

# 基于 Voronoi 图和离散微粒群优化算法的UCAV攻击轨迹决策

丁达理<sup>1,2</sup>, 宋磊<sup>1</sup>, 贺建良<sup>3</sup>, 张洪波<sup>3</sup>

(1. 空军工程大学工程学院, 陕西西安, 710038; 2. 西北工业大学航天学院, 陕西西安, 710072; 3. 光电控制技术重点实验室, 河南洛阳, 471009)

**摘要** 根据UCAV可发射区简化模型,提出了UCAV攻击轨迹决策问题。提出了UCAV攻击轨迹决策的分层设计方法,并将其转化为优化问题。首先根据已知的飞行环境,采用Voronoi图生成初始轨迹;然后通过建立攻击轨迹约束条件模型、飞行距离模型和威胁模型,以飞行距离和威胁代价为优化目标函数,构建了UCAV攻击轨迹决策模型,并在此基础上提出了一种离散微粒群优化(Discrete Particle Swarm Optimization, DPSO)算法对攻击轨迹决策模型进行求解。仿真结果表明,通过Voronoi图和DPSO算法能够较好地解决UCAV攻击轨迹决策问题,并能够在多约束条件下对目标函数进行组合优化。

**关键词** 无人作战飞机;攻击轨迹;离散微粒群优化算法;Voronoi图

**DOI** 10.3969/j.issn.1009-3516.2012.02.001

**中图分类号** V279 **文献标识码** A **文章编号** 1009-3516(2012)02-0001-05

UCAV攻击轨迹决策即在一定的约束条件下,综合考虑UCAV的生存概率和对目标的毁伤效果,以优化目标函数的方式,决策出UCAV快速进入武器系统可发射区的攻击轨迹,以使其攻击效能达到最优或近似最优<sup>[1-3]</sup>。本文采用分层设计的思想,首先根据已知的任务飞行环境(地形和威胁),采用Voronoi图生成初始轨迹;综合考虑各种约束条件,赋予各轨迹相应的权值,构建优化目标函数,采用离散微粒群优化(Discrete Particle Swarm Optimization, DPSO)算法在轨迹空间中搜索出一条最优攻击轨迹。这种设计思想使得UCAV决策系统在实时给出可行轨迹的同时,还能够在任务和约束之间寻找平衡,并且可以及时处理不确定因素。

## 1 基于Voronoi图的初始轨迹生成

可发射区的大小和形状受到载机所采用的攻击方式、机动特性、武器特性(射程、引信启动时间及导引制导能力)、目标探测系统的特性和安全要求等因素的限制,其真实范围非常复杂<sup>[4]</sup>。本文以某UCAV激光制导武器系统为研究对象,为得到可发射区的简化数学模型,假设:①目标固定或低速;②目标位置和威胁点位置均已知且UCAV采用水平机动攻击方式,其攻击速率恒定,可发射区仅与发射距离、航向角有关,在目标已知的情况下,可发射区是确定的。因此,将可发射区简化为由目标距离边界和偏航角侧向边界组成的水平面区域。

图1中可发射区前后边界线1和2分别代表武器发射距离边界(主要由武器导引头最大俯仰角 $\theta_{\max}$ 限制的最大发射距离 $R_{\max}$ 、最小发射区距离 $R_{\min}$ 等参数决定);可发射区左右边界线3和4分别代表UCAV与目标的偏航角侧边界(主要由导引头最大水平视场角偏离角 $\psi_+$ 、 $\psi_-$ )。

Voronoi图(以下简称V图)又叫泰森多边形或Dirichlet图,由一组由连接两邻点直线的垂直平分线组成的连续多边形组成。基于V图进行UCAV初始轨迹生成的基本思想:先将已知的敌方威胁中心位置作为

\* 收稿日期:2011-08-20

基金项目:国家自然科学基金资助项目(61004124);航空科学基金资助项目(20105196016)

作者简介:丁达理(1980-),男,湖南益阳人,讲师,博士,主要从事武器系统运用工程研究。

E-mail: ddl-123@tom.com

点,以威胁大小作为邻近区域的“距离”量度,“距离”越大则所受威胁越小<sup>[5]</sup>。V 图的各条边在相应点的邻域内距威胁源“距离”最大,因而所受威胁相应最小,所以 V 图中的弧即构成飞行器安全性最高的可选择攻击轨迹,线段与线段的交点即构成可飞轨迹节点,从而可以根据威胁源的强度大小给出各条弧的相应权值,最后利用离散 PSO 算法,求取最优飞行轨迹。

无人机在执行打击任务时面临的防空火力主要包括各种地空导弹和高炮等,其威胁度计算模型见式(1),威胁源作用范围内的飞行器所受到的威胁度与飞行器到威胁源的距离构成泊松分布关系<sup>[6]</sup>,即:

$$p(r) = Ke^{-mr} \quad (1)$$

式中: $p(r)$ 为无人机与威胁源距离为 $r$ 时所受到的威胁度; $K$ 为威胁源中心的强度; $m$ 为衰减因子。

通常认为雷达的探测能力与距离的 4 次方成反比。即:

$$p_{\text{Radar}} = \begin{cases} \frac{K_i^{\text{detect}}}{r^4}, & 0 < R \leq R_{\text{max}} \\ 0, & \text{其它} \end{cases}$$

式中: $r$ 为雷达到无人机的距离; $K_i^{\text{detect}}$ 与雷达自身的探测能力有关的一个常数; $R_{\text{max}}$ 为雷达最大探测半径。

因此,将威胁源与 V 图弧之间距离的 4 次方的倒数赋予 V 图相应的弧权值,得到 UCAV 在弧上飞行时受到的威胁。其具体计算公式为:

$$w(l_i) = 1 - \sum_{j=1}^n (1 - p_{j\_value} / D^4(l_i, p_j)) \quad (2)$$

式中: $l_i$ 为 V 图中的弧, $i = 1, 2, \dots, N$ , $N$ 为弧的总数; $n$ 表示与弧 $l_i$ 直接相关的威胁点个数; $p_j$ 为威胁点; $D(l_i, p_j)$ 为威胁点 $p_j$ 到弧 $l_i$ 的距离; $p_{j\_value}$ 为威胁点 $p_j$ 的威胁强度,可以由公式(1)求得。

## 2 UCAV 攻击轨迹建模

定义决策变量 $x_i$ ,当 V 图中弧 $l_i$ 包含在满足约束条件的 UCAV 可行轨迹中时, $x_i = 1$ ,否则 $x_i = 0$ 。由于 UCAV 攻击必须满足武器系统最大视场角和最大机动过载的限制条件,为此,构建 UCAV 约束条件模型为:

$$x_i = \begin{cases} 1, & (\psi_{i-} \leq \psi \leq \psi_{i+}) \& (\vartheta \leq \vartheta_{i0}) \& (a \leq a_0) \\ 0, & \text{else} \end{cases} \quad (3)$$

式(3)表示 UCAV 的 3 个约束条件:①目标视线偏航角 $\psi$ 在探测系统最大允许视场偏航角 $[\psi_{i-}, \psi_{i+}]$ 之间;②目标视线俯仰角 $\vartheta$ 小于探测系统最大允许视场俯仰角 $\vartheta_{i0}$ ;③机动过载小于最大允许值 $a_0$ 。只有满足这 3 个约束条件,才认为 V 图中弧 $l_i$ 为可行轨迹。

UCAV 飞行距离应尽可能短,则 UCAV 飞行距离代价函数为:

$$\min J_1 = \sum_{i=1}^N D(l_i) x_i \quad (4)$$

式中 $D(l_i)$ 为弧 $l_i$ 的长度。

UCAV 攻击目标既要保证攻击效果最优,又要保证自身受到的威胁尽可能小,UCAV 威胁代价模型为:

$$\min J_2 = \sum_{i=1}^N w(l_i) x_i \quad (5)$$

对于采用加权线性累加将目标函数整合为单一目标函数:

$$\min J = m_1 J'_1 + m_2 J'_2 \quad (6)$$

式中: $J'_1$ 、 $J'_2$ 分别是 $J_1$ 、 $J_2$ 量纲转换后的目标函数; $m_1$ 、 $m_2$ 为加权系数,表示威胁和距离在弧的权值中所占的比重,且 $\sum_{i=1}^2 m_i = 1$ 。显然,在算法的优化过程中,每一组权向量将决定目标空间中的一个搜索方向。为了获得尽可能多的非劣解,权值按如下方式确定:

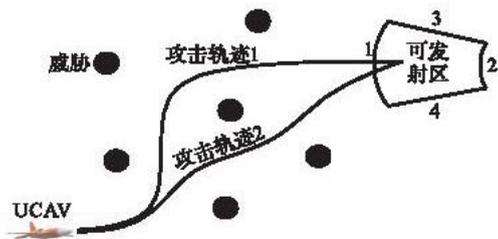


图1 攻击轨迹决策示意图

Fig. 1 Attack trajectory decision - making sketch map

$$m_i = \frac{\text{rand}_i}{\sum_{i=1}^n \text{rand}_i} \quad (7)$$

rand<sub>*i*</sub> 为 0 - 1 之间均匀分布的随机数。种群中的每一个微粒均由式(7)确定一组权值。

### 3 DPSO 算法

由UCAV攻击轨迹模型可以看出,该问题是离散优化问题,不能直接应用标准微粒群优化算法(Particle Swarm Optimization, PSO)<sup>[7-8]</sup>。因此,本文通过对PSO算法进行处理,在PSO优化框架内重新定义微粒的位置、速度及相关的操作,提出了一种离散微粒群优化算法(Discrete PSO, DPSO),以保证算法具有较好的优化性能。

根据微粒群的优化机制可知,微粒的新位置是其速度、自身最佳位置  $P_i$  和群体最佳位置  $P_g$  相互作用的结果<sup>[9-10]</sup>。因此,设计如下的位置更新公式:

$$X_i(k+1) = c_2 \otimes \Phi_{a,b}(c_1 \otimes \Phi_{a,b}(\omega \otimes S_{a,b}(X_i(k)), P_i(k)), P_g(k)) \quad (8)$$

式中: $\omega$ 为惯性系数; $c_1$ 为认知系数; $c_2$ 为社会系数; $X_i(k)$ 表示微粒*i*在第*k*代迭代中的位置; $P_i(k)$ 表示微粒*i*在第*k*代迭代中的自身最佳位置; $P_g(k)$ 表示第*k*代迭代中的群体最佳位置; $S_{a,b}(\cdot)$ 表示对微粒位置矢量的第*a*维分量和第*b*维分量进行交换操作, $\Phi_{a,b}(\cdot)$ 表示对2个微粒位置矢量的第*a*维分量与第*b*维分量之间的数值进行交叉操作, $a, b$ 为随机产生的自然数。

位置更新公式(8)可以认为由3部分组成:

1)微粒对速度的思考:

$$E_i(k) = \omega \otimes S_{a,b}(X_i(k)) = \begin{cases} S_{a,b}(X_i(k)), & \text{rand}() < \omega \\ X_i(k), & \text{rand}() \geq \omega \end{cases} \quad (9)$$

该部分是微粒的惯性部分。其中  $S_{a,b}(X_i(k))$  表示微粒的速度,即为一个概率为  $\omega$  的置换操作。如果产生的随机数  $\text{rand}() < \omega$ ,则随机交换微粒位置的第  $a$  维与第  $b$  维分量,即  $E_i(k) = S_{a,b}(X_i(k))$ ,见图2;否则, $E_i(k) = X_i(k)$ 。

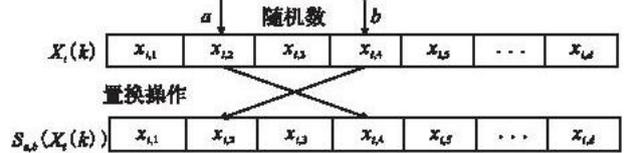


图2 目标置换操作

Fig.2 Target transposition operation

2)微粒对自身的认知

$$F_i(k) = c_1 \otimes \Phi_{a,b}(E_i(k), P_i(k)) = \begin{cases} \Phi_{a,b}(E_i(k), P_i(k)), & \text{rand}() < c_1 \\ E_i(k), & \text{rand}() \geq c_1 \end{cases} \quad (10)$$

该部分是微粒的认知部分,可理解为根据  $P_i(k)$  调整自身位置。 $c_1 \otimes \Phi_{a,b}(E_i(k), P_i(k))$  表示一个概率为  $c_1$  的交叉操作。如果产生的随机数  $\text{rand}() < c_1$ ,则将  $E_i(k)$  的第  $a$  维与第  $b$  维之间的分量用  $P_i(k)$  对应的分量替换,即交叉操作,见图3;否则, $F_i(k) = E_i(k)$ 。

3)微粒向社会的学习:

$$X_i(k+1) = c_2 \otimes \Phi_{a,b}(F_i(k), P_g(k)) = \begin{cases} \Phi_{a,b}(F_i(k), P_g(k)), & \text{rand}() < c_2 \\ F_i(k), & \text{rand}() \geq c_2 \end{cases} \quad (11)$$

该部分是微粒的社会部分,可理解为根据  $P_g(k)$  调整自身位置。 $c_2 \otimes \Phi_{a,b}(F_i(k), P_g(k))$  表示一个概率为  $c_2$  的交叉操作。如果产生的随机数  $\text{rand}() < c_2$ ,则将  $F_i(k)$  的第  $a$  维与第  $b$  维之间的分量用  $P_g(k)$  对应的分量替换,即交叉操作,见图4;否则, $X_i(k+1) = F_i(k)$ 。

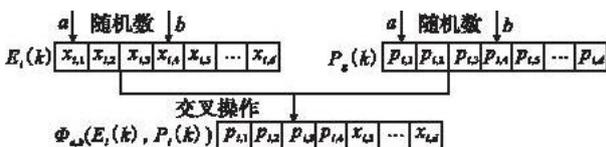


图3 与自身最佳位置的交叉操作

Fig.3 The intercrossing operation with itself optimal position

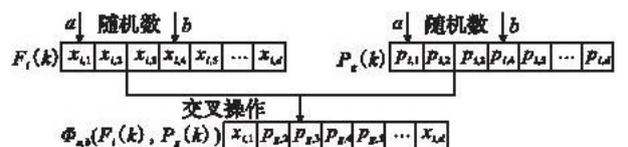


图4 与群体最佳位置的交叉操作

Fig.4 The intercrossing operation with colony optimal position

结合攻击轨迹建模,以威胁代价  $J$  (由式(6)计算得) 为粒子群算法的适应度值,  $V$  图中所有满足约束条件的可行弧的排列作为微粒的位置矢量,每一维对应一段弧,因此,记微粒的位置矢量  $\mathbf{X}_i$  表示轨迹决策模型一个可行解。当  $x_{i,j} = 1$  表示无人机经该弧段飞向攻击区,否则不经过。

综上所述,可以得到基于 DPSO 算法的 UCAV 攻击轨迹决策的具体步骤:

步骤 1 初始化参数  $\omega, c_1, c_2$ 。

步骤 2 初始化微粒的位置矢量  $\mathbf{X}_i(0) = (x_{i,1}, x_{i,2}, \dots, x_{i,d})$ , 每一维随机取 0 或 1。

步骤 3 令  $\mathbf{P}_i(0) = \mathbf{X}_i(0)$ ; 根据目标函数式(6), 评价各微粒的适应度值, 并根据  $J(\mathbf{P}_g(0)) = \min\{J(\mathbf{X}_1(0)), J(\mathbf{X}_2(0)), \dots, J(\mathbf{X}_m(0))\}$  确定群体最佳位置  $\mathbf{P}_g(0)$ ;

步骤 4 重复执行下列步骤, 直到群体最佳位置的适应度值满足要求:

步骤 4.1 令  $k = k + 1$ , 随机产生  $\text{rand}()$ 、 $a, b$ , 采用公式(8)更新  $\mathbf{X}_i(k)$ ;

步骤 4.2 计算各微粒的适应度值  $J(\mathbf{X}_1(k)), J(\mathbf{X}_2(k)), \dots, J(\mathbf{X}_m(k))$ ;

步骤 4.3 更新各微粒的  $\mathbf{P}_i(k)$  以及群体的  $\mathbf{P}_g(k)$ ;

步骤 5 输出与群体最佳位置对应的  $V$  图中弧序列, 即 UCAV 进入可发射区的最优轨迹。

## 4 仿真分析

假设在  $50 \times 50 \text{ km}^2$  的数字地图上分布有 26 个威胁点  $p_i$ , 其中  $A$  点为打击目标位置, 弧 1, 2, 3, 4 围成的区域为  $A$  点的可发射区, 见图 5。

在仿真验证中粒子个数为 50, 维数为 56, 将  $V$  图中弧段按照 1-56 的顺序依次编号, 并设初值  $m_1 = 0.4, m_2 = 0.6, \omega = 0.3, c_1 = 0.5, c_2 = 0.5$ 。最大俯仰角限制为  $\vartheta \leq 60^\circ$ , 最大偏航角限制为  $[-90^\circ, +90^\circ]$ , 最大过载限制为  $15 \text{ m/s}^2$ 。仿真中适应度函数  $J(X_i(k))$  为从起始点依次经历粒子中的节点能到达可发射区所有连通的路径的最小值, 其适应度曲线见图 6。由图可见, 适应度函数随迭代次数的增加迅速降低, 当迭代达到 97 次时, 算法已经收敛, 并获得最优路径。仿真过程输出: 适应度值为 38.5, 总飞行路程 82.1 km, 仿真耗时 9.3 s。

图 5 中, 粗黑线代表仿真得到的最优 UCAV 攻击轨迹。可以看出, 仿真得到的优化攻击轨迹在满足约束条件要求的前提下具有威胁代价最小、飞行距离最短的特点, 实现了 UCAV 攻击轨迹决策的优化与求解。

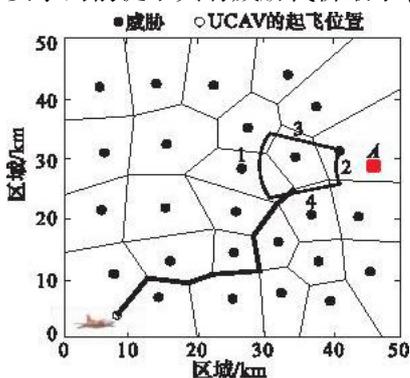


图 5 UCAV 攻击轨迹决策结果

Fig. 5 The result of attack trajectory decision-making

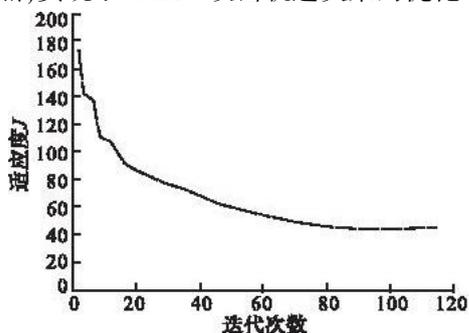


图 6 适应度函数拟合曲线图

Fig. 6 The adaptable degree function fitting curve graph

## 5 结论

本文针对 UCAV 进入可发射区的特点, 根据已知的目标和威胁, 采用  $V$  图生成初始轨迹, 然后根据建立的 UCAV 攻击轨迹决策模型, 采用一种改进型的 DPSO 算法进行求解。仿真结果表明, 本文设计的 DPSO 算法可以较好地解决 UCAV 攻击轨迹决策问题。同时, 该算法也可用于求解多约束的组合优化问题。

## 参考文献(References):

- [1] Reid A L, Meir P, Mark J M. Path planning by unmanned air vehicles for engaging an integrated radar network[R]. AIAA 2005 – 6191.
- [2] Serdar E, Mahmut K, Gary G. Advanced task assignment for unmanned combat aerial vehicles targeting cost efficiency and survivability[R]. AIAA 2008 – 873.
- [3] 王庆江, 高晓光. 激光制导炸弹的仿真及其有效投弹区的计算[J]. 火力指挥与控制, 2008, 33(8): 44 – 47.  
WANG Qingjiang, GAO Xiaoguang. Simulation and calculation of effective attack area about laser – guided bomb[J]. Fire control and command control, 2008, 33(8): 44 – 47. (in Chinese)
- [4] 叶媛媛, 闵春平, 沈林成, 等. 基于 Voronoi 图的无人机空域任务规划方法研究[J]. 系统仿真学报, 2005, 17(6): 1353 – 1356.  
YE Yuanyuan, MIN Chunping, SHEN Lincheng, et al. Voronoi diagram based spatial mission planning for UAVs[J]. Journal of system simulation, 2005, 17(6): 1353 – 1356.
- [5] Wein R, Berg J P, Halperin D. The visibility – voronoi complex and its applications[C]//European workshop on computational geometry. London: [s. n.], 2007: 66 – 87.
- [6] Eun Y, Bang H. Cooperative control of multiple unmanned aerial vehicles using the potential field theory[J]. Journal of aircraft, 2006, 43(6): 1805 – 1813.
- [7] 叶文, 朱爱红, 潘长鹏, 等. 多UCAV 协同目标分配算法研究[J]. 系统工程与电子技术, 2010, 32(1): 104 – 108.  
YE Wen, ZHU Aihong, PAN Changpeng, et al. Cooperation mission assignment algorithm for multi – UCAV[J]. Systems engineering and electronic, 2010, 32(1): 104 – 108. (in Chinese)
- [8] Curz J B, Jr C, Chen G. Particle swarm optimization for resource allocation in UAV cooperative control[R]. AIAA 2004 – 5250.
- [9] Parsopoulos K E, Vrahatis M N. Particle swarm optimization method in multi – objective problems[C]//Proc of ACM symposium on applied computing. Madrid, Spain: ACM press, 2002: 603 – 607.
- [10] 纪震, 廖惠连, 吴青华. 粒子群算法及应用[M]. 北京: 科学出版社, 2009: 16 – 25.  
JI Zhen, LIAO Huilian, WU Qinghua. Particle swarm arithmetic and application[M]. Beijing: Science press, 2009: 16 – 25. (in Chinese)

(编辑: 徐敏)

## Attack Trajectory Decision – Making for UCAV Based on Voronoi Diagram and DPSO Algorithm

DING Da – li<sup>1,2</sup>, SONG Lei<sup>1</sup>, HE Jian – liang<sup>3</sup>, ZHANG Hong – bo<sup>3</sup>

(1. Engineering Institute, Air Force Engineering University, Xi'an 710038, China; 2. College of Astronautics, Northwestern Polytechnical University, Xi'an 710072, China; 3. Science and Technology on Electro – optic Control Laboratory, Luoyang 471009, Henan, China)

**Abstract:** According to simplified model of UCAV launching area, the attack trajectory decision – making for UCAV is introduced. A hierarchical trajectory – designing approach is proposed, and the problem is changed into optimization design. Firstly, the original trajectory is created by Voronoi diagram based on the known flight environment. Secondly, attack trajectory decision making model of UCAV is built based on attack trajectory constraint conditions, flight distance model and threaten model. And the discrete particle swarm optimization is presented to solve it. The simulation results show that DPSO algorithm is simple and can be used to effectively solve the problem of attack trajectory decision – making for UCAV.

**Key words:** UCAV; attack trajectory; discrete particle swarm optimization (DPSO); Voronoi diagram