[\left(1-\frac{j}{n}\right)\cdot \text{rand1}_k\cdot v_{kr}(t)+\left(1+\frac{j}{n}\right)\cdot \text{rand2}_k\cdot (p_{bd}(t)-x_{jd}(t))\right]$$

$$c_{kr}(t+1)=v_{kr}(t+1)+c_{kr}(t) \tag{19}$$

The position $c_{kr}(t+1)$ may not be an integer vector because of the constriction coefficient and the random number in the updating formulae. Thus, there is a modification of position proposed as

$$c_{kr}(t+1)=\begin{cases} N & c_{kr}(t+1)>N \\ 1 & c_{kr}(t+1)<1 \\ \lfloor c_{kr}(t+1)\rfloor & \text{Others} \end{cases} \tag{20}$$

If the position value $c_{kr}(t+1)$ is bigger than the target number, it is restricted in the last target. If the position value $c_{kr}(t+1)$ is less than 1, it is restricted in the first target. Otherwise, all the position values are rounded down to make sure the whole positions are integer within the range.

It is essential to restrict velocity vector in a certain range to make sure that position vector is not updated too fast

$$v_{kr}(t+1)=\begin{cases} v_{\max,r} & v_{kr}(t+1)>v_{\max,r} \\ -v_{\max,r} & v_{kr}(t+1)<v_{\max,r} \\ v_{kr}(t+1) & \text{others} \end{cases}$$

$$v_{\max}=N-1 \tag{21}$$

According to the coordinated attack tactics above, more constraint conditions are taken into consideration. Each missile can only attack one enemy fighter target. Each target is attacked twice at most. The Boolean value of the missile is constrained as

$$\begin{aligned} &\sum_{j=1}^{n} X_{rj}=1 \quad r=1,2,\cdots,Z \\ &\sum_{r=1}^{Z} X_{rj}\leqslant 2 \quad j=1,2,\cdots,N \end{aligned} \tag{22}$$

This series of constraints are used to check the solution of MTA problem and make some adjustments if necessary. The steps are as follows:

Step 1 Denote a set A which includes all the values need to be changed. If the same position value exists in the position vector $\boldsymbol{\pi}_k$ more than twice, two of them are chosen randomly and others are saved in set A.

Step 2 Denote two sets S_0 and S_1. S_0 includes targets in set [1 N_red] which have not appeared in the solution before. S_1 includes targets in set [1 N_red] that have appeared in the solution only once.

Step 3 Make some adjustments to set A. Assume that the value of the position c_{kr} needs to be changed and the updated position value is c_s. c_s should belongs to $\{S_0 \quad S_1\}$. The principle of choosing targets is given by

$$c_s=\arg\min\{d(c_s \quad c_{kr})\} \tag{23}$$

where $d(c_s \quad c_{kr})$ is the distance between c_{kr} and c_s. Then, the element c_{kr} is removed from set A.

Step 4 Update the two sets S_0 and S_1. If there is $c_s\in S_0$, c_s would be saved in S_1 and removed from S_0. If there is $c_s\in S_1$, the elements in S_0 and S_1 would not be changed.

Step 5 Repeat Steps 3, 4 until set A becomes a null set.

4 Simulation Experiment of IPSO for CMTA

Assume that our fighters B and enemy fighters R are in a BVR air combat. Our fighters B adopt CMTA strategy. In this simulation, there are four our fighters and each fighter has four missiles. Thus, the number of the missiles to attack the enemy fighter targets is 16. The velocity of our fighters is 300 m/s. The effective striking distance of missiles carried by our fighters is 70 km. The maximum tracking range of our fighters is 120 km. There are fourteen enemy fighter targets. The velocity of enemy fighters is 300 m/s. The effective striking distance of missiles carried by our fighters is the same as that carried by the enemy fighters. The maximum tracking range of our fighters is the same as that of the enemy fighters. In a random scenario, our fighters and enemy fighters aviate face to face. The air combat situation is shown in Fig. 3.

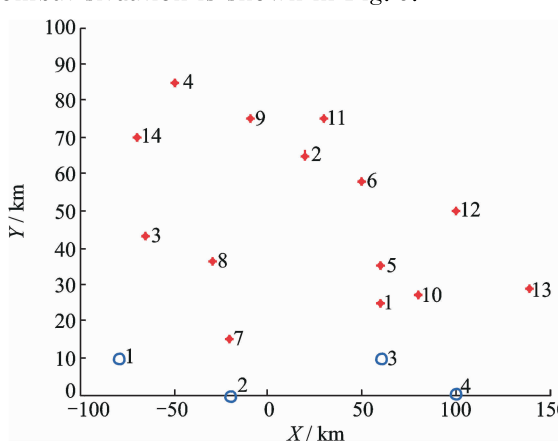

Fig. 3 The air combat situation

Then, the IPSO algorithm designed above is used to present a DM proposal of MTA problem in CMTA. The traditional PSO algorithm is also simulated here to compare with the IPSO algorithm. The constriction coefficient χ is set to be 1. The assignment of all the missiles is

$$\boldsymbol{\pi}=[2 \quad 8 \quad 7 \quad 3 \quad 5 \quad 6 \quad 1 \quad 5 \quad 13 \quad 10 \quad 12 \quad 10 \quad 1 \quad 13 \quad 14 \quad 13]$$

Fig. 4 illustrates the DM proposal of MTA problem. Based on the IPSO algorithm, the missiles carried by our fighter 1 attack enemy fighters 2, 8, 7 and 3. The missiles carried by our fighter 2 attack enemy fighters 5, 6, 1 and 5. The missiles carried by our fighter 3 attack enemy fighters 13, 10, 12 and 10. The missiles carried by our fighter 4 attack enemy fighters 1, 13, 14 and 13. The repeated numbers imply that these enemy fighters threaten our fighters too much and are attacked twice as a result. Some enemy fighters are not attacked because their threat values do not reach the threat threshold value. With the traditional PSO algorithm employed, the missiles carried by our fighter 1 attack enemy fighters 2, 3, 3 and 7. The missiles carried by our fighter 2 attack enemy fighters 1, 1, 14 and 5. The missiles carried by our fighter 3 attack enemy fighters 13, 13, 8 and 8. The missiles carried by our fighter 4 attack enemy fighters 10, 10, 5 and 6. The IPSO algorithm based DM proposal of MTA problem makes full use of the missiles and destroys more threats.

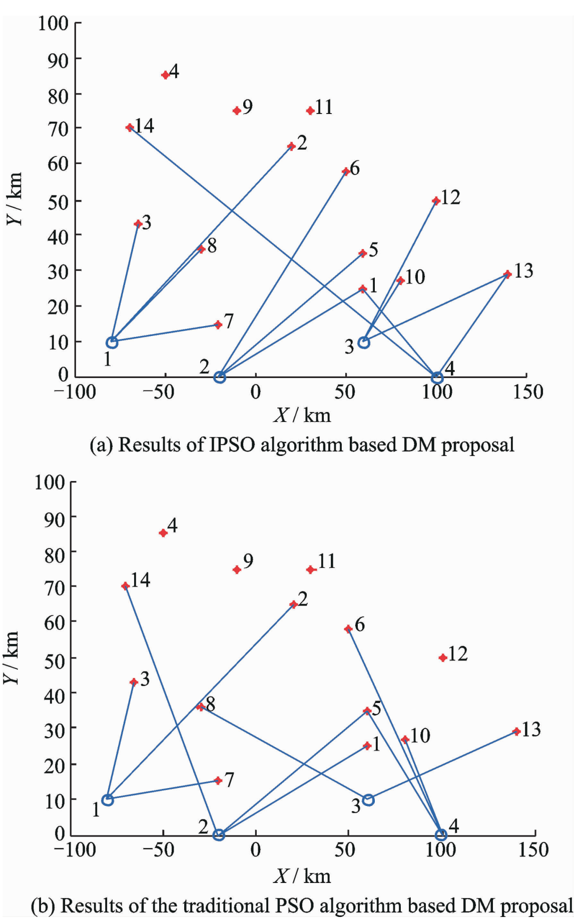

Fig. 4 Results of DM for MTA

Fig. 5 shows the fitness of iteration process. The fitness can decreased to 4.391 5 when using the IPSO algorithm, while the fitness is 4.568 8 with the traditional PSO algorithm. What's more, the DM proposal with the IPSO algorithm is faster than that with the PSO algorithm due to the less iterations when using the IPSO algorithm.

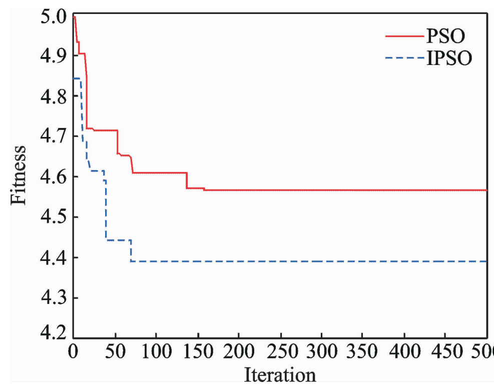

Fig. 5 Fitness of iteration process

5 Conclusions

DM problem for MTA in an air combat is solved by a new improved PSO algorithm which is parametric simple but effective and efficient. The IPSO algorithm is used to minimize fitness function constructed by threat value. Coordinated attack tactics is considered to adjust DM proposal to reach better strike effect. It exhibits better performance to CMTA in an air combat with the IPSO algorithm compared with the traditional PSO algorithm.

Acknowledgement

This work was jointly granted by the Science and Technology on Avionics Integration Laboratory and the Aeronautical Science Foundation of China (No. 2016ZC15008).

References:

[1] AKBARI S, MENHAJM B. A new framework support system for air to air tasks [C]// IEEE Proceedings of the International Conference on SMC. Nashville, TN, USA:[s. n.], 2000, 3: 2019-2022.

[2] SECAREA V V, KRIKORIAN H F. Adaptive multiple target attack planning in dynamically changing hostile environments [C]// IEEE Proceedings of the

National Aerospace and Electronics Conference. Dayton, OH, USA: [s. n.], 1990, 3: 29-34.

[3] LUO D L, WU W H, SHEN C L. A survey on decision-making for multi-target attacking in air combat [J]. Electronic Optics and Control, 2005, 12(4): 4-8.

[4] ZHOU S Y, WU W H, ZHANG N, et al. Overview of autonomous air combat maneuver decision [J]. Aeronautical Computing Technique, 2012, 42(1): 27-31.

[5] HUANG H Q, WANG Y, ZHOU H, et al. Multi-UCAV cooperative autonomous attack path planning method under uncertain environment [C]// Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC). Xi'an: IEEE, 2016: 573-579.

[6] ZHANG Y, LI C. Coordinated attack strategy of network sub-munitions based on particle swarm optimization [J]. Journal of Detection and Control, 2015, 37(1): 99-103.

[7] LI Z W, CHANG Y Z, SUN Y Y, et al. A decision-making for multiple target attack based on characteristic of future long-range cooperative air combat [J]. Fire Control and Command Control, 2016, 47(2): 36-40.

[8] LUO D L, SHEN C L, WANG B, et al. Air combat decision-making for cooperative multiple target attack using heuristic adaptive genetic algorithm [C]// Proceedings of the Fourth International Conference on Machine Learning and Cybernetics. Guangzhou: IEEE, 2005: 473-478.

[9] ZHANG T, YU L, WEI X Z, et al. Decision-making for cooperative multiple target attack based on adaptive pseudo-parallel genetic algorithm [J]. Fire Control and Command Control, 2013, 38(5): 137-140.

[10] LI Linsen, TONG Mingan. Air combat decision of cooperative multi-target attack and its neural net realization [J]. Acta Aeronautic et Astronautic Sinica, 1999, 20(4): 309-312. (in Chinese)

[11] LI L S, YU H X, HAN Z G, et al. Application of a type of associated neural network in cooperative air to air combat analysis [J]. Flight Dynamics, 2000, 18(1): 81-84.

[12] GENG Y L, JING C S, LI W H. Multi-fighter coordinated multi-target attack system [J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2004, 21(1): 18-23.

[13] ROGER W S, ALAN E B. Neural network models of air combat maneuvering [D]. New Mexico: New Mexico State University, 1992.

[14] SONG X G, JIANG J, XU H Y. Application of improved simulated annealing genetic algorithm in cooperative ari combat [J]. Journal of Harbin Engineering University, 2017, 38(11): 1762-1768.

[15] AUSTIN F. Game theory for automated maneuvering during air to air combat [J]. Guidance, 1990, 13(6): 1143-1147.

Prof. **Ding Yongfei** joined in the Key Laboratory of Aviation Electronic Systems Integrated Technology. His research is focused on automatic control and integrated avionics system.

Ms. **Yang Liuqing** received her B. Sc. degree in automation from Nanjing University of Aeronautics and Astronautics (NUAA) in 2016. From 2016 to present, she is a postgraduate student of Navigation, Guidance and Control in College of Automation Engineering in NUAA. Her research interest focuses on advanced flight control.

Mr. **Hou Jianyong** is currently a member of China National Aeronautical Radio Electronics Research Institute. His research interest focuses on flight control and avionics of aircrafts.

Ms. **Jin Guting** is currently a member of China National Aeronautical Radio Electronics Research Institute. Her research interest focuses on the cooperation of manned/unmanned aerial vehicles.

Dr. **Zhen Ziyang** is currently an associate professor in College of Automation Engineering, Nanjing University of Aeronautics and Astronautics. His research interests are flight control of UAVs and hypersonic vehicle, preview control and adaptive control.

(Production Editor: Xu Chengting)