

基于改进粒子群优化算法的多无人机与水下潜航器协同侦察路径规划

张文斌¹ 李 杰² 夏元清³

ZHANG Wenbin LI Jie XIA Yuanqing

摘 要

多无人机和水下潜航器之间的协同覆盖路径规划,是军事侦察、海洋监测等领域的一项关键技术。传统“先分区、后规划”方法存在把任务划分与路径规划割裂的问题,致使出现路径冗余和负载不均等情况。基于此,文章提出一种基于改进粒子群优化的协同路径规划方法。首先,建立协同优化框架,实现区域划分与路径规划统一;然后,设计混合编码方式,并建立多目标适应度函数,该函数要全面考虑任务完成时间、覆盖效果、负载均衡等多方面因素;最后,引入路径连续性增强机制,与基于图论的修复策略结合。这些措施能提高解的质量和可行性。仿真实验结果显示,将随机搜索算法当作参照对象进行对比,改进粒子群优化算法驱动的协同路径规划方法,在覆盖率、任务完成时间、负载均衡度、路径平滑性等指标方面,都呈现出较好优势,为异构无人系统协同侦察提供了工程方面的解决办法,具备计算高效、解质量良好的特性。

关键词

覆盖路径规划;协同侦察;粒子群优化;无人机;水下潜航器

doi: 10.3969/j.issn.1672-9528.2026.06.014

0 引言

近年来,伴随海洋资源勘探、水下安全监测^[1]等方面应用需求持续提高,多无人系统协同作业^[2-3]凭借其在任务执行范围、效率、鲁棒性等方面的优越特性,已然成为达成海洋环境立体化侦察的关键技术,并在军事侦察^[4-5]、海洋监测^[6]等诸多领域展现出重要价值。由多无人机^[7](unmanned aerial vehicle, UAV)与自主水下潜航器^[8](autonomous under water vehicle, AUV)共同构成的异构协同^[9]系统,能够达成空海一体化的侦察工作,有效提高对目标区域全域立体覆盖^[10-11]的效率,具有重要的理论和应用意义。

协同路径规划^[12]对于发挥空海协同优势而言是关键所在,它的质量会直接对侦察范围、效率、鲁棒性产生决定性作用。因为无人机与水下潜航器^[13]在运动能力、通信、续航等方面有着本质上的不同,所以路径规划需要同时满足覆盖完整性、时间协同性、负载均衡、各类动力学约束等条件,这就使得问题展现出高维度、强耦合与非线性等复杂特性,传统规划方法很难有效应对这种情况。

对于之前提到的这些问题,现有的方法大致可以分为

以下几类。传统优化方法,如动态规划^[14],它的计算复杂度比较高,因此很难应用到大规模场景中;在连续空间里,基于图论的方法,比如A*算法^[15],其适应性受到一定限制;基于市场机制的方法,如拍卖算法^[16],难以保证路径达到全局最优;元启发式算法,如随机启发式算法^[17],存在收敛速度较慢的情况;基于机器学习的方法,例如强化学习^[18],需要大量训练数据,并且其可解释性相对较弱。和这些方法相比,群体智能优化算法,如遗传算法^[19]、蚁群算法^[20-21]等,由于具备强大的全局搜索能力、对复杂约束的处理潜力,所以被广泛采用。粒子群优化(particle swarm optimization, PSO)算法^[22]具有结构比较简单、收敛速度比较快的特点,在处理连续优化问题的时候呈现出较好的性能。正是因为这样,该算法被认为适宜用于协同路径规划中的高维目标优化问题。

传统PSO算法在面对路径规划这类离散组合问题时,存在局限性。其连续迭代机制很难直接生成可行的路径序列,容易出现无效解。当算法运行到后期阶段时,多样性会逐渐衰减,进而容易陷入局部最优状态。该算法对于路径约束,比如连通性、平滑性、续航能力等方面,处理能力也不足。

针对此情况,本文给出了一种改进的粒子群算法(improved particle swarm optimization, IPSO)优化算法^[23],

1. 安徽建筑大学 安徽合肥 230601

2. 北京理工大学长三角研究院 浙江嘉兴 314019

3. 中原工学院 河南郑州 450007

该算法的创新之处主要体现在以下几个方面:

(1) 此算法设计出了混合编码、连续性感知的更新机制, 达成了区域划分与路径序列的统一表示;

(2) 建立综合考虑覆盖效率、时间协同、负载均衡的多目标适应度函数;

(3) 在算法当中引入路径连续性增强策略、基于图论的修复机制, 借助这样的方式来提高解的质量、可行性;

(4) 算法将局部搜索以及内层优化相互融合起来, 如此一来, 路径的平滑性也获得了进一步的改进。凭借上述这些策略, 该算法在复杂的异构协同场景当中开展路径规划工作时, 其性能得以较为系统的提升。

1 系统建模与区域分解方法

1.1 环境建模与网格分解

对于无人机与水下潜航器进行的协同侦察任务, 本节采用网格分解法^[24]进行环境建模。此方法把连续区域变成均匀网格, 之后将复杂路径规划转变为可计算的图搜索问题, 提高算法效率。优势为能给异构平台提供统一的拓扑表达, 并且通过调节网格粒度, 能够灵活平衡计算复杂度和规划精度。

(1) 任务区域定义

把任务区域设定成一个凸多边形, 称其为区域 A 。对这个区域进行离散化处理, 让它变成多个大小一样的正方形栅格。每一个这样的栅格就是一个单位区域。对于上述由栅格构成的顶点集合定义为:

$$P = \{p_1, p_2, \dots, p_n\}, p_i = (x_i, y_i) \in R^2 \quad (1)$$

通过网格分解生成的节点集合为:

$$N = \{n_1, n_2, \dots, n_m\}, n_j = (x_j, y_j) \in A \quad (2)$$

式中: m 为总节点数, 由网格大小 g 决定:

$$m = f(A, g) = \left\lceil \frac{\text{Area}(A)}{g^2} \right\rceil \quad (3)$$

(2) 边连接关系建模

节点之间的连接关系通过边集合 E 表示:

$$E = \{(i, j, d_{ij}) | n_i, n_j \in N, \|n_i - n_j\| \leq \sqrt{2}g \cdot \alpha\} \quad (4)$$

式中: d_{ij} 表示欧几里得距离, $\alpha > 1$ 表示连接容差系数, 通常取 1.01~1.1, 以此保证相邻网格点能够连通。

1.2 智能体建模

在协同侦察任务当中, 每个智能体所具备的运动能力、侦察特性都是不一样的。本文针对无人机和水下潜航器进行统一建模。

(1) 智能体基本属性

设系统中存在多个智能体。将这些智能体定义为集合:

$$U = \{u_1, u_2, \dots, u_K\} \quad (5)$$

(2) 路径表示与时间计算

智能体 U_k 的路径表示为有序节点序列:

$$\text{Path}_k = [s_{k1}, s_{k2}, \dots, s_{kL_k}], s_{ki} \in N \quad (6)$$

式中: L_k 为路径 Path_k 所包含的节点总数。

路径总时间 T_k 由移动时间和转向时间组成:

$$T_k = T_k^{\text{move}} + T_k^{\text{turn}} \quad (7)$$

移动时间计算为路径中所有线段移动时间的总和:

$$T_k^{\text{move}} = \sum_{i=1}^{L_k-1} \frac{d(s_{ki}, s_{k(i+1)})}{v_k} \quad (8)$$

式中: $d(s_{ki}, s_{k(i+1)})$ 为连续节点 s_{ki} 与 $s_{k(i+1)}$ 之间的欧几里得距离。

转向时间计算为路径中所有转向点处转向时间的总和:

$$T_k^{\text{turn}} = \sum_{i=2}^{L_k-1} \frac{\theta(s_{k(i-1)}, s_{ki}, s_{k(i+1)})}{\omega_k} \quad (9)$$

转向角度 θ_{ki} 通过向量叉积和点积计算:

$$\theta_{ki} = \arccos \left(\frac{\vec{V}_{i-1,i} \cdot \vec{V}_{i,i+1}}{\|\vec{V}_{i-1,i}\| \|\vec{V}_{i,i+1}\|} \right) \quad (10)$$

式中: $\vec{V}_{i-1,i}$ 和 $\vec{V}_{i,i+1}$ 表示从节点 $s_{k(i-1)}$ 到 s_{ki} 以及从 s_{ki} 到 $s_{k(i+1)}$ 的向量。该模型可以精确呈现智能体在实际运动过程中的时间消耗情形, 这种时间消耗源自运动方向的变化。

1.3 协同侦察目标建模

协同侦察任务的优化目标具有多维特性, 故而要全面考量覆盖完整性、任务效率、资源均衡利用等诸多方面。

(1) 覆盖要求

全覆盖约束有着这样的具体要求: 每个节点都需要至少被一个智能体来访问:

$$\bigcup_{k=1}^K \{s_{k1}, s_{k2}, \dots, s_{kL_k}\} = N \quad (11)$$

重复覆盖惩罚有着如下具体定义: 虽说允许出现重复覆盖的情况, 然而这种情况要尽可能地加以减少:

$$R = \frac{\sum_{k=1}^K L_k - |\bigcup_{k=1}^K \text{Path}_k|}{\sum_{k=1}^K L_k} \quad (12)$$

(2) 路径有效性约束

路径必须由有效的边连接组成:

$$\forall i \in [1, L_k - 1], (s_{ki}, s_{k(i+1)}) \in E \quad (13)$$

(3) 续航约束

每个智能体的路径时间不能超过其最大续航:

$$T_k \leq T_k^{\text{max}}, \forall k \in [1, K] \quad (14)$$

(4) 优化目标函数

综合考虑任务效率、覆盖质量和资源利用, 设计多目标优化函数:

$$\min F = w_c(1-C)^2 + w_t T_{\text{max}} + w_b B + w_s S \quad (15)$$

式中: C 为覆盖率, T 为最大完成时间, B 为负载均衡度, 基于路径长度标准差计算, S 为路径平滑度惩罚, w_c 、 w_l 、 w_b 、 w_s 为权重系数。

2 问题复杂性与求解策略分析

本章围绕多智能体协同覆盖路径规划问题的难点^[25]展开分析, 主要从两个方面考虑, 即计算复杂性和约束耦合性。在剖析这两方面之后, 会指出当下方法存在的不足之处, 并且阐述像粒子群算法这类方法的适用情况。会给出本文运用的分层求解框架。

2.1 问题复杂性分析

多智能体协同覆盖路径规划问题本质上属于复杂的组合优化问题, 其复杂性主要表现在计算复杂度、约束耦合性两个方面。

(1) 计算复杂度极高

在系统当中存在 N 个离散节点以及 M 个智能体的情况下, 其潜在的路径分配方案的数量会变得极为庞大, 这种状况也就成为在求解过程中最为关键的挑战所在。

(2) 多重约束紧密耦合

本问题涉及几何约束、时间约束、动力学约束、协同约束等诸多方面。这些约束相互关联, 呈现出紧密的耦合特性。此种情形使得求解难度加大。

2.2 求解策略与框架设计

整合上述分析内容之后, 本文给出了一个分层优化框架, 此框架把全局搜索和局部精细化结合起来。

(1) 外层优化(全局搜索)

运用改进后的 PSO 算法进行全局优化工作, 着重针对节点到智能体的分配情况、路径的大致走向确定问题、负载均衡的调节方面予以处理。

(2) 内层优化(局部精细化)

结合局部搜索策略进行优化工作, 其中包括路径平滑性、转向角度最小化、冗余节点消除等方面。

(3) 修复机制

设计专门的修复策略。该策略主要围绕三个要点来展开, 其中要点一是要保证实现百分之百的覆盖率, 要点二是要确保路径具备有效性, 要点三是要满足续航约束条件。

3 改进粒子群优化算法设计

3.1 算法整体框架

改进后的粒子群优化算法执行流程是一个循环过程。这个循环过程包括初始化、迭代优化、局部搜索、解决方案修复等多个环节。其整体框架如图 1 所示。

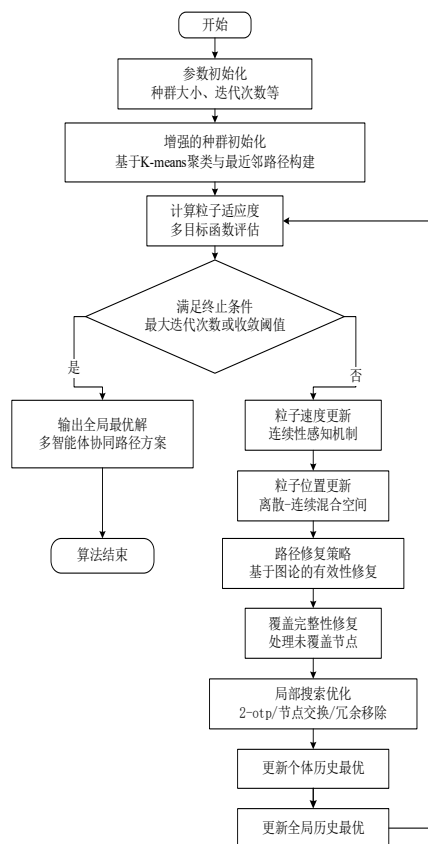


图 1 改进粒子群算法流程图

此框架的设计依照“初始化—迭代—修复”循环优化思路来进行。流程一开始是增强的种群初始化模块。这个模块借助 K-means 聚类、最近邻方法来生成高质量的初始路径。之后, 算法进入到核心的迭代循环阶段。在每一个循环里, 粒子先进行连续性感知的速度更新与位置更新。接着, 算法进入一个关键的修复与优化层。在这个层中, 依次进行基于图论的有效性修复、覆盖完整性修复、局部搜索操作。这些操作是为了保证解能够满足所有约束条件, 并且进一步提高解的质量。循环的最后一步, 算法更新历史最优解, 然后进入下一轮迭代。上述过程不断持续, 直到满足终止条件。满足终止条件后, 算法输出最终的路径方案。此结构把全局搜索和局部修复紧密结合起来, 用来系统性地解决复杂的协同路径规划问题。

3.2 粒子编码与解码策略

因为路径规划问题有离散和连续特性相互混合的情形, 所以本文设计了混合编码方案。

(1) 混合编码设计

每个粒子代表一个完整的解决方案。该方案即所有智能体的路径集合:

$$P_i = \{\text{Path}_1, \text{Path}_2, \dots, \text{Path}_k\} \quad (16)$$

式中: $\text{Path}_k = [s_{k1}, s_{k2}, \dots, s_{kl_k}]$ 为智能体 k 的路径节点序列,

$s_{ij} \in V$ 为任务区域中的节点, L_k 为该路径的节点总数。

粒子的速度 V_i 编码为路径节点被调整的概率为:

$$\text{Vel}_k = [p_{k1}, p_{k2}, \dots, p_{kL_k}] \quad (17)$$

式中: $p_{ki} \in [0, 1]$ 表示智能体 k 的路径中第 i 个节点被调整的概率。

(2) 解码策略

解码过程可以把粒子的连续位置向量转化成所有智能体的可行路径集合。此过程分成以下两步:

节点分配: 借助 Softmax 函数, 依据粒子位置来计算每个节点针对各个智能体的归属概率。接着, 把节点分配给概率处于最高状态的智能体, 以此达成任务区域的柔性划分。

路径生成: 针对为每个智能体所分配到的节点集来建立访问序列。具体运用的是连续性优先的最近邻法。此方法从随机起点起始, 循环执行如下操作: 把与当前路径末端欧氏距离最近且连通的未访问节点添加至路径当中。要是两个节点间没有直接有效边存在, 那就去搜寻最短连通路并将在中间节点插入进去。上述操作是为了保证路径具备物理可达性。

3.3 适应度函数设计

适应度函数是引导算法搜索方向的核心。本文定义如下多目标适应度函数:

$$\text{Fitness} = w_c(1-C)^2 + w_t T_{\max} + w_e E + w_b B + w_s S + w_v V \quad (18)$$

式中: w_c 、 w_t 、 w_e 、 w_b 、 w_s 和 w_v 为各分项权重系数, 用于平衡不同优化目标的重要性。

覆盖率为:

$$C = \frac{|\bigcup_{k=1}^K \text{Path}_k|}{|N|} \quad (19)$$

最大完成时间为:

$$T_{\max} = \max_{k=1}^K T_k \quad (20)$$

续航约束违反惩罚:

$$E = \sum_{k=1}^K \max(0, T_k - T_k^{\max}) \quad (21)$$

负载均衡为:

$$B = \frac{\sigma(L)}{\mu(L)}, L = [L_1, L_2, \dots, L_K] \quad (22)$$

平滑度为:

$$S = \frac{1}{K} \sum_{k=1}^K \frac{1}{\theta_k} \quad (23)$$

有效性为:

$$V = \sum_{k=1}^K \mathbb{I}[\text{Path}_k \text{ 无效}] \quad (24)$$

3.4 改进的粒子更新机制

粒子更新是 PSO 算法的核心操作, 本节引入连续性感知

机制, 使粒子在搜索过程中不仅考虑位置优劣, 还关注路径的连续性与平滑性, 从而生成更符合实际应用需求的路径。

(1) 连续性感知速度更新

传统 PSO 速度更新公式:

$$v_{id}(t+1) = w \cdot v_{id}(t) + c_1 r_1 (p_{id} - x_{id}(t)) + c_2 r_2 (p_{gd} - x_{id}(t)) \quad (25)$$

改进为连续性感知更新:

$$v_{id}(t+1) = w \cdot v_{id}(t) + c_1 r_1 (1 + \Delta C_1)(p_{id} - x_{id}(t)) + c_2 r_2 (1 + \Delta C_1)(p_{gd} - x_{id}(t)) + \alpha \Delta C \quad (26)$$

式中: ΔC 为连续性感知因子, 将节点替换给路径平滑度和连续性带来的影响进行了量化; w 为惯性权重, c_1 、 c_2 为学习因子, r_1 、 r_2 为 $[0,1]$ 范围内的随机数, α 为连续性感知系数, ΔC_1 为连续性改进度量。

(2) 位置更新与修复

位置更新为:

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (27)$$

位置更新完成以后, 要施行一系列修复策略。一是有效性修复, 也就是检查路径里的无效边并加以修正; 二是连续性修复, 通过插入必要的中间节点来恢复路径的连通性; 三是覆盖修复, 去识别那些还没有被覆盖的节点, 然后将其插入适当的路径当中; 四是平滑度优化, 针对路径里面的尖锐转折做出局部的调整。

3.5 增强的种群初始化

采用以下策略。这些策略用于生成高质量的初始粒子。

(1) 基于区域划分的初始化

运用 K-means 聚类方式。此方法会把节点划分成多个区域。接着依据智能体的速度去分配各个区域内节点的数量。

(2) 连续性路径构建

运用最近邻法来建立连续路径。先从区域中心附近挑选起点。之后不断重复进行如下操作: 挑选距离当前节点最近的未访问节点, 要是两个节点之间没办法直接连接, 那就去搜寻一条最短路径来实现连接。并且, 要考虑到续航方面的约束, 在需要的时候提前结束路径的建立。

3.6 局部搜索与路径优化

在全局搜索的基础之上, 进一步引入局部的精细化操作:

(1) 2-opt 局部优化

对每条路径执行 2-opt 优化。该优化尝试所有可能的边交换。

(2) 节点交换优化

交换相邻节点。通过这一操作改善局部的平滑度。

(3) 冗余节点移除

识别共线的中间节点。随后将这些节点予以移除。

3.7 内层优化（局部精细化）

在通过全局搜索确定了大致的路径分配之后，内层优化便承担起对路径进行局部精细化处理的任务。此处理从以下三个方面着手进一步提高路径质量。路径平滑性优化：运用B样条曲线插值法。该方法针对路径节点实施平滑操作。其目的在于减少不必要的转折点，进而增强路径的连续性和可飞行性。在平滑进程中，要维持路径节点之间的连通性不发生改变。转向角度最小化：采用局部节点重排策略。该策略调整路径中相邻节点的顺序。调整的目的是让相邻线段之间的夹角尽量接近180度。如此一来能够降低智能体在转向时所产生的能量消耗与时间消耗。冗余节点消除：遍历路径节点序列。识别处于同一直线段上的中间节点也就是共线节点。然后将这些节点移除。经过上述一系列操作之后，路径结构能够得以简化，同时路径长度、计算复杂度都会有所降低。

3.8 修复机制

为确保生成的路径满足所有约束条件，设计了三层修复策略。覆盖完整性修复：遍历所有任务节点，若存在未被任何智能体访问的节点，则将其插入距离最近的可行路径中；路径有效性修复：基于图论中的最短路径算法（如Dijkstra算法^[26]），对路径中不连续的边进行修复，确保所有路径均由有效边连接而成；续航约束修复：若某智能体路径总时间超过其最大续航，则通过移除路径末端部分节点、将部分节点重新分配给其他负载较轻的智能体或者重新计算路径时间，直至满足续航限制。

4 仿真实验与结果分析

本章通过系统的仿真实验，对所提出的改进粒子群优化算法进行全面的性能评估。实验从覆盖率、任务完成时间、负载均衡、路径质量及收敛性能等多个维度，将IPSO算法与增强随机搜索启发式算法进行对比分析，验证本文算法的有效性与优越性。

4.1 实验环境与参数设置

4.1.1 硬件与软件环境

此次实验于MATLAB R2024a的环境当中展开。其硬件配置为：Intel i5-13400F处理器、32 GB内存、Windows 10操作系统。所有算法都于同一环境运行，以此保证对比实验具备公平性。

4.1.2 任务场景设置

定义一个五边形任务区域。该区域的顶点坐标集合为：

$$P = \{(0,0),(100,0),(100,100),(50,150),(0,100)\}$$
 (28)

运用网格分解方式，把连续区域转变为由覆盖节点所组成的离散形态。将网格尺寸设定为8.40 m。经过这一过程，

总共产生了180个覆盖节点，并且还建立了643条连接边。区域分解的具体结果、整个过程如图2~3所示。

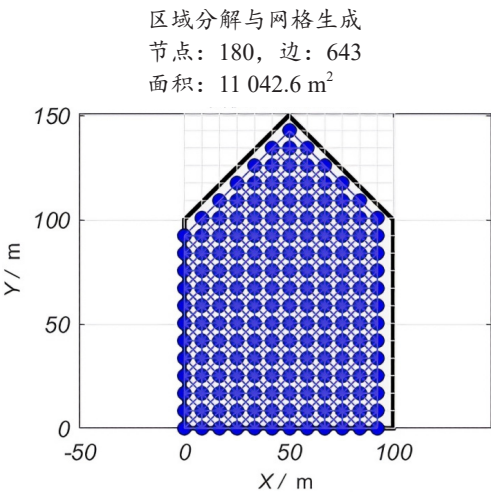


图2 区域分解图

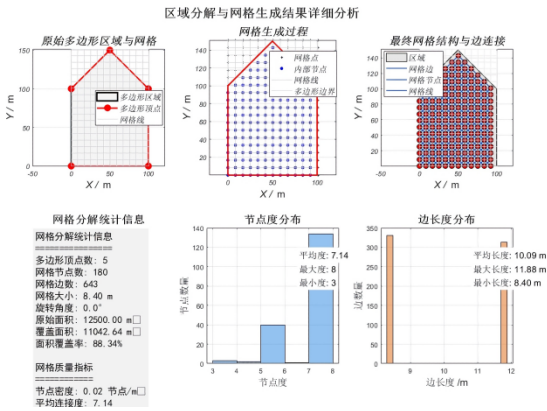


图3 区域分解与网络生成结果详细分析图

4.1.3 智能体参数配置

配置4架无人机和2架水下潜航器组成异构智能体系统，具体参数如表1所示。

表1 智能体参数

智能体	类型	速度 /(m·s ⁻¹)	角速度 /(rad·s ⁻¹)	续航 /s	侦察半径 /m
UAV1	无人机	4.0	0.5	1 800	15
UAV2	无人机	4.0	0.6	1 700	15
UAV3	无人机	5.0	0.7	1 600	15
UAV4	无人机	5.0	0.8	1 500	15
AUV1	水下潜航器	1.5	0.3	3 600	12
AUV2	水下潜航器	1.5	0.3	3 600	12

4.1.4 算法参数设置

IPSO算法参数：种群大小为50；最大迭代次数为100；惯性权重为0.729；学习因子：C₁=1.494，C₂=1.494。

RSH算法参数：最大迭代次数为100；概率乘数为0.1；局部搜索深度为150。

4.2 性能评估指标

为全面评估算法性能,建立以下主要性能指标体系:

覆盖率 (Coverage Rate):

$$C = \frac{N_c}{N_t} \times 100\% \quad (29)$$

式中: N_c 为被覆盖节点数, N_t 为总节点数。

最大完成时间 (Maximum Completion Time):

$$T_{\max} = \max_{k=1,2,\dots,K} T_k \quad (30)$$

式中: T_k 为第 k 个智能体的完成时间。计算时间为 (Computation Time): 算法从开始运行到输出结果的实际计算时间。

负载均衡度为 (Load Balance Degree):

$$B = \frac{\sigma(L)}{\mu(L)}, L = [L_1, L_2, \dots, L_K] \quad (31)$$

式中: σ_L 为路径长度标准差, μ_L 为路径长度均值。

路径平滑度 (Path Smoothness):

$$S = \frac{1}{K} \sum_{k=1}^K \frac{1}{1 + \bar{\theta}_k} \quad (32)$$

式中: $\bar{\theta}$ 为第 k 个智能体路径的平均转向角度。

节点重复率 (Node Repetition Rate):

$$R = \frac{N_r}{N_v} \times 100\% \quad (33)$$

式中: N_r 为重复访问节点数, N_v 为总访问节点数。

4.3 实验结果与对比分析

4.3.1 覆盖率分析

进行了 10 次独立实验并进行统计,IPSO 算法在覆盖率这一方面展现出比较出色的表现。其平均覆盖率达到 99.95%,标准差仅为 0.03%,最小覆盖率为 99.88%。与之相比,RSH 算法的平均覆盖率为 99.85%,标准差是 0.12%,最小覆盖率为 99.65%,稳定性相对较欠佳,关于覆盖率性能的对比情况如表 2 所示。

表 2 覆盖率性能对比

单位: %			
算法	平均覆盖率	覆盖率标准差	最小覆盖率
IPSO	99.95	0.03	99.88
RSH	99.85	0.12	99.65

4.3.2 任务完成时间分析

统计任务完成时间之后得出结果表明,IPSO 算法在最大完成时间这项指标上,和 RSH 算法相比,平均值下降了 6.5% 到 7.5%。这体现出效率有所提高。关于完成任务时间的性能对比情况,如表 3 所示。

表 3 完成时间性能对比

算法	平均完成时间 /s	改进幅度 /%
IPSO	185~195	—
RSH	200~210	6.5~7.5

4.3.3 负载均衡性能分析

IPSO 算法增加了负载均衡权重项。此权重项可让各智能体的任务量分布更为均匀,其负载均衡度为 0.813。该数值与 RSH 算法的 0.588 相比高很多。负载均衡度的计算结果如图 4 所示。

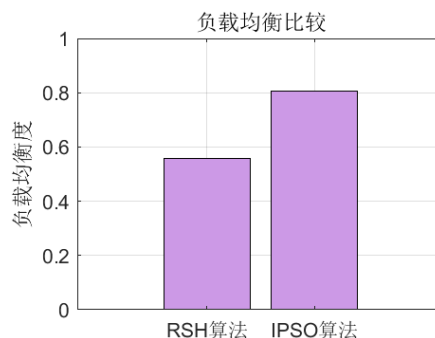


图 4 负载均衡对比

4.3.4 路径质量综合分析

路径质量是衡量协同侦察路径规划效果的关键指标之一。该指标主要体现在路径平滑度、路径交叉情况以及节点重复率等方面。IPSO 与 RSH 这两种算法所得到的路径结果图,分别如图 5~6 所示。算法路径质量的对比情况如图 7 所示。

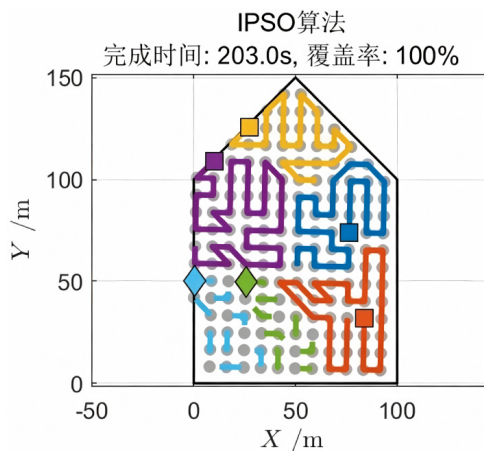


图 5 IPSO 算法路径图

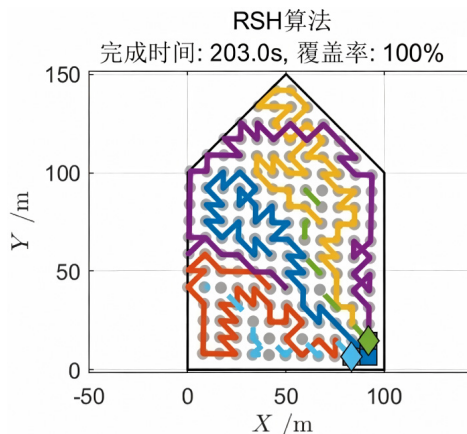


图 6 RSH 算法路径图

根据所呈现的结果显示,IPSO 算法所生成的路径平滑度较高,路径交叉数量为零,节点重复率被控制在较低水平;与之相对,RSH 算法的路径平滑度较低,并且存在多处路径交叉以及节点重复访问的现象。由图 5~6 可见,IPSO 算法在路径连续性与合理性方面展现出优势。进一步而言,图 7 对路径质量的各项指标进行了量化对比,IPSO 算法在路径平滑性、连通性以及简洁性方面均表现出更优的性能。这验证了其在复杂环境下生成高质量路径的能力。

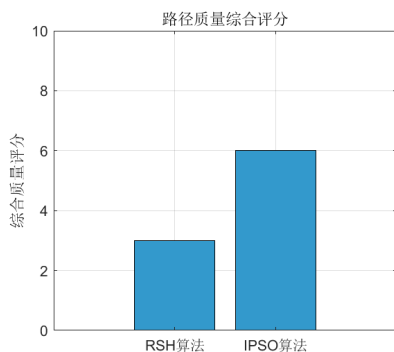


图 7 路径质量对比

4.3.5 收敛性能分析

收敛性能反映了算法的优化效率与稳定性。实验结果如图 8 所示。

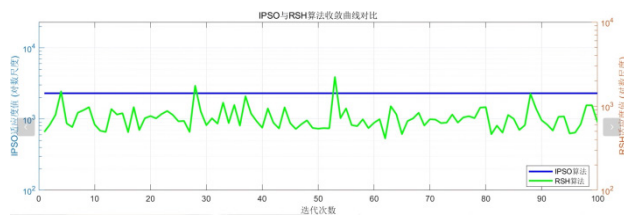


图 8 算法收敛曲线对比

IPSO 算法在大约 40~50 代达到收敛状态。收敛之后,其适应度值保持稳定,波动范围小于 2%。相比之下,RSH 算法的收敛过程则较为缓慢。该算法在 70~80 代区间内仍处于优化过程之中。并且,其在后期表现出的波动较大,波动范围约为 8%。

4.4 异构协同效果深入分析

4.4.1 速度适应性验证

IPSO 算法能够根据智能体之间的速度差异,自动调整任务的分配方式。具体而言:

对于高速无人机($v=5.0$ m/s),其平均覆盖节点数 35.2 个,主要承担大范围侦察任务;

对于中速无人机($v=4.0$ m/s),其平均覆盖节点数 28.4 个,负责执行中等范围任务;

对于低速潜航器($v=1.5$ m/s):其平均覆盖节点数 18.3 个,侧重于精细区域的侦察工作。

上述任务分配比例与各类智能体的速度比基本保持一

致。由此可以验证,该算法具有针对速度差异的自适应性。

4.4.2 续航约束满足分析

所有智能体均在其各自的续航时间范围内完成了所分配的任务:

对于无人机而言,其续航利用率的平均值为 84.3%,最高值达到 92.1%;对于潜航器而言,其续航利用率的平均值为 63.8%,最高值达到 71.5%。此外,未出现续航违反的情况:在所有运行实例中,满足续航约束的比例为 100%。

4.4.3 算法鲁棒性测试

对算法的鲁棒性展开测试,具体做法是改变任务区域的几何形状以及智能体的数量。当区域形状与智能体数量出现变化时,IPSO 算法依旧可维持稳定的性能表现。IPSO 算法在这一条件之下的路径效果如图 9 所示。

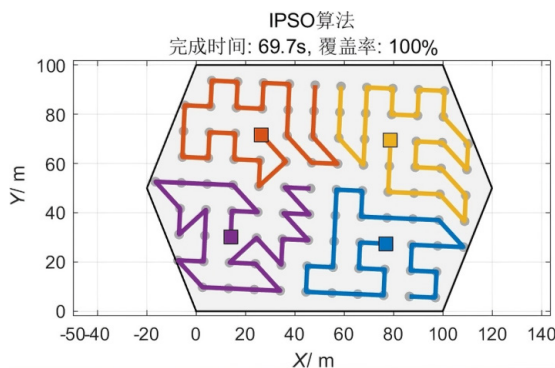


图 9 IPSO 算法路径图

4.5 综合性能评价

经过多维分析可知,IPSO 算法在多个方面都比 RSH 算法有优势。这些方面有覆盖率、任务效率、负载均衡、路径质量还有收敛性能。之所以能有这样的改进,是因为有统一的优化框架、多目标适应度函数、增强的搜索机制和有效的约束处理策略。该算法为复杂环境下的多智能体协同侦察提供了一种高效可靠的解决办法。

5 结论

本文聚焦于异构多无人系统协同覆盖路径规划这一问题,进而提出了一种改进的粒子群优化算法也就是 IPSO 算法。该算法借助一系列创新性设计,成功克服了传统“先分区、后规划”方法在路径冗余、负载不均、约束满足等方面存在的不足。

(1) 算法框架创新:搭建起区域划分与路径规划相互统一的协同优化框架,达成了任务分配与路径生成的深度融合。防止了因阶段割裂而引发的性能损耗。

(2) 在编码与更新机制方面的创新如下:具体设计了混合编码方案,并且还设计了可感知连续性的粒子更新策略,这样的设计能够让算法同时对离散任务分配问题以及连续路

径优化问题加以处理,而且该机制对于提升路径的物理可行性以及平滑性也是有帮助的。

(3) 在多目标优化模型方面有所创新,构建了一个多目标适应度函数,此函数综合考量了任务完成时间、覆盖效果、负载均衡以及路径平滑度等因素,并且借助权重调节这一方式,达成了在不同应用场景之下的优化平衡状态。

(4) 在约束处理机制这一领域呈现出创新之处,引入了借助图论建立的路径修复策略,并且还引入了续航约束处理机制。这些机制能够提高算法于复杂约束条件之下的求解可行性、鲁棒性。

仿真实验结果显示,和增强随机搜索启发式(RSH)算法相比,IPSO算法在多个关键性能指标方面都展现出了一定的优势。具体来讲:

覆盖率:IPSO算法的平均覆盖率是99.95%,比RSH算法的99.85%更优,而且展现出更高的稳定性。

任务效率:IPSO算法让最大完成时间的平均值缩短了6.5%到7.5%。

负载均衡:IPSO算法的负载均衡度提高到0.813,和RSH算法的0.588相比有优势。

路径质量:IPSO算法所形成的路径具有更高的平滑度,其路径交叉数为零,并且节点重复率相对较低。

收敛性能:IPSO算法的收敛速度比较快,收敛之后所呈现出的稳定性更好,其波动范围小于2%。

通过综合分析可知,本文所提出的改进粒子群优化(IPSO)算法,在理论方面达成了区域划分、路径规划的一体化协同建模与优化工作。在应用领域,针对复杂环境下异构无人系统协同侦察的路径规划问题,给出了高效且均衡、具有可扩展性的工程解决办法。仿真结果证实了该算法具备优越性与鲁棒性,还展现出良好的工程应用潜力。

参考文献:

- [1] 闫敬,关新平.海上无人系统跨域集群发展现状及其关键技术[J].自动化学报,2025,51(4):744-761.
- [2] 王亮,徐同乐,肖玉杰,等.海洋自主无人系统跨域协同任务规划模型与技术发展研究[J].兵工学报,2024,45(S2):1-8.
- [3] 孔国杰,冯时,于会龙,等.无人集群系统协同运动规划技术综述[J].兵工学报,2023,44(1):11-26.
- [4] Zhang Yun, You Shixun, Yan Yunbin, et al. Finding a new balance point: intelligent optimization of multi-target cognitive electronic reconnaissance strategy for unmanned aerial vehicles[J]. IEEE Access, 2024, 12: 183252-183264.
- [5] 洪芳宇,张涛,杨昊,等.面向点与区域目标联合成像侦察的多无人机协同任务规划[J].控制与决策,2026,41(4):1122-

1134.

- [6] Giusti E, Capria A, Saverino A L, et al. A Drone-based multi-sensory payload for maritime pollutants detections[J]. IEEE Aerospace and Electronic Systems Magazine, 2023, 38(3): 4-18.
- [7] 张玉强,孙盛智,张琪.无人机系统关键技术发展综述[J].火力与指挥控制,2025,50(8):1-9.
- [8] 王浩亮,于德智,卢丽宇,等.基于自适应视距制导的无人潜航器三维协同路径跟踪控制[J].控制与决策,2025,40(1): 242-251.
- [9] Liu Haishi, Chen Qiyong, Pan Nan, et al. Three-dimensional mountain complex terrain and heterogeneous Multi-UAV cooperative combat mission planning[J]. IEEE Access, 2020, 8: 197407-197419.
- [10] Ramezani M, Atashgah M A A, Rezaee A. A fault-tolerant multi-agent reinforcement learning framework for unmanned aerial vehicles-unmanned ground vehicle coverage path planning[J]. Drones, 2024, 8(10): 537.
- [11] 黄泽贤,江好胜,吴健,等.面向侦察任务的多无人机覆盖路径规划与控制[J/OL].信息与控制,1-14[2026-05-29]. <https://doi.org/10.13976/j.cnki.xk.2025.0231>.
- [12] 郭兴,李擎,姚其家,等.异构无人系统协同控制研究进展[J].工程科学学报,2025,47(1):66-78.
- [13] Yin Shihong, Hu Jiabao, Xiang Zhengrong. Multi-objective collaborative path planning for multiple water-air unmanned vehicles in cramped environments[J]. Expert Systems with Applications, 2025, 292: 128625.
- [14] 傅培宽,刘秋兵,穆荣健,等.动态环境下两栖无人战车实时路径规划算法[J].火力与指挥控制,2025,50(6):114-122.
- [15] 王永成,杨明漾,张国辉.基于改进A星算法对自动导引小车路径规划研究[J].火力与指挥控制,2021,46(8):130-138.
- [16] 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.
- [17] Cho S W, Park H J, Lee H, et al. Coverage path planning for multiple unmanned aerial vehicles in maritime search and rescue operations[J]. Computers & Industrial Engineering, 2021, 161: 107612.
- [18] 韩靖童,余倩,刘源.基于深度强化学习的海上搜救覆盖路径规划算法应用[J].信息与控制,2025,54(4):545-555.
- [19] 武晓雯,郭孟营,胡阿建,等.基于改进遗传粒子群算法的无人机路径规划[J].仪器仪表学报,2025,46(4):315-325.

基于混合遗传算法的航空货运仓储调度优化研究

李知微^{1,2} 喻忠全^{1,2} 何雨松^{1,2} 林虹秀^{1,2}
LI Zhiwei YU Zhongquan HE Yusong LIN Hongxiu

摘要

随着航空货物吞吐量激增,针对航空货运立体仓储中堆垛机调度效率低、路径冗余、多目标冲突等问题,文章提出一种融合遗传算法全局探索、NSGA-II 非支配选择与粒子群局部增强的混合多目标进化模型(HGAM)。该框架采用“分层片段编码”保证解的可行性,引入三指标融合适应度动态平衡收敛性与多样性,并通过嵌入式 PSO 每 20 代对外部档案进行片段级局部增强,实现“全局-局部”双向信息流动。基于 13 组真实航空货运任务场景下的实验结果表明:HGAM 在 89 代即收敛,最优目标函数值 294.17 s,显著优于 GA 与 NSGA-II 算法。模型将堆垛机总路径由 423.67 m 降至 346.33 m,为航空货运高密度存储环境下多目标调度优化提供了可扩展的新范式。

关键词

航空货运; 立体仓储; 调度优化; 混合遗传算法; HGAM

doi: 10.3969/j.issn.1672-9528.2026.06.015

0 引言

近年来,航空货运业务订单规模稳步攀升,货物种类呈现多元化、复杂化特征,这对仓储物流管理的时效性与精准性有着更为严苛的要求^[1]。面对行业发展带来的现实挑战,密集存储相关技术逐渐成为破解仓储空间利用率不足、作业效率偏低等痛点的主流研究方向。自动化立体仓库作为密集

存储体系的典型应用形式,凭借存储空间利用率高、自动化作业能力强、系统响应迅速等突出特点,已在现代物流管控体系中占据核心地位。

区别于普通仓储场景,航空货运具备货物周转频次高、品类跨度大、出入库节奏快等独有特征,依靠传统人工完成出入库作业,普遍存在人力消耗大、作业流程繁琐、人为操作失误率高以及长期运营成本居高不下等问题^[2]。为改善这一现状,各大物流中心引入 RGV、AGV 等自动化设备以优化物料搬运与库内任务调度,有效提升货物存取的作业效率与执行精准度^[3]。将智能调度优化策略融入密集仓储运行体

1. 中国民用航空总局第二研究所 四川成都 610041
2. 民航成都物流技术有限公司 四川成都 611430

- [20] 肖鹏,田润澜,李赫,等.基于改进蚁群算法的无人机通信侦察航迹规划[J].控制与决策,2025,40(11):3239-3252.
- [21] 孙哲,马胜男,解相朋,等.基于仿生算法的多式联运路径规划方法综述[J].控制与决策,2025,40(2):375-386.
- [22] Zhao Jiale, Deng Chaoshuo, Yu Huanhuan, et al. Path planning of unmanned vehicles based on adaptive particle swarm optimization algorithm[J]. Computer Communications, 2024, 216: 112-129.
- [23] 周宏宇,王小刚,单永志,等.基于改进粒子群算法的飞行器协同轨迹规划[J].自动化学报,2022,48(11):2670-2676.
- [24] 戴健,许菲,陈琪锋.多无人机协同搜索区域划分与路径规划[J].航空学报,2020,41(S1):149-156.
- [25] 尹洪玉,吴宇,梁天骄.固定翼无人机巡逻覆盖协同路径规划方法[J].航空学报,2024,45(6):329-343.

- [26] 欧丽珍,杨旭,李新梦,等.基于 Dijkstra 改进遗传算法的定向越野赛道设计[J].火力与指挥控制,2022,47(4):29-33.

【作者简介】

张文斌(1999—),男,安徽合肥人,硕士研究生,研究方向:多无人机协同侦察路径规划。

李杰(1988—),男,湖南娄底人,博士,副研究员,研究方向:智能无人机与集群技术。

夏元清(1971—),男,安徽天长人,博士后,中原工学院党委副书记、校长,教授,博士生导师,研究方向:多源信息复杂系统的信息处理与控制、云控制与决策理论及其应用、天空地一体化网络环境下多运动体跨域协同控制与智能决策等。

(收稿日期:2025-12-31 修回日期:2026-06-10)