

基于分解的多目标布谷鸟搜索算法求解 多无人机协同任务分配问题

陈亮^{1,2} 卢天鸣³ 曹林^{1,4} 白景波¹

(1.陆军工程大学 野战工程学院,江苏 南京 210007; 2.军事交通学院 汽车士官学校,安徽 蚌埠 233011;
3.中国电子科技集团 第二十八研究所,江苏 南京 210007; 4.中国人民解放军 66072 部队,北京 100144)

摘 要: 多无人机协同任务分配问题是一个多目标优化问题,将多目标优化问题转化为单目标优化问题的传统方法易造成决策的主观性和片面性。为帮助决策者做出科学决策,提出了一种基于分解的多目标布谷鸟搜索算法用于求解多无人机协同任务分配问题。通过对多无人机协同任务分配问题分析,建立了多无人机协同任务分配模型。将布谷鸟搜索算法的两个关键组件转化为多目标优化算法的繁殖算子,并结合一种自适应算子选择策略,构成了多目标布谷鸟搜索算法。设计了一种新的编码方案,将带约束的多目标优化问题转为无约束的多目标优化问题。仿真实验表明,多目标布谷鸟搜索算法能有效求解多无人机协同任务分配问题。

关键词: 基于分解的多目标优化算法; 协同任务分配; 布谷鸟搜索算法; 无人机

DOI: 10.19949/j.ams.mora.20200612.01

中图分类号: E911 文献标志码: A 文章编号: 1672-8211(2021)02-0005-08

1 引言

多无人机(Unmanned Aerial Vehicle, UAV)协同任务分配是近些年 UAV 研究的热点问题。多 UAV 协同任务分配是实现 UAV 协同作战的关键技术,其研究目的是实现最佳的 UAV 与任务间的配对,最大化任务效能。目前,衡量任务效能的指标主要分为两大类:预期收益和预期代价。显然这两类指标相互冲突,因此同时考虑此两类目标的多 UAV 协同任务分配问题本质上是一个多目标优化问题(Multiobjective Optimization Problem, MOP)。由于 MOP 的多个目标函数相互冲突,因此 MOP 的最优解(Pareto 最优解)通常有多个。MOP 的所有 Pareto 最优解的集合称为 Pareto 最优解集(Pareto

Optimal Set, PS)。PS 在目标空间的映射的集合称为 Pareto 最优前沿(Pareto Optimal Front, PF)。

目前,求解多 UAV 协同任务分配问题的方法主要有遗传算法^[1,2]、粒子群算法^[3,4]、灰狼算法^[5]和拍卖算法^[6]等。绝大多数算法都是通过对不同目标赋予一定的权重,然后加权求和,从而将 MOP 转化为单目标优化问题。虽然这种方法简化了问题,实际应用效果尚可,但也存在明显的不足:一方面,目标权重的确定带有较强的主观性;另一方面,决策者受限于方案的单一性,无法获得更全面的信息。尽管可以通过设置不同的权重多次求解,弥补不足,但是多次求解计算开销比较大,过程相互独立,难以信息共享,可能导致得到的结果不可比较,令决策者难以做出有效的决策。多目标优化算法

收稿日期: 2020-06-12; 修回日期: 2021-03-04

作者简介: 陈亮(1981—),男,讲师,博士研究生,主要研究方向为军事运筹和军事评估;卢天鸣(1993—),男,助理工程师,硕士,主要研究方向为系统集成;曹林(1977—),男,工程师,博士,主要研究方向为军事运筹;白景波(1982—),男,讲师,博士,主要研究方向为军事运筹。

©1994-2021 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

(Multiobjective Optimization Algorithm, MOOA) 可以一次求解得到 MOP 的多个 Pareto 最优解, 是一种求解 MOP 的较为有效的算法。决策者可以全面比较不同的解, 科学地做出决策。

基于分解的多目标优化算法 (Multiobjective Optimization Algorithm Based on Decomposition, MOOA/D)^[7] 是 MOOA 家族中一类高效算法。MOOA/D 将 MOP 分解为多个单目标子问题, 利用子问题间的邻域关系共享信息, 有效降低了问题求解的难度。为了提高 MOOA/D 的性能, 众多学者从权重向量^[7]、繁殖算子^[8]、分解方法^[9]、配对策略^[10]和替换策略^[11]等多方面对 MOOA/D 算法进行改进, 均取得了不错的效果。

本文在分析作战环境下多 UAV 协同任务分配问题的基础上, 建立了多 UAV 协同任务分配问题模型。为了有效求解 MOP, 提出了基于分解的多目标布谷鸟搜索算法 (Multiobjective Optimization Algorithm Based on Decomposition with Cuckoo Search, MOOA/D-CS)。针对带约束的 MOP 求解困难问题, 设计了一种新的编码方法, 将带约束 MOP 转化为无约束的 MOP。与多个基于 MOOA/D 框架的改进算法的结果比较表明, MOOA/D-CS 在求解 MOP 上具有竞争力, 并可有效地求解多 UAV 协同任务分配问题。

2 多 UAV 协同任务分配问题模型

2.1 问题描述

战场环境下, 有 L 架 UAV 组成的编队 $U = \{U_1, U_2, \dots, U_L\}$, 计划对 M 个目标 $T = \{T_1, T_2, \dots, T_M\}$ 展开攻击, $L > M$ 。战场中同时存在 K 个威胁源 $H = \{H_1, H_2, \dots, H_K\}$, 威胁源可协防一定范围内的被攻击目标, 对侵入 UAV 实施攻击。威胁源与目标之间的协防关系可用协防矩阵 $A = (a_{kj})_{K \times M}$ 表示。 A 是 0-1 矩阵, $a_{kj} = 1$ 表示 H_i 与 T_j 存在协防关系; $a_{kj} = 0$ 表示 H_i 与 T_j 间不存在协防关系。基于以上信息, 对 L 架 UAV 分配攻击目标。要求每一架 UAV 分配一个目标, 每个目标至少分配一架 UAV。最终得到一个 UAV 与目标间的 0-1 分配矩阵 $X = (x_{ij})_{L \times M}$, $x_{ij} = 1$ 表示 U_i 攻击目标 T_j , $x_{ij} = 0$ 表示 U_i 不攻击目标 T_j 。

2.2 威胁代价

本文选择威胁代价和攻击收益作为评估无人机作战效能的指标^[12]。

威胁代价是指 UAV 从基地起飞到发起攻击过程中, 目标和威胁源对 UAV 产生的威胁。威胁代价可用 U_i 攻击 T_j 时被命中的损失价值表示:

$$C_{ij} = \frac{d_{ij}}{d_{\max}} \left[1 - (1 - r_{ij}) \prod_{k=1}^K (1 - a_{kj} h_{ki}) \right] x_{ij} Q_i \quad (1)$$

式(1)中, C_{ij} 表示 U_i 攻击 T_j 的威胁代价; d_{ij} 表示 U_i 与 T_j 之间的欧式距离; d_{\max} 表示 UAV 与所有打击目标的欧式距离最大值; Q_i 表示 U_i 的价值; $[1 - (1 - r_{ij}) \prod_{k=1}^K (1 - a_{kj} h_{ki})] x_{ij}$ 表示 U_i 被 T_j 和 H_i 命中的概率。其中 r_{ij} 表示 U_i 被 T_j 命中的概率, h_{ki} 表示 U_i 被 H_k 命中的概率;。

2.3 攻击收益

攻击收益是指 UAV 在成功执行任务后摧毁的目标价值。 U_i 攻击目标 T_j 的攻击收益 D_{ij} 可表示为^[12]:

$$D_{ij} = \frac{\lambda_i T_j^w}{\max_{j \in \{1, 2, \dots, M\}} \{T_j^w\}} \quad (2)$$

式(2)中, $\lambda_i = 1 - \prod_{i=1}^L (1 - x_{ij} P_{ij})$ 表示 U_i 成功摧毁 T_j 的概率, 其中 $P_{ij} = (1 - r_{ij}) e_i$ 表示 U_i 攻击 T_j 的毁伤概率, r_{ij} 表示 U_i 被 T_j 命中的概率, e_i 表示 U_i 的杀伤概率; T_j^w 为目标 T_j 的价值。

2.4 约束条件

$$\sum_{j=1}^M x_{ij} = 1 \quad i = 1, 2, \dots, L \quad (3)$$

$$\sum_{i=1}^L x_{ij} \geq 1 \quad j = 1, 2, \dots, M \quad (4)$$

式(3)表示每架 UAV 有且仅有一个攻击目标。式(4)表示每个目标至少被一架 UAV 攻击。

综合以上分析, 可建立多 UAV 协同任务分配问题模型:

$$\begin{aligned} \min F(x) &= [f_1(x) \quad f_2(x)]^T \\ \text{s.t.} &\begin{cases} \sum_{j=1}^M x_{ij} = 1 \quad i = 1, 2, \dots, L \\ \sum_{i=1}^L x_{ij} \geq 1 \quad j = 1, 2, \dots, M \end{cases} \end{aligned} \quad (5)$$

$$\text{式(5)中 } f_1(x) = \frac{\sum_{i=1}^L \sum_{j=1}^M C_{ij}}{L}, f_2(x) = 1 - \frac{\sum_{i=1}^L \sum_{j=1}^M D_{ij}}{M}.$$

2.5 决策准则

在决策者无偏好的情况下,可设置自主决策准则选取某一个分配方案。本文选择最大化攻击收益与威胁代价比 ξ 来确定最终的分配方案 ξ 可按下式计算:

$$\xi = \frac{1 - f_2(x)}{f_1(x)} \quad (6)$$

3 基于 MOOA/D - CS 的无人机协同任务分配算法

3.1 布谷鸟搜索算法

受布谷鸟繁殖行为和昆虫莱维飞行的启发, Yang 等^[13] 提出了 CS 算法。CS 算法由 Lévy flights 随机游走和偏好随机游走两个关键部件组成。Lévy flights 随机游走按照下式生成新解:

$$x_{i+1} = x_i + \alpha_0 \times (x_i - x_g) \otimes \text{Lévy}(\beta) \quad (7)$$

式(7)中 x_i 表示第 i 个解; α_0 是与问题规模有关的步长参数; x_g 表示全局最优解; \otimes 表示点乘积 Lévy(β) 服从 Lévy 概率分布:

$$\text{Lévy}(\beta) = \frac{\varphi \times u}{|v|^{\frac{1}{\beta}}} \quad (8)$$

式(8)中 $\beta \in (0, 2]$ 是控制参数; u 和 v 服从标准正态分布。

$$\varphi = \left[\frac{\Gamma(1 + \beta) \times \sin\left(\frac{\pi \times \beta}{2}\right)}{\Gamma\left(\frac{1 + \beta}{2}\right) \times \beta \times 2^{\frac{(\beta-1)}{2}}}\right]^{\frac{1}{\beta}} \quad (9)$$

式(9)中 $\Gamma(\cdot)$ 表示伽马函数。

偏好随机游走按照下式生成新的解:

$$x_{i+1} = x_i + r \times H(p_a - \varepsilon) \otimes (x_j - x_k) \quad (10)$$

式(10)中 $H(\cdot)$ 是 Heaviside 函数 r 和 ε 均为 $(0, 1)$ 区间的均匀分布随机数 x_j 和 x_k 是从种群中选择的不同于 x_i 的相异随机解, p_a 是控制参数。

3.2 全局随机游走算子和局部游走算子

CS 算法的 Lévy flights 随机游走和偏好随机游走分别承担全局搜索和局部搜索作用,将其转换为繁殖算子并用于 MOOA/D 中,将有希望提高算法性能。对 Lévy flights 随机游走进行修改,得到新的交叉算子:

$$y = x_i + \alpha_0 \times (x_i - x_{r1}) \otimes \text{Lévy}(\beta) \quad (11)$$

式(11)中 x_i 表示父代的第 i 个个体, x_{r1} 表示父代种群中不同于 x_i 的随机个体。需要指出的是当 y 的任意维越界,将会被重置为边界值。将以上交叉算子与多项式变异算子^[14] 结合,组成新的繁殖算子,称作 GRW(Global Random Walk) 算子。

局部随机游走可直接改写为新的交叉算子:

$$y = x_i + r \times H(p_a - \varepsilon) \otimes (x_{r1} - x_{r2}) \quad (12)$$

式(12)中 x_{r1} 和 x_{r2} 是从种群中随机选择的不同于 x_i 的个体。如果 y 的任意维越界,同样将会被重置为边界值。该算子直接作为新的繁殖算子,记作 LRW(Local Random Walk) 算子。

3.3 自适应算子选择

GRW 算子擅长全局探索, LRW 算子擅长局部开发,两个算子在功能上具有互补性。因此,可将这两个算子组成一个算子池,在进化过程中,根据算子的性能自适应选择合适的繁殖算子。本文采用一种基于多臂赌博机的自适应算子选择策略^[15] 选择繁殖算子。

首先,建立一个二维数组 SW 存放繁殖算子使用过程中产生的有效信息。 $SW(1)(j)$ ($j = 1, 2, \dots, W$) 记录选择的算子 op_i 的序号, $SW(2)(j)$ ($j = 1, 2, \dots, W$) 记录选择的算子 op_i 获得的适应度改进率(Fitness Improvement Rate, FIR)。FIR _{i} 按照下式计算:

$$FIR_i = \frac{pf_i - cf_i}{pf_i} \quad (13)$$

式(13)中 pf_i 是父代的适应值, cf_i 是子代的适应值。数组 SW 遵循先进先出的原则:最新获得的 FIR 将会被放在数组的尾部,当存放的元素超出了数组的容量,数组中的第一列将会被移除。奖励 $Reward_i$ ($i = 1, 2$) 是存放于数组 $SW(2)(W)$ 由算子 op_i 获得的所有 FIR 的累和。然后,按照奖励 $Reward_i$ ($i = 1, 2$) 给算子 op_i 分配信用 FRR_i :

$$FRR_i = \frac{Reward_i}{\sum_{i=1}^2 Reward_i} \quad (14)$$

最后,按照下式选择算子:

$$op_i = \arg \max_{i \in \{1, 2\}} \left(FRR_i + C \times \sqrt{\frac{2 \times \ln \sum_{j=1}^2 n_j}{n_i}} \right) \quad (15)$$

式(15)中 n_i 是最近 W 次选择中算子 op_i 被选择的次数; C 是比例因子。

3.4 个体编码

为了解决本文多目标分配问题,需要对 MOOA/D-CS 算法中的个体进行编码,从而建立起实际分配方案与算法决策变量的映射关系。本文中的问题是带约束的 MOP,虽然目前有很多处理带约束的 MOP 的算法,但是这些算法各有局限^[16,17]。因此,本文考虑通过编码将约束问题转化为无约束问题。

本文采取实数编码方式。种群个体的维数 D 等于 UAV 数目 L ,个体的每一维由 $[0,1]$ 间的实数表示。个体序号表示 UAV 的编码,各维实数的大小关系和相对位置共同确定攻击目标编码。下面以 6 架 UAV 4 个目标为例进行说明,如图 1 所示。

① 对各维实数按升序排列,排列的前 4 个序号分别

对应 4 个目标的编码,垂直向下箭头显示了对应关系;② 将排列序号为 4—6 对应的实数和实数 1 标记在区间 $[0.45,1]$ 上;③ 将区间 $[0,1]$ 按照 $1/M$ (M 表示攻击目标数) 等距划分,并将间距按顺序编号。将区间 $[0.45,1]$ 映射到区间 $[0,1]$,并确定序号 5—6 对应的实数落在新区间的编号。例如序号 5 对应的实数 0.69 映射到区间 $[0,1]$ 的实数为 $\frac{1}{(1-0.45)} \times (0.69 - 0.45) = 0.3818$,落在第 2 个间隔中,那么 2 即为目标的编号。同理可确定序号 6 对应的实数 0.86 落在第 3 个间隔中,3 即为目标的编号。按以上编码方法,可以确保每一架 UAV 有且仅有一个攻击目标,并且每个目标至少分配一架 UAV。

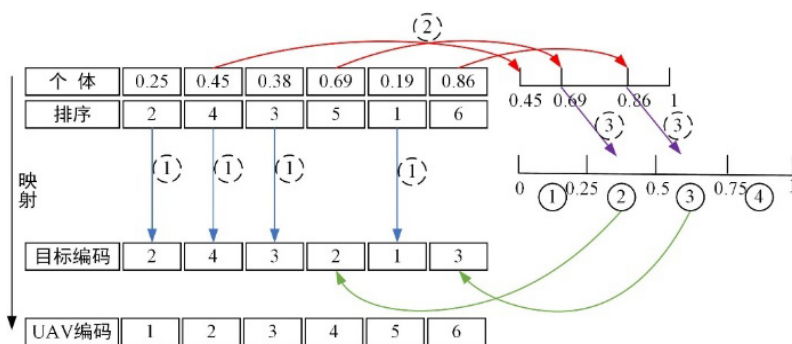


图 1 种群个体与分配方案的映射关系

3.5 算法流程

本文提出的基于 MOOA/D^[18] 框架的 MOOA/D-CS 算法流程可描述如下:

步骤 1 生成 N 个均匀分布的权重向量 $\omega^1, \omega^2, \dots, \omega^N$; 按照欧式距离为每个权重向量 ω^i 确定最近的 T 个权重向量 $\omega^{i_1}, \omega^{i_2}, \dots, \omega^{i_T}$ 作为其邻域, 记作 $B(i) = \{i_1, i_2, \dots, i_T\}$;

步骤 2 按照 3.4 节编码方式随机初始化种群 $\{x_1, x_2, \dots, x_N\}$, 计算理想点 $z^* = [z_1^*, z_2^*, \dots, z_m^*]$;

步骤 3 对于当前个体 x^i 使用式(15) 选择繁殖算子 op_i ;

步骤 4 按照概率 δ 选择邻域或父代作为配对池;

步骤 5 从配对池中随机选择两个不同于 x_i 的个体 x_{r1} 和 x_{r2} 并使用繁殖算子 op_i 生成子代 y ;

步骤 6 评估 y 并更新 z^* ;

步骤 7 按照聚合函数值, 用 y 替换父代种群中 n_r 个个体;

步骤 8 更新 SW 及其他相关参数;

步骤 9 若满足终止条件, 对种群 $\{x_1, x_2, \dots,$

$x_N\}$ 执行非支配排序, 得到并输出 Pareto 最优解集 PS 和 Pareto 最优前沿 PF, 否则转步骤 2;

步骤 10 按照一定的决策准则从 PF 中选择最优解, 并按照该最优解确定规划方案。

4 仿真实验

为了检验 MOOA/D-CS 的性能和求解多 UAV 协同任务分配问题的有效性, 本文给出了两个仿真实验。实验一: MOOA/D-CS 与 MOOA/D 变体在标准测试函数上进行对比分析。实验二: MOOA/D-CS 和对比算法在一个多 UAV 协同任务分配问题实例上对比分析。

4.1 MOOA/D-CS 算法性能实验分析

4.1.1 测试函数与性能指标

为了验证 MOOA/D-CS 算法的性能, 选择 WFG^[19] 标准测试函数集进行实验。WFG 测试函数的位置相关决策变量和距离相关决策变量分别设置为 8 和 2。选择 IGD(Inverted Generational

Distance)^[20] 和 HV(Hyper Volume)^[21] 作为评价指标,两个评价指标均能同时衡量算法的收敛性和多样性。

IGD: PF^* 是一组沿着真实 PF 均匀采样的 Pareto 最优解, PF 是算法获取的 Pareto 最优解集。那么 IGD 可按式定义^[22]:

$$IGD(PF, PF^*) = \frac{\sum_{v \in PF^*} d(v, PF)}{|PF^*|} \quad (16)$$

式(16)中 $d(v, PF)$ 表示解 v 到 PF 中的解的最小欧式距离; $|PF^*|$ 是 PF^* 的基。IGD 值越小, PF 越接近真实值。实验中,选择 1 000 个均匀分布在真实 PF 上的点组成 PF^* 。

HV: PF 是算法获取的 Pareto 最优解集 $z^* = (z_1^*, z_2^*, \dots, z_m^*)^T$ 是最差点(z^* 每一维的值均大于或等于 PF 上的最大值)。HV 是由 PF 中的点与 z^* 在目标空间中围成的超立方体的体积。

$$HV(PF) = volume(\bigcup_{z \in PF} [z_1, z_1^*] \times \dots \times [z_m, z_m^*]) \quad (17)$$

HV 越大 PF 越接近真实值。实验中,最差点 z^* 设置为(1.0, ..., 1.0)。

4.1.2 实验参数设置

实验中,选择 MOEA/D-SBX^[23] 和 MOEA/D-

DE^[16] 进行对比分析。算法参数设置见表 1。其中 η_c 和 p_c 分别是模拟二项式交叉的分布指数和交叉概率;多项式变异算子的分布指数和变异率分别为 η_m 和 p_m ; F 是 DE 算子的比例因子; CR 是 DE 算子的交叉常量; T 表示邻域大小; n_r 表示可被子代替换的父代的最大数目; 概率 δ 控制父代个体来自邻域或来自整个种群的比例; 参数 W, C 在 3.3 节已给出。此外,种群的规模设置为 $N = 100$, 最大函数评价次数设置为 25 000。

表 1 算法参数设置

MOEA/D-SBX	$p_c = 1, \eta_c = 20, T = 0.1 \times N, n_r = \infty, \delta = 1$
MOEA/D-DE	$F = 0.5, CR = 1.0, p_m = 1/n, \eta_m = 20, T = 0.1 \times N, n_r = 2, \delta = 0.9$
MOEA/D-CS	$\alpha_0 = 1, \beta = 1.5, p_a = 0.75, p_m = 1/n, \eta_m = 20, T = 0.1 \times N, n_r = 2, \delta = 0.9, W = 0.5 \times N, C = 1.0$

为了减少随机性的影响,每个测试问题的结果均是算法独立运行 30 次获得的结果的平均值,并给出显著性水平 0.05 的 Wilcoxon 秩和检验。

表 2 给出了有关算法关于 IGD 和 HV 的平均值和标准差,并用阴影突出显示 IGD 和 HV 的最佳结果。

表 2 算法在 WFG 测试函数上关于 IGD 和 HV 的对比结果

问题/算法	IGD			HV		
	MOEA/D-SBX	MOEA/D-DE	MOEA/D-CS	MOEA/D-SBX	MOEA/D-DE	MOEA/D-CS
WFG1 Mean	4.63E-01-	2.61E-02-	1.23E-02	4.78E-01-	6.90E-01-	6.96E-01
std	3.91E-01	2.55E-02	9.90E-04	1.78E-01	9.46E-03	3.20E-04
WFG2 Mean	6.50E-01-	3.46E-01-	1.56E-01	5.41E-01-	5.90E-01-	6.19E-01
std	1.64E-01	2.30E-01	9.00E-02	2.82E-02	3.58E-02	9.13E-03
WFG3 Mean	1.94E-02-	1.13E-02+	1.14E-02	5.75E-01-	5.82E-01+	5.82E-01
std	7.12E-03	6.34E-05	1.03E-04	4.01E-03	9.26E-05	1.20E-04
WFG4 Mean	1.38E-02+	1.77E-02≈	1.55E-02	3.44E-01+	3.40E-01≈	3.42E-01
std	1.99E-03	3.16E-03	1.90E-03	1.15E-03	1.75E-03	1.49E-03
WFG5 Mean	7.13E-02≈	7.11E-02≈	6.74E-02	3.06E-01≈	3.06E-01≈	3.08E-01
std	1.39E-03	6.07E-04	9.27E-03	6.96E-04	3.60E-04	4.96E-03
WFG6 Mean	3.42E-02-	3.27E-02-	2.28E-02	3.33E-01-	3.35E-01-	3.41E-01
std	1.87E-02	9.95E-03	1.57E-02	5.56E-03	5.04E-03	7.43E-03
WFG7 Mean	1.78E-01-	8.25E-02-	3.55E-02	3.05E-01-	3.25E-01-	3.36E-01
std	7.74E-02	7.77E-02	1.81E-02	9.90E-03	1.54E-02	8.21E-03
WFG8 Mean	2.51E-01-	5.20E-02-	3.46E-02	2.74E-01-	3.25E-01-	3.32E-01
std	1.99E-01	6.61E-02	5.50E-03	3.97E-02	2.76E-02	1.96E-03
WFG9 Mean	3.97E-02-	2.59E-02-	2.27E-02	3.31E-01-	3.38E-01-	3.39E-01
std	1.81E-02	3.53E-03	2.34E-03	7.22E-03	1.10E-03	7.36E-04
-/+ / ≈	7/1/1	6/1/2		7/1/1	6/1/2	

-/+ / ≈ 表示在 5% 显著水平下 Wilcoxon 秩和检验下,对应算法的性能明显劣于/相当于/优于 MOEA/D-CS 算法

从表 2 可以看出,MOOA/D-CS 在 7/1/1 个函数上明显优于/相当于/劣于 MOEA/D-SBX; 在 6/2/1 个函数上明显优于/相当于/劣于 MOEA/D-DE。以上结果表明,MOOA/D-CS 算法性能优于 MOEA/D-SBX 和 MOEA/D-DE。

4.2 UAV 协同任务分配实验

假设在 $100\text{ km} \times 100\text{ km}$ 的战场环境中,有 4 个 UAV 基地,每个基地内均有 3 架 UAV,有 8 个待攻击目标和 4 个威胁源。威胁源对一定范围内的目标具有协防功能,一旦目标受到 UAV 攻击,受攻击目标和威胁源可对发动攻击的 UAV 进行反击。表 3、表 4 和表 5 分别给出了 UAV、受攻击目标和威胁源的相关信息。表 6 给出了威胁源与目标的协防关系。现在基地内 12 架 UAV 需要对 8 个目标发起攻击,要求每架 UAV 只攻击一个目标,每个目标至少分配一架 UAV。为 UAV 规划出合理的行动方案,使得总的攻击收益最大,威胁代价最小。

表 3 无人机信息表

基地编号	无人机编号	位置	价值	命中概率
1	1	(5, 5)	0.95	0.85
1	2	(5, 5)	0.8	0.75
1	3	(5, 5)	1	0.35
2	4	(5, 95)	0.8	0.7
2	5	(5, 95)	0.9	0.8
2	6	(5, 95)	1	0.45
3	7	(95, 95)	0.9	0.8
3	8	(95, 95)	0.75	0.7
3	9	(95, 95)	1	0.25
4	10	(95, 5)	0.85	0.75
4	11	(95, 5)	0.65	0.55
4	12	(95, 5)	1	0.55

表 4 目标信息表

目标编号	位置	价值	命中概率
1	(52, 62)	0.85	0.40
2	(33, 30)	0.75	0.35
3	(40, 45)	0.80	0.30
4	(15, 55)	0.80	0.35
5	(65, 35)	0.95	0.45
6	(85, 40)	0.75	0.25
7	(29, 78)	0.78	0.35
8	(75, 65)	0.87	0.47

表 5 威胁源信息表

威胁编号	位置	命中概率
1	(50, 30)	0.30
2	(70, 50)	0.45
3	(30, 60)	0.40
4	(20, 40)	0.20

表 6 协防关系表

威胁	目标							
	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
H_1	0	1	1	0	1	0	0	0
H_2	1	0	0	0	1	1	0	1
H_3	1	0	1	1	0	0	1	0
H_4	0	1	1	1	0	0	0	0

为了检验 MOOA/D-CS 求解多 UAV 协同任务分配问题的有效性,选择 MOEA/D-SBX 和 MOEA/D-DE 算法与 MOOA/D-CS 进行对比实验。参数设置与 4.1 节中相同。

4.2.1 性能比较

表 7 给出了 3 种算法求解该 UAV 协同任务分配问题的 HV 值的均值和方差。

表 7 不同算法求解结果对比

	MOEA/D-SBX	MOEA/D-DE	MOOA/D-CS
Mean	3.73E-01-	3.78E-01-	3.83E-01
std	7.44E-03	5.06E-03	2.25E-03

从表 7 可以看出,MOOA/D-CS 的性能优于另外两种算法,并且 Wilcoxon 秩和检验表明这种差距具有统计学意义。图 2 给出了 HV 的进化曲线图,图中横坐标是函数评价次数,纵坐标是对 HV 取 10 为底的对数的值。从图 2 可以看出,MOOA/D-CS 明显优于另外两种算法。

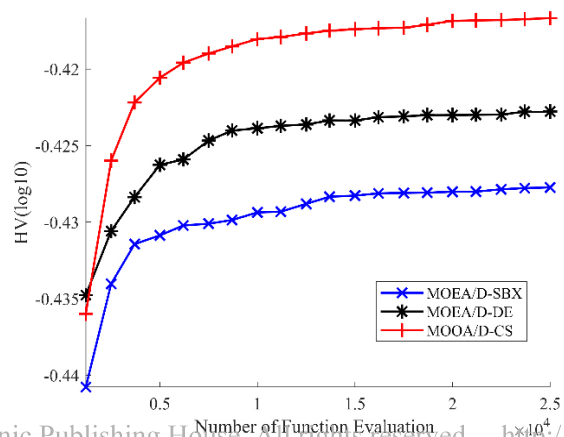


图 2 不同算法求解结果进化曲线

4.2.2 方案比较

决策者在无偏好时,选择 ξ 值最大的个体作为决策方案。表 8 给出了 4 种方案的目标值和 ξ 值。从表 8 可以发现: 本文提出算法的方案 2 和方案 4 优于另外两种算法的最优方案(MOEA/D - SBX 算法的方案 3 和 MOEA/D - DE 算法的方案 2)。在 ξ 取值为 1.768 8(方案 2) 的条件下,对应的分配结果如图 3 所示。

表 8 不同方案的目标值和 ξ 值

参数		方案 1	方案 2	方案 3	方案 4
MOEA/D - SBX	f_1	0.289 5	0.271 1	0.245 8	0.284 0
	f_2	0.567 0	0.568 6	0.584 3	0.576 1
	ξ	1.495 3	1.591 2	1.691 1	1.492 9
MOEA/D - DE	f_1	0.269 3	0.250 8	0.265 6	0.245 5
	f_2	0.563 2	0.570 7	0.582 4	0.581 5
	ξ	1.621 9	1.711 4	1.572 4	1.704 8
MOOA/D - CS	f_1	0.245 8	0.244 8	0.265 1	0.244 8
	f_2	0.584 3	0.567 1	0.570 4	0.567 1
	ξ	1.691 1	1.768 8	1.620 5	1.768 8

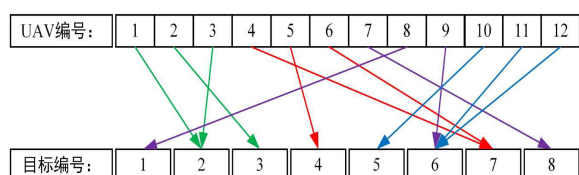


图 3 MOOA/D - CS 算法方案 2 分配结果

决策者的偏好实际上是决策者对不同目标权重的选择。在 MOOA/D 框架下,可以方便地按照决策者的偏好确定方案。决策者的偏好可以按照 $(\lambda_1, \lambda_2, \dots, \lambda_m)$ 的形式给出, λ_i 表示第 i 个目标的权重 $0 \leq \lambda_i \leq 1$ 且 $\sum_{i=1}^m \lambda_i = 1$ 。

$(\lambda_1, \lambda_2)^T = (0, 1)^T$ 和 $(\lambda_1, \lambda_2)^T = (1, 0)^T$ 是两种极端偏好。 $(\lambda_1, \lambda_2)^T = (0, 1)^T$ 表示决策者完全不在意威胁, 希望取得最大的收益; $(\lambda_2, \lambda_1)^T = (1, 0)^T$ 表示决策者希望无人机执行打击任务时能最大限度避免损毁。在极端偏好条件下, MOOA/D - CS 的一个优化结果对应的目标值为: $(\lambda_1, \lambda_2)^T = (0, 1)^T$ 时, $(f_1, f_2)^T = (0.380 7, 0.441 6)^T$; $(\lambda_1, \lambda_2)^T = (1, 0)^T$ 时, $(f_1, f_2)^T = (0.243 8, 0.590 6)^T$ 。对应的分配方案如图 4 和图 5 所示。

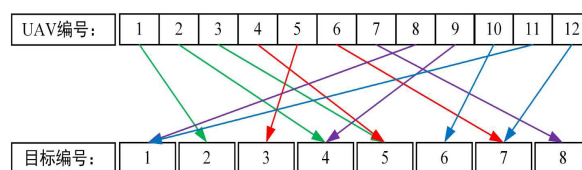


图 4 收益最大偏好对应的分配结果

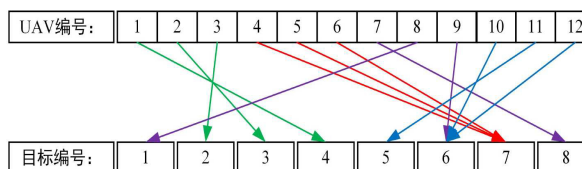


图 5 损失最小偏好对应的分配结果

从图 4 可以看出, 在偏好为 $(\lambda_1, \lambda_2)^T = (0, 1)^T$ 时, 为了获得更大的收益, UAV 攻击距离更远而价值更高的目标, 获得的收益为 0.558 4, 付出的代价为 0.380 7。在偏好为 $(\lambda_1, \lambda_2)^T = (1, 0)^T$ 时, 为了规避威胁, UAV 选择距离较近或受到较少保护的目标, 例如目标 6 和目标 7, 此决策下, 获得的收益为 0.409 4, 付出的代价为 0.243 8。决策者可以全面比较多个分配方案, 做出科学的决策。

5 结束语

本文在分析多 UAV 协同任务分配问题的基础上, 建立了多 UAV 协同任务分配问题模型。为了求解该问题, 提出了 MOOA/D-CS。该算法设计了两种新的繁殖算子, 新的繁殖算子来源于 CS 算法的 Lévy flights 随机游走和偏好随机游走操作, 并采用了一种基于多臂赌博机的自适应算子选择策略。在标准测试函数集上的实验验证了 MOOA/D-CS 算法的良好性能。提出了一种新的编码方式, 将带约束的 MOP 转化为无约束 MOP, 降低了问题的求解难度。通过多 UAV 协同任务分配问题的仿真实验, 验证了模型和 MOOA/D-CS 的有效性, 为多 UAV 协同任务分配智能决策提供了方法支持。

参考文献

- [1] 陈志旺, 夏顺, 李建雄, 等. 考虑分配次序的无人机协同目标分配建模与遗传算法求解[J]. 控制理论与应用, 2019, 36(7):

- 1072–1082.
- [2] 王钦钊,程金勇,李小龙.多无人机协同任务规划方法[J].火力与指挥控制,2018,43(3):86–89,94.
- [3] 蒋硕,袁小平.改进 PSO 算法在多无人机协同任务分配中的应用[J].计算机应用研究,2019,36(11):3344–3347,3360.
- [4] 董海霞,邹杰.基于 AMPSO 算法的无人机任务分配问题研究[J].电光与控制,2018,25(2):28–32.
- [5] 向庭立,王红军,史英春.区域覆盖的多机协同探测任务分配策略[J].空军工程大学学报(自然科学版),2019,20(6):33–38,71.
- [6] 许可,宫华,秦新立,等.基于分布式拍卖算法的多无人机分组任务分配[J].信息与控制,2018,47(3):341–346.
- [7] MA X, YU Y, LI X, et al. A survey of weight vector adjustment methods for decomposition-based multiobjective evolutionary algorithms[J]. IEEE Transactions on Evolutionary Computation, 2020, 24(4): 634–649.
- [8] ZALIK K R, ZALIK B. Multi-objective evolutionary algorithm using problem-specific genetic operators for community detection in networks [J]. Neural Comput Appl, 2018, 30(9): 2907–2920.
- [9] WU M, KE L, KWONG S, et al. Learning to decompose: A paradigm for decomposition-based multiobjective optimization [J]. IEEE Transactions on Evolutionary Computation, 2019, 23(3): 376–390.
- [10] ISHIBUCHI H, AKEDO N, NOJIMA Y. Relation between neighborhood size and MOEA/D performance on many-objective problems [C]//International Conference on Evolutionary Multi-Criterion Optimization. Springer Berlin Heidelberg, 2013.
- [11] WANG Z, ZHANG Q, GONG M, et al. A replacement strategy for balancing convergence and diversity in MOEA/D [C]//IEEE Congr Evol Comput, 2014.
- [12] 赵辉,李牧东,韩统,等.基于多目标 MQABC 算法的无人机协同任务分配[J].华中科技大学学报(自然科学版),2016,44(3):121–126.
- [13] YANG X S, DEB S. Engineering optimisation by cuckoo search [J]. International Journal of Mathematical Modelling and Numerical Optimisation, 2010, 1(4): 330–343.
- [14] DEB K, GOYAL M. A combined genetic adaptive search (GeneAS) for engineering design [J]. Computer Science and Informatics, 1996, 26(4): 30–45.
- [15] LI K, FIALHO A, KWONG S, et al. Adaptive operator selection with bandits for a multiobjective evolutionary algorithm based on decomposition [J]. IEEE Transactions on Evolutionary Computation, 2014, 18(1): 114–130.
- [16] FAN Z, LI W J, CAI X Y, et al. Push and pull search for solving constrained multi-objective optimization problems [J]. Swarm and Evolutionary Computation, 2019, 44(6): 65–79.
- [17] CAI X Y, MEI Z W, FAN Z, et al. A constrained decomposition approach with grids for evolutionary multiobjective optimization [J]. IEEE Transactions on Evolutionary Computation, 2018, 22(4): 564–577.
- [18] LI H, ZHANG Q. Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II [J]. IEEE Transactions on Evolutionary Computation, 2009, 13(2): 284–302.
- [19] TIAN Y, CHENG R, ZHANG X, et al. PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [J]. IEEE Computational Intelligence Magazine, 2017, 12(4): 73–87.
- [20] HUBAND S, HINGSTON P, BARONE L, et al. A review of multiobjective test problems and a scalable test problem toolkit [J]. IEEE Transactions on Evolutionary Computation, 2006, 10(5): 477–506.
- [21] ZITZLER E, THIELE L, LAUMANN S M, et al. Performance assessment of multiobjective optimizers: An analysis and review [J]. IEEE Transactions on Evolutionary Computation, 2003, 7(2): 117–132.
- [22] ZITZLER E, THIELE L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach [J]. IEEE Transactions on Evolutionary Computation, 1999, 3(4): 257–271.
- [23] HANG Q F, LI H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition [J]. IEEE Transactions on Evolutionary Computation, 2007, 11(6): 712–731.

(责任编辑: 刘书江)