

文章编号: 1001-0920(2010)09-1359-05

基于粒子群算法的多无人机任务分配方法

李 炜, 张 伟

(电子科技大学 电子科学技术研究院, 成都 610054)

摘 要: 作为多无人机系统应用的一项关键技术, 任务分配是一个多维互异离散变量的优化问题. 采用混合整数线性规划方法构造优化函数, 并利用群智算法中的粒子群算法来求最优解, 这样可以解决多无人机的任务分配问题. 针对互异性要求进行必要的算法改进. 数值仿真实验表明, 该粒子群算法可以迅速找到优化函数的最优解, 从而高效地实现多无人机的任务分配.

关键词: 多无人机; 任务分配; 粒子群算法; 多维互异变量; 代价函数

中图分类号: V249.122

文献标识码: A

Method of tasks allocation of multi-UAVs based on particles swarm optimization

LI Wei, ZHANG Wei

(Research Institute of Electronic Science and Technology, University of Electronic Science and Technology of China, Chengdu 610054, China. Correspondent: LI Wei, E-mail: aegeanwei@gmail.com)

Abstract: Task allocation is a kind of key technology in the application of multi-UAVs systems, and it is a mutually exclusive multi-dimensions optimal problem. Mixed integer linear program is applied to build the optimal function, and particle swarm optimization(PSO) algorithm is utilized for searching the best solution in order to realize the task allocation of multi-UAVs system. In the PSO algorithm, it is necessary to make some modification to suit for the acquirement of mutually exclusive. The numerical simulation shows that the presented algorithm can find the optimal solution rapidly and accomplish the task allocation of multi-UAVs efficiently.

Key words: Multi-UAVs; Tasks allocation; PSO; Mutually exclusive; Cost function

1 引 言

由多个简单无人机通过协同、协作组成集群系统来完成单个无人机无法完成或难以完成的工作^[1], 已经成为各国研究者的普遍共识. 因此, 对于多无人机系统的研究受到了广泛关注. 多无人机可实现多种作战任务, 而基本的工作状态包括 4 种: 1) 搜索目标状态: 遵照事先规划的搜索路径飞行, 寻找地面目标; 2) 识别目标状态: 利用目标识别算法判断目标为真目标、被击中的目标还是暂时无法确切判断类型的目标; 3) 攻击目标状态: 对于已经被识别为真的目标, 无人机进入交会阶段, 对其实施攻击; 4) 毁伤评估状态: 判断被攻击的目标是否还具备战斗力^[2-4].

针对无人机在某一时刻发现的目标, 如何能够合理地将无人机以最佳的任务状态分配给最适合的目

标是能否发挥多无人机协同工作效能的关键. 国内外对该问题的研究已经成为热点. 其中有代表性的研究成果包括: 1) 采用网络最优模型进行任务分配的方法, 它是模拟货物的供应需求的转换和变化, 从而实现无人机和目标之间的任务分配^[5,6]; 2) 利用多智能体的方法, 将每个无人机都作为一个智能体, 通过相互之间的交互实现任务的分配^[7,8]; 3) 利用禁忌搜索的方法, 依次对无人机分配任务, 反复迭代, 直至找到最优解^[9]; 4) 采用混合整数线性规划方法, 该方法可以找到一个最优解, 但计算量较大^[10,11]. 因此, 在此基础上, 有很多研究者结合蚁群算法、退火算法、遗传算法等最优化方法找到最优解. 目前, 通过模拟生物群体的行为来解决计算问题已经成为了新的研究热点. 其中粒子群算法作为一种具有较强寻优能力、简

收稿日期: 2009-08-27; 修回日期: 2009-10-23.

基金项目: 国防预研项目(402030203); 电子科技大学中青年学术带头人培养计划项目.

作者简介: 李炜(1980—), 男, 内蒙古包头人, 助理研究员, 博士, 从事多机协同、智能控制等研究; 张伟(1974—), 男, 四川达州人, 副研究员, 博士, 从事智能系统、仿真技术等研究.

单通用,且鲁棒性强的方法被广泛应用于科学研究和工程实践中.本文的目标就是采用整形线性规划方法构造最优函数,利用群智算法中的粒子群算法求出最优解,从而实现多无人机的目标分配.

2 构造任务分配函数

无人机根据识别目标的类型所需执行的任务包括 3 种: 1) 对于真目标, 执行攻击任务; 2) 对于被攻击过的目标, 执行毁伤评估任务; 3) 对于无法判断类型的目标, 执行再识别的任务. 无人机根据自身与目标所处的位置和角度对目标进行分类识别, 并且确定分类识别的置信度. 如果识别结果低于预置的置信域, 则报告为暂时无法识别类型的目标, 这样选择无人机执行再识别的任务; 一旦目标被确认为真目标, 则无人机可以选择该目标进行交会; 一旦无人机和目标满足了交会条件, 无人机即可对目标进行攻击; 目标被攻击后, 其他无人机可对其进行毁伤评估, 判断其状态, 完全被摧毁或是仍然可以工作.

从上述的描述可以看出, 无人机对目标识别分类的准确程度是影响任务分配的首要因素. 没有对目标正确的识别也就不会有合理的任务分配. 为了简化目的, 本文在建模时, 认为无人机可以正确地对目标进行识别分类, 而不会出现识别类型错误的情况. 此外, 值得注意的是, 本文研究的对象是微小型无人机, 其对目标的攻击是通过自身的战斗部起爆来实现的, 即无人机实施攻击也意味着本身生命周期的结束. 无人机的任务分配过程中需要考虑的因素包括: 任务的优先级、时间约束以及任务路径的可飞性. 任务的优先级是指对于每个被发现的目标根据其识别类型顺次执行相应的任务, 例如经识别认为是真目标时, 则应该先执行攻击任务, 然后才能执行毁伤评估. 时间约束是指给定的任务需要在一个特定的时间域内完成, 超过了这个时间域, 就可能意味着任务的失败, 特别是在打击时间敏感目标的过程中, 如果完成任务所需的时间过长, 则很可能会丢失目标, 从而使任务失败. 如果假设所有无人机的飞行速度相同, 那么任务路径越短, 所对应的任务完成时间也就越短. 任务路径的可飞性是指无人机在执行任务时所需飞过的路径满足运动学和动力学约束条件.

2.1 距离因素

如果假设无人机飞行速度相同, 则时间约束条件就转化为无人机与目标之间的距离因素. 无人机与目标的距离越近, 则完成任务的时间就越短, 因此将该无人机分配给此目标的机率也就越大. 用数学表达式表示如下: 设距离因素对任务分配的影响为 α_1 , 则

$$a_1 = w_1 d_{ij} / d_{\max}. \quad (1)$$

其中: d_{ij} 表示第 j 个无人机到第 i 个目标直接的距离; d_{\max} 表示所有无人机中相对第 i 个目标的最大距离; w_1 为 $(0 - 1)$ 之间的一个数, 表示距离因素对于任务分配影响程度的权重值, 数值越大表明距离因素对任务分配的结果影响越大, 反之则越小.

2.2 角度因素

任务路径的可飞性要求飞行路径的最大曲率要满足无人机飞行性能的要求. 在任务分配时, 无人机所产生的路径受到多种不确定因素的影响, 很难精确地对可飞路径加以量化描述. 本文主要根据无人机当前的飞行方向与目标视线之间的夹角来描述产生任务路径的可飞性. 如图 1 所示, θ 表示无人机当前飞行方向与目标视线之间的夹角, 虚线表示任务路径. 由图 1 可以看出, 目标 1 与无人机 1 生成的夹角 θ_{11} 要小于目标 2 与无人机 1 生成的夹角 θ_{21} , 相应的任务路径的曲率也较小, 可飞性也就相应的越好; 而对目标 2 而言, 与无人机 1 生成的夹角 θ_{21} 要大于与无人机 2 生成的夹角 θ_{22} , 相应的任务路径曲率也较大, 所以可飞性也就越差. 因此, 利用无人机当前的飞行方向与目标视线之间的夹角可大致地定量描述产生任务路径的可飞性.

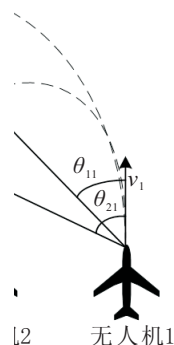


图 1 无人机可飞性示意图

设无人机当前的飞行方向与目标视线之间的夹角对任务分配的影响为

$$a_2 = w_2 \theta_{ij} / \theta_{\max}. \quad (2)$$

其中: θ_{ij} 为第 j 个无人机的飞行方向与第 i 个目标视线的夹角; θ_{\max} 为无人机飞行方向与目标视线的最大夹角, 根据夹角的定义, $\theta_{\max} = 180$; w_2 为飞行方向与目标实现夹角对于任务分配结果的影响程度, 数值越大, 表明对结果的影响越显著, 反之则越微弱.

2.3 时间因素

无人机剩余飞行时间主要用来评价执行任务的代价. 例如在执行攻击任务时, 选择一架刚出发的无人机要比选择一架燃油即将耗尽的无人机付出的代价大得多. 因此, 在进行任务分配时, 尽可能将作战任务分配给剩余飞行时间少的无人机. 对任务分配的影

响为

$$a_3 = w_3 \frac{t_j}{t_{\max}}. \quad (3)$$

其中: t_j 为无人机 j 的剩余飞行时间; t_{\max} 为无人机最大的飞行时间; w_3 为无人机剩余飞行时间对任务分配结果的影响程度, 数值越大, 表明剩余飞行时间对结果的影响越大, 反之越小.

综上所述, 可构造无人机任务分配的代价函数为

$$C(x_{ij}) = P_i(\alpha_1 + \alpha_2 + \alpha_3) = P_i\left(w_1 \frac{d_{ij}}{d_{\max}} + w_2 \frac{\theta_{ij}}{\theta_{\max}} + w_3 \frac{t_j}{t_{\max}}\right). \quad (4)$$

其中: x_{ij} 为 j 无人机作用于 i 目标; P_i 为目标的识别概率, 当 P_i 落入不同的识别置信区域时, 分别判定为无法识别类型的目标、真目标、被攻击的目标; 相对应地, 将 $C(x_{ij})$ 的定义域写为 $\{R_{\text{cost}}, A_{\text{cost}}, B_{\text{cost}}\}$, 分别为再识别任务代价、攻击任务代价和毁伤评估任务代价. 对应不同的目标类型, 式中的权重取值不同.

2.4 优化问题

由此可将任务规划问题转化为寻找下列优化问题的最优解:

$$\text{Cost} = \min \sum_{i=0}^n C(x_{ij})x_{ij}. \quad (5)$$

$$\text{s.t. } C(x_{ij}) \in \{R_{\text{cost}}, A_{\text{cost}}, B_{\text{cost}}\}; \quad (6)$$

$$\sum_i^n x_{ij} = 1, \quad j = 1, 2, \dots; \quad (7)$$

$$\sum_j^m x_{ij} = 1 \text{ or } 0, \quad i = 1, 2, \dots; \quad (8)$$

$$x_{ij} \in \{0, 1\}, i = 1, 2, \dots, j = 1, 2, \dots. \quad (9)$$

其中: n 为目标的数量, m 为无人机的数量, i 为目标编号, j 为无人机编号, $C(x_{ij})$ 为 i 无人机与 j 目标之间的代价系数. 约束条件 (6) 表示代价系数只能是 3 种代价系数之一; 约束条件 (7) 表示对于任意一个目标, 当前时刻只能有一个无人机对其发生作用; 约束条件 (8) 表示在任意时刻, 任意一个无人机至多对一个目标发生作用, 并假设无人机的数量大于等于各类目标的总数量, 对于未与目标配对的无人机则继续执行搜索任务; 约束条件 (9) 表示 $x_{ij} = 0$ 为 (0-1) 变量, 当 j 无人机作用于 i 目标时, $x_{ij} = 1$, 否则 $x_{ij} = 0$, 表示如下:

$$x = \begin{cases} 1, & \text{子弹药 } j \text{ 作用于目标 } i; \\ 0, & \text{否则.} \end{cases}$$

3 粒子群算法求解

粒子群优化算法是由美国的 Kennedy 和 Elberhart 受到鸟群觅食行为的启发, 于 1995 年提出的. 最初的设想是仿真简单的社会系统, 研究解释复杂的社会行

为, 后来发现粒子群优化算法可用于复杂优化问题的求解. 在粒子群优化中, 被称为粒子的个体是通过超维搜索空间“流动”的, 粒子在搜索空间内的位置变化是以一个个体成功地超过其他个体的社会心理意向为基础的. 因此, 群中粒子的变化受到其邻近粒子经验或知识的影响, 一个粒子的搜索行为受到群中其他粒子的搜索行为的影响.

粒子群算法具有计算的快速性和算法本身的易实现性, 由于粒子群迁移过程是有方向性的, 搜索过程中运用反馈原理并采用并行计算技术, 具有较高的搜索效率, 这是其他算法所不具备的优势.

无人机的任务分配问题是离散变量问题, 因此利用粒子群算法来进行优化问题求解时, 需要将更新的速度离散化, 粒子的位置也随之被离散化. 具体方法是利用 Miranda 和 Fonscca 提出的概率舍入的方法^[12], 按速度更新后的数值到取值空间内每个整数的距离的概率比例进行舍入.

3.1 基本优化算法

将 $\sum_{i=0}^n x_{ij}$ 记为微粒 X_k , 其位置变化表现为编号为 i 的目标和编号为 j 的无人机之间的任务配对情况. 它经历过的最好位置记为 p_{best} , 在群体所有微粒经历过的最好位置的索引号记为 g_{best} .

微粒 k 的速度用 $V_k = (v_{k1}, v_{k2}, \dots, v_{kn})$ 表示, 对于每一代, v_{ki} 根据如下方程迭代:

$$v_{ki}(t+1) = w_p v_{ki}(t) + c_1 r_1 (p_{ki}(t) - X_{ki}(t)) + c_2 r_2 (p_{gi}(t) - X_{gi}(t)), \quad (10)$$

$$X_{ki}(t+1) = X_{ki}(t) + v_{ki}(t+1). \quad (11)$$

其中: w_p 为线性递减权重^[13]; c_1, c_2 为学习常数, r_1, r_2 为介于 0 到 1 的随机数; $p_{ki}(t)$ 为每个微粒到目前为止, 所出现的最优位置; $p_{gi}(t)$ 为所有微粒到目前位置, 所出现的最优位置.

在搜索时, 微粒的位置被最大位置和最小位置限制, 如果某微粒在某维的位置超出该维的最大位置或者最小位置(弹目的任务分配超过了目标数或无人机数范围), 则该微粒的位置被限制为该维的最大位置或最小位置. 同样, 微粒的速度也被限制于最大速度和最小速度之间^[14], 具体公式如下:

$$v_{ki} = \begin{cases} v_{\max}, & v_{ki} > v_{\max}; \\ -v_{\max}, & v_{ki} < -v_{\max}. \end{cases}$$

其中: v_{\max} 为设定的最大速度向量. 当粒子速度过大时, 将可导正回适当速度向量, 如粒子往负向速度过大时, 则会限定最大负向速度.

3.2 粒子群改进算法流程

在任务分配的求解问题中, 变量的取值是离散

的, 所以为非连续的优化问题. 在利用PSO算法进行优化时, 必须进行变量的离散化: 一方面需要将粒子的位置离散化, 另一方面也要将更新速度离散化. 同时更为重要的是, 在利用PSO算法时, 必须要满足约束条件(7)和(8)的要求. 因此粒子在选择新位置的时候, 并不是在全定义域内进行选择. 粒子移动的位置表示该粒子对于任务分配的结果, 目标的数目也是粒子的维数, 粒子的更新速度为无人机和目标之间的任务配对的变化情况. 根据约束条件要求, 任意目标只能有一个无人机对其发生作用, 任一无人机至多也只能对一个目标执行任务. 因此在PSO算法的寻优过程中, 微粒各维的最终位置应该是互异的, 在本文中采用多次迭代的方法, 直到各维变量的位置都不同时, 才继续往下执行, 具体的算法流程如图2所示.

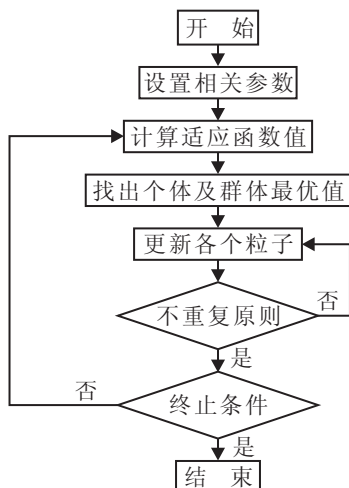


图2 粒子群算法流程图

Step1: 初始化设置微粒群的规模, 惯性权值, 加速系数, 最大允许迭代次数.

Step2: 针对每个微粒, 生成随机的初始无人机和目标的任务配对, 并计算目标评价函数.

Step2.1: 依次对每个目标分配一个无人机来执行作战任务, 如果无人机和目标的任务分配出现同一个无人机对两个不同的目标执行任务时, 则再重新分配, 直到此类情况不再出现(不重复原则).

Step2.2: 当每个微粒的初始弹目任务分配满足要求时, 按目标评价函数评价各微粒的初始适应值.

Step3: 根据公式计算各微粒新的速度, 并对各微粒新的速度进行限幅处理.

Step4: 根据公式计算各微粒新的位置, 并对各微粒新的位置进行限幅处理.

Step5: 当微粒处于新位置时, 同时存在微粒对于无人机和目标的任务分配出现 Step2.1 的情况时, 重新执行 Step3 和 Step4, 直到满足要求.

Step6: 按目标评价函数重新评价各微粒适应值.

Step7: 对每个微粒, 比较其当前适应值和其个体

经历过的最好适应值, 若当前适应值更优, 则令当前适应值为其个体历史最好的适应值, 并保存当前位置为其个体历史最好位置(个体最优弹目任务分配).

Step8: 比较群体所有微粒的当前适应值和全局历史最好的适应值, 若某微粒的当前适应值更优, 则令该微粒的当前适应值为全局历史最好适应值, 并保存该微粒的当前位置为全局历史最好位置(全局历史最优弹目任务分配).

Step9: 若满足停止条件, 则搜索停止, 输出搜索结果; 否则, 返回 Step3 继续搜索.

Step10: g_{best} 为搜索到的最优值.

粒子群改进算法伪码如下:

初始化

粒子数量、最大迭代次数、结束标志阈值、目标数量、无人机数量;

For $i = 1$ 粒子数

计算每个粒子的初始位置(为每个目标分配无人机);

计算粒子初始移动速度;

End;

Do

For $i = 1$ 粒子数

根据目标代价函数计算粒子 p_i 的个体适应值;

If 此适应值优于此前该粒子最好位置 p_{best} 的适应值

更新 p_{best} 值;

End;

End;

从所有粒子中找出个体适应值最大的位置 p_{best} 赋给 g_{best} , 并记录该粒子的任务分配结果;

For $i = 1$ 粒子数

根据公式更新 p_i 的速度, 速度被限制在 $[-v_{max}, v_{max}]$ 之间;

根据公式更新 p_i 的位置, 位置被限制在可行域上(可行的任务分配);

End;

While(当最优适应值大于结束标志域且迭代次数小于最大迭代数)

输出最优结果.

4 数值仿真

为了验证本文提出的任务分配算法的性能, 设定静止目标数量为 5 个, 分别处于(300, 10), (350, 10), (400, 10), (450, 10)和(500, 10). 其中: 目标 3 为无法识别的目标, 目标 4 为被攻击过的目标, 其他目标都为

真目标. 无人机的数量为50个, 其 x 坐标50开始, 依次增大10; y 坐标都为500, 速度方向都是指向 y 轴的负半轴. 分别设粒子数量为1000, 2000, 5000, 最大迭代次数为100, 惯性权值设为0.5. 经过多次计算得出的最终结果为: 第25号无人机对第1个目标执行攻击任务, 第30号无人机对第2个目标执行攻击任务, 第35号无人机对第3个目标执行再识别任务, 第40号无人机对第4个目标执行毁伤评估的任务, 第45号无人机对第5个目标执行攻击任务, 最终的代价函数适应值为0.78.

在粒子数设为1000时, 经过100次的迭代会出现结果不最优的情况. 而当粒子数设为5000时, 每次计算都可以获得最优解. 由此可以看出, 粒子数的规模对寻找最优解有直接的影响, 如图3所示.

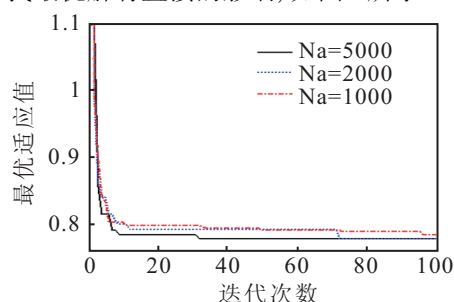


图3 适应函数值收敛曲线

由图3可知, 在粒子数较多时, 经过较少的迭代次数(30代)就可以收敛到最优值, 而当粒子较少时, 一般需要经过更多次的迭代才可以收敛到最优值.

将无人机的飞行方向改为从与 x 轴正向呈 -65° 开始, 以一度的差值依次递减. 经过多次计算得到的结果为: 第25号无人机对第1个目标执行攻击任务, 第32号无人机对第2个目标执行攻击任务, 第35号无人机对第3个目标执行再识别任务, 第38号无人机对第4个目标执行毁伤评估的任务, 第40号无人机对第5个目标执行攻击任务, 最终的代价函数适应值为1.187. 此外, 在此条件下, 将权重值 w_2 从1.5调整到0.5时, 最优结果又变为: 第23号无人机对第1个目标执行攻击任务, 第25号无人机对第2个目标执行攻击任务, 第28号无人机对第3个目标执行再识别任务, 第30号无人机对第4个目标执行毁伤评估的任务, 第32号无人机对第5个目标执行攻击任务, 最终的代价函数适应值为0.862.

5 结 论

多无人机的任务分配是多维互异离散变量寻优问题, 对其求解是一项非常耗费计算的工作. 利用概念清晰、搜索速度快、易于实现的粒子群算法来进行求解是一项非常好的选择. 本文通过对速度更新和位置更新的多次迭代来适应多维变量互异性的要求, 获得了比较好的寻优效果. 从文中的数值仿真结果可以

看出, 在1000~2000个粒子群规模的条件下, 经过几十次迭代就可以找到最优解, 耗费计算资源小, 可满足机载计算机甚至是单片机的运算要求. 因此, 利用粒子群算法可以高效快速地解决多人机的任务分配问题.

参考文献(References)

- [1] David Frelinger, Joel Kvitky, William Stanley. Proliferated autonomous weapons[R]. Santa Monica: Rand Corporation.
- [2] Corey Schumacher, Phillip R Chandler, Steven Rasmussen. Task allocation for wide area search munitions[D]. Anchorage: Wright-Patterson Air Force Base, 2002.
- [3] Robert Dunkel III. Investigation of cooperative behavior in autonomous wide area search munitions[D]. Ohio: Air Force Institute of Technology, 2002.
- [4] Orhan Gozaydin. Analysis of cooperative behavior for autonomous wide area search munitions[R]. Ohio: Air Force Institute of Technology, 2002.
- [5] Kendall Nygard, Phillip Chandler, Meir Pachter. Dynamic network flow optimization models for air vehicle resource allocation[C]. American Control Conf. Arlington, 2001: 25-27.
- [6] Schumacher C J, Chandler P R, Rasmussen S J. Task allocation for wide area search munitions via iterative network flow optimization [C]. Proc of the AIAA Guidance, Navigation, and Control Conf. Monterey, 2002: 3472-3477.
- [7] 姚宗信, 李明, 陈宗基. 多机协同作战任务决策方法多智能体结构框架[J]. 电光与控制, 2008, 15(3): 1-4.
(Yao Z X, Li M, Chen Z J. Multi-agent framework of mission decision-making method for multi-aircraft cooperative combat[J]. Electronics Optics and Control, 2008, 15(3): 1-4.)
- [8] 曹菊红, 高晓光. 多架无人机协同作战智能指挥控制系统[J]. 火力与指挥控制, 2003, 28(5): 22-24.
(Cao J H, Gao X G. Agent-based design for multi-ucav intelligent command and control cooperative system[J]. Fire Control and Command Control, 2003, 28(5): 22-24.)
- [9] Ryan J L, Bailey T G, Moore J T, et al. Unmanned aerial vehicles (UAV) route selection using reactive Tabu search [J]. Military Operations Research, 1999, (4): 5-24.
- [10] Richards A, Bellingham J, Tillerson M, et al. Coordination and control of multiple UAVs [C]. Proc of the AIAA Guidance, Navigation, and Control Conf. Monterey, 2002.
- [11] Schumacher C J, Chandler P R, Pachter M, et al. Constrained optimization for UAV task assignment [C]. Proc of the AIAA Guidance, Navigation, and Control Conference. Providence, 2004.

(下转第1368页)