http://bhxb.buaa.edu.cn jbuaa@buaa.edu.cn
DOI: 10.13700/j.bh.1001-5965.2017.0589

基于 SAGWO 算法的 UCAVs 动态协同任务分配

魏政磊[1,*], 赵辉[1], 黄汉桥[2], 王骁飞[1], 周瑞[1]
(1. 空军工程大学 航空工程学院, 西安 710038; 2. 西北工业大学 无人机系统技术研究院, 西安 710072)

摘 要: 通过分析无人作战飞机(UCAV)优势概率和任务联合威胁以及定义任务时间,建立了以目标价值毁伤、编队损耗代价和时间消耗为性能指标的多无人作战飞机(UCAVs)多约束动态任务分配数学模型,采用改进的灰狼优化(GWO)算法对数学模型进行求解;针对基本 GWO 算法求解早熟的缺点,给出了自适应调整策略和跳出局部最优策略,引入了二次曲线控制方法;对 UCAVs 动态协同任务分配特点,设计了目标任务序列编码方式,提出了基于自适应 GWO(SAGWO)算法的 UCAVs 多目标动态任务分配方法。从静态与动态 2 种情况分别对该方法进行仿真验证;仿真结果表明,该方法是有效的,相比较于其他算法,其优化过程快速精准。

关 键 词: 多无人作战飞机(UCAVs); 动态协同任务分配; 目标依赖矩阵; 任务时间片; 自适应灰狼优化(SAGWO)算法
中图分类号: V279
文献标识码: A **文章编号**: 1001-5965(2018)08-1651-14

随着作战环境的复杂化、作战过程的动态化、作战任务的多样化,多无人作战飞机(Unmanned Combat Aerial Vehicles, UCAVs)协同攻击多目标将是一种未来的重要作战模式。UCAVs 协同任务分配是 UCAV 协同作战的关键技术之一,其目的是在一定战场环境和任务要求下,考虑执行任务的约束条件和技术指标,将目标合理地分配给不同的 UCAV,使得整个编队作战效能达到最大。

目前,对 UCAVs 协同任务分配技术的研究已经成为 UCAV 作战研究中的热点。针对任务分配问题,文献[1]采用多种协商机制建立分布式 UCAVs 控制系统;文献[2]提出了基于多层树的协同攻击多目标模型;也可以采用智能优化算法对 UCAVs 任务分配问题求解,比如萤火虫优化算法[3]、粒子群算法[4]等。在实际的 UCAV 协同任务执行过程中,可能会出现任务获取信息不完整,或者在任务执行过程中存在战场环境和任务需求以及任务信息的变化,导致现有的任务分配方案对实时环境的不适应[5-6]。同时,相关文献中的任务分配算法模型同时或部分存在以下问题,如:①未考虑目标之间的支援关系;②未考虑电子干扰下的任务分配;③动态的任务分配没有用时间指标体现出来。

针对动态任务分配的特点,同时考虑到以上存在的问题,本文采用了时间片下基于改进灰狼优化(Grey Wolf Optimization, GWO)算法[7]的任务分配方法。通过对 UCAV 编队信息和任务目标信息的分析,提出了基于时间片的多约束动态任务分配模型;针对动态任务分配数学模型,采用了自适应灰狼优化(Self-Adaptive GWO, SAGWO)算法进行求解;最后进行了仿真验证。

收稿日期: 2017-09-21; **录用日期**: 2017-12-11; **网络出版时间**: 2018-01-31 16:34
网络出版地址: kns.cnki.net/kcms/detail/11.2625.V.20180131.1611.001.html
基金项目: 国家自然科学基金(61601505); 航空科学基金(20155196022)
* **通信作者**. E-mail: zhenglei_wei@126.com

引用格式: 魏政磊, 赵辉, 黄汉桥, 等. 基于SAGWO算法的UCAVs动态协同任务分配[J]. 北京航空航天大学学报, 2018, 44(8): 1651-1664. WEI Z L, ZHAO H, HUANG H Q, et al. Dynamic UCAVs cooperative task allocation based on SAGWO algorithm [J]. Journal of Beijing University of Aeronautics and Astronautics, 2018, 44(8): 1651-1664 (in Chinese).

http://bhxb.buaa.edu.cn

1 UCAVs 动态任务分配模型

本文以 UCAV 执行空对地打击任务为背景，假设我方有 M 架无人作战飞机 $V=\{V_i, i=1,2,\cdots,M\}$，敌方地面有 N 个确定目标 $T=\{T_j, j=1,2,\cdots,N\}$，其中 V_i 表示我方的第 i 个 UCAV，T_j 表示地面第 j 个任务目标。设目标的分配矩阵为 $\boldsymbol{X}_{M\times N}$，当 X_{ij} 为 1 时，表示第 i 个 UCAV 对第 j 个任务目标进行攻击；X_{ij} 为 0 时，表示 V_i 未分配到 T_j。假设我方第 i 个 UCAV 可携带 L_i 枚导弹，则编队可携带的导弹数量为 $L=\sum_{i=1}^{M} L_i$。

假设攻击目标的 L 枚导弹构成集合 $K=\{k, k=1,2,\cdots,L\}$，则 UCAV 编队的第 k 枚导弹与无人机 V_i 的第 l 枚导弹对应的关系为

$$k=\sum_{i=1}^{i-1} L_i + l \tag{1}$$

式中：$i=\{1,2,\cdots,M\}$；$l=\{1,2,\cdots,L_i\}$。

1.1 UCAV 优势概率

UCAV 对目标的优势概率与 UCAV 距离威胁因子、速度威胁因子和角度威胁因子有关。UCAV 编队中 V_i 对目标 T_j 的优势概率可以表示为

$$P_{\mathrm{K}ij}=c_1 T_{ij,D} T_{ij,\theta} + c_2 T_{ij,v} \tag{2}$$

式中：c_1 和 c_2 分别为无人机 V_i 相对任务目标 T_j 的距离-角度威胁和速度威胁的权重；$T_{ij,D}$、$T_{ij,\theta}$ 和 $T_{ij,v}$ 分别为距离威胁概率、角度威胁概率和速度威胁概率[8]。

距离威胁概率为

$$T_{ij,D}=\begin{cases}1 & d_{ij}\leqslant r_{\mathrm{W}}\\ \dfrac{R_{\mathrm{W}\max}-d_{ij}}{R_{\mathrm{W}\max}-r_{\mathrm{W}}} & r_{\mathrm{W}}<d_{ij}\leqslant R_{\mathrm{W}\max}\\ 0 & d_{ij}>R_{\mathrm{W}\max}\end{cases} \tag{3}$$

式中：d_{ij} 为 V_i 相对于 T_j 的距离；r_{W} 为我方 UCAV 导弹的有效作用距离；$R_{\mathrm{W}\max}$ 为 UCAV 的最大跟踪距离。

角度威胁概率为

$$T_{ij,\theta}=\exp(-\lambda_1(\pi\theta_{ij}/180)^{\lambda_2}) \tag{4}$$

式中：θ_{ij} 为 UCAV 速度方向与视线之间的夹角；λ_1 和 λ_2 为正常数。

速度威胁概率为

$$T_{ij,v}=\begin{cases}1 & \boldsymbol{v}_{T_j}\leqslant 0.5\boldsymbol{v}_i\\ 1.5-|\boldsymbol{v}_{T_j}|/|\boldsymbol{v}_i| & 0.5\boldsymbol{v}_i<\boldsymbol{v}_{T_j}\leqslant 1.4\boldsymbol{v}_i\\ 0.1 & \boldsymbol{v}_{T_j}>1.4\boldsymbol{v}_i\end{cases} \tag{5}$$

式中：$\boldsymbol{v}_i$ 和 $\boldsymbol{v}_{T_j}$ 分别为第 i 个 UCAV 和任务目标 T_j 的速度矢量。

1.2 任务威胁分析

战场目标威胁信息分析对 UCAV 执行任务成功率是非常重要的。对于目标威胁特性，将目标威胁分为探测类目标威胁和杀伤类武器威胁[9]。探测类目标威胁一般指雷达威胁，杀伤类武器威胁包括地空导弹和高炮，其威胁概率采用文献[7]的模型。

当 UCAV 在多目标威胁空间中执行任务时，面临的威胁程度增加。这种威胁程度增加的原因不仅仅是由于多个威胁源独立探测和杀伤的叠加效果，而且是由于整个目标网络化信息管理与共享。因此，提出了目标依赖矩阵来反映目标之间的关系[10]。

定义 1 $\boldsymbol{A}_{N\times N}$ 为目标内部之间依赖矩阵。矩阵元素 a_{ij} 为第 i 个目标对第 j 个目标的依赖程度，也可以说第 j 个目标对第 i 目标的信息共享或者火力支援能力。

对于多个目标的威胁，考虑到目标之间相互支援的情况，则目标对 UCAV 的威胁概率为

$$P_{\mathrm{T}ij}=\left[1-\prod(1-a_{ik,\mathrm{D}}P_{\mathrm{D}}(k,j))\right]\cdot\left[1-\prod(1-a_{im,\mathrm{G}}P_{\mathrm{G}}(m,j))\right] \tag{6}$$

式中：$P_{\mathrm{T}ij}$ 为目标 i 对无人作战飞机 j 的威胁概率；$a_{ik,\mathrm{D}}$ 为任务目标中探测类武器对其他武器的依赖程度；$a_{im,\mathrm{G}}$ 为任务目标中攻击类武器对其他武器的依赖程度；$P_{\mathrm{D}}(k,j)$ 为单个探测类武器的威胁概率；$P_{\mathrm{G}}(m,j)$ 为单个攻击类武器的威胁概率（包括地空导弹威胁概率与高炮威胁概率）。

对于攻击类目标，考虑到信息资源相互共享与火力支援，同时其威胁过程是即发现就攻击，探测概率为 1，因此，只计算攻击类目标对 UCAV 的威胁概率：

$$P_{\mathrm{T}ij}=1-\prod(1-a_{im,\mathrm{G}}P_{\mathrm{G}}(m,j)) \tag{7}$$

1.3 任务时间分析

UCAV 完成任务的时间主要是根据任务类型、可执行该任务的 UCAV 特性及载荷类型等先验经验得到完成各项任务的统计时间。观测某一时刻的分配情况、任务目标对 UCAV 的威胁、UCAV 对任务目标的杀伤情况以及完成任务的效果需要引入时间片这个概念。

定义 2 任务时间片[11]是指对事件变量集的某一次观察。在协同攻击多目标的任务分配问题中，时间片就是每隔一定的时间去观测 UCAV 的任务情况。时间片的间隔越小，意味着系统观

http://bhxb.buaa.edu.cn

测分配结果、目标威胁程度、UCAV 的杀伤情况以及对各方兵力信息的统计分析频率越高,那么决策越精细,其实时性提高;但是间隔小的时间片会引起信息量的增大,系统处理数据的时间增长,不利于战场快速决策,因此,要选取一定任务时间片是分配决策的关键问题。

定义 3　UCAV 飞行矩阵 $\boldsymbol{T}_{\mathrm{fly}}$ 是 $(M+N)\times N$ 矩阵,行包含 UCAV 和目标,列包含目标,t_{ij} 表示节点 i 到节点 j 的时间消耗。$\boldsymbol{T}_{\mathrm{fly}}$ 取决于 UCAV 的速度和位置、任务目标的位置以及其机动性,$\boldsymbol{T}_{\mathrm{fly}}$ 可表示为

$$\boldsymbol{T}_{\mathrm{fly}}=\begin{bmatrix} t_{11} & t_{12} & \cdots & t_{1N} & V_1\\ t_{21} & t_{22} & \cdots & t_{2N} & V_2\\ \vdots & \vdots & & \vdots & \vdots\\ t_{M1} & t_{M2} & \cdots & t_{MN} & V_M\\ t_{(M+1)1} & t_{(M+1)2} & \cdots & t_{(M+1)N} & T_1\\ t_{(M+2)1} & t_{(M+2)2} & \cdots & t_{(M+2)N} & T_2\\ \vdots & \vdots & & \vdots & \vdots\\ t_{(M+N)1} & t_{(M+N)2} & \cdots & t_{(M+N)N} & T_N\\ T_1 & T_2 & \cdots & T_N & \end{bmatrix}$$

对于 $\boldsymbol{T}_{\mathrm{fly}}$ 元素 t_{ij} 的计算主要与在时间片结束时刻 UCAV 的速度和位置、任务目标的位置有关,如下:

$$t_{ij}=\frac{|\boldsymbol{X}_i^{\mathrm{d}}(t)-\boldsymbol{X}_j^{\mathrm{d}}(t)|}{|\boldsymbol{v}_k|} \tag{8}$$

式中:$\boldsymbol{X}_i^{\mathrm{d}}(t)$ 和 $\boldsymbol{X}_j^{\mathrm{d}}(t)$ 分别为 t 时间片第 i 个节点和第 j 个节点的位置。

定义 4　任务累计完成时间 t_i[12] 是指第 i 个 UCAV 在执行任务集 $S_i=\{T_{i,1},T_{i,2},\cdots,T_{i,j}\}$ 的累计完成时间。t_i 可表示为

$$t_i=t_{iT_{i,1}}+t_{T_{i,1}T_{i,2}}+\cdots+t_{T_{i,j-1}T_{i,j}} \tag{9}$$

式中:$t_{iT_{i,1}}$、$t_{T_{i,1}T_{i,2}}$ 和 $t_{T_{i,j-1}T_{i,j}}$ 分别为第 i 个 UCAV 从所在位置节点到目标 $T_{i,1}$ 节点的时间,$T_{i,1}$ 到 $T_{i,2}$ 的时间和 $T_{i,j-1}$ 到 $T_{i,j}$ 的时间。

假设给某一架 UCAV 分配有 j 个目标,则分配方案有 A_j^j 个。通过计算这 A_j^j 个方案的执行任务累计时间,然后根据优先判断原则,选择最小的 t_i,这个 t_i 就是任务优先判断后的执行任务累计时间,对应的分配方案也就是这架 UCAV 的执行任务次序[13]。

1.4　协同任务分配评价指标

UCAVs 协同对地攻击任务目标是以整个编队的作战效能作为最优目标,而评价整个编队的作战效能指标有目标价值毁伤(Target Value Damage,TVD)、UCAV 编队损耗代价(UCAV attrition)和执行任务预计消耗时间(Task Expending Time,TET)。

1)目标价值毁伤最大化

无人机 V_i 完成任务 T_j 的目标毁伤收益值的大小体现了任务的重要程度和 UCAV 的执行能力,也可以量化目标价值毁伤值。编队对多目标的协同打击中,产生总的价值毁伤为

$$f_1=\sum_{i=1}^{m}V_{T_j}\left(1-\prod_{j=1}^{b}(1-P_{\mathrm{K}ij})^{X_{ij}}\right) \tag{10}$$

式中:f_1 为编队完成任务目标的价值毁伤;V_{T_j} 表示摧毁目标 T_j 获得的价值量,其由指挥人员预先设定,在作战过程中随着态势的变化和作战意图的变化可以进行调整。

2)UCAV 编队损耗代价最小化

如果在 UCAV 编队中无人机携带软杀伤武器,那么无人机会因为欺骗和干扰设备而增大无人机的生存概率。假设第 i 架无人机 V_i 携带 n_i 个欺骗和干扰设备,则该机的损耗代价为

$$f_{2i}=\frac{V_{V_i}\left[1-\prod_{i=1}^{c}(1-P_{\mathrm{T}ji})^{X_{ij}}\right]}{n_i+1} \tag{11}$$

式中:V_{V_i} 为无人机 V_i 的价值量。

在整个 UCAV 编队攻击过程中所遭受的损耗代价为

$$f_2=\sum_{j=1}^{n}\frac{V_{V_i}\left[1-\prod_{i=1}^{c}(1-P_{\mathrm{T}ji})^{X_{ij}}\right]}{n_i+1} \tag{12}$$

3)任务时间消耗最短

执行任务的时间可以用 UCAV 与任务目标间的距离来衡量,距离目标越短,则执行任务消耗的时间越短。任务消耗时间最短模型如下:

$$f_3=\sum_{i=1}^{m}t_i \tag{13}$$

式中:f_3 为任务时间消耗指标。

4)总评价指标

在多机协同过程中,设计总评价指标函数时,考虑到任务目标价值毁伤最大化、UCAV 编队损耗代价最小化以及任务时间消耗最短,因此,分配模型的总体评价指标函数为

$$J(\Pi)=\min\frac{\omega_1 f_2+\omega_2 f_3}{f_1} \tag{14}$$

式中:$J(\Pi)$ 为 UCAV 编队协同攻击多目标的总体评价指标函数,Π 为分配方案;ω_1 和 ω_2 分别为衡量减少 UCAV 代价损耗和时间损耗的权重,其来

http://bhxb.buaa.edu.cn

源于决策者的偏好。协同攻击多目标分配问题实质就是寻找一个方案 Π 使评价指标函数最小,使问题转化为寻优问题。

约束条件:

1) $N \leqslant \sum_{i=1}^{M} L_i$,UCAV 任务目标的总数量不能超过编队的导弹数量。

2) $X_{1j} + X_{2j} + \cdots + X_{Lj} = 1$,不允许多个 UCAV 导弹攻击同一个任务目标。

3) $c_i \leqslant L_i$,单架 UCAV 执行任务数量 c_i 不超过 UCAV 可携带最大导弹数量 L_i。

4) 由于 UCAV 性能和挂架的限制,UCAV 和其挂载的武器或者导弹之间存在约束,UCAV 影响武器效能的发挥可以用 UCAV-武器适应度表进行描述,该适应度表中的元素用[0,1]之间的数表示,数值的大小表示武器攻击效能发挥程度,数值越小表明 UCAV 越不利于武器的发挥。

2 自适应灰狼优化算法

2.1 基本 GWO 算法及改进策略

受灰狼的社会等级制度和猎食行为的启发,Seyedali 等提出了 GWO 算法[14]。同其他启发式优化算法相比,GWO 算法具有实现简单、全局优化能力强、收敛速度快等特点,一经提出便受到了众多学者的关注与研究[15-17]。

在寻找最优解过程中,根据灰狼猎食的 3 个重要步骤,即接近、包围、攻击猎物,GWO 算法建立了一个数学模型:狼群中每一个个体都是一个潜在解,其中 α 狼的位置是最好的解,β 和 δ 的位置分别为优解和次优解,其他的候选解是 ω 的位置。猎食过程如图 1 所示,C_1、C_2 和 C_3 分别为 α、β 和 δ 狼的位置系数;a_1、a_2、a_3 分别为 α、β 和 δ 狼的控制参数;D_α、D_β 和 D_δ 分别为 α、β 和 δ 狼到 ω 狼的距离。在文献[14]中,通过大量的仿真实验表明,GWO 在函数优化问题时具有较快的收敛速度,但是控制参数 a 线性变化忽略了要解决的优化问题多样性,尤其在解决多峰值函数问题时容易陷入局部最优,很难达到全局最优。

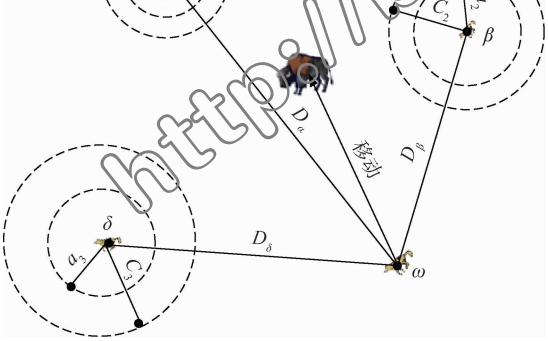


图 1 GWO 算法猎食过程

Fig. 1 Hunting process of GWO algorithm

针对以上缺点与不足,本文提出了自适应调整策略,通过学习种群最优位置信息去控制种群的搜索方向;另一方面,改进算法利用最优个体与当前个体之间的位置矢量差,跳出局部最优区域来保持种群多样性。对于控制参数 a(包括 a_1、a_2 和 a_3)来说,根据文献[18-20]的分析,为了确保算法在多种条件下具有好的优化结果,a 采用二次曲线策略。

1) 自适应调整策略

为了进一步提高算法的收敛速度,采用自适应调整策略,将当前个体的适应度值 J_i 与灰狼群的平均适应度值 J_{avg} 进行比较,如果 J_i 优于 J_{avg},继续使用原策略更新灰狼位置;如果 J_i 次于 J_{avg},利用式(15)变异灰狼位置[7]。

$$\boldsymbol{X}(g+1) = \begin{cases} \dfrac{(1/J_\alpha)\boldsymbol{X}_1 + (1/J_\beta)\boldsymbol{X}_2 + (1/J_\delta)\boldsymbol{X}_3}{a_J} & J_i \leqslant J_{\text{avg}} \\ \dfrac{\boldsymbol{X}_1 + \boldsymbol{X}_2 + \boldsymbol{X}_3}{3} & J_i > J_{\text{avg}} \end{cases} \tag{15}$$

式中:$\boldsymbol{X}(g+1)$ 为第 g 代位置更新后的灰狼位置,即第 $g+1$ 代灰狼位置,g 为当前迭代次数;$\boldsymbol{X}_1$、$\boldsymbol{X}_2$ 和 $\boldsymbol{X}_3$ 分别为当前灰狼到 α、β 和 δ 灰狼的距离;a_J 为 $1/J_\alpha$、$1/J_\beta$ 与 $1/J_\delta$ 之和,J_α、J_β 和 J_δ 分别为 α、β 和 δ 的适应度值。

2) 跳出局部最优策略

针对原种群多样性较差,该策略对更新后的位置进行重置,通过最优个体与当前个体之间的位置矢量差随机控制当前个体不在当前局部最优解的区域内[21]:

$$\boldsymbol{X}_i'(g+1) = \boldsymbol{X}_\alpha(g) + |\boldsymbol{X}_\alpha(g) - \boldsymbol{X}_i(g)|\boldsymbol{r} \tag{16}$$

式中:$\boldsymbol{X}_i'(g+1)$ 和 $\boldsymbol{X}_i(g)$ 分别为跳出局部最优的智能个体位置;$\boldsymbol{X}_\alpha(g)$ 为当代最优解;$\boldsymbol{r}$ 为[-2,-1]和[1,2]的随机向量。

3) 控制参数非线性调整策略

对于基本的 GWO 算法,选择二次曲线来设计控制参数 a 的调整策略。利用二次曲线在区间[0,1]上非线性递增,将 GWO 算法的迭代次数映射到[0,1]上,控制参数调整策略为

$$a = a_{\max} - (a_{\max} - a_{\min})(g/g_{\max})^2 \tag{17}$$

式中:$a_{\max}$ 和 $a_{\min}$ 分别为控制参数的最大值和最小值,一般分别取 2 和 0;$g_{\max}$ 为最大迭代次数。

http://bhxb.buaa.edu.cn

2.2 SAGWO 算法流程

由于 GWO 存在收敛速度慢,容易陷入局部最优的缺点,结合上面策略,提出了 SAGWO 算法。下面给出了 SAGWO 算法的流程:

步骤 1　初始化灰狼种群,即随机产生 n 个智能个体的位置;初始化 a、距离系数 $\boldsymbol{A}$ 和位置系数 $\boldsymbol{C}$[14];初始化 $\boldsymbol{X}_\alpha$、$\boldsymbol{X}_\beta$ 和 $\boldsymbol{X}_\delta$ 的值。

步骤 2　处理越界智能个体,计算每个智能体的适应度值,计算灰狼群的平均适应度值 J_{avg}。

步骤 3　更新当前迭代期间的 $\boldsymbol{X}_\alpha$、$\boldsymbol{X}_\beta$ 和 $\boldsymbol{X}_\delta$。

步骤 4　根据式(17),计算各个优解的 a、$\boldsymbol{A}$ 和 $\boldsymbol{C}$。

步骤 5　采用式(15)的策略更新当前智能个体的位置。

步骤 6　通过式(16),使智能个体跳出局部最优解的区域。

步骤 7　输出最优解 $\boldsymbol{X}_\alpha$ 和最优值 J_α。

步骤 8　如果达到结束条件(最大迭代次数或连续 5 次结果相差小于一定的精度),则结束;否则转到步骤 2。

2.3 SAGWO 算法的收敛性分析

SAGWO 算法也是一种随机算法,可以通过随机算法的收敛准则来判断该算法是否收敛。文献[22]给出了随机优化算法求解最小问题的收敛准则。假设对于优化问题 $<S,f>$,S 和 f 分别为解空间和优化问题目标函数。算法 F_{al} 满足条件 1 和条件 2,则算法 F_{al} 产生的序列 $\{x_k\}_{k=0}^{\infty}$ 有:$\lim\limits_{k\to\infty}P(x_k\in R_{\varepsilon,M})=1$,也就是算法 F_{al} 收敛,其中,$P(x_k\in R_{\varepsilon,M})$ 是算法 F_{al} 第 k 次的搜索结果 x_k 在 $R_{\varepsilon,M}$ 上的概率测度,x_k 为第 k 次迭代的结果,条件 1 与条件 2 分别为:

条件 1　若 $f(F_{\text{al}}(x,\xi))\leqslant f(x)$,且 $\xi\in S$,则有 $f(F_{\text{al}}(x,\xi))\leqslant f(\xi)$。

条件 2　$\forall B\in S$, s. t. $v(B)>0$, $\prod\limits_{k=0}^{\infty}(1-u_k(B))=0$。

其中:$x_{k+1}=F_{\text{al}}(x_k,\xi)$ 为下一次迭代结果;S 为解空间;f 为优化函数;ξ 为迭代过程中得到的解;$v(B)$ 为集合 B 的 Lebesgue 测度;$u_k(B)$ 为算法 F_{al} 第 k 次迭代解在集合 B 上的概率测度。

定理 1　SAGWO 算法收敛于全局最优。

证明　首先,SAGWO 算法每次迭代都保存了灰狼群体的最优位置,保证了目标函数的适应度值的非增性,所以算法的收敛满足条件 1。

其次,再需要证明 SAGWO 算法满足条件 2,便可根据文献[23]的随机优化算法收敛定理,得到 SAGWO 算法收敛于全局最优。根据文献[14]的灰狼群体位置更新公式转换,可得

$$\frac{\|\boldsymbol{X}_\alpha(g)-\boldsymbol{X}_1(g)\|}{\|\boldsymbol{X}_\alpha(g)-2r_2\boldsymbol{X}(g)\|}=|\boldsymbol{A}(g)|=a(g)\,|2r_1-1| \tag{18}$$

式中:$\boldsymbol{X}_1(g)$ 为更新 $\boldsymbol{X}(g)$ 而产生与最优解有关的中间变量;r_1 与 r_2 为随机变量。

由式(18)对每一代左右累乘,可得

$$\prod_{g=1}^{\infty}\frac{\|\boldsymbol{X}_\alpha(g)-\boldsymbol{X}_1(g)\|}{\|\boldsymbol{X}_\alpha(g)-2r_2\boldsymbol{X}(g)\|}=\prod_{g=1}^{\infty}a(g)\,|2r_1-1|\leqslant\prod_{g=1}^{\infty}a(g) \tag{19}$$

对于式(19)等号后的部分,根据式(17)可知,$\lim\limits_{t\to\infty}a(g)=0$,则有

$$\lim_{t\to\infty}\prod_{t=1}^{\infty}a(g)=0 \tag{20}$$

根据式(19)与式(20),可得

$$\lim_{g\to\infty}\prod_{g=1}^{\infty}\frac{\|\boldsymbol{X}_\alpha(g)-\boldsymbol{X}_1(g)\|}{\|\boldsymbol{X}_\alpha(g)-2r_2\boldsymbol{X}(g)\|}=0 \tag{21}$$

因此必有

$$\lim_{g\to\infty}\|\boldsymbol{X}_\alpha(g)-\boldsymbol{X}_1(g)\|=0 \tag{22}$$

同理可得,优解、次优解、$\boldsymbol{X}_2$ 与 $\boldsymbol{X}_3$ 收敛于最优解 $\boldsymbol{X}_\alpha$。由式(15)可知,当 $g\to\infty$ 时,$\boldsymbol{X}(g+1)$ 收敛于最优解 $\boldsymbol{X}_\alpha$。

SAGWO 算法的单个灰狼智能体搜素空间是有限的,则由 n 个灰狼组成的群体的状态也是有限的。当前的群体状态仅与上一时刻的群体状态有关,所以,狼群状态具有 Markov 性。狼群的转移状态与时间 g 无关,则狼群位置的状态序列是齐次的。综上述分析,狼群位置的状态序列是有限齐次 Markov 链。

设优化问题 $<S,f>$ 的全局最优解为 $\boldsymbol{X}_\alpha$,定义狼群位置的最优状态集为

$$H=\{\boldsymbol{q}=(X_{\alpha 1},X_{\alpha 2},\cdots,X_{\alpha n})\} \tag{23}$$

对于狼群状态序列 $\{\boldsymbol{X}_\alpha(t);g\geqslant 0\}$,最优状态集 H 是群体状态空间 S 上的一个闭集。由式(22)和狼群状态序列是有限齐次 Markov 链可知,S 中不存在 H 之外的非闭空间,即群体状态序列必将进入最优状态集 H,连续无穷次搜索不到全局最优解的可能性是 0,则满足 2.3 节条件 2,故 SAGWO 算法收敛于全局最优。　**证毕**

2.4 SAGWO 算法的复杂度分析

根据 SAGWO 算法的特点与执行步骤,在这里讨论算法每一步时间复杂度。在 2.2 节的步骤 1中,对 D 维空间下的 n 个灰狼进行初始化,需要 nD 次运算;在步骤 2 中,计算狼群的适应度值

http://bhxb.buaa.edu.cn

需要 n 次运算,一般的适应度函数的复杂度为 $O(D)$,还要计算种群适应度值平均值,需要 n 次运算;在步骤 3 中,通过比较得到 3 个优解,最多需要 $3n-3$ 次运算;在步骤 4 中,计算更新各个优解的 a、$\boldsymbol{A}$ 和 $\boldsymbol{C}$ 需要 $3nD$;在步骤 5 中,计算各个狼到 $\boldsymbol{X}_\alpha$、$\boldsymbol{X}_\beta$ 和 $\boldsymbol{X}_\delta$ 的距离需要 $3nD$ 次运算,根据式(15)需要 $D+n$,包括计算更新位置与比较;在步骤 6 中,跳出局部最优策略需要 nD;在步骤 7 和 8 中,输出最优解与判断算法是否满足终止条件,其复杂度均为常数。由于算法最多执行 g 次,则 SAGWO 算法的时间复杂度为 $O(((3n+1)\cdot D+n)g)\approx O(3nDg)$。

3 UCAVs 动态任务分配方法

3.1 目标任务序列编码

在 GWO 算法中,每个智能个体就是一个备选解,多个智能个体通过种群搜索和其位置更新来寻优。协同攻击多目标的任务决策关键在于确定:哪架 UCAV 来执行哪些任务和 UCAV 执行任务的时序[23]。

对于 UCAV 导弹与执行目标之间的关系,第 1 个 L 维向量的数值代表该 UCAV 用空地导弹 $k=\sum_{i=1}^{i-1}L_i+l$ 攻击的目标的序号;第 2 个 L 维向量的数值表示 UCAV 执行该任务在该 UCAV 任务集中的次序,可以通过任务累计时间去判别任务的优先执行次序。例如,如图 2 所示,第 2 架 UCAV 要执行的任务是目标 1 和目标 5,其中导弹 3 执行任务目标 1,导弹 4 执行任务目标 5,而根据任务累计时间的分析,UCAV 首先执行目标 5,然后执行目标 1。

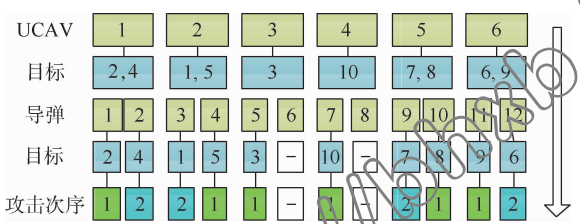


图 2 目标任务序列编码

Fig. 2 Target task sequence coding

3.2 多目标动态任务分配流程

UCAV 在执行任务过程中,所有的元素都可能随着时间发生变化。设定动态的任务分配的适应度函数为

$$J(\Pi(t),t)=\min\frac{\omega_1 f_2(X_{\text{missile-T}}(t),t)+\omega_2 f_3(X_{\text{order}}(t),t)}{f_1(X_{\text{missile-T}}(t),t)} \tag{24}$$

其中:$\boldsymbol{X}_{\text{missile-T}}(t)$ 为 UCAV 携带的导弹对任务目标的分配方案;$X_{\text{order}}(t)$ 为 UCAV 执行任务的顺序方案;t 表示执行任务中的时刻。

具体基于 SAGWO 算法的 UCAV 协同攻击多目标动态任务分配流程如下:

步骤 1 定义任务时间片长度 t_{slice}。根据估算最小的任务飞行时间 $\min(t_{ij})$,将其划分为 n_t 个时间片,因此,可以计算得到

$$t_{\text{slice}}=\frac{\min(t_{ij})}{n_t} \tag{25}$$

步骤 2 初始化 UCAV 与目标的信息。在分配开始之前,获取初始 UCAV 的位置、数量、速度以及其武器信息,探测得到目标的位置、数量和机动性。计算目标的威胁概率和 UCAV 对目标的毁伤概率。

步骤 3 初始化参数。设置 GWO 的基本参数,控制参数最大值 $a_{\max}$ 和最小值 $a_{\min}$,GWO 参数 $\boldsymbol{A}$、$\boldsymbol{C}$,搜索方程的惯性权重 ω,最大迭代次数 $g_{\max}$,搜索空间智能个体数为 n_{agent}。

步骤 4 随机生成一组实数然后基于编码生成初始分配方案。

步骤 5 当 $t=0$ 时,根据 2.2 节的步骤 2 ~ 步骤 8,计算初始最优分配方案。

步骤 6 UCAV 根据初始分配方案,对将要执行的任务目标进行动态追踪。

步骤 7 在第一个时间片开始时,获取 UCAV 失效信息与目标是否发生变化。

步骤 8 在时间片内,根据变化的信息,随机生成一组分配方案,重复步骤 5,寻求信息变化后的最优分配方案。

步骤 9 在时间片末,根据步骤 8 的最优方案,利用追踪法更新 UCAV 的位置信息,并且输出最优分配结果。

步骤 10 在下一个时间片或者在 $t+t_{\text{slice}}$ 时刻,重复步骤 7 ~ 步骤 9,直到 $t\geqslant\min(t_{ij})$ 或者开始执行航路规划层的时候,输出最终结果。

4 仿真实验及分析

4.1 任务想定

为验证算法的有效性,针对 6 架 UCAV 对 8 个具有威胁的敌方目标的任务分配进行仿真实验,如图 3 所示,x 和 y 分别为横向和纵向坐标。其中,UCAV 编队拥有 6 架无人机,每架无人机携带 2 枚空地导弹,其攻击目标的任务是毁伤目标,相关属性见表 1 和表 2;敌方由雷达(Radar)、SAM(Surface-to-Air Missile)防空导弹和 AAGun

http://bhxb.buaa.edu.cn

(Anti-Aircraft Gun)防空炮 3 种类型目标组成,每个目标最多只能被我方 1 个 UCAV 攻击,其相关属性见表 3 和表 4。A 型空对地导弹的性能参数:有效作用距离 $r_{W\text{-}A}$ = 10 km、最大跟踪距离 $R_{Wmax\text{-}A}$ = 60 km;B 型空对地导弹的性能参数:$r_{W\text{-}B}$ = 10 km、$R_{Wmax\text{-}B}$ = 80 km。表 2 表示 UCAV 与机载导弹之间约束。

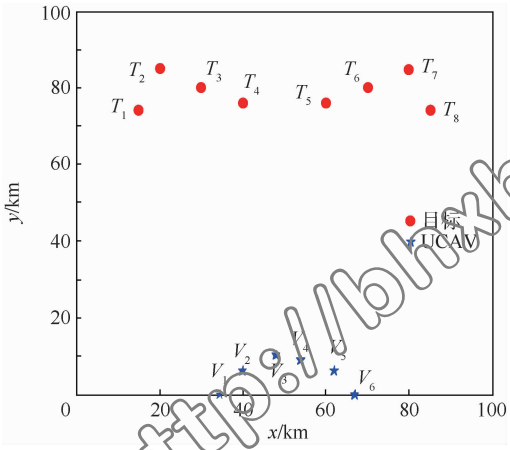


图 3　仿真环境初始设置

Fig. 3　Initial setting of simulation environment

表 1　UCAV 编队信息设置

Table 1　Information setting of UCAV formation

UCAV	位置/km	导弹数量×型号	软杀伤武器数量	速度/(m·s^{-1})	价值量
V_1	(35,0)	2×A-AGM	2	238	0.8
V_2	(40,6)	2×B-AGM	1	238	0.9
V_3	(48,10)	2×A-AGM	2	238	0.95
V_4	(54,9)	2×B-AGM	1	238	0.95
V_5	(62,6)	2×A-AGM	2	238	0.85
V_6	(67,0)	2×B-AGM	0	238	0.8

表 2　UCAV 的武器适应度

Table 2　Weapon fitness of UCAV

导弹编号	V_1	V_2	V_3	V_4	V_5	V_6
1	0.98	0.95	0.99	0.99	0.99	0.95
2	1	1	0.96	0.97	1	1

表 3　目标信息设置

Table 3　Information setting of targets

目　标	类　型	位置/km	价值量
T_1	SAM	(15,74)	0.75
T_2	Radar	(20,85)	0.7
T_3	SAM	(30,80)	0.75
T_4	AAGun	(40,76)	0.5
T_5	AAGun	(60,76)	0.5
T_6	AAGun	(70,80)	0.5
T_7	SAM	(80,85)	0.75
T_8	Radar	(85,74)	0.7

表 4　目标间依赖矩阵

Table 4　Dependence matrix of targets

目　标	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
T_1	1	0.2	0.9	0.95	0.95	0.95	0.9	0.1
T_2	0.2	1	0.2	0.1	0.1	0.1	0.1	0.5
T_3	0.9	0.8	1	0.95	0.95	0.95	0.9	0.8
T_4	0.1	0.1	0.3	1	0.1	0.1	0.1	0.1
T_5	0.1	0.1	0.3	0.1	1	0.1	0.1	0.1
T_6	0.1	0.1	0.3	0.1	0.1	1	0.1	0.1
T_7	0.9	0.8	0.95	0.95	0.95	0.9	1	0.8
T_8	0.2	1	0.2	0.1	0.1	0.1	0.1	1

表 3 表示敌方目标的编号、类型、位置信息以及价值量,由文献[7,9]可知,雷达目标性能参数:雷达参数 $K_R = 1.0 \times 10^{12}$ m^4、最大作用距离 R_{max} = 100 km;防空导弹 SAM 性能参数:杀伤区近界水平距离 d_{sj} = 30 km,杀伤区远界水平距离 d_{sy} = 100 km;高炮 AAGun 性能参数:有效火力半径 R_g = 10 km,圆锥死界半径 r_g = 0.9 km。表 4 表示目标间相互支援能力。

SAGWO 算法和 GWO 算法种群数大小设置为 30,迭代次数为 100 次,控制参数值的最大值 a_{max}、最小值 a_{min} 分别取值为 2 和 0。对于 IPSO、ACA、GA、GWO 算法,其算法参数设置与文献[7]相同,便于比较参考。

任务分配模型中的威胁优势权重 c_1 和 c_2 分别为 0.6 和 0.4,总体评价指标函数中的衡量减少 UCAV 代价损耗和时间损耗的权重 ω_1 和 ω_2 分别为 0.6 和 0.4。

本文采用 MATLAB 2014a 进行仿真,运行环境为 Intel(R) Core(TM) i5-3470 处理器,操作系统为 Windows7。

4.2　静态任务分配仿真

为了验证 SAGWO 算法的求解有效性,对该实验运行 30 次,得到最优分配方案、算法耗时和平均最优函数值。根据优势概率模型,计算得到我方无人机编队各个 UCAV 对敌方目标的优势概率,如表 5 所示;根据目标对 UCAV 的威胁模型,求出各目标对 UCAV 的威胁概率,如表 6 所示。

仿真结果:SAGWO 算法耗时为 0.281 06 s,最优函数值为 0.733 8,最优任务分配结果如表 7 所示,具体分配如图 4 所示。

由图 5 可知,最优适应度值随着迭代次数的增加,渐趋于稳定,在第 42 代以后最优适应度值基本不再变化,其值为 0.733 8;平均函数值也随着迭代次数增加而趋于稳定,虽然有微小波动,但是总体趋势不再变化。由图6可知,任务分配的

http://bhxb.buaa.edu.cn

表 5 UCAV 对目标的优势概率

Table 5 Advantage probability of UCAV to target

UCAV	导弹编号	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
V_1	1	0.398	0.380	0.399	0.398	0.387	0.398	0.390	0.388
	2	0.400	0.406	0.421	0.400	0.408	0.400	0.391	0.403
V_2	3	0.459	0.395	0.441	0.487	0.457	0.396	0.395	0.389
	4	0.465	0.420	0.446	0.485	0.461	0.401	0.400	0.400
V_3	5	0.399	0.396	0.411	0.399	0.401	0.386	0.395	0.399
	6	0.396	0.394	0.396	0.369	0.390	0.390	0.396	0.396
V_4	7	0.435	0.398	0.442	0.498	0.508	0.461	0.399	0.467
	8	0.434	0.397	0.440	0.496	0.506	0.458	0.397	0.465
V_5	9	0.399	0.400	0.394	0.399	0.407	0.399	0.402	0.399
	10	0.403	0.400	0.404	0.405	0.400	0.411	0.400	0.400
V_6	11	0.395	0.395	0.387	0.395	0.426	0.396	0.395	0.430
	12	0.410	0.405	0.400	0.399	0.431	0.404	0.424	0.432

表 6 目标对 UCAV 的威胁概率

Table 6 Threat probability of target to UCAV

目 标	V_1	V_2	V_3	V_4	V_5	V_6
T_1	0.528	0.651	0.705	0.668	0.573	0.400
T_2	0.124	0.159	0.175	0.160	0.126	0.079
T_3	0.523	0.649	0.706	0.669	0.575	0.421
T_4	0.121	0.158	0.176	0.163	0.130	0.083
T_5	0.120	0.157	0.201	0.162	0.130	0.082
T_6	0.122	0.157	0.217	0.154	0.130	0.101
T_7	0.514	0.638	0.695	0.660	0.567	0.397
T_8	0.124	0.159	0.175	0.161	0.127	0.080

表 7 UCAVs 最优任务分配

Table 7 Best task allocation of UCAVs

目 标	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
UCAV	V_2	V_1	V_2	V_3	V_4	V_6	V_5	V_4

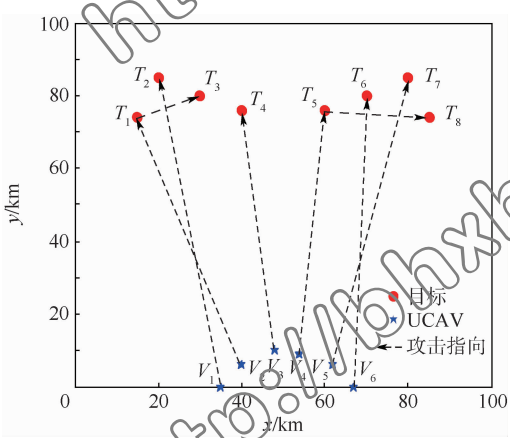


图 4 UCAV 编队攻击目标的任务分配

Fig. 4 Attack target task allocation of UCAV formation

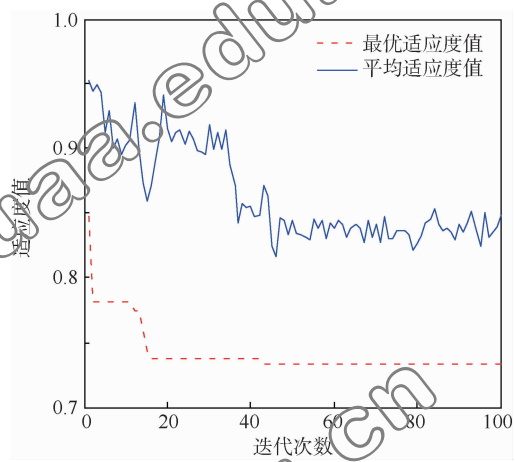


图 5 SAGWO 算法的适应度值变化曲线

Fig. 5 Fitness variation curves of SAGWO algorithm

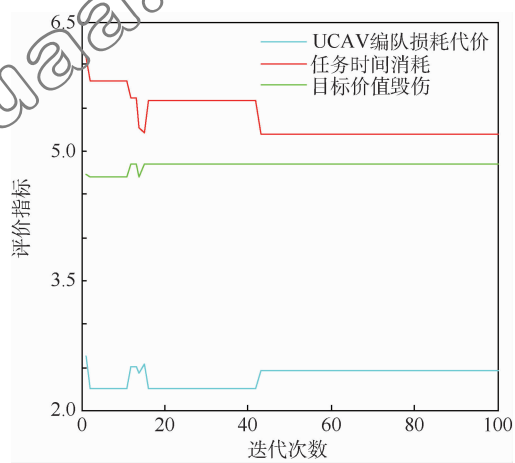


图 6 SAGWO 算法各评价指标曲线

Fig. 6 Evaluation index curves of SAGWO algorithm

各个指标值在最开始有波动,但是随着迭代次数增加,各指标值区趋于稳定,编队损耗代价值随着迭代增加而减小,目标价值毁伤值相对原来变化不大,任务时间消耗值比迭代初期减少,基本符合任务分配的目标毁伤最大化、编队损耗最小化、任务时间消耗最小化的原则。综合以上分析,SAGWO 算法对任务分配问题是有效的。

为了验证改进算法的求解质量与求解稳定性,将 SAGWO 算法与 IPSO、ACO、GA、GWO 算法

http://bhxb.buaa.edu.cn

相比较[7],并且应用于不同规模。从图 7 可以分析出,在分配方案变化不大的基础上,5 种算法的最优函数值随着迭代次数的增加渐渐趋于稳定,只是求解精度与求解速度不一样,GWO、ACO 和 IPSO 算法相对于其他 2 种算法过早的收敛,陷入局部最优;GA 算法在 58 代时达到最优解,但是收敛速度较慢,SAGWO 算法在 43 代达到最优解,相较于 GA 算法收敛速度较快。图 8 是针对不同规模的任务分配问题,采用不同算法的优化时间对比结果,从其可以看出,基于各个算法的任务分配优化时间随着任务分配规模增加而增加,其中 GA、ACO、IPSO 算法对问题规模增加更加敏感,优化时间增加得更快,而 SAGWO 算法与 GWO 算法的优化时间增加程度基本上一样。通过以上基于不同算法的任务分配实验,进一步验证了 SAGWO 算法的求解稳定性和有效性,也验证了 SAGWO 算法的求解精度好于其他 4 种算法,而且优化时间也好于 GA、ACO、IPSO 算法。

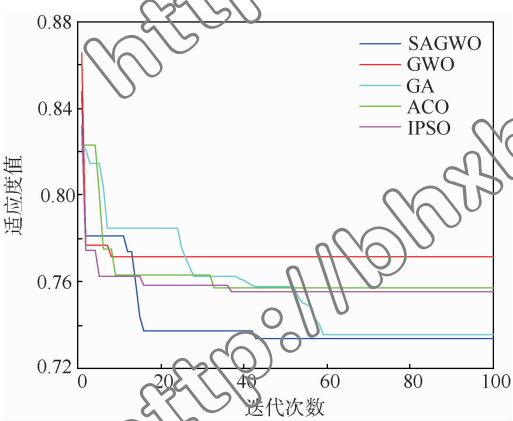


图 7　基于不同算法的 6 架 UCAV 编队攻击 8 目标适应度值变化曲线

Fig. 7　Fitness variation curves of 6-UCAV formation attacking 8 targets based on different algorithms

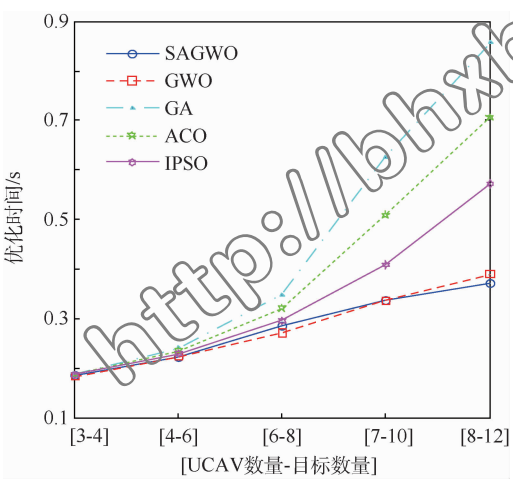


图 8　不同规模任务分配优化时间对比

Fig. 8　Comparison of optimization time for different scales of task allocation

4.3　动态任务分配仿真

1）基于目标移动的动态任务分配仿真

为了验证基于目标移动的动态任务分配有效性,对 6 架 UCAV 编队攻击 8 目标的动态任务分配实验运行 30 次,得到一定时间片末的最优分配方案、算法耗时和平均最优函数值。首先通过 UCAV 飞行时间矩阵定义得到表 8,根据时间片定义与分析,确定最小飞行时间 $\min(t_{ij})$ 和时间片数 n 分别为 279.3 和 270,得到时间片长度为 1 s(取整数,方便计算),满足大于每次任务分配优化时间。其次设置机动目标参数,如表 9 所示,合速度为 x、y 方向速度平方和的开方。在实验中,取定前 240 个时间片进行仿真与分析,首先对整个过程的动态任务分配进行分析,选取 240 个时间片其中的 4 个时间片与 0 时刻进行对比分析,然后对 240 个时间片的函数值曲线进行分析,最后对第 240 个时间片优化函数值曲线和指标值变化曲线进行分析。

图 9 和表 10 是针对基于目标移动的动态任务分配问题的任务分配图和其中 5 个时间片的任务分配方案。从图 9 可以看出,经过 240 个时间片,UCAV 根据自己的优势去攻击相应的目标,分配结果比较均匀。表 10 中的 5 个时间片任务分配方案随着时间片的变化,其有一定的变动,这是因为目标位置发生变化,UCAV 也发生变化,导致分配方案也发生变化。

表 8　UCAV 飞行时间矩阵

Table 8　Fly time matrix of UCAV

UCAV	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
V_1	322.1	362.7	336.8	320.0	336.2	366.9	404.1	375.3
V_2	304.4	342.4	313.7	294.1	305.9	335.5	372.1	342.6
V_3	302.5	336.4	303.7	279.3	281.8	308.3	342.6	310.6
V_4	318.5	349.8	314.9	287.6	282.6	305.8	337.5	302.6
V_5	347.3	375.9	338.8	308.3	294.2	312.7	340.4	301.6
V_6	380.1	408.1	370.3	338.9	320.7	336.4	361.3	320.0

表 9　机动目标速度

Table 9　Speed of maneuvering targets

目　标	合速度/($m \cdot s^{-1}$)	x 方向速度/($m \cdot s^{-1}$)	y 方向速度/($m \cdot s^{-1}$)
T_1	25	$25\sin(\pi t/180)$	$25\cos(\pi t/180)$
T_2	10	$10\sin(\pi/4)$	$-10\cos(\pi/4)$
T_3	25	$-25\sin(\pi t/180)$	$-25\cos(\pi t/180)$
T_4	7	-7	0
T_5	7	$7\sin(\pi/3)$	$7\cos(\pi/3)$
T_6	7	7	0
T_7	25	$-25\sin(\pi/6)$	$-25\cos(\pi/6)$
T_8	10	-10	0

http://bhxb.buaa.edu.cn

表 11 是针对基于目标移动的动态任务分配问题选取其中 5 个时间片的优化时间和适应度值进行比较分析，图 10 表示 240 个时间片的平均函数值与最优函数值随时间片变化的曲线。通过以上结果分析，随着时间变化，基于移动目标的任务分配方法是有效的，而且任务分配方案也是变化的。图 11 为第 240 个时间片的最优适应度值、平均值以及各评价指标值随迭代次数增加的变化趋势。通过图 11 的结果分析，时间片内基于 SAGWO 算法的任务分配是有效的，而且趋于稳定，得到最优值。

2）基于 UCAV 失效的动态任务分配仿真

在第240个时间片，V_4被击伤，退出战场，通过仿真实验，得到表 12、表 13 和图 12。表 12 中“0”表示无目标可执行。从表 13 的仿真结果可以知道，随着 UCAV 减少一架，优化时间消耗也减少。从图 12 中可以看出，随着迭代次数的增加，最优函数值和平均函数值趋于稳定，任务分配的各个指标值也渐趋于稳定，而且其值走向符合编队损耗代价值最小化、任务时间消耗最小化和目标价值毁伤最小化的原则。

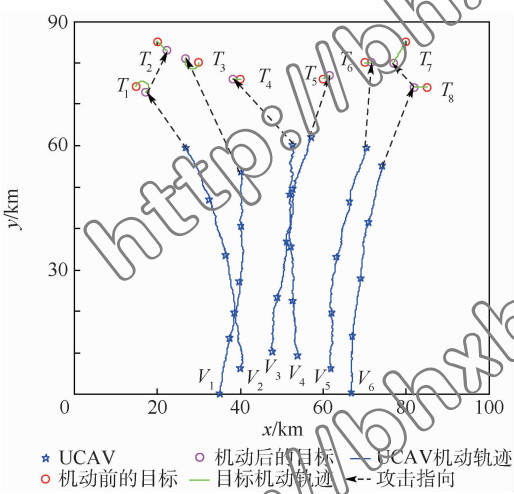


图 9 基于目标移动的动态任务分配

Fig. 9 Dynamic task allocation based on target movement

表 10 5 个时间片的任务分配方案比较

Table 10 Comparison of task allocation scheme for 5 time slices

时间/s	V_1	V_2	V_3	V_4	V_5	V_6
0	T_1	T_3，T_2	T_4	T_5，T_8	T_7	T_6
60	T_2	T_1，T_3	T_4	T_5	T_6，T_7	T_8
120	T_3	T_1，T_2	T_4	T_5	T_6，T_7	T_8
180	T_3，T_2	T_1	T_5	T_4	T_6	T_8，T_7
240	T_3	T_1，T_2	T_5	T_4	T_6	T_8，T_7

表 11 5 个时间片的任务分配优化时间消耗与适应度值

Table 11 Task allocation optimization time and fitness for 5 time slices

时间/s	优化时间/s	最优适应度值	平均适应度值
0	0.286 8	0.771 5	0.868 2
60	0.269 3	0.817 6	0.971 2
120	0.263 8	0.702 6	0.907 6
180	0.263 5	0.546 4	0.732 2
240	0.262 5	0.323 5	0.498 0

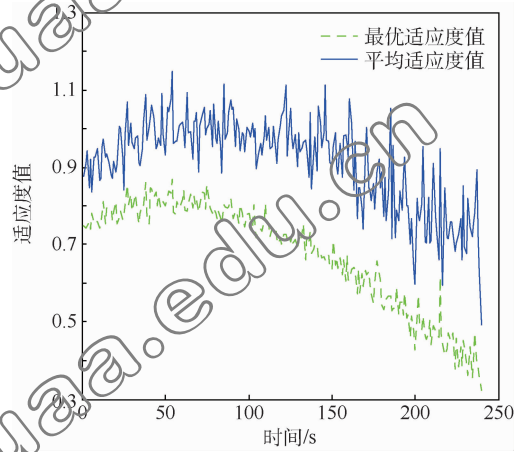


图 10 240 个时间片的适应度值变化曲线

Fig. 10 Fitness curve for 240 time slices

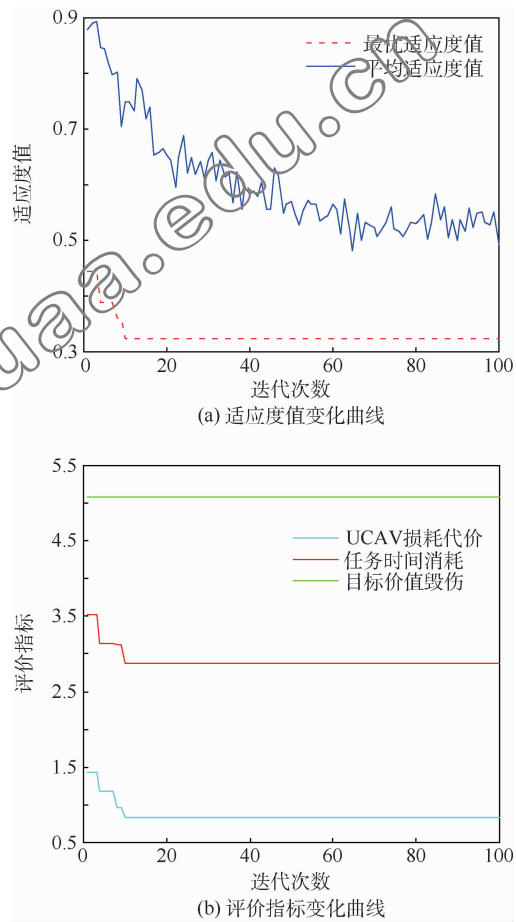


图 11 第 240 个时间片的优化结果

Fig. 11 Optimized results fitness in the 240th time slice

http://bhxb.buaa.edu.cn

表 12　一架 UCAV 失效前后的任务分配方案比较

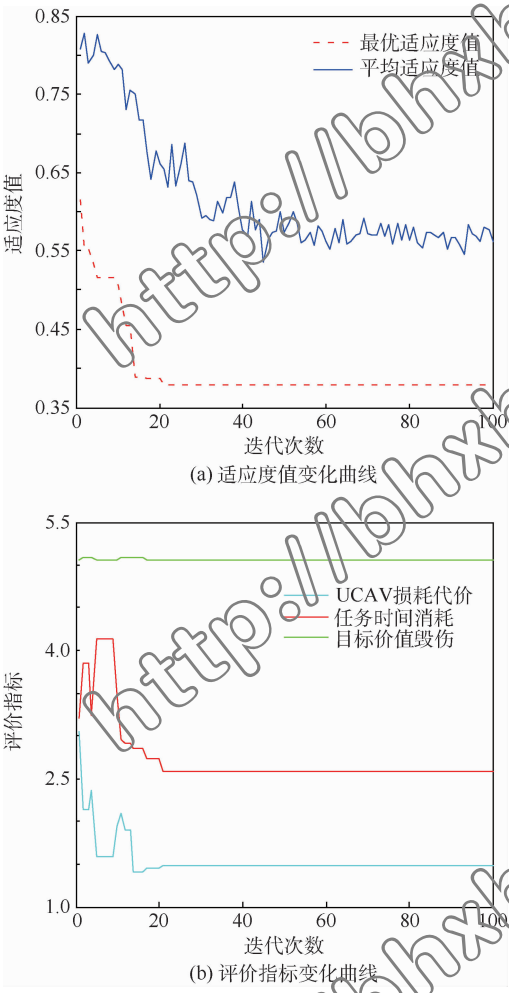
Table 12　Comparison of task allocation scheme before and after invalidation of an UCAV

UCAV	V_1		V_2		V_3		V_4		V_5		V_6	
	失效前	失效后	失效前	失效后	失效前	失效后	失效前	失效后	失效前	失效后	失效前	失效后
目　标	T_3	T_4, T_3	T_1, T_2	T_1, T_2	T_5	T_5	T_4	0	T_6	T_6	T_8, T_7	T_8, T_7

表 13　一架 UCAV 失效前后的任务分配优化结果对比

Table 13　Comparison of task allocation optimized result before and after invalidation of an UCAV

优化指标	优化时间/s		最优适应度值		平均适应度值	
	失效前	失效后	失效前	失效后	失效前	失效后
数　值	0.262 5	0.249 3	0.323 5	0.376 4	0.498 0	0.542 0



(a) 适应度值变化曲线

(b) 评价指标变化曲线

图 12　一架 UCAV 失效后一个时间片内的优化结果

Fig. 12　Optimized result in a time slice after invalidation of an UCAV

3）基于新增 2 个目标的动态任务分配仿真

对新增 2 个目标的性能参数如表 14 和表 15 所示。在第 240 个时间片之后,发现 2 个新目标——T_9 和 T_{10},通过仿真实验,得到图 13、表 16 和表 17。从表 17 可以看出,随着在 240 时间片之后增加 2 目标,最优函数值和种群平均函数值都增加,并且优化时间消耗也增加。从图 13 可知,任务分配的最优函数值和平均函数值随着迭代次数的增加趋于稳定,任务分配的各个指标值随迭代次数增加而渐趋于稳定,而且其值走向符合任务分配各指标优化原则,提高打击敌方目标效费比。

表 14　新增目标的信息设置

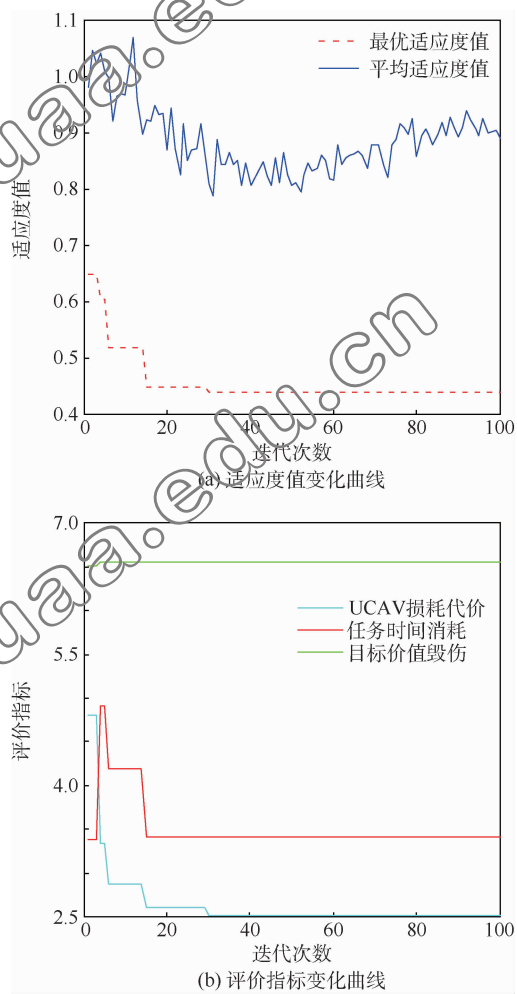
Table 14　Information setting of increased targets

目　标	类　型	位置/km	价 值 量	速度/(m·s^{-1})
T_9	SAM	(47,53)	0.75	(−25,0)
T_{10}	SAM	(69,68)	0.75	(25,0)

表 15　新增目标依赖矩阵

Table 15　Dependence matrix of increased targets

目　标	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}
T_9	0.5	0.2	0.6	0.3	0.1	0.1	0.6	0.1	1	0.9
T_{10}	0.4	0.1	0.55	0.2	0.3	0.2	0.7	0.1	0.9	1



(a) 适应度值变化曲线

(b) 评价指标变化曲线

图 13　增加目标后一个时间片内的优化结果

Fig. 13　Optimized result in a time slice after increased targets

http://bhxb.buaa.edu.cn

表 16 增加目标前后的任务分配方案比较

Table 16 Comparison of task allocation scheme before and after increased targets

UCAV	V_1		V_2		V_3		V_4		V_5		V_6	
	增加目标前	增加目标后	增加目标前	增加目标后	增加目标前	增加目标后	增加目标前	增加目标后	增加目标前	增加目标后	增加目标前	增加目标后
目 标	T_3	T_3	T_1, T_2	T_1, T_2	T_5	T_9, T_4	T_4	T_{10}, T_5	T_6	T_6	T_8, T_7	T_8, T_7

表 17 增加目标前后的任务分配优化结果对比

Table 17 Comparison of task allocation optimized result before and after increased targets

优化指标	优化时间/s		最优适应度值		平均适应度值	
	增加目标前	增加目标后	增加目标前	增加目标后	增加目标前	增加目标后
数 值	0.262 5	0.287 1	0.323 5	0.441 6	0.498 0	0.883 0

5 结 论

1）针对 UCAVs 协同对地打击多目标任务分配问题的实时性、高效性以及复杂性，对 UCAV 编队信息以及对任务目标信息进行分析，设计了基于时间片的多约束动态任务分配模型。

2）传统算法对动态任务分配问题求解速度慢、求解精度不高的缺点，提出了一种基于 SAGWO 算法的动态任务分配方法，同时证明了改进算法全局收敛，并分析了时间复杂度。

3）本文对所设计的基于 SAGWO 的任务分配从静态与动态 2 种情况进行仿真验证。仿真结果表明，SAGWO 算法可快速精确地求解出动态任务分配，且分配结果符合作战要求。由于篇幅有限，对于 UCAVs 武器分配尚且研究有限，需要进一步深入研究。

参考文献（References）

[1] 龙涛，朱华勇，沈林成. 多 UCAV 协同中基于协商的分布式任务分配研究[J]. 宇航学报，2006，27(3)：457-462.
LONG T，ZHU H Y，SHEN L C. Negotiation-based distributed task allocation for cooperative multiple unmanned combat aerial vehicles[J]. Journal of Astronautics，2006，27(3)：457-462(in Chinese).

[2] 杨啸天，刘小军，冯金富，等. 不确定环境下空地多目标攻击优先权决策[J]. 南京理工大学学报，2012，36(4)：567-571.
YANG X T，LIU X J，FENG J F，et al. Priority decision of air to surface multi-target attack under uncertainty[J]. Journal of Nanjing University of Science and Technology，2012，36(4)：567-571(in Chinese).

[3] 王永泉，罗建军. 基于多群体改进萤火虫算法的 UCAV 协同多目标分配[J]. 西北工业大学学报，2014，32(3)：451-455.
WANG Y Q，LUO J J. Target assignment in cooperative attacking of UCAVs based on multi-intelligence improved glowworm swarm optimization algorithm[J]. Journal of Northwestern Polytechnical University，2014，32(3)：451-455(in Chinese).

[4] 颜骥，李相民，刘波. 基于离散粒子群-郭涛算法分配多无人机协同任务[J]. 国防科技大学学报，2015，37(4)：165-171.
YAN J，LI X M，LIU B. Cooperative task allocation of multi-UAVs with mixed DPSO-GT algorithm[J]. Journal of National University of Defense Technology，2015，37(4)：165-171(in Chinese).

[5] RABBATH C A，GOGNON E，LAUZON M. On the cooperative control of multiple unmanned aerial vehicles[J]. IEEE Canadian Review，2004(46)：15-19.

[6] AMATO P，FARINA M. An alift-inspired evoluntionary algorithm for dynamic multiobjective optimization problems[J]. Soft Computing Methodologies and Applications，2005，32：113-25.

[7] 魏政磊，赵辉，韩邦杰，等. 基于自适应 GWO 的多 UCAV 协同攻击目标决策[J]. 计算机工程与应用，2016，52(18)：257-261.
WEI Z L，ZHAO H，HAN B J，et al. Research on cooperative attack decision of unmanned combat aerial vehicles using self-adaptive grey wolf optimization[J]. Computer Engineering and Application，2016，52(18)：257-261(in Chinese).

[8] 罗德林，段海滨，吴顺祥，等. 基于启发式蚁群算法的协同多目标攻击空战决策研究[J]. 航空学报，2006，27(6)：1166-1170.
LUO D L，DUAN H B，WU S X，et al. Research on air combat decision-making for cooperative multiple target attack using heuristic ant colony algorithm[J]. Acta Aeronautica et Astronautica Sinica，2006，27(6)：1166-1170(in Chinese).

[9] 郑昌文，严平，丁明跃. 飞行器航迹规划研究现状与趋势[J]. 宇航学报，2007，28(6)：1441-1446.
ZHENG C W，YAN P，DING M Y. Research status and trend of route planning for flying vehicles[J]. Journal of Astronautics，2007，28(6)：1441-1446(in Chinese).

[10] 王宝龙，黄考利，马立元，等. 基于依赖矩阵的测试性分析[J]. 计算机测量与控制，2011，19(6)：1260-1265.
WANG B L，HUANG K L，MA L Y，et al. Dependency matrix based testability analysis[J]. Computer Measurement & Control，2011，19(6)：1260-1265(in Chinese).

[11] 王庆江，彭军，曾儒伟，等. 无人机对地多目标攻击决策研究[J]. 电光与控制，2014，21(11)：57-61.
WANG Q J，PENG J，ZENG R W，et al. Decision-making of UAVs for air-to-ground multi-target attacking[J]. Electronics Optics & Control，2014，21(11)：57-61(in Chinese).

[12] 潘峰，陈杰，任智平，等. 基于计算智能方法的无人机任务指派约束优化模型研究[J]. 兵工学报，2009，30(12)：1706-1713.
PAN F，CHEN J，REN Z P，et al. Research on vehicle assign-

http://bhxb.buaa.edu.cn

ment model for constraints handing based on computational intelligence algorithms[J]. Acta Armamentarii, 2009, 30(12): 1706-1713(in Chinese).

[13] CHANDLER P R, PACHTER M, SWAROOP D, et al. Complexity in UAV cooperative control[C]// The Proceedings of American Control Conference. Piscataway, NJ: IEEE Press, 2012: 1831-1836.

[14] SEYEDALI M, SEYED M M, ANDREW L. Grey wolf optimizer [J]. Advances in Engineering Software, 2014, 69: 46-61.

[15] SEYEDALI M, SHAHRZAD S, SEYED M M, et al. Multi-objective grey wolf optimizer: A novel algorithm for multi-criterion optimization[J]. Expert Systems With Application, 2016, 47: 106-119.

[16] ZHU A J, XU C P. Hybridizing grey wolf optimization with differential evolution for global optimization and test scheduling for 3D stacked SoC[J]. Journal of Systems Engineering and Electronics, 2015, 26(2): 317-328.

[17] 龙文,赵东泉,徐松金. 求解约束优化问题的改进灰狼优化算法[J]. 计算机应用,2015,35(9):2590-2595.
LONG W, ZHAO D Q, XU S J. Improved grey wolf optimization algorithm for constrained optimization problem[J]. Journal of Computer Application, 2015, 35(9): 2590-2595(in Chinese).

[18] 魏政磊,赵辉,李牧东,等. 控制参数值非线性调整策略的灰狼优化算法[J]. 空军工程大学学报(自然科学版),2016,17(3):104-110.
WEI Z L, ZHAO H, LI M D, et al. A grey wolf optimization algorithm based on nonlinear adjustment strategy of control parameter[J]. Journal of Air Force Engineering University(Natural Science Edition), 2016, 17(3): 104-110(in Chinese).

[19] 周敏,李太勇. 粒子群优化算法中的惯性权重值非线性调整[J]. 计算机工程,2011,37(5):68-72.
ZHOU M, LI T Y. Nonlinear adjustment strategy of inertia weight in particle swarm optimization algorithm[J]. Computer Engineering, 2011, 37(5): 68-72(in Chinese).

[20] CHATTERJEE A, SIARRY P. Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization[J]. Computers & Operations Research, 2006, 33(3): 859-871.

[21] 李牧东,赵辉,翁兴伟. 具有广泛学习策略的回溯搜索优化算法[J]. 系统工程与电子技术,2015,37(4):958-963.
LI M D, ZHAO H, WENG X W. Backtracking search optimization algorithm with comprehensive learning strategy[J]. Systems Engineering and Electronics, 2015, 37(4): 958-963(in Chinese).

[22] SOLIS F, WETS R. Minimization by random search techniques [J]. Mathematics of Operations Research, 1981, 6(1): 19-30.

[23] TAL S, STEVEN J R, ANDREW G S, et al. Multiple task assignments for cooperating uninhabited aerial vehicles using genetic algorithms[J]. Computers & Operations Research, 2006, 33(11): 3252-3269.

作者简介:

魏政磊　男,博士研究生。主要研究方向:无人机作战系统与智能优化算法。

赵辉　男,博士,教授,博士生导师。主要研究方向:无人机作战系统与智能优化算法。

黄汉桥　男,博士,副教授。主要研究方向:无人机作战系统。

王骁飞　男,博士研究生。主要研究方向:无人机作战系统与智能优化算法。

周瑞　男,硕士研究生。主要研究方向:无人机作战系统。

http://bhxb.buaa.edu.cn

Dynamic UCAVs cooperative task allocation based on SAGWO algorithm

WEI Zhenglei[1,*], ZHAO Hui[1], HUANG Hanqiao[2], WANG Xiaofei[1], ZHOU Rui[1]

(1. Aeronautics Engineeering Couege, Air Force Engineering University, Xi'an 710038, China;
2. Unmanned System Research Institute, Northwestern Polytechnical University, Xi'an 710072, China)

Abstract: Through analyzing unmanned combat aerial vechicle (UCAV) advantage probability and task joint threat and defining task time, the task allocation model for UCAVs with multi-constraint dynamic task allocation is built up, which takes target value damage, UCAV attrition and task expending time as the performance indexes, and the improved grey wolf optimization (GWO) algorithm is used to solve the model. Aimed at the flaw of early convergence from the original algorithm, the GWO algorithm is improved by proposing a self-adaptive adjustment strategy and a step-out local optimum strategy, using quadratic curve control method. According to the characteristics of UCAVs dynamic cooperative task allocation, target task sequence coding is designed to present the UCAVs dynamic task allocation method based on self-adaptive GWO (SAGWO) algorithm. Finally, the simulation results for static and dynamic task allocation show that the task allocation method based on SAGWO algorithm is valid, and compared with other algorithms, the optimizing process is rapid and accurate.

Keywords: unmanned combat aerial vehicles (UCAVs); dynamic cooperative task allocation; target dependence matrix; task time slice; self-adaptive grey wolf optimization (SAGWO) algorithm

http://bhxb.buaa.edu.cn

Received: 2017-09-21; **Accepted**: 2017-12-11; **Published online**: 2018-01-31 16:34

URL: kns.cnki.net/kcms/detail/11.2625.V.20180131.1611.001.html

Foundation items: National Natural Science Foundation of China (61601505); Aeronautical Science Foundation of China (20155196022)

* **Corresponding author**. E-mail: zhenglei_wei@126.com