



沈阳航空航天大学

工程硕士学位论文

论文题目 基于改进 **RRT** 算法的无人机航迹
规划方法研究

研究生 刘奎武

工程领域 控制工程

研究方向 任务分配与路径规划

指导教师(学校) 陈侠 教授

指导教师(企业) 郝杰 研究员

二零二二年五月

分类号_____

密级_____公开_____

UDC _____

学 位 论 文

基于改进 RRT 算法的无人机航迹规划方法研究

研 究 生 姓 名 : 刘奎武

工程领域名称 : 控制工程

研 究 方 向 : 任务分配与路径规划

论 文 类 型 : 基础研究

申请学位类别 : 工 程 硕 士

指导教师(学校) : 陈侠 教授 沈阳航空航天大学

指导教师(企业) : 郝杰 研究员 沈阳申蓝航空科技有限公司

论文提交日期 : 2022 年 05 月 15 日

论文答辩日期 : 2022 年 05 月 27 日

沈 阳 航 空 航 天 大 学

2022 年 5 月

SHENYANG AEROSPACE UNIVERSITY

THESIS FOR MASTER'S DEGREE

**RESEARCH ON PATH PLANNING OF UAV
BASED ON IMPROVED RRT ALGORITHM**

Candidate: Kuiwu Liu

Supervisor: Xia Chen

Specialty: Control Engineering

Date: May, 2022

原创性声明

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立完成的。除文中已经注明引用的内容外，本论文不包含其他个人或集体已经发表或撰写过的作品或成果，也不包含本人为获得其他学位而使用过的成果。对本文研究做出重要贡献的个人或集体均已在论文中进行了说明并表示谢意。本声明的法律后果由本人承担。

论文作者签名：刘奎武

2022 年 5 月 15 日

摘 要

随着科学技术的发展,无人机在各个领域里受到了广泛的关注。作为无人机的核心技术,航迹规划一直都是国内外学者研究的重点。因此,研究航迹规划方法具有重要的理论价值及实际意义。本文研究内容主要分为三个方面:

针对静态威胁下无人机全局航迹规划问题进行了研究。针对传统 RRT 算法的不足,提出了一种基于 RRT 算法与改进人工势场法相融合的航迹规划算法,以解决搜索范围随机性强和收敛速度较慢的缺点。首先,引入目标偏置来引导随机采样点的生成,减少采样次数。同时结合改进人工势场法将目标点、采样点和障碍物的合力方向作为搜索树的生长方向,加快路径搜索的效率。然后使用遗传算法对路径进行了优化,获得较优路径,解决了路径随机性问题。

针对移动威胁下无人机动态航迹规划问题进行了研究。建立了无人机和移动威胁的运动模型,利用改进 RRT 算法获得一条全局航迹。当环境中出现移动威胁时,依据无人机与障碍物之间的速度矢量关系,考虑了四种可能与障碍物发生碰撞的情况,并设计了相应的避障策略。使用速度障碍方法对航迹进行重规划,快速规避障碍物。

针对突发威胁下无人机三维航迹规划问题进行研究。将改进 RRT 算法由二维扩展到三维,建立了无人机在三维环境下的运动模型和环境威胁模型,给出了基于改进 RRT 算法的三维航迹规划算法。同时,考虑了出现突发威胁的情况,给出了局部路径重规划方法。将改进算法与原 RRT 等算法进行比较,仿真验证了该算法能够提高搜索速度,并能快速规避突发威胁。

关键词: RRT 算法; 人工势场法; 遗传算法; 移动威胁; 速度障碍法

Abstract

With the development of science and technology, UAVs have received extensive attention in various fields. As the core of UAV technology, path planning has always been the focus of domestic and foreign scholars. Therefore, the study of path planning has important theoretical value and practical significance. The research content of this article is mainly divided into three aspects:

The global path planning of UAV under static obstacle is studied. In view of the disadvantages of the traditional RRT algorithm, the RRT algorithm combined with Artificial Potential field method for UAV path planning is presented to solve the disadvantages of the high randomness and the slow convergence rate. First of all, the Goal bias is adopted to optimize the generating position of random sampling points which reduce the sampling numbers. Meanwhile, the Artificial Potential field is introduced to make the generation direction of the forces resultant that is generated by target point, random sampling point and the obstacles as the growth direction of search tree. The efficiency of path search is accelerated. After that, Genetic Algorithm is used to optimize the path to obtain a better path and solve the randomness of the path.

The problem of UAV dynamic path planning under moving obstacle is studied. The motion model of UAV and moving obstacle is established, and a global path is obtained by improved RRT algorithm. When a moving obstacle occurs in the environment, according to the velocity vector relationship between UAV and the moving obstacle, four possible situations collisions with the obstacle are considered and corresponding obstacle avoidance strategies are designed. The Velocity Obstacle method is used to replan the path and avoid the obstacle quickly.

The 3D path planning of UAV under sudden obstacle is studied. The improved RRT algorithm is extended from 2D to 3D. The UAV motion model and the obstacle model in 3D environment are established. A 3D path planning algorithm based on improved RRT algorithm is presented. At the same time, considering the emergence of the sudden obstacle, a local path replanning method is presented. By comparing the improved algorithm with the original RRT algorithm, the simulation results show that the proposed algorithm can improve the search speed and avoid sudden obstacle quickly.

Keywords: RRT algorithm; Artificial Potential field; Genetic Algorithm; moving obstacle; Velocity Obstacle

目 录

摘 要	I
ABSTRACT	II
第 1 章 绪论	1
1.1 研究背景与意义	1
1.2 国内外相关研究进展	2
1.3 本文主要研究思路	5
1.4 论文结构安排	6
第 2 章 航迹规划基础理论概述	7
2.1 无人机航迹规划问题分析	7
2.2 无人机航迹规划分类	8
2.3 常见航迹规划算法基本原理介绍	9
2.3.1 传统航迹规划算法	9
2.3.2 智能优化算法	10
2.4 本章小结	11
第 3 章 基于改进 RRT 算法的无人机二维航迹规划	12
3.1 改进 RRT 算法	12
3.1.1 基本 RRT 算法	12
3.1.2 RRT 算法改进	14
3.1.3 无人机航迹约束	16
3.1.4 改进 RRT 算法流程	17
3.2 基于遗传算法的航迹优化	19
3.3 仿真分析	21
3.4 本章小结	25
第 4 章 移动威胁下的无人机动态航迹规划	27
4.1 移动威胁下的无人机航迹规划流程	27
4.2 速度障碍法原理	28
4.3 移动障碍规避策略	29
4.3.1 无人机与移动障碍运动模型	29
4.3.2 障碍规避策略	30

4.4 仿真分析	34
4.5 本章小结	37
第 5 章 基于改进 RRT 算法的无人机三维航迹规划	38
5.1 无人机运动模型和三维环境建模	38
5.2 突发威胁模型的建立	39
5.3 基于改进 RRT 算法的三维航迹规划	39
5.3.1 无人机三维航迹约束	39
5.3.2 代价函数模型	41
5.3.3 突发威胁航迹重规划	41
5.3.4 动态环境下改进 RRT 算法三维航迹规划流程	43
5.4 航迹平滑优化策略	44
5.5 仿真分析	45
5.6 本章小结	52
结 论	53
参 考 文 献	55
致 谢	58
攻读硕士期间发表(含录用)的学术论文	59

第1章 绪 论

1.1 研究背景与意义

无人驾驶飞机(Unmanned Aerial Vehicle)简称“无人机”，缩写为“UAV”，是一种没有飞行员驾驶，通过地面站远程控制或依靠编程自主控制的飞行器，其主要作用是比普通飞机更安全，更有效地执行任务。无人机机动性强、造价低、灵活、可扩展性强，使其应用领域越来越广泛，例如民用和军事等领域^[1-3]。航迹规划是无人机控制系统不可缺少的部分，一直是各国学者广泛关注的问题。随着无人机飞行任务要求的提高，飞行环境信息变得越来越复杂，如何高效快速规划出一条可行航迹一直是无人机技术的重要研究课题。

近年来，无人机在民用领域应用的研究取得了较大进展。例如发生自然灾害后的搜救行动，搜救行动通常需要注意时效性，任何时间延误都可能会造成难以承受的后果，如受灾人员的伤亡等。然而，恶劣的搜救环境往往会对救援人员造成伤害，而无人机具有灵活、快速的特点，可以在付出较小的代价下，提供强力协助^[4]。在农业生产方面，使用无人机系统(UAS)作为传感和通信平台，是精密农业^[5]领域具有巨大潜力的突破性技术。它作为一种低成本的替代技术被引入到环境监测、高空间分辨率和图像采集等应用中。如今，无人机在农业生产中的应用正在扩大，它能够帮助种植者对农场进行监测和提供决策支持。无人机被用于各种农业实践中，如灌溉、施肥、农药、杂草管理等^[6]。无人机技术已经获得了飞速发展，由于其可以减少环境污染和道路拥堵所造成的负面影响，受到了交通运输部门的高度关注，这对传统运输系统的物流转运方式产生很大影响。无人机可以代替人工运输一些小型工业产品^[7]。此外，利用无人机自主、模块化、快速、高反应性的能力，也可以在受灾地区紧急商品运输中起到重要作用^[8]。

无人机广泛应用于军队之中，无人机作战的概念已广为人知。长久以来，由于无人机有着可以远程操控的优势，一直被用来侦察、打击某些基础设施。比如美国多次使用无人机攻击恐怖分子或摧毁物理目标。在过去的几十年中，美国国防部几乎在每次军事行动中都使用了无人机，其主要目的是对行动目标进行监视、侦察并获取情报^[9]。目前，已有 100 多个国家使用军用无人机执行任务，例如在无人机上配备最新一代的摄像机，提供精确的地形侦察，用于攻击和救援任务等。现代科技的发展，使无人机的技术水平不断提高，人们不只局限于远程操控无人机，更多的情况下会在无人机上写入控制决策算法程序，使无人机不再需要实时远程控制，在遇到突发情况时，无人机可以通过算法评估战场威胁，规避突发情况，完成侦察或打击任务^[10]。在现代战争中，无人机将在战场中起到重要作用，减少人员伤亡，增大任务成功概率。无人机已经成为最有效的军事

手段之一。无人机技术的发展程度将在衡量一个国家综合实力时占据重要权重。因此,作为无人机技术研究的重要一环,自主飞行技术研究成为了各国学者争先研究的热点。

航迹规划算法是无人机安全飞行、高效实现任务目标的重要保证。航迹规划算法就是在给定的环境下,规划出一条避免与障碍物碰撞且满足飞行曲率的航迹。随着各领域无人机使用率的提高,任务内容复杂化,环境信息多样化,使无人机航迹规划算法的智能化水平要求逐渐提高。一般来说,航迹规划需要注意以下几点:路径最优,航迹平滑,规避障碍等。随着传感器的飞速发展,无人机对环境的感知越来越准确,可获得的信息也越来越多,这就使得规划出的航迹也变得更加的安全和有效^[11]。研究航迹规划是一个综合性非常高的课题,其中涉及了许多其他领域,如人工智能、飞行力学、导航制导等。通常情况下,航迹规划可以先收集已有的障碍威胁信息,将这些信息数字化,再使用航迹规划算法获得满足环境要求的飞行航迹,并保存上传到无人机系统中完成自主飞行。然而,在实际飞行过程中,突发事件的出现改变了环境障碍信息,已经规划好的航迹不再满足无人机的飞行要求。所以,无人机必须能够在线重规划,在面对突发情况时,能够实时规划出可行航迹,躲避突发威胁,到达任务地点,完成飞行任务。

如今,无人机技术水平已经取得很大进步,应用领域更加广泛。然而在现代化战争中,战场环境复杂化以及日常生活中人们对于无人机执行任务要求的提高,作为无人机技术重要的一部分,如何安全高效规划出可行航迹,提高任务效率,是一个亟待解决的问题。因此,无人机航迹规划的研究有着重要的理论与实际意义^[12]。

1.2 国内外相关研究进展

近年来,各国学者提出了许多航迹规划方法,例如 Dijkstra 算法^[13]、A*算法^[14]和人工势场法(Artificial Potential Field)^[15]等传统航迹规划算法,以及遗传算法(Genetic Algorithm)^[16]、粒子群算法(Particle Swarm Optimization)^[17],蚁群算法(Ant Colony Optimization)^[18],灰狼算法(Grey Wolf Algorithm)^[19-20]等智能优化算法。然而,对于日益复杂的战场环境以及无人机任务需求的多样性,原来的算法已经很难满足无人机航迹规划要求。于是,有关学者开始对算法进行改进来满足无人机飞行要求,下面将对几种常用的航迹规划算法进行介绍。

(1) Dijkstra 算法。Dijkstra 算法是图论中的经典算法。Dong 等人使用 Voronoi 图对威胁进行分割,将 Dijkstra 算法中的顶点替换为各威胁点之间连线中垂线的交点。对地图的预处理减少了顶点的数量,同时 Voronoi 图的特性保证了生成航迹不会与地图中的障碍发生碰撞。但是生成的航迹会出现偏航角较大的情况,对无人机的转弯能力要求很高^[21]。Maini 等人采用可视图处理障碍物,将障碍物转化为不规则多边形,多边形的每个顶点作为 Dijkstra 算法中的顶点,考虑了无人机最大偏航角,给出对应的转弯约束条件,改进后的算法能够加快 Dijkstra 算法的运算速度,但是由于多边形的顶点靠近

障碍物，会对无人机的飞行安全产生影响。同时，由于可视图不考虑障碍物运动方向，会导致 Dijkstra 算法陷入局部最优，找不到航迹^[22]。

(2) A*算法。A*算法是一种智能启发式搜索算法。Szczurba 等人改进了 A*算法的代价函数，将航迹的约束条件加入到代价函数中，减少了搜索的节点数量，提高了算法的搜索效率^[23]。Guruji 等人给出了一种改进高效 A*算法，减少了路径处理时间和路径长度^[24]。陈晓宏等人将网络组织的立体空间剖分网格理论与 A*算法相结合，设计了一种路径代价惩罚因子，改进了代价函数，使算法更具有指向性，同时考虑环境中障碍物分布，采用分段变步长航迹规划，将航迹规划分为巡航段和突防段，巡航段采用大步长，突防段采用小步长，减少了 A*算法的计算量^[25]。杨明亮等人对 A*算法中 $h(n)$ 的权重系数进行改进，同时融合了 DWA 算法考虑了局部路径重规划问题，解决了 A*算法的局部最优问题^[26]。张阳伟等人提出了一种改进环境建模的双向 A*算法，采用线性四叉树进行环境建模，降低了搜索节点的数量，在启发函数中加入三阶贝塞尔曲线平滑，优化新节点选取，采用双向变步长 A*算法提高了路径搜索效率^[27]。

(3) 人工势场法。人工势场法是一种模拟电势场分布的路径规划方法。Zhu 等人提出了一种三维空间规避障碍的无人机路径规划算法，对传统人工势场法进行改进，在算法中加入无人机飞行物理约束，不满足约束的势场力将被修正，改进后的算法能够精确规避障碍^[28]。Chen 等人对传统人工势场法进行改进，针对目标不可达问题，对斥力场函数进行改进，将目标点与无人机两者之间的距离加入到斥力场函数中，当距离变大时，斥力减小，保证无人机能够到达目标^[29]。李廷珍等人提出了一种三维环境下改进人工势场法的无人机航迹规划方法。针对算法中出现斥力大于引力的情况，在斥力场函数中加入距离因子。结合无人机性能约束对局部极小值区域进行规避，设计了一种航迹点回溯法，解决局部极小值问题^[30]。韩尧等人在传统航迹规划的基础上，设计了一种角度和速度参量调节算法，解决了不可达和震荡问题。同时，在原算法中加入了辅助避障力，当无人机探测到障碍物时，无人机能够提前一步进行障碍规避操作，避开局部极小值点^[31]。

(4) 遗传算法。遗传算法是一种基于种群的非确定性智能优化算法。Pehlivanoglu 等人改进了传统遗传算法的初始种群生成方式，使用蚁群算法生成一个次优路径，利用该次优路径生成遗传算法的初始种群，采用 Voronoi 和聚类方法处理次优路径中与障碍碰撞的点，加快遗传算法的收敛过程^[32]。文献[33]对遗传算法传统的突变方式进行改进，提出了振动突变算子的周期性突变方式，解决了遗传算法过早收敛的问题。同时，构造了 Voronoi 图，并基于 Voronoi 顶点生成了初始种群，加快了算法收敛速度。程泽新等人将改进差异进化变异策略用于遗传算法的种群变异阶段，增加了种群多样性，将变异产生的个体输入到模拟退火算法中，输出的最优解加入到原种群中，解决了局部最优问

题^[34]。Li 等人提出了一种具有优化搜索区域的层次遗传算法，根据任务区域内威胁源的分布情况，优化航迹点生成的区域，减少了算法的运行时间。在适应度函数中添加了航向修正约束和最小转弯半径约束，改善了生成航迹转弯较多的情况^[35]。黄书召等人同时对选择算子，交叉算子和变异算子进行改进，在轮盘赌选择的基础上，选择算子又新增了锦标赛选择方式，设计了非对称映射交叉算子，使用两种截断方法处理重复基因，提出了一种启发式多次变异算法子，避免了算法早收敛的问题，同时加快了算法搜索速度^[36]。

(5) 蚁群算法。蚁群算法是一种模拟蚂蚁群体社会行为的群智能优化算法。Chen 等人提出了一种快速、高效的 UCAV 改进蚁群航迹规划算法，通过改进转移概率来提高选择过程的速度，当一条航迹有超过三分之一的蚂蚁选择时，设计一个算子来代替信息素减少率，降低出现局部最优解的概率^[37]。李喜刚等人提出了一种改进蚁群无人机航迹规划算法，改进了信息素挥发系数，引入信息增量调节因子对次优和其他路径进行区分，加快了算法的收敛，并提出了双向搜索策略，使改进算法能够更快的获得可行航迹^[38]。类比于蚁群算法，人工蜂群算法是模仿蜜蜂行为提出的一种算法，蜜蜂之间的信息交换依赖于一种被称为摇摆舞的行为。文献[39]提出了一种基于多策略进化学习的人工蜂群算法的路径规划方法，给出了一种基于类脑机制的进化学习框架，设计了一个基于认知机制的多策略进化数据库，取代了传统的随机搜索方法，提高了人工蜂群算法的优化效果，减少了原有算法的迭代次数。

(6) 粒子群算法。粒子群算法是一种模拟鸟群觅食的航迹规划方法。方群等人对粒子群算法中惯性权值进行改进，提出了一种阶梯式调整策略，避免了复杂的函数运算，加快了算法的优化速度，同时重新定义了局部最优的判定条件，设计了一种脱离局部最优解策略，使算法能够避免局部最优，同时保证算法的稳定性^[40]。Liu 等人构造了一个自适应灵敏度决策区域，确定具有高概率成为最优解的潜在粒子位置，并删除其他候选粒子，提高算法计算能力。然后将粒子的搜索空间限制在一个有限的边界内，以避免算法早熟^[41]。鲁亮亮等人提出了一种基于 APSODE-MS 算法的无人机航迹规划方法，首先设计一种分段非线性函数对惯性权重进行调整，将空间中的粒子进行分类，并分配不同的惯性权重，然后将差分进化算法与改进粒子群算法结合，将单个粒子与个体最优和群体最优的相关性转换为多个粒子之间信息交流，同时由于差分进化算法的机制，影响粒子进化方向，提高了全局搜索能力^[42]。Girija 等人提出了一种在障碍物分布密集的环境下，基于粒子群和人工势场法结合的混合航迹规划算法，与基本粒子群算法相比，改进算法在取得相同的评价函数值时，用时更少^[43]。

(7) 灰狼算法。灰狼算法是根据灰狼群体捕猎行为提出的算法。Qu 等人提出了一种基于灰狼优化算法(GWO)的航迹规划算法，简化了 GWO 算法部分，保留了算法的搜索

能力,同时使用 GWO 算法对共生生物搜索算法(SOS)算法的共生阶段进行了改进,提高了开发能力,融合算法加快了算法收敛速度^[44]。李彦苍等人使用莱维飞行改进了探狼的搜索方式,在奔袭和围攻时,根据人工狼与头狼的距离选择不同步长前进,步长的选取与距离成反比,改进算法能够提高收敛速度并能避免局部最优^[45]。滕志军等人将灰狼优化算法与粒子群算法结合。对狼群初始化位置进行改进,使用混沌序列产生初始狼群位置,设计了一种参数非线性控制策略来调节惯性权重,同时使粒子群中个体的最佳价值和狼群的最佳价值共同决定每个灰狼的位置信息,提高了算法的搜索能力^[46]。

综上所述,航迹规划算法虽然已经取得一些研究成果,然而在日益复杂的飞行环境下,算法的快速性一直是航迹规划性能的重要指标。因此,如何快速获得较优的无人机航迹仍为研究热点。RRT 算法是由 LaValle 提出的一种快速搜索算法,它避免了准确的环境模型,并可以在较短的时间内找到一条路径^[47]。RRT 算法便于建模,搜索树拓展过程简单,算法能够满足实时性要求,但是该算法找不到最优路径,同时全局采样导致收敛缓慢,存在大量冗余点,不能满足无人机飞行要求。针对这些问题不少学者进行了改进。成浩浩等人利用概率引导随机树的生长方向和动态步长搜索未知空间,通过双层平滑优化策略规划路径,获得无人机飞行的可行路径^[48]。Faris 等人将 RRT 算法生成的路径点序列集输入四次样条曲线,获得了一条平滑的动态可行路径^[49]。尹高扬等人改进了原始 RRT 算法最近邻点的选取策略,但是仍然采取全局随机采样策略,在一定程度上增加了算法的搜索时间^[50]。刘成菊等人在目标偏置 RRT 算法的基础上加入了人工势场法的引力思想,给出了机器人的路径规划算法,该算法加快了路径搜索速度,但是并未考虑障碍物的斥力作用,导致障碍物周围存在过多冗余点^[51]。虽然已有学者对 RRT 算法进行了改进,但仍需要提高该算法的效率,原算法还存在路径随机性较大的问题。如何改进 RRT 算法仍然具有重要研究意义。

1.3 本文主要研究思路

目前,国内外关于无人机航迹规划的研究已经取得了一些成果。但相关算法研究仍存在不足,针对动态环境下的航迹规划研究成果还相对较少,目前已有的 RRT 算法存在着搜索效率低、计算量大、运行时间较长等不足,本文主要研究以下三个方面:

(1) 基于改进 RRT 算法的无人机二维航迹规划研究。本文提出一种将改进 RRT 算法、人工势场法及遗传算法相融合的无人机航迹规划方法。首先,利用目标偏置来改变随机采样点的生成方式,在原有全局采样的基础上,目标点有一定概率作为采样点,提高了搜索效率。然后引入人工势场法来优化新节点的生成方式,目标点和随机采样点作为最近邻点的引力,障碍物作为最近邻点的斥力,将三者的合力方向作为新节点的生成方向。最后规划出基于改进 RRT 算法的航迹。针对所生成航迹冗余点过多,路径曲率

不满足飞行要求的问题,采用遗传算法对生成的航迹进行优化,将改进后的 RRT 算法生成的一组航迹点序列作为遗传算法的初始种群,建立适应度函数,通过选择、交叉和变异操作获得较优无人机飞行航迹,在优化部分解决了路径随机性问题,且改进后算法生成的路径长度更短,消耗时间更少。

(2) 移动威胁下的无人机动态航迹规划研究。在获得静态威胁环境下的全局航迹后,需要考虑环境中出现移动威胁的情况。首先,使用改进 RRT 算法获得全局航迹,然后考虑环境中出现的移动威胁,建立移动威胁和无人机的运动模型。根据无人机和移动障碍速度之间大小与方向的关系,设计了四种不同碰撞情形,并给出了相对应的避障策略。最后采用速度障碍法(Velocity Obstacle,VO)进行移动障碍规避,完成实时路径重规划。

(3) 基于改进 RRT 算法的无人机三维航迹规划研究。针对改进 RRT 算法在三维环境中实现的问题,建立了无人机在三维环境下的运动模型以及环境威胁模型,将航迹规划过程中的物理约束具体化,当无人机航迹受突发威胁影响时,建立了突发威胁模型,并给出了避障策略,最后将改进算法与原始 RRT 算法和 tRRT 算法进行比较,通过仿真验证了改进算法能够快速到达目标点,并能准确规避突发威胁。

1.4 论文结构安排

第1章,绪论。本章介绍了无人机的应用领域,航迹规划算法的研究背景与意义。以及航迹规划算法的国内外研究现状。最后介绍了本文研究内容和章节安排。

第2章,航迹规划基础理论概述。本章主要对航迹规划问题进行分析,介绍了无人机航迹规划的分类,通过传统航迹规划算法和智能优化算法两方面对典型的规划算法的基本原理进行介绍。

第3章,基于改进RRT算法的无人机二维航迹规划。本章针对传统RRT算法的不足进行改进,引入目标偏置法来改进随机采样点的选取方式,结合人工势场法改进新树节点的生成方式。使用遗传算法对生成航迹进行优化,获得了较优路径。

第4章,移动威胁下的无人机动态航迹规划。本章根据运动空间中无人机与移动威胁之间的速度矢量关系,考虑了四种可能与障碍碰撞的情况,并设计了相应避障策略。在改进RRT算法生成全局航迹的前提下,使用速度障碍法进行局部重规划,验证了四种避障策略的可行性。

第5章,基于改进RRT算法的无人机三维航迹规划。本章研究了改进RRT算法在三维空间的实现以及算法在出现突发威胁时在线重规划的能力。首先,建立了三维无人机运动模型、环境威胁模型和突发威胁模型。然后分析了无人机三维航迹规划约束条件,并建立了代价函数模型。最后给出了突发威胁航迹重规划策略。仿真验证了改进算法在三维环境中的有效性,并且能够完成在线重规划。

结论:对全文工作的总结,并指出了存在的不足和未来要解决的问题。

第2章 航迹规划基础理论概述

航迹规划是无人机任务分配系统的重要组成部分。航迹规划就是根据环境信息，规划出一条避免与障碍物碰撞的航迹。同时，生成的航迹必须要考虑无人机性能，如最远飞行距离，最大偏转角度等，还需要规避地图中的各种障碍威胁。航迹规划可以划分为三个部分：首先获取地图中的障碍威胁信息，对障碍信息进行转化，能够被计算机识别，然后根据不同任务要求选择合适的航迹规划算法，最后使用三次样条等优化算法对生成航迹进行平滑^[52]。作为无人机执行任务的基础，近些年来，航迹规划算法受到了大量学者的关注。本章主要介绍了无人机航迹规划问题分析和航迹规划的分类，以及几种广泛应用的无人航迹规划算法。

2.1 无人机航迹规划问题分析

航迹规划问题是一个包含多个优化目标和多个约束的非线性规划问题^[53]。任务内容决定选取的约束条件，合适的约束条件和精确的数学模型可以极大提高算法的效率。无人机航迹规划地图环境分为二维和三维，在某些情况下，无人机执行任务时飞行高度不会改变，这样就可以将环境障碍等投影到二维平面，避免三维复杂情况，便于航迹规划算法的求解。目前，大多数研究都采用分层规划的思想，将航迹规划分成两个步骤。第一步，根据已知地图环境和任务要求，提前为无人机规划出一条满足飞行要求的航迹，此时应最大限度考虑飞行时间问题；第二步，当无人机按照预定轨迹飞行时，考虑环境中存在未知威胁或突发威胁信息，无人机实时监测，完成局部动态规划，并尽量缩短规划时间，确保实时性。

无人机航迹规划详细步骤如下：环境建模，航迹表示，确定约束条件，设计目标函数，航迹规划算法选择和航迹平滑。

(1) 环境建模。精确的环境是无人机航迹规划的基础，将实际地图环境等效转化为计算机能够识别的环境模型，转化后的环境模型需要最大限度的还原地图中的地形，障碍等信息，同时当地图中的障碍物等信息发生改变时，环境模型能够及时更新，保证航迹安全。常用的环境建模方法包括可视图法，该方法将环境威胁转化为凸多边形，以多边形顶点和边代替威胁边界。栅格图法，该方法将规划空间分成相同大小的单元，将地图信息数字化，单元的尺寸与地图精度和算法计算量成反比，选定栅格尺寸时需要注意算法搜索速度。

(2) 航迹表示。航迹规划算法生成的航迹可以分成两种表示方法。第一种，根据无人机时变连续运动学方程，生成一条连续航迹，这种情况下，不需要进行航迹平滑；另一

种是用离散运动学方程进行求解,生成的航迹由航迹点组成,通过航迹点连接成一条连续折线航迹。

(3) 确定约束条件。航迹规划算法应考虑无人机自身性能约束和外界环境干扰,一般来说,无人机的自身性能包括最大偏航角、最大俯仰角、最远飞行距离和最小最大转弯半径等。合理的航迹有利于无人机执行飞行任务。外界环境干扰主要分为地形干扰,火炮、雷达、导弹威胁,极端天气等,准确的约束条件建模是无人机航迹规划成功的重要基础。

(4) 设计目标函数。目标函数是对生成航迹优劣评价的重要指标。目标函数的设计必须满足无人机的性能指标,这样算法生成的航迹才能满足无人机飞行要求。无人机在飞行过程中必须要考虑自身携带燃料的消耗,这决定了无人机的最大航程,在航迹规划算法中,一般用航迹点之间的欧式距离之和表示无人机的航程,通常情况下,航迹越短,无人机获得收益最大。无人机自身安全是完成任务的重要保障,航迹规划算法根据障碍物的威胁范围,设立相应禁飞区,保证了生成航迹的安全性。根据不同的任务要求和环境的复杂程度,对规划时间的要求不同,离线航迹规划对时间的要求不高,但是在线规划需要算法能够快速规划出航迹。综合以上无人机的性能指标,根据不同任务要求建立合适的目标函数。

(5) 航迹规划算法选择。根据执行任务的方式,选择航迹规划算法,规划出的航迹需要保证飞机的安全和飞行曲率。

(6) 航迹平滑。经过规划算法生成的航迹不一定能直接用于无人机飞行,比如航迹冗余点过多,偏航角和俯仰角变化很大等等。在真实场景中,飞机不能频繁改变飞行姿态,因此,有必要对生成航迹进行平滑处理,消除航迹中的冗余点保证无人机正常飞行。常用的平滑算法包括贪婪算法、三次 B 样条法和贝塞尔曲线(Bezier Curve)等。

2.2 无人机航迹规划分类

无人机技术可以分为硬件部分和软件编程控制,硬件部分是指无人机机身、中央控制器、传感器等零件的研究、制作以及后续的组装,而软件编程包括使用航迹规划算法生成无人机可行航迹,控制系统通过该信息控制无人机飞行^[54]。无人机航迹规划发展至今已经产生了许多分支,无人机往往会因为环境和任务的不同而选择不同的算法。

根据无人机执行任务的性质可以将航迹规划分为全局航迹规划和局部航迹规划^[55]。全局航迹规划是指已知地图环境,在无人机起飞前,使用规划算法生成一条可行航迹,并将航迹信息上传到控制系统^[56]。局部航迹规划通常是无人机在飞行过程中,环境出现突发事件如突发威胁,需要对传感器获得突发威胁的信息进行分析,并使用算法重新规划出一条航迹,这种情况要求航迹规划算法运算时间很短。

根据无人机执行任务的数量分为单无人机航迹规划和多无人机航迹规划^[57]。无人机

在执行侦察或监视任务时,为了保证自身不被敌方发现,通常采用单个无人机执行任务,使用的无人机必须具有高速度,高续航的特点,这种情况可以称之为单无人机航迹规划。多无人机航迹规划是指当无人机执行火力支援或者定点摧毁任务时^[58-59],考虑到单架无人机能力有限,可能无法完成攻击任务,为了提高任务的成功率需要使用多架无人机共同打击目标^[60],使用的航迹规划算法必须考虑到多架无人机的协同关系,建立的目标函数为了保证效益最大化,所以对单架无人机来说可能并不是最优解。

2.3 常见航迹规划算法基本原理介绍

在上一节中,介绍了航迹规划算法的分类。近年来,针对不同航迹规划类别,国内外学者提出了许多航迹规划算法,利用算法来获得可行解,将无人机约束条件融入到路径规划算法中,再应用到无人机上。接下来本节将通过传统航迹规划算法和智能优化算法两方面对一些常用规划算法的基本原理进行介绍。

2.3.1 传统航迹规划算法

(1) Dijkstra 算法: Dijkstra 算法最早提出于 1959 年,是图论中的经典算法,主要是用于求解最短路径。首先需要构造有权图,将空间中的各个顶点与起始点的距离储存起来,从起点开始,遍历与其相邻的顶点,选出与起点距离最近的点,将该点单独存放到一个空集 T 中,然后以该点作为起点,在与其相邻的点中选择距离最近点存入到 T 中,需要注意的是每个顶点只能选择一次,经过多次迭代,直到目标顶点被找到。该算法易于实现,在范围较小的环境中搜索效果很好。但是,每更换一次起点就必须重新计算各个顶点到起点之间的距离,当地图范围变大时,算法中顶点的数量会大幅增加,导致算法的运算时间和占用内存显著增加,因此,不适用于大范围环境搜索路径,通常将该算法与其他算法结合使用。目前对于 Dijkstra 算法的改进,大多数是减少算法中的顶点的数量,通常使用 Voronoi 图、概率地图或可视图对地图环境进行预处理,然而若想要获得安全的航迹,譬如 Voronoi 图法,会导致生成的航迹过长且不能满足无人机飞行曲率,若想要较短路径,如可视图法,则会影响无人机的飞行安全。

(2) A*算法: A*算法是一种基于栅格的智能启发式搜索算法。以 Dijkstra 算法为基础,采用启发式思想,引入评价函数,通过评价函数来计算每个节点的优先级。首先,建立了 OPEN 表和 CLOSE 表,将起点存放 to OPEN 表中,并设置为最高优先级,寻找起点周围可以到达的点,并将这些点加入到 OPEN 表中,然后将起点存放 to CLOSE 表中,使用评价函数计算 OPEN 表中的点,将评价值最小的点取出放入 CLOSE 表中,同时将该点周围的可行点放入到 OPEN 表中,舍去与 OPEN 表中重复的节点。重复上述过程,直到终点出现在 OPEN 表中。该算法具有较强的全局搜索能力和回溯性,但是, A*算法计算量大,运行时间长,实时性较差。

(3) 人工势场法：人工势场法是一种模拟电势场分布的路径规划方法。人工势场法的思想是给每个障碍物分配一个类似于静电势的函数，然后以最小势能的形式推导出自由空间的拓扑结构。将无人机看成质点，引力场使无人机向目标点飞行，斥力场使无人机远离障碍物，避免与障碍物碰撞，在引力场与斥力场共同引导下，无人机通过势场梯度下降的方向到达目标点，获得一条可行航迹。人工势场法具有良好的实时性，在航迹规划中被广泛使用。但是，当无人机距离威胁太近时，斥力会大于引力，算法会停滞或震动，无人机将不能到达目标点，这也称为局部极小值问题。

(4) Voronoi 图：Voronoi 图^[61]又叫泰森多边形，是将搜索空间中相邻点连线的垂直平分线组合在一起的多边形。在航迹规划中，将障碍视为点，将所有相邻点的垂直平分线连接在一起，构成多边形“网状图”，然后获得一条可行的路径。刘振等人提出一种改进 Voronoi 图，将障碍物的威胁进行评级，采用动态贝叶斯网络对威胁等级进行感知，当障碍物威胁等级变化时，重新构造 Voronoi 图，改进后的算法能够规避环境中的突发威胁，实现路径重规划^[62]。Chen 等人根据已知的雷达威胁分布，利用威胁中心构造生成 Voronoi 图，以雷达威胁成本和燃料成本确定了 Voronoi 图各边的代价函数，同时 Dijkstra 算法满足了实时路径规划的优化问题^[63]。

2.3.2 智能优化算法

(1) 遗传算法：遗传算法是一种基于种群的非确定性智能优化算法，其灵感来源于达尔文的“适者生存”理论。遗传算法原理基于生物学中的遗传和变异，上一代的优秀基因通过遗传传递给下一代，下一代中个体的变异促进种群的进化。在航迹规划中，一组航迹点序列就是一条染色体，其中的航迹点作为单个基因，在建立初始种群后，首先计算种群所有个体的适应度，选取优秀个体组成新的种群，然后种群个体之间进行基因交叉，完成信息交换，接下来种群个体内部进行基因突变，改善种群质量，在经过上述三个操作后，用适应度函数对种群进行评价，如果满足截止条件，则输出最优解，否则，继续上述操作。遗传算法全局搜索能力很强，但是收敛速度慢。

(2) 蚁群算法：蚁群算法是一种通过模拟自然界蚂蚁寻找路径的过程的群智能优化算法。自然界中，蚂蚁总能根据环境变化实时更新寻找食物的路径，这种现象发生的根本原因是蚂蚁会释放一种称为信息素的物质，这种物质能够在蚂蚁之间传递信息，信息素浓度越高对蚂蚁吸引力越大，这是一种正反馈。在航迹规划中，将若干蚂蚁放置在起点，每只蚂蚁都按照设定的转移概率前进，当所有蚂蚁都到达目标点后，在所得航迹中找到最小代价航迹，以该航迹为基准，使用信息素修改规则修正其他路径中的信息素浓度，然后在起点重新进行航迹规划，当到达最大迭代次数后，输出最优航迹。由于蚁群算法中寻找新节点时概率是随机的，会出现算法收敛速度较慢的情况，由于正反馈机制，当搜索空间变大时，会出现局部最优问题。

(3)粒子群算法：粒子群算法是一种连续非线性函数的优化方法，该算法来源于对鸟群觅食这一社会群体行为的研究。每个个体都是优化问题的一个可行解，每个个体的下一步行动由个体自身最优解和群体最优解共同决定，单独个体由速度和位置信息表示，每次迭代，每个个体都会根据群体中的最优个体调整自己的速度和位置，同时群体位置也会相应改变，当满足终止条件时，算法结束。作为一种不确定算法，粒子群算法对于某些特定问题有着独特优势，便于求解全局最优解，同时群体之间的信息沟通对环境的适应能力较强。然而由于个体之间信息的单向传播，算法容易出现早熟和局部寻优能力较弱等问题。

(4)灰狼算法：狼群算法是根据狼群的捕食行为所提出的。整个狼群由头狼，探狼，猛狼组成。头狼由竞争产生，通过接收狼群探查的信息进行决策，探狼负责搜寻猎物信息，通过猎物留下的气味进行追踪，猛狼则是根据头狼的指令对猎物发动攻击。在航迹规划中，狼群中的最优解作为头狼，当探狼中感知猎物的气味浓度大于头狼或达到最大游走次数，猛狼会向当前探狼位置前进，狼群执行围攻行为，完成一次围猎后更换头狼，进行群体更新，当到达目标点或最大迭代次数，输出头狼位置，完成航迹规划。狼群算法具有较高的搜索速度，但是容易陷入局部最优并且精度不高。

2.4 本章小结

本章叙述了无人机航迹规划的问题分析和航迹规划的详细步骤。最后阐述了航迹规划中常用的一些典型算法的基本原理和优缺点。

第3章 基于改进 RRT 算法的无人机二维航迹规划

航迹规划作为无人机技术的重要部分，算法的研发与改进受到了国内外学者的广泛重视。在第二章中，介绍了一些常见的航迹规划算法，比如人工势场法、A*算法、Dijkstra 算法等传统的航迹规划算法，以及遗传算法、蚁群算法等智能优化算法，这些算法虽然能找一个最优航迹或次优航迹，但运行时间上会随着环境模型的增大呈现指数增长。然而，在实际战场环境中，随着战场环境的日益复杂，无人机执行任务要求的提高，上述航迹规划算法已经不能满足飞行环境实际要求。RRT 算法不需要精确的环境建模，随机采样点全局采样，不会出现局部最优问题。然而，该算法对整个采样空间进行搜索，导致算法搜索效率低，生成的航迹质量差，且每次生成的航迹有较大差异。针对以上问题，不少学者对其进行改进，比如采用动态步长可以加快搜索树的生长速度，然而由于 RRT 算法随机采样点的选取遍布整个搜索空间，动态步长仍然不能解决航迹拐点较多的问题。而 RRT*算法是对 RRT 算法最近邻点的选取进行优化衍生的算法，该算法虽然能生成一条较优航迹，但是每次生成新的节点后都要重新选取最近邻点并对搜索树重新布线，这极大的增加了算法的运行时间。tRRT 算法通过引入目标点对最近邻点的引力可以提高生长树的指向性，采用目标点引力优化随机树的生长方向，虽然减少航迹拐点，但在障碍物周围会产生较多无用树节点，增加了算法不必要的运行时间。

针对以上改进 RRT 算法存在的不足，本章以原始 RRT 算法为基础，同时对随机采样点的选取，新节点的生成方式进行改进，引入目标偏置使目标点概率成为随机采样点，增强搜索树的指向性。在计算新节点时，不仅考虑目标点对最近邻点的引力，还考虑了障碍物对其的斥力，减少了冗余树节点的数量，缩短算法的运行时间。同时采用遗传算法对生成航迹进行处理，改进算法生成的航迹代替遗传算法随机生成的初始种群，此时初始种群比较接近于最优解，能够减少遗传算法运行时间，仿真结果表明，本文改进算法能够生成较优航迹，改善了 RRT 算法航迹随机性大的问题。

3.1 改进 RRT 算法

3.1.1 基本 RRT 算法

RRT 算法是在空间中随机采样构造搜索树，搜索树从飞行起点扩展到目标点，在节点扩展过程中应满足 UAV 的性能约束。算法的扩展示意图如图 3.1 所示。

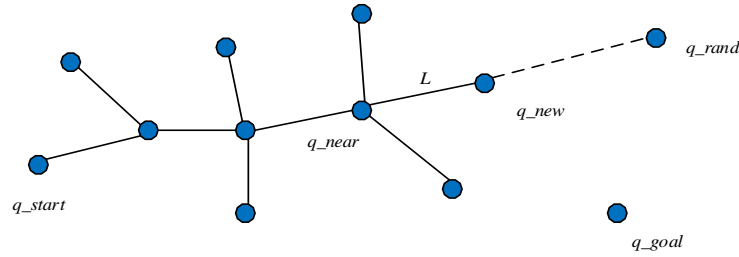


图 3.1 RRT 算法节点扩展示意图

在图 3.1 中, q_{start} 和 q_{goal} 分别为飞行起点和目标点, q_{rand} 代表空间中的随机采样点, q_{near} 为搜索树中的最近邻点, q_{new} 是扩展后生成的新节点, L 为无人机的飞行步长。RRT 算法的流程图如图 3.2 所示。

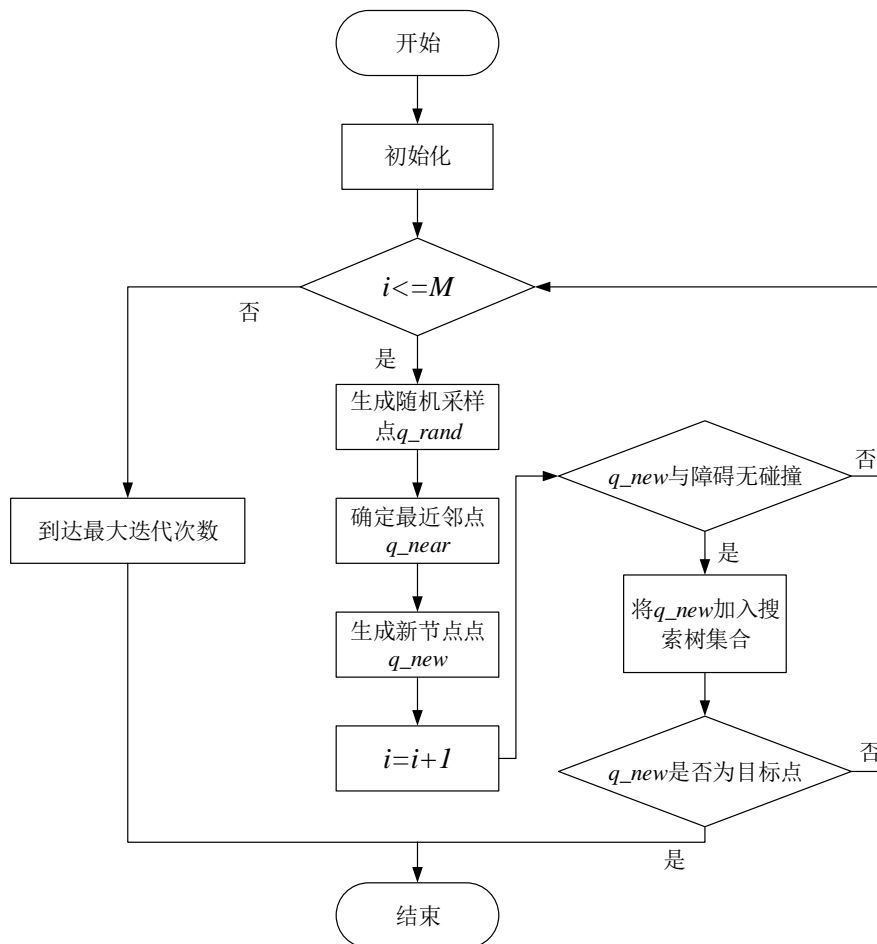


图 3.2 RRT 算法流程图

首先进行初始化设置, q_{start} 作为扩展树的第一个最近邻点 q_{near} , M 是最大迭代

次数，然后在空间随机选取一点作为 q_{rand} ，根据 q_{rand} 与搜索树集合中的点的欧式距离选取最近邻点 q_{near} ，以 q_{near} 为起点朝 q_{rand} 方向扩展步长 L 获得新节点 q_{new} ，如果 q_{near} 与 q_{new} 的连线途经障碍或 q_{new} 处于障碍中，则需要重新选择 q_{near} ，反之将 q_{new} 储存在搜索树集合中。当 q_{new} 到达目标点或到达最大迭代次数 M ，算法终止。

3.1.2 RRT 算法改进

本文主要在两个方面对 RRT 算法进行改进：随机采样点 q_{rand} 的选取和新节点 q_{new} 的生成。在原始 RRT 算法中， q_{rand} 由搜索空间随机采样获得，本文对 q_{rand} 选取做了改进，引入目标偏置思想，改进后的 q_{rand} 依照公式(3.1)选取。

$$q_{rand} = \begin{cases} \text{随机采样} & \text{if } rand() > \delta \\ q_{goal} & \text{else} \end{cases} \quad (3.1)$$

其中 $rand()$ 是 0-1 间的随机数， δ 是设定的目标偏置阈值，当 $rand()$ 值大于设定阈值时，在采样空间中选取随机点作为 q_{rand} ，反之将 q_{goal} 作为随机采样点。

在原始 RRT 的基础上，通过加入人工势场法改进 q_{new} 的生成方式，加快算法的收敛速度。Khatib 首次提出了人工势场法，其中包括引力场 $U_{att}(q)$ 和斥力场 $U_{rep}(q)$ 。假设 p 是无人机的一个航迹点，则引力势场函数、斥力势场函数和合力势场函数表示如下：

$$U_{att}(p) = \frac{1}{2} k_a \rho_g^2(p) \quad (3.2)$$

$$U_{rep}(p) = \begin{cases} \frac{1}{2} k_r \left(\frac{1}{\rho(p)} - \frac{1}{\rho_0} \right)^2 & \rho(p) \leq \rho_0 \\ 0 & \rho(p) > \rho_0 \end{cases} \quad (3.3)$$

$$U_{total} = \sum U_{rep} + \sum U_{att} \quad (3.4)$$

其中 k_a 是引力场增益常数， k_r 是斥力场增益常数， ρ_0 是航迹点 p 距离威胁范围的最大安全距离， $\rho_g(p)$ 和 $\rho(p)$ 分别表示点 p 与目标点和最近威胁中心的欧氏距离。引力和斥力的大小分别是引力场函数和斥力场函数的负梯度。根据式(3.2)-(3.4)可知，引力和斥力的大小为：

$$F_{att}(p) = k_a \rho_g(p) \quad (3.5)$$

$$F_{rep}(p) = \begin{cases} k_r \left(\frac{1}{\rho(p)} - \frac{1}{\rho_0} \right) \frac{\partial \rho(p)}{\rho^2(p) \partial x} & \rho(p) \leq \rho_0 \\ 0 & \rho(p) > \rho_0 \end{cases} \quad (3.6)$$

$$F_{total} = \sum F_{rep} + \sum F_{att} \quad (3.7)$$

通过结合人工势场法，目标点 q_{goal} 和随机采样点 q_{rand} 对 q_{near} 产生引力，障碍物对 q_{near} 产生斥力，三者的合力决定了 q_{new} 的生成方向， q_{near} 的受力示意图如图 3.3 所示。 F_{att1} 和 F_{att2} 分别为 q_{rand} 和 q_{goal} 对 q_{near} 的引力， F_{rep} 是障碍物对 q_{near} 的斥力，根据平行四边形法则求出合力方向，即 q_{new} 的扩展方向。

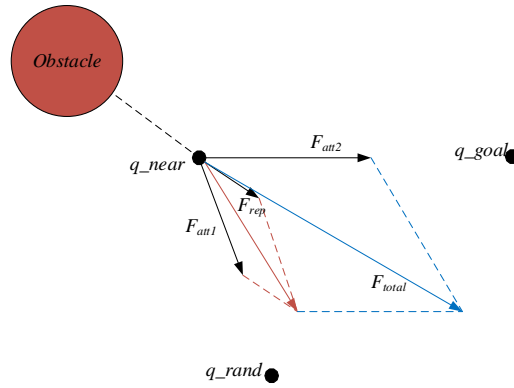


图 3.3 q_{near} 受力示意图

然而，当无人机距离障碍物比较近时，其所受的排斥力会变得很大。当排斥力大于吸引力时，无人机将不能到达目标点。因此，本文改进了势场函数，改进后的势场函数为：

$$U_{att}(p) = \frac{1}{2} k_a \rho_g^2(p, p_{goal}) \quad (3.8)$$

$$U_{rep}(p) = \begin{cases} \frac{1}{2} k_r \left(\frac{1}{\rho(p, p_{obs})} - \frac{1}{\rho_0} \right)^2 \rho_g^n(p, p_{goal}) & \rho(p, p_{obs}) \leq \rho_0 \\ 0 & \rho(p, p_{obs}) > \rho_0 \end{cases} \quad (3.9)$$

其中 p , p_{goal} , p_{obs} 分别代表 q_{near} , q_{goal} 和障碍物的位置。 n 是一个正整数， $\rho(p, p_{obs})$ 是 q_{near} 距其最近的障碍物之间的欧氏距离， $\rho_g(p, p_{goal})$ 是 q_{near} 和 q_{goal} 之间的距离。当无人机靠近障碍物时，障碍物产生的斥力会随着 $\rho_g(p, p_{goal})$ 的减小而变小，这样避免了斥力大于引力的现象。改进后的斥力表达式为：

$$F_{rep}(p) = -\nabla U_{rep}(p) = F_{rep1} n_{OR} + F_{rep2} n_{RG} \quad (3.10)$$

$$F_{rep1} = k_r \left(\frac{1}{\rho(p, p_{obs})} - \frac{1}{\rho_0} \right) \frac{\rho_g^n(p, p_{goal})}{\rho^2(p, p_{obs})} \quad (3.11)$$

$$F_{rep2} = \frac{n}{2} k_r \left(\frac{1}{\rho(p, p_{obs})} - \frac{1}{\rho_0} \right)^2 \rho^{n-1}(p, p_{goal}) \quad (3.12)$$

$n_{OR} = \nabla \rho(p, p_{obs})$ 和 $n_{RG} = -\nabla \rho(p, p_{goal})$ 是两个单位向量, 分别为 q_{near} 和 q_{goal} 的方向向量, 以及 q_{near} 和障碍物之间的方向向量。当无人机接近目标点时, 与其最近的障碍物之间的斥力就会变得很小, 可以保证无人机能够到达目标点。在求解 q_{new} 时, 首先, 需要分别计算 q_{rand} 和 q_{goal} 对 q_{near} 的引力, 然后计算障碍物对 q_{near} 的斥力, 然后将合力 F_{total} 分解为 x 轴和 y 轴两个方向的力, 分别用 F_x , F_y 表示。假设 q_{near} 的坐标为 (x_c, y_c) , 扩展步长由 q_{near} 在 x , y 轴上受到的合力的分量决定。选择每个坐标轴上的合力分量最大值的绝对值作为 F_{max} , 则无人机的扩展步长可以确定为:

$$\begin{cases} k = \frac{l}{F_{max}} \\ l_x = kF_x, \quad l_y = kF_y \end{cases} \quad (3.13)$$

l 是无人机的扩展步长, k 是比例系数, 由合力分量最大值与 l 确定, l_x 和 l_y 分别为无人机在 x , y 轴上的前进分量, 这样就可以得到 q_{new} 的坐标, 令 q_{new} 的坐标为 (x_n, y_n) , 则 q_{new} 的坐标方程为:

$$\begin{cases} x_n = x_c + l_x \\ y_n = y_c + l_y \end{cases} \quad (3.14)$$

3.1.3 无人机航迹约束

在无人机航迹规划过程中, 生成航迹的可行性是建立在无人机能够满足自身性能参数条件下的, 除了要避免与环境中的障碍物碰撞, 保证无人机飞行安全, 还需要满足无人机自身的性能指标。无人机的机身结构和能耗等因素决定了无人机的物理性能约束, 为了保证无人机正常飞行, 在飞行过程中需要满足基本的航迹约束条件。通常情况下, 无人机的航迹约束包括最小航迹段约束, 最大偏航角约束, 最大俯仰角约束以及最大航迹长度约束, 接下来将分别对这些航迹约束条件进行定义。首先, 定义一条航迹共有 m 个航迹点, 定义 $q_{i-1}, q_i, q_{i+1} (i=2 \dots m-1)$ 为航迹中的连续三点, 定义三点坐标 $q_i = (x_i, y_i)$, $q_{i-1} = (x_{i-1}, y_{i-1})$, $q_{i+1} = (x_{i+1}, y_{i+1})$ 。

(1) 最小航迹段约束: 无人机的航迹是由若干航迹段组成, 从起点到终点由于障碍物的存在无法使规划的航迹成为直线, 需要绕过障碍区, 因此在无人机进行转弯飞行时, 为保证机身安全, 需要继续飞行一定长度进行缓冲。缓冲距离随着转弯角度变化而变化, 但是由于机身参数限制, 该距离存在一个最小值, 每小段航迹长度都应该大于这个距离。在二维航迹规划中, 最小航迹段长度定义为 l_{min} , 无人机航迹段的长度 $\overline{q_{i-1}q_i}$ 表示为 l_i ,

最小航迹段约束可以表示为:

$$\begin{cases} l_i = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \\ l_i \geq l_{\min} \end{cases} \quad (3.15)$$

(2) 最大偏航角约束: 最大偏航角是指在无人机以地面坐标系 XOY 平面为基准, 飞行速度不变, 无人机转弯前和转弯后方向之间的夹角, 是无人机能否正常飞行的主要评价指标之一。如果无人机在水平方向转弯的角度大于最大偏航角, 将会影响无人机自身安全。无人机的偏航角影响转弯半径, 偏航角大小与转弯半径正相关。为了保证飞行安全, 无人机需要保证转弯时, 偏航角小于设定的最大偏航角。已知连续三航迹点为 q_i 、 q_{i-1} 、 q_{i+1} , 将第 i 段航迹在地面坐标系 XOY 面水平投影定义为 $\vec{a}_i = [x_i - x_{i-1}, y_i - y_{i-1}]^T$, 将无人机允许最大偏航角定义为 ϕ_{\max} 。第 i 个航迹点无人机偏航角为航迹段 $\overrightarrow{q_{i-1}q_i}$ 与 $\overrightarrow{q_iq_{i+1}}$ 夹角, 表示为 ϕ_i , 最大偏航角约束表示为。

$$\begin{cases} \phi_i = \arccos \left(\frac{(x_i - x_{i-1})(x_{i+1} - x_i) + (y_i - y_{i-1})(y_{i+1} - y_i)}{\sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}} \right) \\ \phi_i \leq \phi_{\max} \end{cases} \quad (3.16)$$

(3) 最大航迹长度: 无人机受最大航程的约束, 在无人机飞行过程中, 飞行轨迹并不全是直线距离, 需要避开规划空间的障碍物, 因此飞行路径是曲折的, 这样无形中加大了飞行的起点和终点间的直线距离。由于无人机自身携带燃料容量受限, 飞行时间不能过长, 因此飞行距离受到约束, 在进行无人机航迹规划任务时, 需要使规划的航迹尽可能短。将无人机能够飞行的最远距离定义为 L_{\max} , 由(1)可知, 航迹段长度为 l_i , 则无人机飞行航迹总长度为 L , 则无人机最大航迹长度约束为。

$$\begin{cases} L = \sum_{i=1}^m l_i \quad (i = 2, \dots, m) \\ L \leq L_{\max} \end{cases} \quad (3.17)$$

3.1.4 改进 RRT 算法流程

Step1: 算法参数初始化, 定义无人机航迹规划起始点 q_{start} 和目标点 q_{goal} , 最大迭代次数 M , 目标偏置阈值 $\delta, \delta \in [0, 1]$ 。

Step2: 选取随机采样点 q_{rand} 。根据公式(3.1)选取随机采样点, 当 $\text{rand}() > \delta$ 时, 算法在空间中随机选取采样点。反之, 选择目标点 q_{goal} 为随机采样点。

Step3: 选取最近邻点 q_{near} 。分别计算随机采样点 q_{rand} 与搜索树内节点的欧

氏距离，距离最小的点选取为最近邻点 q_{near} 。

Step4: 计算新节点 q_{new} 。分别计算目标点 q_{goal} 、随机采样点 q_{rand} 对最近邻点 q_{near} 的引力，以及障碍物对 q_{near} 的斥力，根据公式(3.13)，公式(3.14)计算得到新节点 q_{new} 。

Step5: 判断 q_{new} 是否与环境中的障碍发生碰撞。若无碰撞，将 q_{new} 加入到搜索树中，并计算起始点到 q_{new} 的距离。若碰撞，转到 Step2。

Step6: 判断 q_{new} 是否到达目标点。若 q_{new} 与 q_{goal} 之间的距离小于设定值，则表明到达目标点。反之，转到 Step2。

Step7: 从目标点开始，在搜索树中逐点向上回溯，直到起始点。输出航迹点集合，航迹规划完成。

改进 RRT 算法流程图如图 3.4 所示。

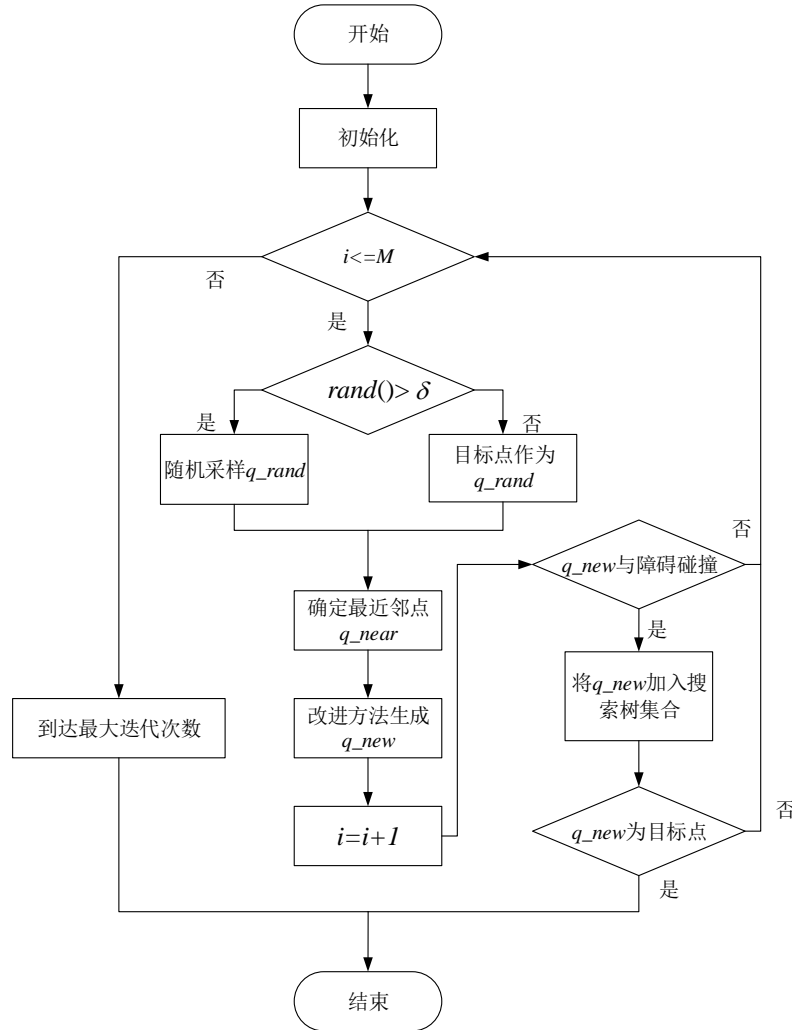


图 3.4 改进 RRT 算法流程图

3.2 基于遗传算法的航迹优化

遗传算法是一种通过模拟自然演化过程来寻找最优解的方法，通常从需要求解的问题中找到多种可能的解决方案，形成一组解集。然后，将该解集视为一个种群，模拟生物进化中的自然选择和遗传机制。通过适应度函数评价每个个体，经过选择，交叉，变异等操作，最终获得问题的最优解。本文将改进后的 RRT 算法生成的一组航迹点作为初始种群。基于遗传算法的航迹优化过程如下，算法流程图如图 3.5 所示：

Step1: 初始化种群。将改进算法生成的一条完整航迹作为种群中的个体，其中的航迹点对应个体中的基因。初始种群由 10 条改进算法生成的航迹组成。

Step2: 选择。这是在输入种群中选择具有高活力的个体的过程。用适应度函数作为评价标准，计算每个个体的适应度值。高适应度的个体更有可能被选择并产生一个或多个后代，个体种群的数量保持不变。

Step3: 交叉。模拟生物进化过程中的繁殖过程，通过基因串的交流，产生新的个体。交叉概率决定个体之间是否基因交换。在基因总数范围内随机生成一个整数，作为个体之间的单点交叉起点，交叉体现了信息交换的概念。

Step4: 变异。模拟自然界中的基因突变现象，它可以以很小的概率改变个体中任何位置的基因。如果只有选择和交叉，将不能生成初始种群外的个体，可以找到局部最优解，但不能得到全局最优解。一旦发生突变，突变位置将随机生成一个航迹点来替换原来的航迹点。

Step5: 通过 Step2-Step4，算法生成新的种群。分别计算新种群中每个个体的适应度函数值。如果个体间的适应度差小于一个阈值，则认为已找到最优解，输出最优个体。否则，回到 Step2 进行下一轮计算，直到满足阈值条件。

适应度函数直接影响收敛速度，是确定群体中个体优劣程度的指标。本文适应度函数由路径长度函数、路径平滑函数和路径安全函数组成。

$L(p)$ 是路径总长度，由 3.1.3 节可知，路径长度函数定义为 $L(p) = \sum_{i=1}^m l_i \quad (i = 2, \dots, m)$,

$S(p)$ 是航迹的平滑度，是根据航迹段之间的夹角之和来判断航迹的平滑性，示意图如图 3.6 所示，航迹段 $\overline{l_{i-1}l_i}$ 与 $\overline{l_i l_{i+1}}$ 的夹角即为 $A(l_{i-1}, l_i)$ ， $A(l_{i-1}, l_i)$ 由公式(3.16)计算获得，其角度数值均取正值。路径平滑函数定义为：

$$S(p) = \sum_{i=1}^{m-1} A(l_{i-1}, l_i) \quad (3.18)$$

$R(p)$ 代表航迹的安全度， D_i 是第 i 个航迹点与距离它最近的障碍的距离。路径安全函数定义为：

$$R(p) = \sum_{i=1}^{m-1} D_i \quad (3.19)$$

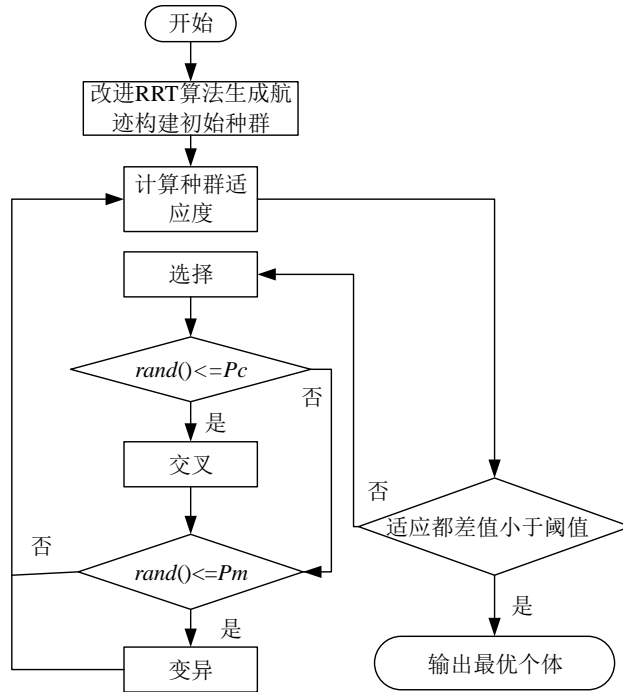


图 3.5 遗传算法的航迹优化流程图

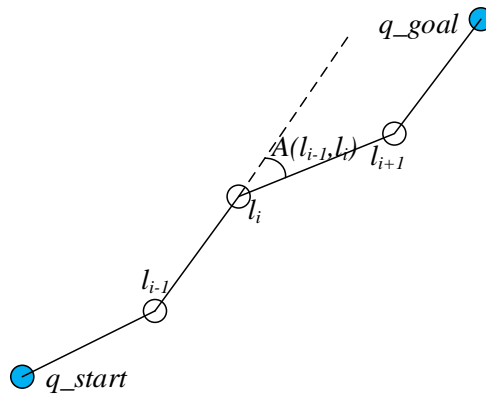


图 3.6 平滑度示意图

适应度函数可以表示为:

$$fit(p) = \frac{1}{\alpha \cdot L(p) + \beta \cdot S(p) + \gamma \cdot R(p)} \quad (3.20)$$

$fit(p)$ 是适应度函数值, α 、 β 和 γ 分别代表着路径长度, 路径平滑度和路径安全性

的权重因子。通过适应度函数分别评估每个个体。选取个体中适应度函数值最大的个体作为最佳个体,交叉概率设置 P_c 为 0.6,突变概率 P_m 设为 0.01。

3.3 仿真分析

本文使用 MATLAB 来验证该改进算法的有效性。将本文算法分别与原始 RRT 算法和只加入目标引力 RRT(tRRT)算法进行比较。将算法分别在简单环境和复杂环境下进行了验证,障碍物分布信息如表 3.1, 3.2 所示。

表 3.1 简单环境下障碍物信息

序号	位置/km	半径/km
1	(50,30)	10
2	(50,80)	10

仿真地图设置为 $100km*100km$ 的二维空间。起点坐标设置为(1km, 1km), 目标点设置为(90km, 90km)。地图中的红色圆圈代表着障碍物, 无人机不能在该区域通过。简单环境中的仿真结果如图 3.7 所示, 蓝色线条表示随机树的枝条, 红色线条为算法规划出的无人机可行航迹, 本文将随机搜索树生成一条树枝定义为算法进行一次迭代。

表 3.2 复杂环境下障碍物信息

序号	位置/km	半径/km
1	(50,30)	10
2	(20,40)	10
3	(50,80)	10
4	(40,50)	10
5	(80,40)	10
6	(70,70)	10

在图 3.7(a)中, 由于原始 RRT 算法随机采样, 地图中存在大量冗余节点, 生成的航迹存在较多拐点。图 3.7(b)是 tRRT 算法的仿真结果, 与原始 RRT 算法相比, 随机树的生长方向更具有指向性, 航迹中的拐点数减少, 加快了原算法的搜索速度。虽然 tRRT 算法提高了搜索效率, 但是, 由于缺少障碍物的斥力, 该算法在障碍物周围出现冗余点。图 3.7(c)为本文算法, 由图可知障碍物作用的加入, 障碍物周围的多余树节点显著减少, 与 tRRT 算法相比, 搜索效率更高, 无人机能更快到达目标点。

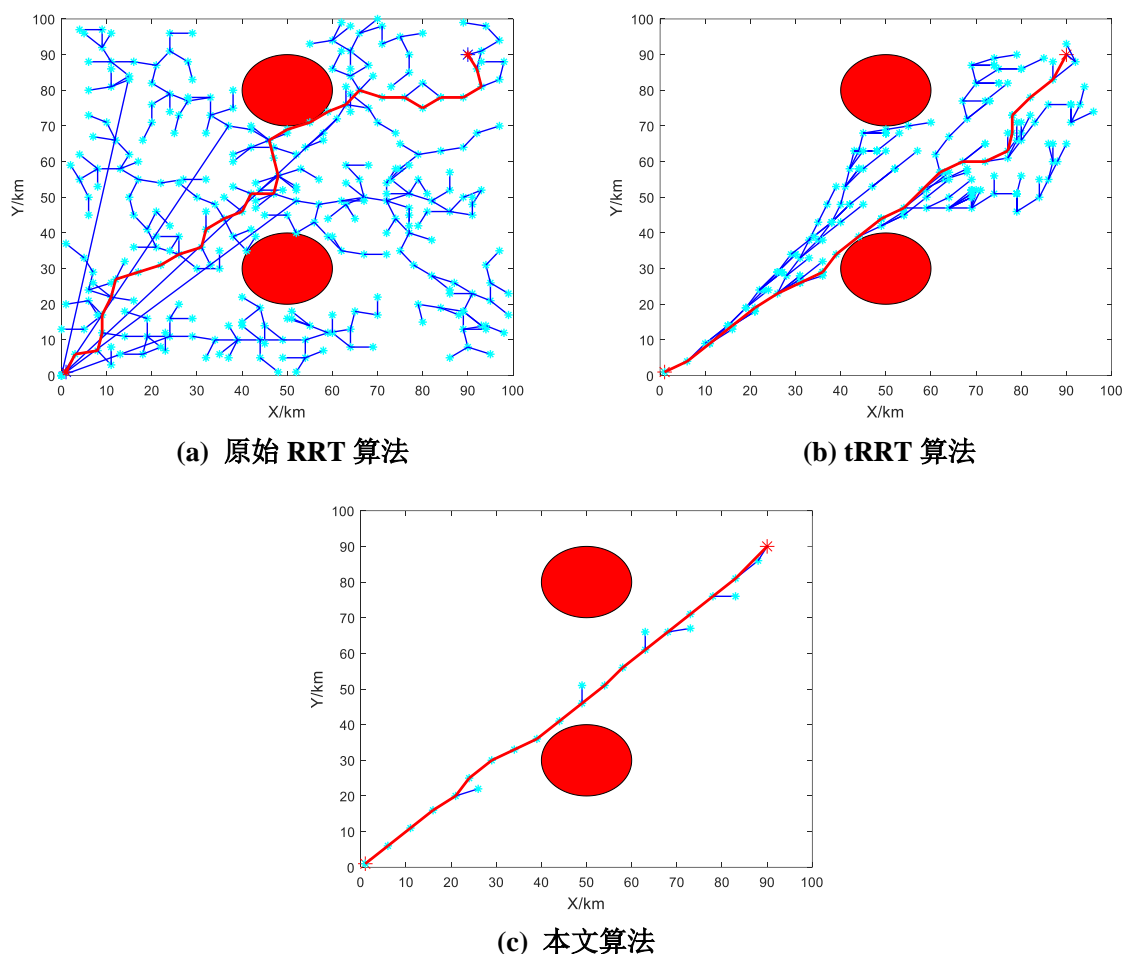


图 3.7 简单环境下仿真

在障碍物多的复杂环境下，仿真结果如图 3.8 所示。随着环境变得越来越复杂，三种算法的树节点的数量就会增加，这表明迭代的次数就会增加。在图 3.8 中蓝色线条表示随机树的枝条，红色线条是无人机的飞行航迹。图 3.8(a)为原始 RRT 算法规划航迹，由于全局采样，导致搜索树探索整个空间，且生成的航迹较为曲折，不满足无人机的飞行要求，图 3.8(b)为 tRRT 算法生成航迹，与图 3.8(a)相比，目标偏置缩小了搜索树的搜索范围，目标点的引力作用加快了搜索树向目标点的生长速度。图 3.8(c)为本文算法生成航迹，在目标偏置和目标引力的基础上，本文算法增加了障碍物的斥力作用，与图 3.8(b)相比，减少了障碍物周围无效树节点的数量，提高了生成航迹质量。通过两种环境下算法的对比仿真，结果表明本文算法极大提高了算法的搜索效率，生成的节点数量和迭代次数均有显著减少。

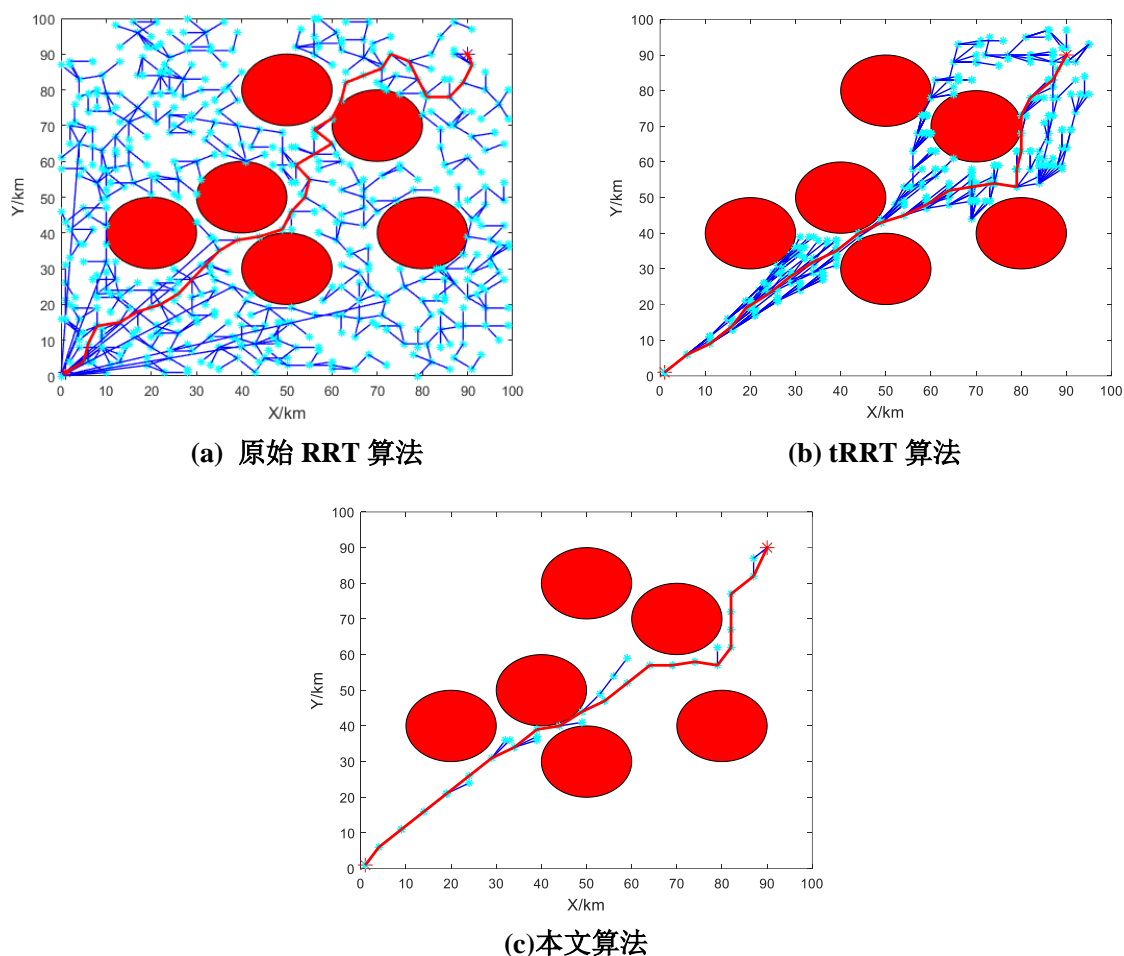


图 3.8 复杂环境下仿真

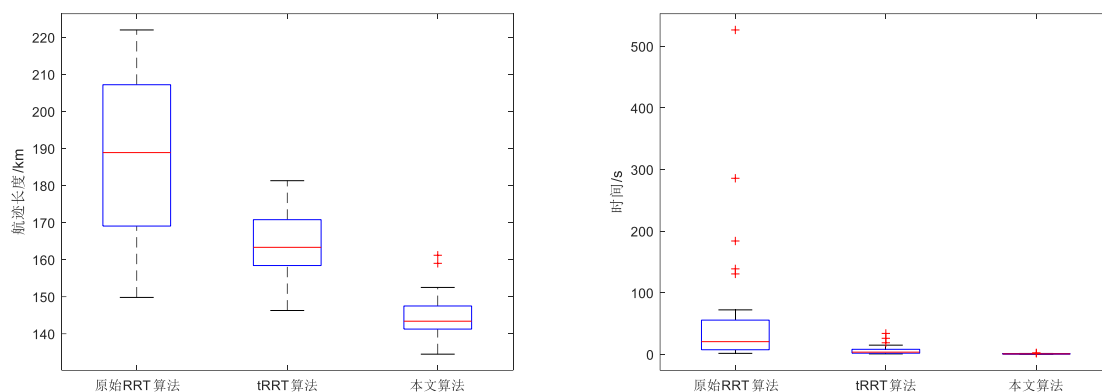
为了验证算法的稳定性, 本文对每个算法在复杂环境下进行 30 次实验, 记录算法的运行时间、生成航迹的长度和迭代次数。实验均值及对比结果如表 3.3 和图 3.9 所示。

表 3.3 30 次实验平均值

算法	平均航迹长度/km	平均运行时间/s	平均迭代次数/次
原始 RRT 算法	188.8892	20.5853	782
tRRT 算法	163.3237	3.5687	273
本文算法	143.3895	0.8306	68

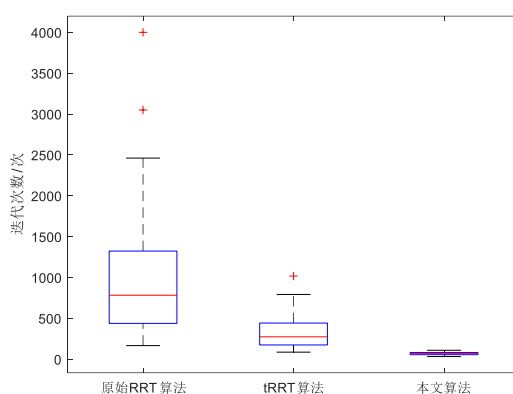
图 3.9 中, 红色加号为每组实验数据中的异常值, 在计算均值时, 为了减少误差, 已将异常值剔除。通过分析图 3.9 和表 3.3 数据, 与原始 RRT 算法相比, 本文算法生成的平均航迹长度减少了 24.08%, 平均运行时间减少了 95.97%, 平均迭代次数减少了 91.30 %。与 tRRT 算法相比, 航迹长度均值减少了 12.21%, 运行时间均值减少了 76.73%,

迭代次数均值减少了 75.10%，这表明本文算法能够加快算法搜索速度，减少迭代次数，提高算法生成航迹质量，同时证明了算法的稳定性。



(a) 航迹长度比较

(b) 运行时间比较



(c) 迭代次数比较

图 3.9 仿真数据比较

在图 3.8(c)中，虽然改进算法生成了一个可行的航迹，但在航迹中仍然存在冗余点，路径的曲率也不能满足无人机的飞行条件。因此，生成航迹需要进一步平滑处理。本文将改进算法生成的航迹作为遗传算法输入种群，经过选择，交叉，变异获得了一条较优航迹，航迹长度与改进算法的平均航迹长度进行比较，为了验证优化算法有效性，在已有图 3.8 环境下，新增一种环境对优化算法进行验证，图 3.8 环境称为环境 1，新增环境称为环境 2，仿真结果及优化前后航迹长度对比分别如图 3.10、图 3.11 和表 3.4 所示。

表 3.4 优化前后航迹长度值对比

仿真环境	优化前	优化后
环境 1	144.3232km	133.7346km
环境 2	133.9250km	130.5890km

图 3.10(a)(b)分别为环境1时优化算法生成的较优航迹和适应度差值曲线图，图 3.11(a)(b)为环境2时算法优化生成的航迹和适应度差值曲线。为了满足无人机的飞行要求，使用三次 B 样条法对生成的航迹进行平滑处理，蓝色线条为遗传算法优化后生成的较优航迹，绿色线条为平滑后的无人机飞行航迹。本文将改进 RRT 算法生成的航迹点序列集合作为输入种群，由图 3.10(b)和图 3.11(b)可知，环境1时，遗传算法经过6次进化完成了优化，环境2时，13次进化完成了优化，这表明初始种群的质量能够缩短遗传算法的运行时间。由表4数据可知，优化后的航迹长度环境1较之前减少了7.34%，环境2减少了2.49%，获得了较优航迹。

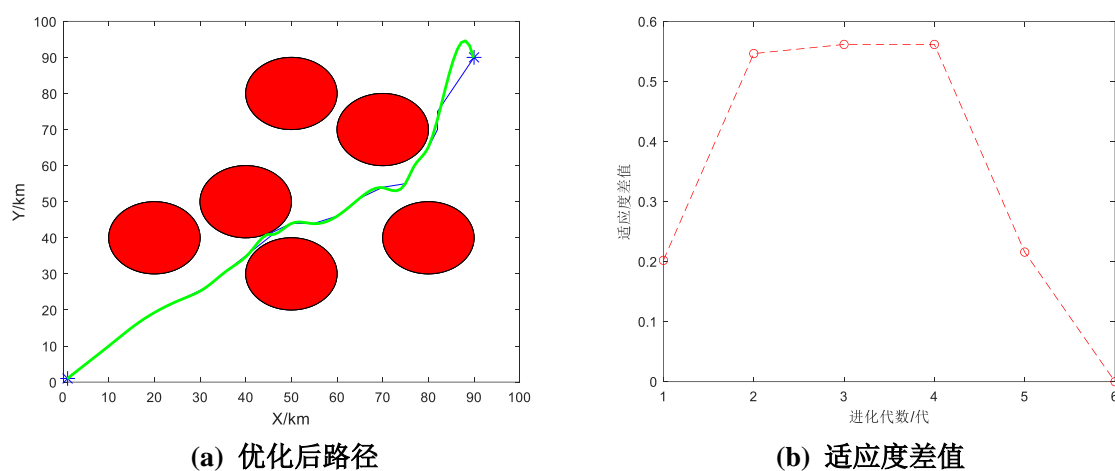


图 3.10 环境1遗传算法仿真结果

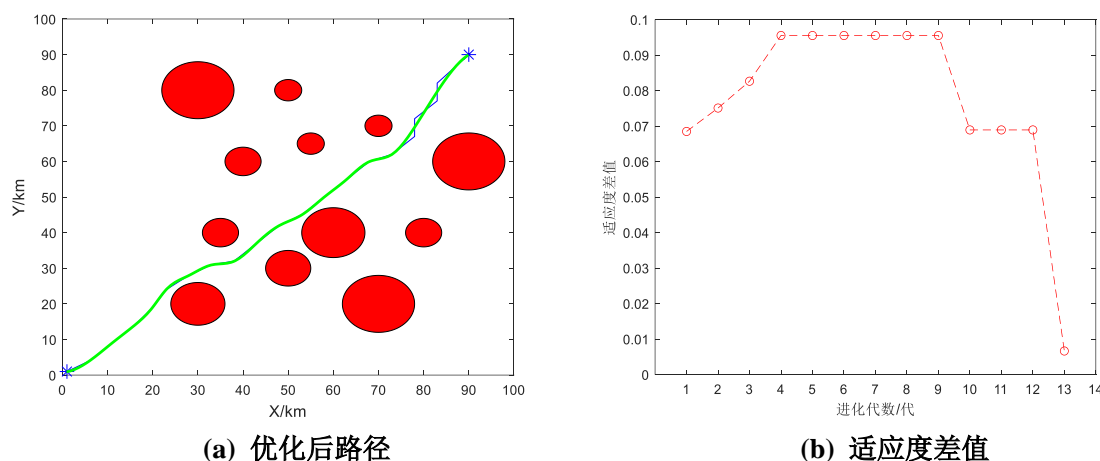


图 3.11 环境2遗传算法仿真结果

3.4 本章小结

本章提出了一种将改进 RRT 算法、人工势场法及遗传算法相融合的无人机航迹规

划方法。首先，针对原始 RRT 算法规划速率较低的问题，引入目标偏置改变 q_{rand} 点的选取方式，目标点有一定概率作为采样点，提高了搜索效率。然后优化 q_{new} 的生成方式，目标点和 q_{rand} 对 q_{near} 产生引力，障碍物对 q_{near} 产生斥力，将三者的合力方向作为新节点的生成方向，最后获得无人机可行航迹。针对所生成航迹冗余点过多，路径曲率不满足飞行要求的问题，采用遗传算法对生成的航迹进行优化，将改进后的 RRT 算法生成的一组路径点序列作为遗传算法的初始种群，通过选择、交叉和变异操作获得种群的较优无人机飞行航迹，解决了路径随机性较大问题。仿真结果表明改进算法极大提高了算法的搜索效率，生成的节点数量和迭代次数均有显著减少，经过遗传算法优化之后的航迹长度减少，获得较优航迹。

第4章 移动威胁下的无人机动态航迹规划

无人机航迹规划一直是无人机系统研究中的重要内容，航迹规划是无人机安全飞行的重要保证，可以分为静态航迹规划和动态航迹规划。静态航迹规划是指在已知障碍物信息的环境下，使用算法生成一条满足无人机飞行要求的航迹，也可以叫做全局航迹规划。当环境中出现移动威胁时，则需要进行局部航迹规划，这也称为动态航迹规划。随着科学技术的不断发展和无人机飞行环境的日益复杂，全局航迹规划已不能满足复杂的环境需求。在无人机飞行的过程中，不仅仅只存在已知静态威胁，还有可能会遇到移动威胁，如敌方的无人机、地对空导弹等，具有移动威胁的动态环境可以更好地映射无人机在真实环境中飞行情况。因此，如何同时规避静态和移动威胁的航迹规划是一个重要研究内容。针对这一问题，许多学者针对规避移动威胁的路径规划算法进行了研究。例如模型预测控制算法(MPC)、动态窗口法等算法，其中 MPC 法被用来进行无人机动态航迹规划，首先获得一条全局航迹，根据约束条件和无人机预测模型建立目标函数，通过滚动检测实时获取障碍物位置信息，通过目标函数最小代价确定航迹点的生成，该算法能够规避移动威胁，但是实时性较差。动态窗口法是一种在线路径重规划方法，根据机器人的动力学特性，可能产生的运动轨迹所对应的速度集合构建速度窗口，通过评价函数选择速度值，更新机器人的运动方向。然而该算法设置参数较多，且均对路径生成有较大影响，算法收敛较慢，不适用于无人机动态航迹规划。

本章给出了一种无人机规避移动障碍的动态航迹规划方法。首先，在上一章的基础上，使用改进后的 RRT 算法生成一条无人机可行航迹。然后，当环境中出现移动威胁时，考虑障碍物或无人机的速度、加速度等和运动相关的因素，建立无人机与障碍物的运动模型，本章考虑了四种无人机飞行时与移动威胁产生碰撞的情况，根据无人机和移动障碍速度矢量之间的关系设计了对应的避障策略，最后，通过仿真验证了所提方法的有效性及其实用性。

4.1 移动威胁下的无人机航迹规划流程

本章提出了一种移动威胁下的无人机航迹规划方法。首先，根据已知环境中静态障碍物信息，使用第三章改进 RRT 算法生成一条可行航迹。然后，再用遗传算法对航迹进行处理，获得一条较优航迹。当环境中出现移动障碍时，通过分析无人机与障碍物速度方向、大小之间的关系，考虑了四种无人机可能与障碍物发生碰撞的情况，进一步给出了对应的避障策略，实现了移动威胁下无人机局部路径重规划。无人机动态航迹规划模块图如图 4.1 所示。

Step1: 首先, 初始化环境中障碍物位置信息, 设置航迹规划起点和终点。

Step2: 基于改进 RRT 算法的无人机全局航迹规划, 使用改进 RRT 算法生成一条从起点到终点的航迹。

Step3: 生成的航迹经过遗传算法进行处理, 获得较优航迹。

Step4: 当检测到突发移动威胁时, 分析无人机与移动障碍速度之间的矢量关系, 选择对应的避障策略, 使用速度障碍法进行局部路径重规划。

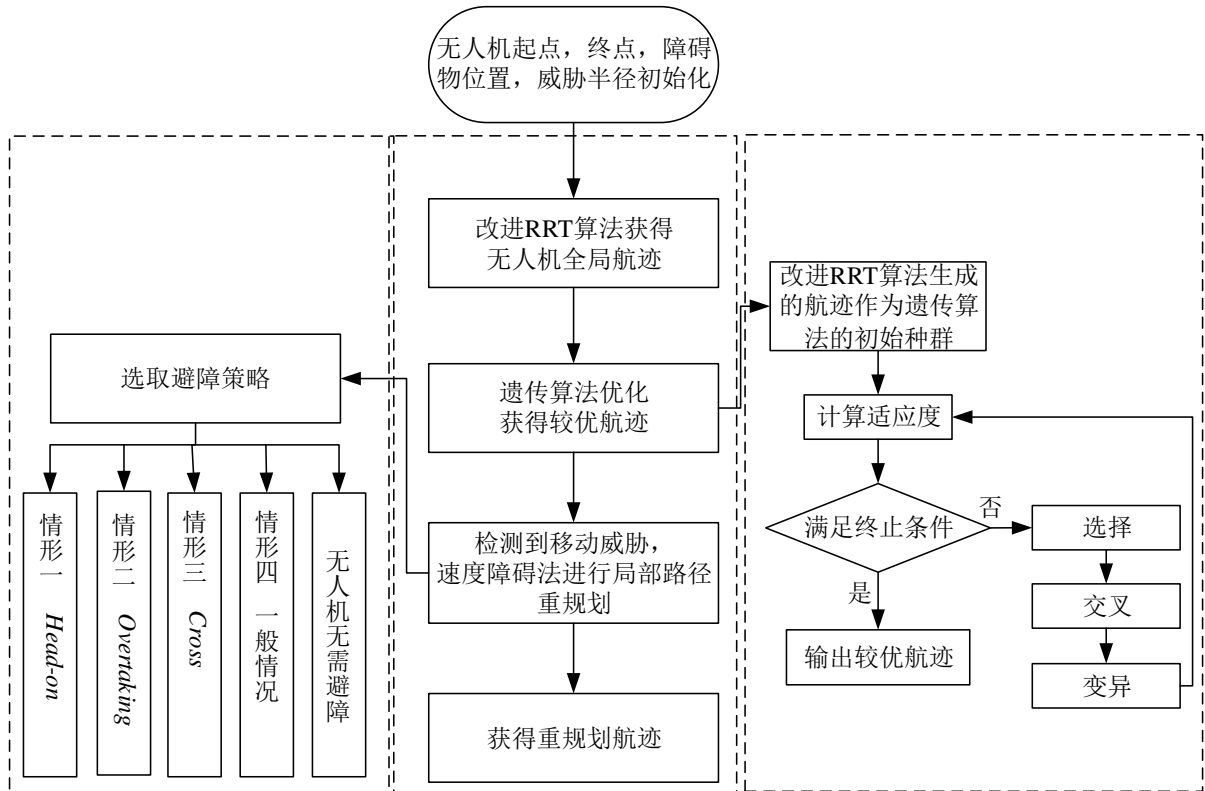


图 4.1 无人机动态航迹规划模块图

4.2 速度障碍法原理

速度障碍法被广泛应用于带有移动障碍物的路径规划, 图 4.2 所示的速度障碍原理图解释了无人机和障碍物速度之间的矢量关系。速度障碍法定义了一个相对速度障碍区域, 将飞机 A 定义为无人机, 速度表示为 v_a , 飞机 B 定义为移动障碍, 速度表示为 v_b , 两者在空间中的运动可以表示为 $R(t) = (x_R(t), v_R(t))$, 其中 $x_R(t) = [x_R, y_R]$ 代表两者在 t 时刻的位置坐标, $v_R(t)$ 代表在 t 时刻的速度, 相邻时间间隔为定义为 Δt , 并假设在 Δt 内两者的速度保持不变, 由图 4.2 可知, A 相对于 B 的速度 v_{ab} 指向锥形区域 $CC_{a,b}$, 红色虚线由 λ_{ab} 表示。当相对速度指向该区域时, 可以认为在有限的时间内, 无人机将与障碍

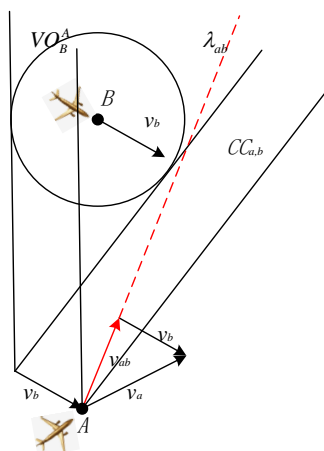


图 4.2 速度障碍法原理图

在计算速度障碍时,为了方便计算,将所有运动物体都看作质点。为了构建碰撞锥,将移动障碍 **B** 映射到无人机的速度空间中,由相对关系,碰撞锥的定义如下:

$$CC_{a,b} = \{v_{ab} \mid \lambda_{ab} \cap B_{ab} \neq \emptyset\} \quad (4.1)$$

$CC_{a,b}$ 碰撞锥定义了无人机在未来碰撞情况下, 无人机与移动障碍物相对速度的集合, 即相对碰撞区。当环境中存在多个移动障碍物时, 就需要用到绝对碰撞区, 定义为:

$$VO_R^A = CC_{a,b} \oplus v_R \quad (4.2)$$

其中 \oplus 表示为闵可夫斯基和，区域 VO_B^A 由相对碰撞区沿 v_b 进行平移，也就是VO(速度障碍)。故只要保证 v_{ab} 在绝对碰撞区之外就可以避免碰撞。本章只考虑环境中出现一个移动威胁的情况。

$$v_{ab} \notin VO_B^A \quad (4.3)$$

4.3 移动障碍规避策略

4.3.1 无人机与移动障碍运动模型

在无人机飞行过程中，如何准确检测障碍物，并避免碰撞是非常重要的。在本文中，必须满足以下障碍检测条件：

$$\| (x_u, y_u) - (x_{obs}, y_{obs}) \| \leq r_{obs} \quad (4.4)$$

在公式(4.4)中, (x_u, y_u) 代表无人机的位置, (x_{obs}, y_{obs}) 代表障碍物的中心位置。当满足公式(4.4)的条件时, 这表明无人机当前位置会与障碍物发生碰撞。因此, 该航迹点不可达, 而无人机将重新计算新节点位置。

在无人机的实时飞行中，必须考虑到移动的障碍。将每个时间步长位置发生变化的移动障碍物设定为二维空间中的点，离散时间下移动障碍物动力学坐标方程如下：

$$\begin{aligned} x_{obs}(k+1) &= x_{obs}(k) + V_{obs} \cos \vartheta \\ y_{obs}(k+1) &= y_{obs}(k) + V_{obs} \sin \vartheta \end{aligned} \quad (4.5)$$

其中 V_{obs} 和 ϑ 是移动障碍物的速度和航向角。在笛卡尔坐标系中，无人机在 t 时刻的速度定义为 V_u ，航向角为 ι ，运动状态定义为 $U(t) = (P_u(t), V_u(t))$ ， $P_u(t) = [x_u, y_u]^T$ 是无人机的位置坐标，则离散时间下无人机的动力学坐标方程为：

$$\begin{aligned} x_u(k+1) &= x_u(k) + V_u \cos \iota \\ y_u(k+1) &= y_u(k) + V_u \sin \iota \end{aligned} \quad (4.6)$$

4.3.2 障碍规避策略

假定无人机与移动障碍物之间的速度矢量关系保持不变。点 O 是障碍物位置，定义为 $P_o(t) = [x_o, y_o]$ ，速度定义为 V_o ，威胁半径为 R 。根据无人机与障碍物速度矢量之间的夹角 η 的大小，设计了 4 种避障情况并给出了相应的避障策略。

情形 1：在图 4.3(a)中，无人机与障碍的速度夹角 η 为 π ，可以定义为无人机飞向障碍物，称为 *Head-on*。在这种情况下，为了躲避障碍物，无人机需要改变原来的飞行方向。避障示意图如图 4.3(b)所示，无人机在障碍物一侧飞行完成障碍规避。

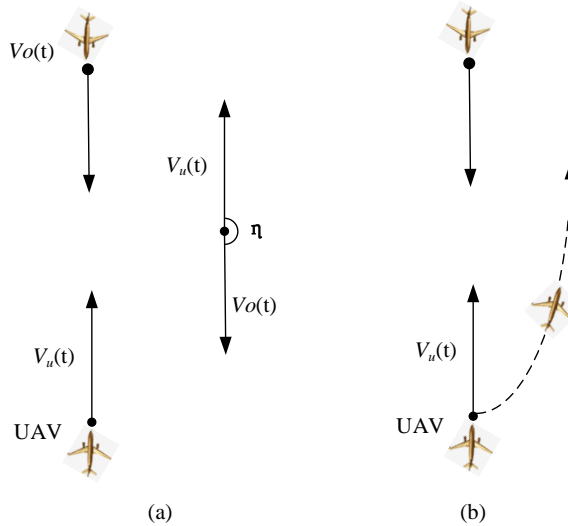


图 4.3 *Head-on* 避障情况及策略

情形 2：在图 4.4(a)中，无人机和障碍物速度之间的夹角 η 为 0。此时无人机和障碍物飞行的方向相同，此情形被称为 *Overtaking*。在这种情况下，根据无人机和障碍物速

度大小的关系,有两种避障情况,当无人机的飞行速度小于障碍物,即 $V_u(t) < V_o(t)$,此时,无人机无需执行避障操作,保持原来飞行方向。当无人机速度大于障碍物,即 $V_u(t) > V_o(t)$,无人机必须在障碍物一侧绕行,以避免碰撞,避障路线如图 4.4(b)的虚线所示。

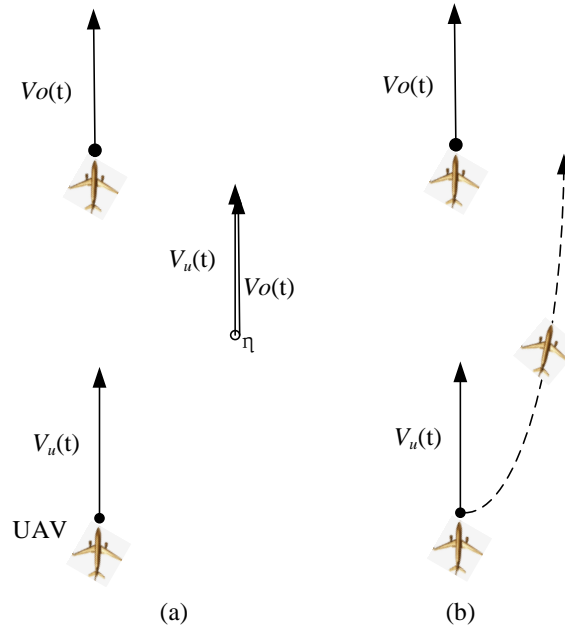


图 4.4 Overtaking 避障情况及策略

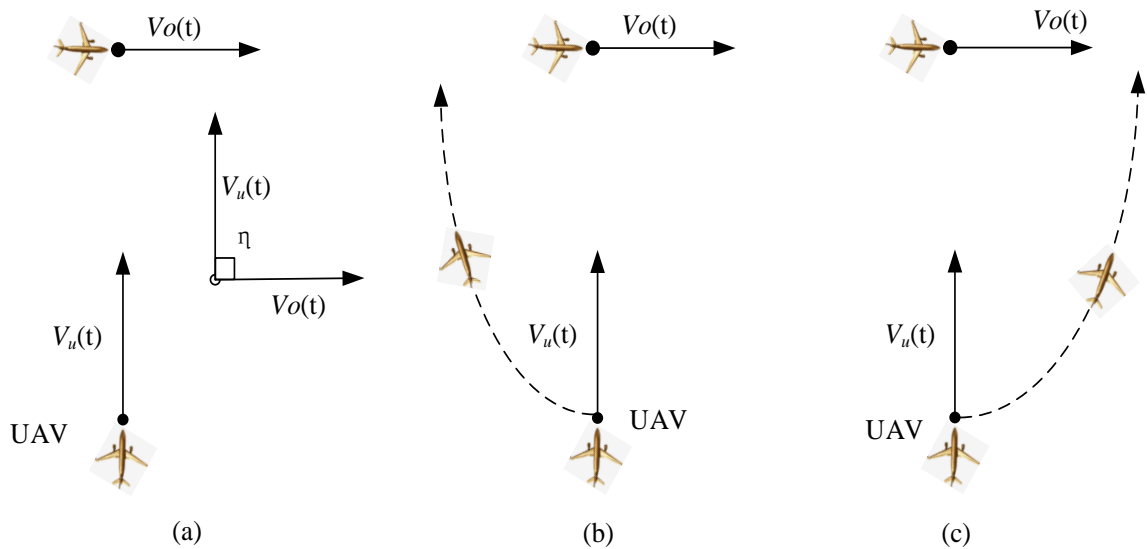


图 4.5 Cross 避障情况及策略

情形 3: 在图 4.5(a)中,无人机和障碍物之间的夹角 η 为 $\pi/2$ 。无人机和障碍物飞行

的方向相同，此情形被称为 *Cross*。此情形分为两种避障策略，当 $V_u(t) < V_o(t)$ 时，如图 4.5(b) 所示，为了避免与障碍物相撞，无人机在障碍物后面飞行完成规避。相反，无人机速度高于障碍物，即 $V_u(t) > V_o(t)$ ，如图 4.5(c) 虚线所示，无人机将在障碍物前方飞过完成避障。

情形 4：在图 4.6 中，无人机与移动障碍物速度方向夹角满足 $0 < \eta < \pi/2$ ，此时，无人机未来将与障碍物在图中红星处相撞。 \mathbf{PB} 和 \mathbf{AB} 分别代表无人机的速度 V_u 和障碍物的速度 V_o 。向量 \mathbf{PA} 即 V_{uo} 是向量 \mathbf{PB} 和 \mathbf{AB} 的矢量差，它的射线被定义为 l_{uo} ，障碍物的威胁半径为 r ，障碍物的威胁范围定义为图 4.6 中的圆形区域 PO ，碰撞锥由点 P 与 PO 的切线构成。当 $l_{uo} \cap PO \neq \emptyset$ 时， l_{uo} 在碰撞锥内，当经过时间 Δt 后，无人机将与障碍物在红星处相撞，此时无人机需要避开移动的障碍物。

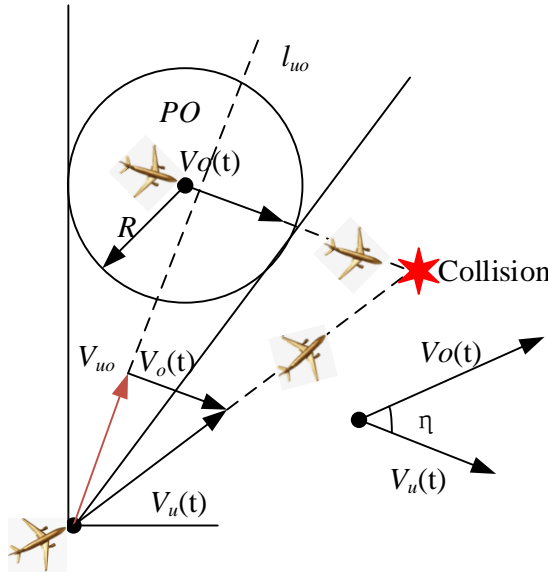


图 4.6 $0 < \eta < \pi/2$ 避障情况

由图 4.7 可知，为了避免与移动障碍物碰撞，无人机的速度方向需要旋转角度 α 来使 l_{uo} 远离碰撞锥 PO ，然而还需要判断无人机速度 V_u 的旋转方向是顺时针还是逆时针。逆时针和顺时针的偏转角 α 设置为 ψ 和 ϕ 。在图 4.7 中， V_u 需要进行顺时针方向的偏转，此时无人机避免碰撞时 V_u 偏转角度 α 计算如下：

$$\hat{V}_{uo} = \frac{\sqrt{r^2 - R^2}}{t} \quad (4.7)$$

$$\psi = \arccos \frac{V_u^2 + V_o^2 - \hat{V}_{uo}^2}{2V_u V_o} - \arccos \frac{V_u^2 + V_o^2 - V_{uo}^2}{2V_u V_o} \quad (4.8)$$

$$\varphi = \arccos \frac{V_u^2 + V_o^2 - V_{uo}^2}{2V_u V_o} - \arccos \frac{V_u^2 + V_o^2 - \hat{V}_{uo}^2}{2V_u V_o} \quad (4.9)$$

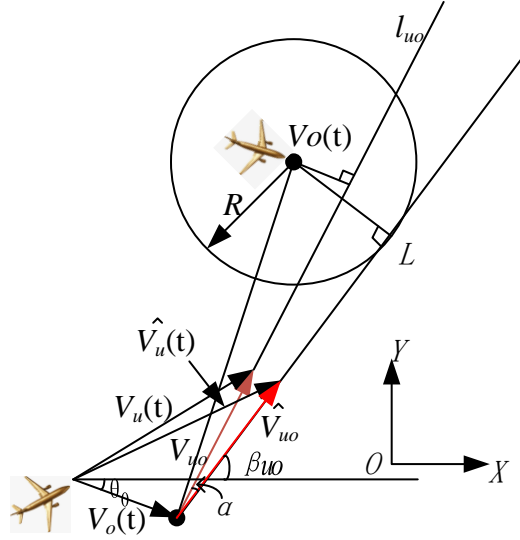
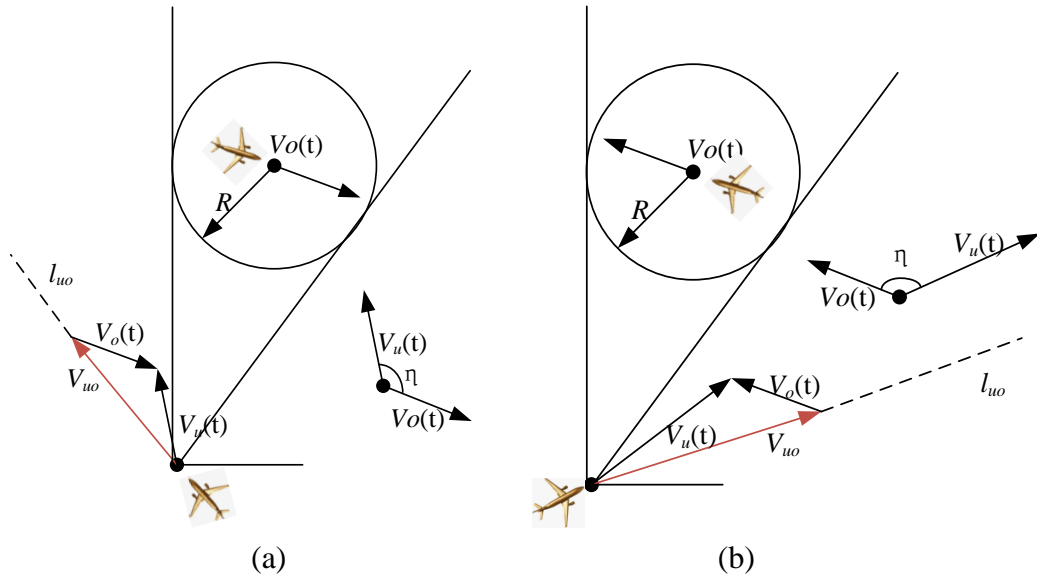


图 4.7 无人机障碍规避示意图

在图 4.7 中， OX 为横坐标轴， OY 为纵坐标轴， θ_u 是无人机速度 V_u 和 OX 之间的夹角， θ_o 是障碍物速度 V_o 和 OX 之间的夹角， d 为障碍物中心 O 与无人机相对速度 V_{uo} 之间的最短距离， α 是无人机躲避障碍物时速度转向角度。 β_{uo} 是 PL 和 OX 之间的夹角。


 图 4.8 $\pi/2 < \eta < \pi$ 避障情况

如图 4.8 所示，当 $\pi/2 < \eta < \pi$ ，图 4.8(a)(b) 分别表示两种情况下无人机和障碍物

速度方向之间的关系， l_{uo} 是两者速度矢量差 V_{uo} 方向的射线，由图 4.8 可知，当 η 满足 $\pi/2 < \eta < \pi$ ， l_{uo} 指向碰撞锥外，也就是说相对速度 V_{uo} 没有指向碰撞区域，这说明无人机接下来的飞行会远离碰撞区域 PO ，未来不会与障碍物发生碰撞。

4.4 仿真分析

当飞行轨迹中出现移动障碍物时，无人机无法沿原路径飞行，需要进行局部路径规划。本文采用速度障碍物法对路径进行重新规划，由于无人机与移动障碍物的速度不同，考虑了四种发生碰撞的情况，并设计了相应的解决策略。为了验证所提出策略的有效性，本文使用 MATLAB 软件，进行仿真验证。障碍物分布信息如表 4.1 所示。

表 4.1 障碍物位置信息

序号	位置/km	半径/km	序号	位置/km	半径/km
1	(55,65)	3	7	(50,30)	5
2	(30,80)	8	8	(35,40)	4
3	(70,20)	8	9	(50,80)	3
4	(90,60)	8	10	(40,60)	4
5	(60,40)	7	11	(80,40)	4
6	(30,20)	6	12	(70,70)	3

首先，采用上一章的改进算法生成一条无人机可行航迹，仿真结果如图 4.9 所示。蓝色菱形为初始航迹飞行起点，黑色菱形为初始航迹终点。绿线为第三章改进算法生成的无人机飞行航迹，红色的圆圈代表地图上的静态障碍。

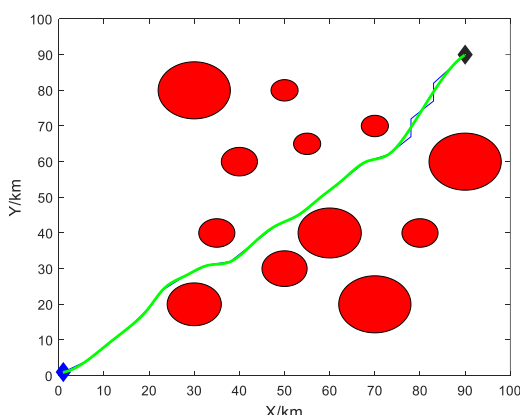


图 4.9 改进算法生成的全局航迹

在图 4.10 中，蓝色星点是一个移动的障碍物，连续的蓝星点代表了移动障碍物的运

行轨迹。名为 *movingobstacle* 的黑点是检测到移动障碍物的位置，也是移动障碍物的运动起点，名为 *Findobstacle* 和 *Avoidobstacle* 的两个黑点分别代表无人机路径重规划的起点和终点。红色星是无人机重规划后新的路径点。在图 4.10 中，无人机与移动障碍物速度方向相反，大小均为 30m/s ，这种情况对应情形 1，无人机与移动障碍物相向而行，由图可知当迎面飞来移动威胁时，无人机沿着连续红色星点飞行，成功规避移动威胁。

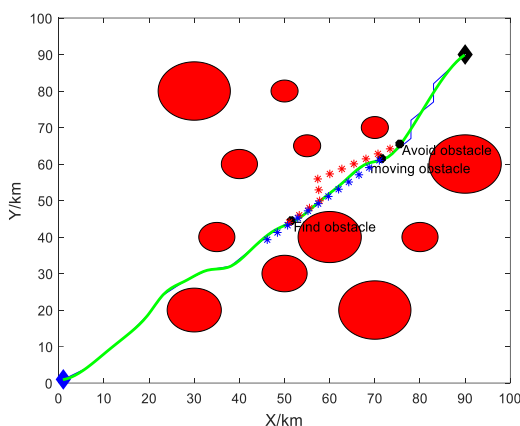


图 4.10 Head-on 情形下避障仿真图

在图 4.11, 4.12 中，无人机与障碍物速度方向相同，对应第二种情况即情形 2，由于两者速度大小的差异，无人机会采用不同的障碍规避策略，在图 4.11 中，无人机速度设定为 15m/s ，障碍移动速度为 30m/s ，此时，无人机速度小于障碍物，所以无人机只需在障碍物后方飞行，不需要避障，但是在图 4.12 中，无人机速度为 30m/s ，障碍物的速度为 15m/s ，无人机飞行速度大于障碍物，此时，无人机会沿着连续红色星点在障碍物侧方飞过，避免与移动威胁碰撞。

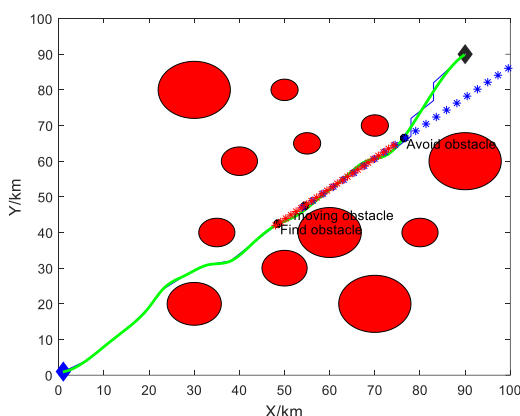


图 4.11 Overtaking 情形下避障仿真图 1

图 4.13 和图 4.14 的仿真结果对应情形 3 即 *Cross* 情况。在图 4.13 中，无人机速度

设定为 $15m/s$ ，障碍物速度设定为 $30m/s$ ，由图 4.13 可知，由于无人机速度小于障碍物，无人机在障碍物后方绕过障碍物完成避障，飞行轨迹如图 4.13 中的连续红色星点所示。在图 4.14 中，无人机速度设定为 $30m/s$ ，障碍物速度设定为 $15m/s$ ，无人机的飞行速度大于障碍物，无人机按照红色星点在障碍物的前方飞过，完成避障。

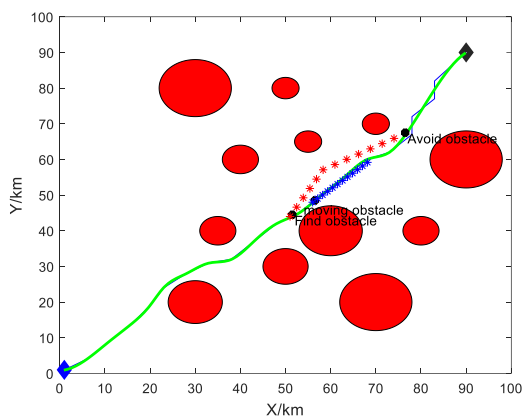


图 4.12 Overtaking 情形下避障仿真图 2

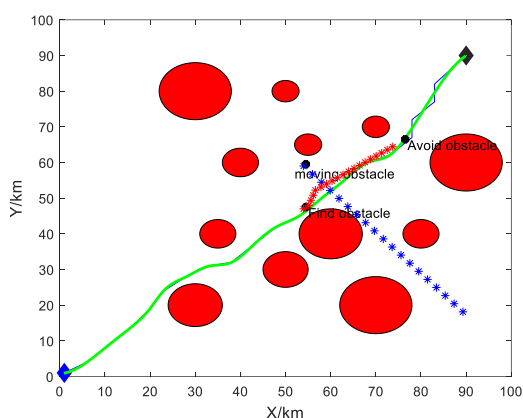


图 4.13 Cross 情形下避障仿真图 1

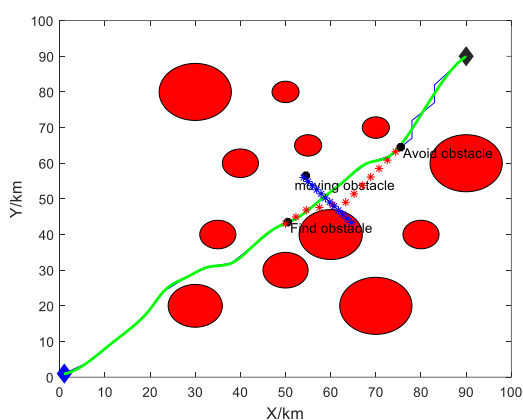


图 4.14 Cross 情形下避障仿真图 2

在图 4.15 中，这是一般情况对应情形 4，无人机与障碍物速度矢量差的方向指向碰

撞锥内,在这种情况下,无人机需要改变方向来避免碰撞,图 4.15 表明,在该种策略下,无人机能够成功规避移动障碍。

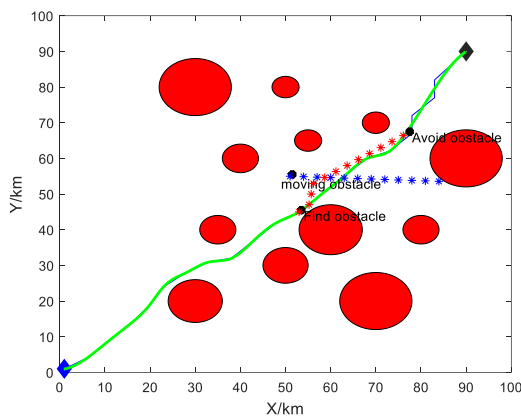


图 4.15 $0 < \eta < \pi/2$ 避障仿真图

4.5 本章小结

本章内容考虑了无人机飞行过程中,环境的移动威胁对航迹的影响,引入了速度障碍法完成局部路径重规划。首先,由第二章的算法获得一条全局较优航迹,当无人机检测到移动威胁时,然后建立移动威胁和无人机的运动模型。考虑环境中出现的移动威胁,根据无人机和移动障碍速度之间大小与方向的关系,设计了四种不同碰撞情形,并给出了相对应的避障策略。采用 VO 法进行移动障碍规避,实现实时航迹规划。仿真结果表明,该算法能够快速规避移动威胁,完成局部路径重规划。

第5章 基于改进 RRT 算法的无人机三维航迹规划

航迹规划是无人机执行任务的关键部分。近年来,针对已有的算法,许多学者对其进行了改进并在二维环境中进行了验证。然而,在实际环境中,无人机飞行过程中需要考虑自身高度问题,避免飞行过低,与建筑物碰撞导致坠毁,同时由于无人机自身性能约束,飞行过程中也要保证机身稳定,俯仰角不会产生大角度变化。尽管改进算法在二维环境中取得了较好效果,但是却没有考虑飞行高度和俯仰角问题。因此,如何将二维航迹规划推广到三维航迹规划具有重要研究意义。在三维航迹规划中,精确的环境建模是航迹规划算法生成安全航迹的重要保证。针对不同的环境,采用不同的建模方法,常见的三维环境建模方法有栅格法、几何建模法和高程建模法。除了环境建模外,还需要注意无人机约束条件的变化,三维环境下,需要加入俯仰角等约束条件。目前,无人机三维航迹规划问题已经引起许多学者的重视。例如,精英蚁群与人工势场法结合的三维航迹规划算法^[64],算法使用了栅格法构建地图环境,但是并未考虑突发威胁情况。量子粒子群三维航迹规划算法^[65],采用几何建模法绘制环境威胁,仿真结果表明该算法在三维空间中比传统粒子群算法搜索效率高。目前,关于无人机三维航迹规划的研究成果较少,本章参考上述三维航迹规划算法中的环境建模方法,结合改进 RRT 算法的约束条件,将改进 RRT 算法扩展到三维空间中,建立了三维环境下无人机的运动模型以及障碍物模型,同时给出了无人机三维航迹规划约束条件,建立了代价函数模型。当环境出现突发威胁时,建立了突发威胁模型,并给出了突发威胁规避策略。仿真结果验证了改进 RRT 算法的有效性,能够快速生成无人机航迹,并成功规避环境中的突发威胁,满足在线重规划要求。

5.1 无人机运动模型和三维环境建模

本文采用地面坐标系和机体坐标系对无人机动态系统进行数学描述,其中地面坐标系以地面固定点为三维坐标原点 O ,无人机每一时刻所处位置用 OX , OY , OZ 轴组成的右手坐标系坐标点表示。无人机的质心作为机体坐标系的原点 O ,机头是 X 轴方向,机身右侧是 Y 轴方向,机腹是 Z 轴方向。无人机飞行的姿态由偏航角,俯仰角描述,偏航角为机体坐标系 X 轴在地面坐标系水平面的投影与地面坐标系 X 轴的夹角,俯仰角为机体轴坐标系的 X 轴与地面坐标系水平面的夹角。

在无人机航迹规划问题中,将飞行中的无人机视作一个质点,则在 t 时刻无人机运动状态用矢量 $X(t)$ 表示:

$$X(t)=[x(t),y(t),z(t),\phi(t),\xi(t)] \quad (5.1)$$

在公式(5.1)中, $x(t), y(t), z(t)$ 为无人机在时刻 t 地面坐标系的位置, $\phi(t), \xi(t)$ 分别为无人机在 t 时刻的偏航角和俯仰角。本文只考虑无人机飞行路径问题, 不对飞行姿态进行定义, 故不考虑滚转角。由质点动力学可得, 在地面坐标系下各物理量的关系如下所示, 其中 V 为无人机的速度。

$$\begin{cases} \dot{x}(t) = V \cos \phi(t) \cos \xi(t) \\ \dot{y}(t) = V \sin \phi(t) \sin \xi(t) \\ \dot{z}(t) = V \sin \xi(t) \end{cases} \quad (5.2)$$

为研究无人机航迹规划问题, 需要建立三维环境模型, 环境模型通常包括地形和障碍区域信息。本文地形建模设置为 $10km \times 10km \times 8km$ 的直角坐标区域; 障碍区域建模采用山峰模型, 数学表达式如下:

$$z(x, y) = \sum_i h_i [\exp(-(\frac{x - x_i}{x_{si}})^2 - (\frac{y - y_i}{y_{si}})^2)] \quad (5.3)$$

式中, h_i 表示第 i 座山峰的高度, (x_i, y_i) 为山峰的中心坐标, x_{si}, y_{si} 分别表示山峰在 x 轴和 y 轴方向的坡度向量, (x, y) 代表地形中每个点投影到平面的点坐标, $z(x, y)$ 表示地形中每个点的高度。

5.2 突发威胁模型的建立

当无人机按照既定轨迹飞行时, 必须要考虑环境中可能出现的突发威胁, 当无人机侦测到突发威胁时, 航迹规划算法必须要有在线重规划的能力。在无人机实际飞行过程中, 突发威胁包含多种情况, 比如敌方雷达探照、防空导弹或是恶劣天气。由于突发威胁的多样性, 很难用具体表达式进行描述, 这里将突发威胁的有效范围简化为空间中的一个球形区域, 突发威胁表达式如下:

$$(x_s - x_{so})^2 + (y_s - y_{so})^2 + (z_s - z_{so})^2 = R_{so}^2 \quad (5.4)$$

其中 (x_s, y_s, z_s) 表示球体表面的点的坐标, (x_{so}, y_{so}, z_{so}) 表示为球体中心点坐标, R_{so} 为突发威胁的影响半径。

5.3 基于改进 RRT 算法的三维航迹规划

5.3.1 无人机三维航迹约束

改进 RRT 算法流程以及航迹约束条件已在第三章说明。在无人机三维航迹规划中, 最大偏航角约束与二维航迹规划一致, 需要注意的是俯仰角约束的加入, 首先将起点坐标定义为 $(x_{start}, y_{start}, z_{start})$, 目标点定义为 $(x_{goal}, y_{goal}, z_{goal})$, q_rand 定义为

$(x_{rand}, y_{rand}, z_{rand})$, q_near 定义为 $(x_{near}, y_{near}, z_{near})$, q_new 定义为 $(x_{new}, y_{new}, z_{new})$, 假设无人机航迹点数量为 n , 将连续三个航迹点定义为 Q_{i-1}, Q_i, Q_{i+1} ($i=2, \dots, n-1$), 航迹点坐标分别定义为 $Q_{i-1}=(x_{i-1}, y_{i-1}, z_{i-1})$, $Q_i=(x_i, y_i, z_i)$, $Q_{i+1}=(x_{i+1}, y_{i+1}, z_{i+1})$ 。

(1) 最小航迹段约束: 最小航迹段长度定义为 l_{\min} , 无人机航迹段的长度 $\overline{Q_{i-1}Q_i}$ 表示为 l_i , 最小航迹段约束为:

$$\begin{cases} l_i = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2} \\ l_i \geq l_{\min} \end{cases} \quad (5.5)$$

(2) 最大偏航角约束: 在三维航迹规划中, 第 i 个航迹点对应的偏航角为航迹段 $\overline{Q_{i-1}Q_i}$ 与 $\overline{Q_iQ_{i+1}}$ 的夹角, 由于只考虑二维平面的偏航角变化, 因此, 该偏航角的求解方式与公式(3.16)一致。

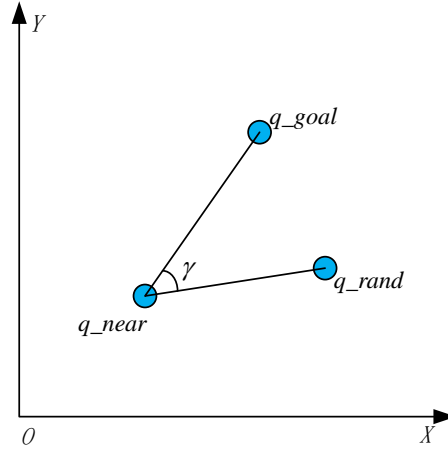
(3) 最大俯仰角角约束: 最大俯仰角是指无人机在地面坐标系 Z 轴方向上升或下降时机体与地面坐标系的夹角, 也是衡量无人机飞行正常与否的主要评价指标。由于无人机自身机动性能限定, 飞行俯仰角变化不能过大, 将无人飞行过程中最大俯仰角值定义为 ξ_{\max} , 第 i 段航迹段的俯仰角定义为 ξ_i , 由(1)可知, 第 i 段航迹可以表示为 $\overline{Q_{i-1}Q_i}$ 。已知两点坐标, 则无人机最大俯仰角约束可以表示为:

$$\begin{cases} \xi_i = \arctan \left(\frac{|z_i - z_{i-1}|}{\sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}} \right) \\ \xi_i \leq \xi_{\max} \end{cases} \quad (5.6)$$

(4) 最大航迹长度: 三维环境下无人机飞行最远航程与二维环境不同, 无人机飞行的航迹段距离为航迹段之和。由(1)可知航迹段长度为 l_i , 则无人机最大航迹约束可以表示为:

$$\begin{cases} L = \sum_{i=1}^n l_i \quad (i=2, \dots, n) \\ L \leq L_{MAX} \end{cases} \quad (5.7)$$

在改进 RRT 算法三维航迹规划中, 为了提高算法的搜索效率, 对 q_rand 的选取范围进行改进, 当生成新的 q_rand 时, 计算 q_rand 和 q_near 之间的连线与目标点和 q_near 之间连线的夹角, 如图 5.1 所示, 将该夹角定义为 γ , 允许的最大值定义为 γ_{\max} , 当选取的 q_rand 不满足 $\gamma \leq \gamma_{\max}$ 时, 重新选取 q_rand , γ 的计算公式为:


 图 5.1 γ 角示意图

$$\gamma = \arccos \left(\frac{(x_{goal} - x_{near})(x_{rand} - x_{near}) + (y_{goal} - y_{near})(y_{rand} - y_{near})}{\sqrt{(x_{goal} - x_{near})^2 + (y_{goal} - y_{near})^2} \sqrt{(x_{rand} - x_{near})^2 + (y_{rand} - y_{near})^2}} \right) \quad (5.8)$$

5.3.2 代价函数模型

航迹代价函数是评价航迹优劣的重要标准。通过建立航迹代价函数将航迹规划问题转换求解最优值问题。结合 5.3.1 节的航迹约束条件，在满足约束条件的前提下，保证航迹长度最小，航迹代价函数定义为：

$$\begin{cases} J = \omega_1 L + \omega_2 \phi + \omega_3 \xi \\ L = \sum_{i=1}^{n-1} l_i \quad (i = 2, \dots, n) \\ \phi = \sum_{i=1}^{n-2} \phi_i \quad (i = 2, \dots, n) \\ \xi = \sum_{i=1}^{n-1} \xi_i \quad (i = 2, \dots, n) \\ \omega_1 + \omega_2 + \omega_3 = 1 \\ L \leq L_{MAX}, \quad \phi_i \leq \phi_{max}, \quad \xi_i \leq \xi_{max} \end{cases} \quad (5.9)$$

在式(5.9)中， J 为本文设定的代价函数值， $\omega_1, \omega_2, \omega_3$ 分别为表达式中各评价指标的权重系数， n 为航迹点个数， L 为无人机总航程， ϕ 为各航迹段偏航角之和， ξ 为各航迹段俯仰角之和。约束条件保证航迹规划算法在计算过程中生成无人机可行航迹，代价函数值 J 则是对航迹整体进行评价， J 值越小，生成的航迹越好。

5.3.3 突发威胁航迹重规划

当环境中出现突发威胁时，原始航迹已经不能够保证无人机飞行的安全，此时需要

进行航迹重规划，针对于战场环境多变的特性，只针对被突发威胁影响的区域进行航迹重规划，在此之前无人机仍按照原航迹飞行。首先需要找到处于威胁范围内的航迹段，计算每个航迹点与突发威胁中心的欧氏距离，通过式(5.10)判断航迹点是否在威胁范围内。

$$\sqrt{(x_i - x_{so})^2 + (y_i - y_{so})^2 + (z_i - z_{so})^2} \leq R_{so} + \Delta s \quad (5.10)$$

其中 (x_i, y_i, z_i) 航迹点坐标， Δs 是安全距离，当航迹点坐标满足式(5.10)时，说明该航迹点在突发威胁影响范围内，记录原始航迹处于威胁范围的航迹段，该航迹段的起点作为重规划起点，终点作为重规划的目标点。无人机航迹重规划示意图如图 5.2 所示。

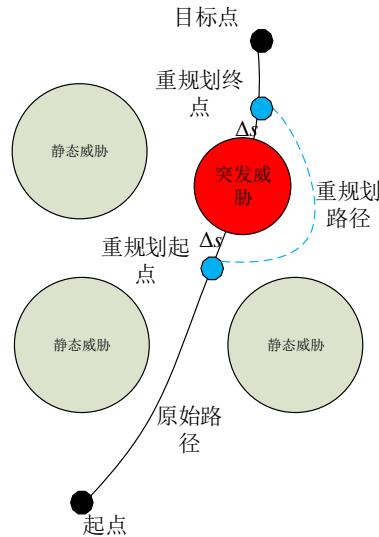


图 5.2 无人机航迹重规划示意图

在图 5.2 中，蓝色点为航迹重规划的起点和终点， Δs 是蓝色点到突发威胁中心的距离与威胁半径的差值，目的是为了保证无人机重规划起点和终点的安全性。当无人机检测到障碍物时，通过计算获得处于威胁范围内的航迹段及各航迹点坐标，选取重规划起点和终点，使用改进 RRT 算法完成航迹重规划。

突发威胁的无人机航迹重规划流程图如图 5.3 所示。

Step1: 改进 RRT 算法生成无人机飞行航迹。

Step2: 突发威胁参数初始化，建立突发威胁模型。

Step3: 判断航迹点是否进入突发威胁影响范围内，是则进入下一步，否则无人机继续按照原航迹飞行。

Step4: 计算获得处于威胁范围内的航迹段。

Step5: 将重规划起点终点等信息输入到改进 RRT 算法中。

Step6: 对重规划航迹进行平滑处理。

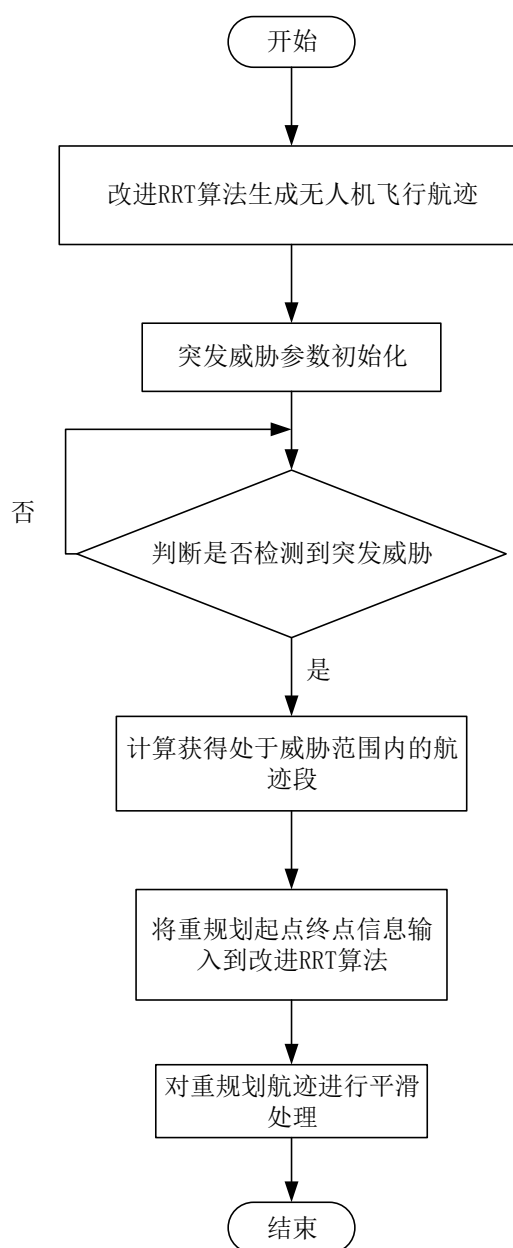


图 5.3 突发威胁下无人机航迹重规划流程图

5.3.4 动态环境下改进 RRT 算法三维航迹规划流程

动态环境下改进 RRT 算法三维航迹规划流程图如图 5.4 所示。

Step1: 初始化地图环境，无人机飞行起点终点位置信息。

Step2: 改进 RRT 算法规划初始航迹。

Step3: 判断无人机是否进入突发威胁影响范围内，是则进入下一步，否则无人机继

续按照原航迹飞行。

Step4: 获取重规划起点终点, 地图环境加入突发威胁信息。

Step5: 使用改进 RRT 算法进行航迹重规划。

Step6: 对重规划航迹进行平滑处理。

Step7: 规避突发威胁后无人机继续按照原航迹飞行。

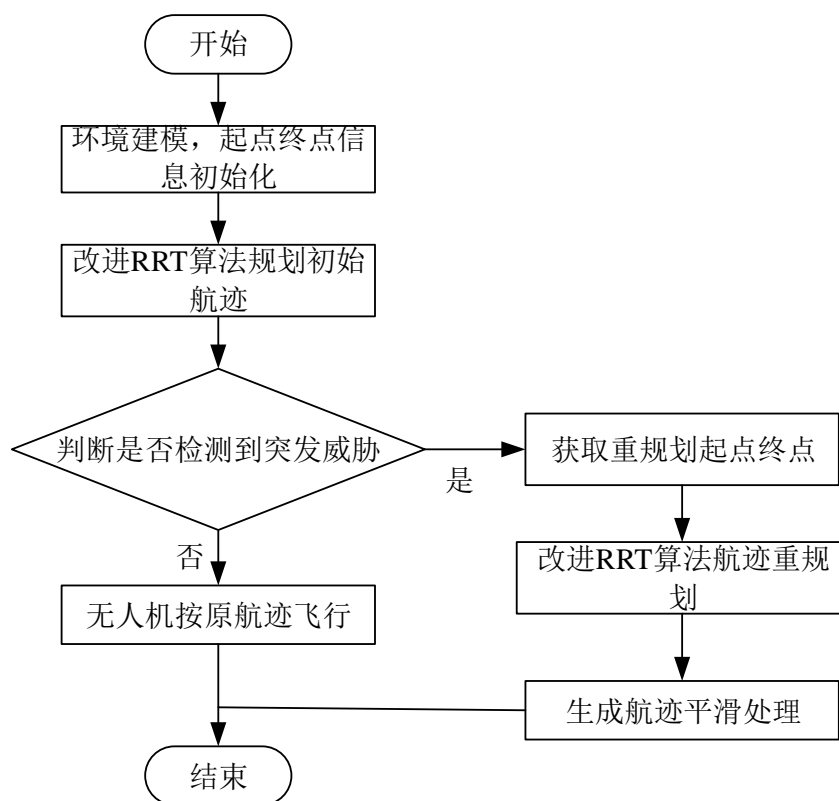


图 5.4 动态环境下改进 RRT 算法三维航迹规划流程图

5.4 航迹平滑优化策略

由于改进 RRT 算法的特性, 算法生成的航迹为一条折线, 同时, 由于飞行环境障碍物的影响, 导致算法所生成航迹的各个节点之间存在很多拐角, 这种不平滑的航迹对无人机飞行过程中的稳定性具有严重的影响。因此, 算法所生成的航迹需要进行平滑处理。

针对这一问题, 本文采用 B 样条平滑优化方法。作为一种基于插值思想的平滑处理方法, B 样条对于局部规划和全局规划都具有良好的优化效果。B 样条曲线具有局部性和连续性, 既具有贝塞尔曲线仿射不变性的优点, 还克服了贝塞尔曲线由于整体表示所带来的局限性。因此, 本文采用 B 样条曲线, 对算法所生成的航迹节点进行平滑处理。

B 样条曲线段可由公式(5.11)和公式(5.13)表示:

$$P(t) = \sum_{k=0}^K P_k F_{k,n}(t) \quad t \in [0,1] \quad (5.11)$$

其中， $P_k (k=0,1,\dots,K)$ 为航迹节点坐标， $F_{k,n}(t)$ 为 n 次 B 样条基函数， n 为最高次数， $t \in [0,1]$ 表示控制参数。基函数是由 1 个节点矢量的非递减参数 u 的序列 $U: u_0 \leq u_1 \leq \dots \leq u_{k+n+1}$ 所决定的 n 次分段多项式。B 样条的基函数采用 Cox-deBoor 递推公式：

$$\begin{cases} N_{k,0}(u) = \begin{cases} 1, & u_k \leq u \leq u_{k+1} \\ 0, & \text{其他} \end{cases} \\ N_{k,n}(u) = \frac{u - u_k}{u_{k+n} - u_k} N_{k,n-1}(u) + \frac{u_{k+n+1} - u}{u_{k+n+1} - u_{k+1}} N_{k+1,n-1}(u) \\ \text{定义 } \frac{0}{0} = 0 \end{cases} \quad (5.12)$$

由公式(5.11)可知，B 样条在对局部的控制顶点进行修改时，只会改变与该控制顶点相连的航迹段形状，这有利于无人机航迹规划算法的生成轨迹的优化。

本文设定 B 样条曲线最高次数 $n=3$ ，三次 B 样条的基函数表示如下：

$$\begin{cases} F_{0,3}(t) = \frac{1}{6}(-t^3 + 3t^2 - 3t + 1) \\ F_{1,3}(t) = \frac{1}{6}(3t^3 - 6t^2 + 4) \\ F_{2,3}(t) = \frac{1}{6}(-3t^3 + 3t^2 + 3t + 1) \\ F_{3,3}(t) = \frac{1}{6}t^3 \end{cases} \quad t \in [0,1] \quad (5.13)$$

5.5 仿真分析

为了验证算法的有效性，本文使用 MATLAB 软件进行算法仿真验证。无人机飞行起点设置为 $(0.5,0.5,0.5)km$ ，目标点设置为 $(8,8,2.5)km$ ，飞行环境为 $10km \times 10km \times 8km$ 的三维空间。本次实验设置两种仿真环境，环境 1 与环境 2 的障碍物参数设置如表 5.1 和表 5.2 表示。绘制的两种环境分别如图 5.5，5.6 所示。本文中，环境威胁均采用山峰表示，无人机不能在山峰内部穿过。

表 5.1 环境 1 障碍物信息

序号	位置/ 10^2m	$x_{si}/10^2m$	$y_{si}/10^2m$
1	(40,30,40)	10	10
2	(50,60,40)	10	10

表 5.2 环境 2 障碍物信息

序号	位置/ 10^2m	$x_{si}/10^2m$	$y_{si}/10^2m$
1	(30,20,40)	9	9
2	(30,50,20)	8	8
3	(40,80,50)	5	5
4	(50,40,40)	8	8
5	(70,70,35)	6	6
6	(80,50,20)	7	7

本文将改进算法与原始 RRT 算法和 tRRT 算法进行比较。 γ_{\max} 设置为 120° ， ξ_{\max} 设置为 70° ，分别在两种环境中进行仿真验证，仿真结果如图 5.7，5.8，5.10，5.11 所示，其中红色线条是原始 RRT 算法规划出的无人机航迹，黑色线条为 tRRT 算法规划出的航迹，绿色线条为本文算法规划的无人机飞行航迹。

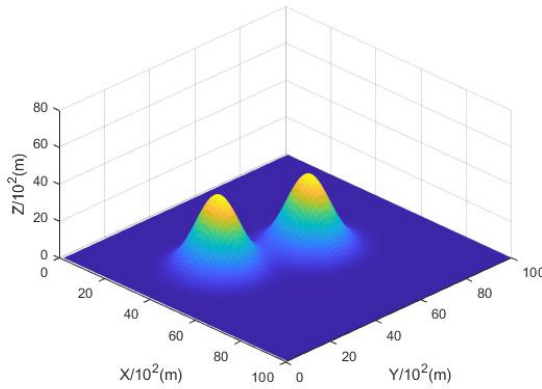


图 5.5 环境 1 仿真图

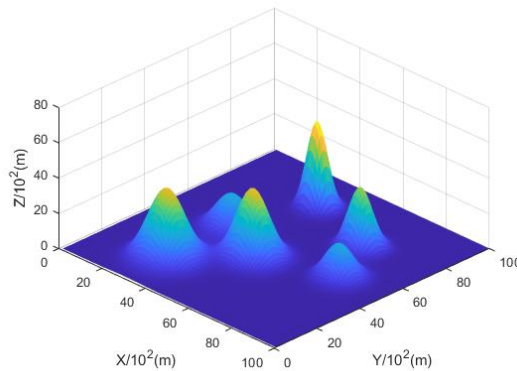


图 5.6 环境 2 仿真图

图 5.7，5.10 是 45° 视角下无人机飞行航迹示意图，图 5.8，5.11 是无人机飞行航迹

俯视图。由图可知，改进后的算法与原算法和 $tRRT$ 算法相比，航迹中拐点的数量显著减少。与原始 RRT 算法相比，势场作用的加入，使搜索树的生长方向指向目标点，加快了算法的搜索速度，与 $tRRT$ 算法相比，虽然 $tRRT$ 算法提高了航迹质量，但是仍存在较多拐点，不能满足无人机的飞行要求。而加入障碍物斥力作用后，本文算法搜索效率更高，无人机能更快到达目标点。

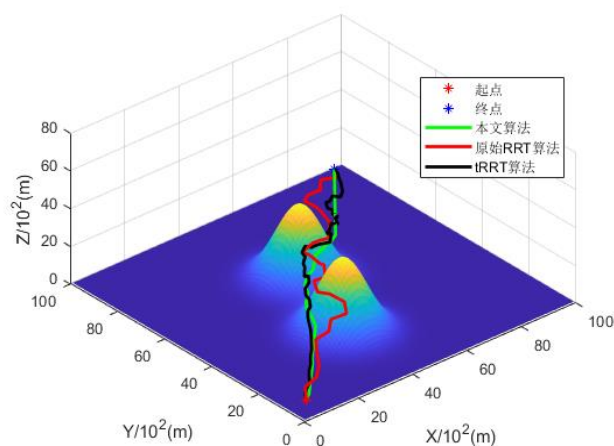


图 5.7 环境 1 仿真图

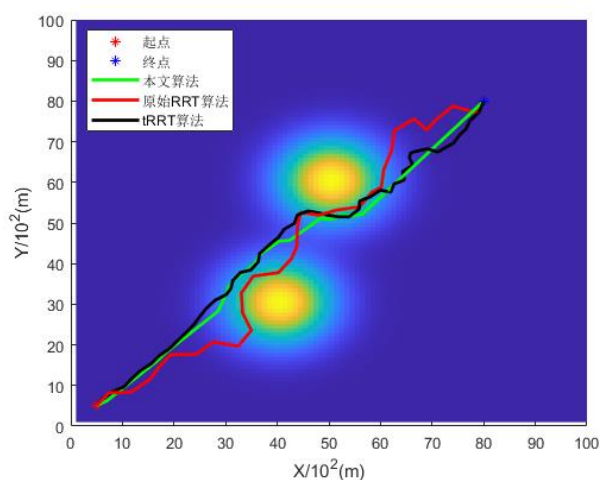


图 5.8 环境 1 俯视仿真图

表 5.3 环境 1 30 次实验平均值

算法	平均航迹长度/ 10^2m	平均运行时间/s	平均迭代次数/次
原始 RRT 算法	152.3711	0.4111	60
$tRRT$ 算法	125.0704	0.2524	51
本文算法	118.1163	0.1019	20

为了验证本文改进算法的有效性, 分别在环境 1 和环境 2 中对三种算法进行 30 次仿真实验。仿真结果如表 5.3、表 5.4 和图 5.9、图 5.12 所示。表 5.3 和图 5.9 为环境 1 下, 三种算法仿真实验数据。由表 5.3 和图 5.9 可知, 通过三种指标数据对比, **tRRT** 算法性能均优于原始 **RRT** 算法, **tRRT** 算法生成的航迹长度较原始 **RRT** 算法减少了 17.92%, 迭代次数减少了 15%, 运行时间减少了 38.60%。本文相较于 **tRRT** 算法, 在生成新节点时, 加入了山峰对无人机的斥力, 减少了山峰周围不必要的树节点。因此本文算法的航迹长度均值较 **tRRT** 算法减少了 5.56%, 迭代次数减少了 60.8%, 运行时间缩短了 59.6%。仿真结果表明, 在同一环境下, 本文算法在航迹长度, 运行时间以及迭代次数三方面均优于 **tRRT** 算法和原始 **RRT** 算法。

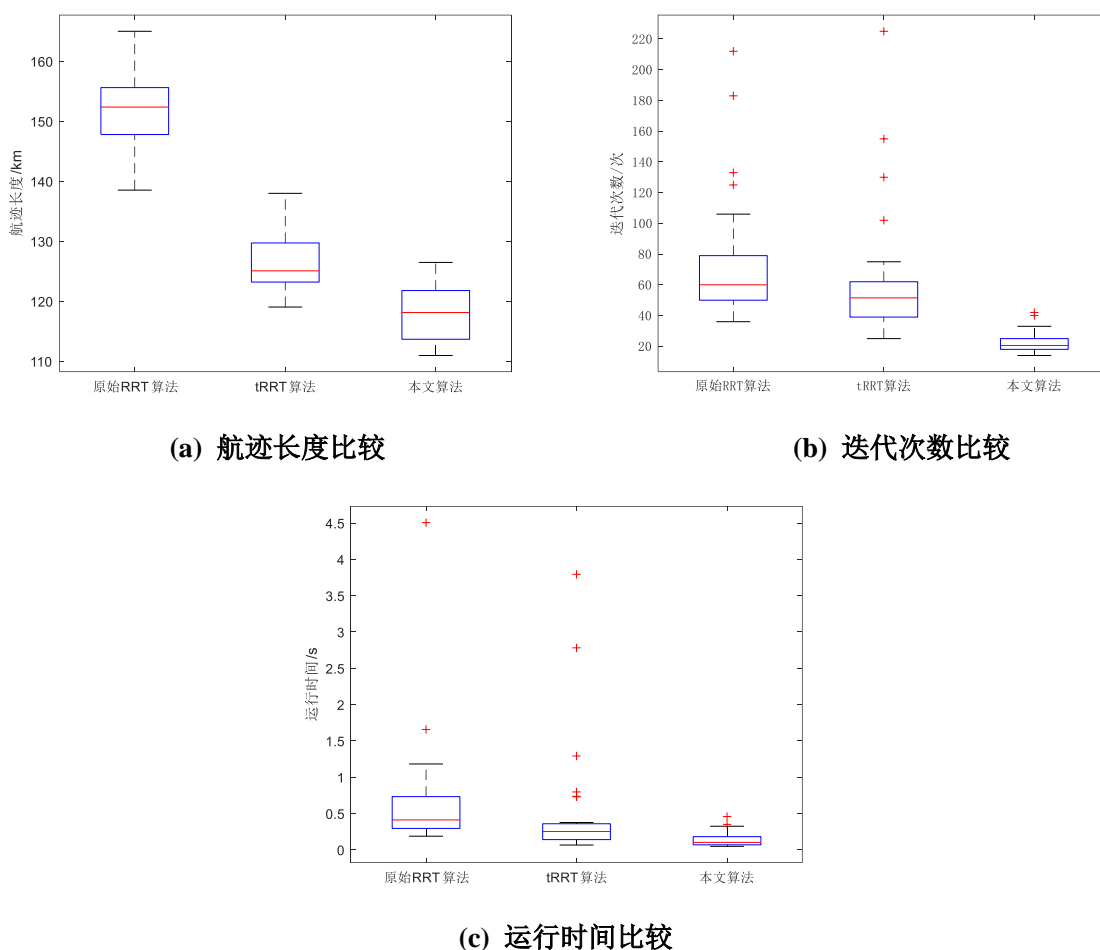


图 5.9 环境 1 仿真数据比较

图 5.12 为环境 2 下三种算法实验数据箱线图。表 5.4 为环境 2 下三种算法 30 次实验指标均值。由图 5.12 可知, 本文算法在航迹长度, 迭代次数和运行时间三种数据比较中, 均优于原始 **RRT** 算法和 **tRRT** 算法, 且数据中异常值数量均少于两种算法, 说明本算法具有较好稳定性。通过分析表 5.4 中数据可知, 本文算法能够缩短航迹长度, 算法

生成航迹长度较原始 RRT 算法减少了 32.38%，较 tRRT 算法减少了 5%，本文算法运行时间较原始 RRT 算法减少了 66.87%，较 tRRT 算法减少了 41.44%，本文算法迭代次数较原始 RRT 算法减少了 79.39%，较 tRRT 算法减少了 61.8%，提高算法搜索速度。两种环境下的算法仿真实验说明了本文算法具有普遍适应性，验证了算法在三维航迹规划中的有效性。

表 5.4 环境 2 30 次实验平均值

算法	平均航迹长度/ 10^2m	平均运行时间/s	平均迭代次数/次
原始 RRT 算法	178.6244	0.7652	165
tRRT 算法	127.1462	0.4329	89
本文算法	120.7849	0.2535	34

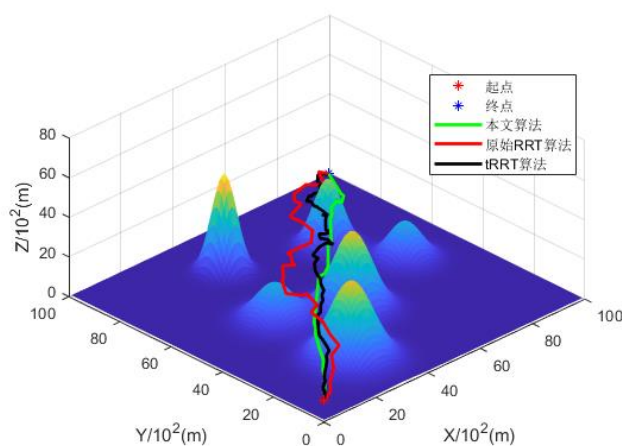


图 5.10 环境 2 仿真图

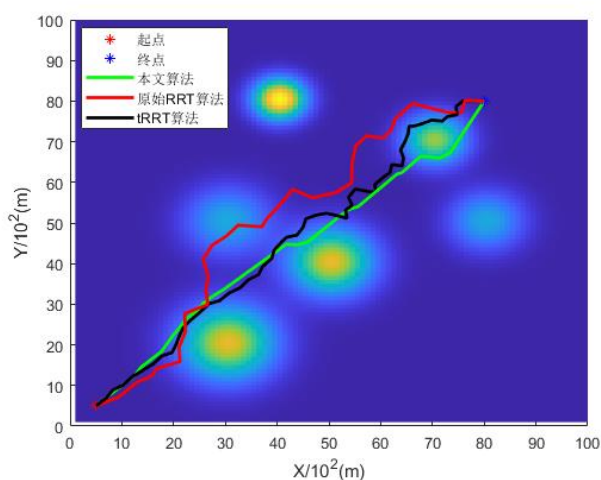


图 5.11 环境 2 俯视仿真图

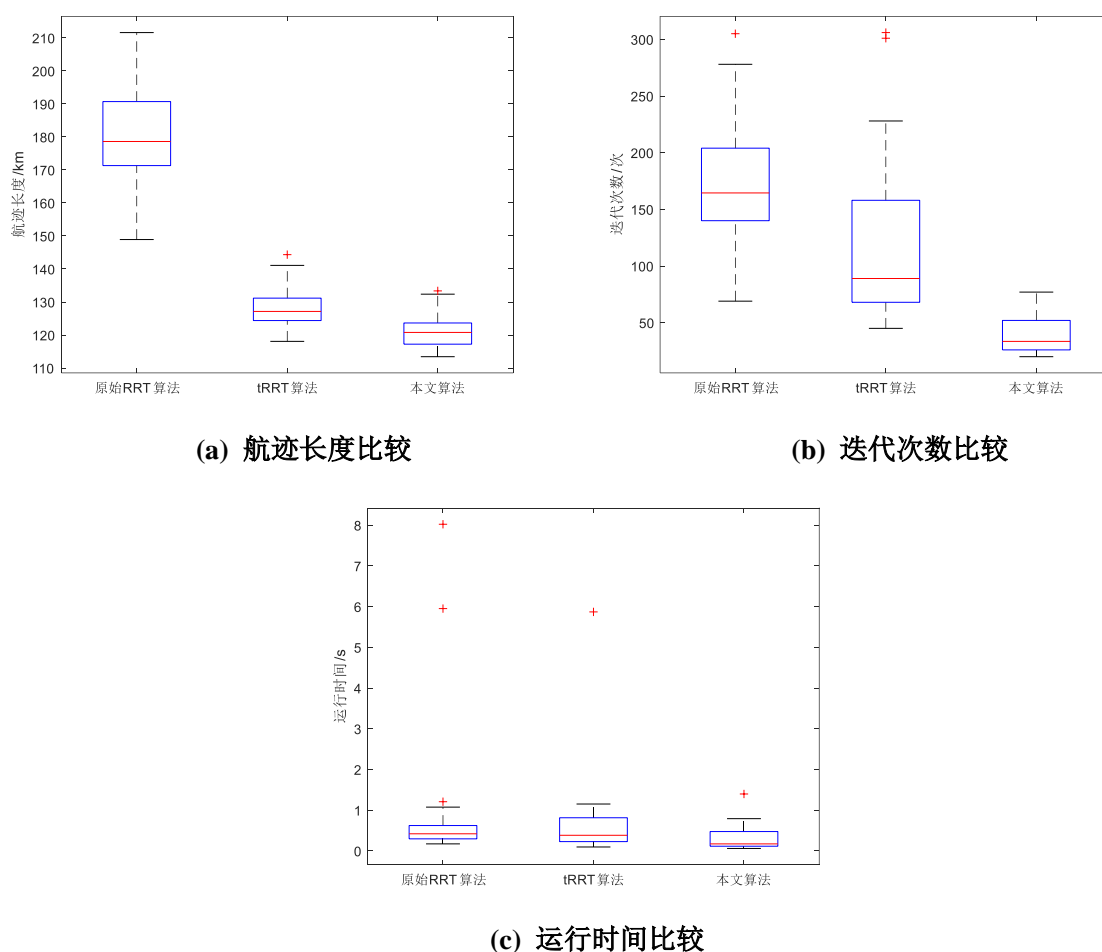


图 5.12 环境 2 仿真数据比较

当无人机按照规划好的航迹飞行时, 预定航迹上突然出现障碍物, 无人机需要进行局部航迹重规划。假定障碍物为球体威胁, 威胁半径设定为 $r = 400m$, 此时, 需要获得无人机重规划的起点和终点, 分别计算所有航迹点与突发威胁的球心的欧式距离, 获得处于威胁范围内的航迹段, 将该航迹段的起点定义为重规划起点, 终点作为重规划的目标点。分别在环境 1 和环境 2 中, 对改进算法进行验证, 仿真结果如图 5.13-5.16 所示。

图 5.13-5.16 中, 绿色菱形为原无人机起点和终点, 红色线条为原无人机航迹, 黑色球体为突发威胁, 白色五角星为重规划起点终点。白色线条为改进 RRT 算法重规划后的航迹, 绿色线条为三次 B 样条优化后飞行航迹。仿真结果表明两种环境下, 改进算法都可以重新规划一条绕过障碍物的新航迹, 完成局部路径重规划。环境 1 下, 本文算法完成重规划时间 0.10 秒, 环境 2 下完成重规划时间为 0.07 秒, 能够满足算法的实时性要求, 同时也验证了本文算法航迹重规划的可行性。

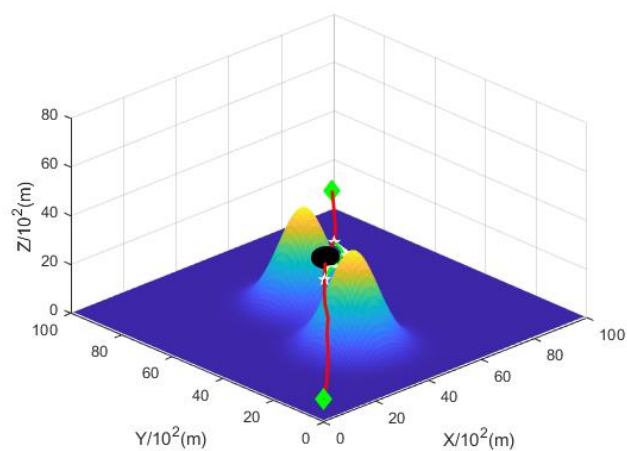


图 5.13 环境 1 突发威胁

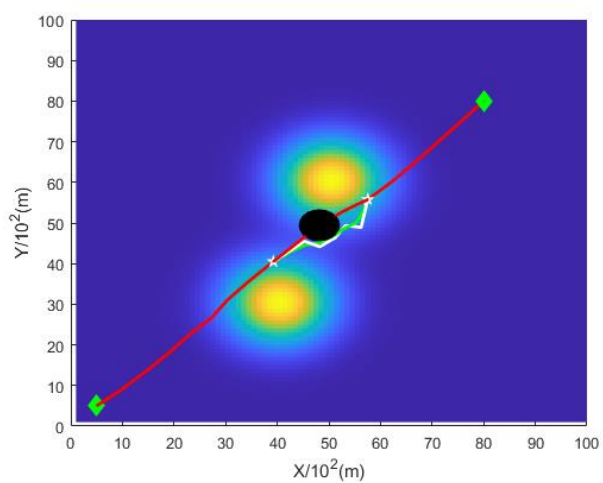


图 5.14 环境 1 突发威胁俯视

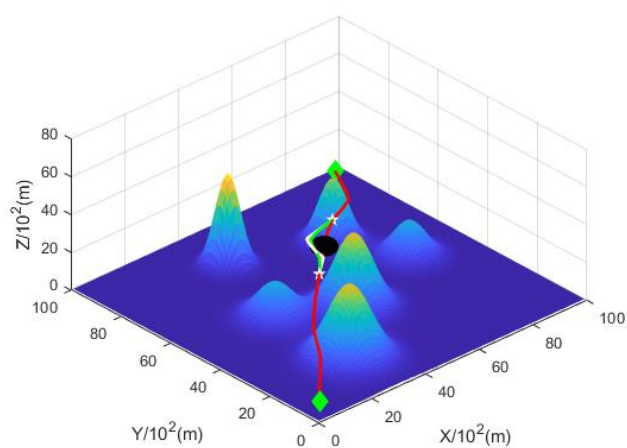


图 5.15 环境 2 突发威胁

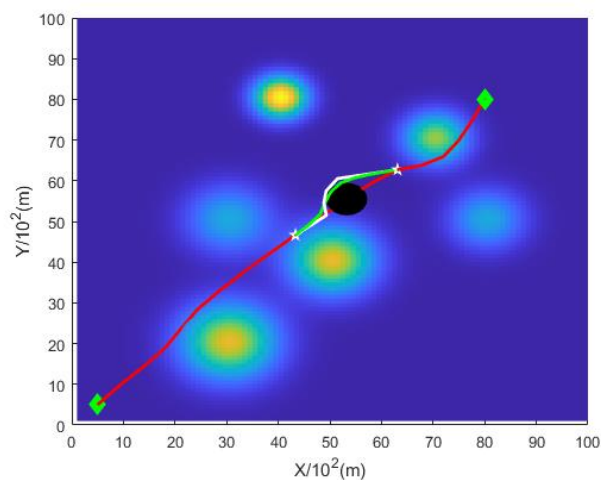


图 5.16 环境 2 突发威胁俯视

5.6 本章小结

本章主要研究了三维环境下改进 RRT 算法的无人机航迹规划。在第三章改进 RRT 算法二维全局航迹规划基础上。本章进一步将无人机二维航迹规划扩展到三维，建立了无人机在三维环境下的运动模型以及环境威胁模型，将航迹规划过程中的物理约束具体化，同时，考虑了无人机飞行过程中出现突发威胁的情况，建立了突发威胁模型，并给出了避障策略，最后将改进算法与 tRRT 算法和原始 RRT 算法进行比较，通过仿真验证了改进算法能够快速到达目标点，并能准确规避突发威胁。

结 论

随着科学技术的不断进步,无人机被广泛用于军事和民用领域,航迹规划是无人机任务规划的重要组成部分之一,是无人机自主航行必要的技术手段。本文以无人机完成飞行任务为背景,针对单无人机规避静态与动态障碍的问题分别进行了研究。

全文所做的主要工作及得到的结论总结如下:

(1) 静态威胁下无人机全局二维航迹规划算法。针对原始 RRT 算法规划速率较低等问题,本文对该算法进行改进,加入目标偏置来改进随机采样点的选取方式,将人工势场法融入到 RRT 算法中,改变了新节点的生成方式,障碍物的斥力使新节点远离威胁,目标点和随机采样点的引力使新节点向目标点方向生长,仿真结果表明改进算法减少了冗余点的数量,提高了算法的搜索效率。针对所生成航迹曲率不满足飞行要求的问题,使用遗传算法对生成的航迹进行优化,将改进后的 RRT 算法生成的航迹作为遗传算法的初始种群,建立了适应度函数模型,经过选择、交叉、变异对航迹点进行处理。仿真结果表明,经过遗传算法优化后,能够获得较优路径,解决了路径随机性问题。

(2) 移动威胁下的无人机航迹规划算法。考虑环境中出现的移动威胁,首先由改进 RRT 算法获得一条全局较优航迹。建立移动威胁和无人机的运动模型,当无人机飞行过程中出现移动威胁,根据无人机和移动障碍速度之间大小与方向的关系,设计了四种不同碰撞情形,并给出了相对应的避障策略。采用 VO 法进行移动障碍规避。仿真结果表明,该算法能够快速规避移动威胁,完成局部路径重规划。

(3) 基于改进 RRT 算法的无人机三维航迹规划。针对改进 RRT 算法三维航迹规划问题,建立了无人机在三维环境下的运动模型以及环境威胁模型,将航迹规划过程中的物理约束具体化,同时,针对无人机飞行过程中出现突发威胁的情况,建立了突发威胁模型,并给出了避障策略,最后将改进算法与 tRRT 和原始 RRT 算法进行比较,通过仿真验证了改进算法在三维环境中的有效性,能够快速到达目标点,并能快速规避突发威胁。

本文针对无人机航迹规划问题进行了一定的研究,取得了一些研究成果,但是,还是存在着需要改进的问题。主要包括以下几个方面:

(1) 本文在二维仿真中所设置威胁的形状为方便计算均为圆形,而现实情况下的威胁也有可能是其他形状的,如何解决这方面的问题还需要进一步深入研究。

(2) 针对移动威胁的航迹规划需要进一步研究, 本文只考虑了二维环境下移动威胁的规避方案, 且移动障碍物的速度大小和方向不发生改变, 之后的研究可以考虑研究用障碍物速度变化情况下的规避策略, 进一步建立三维移动障碍运动模型, 实现三维环境中的移动障碍规避。

(3) 本文针对单一无人机航迹规划。考虑到单无人机执行任务的效率有限。因此, 下一步将研究使用改进 **RRT** 算法完成多无人机协同航迹规划, 以及多无人机目标分配问题。

参考文献

- [1] Radoglou G P., Sarigiannidis P., Lagkas T., et al. A Compilation of UAV Applications for Precision Agriculture [J]. Computer Networks, 2020, 172:107-148.
- [2] Shim D H., Han J S., Yeo H T. A Development of Unmanned Helicopters for Industrial Applications [J]. Journal of Intelligent and Robotic Systems, 2009, 54:407-421.
- [3] Szabolcsi R. The Quadrotor-Based Night Watch bird UAV System Used in The Force Protection Tasks [J]. Nephron Clinical Practice, 2015, 21(3):749-755.
- [4] Waharte S., Trigoni N. Supporting Search and Rescue Operations with UAVs [C]. 2010 International Conference on Emerging Security Technologies, 2010:142-147.
- [5] Zhang C H., Kovacs J M. The application of small unmanned aerial systems for precision agriculture: A review [J]. Precision Agriculture, 2012, 13(6):693-712.
- [6] Boursianis A D., Papadopoulou M S., Diamantoulakis P., et al. Internet of Things (IoT) and Agricultural Unmanned Aerial Vehicles (UAVs) in Smart Farming: A Comprehensive Review [J]. Internet of Things, 2020:100-187.
- [7] Sarkar B., Guchhait R., Sarkar M., et al. How does an industry manage the optimum cash flow within a smart production system with the carbon footprint and carbon emission under logistics framework? [J]. International Journal of Production Economics, 2019, 213:243-257.
- [8] Chowdhury S., Emelogu A., Marufuzzaman M., et al. Drones for disaster response and relief operations: A continuous approximation model [J]. International Journal of Production Economics, 2017, 188:167-184.
- [9] Ak A., Tb B. Military autonomous drones (UAVs)-from fantasy to reality [J]. Legal and Ethical implications, 2021, 59:292-299.
- [10] Liu C., Sun S S., Tao C G., et al. Sliding mode control of multi-agent system with application to UAV air combat [J]. Computers and Electrical Engineering, 2021, 96:107491.
- [11] Liu J Z., Zhang B C., M Z L., et al. Geometric Reinforcement Learning for Path Planning of UAVs [J]. Journal of Intelligent & Robotic Systems: Theory & Application, 2015, 77(2):391-409.
- [12] 张欣,严俊,吴雪松.电子战无人机航电系统及其发展[J].航空电子技术,2012,43(2):8-13.
- [13] 王玉林,魏国亮,鲍海锋.基于 Dijkstra 算法的磁带导引 AGV 路径规划[J].农业装备与车辆工程,2018,56(3):51-54.
- [14] Meng B B. UAV Path Planning Based on Bidirectional Sparse A* Search Algorithm [C]. 2010 International Conference on Intelligent Computation Technology and Automation, 2010:1106-1109.
- [15] Chen Y B., Luo G C., Mei Y S., et al. UAV path planning using artificial potential field method updated by optimal control theory [J]. International Journal of Systems Science, 2016, 47(6):1407-1420.
- [16] 李璠,郝应光.基于改进混沌遗传算法的无人机航迹规划[J].电光与控制,2012,19(8):15-19.
- [17] Kennedy J., Eberhart R. Particle swarm optimization [C]. Proceedings of ICNN95-International Conference on Neural Networks, 1995:1942-1948.
- [18] 段海滨,王道波,朱家强,等.蚁群算法理论及应用研究的进展[J].控制与决策,2004(12):1321-1326+1340.
- [19] 刘永兰,李为民,吴虎胜,等.基于狼群算法的无人机航迹规划[J].系统仿真学报,2015,27(08):1838-1843.
- [20] 吴虎胜,张凤鸣,吴庐山.一种新的群体智能算法-狼群算法[J].系统工程与电子技术,2013,35(11):2430-2438.
- [21] Dong S Y., Zhu X P., Long G Q. Cooperative planning method for swarm UAVs based on hierarchical strategy [C]. 2012 3rd International Conference on System Science, Engineering Design and Manufacturing Informatization, 2012:304-307.
- [22] Maini P., Sujit P B. Path planning for a UAV with kinematic constraints in the presence of polygonal obstacles [C]. 2016 International Conference on Unmanned Aircraft Systems (ICUAS), 2016:62-67.

- [23] Szczerba R J., Galkowski P., Glicktein I S., et al. Robust algorithm for real-time route planning [J]. IEEE transactions on aerospace and electronic systems, 2000, 36(3):869-878.
- [24] Gururji A K., Agarwal H., Parsediya D K. Time-efficient A* Algorithm for Robot Path Planning [J]. Procedia Technology, 2016, 23:144-149.
- [25] 陈晓宏,储飞黄,方胜良,等.基于剖分网格改进 A*算法的航迹规划研究[J/OL].电光与控制:1-6.
- [26] 杨明亮,李宁.改进 A*算法的移动机器人路径规划[J/OL].机械科学与技术:1-7.
- [27] 张阳伟,乔越,李成凤.基于四叉树栅格环境的变步长双向 A* 算法[J].控制工程,2021,28(10):1960-1966.
- [28] Zhu L., Cheng X., Yuan F G. A 3D collision avoidance strategy for UAV with physical constraints [J]. Measurement, 2016, 77:40-49.
- [29] Chen X., Zhang J. The Three-Dimension Path Planning of UAV Based on Improved Artificial Potential Field in Dynamic Environment [C]. 2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics, 2013:144-147.
- [30] 李廷珍,招启军,张夏阳,等.基于改进人工势场法的无人直升机三维航迹规划[J].飞行力学,2022,40(1):69-75.
- [31] 韩尧,李少华.基于改进人工势场法的无人机航迹规划[J].系统工程与电子技术,2021,43(11):3305-3311.
- [32] Pehlivanoglu Y V., Pehlivanoglu P. An enhanced genetic algorithm for path planning of autonomous UAV in target coverage problems [J]. Applied Soft Computing, 2021, 112:107796.
- [33] Pehlivanoglu Y V. A new vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV [J]. Aerospace Science and Technology, 2012, 16(1):47-55.
- [34] 程泽新,李东生,高杨.一种改进遗传算法的无人机航迹规划[J].计算机仿真,2019,36(12):31-35.
- [35] Li J H., Huang Y B., Xu Z., et al. Path planning of UAV based on hierarchical genetic algorithm with optimized search region [C]. 2017 13th IEEE International Conference on Control Automation (ICCA), 2017:1033-1038.
- [36] 黄书召,田军委,乔路,等.基于改进遗传算法的无人机路径规划[J].计算机应用,2021,41(02):390-397.
- [37] Chen M., Wu Q X., Jiang C S. A modified ant optimization algorithm for path planning of UCAV [J]. Applied Soft Computing, 2008, 8(4):1712-1718.
- [38] 李喜刚,蔡远利.基于改进蚁群算法的无人机路径规划[J].飞行力学,2017,35(01):52-56.
- [39] Han Z L., Chen M., Shao S Y., et al. Improved artificial bee colony algorithm-based path planning of unmanned autonomous helicopter using multi-strategy evolutionary learning [J]. Aerospace Science and Technology, 2022, 122:107374.
- [40] 方群,徐青.基于改进粒子群算法的无人机三维航迹规划[J].西北工业大学学报,2017,35(01):66-73.
- [41] Liu Y., Zhang X., Guan X., et al. Adaptive sensitivity decision based path planning algorithm for unmanned aerial vehicle with improved particle swarm optimization [J]. Aerospace Science & Technology, 2016, 58:92-102.
- [42] 鲁亮亮,代冀阳,应进,等.基于 APSODE-MS 算法的无人机航迹规划[J/OL].控制与决策:1-10.
- [43] Girija S., Joshi A. Fast Hybrid PSO-APF Algorithm for Path Planning in Obstacle Rich Environment [J]. IFAC-Papers OnLine, 2019, 52(29):25-30.
- [44] Qu C Z., Gai W D., Zhang J., et al. A novel hybrid grey wolf optimizer algorithm for unmanned aerial vehicle (UAV) path planning [J]. Knowledge-Based Systems, 2020, 194:105530.
- [45] 李彦苍,徐培东.基于自适应步长和莱维飞行策略的改进狼群算法[J/OL].重庆大学学报:1-18.
- [46] 滕志军,吕金玲,郭力文,等.一种基于 Tent 映射的混合灰狼优化的改进算法[J].哈尔滨工业大学学报,2018,50(11):40-49.
- [47] Lavalley S M. Rapidly-Exploring Random Trees: A New Tool for Path Planning [J]. Algorithmic & Computational Robotics New Directions, 1999, 1(1):293-308.
- [48] 成浩浩,杨森,齐晓慧.基于改进 RRT 算法的四旋翼无人机航迹规划[J].计算机工程与设计,2018,39(12):3705-3711.
- [49] Faris J., Ron R., Philipp N. Smooth Path-Generation Around Obstacles Using Quartic Splines and RRTs [J]. IFAC-PapersOnLine, 2017, 50(1):9108-9113.

- [50] 尹高扬,周绍磊,吴青坡.基于改进 RRT 算法的无人机航迹规划[J].电子学报,2017,45(7):1764-1769.
- [51] 刘成菊,韩俊强,安康.基于改进 RRT 算法的 RoboCup 机器人动态路径规划[J].机器人,2017,39(1):8-15.
- [52] 樊娇,雷涛,韩伟,等.无人机航迹规划技术研究综述[J].郑州大学学报,2021,42(03):39-46.
- [53] 王琼,刘美万,任伟建,等.无人机航迹规划常用算法综述[J].吉林大学学报,2019,37(1):58-67.
- [54] 吴菲菲,李睿毓,黄鲁成,等.跨产业技术溢出识别与效应测度研究——以无人机技术为例[J].科学与科学技术管理,2018,39(6):84-98.
- [55] 钱洲元,雷明.面向无人机航迹规划的自适应乌贼算法[J].哈尔滨工业大学学报,2019,51(10):37-46.
- [56] 于泳浩.自适应可变威胁的多无人飞行器协同航迹规划算法[D].沈阳:辽宁大学,2012.
- [57] Philpott B P. Strategic and tactical planning in international marketing policies [J]. Lincoln College Agricultural Economics Research Unit, 1965, 25:33-51.
- [58] 简康.无人机航迹规划算法研究[D].西安:西安电子科技大学,2014.
- [59] 庞强伟,胡永江,李文广,等.多无人机协同侦察任务规划方法研究综述[J].电讯技术,2019,59(6):741-748.
- [60] 邸斌,周锐,董卓宁.考虑信息成功传递概率的多无人机协同目标最优观测与跟踪[J].控制与决策,2016,31(4):616-622.
- [61] Bhattacharya P., Gavrilova M L. Roadmap-Based Path Planning-Using the Voronoi Diagram for a Clearance-Based Shortest Path [J]. Robotics & Automation Magazine IEEE, 2008, 15(2):58-66.
- [62] 刘振,史建国,高晓光.Voronoi 图在航迹规划中的应用[J].航空学报,2008(S1):15-19.
- [63] Chen X., Chen X M. The UAV dynamic path planning algorithm research based on Voronoi diagram [C]. The 26th Chinese Control and Decision Conference (2014 CCDC), 2014:1069-1071.
- [64] 袁梦顺,陈谋,邵书义,等.基于改进精英蚁群算法的无人机三维航迹规划[J].火力与指挥控制,2022,47(2):37-42.
- [65] 赵红超,周洪庆,王书湖.无人机三维航迹规划的量子粒子群优化算法[J].航天控制,2021,39(1):40-45.

致 谢

光阴似箭日月如梭，研究生生涯即将结束，回顾之前的三年学习生涯，我收获颇多，不仅得到了良师也收获了益友。

首先，我要感谢我的导师陈侠教授，陈老师不仅在学术方面对我进行指导与帮助，还在生活上给予我关心与照顾。从研究生一年级开始，确定研究内容，到研究生三年级毕业论文的完成，都离不开陈老师的细心指导。每当我在生活中遇到了难题，陈老师总能为我指引正确的道路，帮助我成长。她的工作方式、解决问题的方法以及面对生活的态度，都是我今后学习和生活的明灯。

其次，我要感谢我所在的科研团队以及生活中的朋友。无论是研讨会上的讨论还是私下的交流，都是我美好的回忆。感谢师兄周琦轩、侯林浩，师姐祁丽杰，是你们热心的帮助让我的研究取得了一定的成果。感谢同届的毛海亮、李露在我失落的时候为我加油打气。也感谢师弟范珈铭、栗杰、赵宇飞的支持与信任。

最后，我还要感谢我的家人和对我提供过帮助的老师 and 同学们，谢谢你们的照顾和陪伴，正因为有你们我才能成为更好的自己，谢谢你们！

刘奎武于沈阳航空航天大学

2022 年 05 月 15 日

攻读硕士期间发表(含录用)的学术论文

- [1] 陈侠, **刘奎武**, 毛海亮. 基于 APF-RRT 算法的无人机航迹规划[J]. 电光与控制.(已录用)(本文硕士学位论文第三章)
- [2] **Liu Kuiwu**, Chen Xia, Li Lu A two-level obstacle avoidance algorithm for UAV path planning [J]. PROCEEDINGS OF THE IMECHE, PART G: JOURNAL OF AEROSPACE ENGINEERING. (一审)(本文硕士学位论文第四章)

版权授权说明

本人授权学校“有权保留送交学位论文的原件，允许学位论文被查阅和借阅，学校可以公布学位论文的全部或部分内容，可以影印、缩印或其他复制手段保存学位论文”；愿意将本人学位论文电子版提交给研究生部指定授权单位收录和使用。学校必须严格按照授权对论文进行处理，不得超越授权对毕业论文进行任意处置。

作者：刘奎武

2022年6月4日

指导教师：陈侠

2022年6月4日