

中图分类号: TP391

论文编号: 1028703 18-SX136

学科分类号: 081101

# 硕士学位论文

## 多机协同任务规划与非完备信息下攻击决策技术

研究生姓名	夏鹏程
学科、专业	控制理论与控制工程
研究方向	无人机协同作战
指导教师	陈谋 教授

南京航空航天大学

研究生院 自动化学院

二〇一八年三月

Nanjing University of Aeronautics and Astronautics  
The Graduate School  
College of Automation Engineering

**Research on Mission Planning & Attack  
Decision-Making under Incomplete  
Information for Multi-UAVs**

A Thesis in

Control Theory and Control Engineering  
by

Xia Pengcheng

Advised by

Professor Chen Mou

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

Master of Engineering  
March, 2018



## 摘 要

近年来,无人机在军事领域的作用越来越显著,在战争中凸显出不可替代的地位,受到了世界各军事大国的广泛的关注。面对现代化战场中任务的多样性与复杂性,无人机的作战模式逐渐由单机作战转向着多机协同作战。本文在多无人机协同作战的背景下,对多机任务规划与攻击决策方面的问题进行了研究。

首先,对无人机协同任务规划进行了分析,描述了其中的任务区域,任务执行方与被执行方,并对规划所受到的任务约束、威胁约束以及航程的预估进行建模分析。针对多机协同攻击决策问题,建立了角度、距离、速度、高度与空战能力 5 个威胁因子对空战态势进行描述,并提出使用区间数表示威胁因子中的不确定信息。

其次,针对多无人机协同任务规划中的任务分配问题,结合规划场景中的多种约束,研究了基于静态链表结构的遗传算法,并使用在了任务分配中,仿真结果表明了设计的有效性。

接着,为了实现协同任务规划中快速有效的航路规划,研究了改进的 A\*算法。对 A\*算法的多种启发函数进行分析与选择,并设计了一种新型地图信息存储方式对 A\*算法进行改进,仿真证明改进提高了算法的规划速度。

随后,任务执行过程中由于事先侦察的不充分、任务临时调整、设备故障等因素会引发如下重分配情况:任务点新增,任务点取消,新增威胁区域,无人机战损以及人为任务调整。研究了基于合同网算法的任务重分配算法,并对合同网中的投标过程进行改进。仿真结果表明了设计的合理性。

然后,综合考虑战场态势信息存在的非完备性和目标意图,提出了一种无人机协同攻击决策方法。利用粗糙集理论对历史空战信息进行规则的挖掘,对比当前需要识别意图的态势信息与规则之间的相似程度,获取目标意图。再将意图结合当前的态势构建攻击分配的模型,并对匈牙利算法进行改进,使其可以在当前这种态势信息不确定的模型中完成目标分配。通过仿真实验证实了改进的有效性。

最后,将上述任务分配、航路规划以及重分配设计相融合,在 Visual Studio 2015 平台上利用 MFC 相关的技术开发设计了协同任务规划系统仿真平台,并演示了协同任务规划的仿真过程。

**关键词:** 无人机, 任务分配, 航路规划, 遗传算法, A\*算法, 合同网, 非完备信息, 意图识别, 粗糙集, 攻击决策, 匈牙利算法

## ABSTRACT

In recent years, the UAV(Unmanned Aerial Vehicle) has become more and more essential in the military field. It plays an irreplaceable role in the war and has attracted the widespread attention of all the major military countries in the world. The operational mode of the UAV has shifted from single-UAV mode to multi-UAVs cooperative mode when faced with the diversity and complexity of the mission in the modern battlefield. In this thesis, the problem of multi-UAVs mission planning and attack decision-making are studied under the background of multi-UAVs cooperative operation.

First of all, the cooperative mission planning of UAVs is analyzed. The mission area property, the tasks and the task executors are described in this part, the models of the task constraints, the map constraints and the voyage estimation are also constructed. According to the problem of multi-UAVs cooperative attack decision-making, five threat factors: angle, distance, velocity, altitude and air combat capability are established to describe the situation of air combat and the interval number is used to represent the uncertain threat factors.

Secondly, in order to solve the task assignment problem in cooperative mission planning, a kind of genetic algorithm based on the static linked list is proposed, and this method can solve the task assignment problem with multiple constraints in the mission area. The simulation results prove the effectiveness of the method.

Then, A\* algorithm is studied to achieve a path planning result fast and effectively in cooperative mission planning. A variety of heuristic functions of A\* algorithm are analyzed and a suitable one is selected. In this part, a new storage method of the mission area map is designed and it accelerates the speed of the algorithm in the simulation.

Next, because of inadequate reconnaissance in advance, temporary task adjustment and equipment failure, the task reassignment should be taken into account. The task reassignment will be triggered during the execution of the tasks when faced with: task point add, task point cancel, newly added threat areas, UAV failure and man-made task adjustment. The task reassignment method is designed using the contract net protocol, which the bidding process is improved in this part. The simulation results show that the design is reasonable.

And then, considering the incompleteness of the battlefield situation information and the intention of the target, a decision-making method of UCAV(Unmanned Combat Aerial Vehicle)

collaborative attack is proposed. In this part, the rough set theory is used to mine the historical air combat information for rules, and then the situation information that need to recognize and the rules are compared to get the intention. After obtaining the intention, the model of the fire distribution is constructed according to the current situation, and the Hungarian algorithm is improved so that it can be used in the current model with uncertain situation information. The simulation results show the effectiveness of the improvement.

Finally, the task assignment, path planning and task reassignment are combined into a software. On the Visual Studio 2015, using MFC, a cooperative mission planning system simulation platform is developed, and the simulation process of the cooperative mission planning can be demonstrated on it.

**Key Words:** UAV, Task assignment, Path planning, Genetic algorithm, A\* algorithm, Contract net, Incomplete information, Intent recognition, Rough set, Attack decision-making, Hungary algorithm.

## 目 录

摘 要 .....	I
ABSTRACT .....	II
目 录 .....	IV
图表清单 .....	VII
注释表 .....	IX
缩略词 .....	X
第一章 绪论 .....	1
1.1 课题研究背景及意义 .....	1
1.2 国内外研究现状 .....	2
1.2.1 无人机的发展历程与现状 .....	2
1.2.2 无人机协同任务规划国内外现状 .....	4
1.2.3 无人机意图识别与攻击决策技术国内外现状 .....	6
1.3 本文主要研究内容 .....	7
第二章 无人机协同作战系统 .....	9
2.1 引言 .....	9
2.2 系统组成 .....	9
2.3 无人机协同任务规划系统分析与建模 .....	10
2.3.1 无人机协同任务规划系统场景描述 .....	10
2.3.2 任务点约束模型 .....	12
2.3.3 地图约束建模 .....	13
2.3.4 航路预估 .....	14
2.4 无人机攻击决策系统分析与建模 .....	19
2.4.1 威胁因子定义 .....	19
2.4.2 无人机攻击决策系统中的信息不完备 .....	21
2.5 本章小结 .....	21
第三章 无人机任务分配技术 .....	22
3.1 引言 .....	22
3.2 任务分配问题分析 .....	22
3.2.1 任务分配前的数据准备 .....	22

3.2.2 任务分配的目的 .....	23
3.2.3 任务分配目标函数 .....	23
3.3 基于静态链表结构的遗传算法 .....	25
3.3.1 染色体编码 .....	26
3.3.2 种群初始化 .....	28
3.3.3 交叉操作 .....	28
3.3.4 变异操作 .....	29
3.3.5 调整染色体使其满足合理性的过程 .....	30
3.4 仿真算例 .....	31
3.5 本章小结 .....	34
第四章 无人机航路规划技术 .....	35
4.1 引言 .....	35
4.2 航路规划研究对象 .....	35
4.3 A*算法 .....	36
4.3.1 算法原理 .....	36
4.3.2 启发函数 .....	37
4.4 A*算法的改进 .....	38
4.5 仿真算例 .....	39
4.5.1 不同启发函数的影响 .....	39
4.5.2 ClosedList 改进结果 .....	42
4.5.3 改进 A*算法用于多编队航路规划 .....	42
4.6 本章小结 .....	43
第五章 无人机任务重分配技术 .....	44
5.1 引言 .....	44
5.2 基于合同网协议的任务重分配 .....	44
5.2.1 合同网协议 .....	44
5.2.2 基于改进合同网算法的编队任务重分配 .....	45
5.3 仿真算例 .....	47
5.4 本章小结 .....	49
第六章 非完备信息下的空中目标意图识别与攻击决策 .....	51
6.1 引言 .....	51
6.2 非完备信息下的空中目标意图识别 .....	51



6.2.1 区间数与联系数 .....	51
6.2.2 粗糙集相关理论 .....	53
6.2.3 非完备区间数决策信息系统中的相似关系定义 .....	53
6.2.4 不完备信息决策表中的规则获取 .....	54
6.2.5 意图识别的流程 .....	56
6.2.6 仿真算例 .....	57
6.3 存在信息不完备条件下的多目标分配决策 .....	59
6.3.1 问题描述 .....	59
6.3.2 目标平衡分配问题的数学模型 .....	60
6.3.3 匈牙利算法的改进 .....	60
6.3.4 不完备信息下的目标平衡分配流程 .....	63
6.3.5 仿真算例 .....	64
6.3.6 改进匈牙利算法应用于非平衡分配 .....	65
6.4 本章小结 .....	66
第七章 无人机协同任务规划系统仿真设计 .....	67
7.1 引言 .....	67
7.2 协同任务规划系统软件架构 .....	67
7.2.1 软件整体结构 .....	67
7.2.2 各模块介绍 .....	69
7.3 协同任务规划系统仿真软件设计 .....	71
7.4 本章小结 .....	75
第八章 总结与展望 .....	76
8.1 本文工作总结 .....	76
8.2 不足与展望 .....	77
参考文献 .....	78
致 谢 .....	84
在学期间的研究成果及发表的学术论文 .....	85

## 图表清单

图 1.1 无人机示意图 .....	4
图 2.1 协同作战系统结构图 .....	9
图 2.2 单任务点时间约束示意图 .....	13
图 2.3 任务点间的时间约束示意图 .....	13
图 2.4 任务点执行时间示意图 .....	14
图 2.5 可达性矩阵内元素求解流程图 .....	16
图 2.6 航路被干扰示意图 .....	17
图 2.7 转换坐标系后航路被干扰示意图 .....	17
图 2.8 绕过威胁区域示意图 .....	18
图 2.9 空战态势图 .....	19
图 3.1 总任务信息分解过程示意图 .....	22
图 3.2 任务分配目的示意图 .....	23
图 3.3 染色体交叉示意图 .....	29
图 3.4 任务点与威胁区域平面投影示意图 .....	33
图 4.1 航路规划细分过程示意图 .....	36
图 4.2 多种启发函数效果示意图 .....	38
图 4.3 不同启发函数下 A*算法效果图 .....	41
图 4.4 多编队航路规划效果 .....	42
图 5.1 基于合同网的算法流程图 .....	45
图 5.2 任务重分配模型图 .....	47
图 5.3 任务场景示意图 .....	48
图 5.4 多种情况下重分配仿真示意图 .....	49
图 6.1 非完备决策信息表获取规则流程图 .....	55
图 6.2 目标意图识别流程图 .....	57
图 6.3 不完备信息下的目标平衡分配算法流程 .....	63
图 7.1 整体软件结构图 .....	68
图 7.2 各模块与系统存储区关系图 .....	68
图 7.3 信息输入内容 .....	69
图 7.4 任务分配与编队分配模块工作流程 .....	70

图 7.5 仿真软件主界面 .....	71
图 7.6 信息输入界面 .....	72
图 7.7 任务分配结果方案选择界面 .....	72
图 7.8 航路规划与任务过程模拟显示界面 .....	73
图 7.9 新增任务点重分配输入界面 .....	74
图 7.10 任务重分配后航路重规划结果 .....	74
图 7.11 任务重分配信息输入界面 .....	75
图 7.12 任务分配结果评估界面 .....	75
表 2.1 任务点 $TP_i$ 内目标信息 .....	11
表 2.2 各类型目标所需执行任务与无人机需求表 .....	12
表 3.1 修改前的静态链表 .....	26
表 3.2 修改后的静态链表 .....	27
表 3.3 染色体结构表 .....	27
表 3.4 无法解码的染色体结构 .....	30
表 3.5 调整后的染色体结构 1 .....	31
表 3.6 调整后的染色体结构 2 .....	31
表 3.7 我方无人机相关信息表 .....	31
表 3.8 任务点坐标与时间约束表 .....	31
表 3.9 任务点内目标信息表 .....	32
表 3.10 地形与气象威胁信息 .....	32
表 3.11 任务分配结果表 .....	33
表 4.1 $h_o(n)$ 与 $h_md(n)$ 在不同威胁区域数下对比表 .....	41
表 4.2 改进前后运行时间对比表 .....	42
表 6.1 非完备区间数决策信息表 .....	55
表 6.2 非完备决策信息表获取规则表 .....	56
表 6.3 待识别的威胁因子信息 .....	57
表 6.4 转化为联系数的威胁因子信息 .....	57
表 6.5 威胁因子信息与规则的贴近度 .....	58
表 6.6 各威胁因子权重分配 .....	58
表 6.7 威胁因子信息与所有规则的综合贴近度 .....	58
表 6.8 意图威胁值表 .....	59

## 注释表

符号	定义	单位
$v$	无人机飞行速度	$Km/h$
$T$	所有任务点的集合	无
$S_{Time}$	任务点任务开始时间	$min$
$E_{Time}$	任务点任务结束时间	$min$
$R_{RA}$	目标雷达探测范围	$Km$
$R_{ATK}$	目标武器打击范围	$Km$
$NUM$	任务所需总无人机数量	架
$discost$	任务总的航程	$Km$

## 缩略词

缩略词	英文全称
UAV	Unmanned Aerial Vehicle
UCAV	Unmanned Combat Aerial Vehicle
SEAD	Suppression of Enemy Air Defenses
SPFA	Shortest Path Faster Algorithm
SMAA	Stochastic Multi-criteria Acceptability Analysis
CCD	Charge Coupled Device
GIS	Geographic Information System
D-S	Dempster-Shafer
MFC	Microsoft Foundation Classes

## 第一章 绪论

### 1.1 课题研究背景及意义

随着空战环境的日趋复杂和任务类型多样化, 高新技术越来越多的应用在军事领域中, 21 世纪的战争是信息权争夺的战争, 而无人机(Unmanned Aerial Vehicle, UAV)将是这场战争中不可缺少的角色<sup>[1][2][3]</sup>。飞行员作为比飞机本身更宝贵的资源, 其培养需要花费巨大的代价, 无人机在战争中的使用则可以大大减少战争中的飞行员牺牲<sup>[4]</sup>。

作为现代空战中主力的有人机, 其出动任务通常都是以编队为单位进行, 现代较大规模的战争中, 一个编队内由预警机、歼击机、电子战飞机、空中加油机等多机种组成<sup>[5]</sup>, 相互协同, 功能互补。无人机的诞生初衷就是为了替代有人机, 减少飞行员的伤亡, 在未来逐渐替代有人机的过程中, 无人机的作战模式将沿用有人机之间的协同编队作战模式, 协同编队作战技术在无人机领域的进一步深入, 对保证现代化战争的胜利有着十分重要的意义。

战争是一场相互消耗资源的竞争, 无人机协同编队作战, 能够提高整体的生存能力, 不论是预先知晓的任务还是突发威胁情况都能够有能力应对。编队内侦察型的无人机为编队成员提供重要的敌方位置与火力等信息, 使编队可以规避掉一些高威胁区域, 电子战型无人机可使敌方雷达失效, 提高任务执行的安全性, 攻击型无人机也可在遇到敌袭时利用自身多种武器保护编队内其他成员的安全。同时, 协同编队作战相对于单机作战有着更大的优势, 面对愈加复杂和广阔的战场环境, 协同侦察可提供更加全面的战场信息, 广阔的侦察范围和发现更加隐蔽的敌方作战力量<sup>[6]</sup>, 协同干扰可以覆盖整个雷达的辐射面, 协同攻击可以实现对目标的多角度打击、对多目标同时打击, 这些都是单机作战所不具备的优势。因此, 无论是从编队的生存能力还是从带来的战场优势的角度来说, 无人机协同编队作战都将是未来无人机作战一个必然的发展趋势<sup>[7]</sup>。

当前世界各国普遍在新一代军用飞机上广泛采用了隐身、伪装、电子干扰等技术, 这样不仅能够提升战机生存能力, 还能给对方收集战机的战场作战相关信息产生一定的难度。在现代化空战环境下, 依靠雷达探测等侦查手段已经很难完全获知敌方目标的全部信息。如何处理并利用我方获取的可能存在不完备不确定特点的信息, 对敌方目标的作战意图进行有效准确的识别和推断是无人作战飞机完成有效打击的重要环节<sup>[9]</sup>, 这将提升对高威胁、高价值目标打击的快速性和准确性, 对取得空中优势, 降低战争成本具有重大意义。数据不完备是许多研究领域中的一个复杂的问题, 也是空战数据处理经常遇到的一个技术难点。空战数据库中一些个体记录在某些属性上可能存在不确定性, 使得一些重要的模式很难被发掘<sup>[10]</sup>, 带来的具体影响如下:

首先,系统中失去了很多有用的信息;其次,系统表现出更加显著的不确定性,其中的确定性成分更难以掌握;再次,这些不确定数据会影响空战数据库的开发利用,导致不可靠的结果输出<sup>[11]</sup>。因此,如何对含有不完备信息的空战信息系统进行处理与合理利用,使其可以指导我方战机作战,是一个非常具有实战价值和研究意义的问题。

随着当前人工智能技术发展以及其广泛的应用前景,可以想见真正能够满足未来战场需求的无人机必须为可以进行独立决策和任务规划的智能武器。未来复杂作战条件下在作战过程中要求无人机能自动进行目标搜索、监视、跟踪和武器分配、发射等作战行动决策,具有相同于有人机的作战应变能力<sup>[12]</sup>,这就需要最大程度地给无人机赋予智能,实现其自主攻击决策,从而在现代信息化战争中抢占战场的先机并对战争的胜负结果产生决定性的作用。现有的决策技术很少考虑到目标意图对火控决策所造成的影响,也很少考虑非完备信息下的决策问题。为了使无人作战飞机(Unmanned Combat Aerial Vehicle, UCAV)能有效的打击敌方机动目标,有必要依据当前和历史态势信息完成对敌方目标攻防意图的感知,同时,有效的应对非完备信息对攻击决策造成的影响,将为我方无人作战飞机的主动出击奠定了坚实基础,有利于提高我方无人作战飞机的空战能力。

综合以上内容,无人机协同作战、非完备信息处理、意图识别、自主决策等主题的研究,将推动无人机在未来取得更广泛应用,也是本文研究的主要内容。

## 1.2 国内外研究现状

### 1.2.1 无人机的发展历程与现状

无人机自诞生至今 80 多年以来,由于结构简单、安全性强、性能高、操作简单、无需承担生命危险、成本低等优点而备受青睐,日趋广泛地应用在了局部战争中,在越南战争、中东战争、海湾战争、科索沃战争及阿富汗战争这五次战争中崭露头角<sup>[13]</sup>。

从智能化程度上来说,早期的无人机系统大都是开环系统,比如“凯特灵空中鱼雷”号,环境中的任何变化,如风干扰,都可能导致其系统无法继续工作<sup>[14]</sup>,现代系统中已普遍采用了闭环系统设计来解决这个问题。与此同时,在无人机实际的应用中又产生了其他的功能需求,如自主目标选择、火力分配,这就需要无人机系统变得更加智能。在 20 世纪 80 年代早期,无人机系统在一定程度上已经拥有了自主决策能力,例如,美国的 AGM-136A“沉默彩虹”反辐射无人机,这款无人机在当时已具备搜索目标以及自动生成攻击航路的能力,其可以从一架飞机上起飞后,在预定的区域上空盘旋搜索目标,一旦侦测到敌人的雷达,立刻对其进行攻击<sup>[15]</sup>。从 20 世纪 90 年代末到现在,无人攻击机的研制越来越得到人们的关注,诞生了如波音 X-45a 这样的机型,其拥有更先进的传感器与武器系统,并被赋予了自主制定任务决策的能力<sup>[16]</sup>。

从数量上而言,世界各国小型无人机数量在本世纪初已超过 4 万架,并以每 10 年翻一番的速度继续发展,至 2010 年全球无人机数量已达到 12 万架,超过全部有人驾驶飞机 1 倍以上<sup>[17]</sup>。目前,美国、俄罗斯、欧洲各国、以色列、中国乃至日本、澳大利亚都在投入大量人力物力进行无人机的研制与实战部署,尤以美国和以色列发展最快。

目前军用领域而言,无人机可以归类但不限制于以下几类:

#### (1) 自杀式无人机

自杀式无人机正如其名,机体内部填装炸药,作战时对目标发动自杀式袭击。其作战原理为无人机发射升空后对敌方进行搜索、识别、跟踪,并在锁定目标后,自动调整航线高度,向目标俯冲,最终引爆炸药将目标摧毁<sup>[18]</sup>。其与巡航导弹存在本质的区别,巡航导弹的打击目标指派不是该导弹自己决定,而是由后方的导弹控制人员决定,导弹不具备主动搜寻目标并确定攻击目标的能力,攻击必须依赖导弹控制人员的指令。自杀式攻击无人机实现了察打一体化,自己寻找目标,并攻击。以色列“哈比”反雷达无人机就是此类无人机,不但可以自主搜寻目标并攻击,目标丢失后还能重新升空继续搜寻<sup>[19]</sup>。

#### (2) 无人侦察机

无人侦察机是当今无人机在军事领域成功应用的代表,无人机战场侦察已经成为情报获取的重要手段<sup>[20]</sup>。从“火蜂”到“全球鹰”<sup>[21]</sup>,从“彩虹”到“翔龙”,各国的侦察无人机都在积极为作战行动提供战场情报,给战场指挥决策提供数据信息的支持。现代战争中,获取的信息量以及信息的质量是掌握战场主动权的关键,无人侦察机由于其有别于有人侦察机等其他侦察手段的多种优势<sup>[22]</sup>,其丰富的挂载载荷选择:多功能雷达、光电探测设备、CCD 相机等可应对不同的战场侦察需求。无人侦察机已成为无人机中占比例最多的机种,也是当今无人机发展的主流。

#### (3) 电子战无人机

电子战无人机又称电子对抗无人机,是通过在无人机上安装专门的电子对抗载荷,对敌方通讯、雷达等系统具有较强的侦察、干扰能力<sup>[23]</sup>。依赖此无人机,不仅可以确定敌方的位置,还可以对敌方无线电设备进行干扰压制,让敌“看不见,听不到”。多架电子战无人机编队协同,可实现对目标辐射源的准确定位,以进行后续的高效干扰,或者引导武器系统对高价值军事目标进行精确打击<sup>[24]</sup>。目前电子战无人机向着软硬杀伤一体化、智能化、小型化的方向发展,在未来信息化的战场上,展现出越来越重要的应用价值<sup>[25]</sup>。

#### (4) 打击型无人机

当前打击型无人机主要还是依赖于无人侦察机的平台,进行相应改装,使其具备打击能力,如美国的“捕食者”无人机,其最早编号为 RQ-1,是一款无人侦察机,后加装武器装备,调整编号为 MQ-1<sup>[26]</sup>,由此改装而来的打击型无人机仍具有大翼展、低速、载荷外挂的侦察



无人机特点,如图 1.1 所示。目前越来越强调打击型无人机的隐身能力、机动性和生存能力,因此,这样的改装只是一种暂时的打击型无人机解决方案,需要打击型无人机尽早摆脱侦察机的平台,使其各方面特性更符合“打击”的主业。



图 1.1 无人机示意图

## 1.2.2 无人机协同任务规划国内外现状

为了拓展无人机在现代战场中更广泛的应用,无人机已经由单纯的飞行平台发展成为作战系统中的子部分。进入21世纪后,美军提出用“无人机系统”取代“无人飞行器”的概念<sup>[27]</sup>。为了实现多无人机平台及其武器传感器之间的协同工作,美国陆军首先发起了无人机自主协同作战项目,并由罗克韦尔科学公司开发并验证多无人机协同作战能力<sup>[28]</sup>,其研究内容主要涉及无人机协同作战的开发,需求范围从协同路径点编队飞行到多种类无人机协同对敌防空压制(Suppression of Enemy Air Defenses, SEAD)等等。2015年,乔治亚理工大学联合美国卓越机动中心、美国陆军坦克自动化研发与工程中心等机构,开展了无人机蜂群协同作战实验研究。该研究认为无人机正在越来越智能化,蜂群协同化作战能够更有效的完成任务和获取信息,同时也更具战场杀伤力,研究的最终目标是确定大量小型无人机组成的蜂群协同完成战场作战任务的能力。相比于无人机领域比较领先的美国,欧洲在无人机方面的研究相对落后。据悉,欧洲的COMETS项目实现了一种由无人直升机和无人飞艇组成的异构无人机系统,主要运用于环境感知,包括火灾探测和监测,以及地形测绘等方面<sup>[29]</sup>。

相对国外的研究成果而言,当下我国的无人机协同作战方面的研究相对落后,在各类无人机兼容性和多场景需求下远不能满足实际的需求。以任务规划方面为例,目前主要依靠任务规划相关的算法工程师依据任务场景的要求去对应设计规划算法,再将结果发布给特定的无人机。国内某中航工业研究所目前已开发出了一套任务规划服务器,可根据根据战场环境,无人机组成等产生任务航路,但该任务规划服务器并不能在线实时产生航路,只能实现离线规划<sup>[30]</sup>。综上所述,当今无人机协同作战是大势所趋,我国在这方面想要做到国际领先水平尚需一定的努力。

现主要从任务分配和路径规划上分析当前的无人机协同任务规划方面研究进展。无人机任务分配是一个比较经典的研究课题,一直受到国内外相关学者广泛的关注<sup>[32][33][34][35]</sup>,国外学

者较早的开始了此方向的研究<sup>[36]</sup>，所使用的方法除遗传算法、粒子群算法等群智能算法外<sup>[33][34]</sup>，还有一些其他创新的方法，Nygard等人针对无人机任务分配提出了一种网络流最优模型，通过将网络优化问题转化为线性规划问题解决了无人机的任务分配决策<sup>[37]</sup>。Sujit等人面对无人机之间无法通信，感知同伴和敌方能力都受到限制时，要实现多无人机联合侦察打击，提出了一种团队理论(Team Theory)来解决这种受限情况下的无人机任务分配问题，并将此方法的效果与使用贪婪算法的效果相对比，结论为即使在传感器受限、无相互通信的情况下，团队理论也有较好的分配效果<sup>[38]</sup>。

国内对于无人机任务分配的研究虽然起步相对国外晚，但是也已有一定数量的相关成果。常见的解决方法主要有：聚类算法<sup>[39]</sup>、分层策略<sup>[40]</sup>、遗传算法<sup>[41][42]</sup>、粒子群算法<sup>[43]</sup>、蚁群算法<sup>[44]</sup>等方法，文献[39]使用了模糊C-均值聚类算法，完成了任务点到各无人机的初分配，具有较好的时效性特点。文献[42]改进了遗传算法，通过设计一种特殊的矩阵形式的染色体编码方式，实现了受时序和任务类型约束的异构无人机机群的任务分配。文献[44]中利用蚁群算法解决了一种带有任务时间、时序约束的任务分配问题，并受现实中蚂蚁分工协作的启发，提出了利用分工机制改进蚁群算法。

路径规划方面，其定义就是给定起始点与目标点，在多种约束条件下，寻找到一条符合特定指标的最优路线<sup>[45]</sup>，在许多领域都有其应用的身影，如地面机器人自主避障行走<sup>[46]</sup>，无人机规避航路中的威胁<sup>[47]</sup>，汽车导航系统等等，甚至在游戏中，用鼠标选择目的地后，游戏角色自动前往目的地也是依赖路径规划完成的<sup>[50]</sup>。路径规划的研究是国内外一个经久不衰的热点，从深度优先搜索、广度优先搜索、迪杰斯特拉算法等经典算法，再到队列优化算法(Shortest Path Faster Algorithm, SPFA)算法、Floyd 算法、Bellman Ford 算法、A\*算法等精确算法，目前已知的成熟导航应用中，大多以此类精确算法作为其核心导航方法。有观点指出，A\*改进算法、国外提出的压缩分层算法(contraction hierarchies)以及高速分层算法(highway hierarchies)支撑了目前主流的路径规划软件。文献[48]中考虑到了气象因素与空中管制的影响，提出使用迪杰斯特拉算法对军用运输机起飞点至降落点之间的最短航路进行规划，文献[49]中因A\*算法方法简单速度快灵活高效等特点，将其运用到无人机航路规划中，实现了对地形以及防空火力的合理规避。同时由于智能算法的兴起，遗传算法、粒子群算法、蚁群算法等也被应用到路径规划中，已有相当数量的相关文献资料产出。其对模型的要求不高，在处理约束方面较为简单，在当下也有一定的应用<sup>[51]</sup>。但是智能优化算法依赖迭代寻优，导致规划的耗时较长，也容易陷入局部最优，同时由于优化算法很多具有一定的概率性，导致时常会产生不相同的结果，而且结果通常只为次优而不是最优解等等。这些缺陷的存在，使得智能优化算法若想在路径规划中得到更广泛的应用，还需要更加深入的研究与改进。

### 1.2.3 无人机意图识别与攻击决策技术国内外现状

2001 年 10 月,“捕食者”无人机首次挂载“地狱火”导弹,在阿富汗进行了一次对地打击任务,拉开了无人机对地打击的序幕<sup>[26]</sup>。自此,无人机的作用逐渐由战场侦察转向战场作战,无人机正式成为空中作战的一员。如何实现无人机的自主攻击决策一直是一个军事领域研究的问题,早在 10 多年前,俄罗斯苏霍伊飞机实验设计局就开始设计智能化的无人战斗机,但由于硬件等相关技术的限制,最终项目流产。依照摩尔定律,10 年后,微处理芯片的处理能力将超越人脑,硬件的存储能力也将达到人脑的水平,这将为无人机的智能化自主化提供足够的平台基础,参照美国国防部所提出的《美国无人机发展路线图 2005—2030》,未来的十年,无人机将完全从人为操纵进化为自主控制自主决策<sup>[52]</sup>。

本文所研究的无人机攻击决策问题是一个基于意图的决策过程,即在作出决定之前,首先需要参考敌人的目的,进行敌方意图的识别。从军事运筹学研究的角度,对意图识别进行定义:根据战场上的传感器和侦察装置获得的情报数据,确定战场上敌方平台的位置,速度,类型和数量,对战场情况以及敌方的行动意图做出判断。意图识别相关的理论研究刚刚起步,已有少部分的研究成果,主要利用了:专家系统,贝叶斯网络,神经网络等较为经典的方法<sup>[53]</sup>,目前随着人工智能方向的发展,也有学者提出使用深度学习的方法进行意图识别<sup>[54]</sup>。

无人机攻击决策就是研究如何将我方无人机的火力合理的分配至敌方无人机的问题。目前,有关无人机空战攻击决策已有许多的研究成果,许多方法被成功应用到其中。国外学者 Austin 早在 1990 年就提出了基于博弈论的一对一空战低速飞行器智能决策方法,决策的过程中综合考虑了双方的航向、位置、速度等因素,此方法已在美国宇航局艾姆斯垂直运动模拟器上完成验证<sup>[55]</sup>。学者 Rosenberger 针对多平台多目标火力分配问题,分别使用了拍卖算法和分支定界策略,为每个平台分配一个相应的打击目标,并对两种方法的效果进行对比,最终选用了分支定界策略<sup>[56]</sup>。国内学者在打击决策方面所使用的方法也比较丰富,如满意决策方法<sup>[57][58]</sup>、贝叶斯网络<sup>[59]</sup>、遗传算法<sup>[60]</sup>等。文献[58]中使用了满意决策方法应用于多无人机协同对地面目标打击分配,通过建立拒绝函数和选择函数,得到一种令人“满意”的分配结果而不是最优的分配结果。文献[59]中根据先验知识,获得节点之间的条件概率和状态转移概率,由此建立了一种基于动态贝叶斯网络的空战决策模型,以此来模拟真实飞行员的决策过程。文献[61]针对超视距空战下的无人机决策问题,对传统粒子群算法中容易陷入局部最优的缺点,在算法中引入遗传算法的变异和交叉操作,提出了遗传粒子群算法,并用仿真证明了方法的可行性。

目前绝大部分的文献中所解决的无人机意图识别与攻击决策问题,都是基于一种完备信息的前提下,非完备信息下的空战意图识别相关的文献数量十分稀少,属于一个新兴研究课题,非完备信息下的攻击决策问题的已有少量相关文献,但所使用的方法远没有完备信息下

的多样。文献[62]中,面对战场态势存在的不确定,首先建立了不确定信息下无人机模糊态势模型,并提出了一种异步一致性拍卖算法用于解决此种态势模型下的空战决策问题。文献[63]中,对于敌我毁伤概率、敌方阵地价值、武器价值等信息都认为是不确定的情况下,定义了一种敌我支付函数,同时使用了纳什均衡方法,为不确定不完备信息下的无人机攻防决策提供了新思路。在其他有限的文献中,多属性方案排序法(Stochastic Multi-criteria Acceptability Analysis, SMAA)<sup>[64]</sup>、粒子群算法<sup>[65]</sup>等也在逐步被应用到非完备信息下的空战决策中。以上的文献研究都具有一定的理论价值,但非完备信息下的空战决策问题是一个崭新的并且非常实际的军事应用课题,目前距离理论成熟有着较大的研究空间,需要探索更多的思路与方法。

### 1.3 本文主要研究内容

本文以无人机协同对地目标侦察攻击以及无人机协同空战为背景,设计了无人机协同作战系统,结合实际场景中对地作战的相关约束与空战中的态势,对多机任务分配、航路规划、任务重分配以及空战意图识别、多机攻击决策等问题进行研究。

第一章为绪论,首先分析了无人机协同作战以及非完备信息下的意图识别与攻击决策问题研究的必要性。简要介绍了无人机的发展历程以及当今世界中主流的无人机类别。针对无人机协同任务规划、意图识别与攻击决策技术的实际应用情况以及理论研究现状进行了分析,并指出了一些不足之处。

第二章为无人机协同作战系统介绍,本章首先将所需要设计的无人机协同作战系统分为协同任务规划系统以及攻击决策系统,对两个子系统进一步的划分为:任务分配模块,航路规划模块,任务重分配模块,意图识别模块以及攻击决策模块,以此作为接下来几章工作的主要内容。接着对协同任务规划系统进行分析,详细描述了场景中的任务点约束与地图约束,设计了航路预估方法,定义了相关的可达性矩阵,为下一章的任务分配部分做好铺垫。针对攻击决策系统中的意图识别与攻击决策问题,根据空战态势信息建立了5种威胁因子,并使用区间数表示威胁因子。

第三章为协同任务规划系统中的任务分配技术。在上一章定义的可达性矩阵的基础上,参照静态链表的结构,设计了一种遗传算法,定义其独特的染色体编码,交叉、变异过程,在相应的目标函数下完成任务的分配。

第四章为协同任务规划系统中的编队航路规划技术,首先介绍了A\*算法的基本内容,对不同的启发函数进行介绍,并通过仿真分析其特点,为系统选择一种合适的启发函数。同时针对A\*算法中的运行速度问题,提出了对ClosedList查找速度提升的改进,一定程度上提高了算法的速度与规划效果。

第五章为协同任务规划系统中的编队任务重分配技术，考虑存在事先侦察所获的战场信息的不充分以及战场中的不确定性，分析了作战过程中需要重分配的情况：任务中新被侦察到的威胁区域、新增任务点、虚假任务点以及一些其他突发状况：无人机战损、人为临时任务调整。介绍并改进了合同网协议，将其使用到上述的重分配中。

第六章为非完备信息下的目标意图识别与空战决策技术。着重研究在具体空战中，在非完备信息的情况下，如何根据历史空战信息识别当前态势下敌方无人机的意图，并将其作为一个重要的参考因素，运用到下一步的多机空战决策中。空战决策中对匈牙利算法进行改进，以实现在态势信息不确定的情况下完成打击分配

第七章为协同任务规划系统仿真软件的具体设计。在 Microsoft Visual Studio 2015 平台上利用 MFC 技术将无人机协同任务规划系统进行设计与实现。

第八章为总结与展望，首先总结了本文的工作内容，同时指出了本文的一些不足之处，对未来的工作提出建议与展望。

## 第二章 无人机协同作战系统

### 2.1 引言

无人机在逐步取代有人机的过程中，其单机作战功能越来越强大，系统集成越加复杂，在战场上的分工也越来越明细。现代战场中复杂多变的情况，需要无人机延续有人机的协同作战模式，这样不仅可以提高无人机系统的信息全面性，也可以实现无人机之间的资源互补，不论从生存能力的提升还是战场优势的建立而言，都具有必要性。一般而言，无人机相互协同所带来的收益将超过单机各自为战带来的收益总和，主要体现在如下三个方面<sup>[5]</sup>：

#### (1) 信息共享

当每架无人机都携带能够感知环境载荷的载荷时，可通过通信网络与编队的其他成员进行信息共享，此时整个编队都可以基于这种全局态势感知。

#### (2) 资源管理

各无人机之间独立作战时，会导致一些任务的重复执行。当无人机进行协同作战时，依靠系统中的分配算法，可以有效地将无人机资源分配到各个任务上。

#### (3) 鲁棒性

如果无人机相互独立地在作战区域中执行任务时，其中一架的退出都会导致一些任务的未执行。协同作战则可实现任务的重新分配，最大程度的保证了任务的完成率。

本文以无人机协同对地目标侦察攻击以及无人机协同空战为研究背景，设计了一个无人机协同作战系统，作战系统对于对地作战和对空作战两种模式，分别设计了无人机协同任务规划系统与无人机攻击决策系统，两个系统的具体组成部分在下一节进行介绍。

### 2.2 系统组成

系统结构如图 2.1 所示，下面分别介绍各模块的内容。

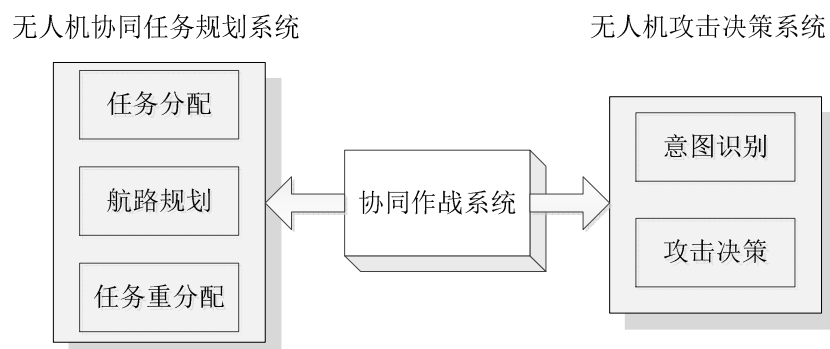


图 2.1 协同作战系统结构图

无人机协同任务规划系统，意在建立对地作战中无人机与作战任务之间的关联和映射关系，使无人机体系作战性能在一定指标下达到最优化水平，并能做到应对作战的不确定性与动态性<sup>[66]</sup>，能进行有效的计划调整与冲突消除，充分发挥出无人机协同作战的优势。

任务分配模块，或称为任务预分配，是根据我方获取的敌方信息，将任务进行合理的分割与分配到我方各编队，本模块主要内容为设计任务分配算法，算法中能够充分考虑到任务场景中的目标多类型、多约束，得到一个合理的分配结果，相关内容详见第3章。

航路规划模块，首先是建立在完成上述任务分配的基础上，对已拥有任务的无人机编队进行任务航路的规划，主要内容为设计一种航路规划算法，使得编队可以在飞行代价最小的航路指引下，依次完成任务。航路规划同时也要为下一步任务重分配负责，当在任务执行过程中，出现突发情况，触发了任务的重分配，此时系统需要及时根据任务的调整情况，快速的进行航路的重新规划。因此，航路规划模块的算法设计，不仅需要考虑规划效果，在实时性方面也有着较高的要求，相关内容详见第4章。

任务重分配模块，任务分配模块主要是在当前已有信息的基础上进行的预分配，在任务执行的过程中，无可避免的会有各种突发情况，兵法有云：知己知彼百战不殆，一个协同任务规划系统不仅需要能够进行任务的合理预规划，还要有着多种突发情况的及时处理能力，这就需要事先总结出可能出现的突发情况，针对每种情况制定相应的应对措施，即任务重分配能力，相关内容详见第5章。

当任务目标是空中的敌方无人机时，在我方对其执行打击任务时，需要考虑攻击决策的问题。无人机攻击决策系统针对具有不确定性和不完备性特点的现代化空对空战场信息环境，建立无人作战飞机空战模型，综合利用各种辅助信息、历史态势信息和当前态势信息完成对敌方目标的意图识别，并据此提出基于意图识别的自主攻击决策算法。所研究的基于意图的自主决策技术有利于提高未来战争对目标打击的突然性、快速性、准确性，提高武器系统的作战效能，降低战争成本，使无人作战飞机做到“先敌发现，先敌开火”，最大程度上消灭敌人并保存自己，相关内容详见第6章。

## 2.3 无人机协同任务规划系统分析与建模

### 2.3.1 无人机协同任务规划系统场景描述

将任务区域内需要执行的任务总和，称为总任务(mission)。对总任务的描述主要通过三个方面：任务区域、任务执行方、任务被执行方。

#### (1) 任务区域

任务区域，即是指定的无人机编队执行所有任务的空间范围限制，本文设定为一定范围内的矩形陆地空域，其他类型区域如海面空域的也可同样分析。

## (2) 任务执行方

无人机作为任务的执行方，受到其自身软硬件设备的限制，这些约束包括：

1) 载荷功能约束：协同任务规划系统中的无人机类型共分为四种：侦察探测无人机(携带光电传感器、多功能雷达)；电子战无人机(携带雷达探测设备、雷达压制设备)；攻击无人机(携带光电吊舱、挂载武器)；毁伤评估无人机(携带 CCD 相机)。

2) 运动学特性约束：这是由无人机的发动机与气动特性等决定的，主要考虑无人机的飞行速度、最小转弯半径和无人机的航程约束。协同任务规划系统中的各类无人机之间仅挂载设备不同，其余动力学属性完全相同，飞行速度设为匀速  $v$ 。

3) 载荷资源约束：对于攻击类无人机而言，主要表现在受到其机载弹药数量的限制，每架只能进行有限次数的打击任务；对于其余类型无人机而言，其载荷资源没有数量约束。

## (3) 任务被执行方：任务点(Task Point)

将拥有任务目标，需要编队对其执行相应任务操作的任务地点定义为一个任务点。对任务点的描述主要通过一下几个方面：任务点编号、任务点坐标、任务点执行时间约束、任务点内目标信息。

任务点编号为任务点相互区分的序号，对于任意一个任务点，记为  $TP_i$ ， $TP_i \in T$ ， $T$  为任务区域内所有任务点的集合。任务点坐标为任务点在任务区域内的空间坐标，对于  $TP_i$ ， $(x(TP_i), y(TP_i))$  为其在任务区域中的坐标。任务点  $TP_i$  执行时间约束记为  $[STime(TP_i), ETime(TP_i)]$ ，具体含义在下文做出说明。任务点内的目标可以为如下四种类型：防空导弹(携带雷达与防空武器)，防空高炮(携带防空武器)，雷达(携带雷达)以及敌方地面建筑(无雷达无防空武器)，同一个任务点内的目标在地理位置上非常靠近，坐标几乎相同，一个任务点内至少有一个目标，任务点上的任务就是我方具体需要对任务点内的目标执行的任务操作，即目标产生任务。

表 2.1 任务点  $TP_i$  内目标信息

编号	坐标(Km)	执行时间约束 (Min)	目标编号	雷达探测范围 (Km)	武器打击范围 (Km)
$TP_i$	$(x(TP_i), y(TP_i))$	$[STime(TP_i), ETime(TP_i)]$	$TP_i(\alpha)$	$R_{RA}(TP_i(\alpha))$	$R_{ATK}(TP_i(\alpha))$
			$TP_i(\beta)$	$R_{RA}(TP_i(\beta))$	$R_{ATK}(TP_i(\beta))$
			...	...	...

任务点内目标信息包括：任务点内目标编号，目标雷达探测范围，目标武器攻击范围，具体用表 2.1 中数据结构表示。表 2.1 中，对任务点  $TP_i$  内的敌方目标进行相应的编号，以相互区分， $TP_i$  中一目标  $TP_i(\alpha)$ ，存在雷达探测范围  $R_{RA}(TP_i(\alpha))$  和武器打击范围



$R_{ATK}(TP_i(\alpha))$  两个属性,  $R_{RA}(TP_i(\alpha))$  大于 0 代表此目标带有雷达, 否则不带有, 同理  $R_{ATK}(TP_i(\alpha))$  大于 0 代表此目标带有对空武器, 否则不带有。通过两个属性的值不仅可以描述当前目标的雷达和武器的性能, 还可以判定这是敌方哪种类型的目标, 例如, 当  $R_{RA}(TP_i(\alpha)) = 0$ ,  $R_{ATK}(TP_i(\alpha)) > 0$  时, 表示此目标无雷达但带有防空武器, 则  $TP_i(\alpha)$  为一防空高炮。下一小节具体对任务点所受到的约束进行分析。

### 2.3.2 任务点约束模型

#### (1) 任务类型约束

系统中以多机组成的编队作为单位进行研究, 编队执行任务时, 由于需要对任务点内每个目标执行多种任务操作, 不同的目标类型需要被执行不同的任务, 对编队内的无人机机载设备类型以及数量有相应的要求, 具体如表 2.2 所示。

表 2.2 各类型目标所需执行任务与无人机需求表

目标类型	能力属性	我方需执行任务	执行所需最少无人机
防空导弹	1、雷达探测 2、对空武器	对目标进行电子压制与定位、目标打击、目标毁伤评估	两架电子战无人机、至少剩余一枚导弹的攻击无人机、一架毁伤评估无人机
防空高炮	1、对空武器	对目标进行探测与定位、目标打击、目标毁伤评估	一架侦察探测无人机、至少剩余一枚导弹的攻击无人机、一架毁伤评估无人机
雷达	1、雷达探测	对目标进行电子压制与定位、目标打击、目标毁伤评估	两架电子战无人机、至少剩余一枚导弹的攻击无人机、一架毁伤评估无人机
敌方地面建筑	无	对目标进行探测与定位、目标打击、目标毁伤评估	一架侦察探测无人机、至少剩余一枚导弹的攻击无人机、一架毁伤评估无人机

#### (2) 任务点时间约束

##### 1) 单任务点时间约束

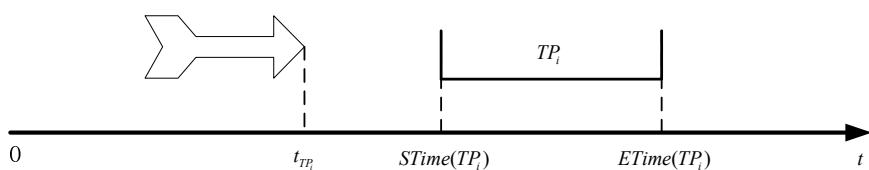


图 2.2 单任务点时间约束示意图

对于任务点  $TP_i$ ，该任务点有属性任务点执行时间约束  $[STime(TP_i), ETime(TP_i)]$ ，无人机编队必须在此时间窗口内执行任务点  $TP_i$  上的任务。如图 2.2 中，无人机编队到达  $TP_i$  的时间设为  $t_{TP_i}$ ，当  $t_{TP_i} < STime(TP_i)$ ，编队需要等待  $STime(TP_i) - t_{TP_i}$  后开始执行任务，由于编队内无人机都为固定翼，在等待的过程中将进行小半径盘旋等待，并在  $ETime(TP_i)$  执行完当前任务点的任务。当  $t_{TP_i} = STime(TP_i)$ ，则编队无需等待，立刻执行任务。如果  $t_{TP_i} > STime(TP_i)$ ，则编队错过了时间窗口，无法完成任务。

### 2) 任务点间的时间约束

多个任务点之间，由于各自执行时间约束，使得任务点之间存在两种时序关系：串行与并行。如图 2.3 所示，对于当前的三个任务点  $\{TP_a, TP_b, TP_c\}$ ，各任务点的执行时间约束如图，存在串行关系  $\{TP_a, TP_b\}$  与  $\{TP_a, TP_c\}$ ，即  $TP_a$  的任务一定在  $TP_b$  与  $TP_c$  的任务之前被执行。存在并行关系  $\{TP_b, TP_c\}$ ，由于任务点  $TP_b$  与  $TP_c$  之间的执行时间存在重合，两个点上的任务必须由不同的编队分别执行，不可存在于同一编队中被顺序执行。

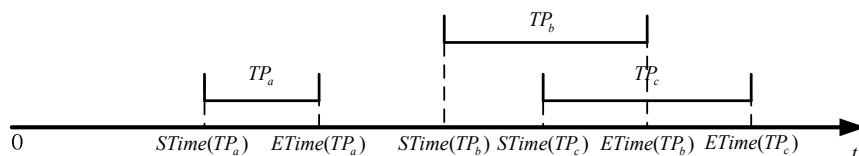


图 2.3 任务点间的时间约束示意图

### 3) 任务点内多任务时序约束

一般而言，编队执行任务点中目标上的任务时，若目标拥有雷达设备，需要首先进行电子压制软杀伤，使其雷达失效，再进行目标定位、目标打击，最后进行毁伤评估。若目标无雷达设备，则首先使用我方雷达进行探测与定位，获取目标具体位置信息后，攻击型无人机面对敌方目标进行相应的火力打击，最后进行毁伤评估。总的来说，任务点内每个目标上任务时序满足电子压制与定位 → 目标打击 → 目标毁伤评估，或者雷达探测与定位 → 目标打击 → 目标毁伤评估的时序要求。

## 2.3.3 地图约束建模

无人机编队在执行任务的过程时，周围环境中会存在多种威胁需要编队规避，其中包括地形、气象、雷达和防空火力等。在进行航路预估和航路规划时，需要这些约束的模型，下面分别进行建模：

(1) 雷达威胁建模<sup>[67]</sup>

由于任务点内的目标可以为防空导弹(携带雷达与防空武器)以及雷达(携带雷达)这类配备雷达的固定目标,在编队航路经过此类任务点时,需要进行规避,防止被敌方所侦察。现假设敌方的携带雷达的设备都安置于开阔地,雷达的有效探测距离的边界可以用相对高度  $H$  与水平距离  $R$  之间确定的一个抛物线近似描述,在水平界面上为一个圆面。

## (2) 防空武器建模

防空武器主要为防空导弹与防空高炮,根据防空导弹的破坏特性,可以认为其杀伤区为一个腰鼓型区域<sup>[71]</sup>。在一定高度上攻击范围最广,并随高度的上下变化而缩小,在一定的程度上为圆面。高炮的攻击范围受到武器仰角范围的限制,杀伤范围为一个曲面和锥面的结合形状,在一定的程度上也为一个圆面。

(3) 地形与气象威胁建模<sup>[67]</sup>

地形威胁主要为高耸的山峰,用圆锥体近似表示,由于假设的场景为海面 and 沿海区域,可能存在台风、降雨等复杂气象情况,编队如果从中穿过可能会对机载设备和无人机本身造成一定破坏,采用圆柱体模型来描述气象威胁。

## 2.3.4 航路预估

协同任务规划系统中首先需要进行的的就是任务分配,所分配给各编队的任务点需要满足如下条件:假设当前一个任务点子集  $\{TP_a, TP_b, \dots, TP_m\} \subseteq T$  分配给一个编队去执行,此集合内的各任务点已按照任务点执行开始时间  $S\text{Time}$  从小到大排序。

若一个任务点子集是可执行的,参照图 2.4,应有如下情况被满足:

(1) 从出发点出发,到达任务点  $TP_a$  的航程假设为  $dis(0)(TP_a)$  (0 代表出发点),编队到达  $TP_a$  时间为  $Tariv(0)(TP_a)$ ,  $Tariv(0)(TP_a) = dis(0)(TP_a) / v$ 。有  $Tariv(0)(TP_a) \leq t_1$ , 其中  $t_1 = S\text{Time}(TP_a)$ 。

(2) 对于任意两个连续的任务点之间,以  $TP_a$  与  $TP_b$  为例,  $TP_a$  与  $TP_b$  之间编队飞行的航程为  $dis(TP_a)(TP_b)$ , 编队执行完  $TP_a$  的任务后到达  $TP_b$  的时间为  $Tariv(TP_a)(TP_b)$ ,  $Tariv(TP_a)(TP_b) = E\text{Time}(TP_a) + dis(TP_a)(TP_b) / v$ 。有  $t_2 = S\text{Time}(TP_b) - E\text{Time}(TP_a)$ , 应满足  $Tariv(TP_a)(TP_b) - E\text{Time}(TP_a) \leq t_2$ 。

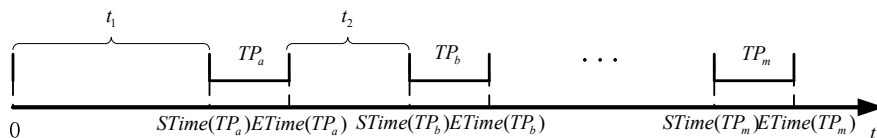


图 2.4 任务点执行时间示意图

定义一种可达性矩阵,以方便判断一个任务点子集是否符合上述条件。可达性定义分为

两种情况：对于任意  $TP_i, TP_j$ ，满足  $TP_i \in T, TP_j \in T$

(1) 根据从出发点到达任务点  $TP_i$  的航程  $dis(0)(TP_i)$ ，进而得到编队到达  $TP_i$  时间  $Tariv(0)(TP_i)$ ，对于任务点  $TP_i$ ，如果  $STime(TP_i) \geq Tariv(0)(TP_i)$ ，则认为任务点  $TP_i$  是出发点可达的，记为 1，否则为出发点不可达，记为  $Inf$ ；

**定义 2.1:** 矩阵  $Czt$  为出发点可达性矩阵，对于  $i=1,2,\dots,n$ 。

$$Czt_i = \begin{cases} 1 & STime(TP_i) \geq Tariv(0)(TP_i) \\ Inf & STime(TP_i) < Tariv(0)(TP_i) \end{cases} \quad (2.1)$$

(2) 根据从任务点  $TP_i$  出发到达任务点  $TP_j$  的航程  $dis(TP_i)(TP_j)$ ，可以得到到达任务点  $TP_j$  的到达时间  $Tariv(TP_i)(TP_j)$ 。对于任务点  $TP_j$ ，如果  $STime(TP_j) \geq Tariv(TP_i)(TP_j)$ ，则认为任务点  $TP_j$  是任务点  $TP_i$  可达的，否则  $TP_j$  为  $TP_i$  不可达，记为  $Inf$ ；

**定义 2.2:** 矩阵  $Ctt$  为任务点可达性矩阵，对于  $i,j=1,2,\dots,n$ 。

$$Ctt_{ij} = \begin{cases} dis(TP_i)(TP_j) & STime(TP_j) \geq Tariv(TP_i)(TP_j) \\ Inf & STime(TP_j) < Tariv(TP_i)(TP_j) \text{ or } i = j \end{cases} \quad (2.2)$$

当地图中没有需要避开的障碍时，无人机编队从某一点出发达到另一点所花费的最少时间取决于两点间的直线距离(假设编队处于匀速飞行状态)。在本文所假设的场景中，任务区域中存在两种类型的威胁区域：(1)静态威胁，主要为地形和气象威胁，静态威胁从始至终都需要编队规避；(2)动态威胁，在整个任务区域中，一个任务点  $TP_i$  内的敌方目标有可能携带雷达或者防空武器，对周围空域会产生一个动态威胁区域，有如下特点：此威胁区域的大小由任务点内目标的雷达以及防空武器最大影响范围决定，在  $ETime(TP_i)$  之前威胁区域将一直存在于任务区域中，航路在附近的我方编队需要进行规避，在  $ETime(TP_i)$  之后，认为敌方目标被消灭，威胁区域不复存在，此时再经过的话便不需要规避。

本章采用一种最坏情况分析法计算可达性矩阵，求解矩阵  $Ctt$  上的每个值的具体过程如图 2.5，描述如下：

Step1: 载入地图信息，输入与所求值对应的任务点  $TP_a$  与  $TP_b$ ；

Step2: 如果  $TP_a$  与  $TP_b$  是同一个任务点，或者  $ETime(TP_a) \geq STime(TP_b)$ ，则定义为不可达，值为  $Inf$ ，结束判定；否则执行 Step3；

Step3: 选取当前第一个任务点  $TP_i, i=1$ ；

Step4: 若  $TP_i$  不为  $TP_a$  以及  $TP_b$  中任何一个，同时  $TP_i$  内目标带有雷达或防空武器，执行 Step5；否则转 Step7；

Step5: 若  $ETime(TP_a) < ETime(TP_i)$ ，则编队可能进入  $TP_i$  雷达探测或攻击能力范围，则转 Step6；否则， $TP_a$  执行完成之前， $TP_i$  已被执行完成， $TP_i$  带有雷达探测或攻击能力已

不复存在，转 Step7；

Step6: 判定  $TP_i$  的威胁区域在空间上与时间上是否干扰到编队的航路，具体的过程描述见下文，如果任务点干扰到了编队航路，保存下任务点的航路绕过点用于航路的预估；

Step7: 判定  $TP_i$  是否为最后一个任务点，即  $i$  是否为  $n$ ，如果是，结束；否则， $i = i + 1$ ，转 Step4。

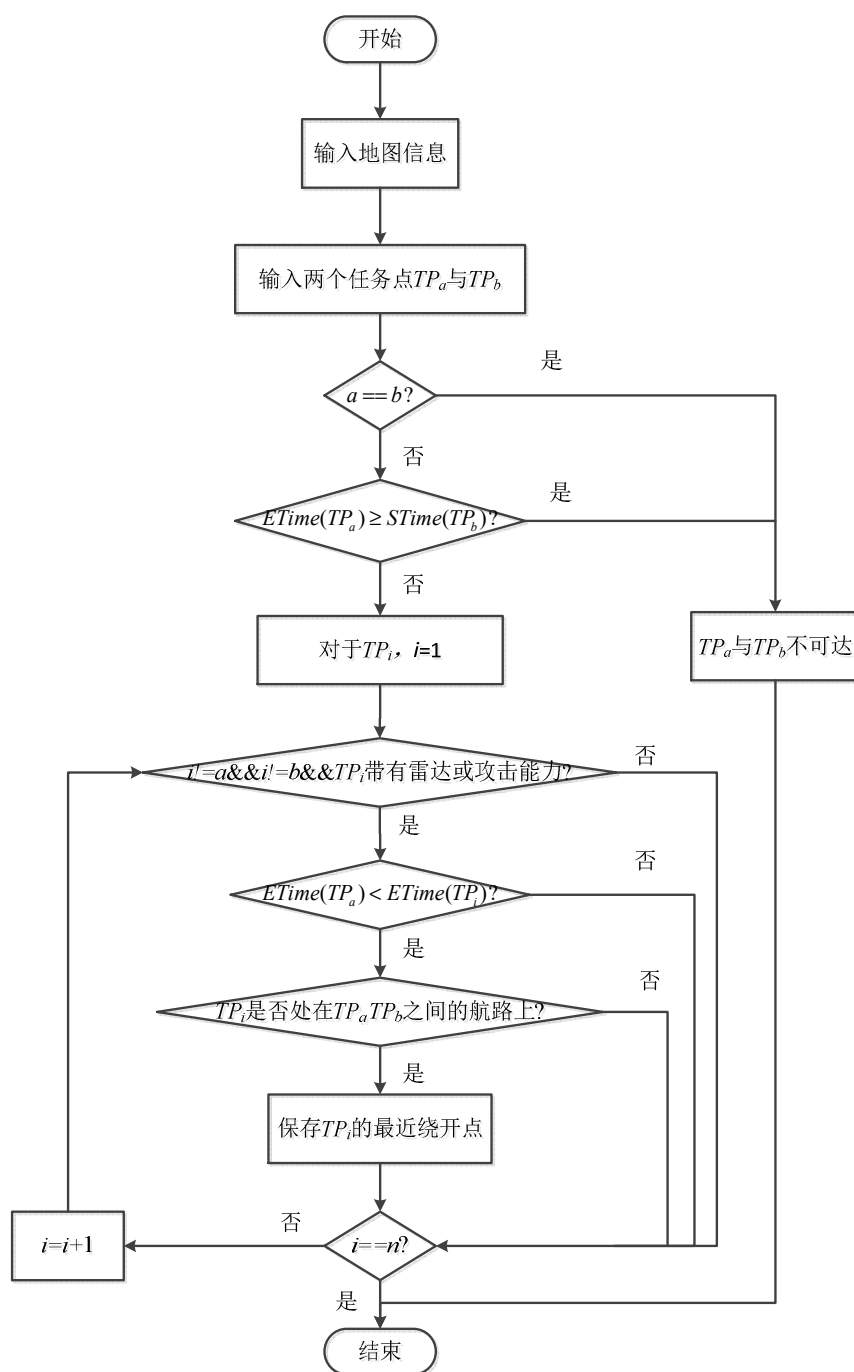


图 2.5 可达性矩阵内元素求解流程图

判定任务点的威胁区域在空间上与时间上是否干扰到编队的航路过程：

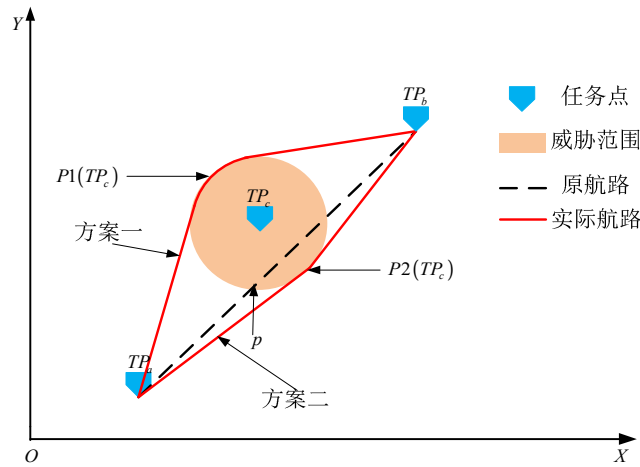


图 2.6 航路被干扰示意图

以图 2.6 为例，确认  $TP_c$  是否干扰到编队从  $TP_a$  到  $TP_b$  的航路时，首先进行空间上的判定， $r = \max(\max(R_{RA}(TP_c(\alpha)), R_{ATK}(TP_c(\alpha))), \max(R_{RA}(TP_c(\beta)), R_{ATK}(TP_c(\beta))), \dots)$  为  $TP_c$  的威胁区域半径，首先将  $TP_c$  转换到以  $TP_a$  为原点， $TP_a$  指向  $TP_b$  方向直线为  $X$  轴正方向的坐标系下，新坐标系下  $TP_c$  的坐标为  $(x'(TP_c), y'(TP_c))$ ，当满足  $0 \leq x'(TP_c) \leq \text{len}(TP_a)(TP_b)$ ， $|y'(TP_c)| < r$ ，如图 2.7，则认为空间上  $TP_c$  的威胁区域阻碍了编队的航路。 $(\text{len}(TP_a)(TP_b))$  表示  $TP_a$  与  $TP_b$  之间的直线距离。

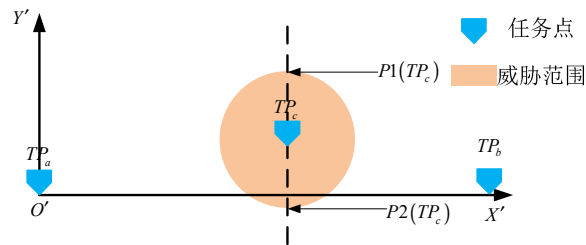


图 2.7 转换坐标系后航路被干扰示意图

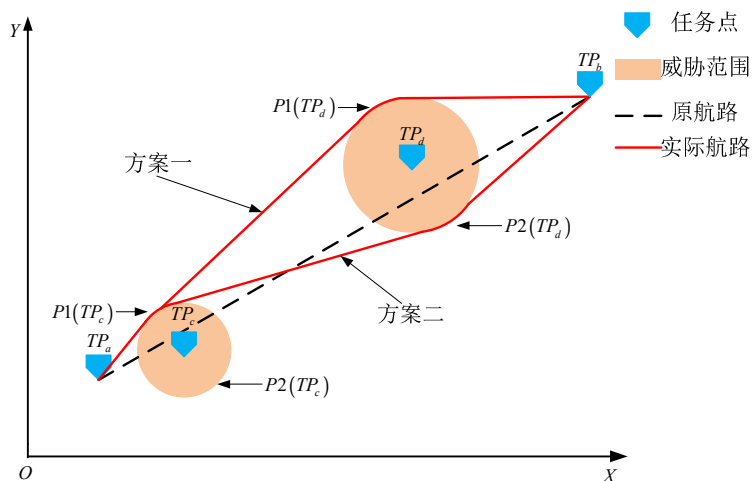
当编队面对可绕过的威胁时，如图 2.6，绕过路径有两种方案，所经过的两个点  $P1(TP_c)$  与  $P2(TP_c)$  为威胁区域被原航路所分割成的两部分上离原航路最远的两个点，称为  $TP_c$  上的绕过点，其在图 2.7 中新坐标系下的坐标为  $(x'(TP_c), y'(TP_c) + r)$  与  $(x'(TP_c), y'(TP_c) - r)$ 。

在空间上阻碍的情况下，再进行时间上的判断，如图 2.6 编队从  $TP_a$  飞向  $TP_b$  的过程中， $p$  为航路与  $TP_c$  的威胁区域的第一个交点，设编队到达点  $p$  时，时间为  $t_p$ ，当  $t_p < ETime(TP_c)$  时，认为  $TP_c$  会阻碍了  $TP_a$  到  $TP_b$  的航路。

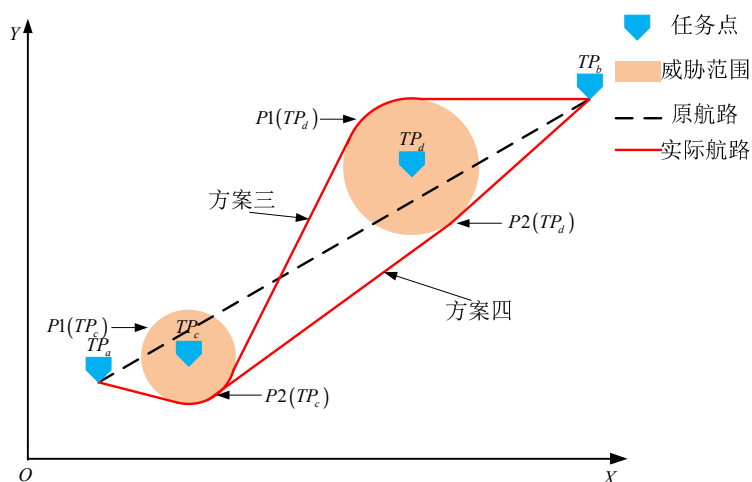
上述过程判定了任务点造成的动态威胁对编队的飞行路径的影响，对于静态威胁，编队的飞行航路必须要对其进行航路规避判断，由于静态威胁是长期存在的，可将其视为一个存在时间为  $[0, +\infty)$  的动态威胁，这样，不论动态与静态威胁，在进行航路距离预估时都可以用

如下过程得到预估的航路，航路预估采取如下策略：

第四章中的航路规划意图规划出的航路为最优航路，各编队既不进入威胁区域，也尽量保证了任务航程的最短，在绕过威胁区域时，所选择的都是最佳绕过方案。航路预估的结果虽不如航路规划的结果准确，但也应该尽量接近实际的航路规划，即在预估时，也选择最佳的绕过方案。对于所要获得航路预估值的任意两个任务点  $TP_a$  和  $TP_b$ ，有一集合  $B = \{TP_i \in T | TP_i \text{ 的威胁区域阻碍了 } TP_a \text{ 到 } TP_b \text{ 的航路}\}$ ，以图 2.8 例， $B = \{TP_c, TP_d\}$ ， $B$  中的每个元素都存在两个绕过点，估算航路时每个元素需要选择其中一个，为了获得全局最优的航路估计值，计算出所有的绕过点选择组合，图 2.8 中有四种绕过方案，选择其中航程最短的情况作为全局最优法得到的航路预估结果输出。求解  $Czt$  矩阵值时，将  $TP_a$  变更为原点，此时其求解方法与求解  $Ctt$  值的过程相似。



(a) 绕过障碍物方案一与二



(b) 绕过障碍物方案三与四

图 2.8 绕过威胁区域示意图

## 2.4 无人机攻击决策系统分析与建模

无人机攻击决策系统所基于的场景为无人机作战飞机UCAV的空对空作战, 下图2.9是空战战场中两敌对无人机的空战态势图<sup>[68]</sup>。其中,  $D$ 为双机目标线, 即我机 $i$ 到敌机 $j$ 的连线;  $\varphi$ 为我机位置角, 即目标线 $D$ 偏离我机 $i$ 速度 $V_i$ 方向的角度;  $\rho$ 为目标进入角, 即目标线 $D$ 偏离敌机 $j$ 速度 $V_j$ 方向的角度。规定 $\varphi$ 和 $\rho$ 右偏为正, 左偏为负,  $0 \leq |\varphi| \leq \pi$ ,  $0 \leq |\rho| \leq \pi$ 。

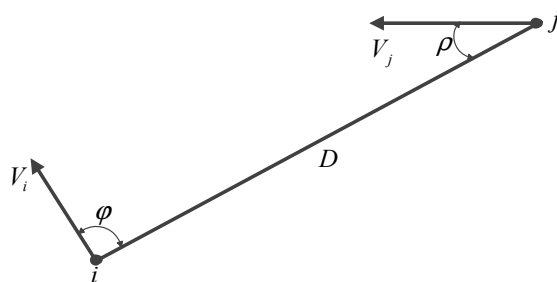


图 2.9 空战态势图

### 2.4.1 威胁因子定义

现在以我机 $i$ 作为研究对象, 根据敌机 $j$ 对我机 $i$ 的态势情况以及双方飞机之间的性能差异定义如下5个威胁因子:

#### (1) 角度威胁因子 $Ta$ <sup>[68]</sup>

角度威胁因子主要考虑敌机与我机之间进入角 $\rho$ 与 $\varphi$ 的关系, 具体定义如下:

$$Ta_{ij} = (\pi + |\varphi| - |\rho|) / 2\pi \quad (2.3)$$

式(2.3)中典型情况分析:  $\varphi = 90^\circ$ ,  $\rho = 90^\circ$ ,  $Ta_{ij} = Ta_{ji} = 0.5$ , 两机均势。

#### (2) 距离威胁因子 $Td$

距离威胁因子主要考虑我机是否处在敌机武器攻击区范围内, 若在则需要考虑两机之间的距离情况。具体定义如下:

$$Td_{ij} = \begin{cases} \exp(-((d_{ij} - R_0^j) / \sigma)^2) & R_{\min}^j < d_{ij} < R_{\max}^j \\ 0.1 & d_{ij} \geq R_{\max}^j \text{ or } d_{ij} \leq R_{\min}^j \end{cases} \quad (2.4)$$

式(2.4)中:  $d_{ij}$ 为沿目标线 $D$ 两机之间的距离,  $R_0^j = (R_{\min}^j + R_{\max}^j) / 2$ 。其中 $R_{\min}^j$ 和 $R_{\max}^j$ 是敌机武器攻击区的近边界和远边界半径。当 $d_{ij} \geq R_{\max}^j$ 或 $d_{ij} \leq R_{\min}^j$ 时,  $Td_{ij} = 0.1$ , 可得 $\sigma = 0.33(R_{\max}^j - R_{\min}^j)$ 。

#### (3) 速度威胁因子 $Tv$ <sup>[68]</sup>

速度威胁因子认为敌机相对于我机速度越快, 速度威胁越大。具体定义如下:



$$Tv_{ij} = \begin{cases} 0.1 & V_j \leq 0.6V_i \\ -0.5 + V_j / V_i & 0.6V_i < V_j < 1.5V_i \\ 1 & V_j \geq 1.5V_i \end{cases} \quad (2.5)$$

式(2.5)中：当两机速度相同时，威胁因子为 0.5，两机均势。

#### (4) 高度威胁因子 $Th$

在空战情况下，高度也是不容忽视的因素。如果一方在空战中占据高空优势，对武器的射速，武器的射程，发射范围都有一定的效果提升。具体定义如下：

$$Th_{ij} = \begin{cases} 0 & h_{ij} < -5km \\ 0.5 + 0.1h_{ij} & -5km \leq h_{ij} \leq 5km \\ 1 & h_{ij} > 5km \end{cases} \quad (2.6)$$

式(2.6)中： $h_{ij}$ =敌机  $j$  飞行高度-我机  $i$  飞行高度。当两机高度相同时，双方均势。

#### (5) 空战能力威胁因子 $E$

空战能力多年来我国一直采用经典的对数法公式，但在权重问题和参数选择问题等方面存在明显的不足，本节采用一种新的空战能力评估模型，公式如下：

$$S = k_1 \times C_p + k_2 \times C_A + k_3 \times C_M \quad (2.7)$$

式中： $S$  为战机的空战能力； $C_p$  为对目标的探测能力， $C_A$  为对目标的攻击能力， $C_M$  为自身的机动能力。 $k_1$ 、 $k_2$ 、 $k_3$  为大于零的权重，且  $k_1 + k_2 + k_3 = 1$ 。 $C_p$ 、 $C_A$ 、 $C_M$  的具体定义如下：

##### 1) 探测能力定义<sup>[69]</sup>

在空战对抗中，考虑传感器的探测能力限制以及目标可能具有电子干扰能力，导致无法准确感知，定义探测能力如下：

$$C_p = \frac{e^{kP_t} - 1}{e^k - 1} \quad (2.8)$$

式(2.8)中， $P_t$  表示当前态势下对目标的探测概率， $k$  为修正  $P_t$  对探测能力影响的参数。

##### 2) 攻击能力定义

现代空战中，导弹是主要的攻击手段，攻击能力的定义主要考虑机载导弹破坏能力  $\eta$  和导弹有效打击能力  $\lambda$  组成：

$$C_A = \gamma \times \eta \quad (2.9)$$

导弹有效打击能力  $\lambda$  的定义如下<sup>[69]</sup>：

$$\gamma = \begin{cases} 0 & R < R_{M \min} \text{ or } R > R_{M \max} \\ \frac{\pi - |\theta|}{\pi} & R_{K \max} \leq R \leq R_{M \max} \\ 1 & R_{M \min} < R < R_{K \max} \end{cases} \quad (2.10)$$

式(2.10)中:  $R_{M \max}$  为导弹射程,  $R_{M \min}$  为导弹最小发射距离,  $R_{K \max}$  为导弹不可逃逸攻击距离,  $R$  为与所要攻击的目标之间的距离,  $\theta$  为战机的速度方向与目标线之间的夹角。

### 3) 机动能力定义<sup>[70]</sup>

战机机动能力  $C_M$  的定义由操纵性能  $\lambda$  和机动性能  $\kappa$  组成, 如下式:

$$C_M = \lambda \times \kappa \quad (2.11)$$

根据上述(2.7)所定义的空战能力评估模型, 定义如下空战能力威胁因子:

$$E_{ij} = S_j / (S_i + S_j) \quad (2.12)$$

表示敌机  $j$  对我机  $i$  的空战能力威胁, 其中  $S_i$  为我机空战能力,  $S_j$  为敌机空战能力。

## 2.4.2 无人机攻击决策系统中的信息不完备

由于空战中无法获得敌方目标的准确相关信息, 导致上述所定义的 5 个威胁因子中存在着信息不确定, 由于区间数可以比较客观的刻画信息的确定性与不确定性<sup>[89]</sup>, 考虑使用区间数来表达空战信息。区间数的相关定义如下:

**定义 2.3<sup>[87]</sup>:** 有  $x^-, x^+ \in R$ , 且  $x^+ \geq x^-$ , 则称  $[x^-, x^+]$  为规范区间数, 简称区间数, 此区间数的上界为  $x^+$ , 下界为  $x^-$ 。区间数不是一个确定的数, 而是一个实数集对, 即

$$[x^-, x^+] = \{x \mid x^- \leq x \leq x^+\} \quad (2.13)$$

$x$  为区间数中的一个可能值, 其值不确定, 取值上界为  $x^+$  下界为  $x^-$ 。当  $x^+ = x^-$  时,  $[x^-, x^+]$  由区间数退化为一个常数。

## 2.5 本章小结

本章首先论述了协同作战的价值, 给出了一种无人机协同作战系统, 系统由协同任务规划系统部分和攻击决策系统部分组成, 分别使用在对地作战和对空作战两种情况下。接着, 介绍了协同任务规划系统部分的分析与建模, 并对系统的任务场景进行描述, 详细说明了任务点的类型约束、时间约束、地图约束等多种约束因素, 定义了可达性矩阵, 设计了航路预估方法, 以方便进行时间约束相关的计算, 下一章将在本章所定义的约束以及可达性矩阵的基础上进行任务分配。最后描述了无人机攻击决策系统中空对空作战的空战态势图, 依照此态势图定义了角度、距离、速度、高度以及空战能力 5 个威胁因子, 由于所定义的 5 个威胁因子存在不确定性, 提出用区间数来将其表述。

## 第三章 无人机任务分配技术

### 3.1 引言

任务分配是协同任务规划系统中一个需要首先设计的部分，多无人机的航路规划与重分配都是建立在任务分配完成的前提下。任务分配，即是将当前我方所需要前往执行任务的任任务点集合进行合理的划分，使得每个任务点子集都可被一个编队依照任务点执行时间全部执行。

任务分配问题本质上是一种组合优化问题，当问题的规模不大时，可以考虑使用穷举法实现，此时可以获得最优解，但是随着问题规模的扩大，穷举法将消耗大量的计算时间，使得穷举法很少被应用到实际中<sup>[71]</sup>。本章研究的任务分配为系统的任务预分配部分，对算法实时性要求不高，同时考虑到遗传算法的全局搜索的优点，其对模型的精确度要求不高，可实现多约束下的寻优，将遗传算法作为本章任务分配的主要算法。通过对算法过程合适的设计与调整，使其结合本文所假设场景，最终可以获得任务分配的结果。

### 3.2 任务分配问题分析

#### 3.2.1 任务分配前的数据准备

当指挥中心向协同任务规划系统发布总任务时，系统首先需要将总任务进行分解和细化，梳理出任务任务分配所需要的信息，流程如图 3.1。

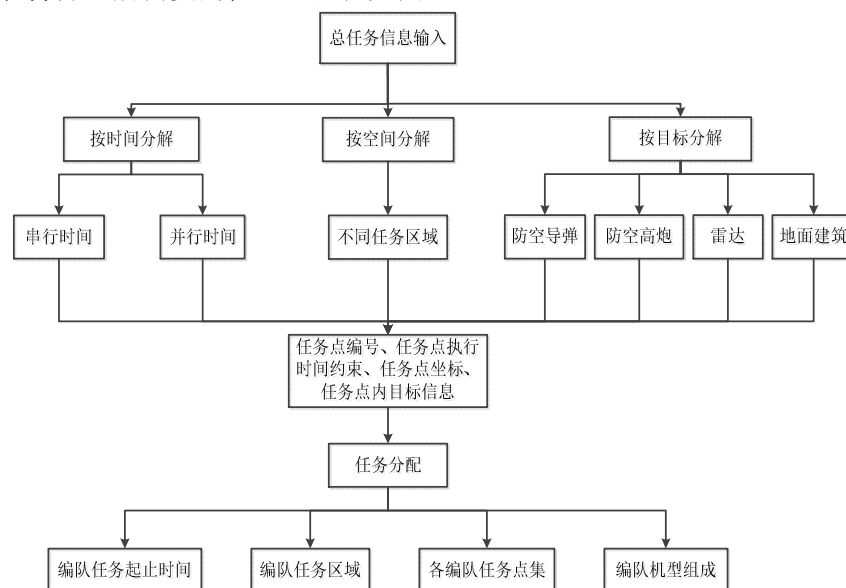


图 3.1 总任务信息分解过程示意图

具体过程如下：

Step1: 根据指挥中心发布的总任务，按时间分解出每个任务点的执行时间约束，得到任务点之间执行的时间串行与并行关系；

Step2: 根据指挥中心发布的总任务，得到任务区域的空间约束，并按空间分解出每个任务点的空间位置；

Step3: 根据指挥中心发布的总任务，分解出每个任务点内敌方目标的组成，并获得此任务点内所需要执行的具体任务内容；

Step4: 综合以上分解出来的内容，整理得到所有的任务点信息，包括任务点编号、任务点执行时间约束、任务点坐标、任务点内目标信息。

### 3.2.2 任务分配的目的

完成数据准备后，接下来可以进行任务分配，本文的分配对象为任务点，任务分配的过程就是将所有任务点进行划分成若干个子集，每一个子集都需满足 2.3.4 小节中的可执行条件。如图 3.2 中，假设当前任务点集合  $T = \{TP_1, TP_2, \dots, TP_n\}$ ，记一个任务点子集为  $T_i, T_i \subseteq T$ 。当分配后得到  $N$  个子集合  $T_1, T_2, \dots, T_N$ ，此时有  $T_1 \cup T_2 \cup \dots \cup T_N = T$ ，同时  $N$  个子集合任意两个之间的交集为空。由于每一个任务点子集都将分配给一个编队执行，则第  $i(i = 1, 2, \dots, N)$  个编队将执行任务点子集  $T_i$  中的任务。当划分出子集后，按照各子集任务点内的任务要求进行相应编队的分配，编队分配的具体过程在下一小节中描述。

分配后可以得到如下结果：编队执行任务的起止时间，编队任务区域，编队所分配到的任务点子集，以及根据任务点子集所确定的编队内无人机成员组成。

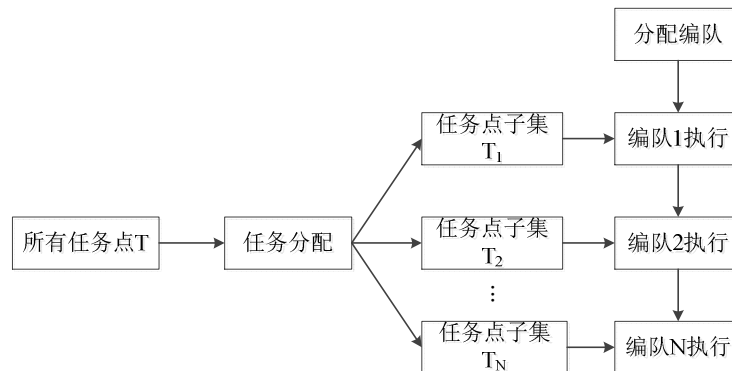


图 3.2 任务分配目的示意图

### 3.2.3 任务分配目标函数

本文的任务分配问题是一个多约束问题，前文已经对约束条件进行了定义和归类。对于这样的多约束问题，采用一种综合评价指标，即任务效能，来综合反应无人机完成任务所获

得的收益指标和付出的代价指标。

首先考虑代价指标，本章从无人机数量使用和航程代价两方面来决定当前分配方式的代价指标。确认无人机使用数量过程如下描述：对于每个编队  $i(i=1,2,\dots,N)$ ，其中的无人机组成确认过程如下：

Step1: 输入当前编队的任务点子集  $T_i$  与无人机需求矩阵，无人机需求矩阵用于保存当前编队对每类无人机需求数量，初始全为 0；

Step2: 选择任务点子集  $T_i$  中第一个任务点；

Step3: 选择当前任务点中的第一个目标；

Step4: 判断此目标雷达探测范围是否大于 0，如果是，转 Step5；否则转 Step6；

Step5: 设置当前编队电子战无人机数量为 2，携带弹药数量+1，毁伤评估类无人机数量为 1，转 Step7；

Step6: 设置当前编队侦察探测类无人机数量为 1，携带弹药数量+1，毁伤评估类无人机数量为 1，转 Step7；

Step7: 判断当前目标是否为当前任务点中最后一个目标，如果是，转 Step8；否则，选择任务点中的下一个目标，转 Step4；

Step8: 判断当前任务点是否为编队任务点子集中的最后一个任务点，如果是，输出无人机需求矩阵；否则，选择编队任务点子集中下一个任务点，转 Step3。

无人机需求矩阵得到的值中，除攻击型无人机外都是明确的架数，攻击型无人机数量的为  $ceil$  (编队需携带弹药数/每架攻击型无人机带弹量)，其中  $ceil$  函数为向上取整函数。

假设第  $i$  个编队所需无人机数量为  $Num(T_i)$ ，此时所需的总无人机数量为：

$$NUM = \sum_{i=1}^N Num(T_i) \quad (3.1)$$

航程代价的确认比较简单，对于每个编队而言，其任务点子集中执行时间约束的结束时间最晚的一个任务点，决定了编队在执行任务过程中飞行的距离，假设第  $i$  个编队的航程代价为  $distance(T_i)$ ，则此时总的航程代价为

$$discost = \sum_{i=1}^N distance(T_i) \quad (3.2)$$

总的代价指标计算公式为：

$$C = a_1 * NUM + a_2 * discost \quad (3.3)$$

无人机使用的越多，航路代价越大，总的任务代价指标也越大。 $a_1$  与  $a_2$  为大于零的调节系数，用于调整无人机数量使用和航程代价对代价指标的影响。

编队完成任务得到的收益，首先与其任务点子集内的目标价值有关，还与编队执行任务的成功率有关，对于执行任务点子集  $T_j$  的编队  $j$ ，其执行  $TP_i(TP_i \in T_j)$  中任务的成功率记为

$P_j(TP_i)$ ,  $TP_i$  的价值为  $Val(TP_i)$ , 则此编队执行  $T_j$  中所有任务后的所得收益为:

$$G_j = \sum P_j(TP_i) * Val(TP_i), \forall TP_i \in T_j \quad (3.4)$$

此时总的收益指标计算公式为:

$$G = \sum_{j=1}^N G_j \quad (3.5)$$

最终确定的目标函数为:

$$Z = \max(k_1 * G - k_2 * C) \quad (3.6)$$

任务分配目标就是实现目标函数的尽可能大, 此时收益指标值尽可能高, 代价指标值尽可能低,  $k_1$  与  $k_2$  为大于零的调节系数, 用于调整收益指标值与代价指标值对优化目标的影响, 两者值的变化可以反应当前对收益或是代价的侧重变化。

### 3.3 基于静态链表结构的遗传算法

遗传算法借鉴了生物学中进化论“适者生存”的观点, 最早由美国 Michigan 大学的 Holland 教授提出, 1975 年其出版了专著《Adaptation in Natural and Artificial Systems》标志着遗传算法的诞生<sup>[73]</sup>。

遗传算法的基本流程如下:

Step1: 产生初始种群, 随机产生一定数量的染色体个体, 形成初始种群;

Step2: 适应度计算, 根据所设定的目标函数, 计算每个个体相应的适应度值, 用以判断染色体的好坏;

Step3: 复制, 从当前的种群中挑选优秀的染色体, 将其复制进入下一代个体;

Step4: 交叉, 挑选部分染色体之间进行交叉换位操作, 由此产生一定数量的下一代的新个体;

Step5: 变异, 模拟生物学中的基因突变, 对被选中的个体的染色体某位基因进行随机变化, 增加种群的多样性;

Step6: 新种群产生, 通过上述的复制、交叉、变异操作得到一个新的种群;

Step7: 终止条件判断, 判断当前种群是否满足终止条件, 如果是, 则将此时种群中最优的结果解码输出, 如果否, 返回执行 Step3。

遗传算法中需要确定的运行参数:

(1) 编码: 设计编码方式用以表示所研究问题的解, 即染色体;

(2) 种群数量: 种群中所含个体的数量, 不宜过大或者过小, 一般取 50 至 100;

(3) 交叉概率: 交叉概率影响种群个体间交叉操作执行的可能性, 是产生新个体的主要方式, 一般取 0.4 至 0.9;

(4) 变异概率：变异概率若取得较大，会破坏种群中已有的较好的模式，若取得过小，通过变异产生新个体的能力又会较差，一般选取 0.0001 至 0.01；

(5) 终止进化代数：遗传算法依赖终止进化代数停止运行，并输出种群内的最优解，一般取 100 至 500。

本节主要设计了一种基于静态链表结构的遗传算法，参照静态链表的数据结构设计相应的染色体编码，并且针对染色体的编码方式设计了相应的交叉与变异操作方式，最终进行仿真证明了设计的可行性。首先介绍染色体的编码方式。

### 3.3.1 染色体编码

本章的编码方式参考数据结构<sup>[74]</sup>中的静态链表的结构。由于在部分高级程序设计语言中并不存在指针类型，则无法在其中使用链表结构，为此，专门设计了静态链表。静态链表的具体内容如下：在如下的表 3.1 中，表述了一个数组，数组的一个分量(即一行)表示一个结点，每个分量由索引、数据域和指针域组成，指针域内存放当前分量所指向的下一个分量的索引。数组索引为 0 的分量定义为头结点，代表此静态链表的第一个结点。这样设计的数据结构在执行数据的插入和删除操作时不需要移动元素，只需要对分量的指针域内的值进行相应的修改。例如，向表 3.1 中索引为 4 的“D”元素后插入数据元素“S”时，将指针域由 5 改为 9，同时将索引为 9 的分量的指针域改为 5 即可，同理对于删除数据元素“N”的情况，将指针域指向他的分量“O”的指针域的值改成所需删除的分量的指针域的值即可，表 3.2 中显示了最终结果。

表 3.1 修改前的静态链表

索引	数据域	指针域
0		1
1	U	2
2	A	3
3	V	4
4	D	5
5	R	6
6	O	7
7	N	8
8	E	0
9		

表 3.2 修改后的静态链表

索引	数据域	指针域
0		1
1	U	2
2	A	3
3	V	4
4	D	9
5	R	6
6	O	8
7	N	8
8	E	0
9	S	5

本章所设计的染色体编码中舍弃了表 3.1 的数据域部分,保留了索引部分和指针域部分,索引从 1 开始,每一个索引对应一个任务点,如当索引为 2 时,对应任务点  $TP_2$ 。表 3.3 中为一个染色体实例,其中任务点数量为 6,对应每个任务点的索引值在染色体中从左至右依次排序,指针域内的值除 0 和-1 之外,其余都为 1 到 6 中的值。

现对指针域中的值的情况以及含义进行具体说明,假设当前任务点集合  $T = \{TP_1, TP_2, \dots, TP_n\}$ , 指针域内的值存在三种情况:

- (1)  $1 \sim n$  之间的正整数,用以表述所指向的下一个任务点所对应的索引;
- (2) -1, 表示当前索引对应的任务点已无其他可达任务点,即  $Ctt$  矩阵此行的值全为  $Inf$ ;
- (3) 0, 不再指向其他分量,即当前索引对应的任务点之后不再指向其他任务点的索引。

同时指针域内的值还需满足如下条件:当索引为  $i$  的基因指针域值为  $1 \sim n$  内的正整数  $j$  时,  $Ctt_{ij} \neq Inf$ , 同时染色体指针域中除了 0 和-1 之外,其余数字出现次数不得超过一次。

表 3.3 染色体结构表

索引	1	2	3	4	5	6
指针域	3	4	5	6	0	-1

上表中的染色体,若其可达性矩阵描述如下,其中  $x_i$  表示一个正实数:



$$Czt = \begin{bmatrix} 1 & 1 & 1 & 1 & Inf & Inf \end{bmatrix}, Ctt = \begin{bmatrix} Inf & x_1 & x_2 & x_3 & x_4 & x_5 \\ Inf & Inf & x_6 & x_7 & x_8 & x_9 \\ Inf & Inf & Inf & Inf & x_{10} & x_{11} \\ Inf & Inf & Inf & Inf & x_{12} & x_{13} \\ Inf & Inf & Inf & Inf & Inf & x_{14} \\ Inf & Inf & Inf & Inf & Inf & Inf \end{bmatrix}$$

则可通过如下步骤将其解码成任务集信息:

Step1: 定义一个集合  $TF$ ,  $TF = \emptyset$ , 当前编队为  $i, i = 1$ ;

Step2: 新建任务点子集  $T_i$ , 找出任务点  $TP_j$ , 其满足  $STime(TP_j) = \min(STime(TP_k))$  且  $Czt_j = 1$ , 其中  $TP_k \in T$  且  $TP_k \notin TF$ 。将  $TP_j$  加入集合  $T_i$  以及  $TF$  中;

Step3: 由任务点  $TP_j$  开始, 根据其指针域中的值, 查找到下一个任务点设为  $TP_m$ , 并将其  $TP_m$  加入任务子集  $T_i$  以及集合  $TF$  中。根据  $TP_m$  指针域中的值, 查找到下一个任务点将其加入任务子集  $T_i$  以及集合  $TF$  中, 如此反复, 直到当前任务点指针域内值为 0 或 -1。将当前任务子集  $T_i$  作为编队  $i$  的任务点子集并保存,  $i = i + 1$ ;

Step4: 如果此时  $TF \neq T$ , 转 Step2; 否则, 得到所有的编队的任务点子集。

假设当前  $STime(TP_1) < STime(TP_2) < \dots < STime(TP_6)$ , 则按上述方法可解码得到两个子任务点子集, 分别为  $\{TP_1, TP_3, TP_5\}$  与  $\{TP_2, TP_4, TP_6\}$ 。

### 3.3.2 种群初始化

种群初始化过程中, 将产生一定数量的染色体作为种群, 每个染色体随机产生, 同时符合上述编码规则, 每个染色体都可以译码得到一种任务分配方式。每个染色体产生的过程描述如下:

对于染色体的第  $i (i = 1, 2, \dots, n)$  个基因, 根据  $Ctt$  矩阵找出当前  $TP_i$  是否有可达的任务点, 如果有, 从可达的任务点中选择一个任务点, 如果此任务点没有被选择过(没有出现在染色体的基因中), 则将其作为染色体第  $i$  个基因上的指针域值; 如果此任务点已被选择, 染色体第  $i$  个基因上的指针域值设为 0。如果当前  $TP_i$  无可达任务点, 则将其染色体第  $i$  个基因上的指针域值设为 -1。

重复执行上述过程多次, 即可得到满足种群数量要求的染色体。

### 3.3.3 交叉操作

在交叉过程中, 使用类似于单点交叉方法, 选择染色体的一个基因作为交叉点。亲代染色体的选择通过转轮盘的方法, 随机选择两个产生一对子代染色体。一个亲代染色体被随机选中的概率, 由其适应函数值决定, 染色体越接近目标函数的解, 适应函数值越高。染色体

的交叉点选择应为随机，但是当两个亲代染色体在所选择的交叉点上指针域值相同时，则需要重新产生交叉点位置，以保证交叉操作的有效性。当选中了两个亲代染色体执行交叉操作，并选中了合适的交叉点，交叉的具体过程设计如下：

Step1: 相互交换两个亲代染色体被选中交叉点处的指针域值；

Step2: 调整交换后的两个亲代染色体的合理性，具体过程在 3.3.5 小节中描述；

Step3: 得到交叉后的两个新染色体，结束这两个染色体的交叉过程。

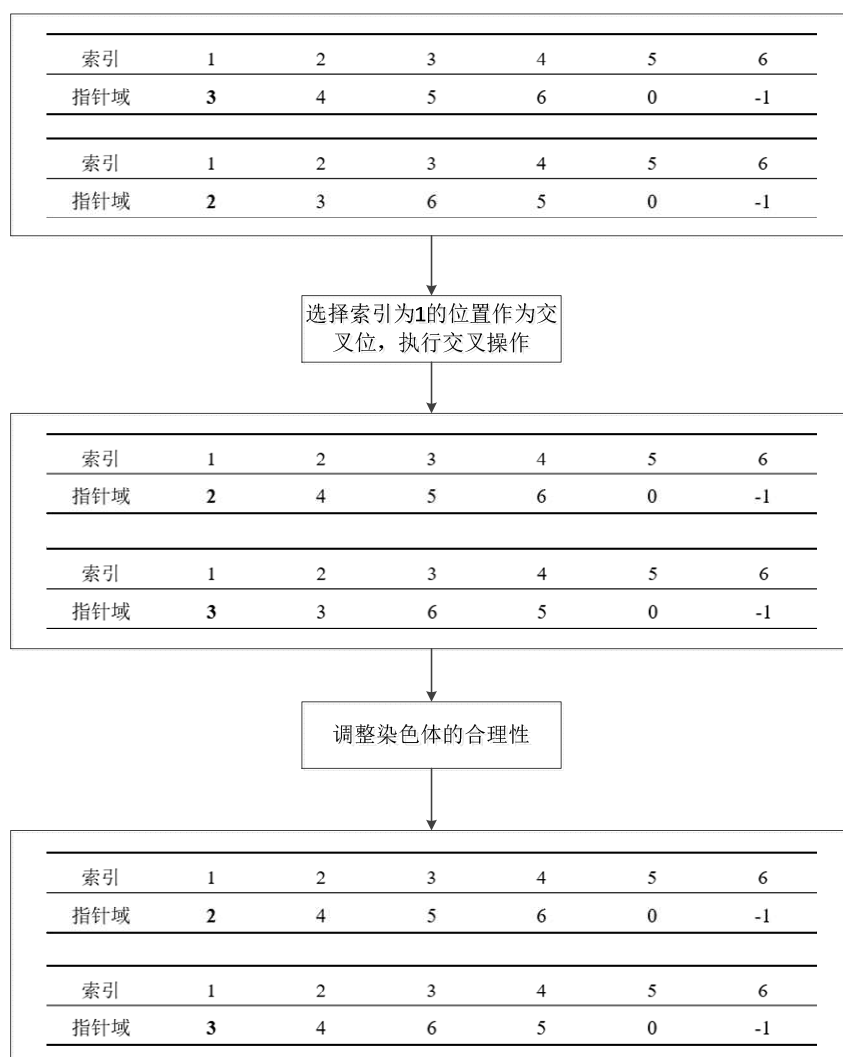


图 3.3 染色体交叉示意图

图 3.3 为一个交叉过程的简单例子，可以发现，执行染色体交叉操作之后，得到了与亲代不相同的子代染色体。

### 3.3.4 变异操作

一个染色体发生变异是随机的，染色体发生变异的基因是也随机选择的，变异操作采取

基本位变异,种群内每个染色体以变异概率  $P_m$  被选中,并随机指定某一位基因进行变异操作。

当一个染色体被选中需要变异时,变异过程描述如下:

Step1: 若当前变异的染色体所有指针域值都为-1,则此种情况下染色体无法产生变异,结束;否则执行下一步;

Step2: 在染色体上随机产生一个变异基因,若当前此基因上指针域值为-1,则其无法变异,重新选择,直到选中一个基因位上指针域值不为-1;

Step3: 对于当前选中的基因上索引值所对应的任务点,从  $Ctt$  矩阵中查找出其可达的任务点,如果其可达任务点只有一个,则若当前此位指针域值不为 0,将其置为 0;若当前此位指针域值为 0,将其置为唯一可达任务点。如果其可达任务点不止一个,则随机选择一个可达任务点,并保证与先前基因位上的指针域值不相同;

Step4: 调整染色体使其满足合理性;

Step5: 结束此染色体的变异操作。

### 3.3.5 调整染色体使其满足合理性的过程

当染色体经过交叉或者变异可能会产生无法解码的情况,主要原因是在交叉和变异的过程中,染色体的基因上可能会产生相同的指针域值(非-1与0)。如表 3.4 所示的染色体是由表 3.3 中的染色体变异第二位基因而来,其指针域内的值由原先的 4 变为 3,此种情况下,此染色体本身是一条“错误”的染色体,无法解码,需要进行相应的调整。

表 3.4 无法解码的染色体结构

索引	1	2	3	4	5	6
指针域	3	3	5	6	0	-1

调整过程描述如下:对于任何一个经过交叉或者变异的染色体,判断其是否需要进行合理性调整。若其为经过交叉的染色体,针对其未交叉的部分的每一位基因上不为 0 和-1 的指针域值,如果它的指针域值与染色体内交叉部分基因的指针域值存在相同,则查找此基因索引值对应的任务点的所有可达任务点,如果其中存在尚未在染色体的其他基因的指针域值上出现的任务点,将此基因指针域值调整为此任务点;如果可达任务点中不存在未出现的,将基因指针域值调整为 0;

若其为经过变异的染色体,针对其未变异的部分的每一位基因上不为 0 和-1 的指针域值,如果它的指针域值与染色体内变异的部分基因的指针域值存在相同,则查找此基因索引值对应的任务点的所有可达任务点,如果其中存在尚未在染色体的其他基因的指针域值上出现的任务点,将此基因指针域值调整为此任务点;如果可达任务点中不存在未出现的,将基因指针域值调整为 0。

以表 3.4 中经过变异的染色体为例, 第二位基因为经过变异的基因。从左至右检查每一位基因, 发现第一位基因的指针域值与变异部分指针域值冲突, 则查找出任务点  $TP_1$  的所有可达任务点  $\{TP_2, TP_3, TP_4, TP_5, TP_6\}$ , 其中  $TP_2, TP_4$  尚未出现, 则有以下表 3.5 与表 3.6 两种调整方式。

表 3.5 调整后的染色体结构 1

索引	1	2	3	4	5	6
指针域	2	3	5	6	0	-1

表 3.6 调整后的染色体结构 2

索引	1	2	3	4	5	6
指针域	4	3	5	6	0	-1

### 3.4 仿真算例

现运用 MATLAB R2014a 进行相关程序设计, 进行仿真, 以验证上述的遗传算法在任务分配中的可行性。我方各类无人机的速度、航程、资源约束等信息如表 3.7 中所示,

表 3.7 我方无人机相关信息表

无人机类型	速度(Km/h)	航程(Km)	资源限制	数量(架)
侦察探测类	150	5500	无	充足
电子战类	150	5500	两架才可定位	充足
攻击类	150	4000	每架 4 枚导弹	充足
毁伤评估类	150	2200	无	充足

表 3.8 任务点坐标与时间约束表

任务点编号	平面 X 坐标(Km)	平面 Y 坐标(Km)	执行时间约束(min)
$TP_1$	50	50	[30,40]
$TP_2$	50	150	[110,120]
$TP_3$	150	50	[110,120]
$TP_4$	120	350	[300,320]
$TP_5$	300	350	[240,260]
$TP_6$	300	200	[240,250]
$TP_7$	100	250	[190,200]
$TP_8$	400	450	[450,470]
$TP_9$	400	100	[340,360]

我方所需要执行的 9 个任务点的坐标信息与执行时间约束如表 3.8 所示, 各任务点位置

为以我方基地为坐标轴原点的坐标系下的位置,任务点执行时间约束为以无人机起飞时间为基准的相对时间。

表 3.9 中为各任务点内部的敌方目标的信息表,每个任务点内的目标都有特定的编号,并且通过其雷达探测范围与武器打击范围可以确定其类型,同类型目标其探测与打击能力也不尽相同。

表 3.9 任务点内目标信息表

任务点编号	目标编号	雷达探测范围(Km)	武器打击范围(Km)
$TP_1$	$TP_1(1)$	0	0
$TP_2$	$TP_2(1)$	20	15
$TP_3$	$TP_3(1)$	15	15
	$TP_3(2)$	0	0
	$TP_3(3)$	0	0
$TP_4$	$TP_4(1)$	30	20
$TP_5$	$TP_5(1)$	0	0
$TP_6$	$TP_6(1)$	0	0
$TP_7$	$TP_7(1)$	30	30
$TP_8$	$TP_8(1)$	30	0
	$TP_8(2)$	20	0
$TP_9$	$TP_9(1)$	30	0
	$TP_9(2)$	20	20

上述的任务点的平面投影如图 3.4 所示,每个任务点位置在图中已标识,由于任务点内的目标可能存在雷达探测与武器攻击的威胁,因此,在任务点上表示出两种威胁的最大范围。

地图中存在着地形与气象等静态威胁的存在,其威胁区域的平面投影也为圆形,其中心坐标与威胁范围如表 3.10,在图 3.4 中用同心圆标识。

表 3.10 地形与气象威胁信息

地图威胁编号	平面 X 坐标(Km)	平面 Y 坐标(Km)	威胁范围(Km)
1	65	210	20
2	130	80	18
3	270	315	25

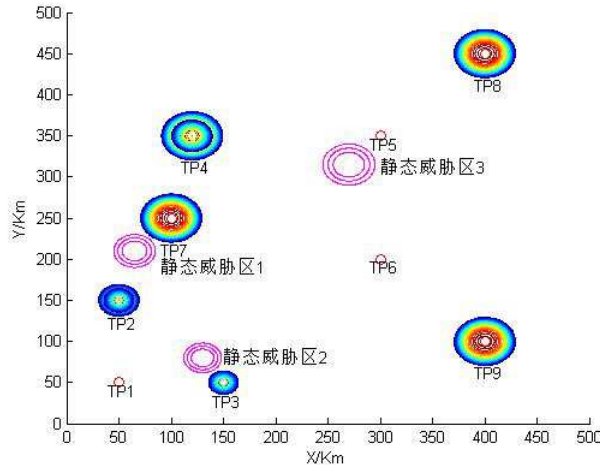


图 3.4 任务点与威胁区域平面投影示意图

根据上述信息获取如下可达性矩阵用以进行染色体的编码与解码

$$C_{zt} = [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]$$

$$C_{tt} = \begin{bmatrix} Inf & 100 & 100 & 313.24 & 397.51 & 291.55 & 206.16 & 533.05 & 353.79 \\ Inf & Inf & Inf & 224.26 & Inf & 254.95 & 113.80 & 462.40 & 353.55 \\ Inf & Inf & Inf & 301.50 & Inf & 212.13 & Inf & 471.70 & 254.95 \\ Inf & Inf & Inf & Inf & Inf & Inf & Inf & 297.32 & Inf \\ Inf & Inf & Inf & Inf & Inf & Inf & Inf & 141.42 & Inf \\ Inf & Inf & Inf & Inf & Inf & Inf & Inf & 269.26 & 141.42 \\ Inf & Inf & Inf & 101.98 & Inf & Inf & Inf & 360.56 & 335.41 \\ Inf & Inf & Inf & Inf & Inf & Inf & Inf & Inf & Inf \\ Inf & Inf & Inf & Inf & Inf & Inf & Inf & Inf & Inf \end{bmatrix}$$

遗传算法种群数量选择 100，交叉概率 0.5，变异概率 0.015，终止迭代代数 200 代，目标函数  $Z$  中调节参数  $a_1$ ， $a_2$ ， $k_1$ ， $k_2$ ，使得  $a_1$  与  $k_2$  的值较大，此时无人机使用数量对目标函数的影响最大，无人机使用越少，目标函数越优。分配结果如表 3.11 所示。

表 3.11 任务分配结果表

编队	分配任务点
$F_1$	$\{TP_1, TP_5\}$
$F_2$	$\{TP_2, TP_4, TP_7, TP_8\}$
$F_3$	$\{TP_3, TP_6, TP_9\}$

此时无人机的使用数量为：侦察探测无人机 2 架，电子战无人机 4 架，攻击无人机 5 架，毁伤评估无人机 3 架，一共 14 架无人机参与任务。

### 3.5 本章小结

本章在第二章中所建立的无人机协同任务规划系统模型以及可达性矩阵等的基础上,提出了一种基于静态链表结构编码的遗传算法,确定了由所获收益与付出代价相综合的目标函数,设计了染色体的交叉、变异过程,用以解决无人机编队的任务预分配问题,最后通过仿真实验,证明了方法的可行性。

## 第四章 无人机航路规划技术

### 4.1 引言

在任务分配阶段,为了能够得到任务点之间的可达性关系,对航路进行了预估,而未进行航路的规划。航路规划作为整个无人机任务规划仿真系统的一部分,主要目的是为了根据当前任务分配结果、战场信息、我方无人机的性能等信息,为无人机编队规划出航路,有效的提高无人机的作战效能与生存能力。

正如绪论中所言,目前的航路规划有着多样的方法选择,这些方法各有优缺点,并不存在一种完美的算法可以适用于各种情景下。 $A^*$ 算法作为目前路径规划中相对成熟与成功的算法,有着运算速度快、方法简洁稳定、搜索结果较优等特点,其应用也相对其他规划算法更加广泛。文献[46]中,将  $A^*$ 算法运用到移动机器人的避障问题中,文献[75]中,研究了  $A^*$ 算法在游戏地图中路径搜索方面的应用,文献[76]中,研究了在地理信息系统(Geographic Information System ,GIS)中  $A^*$ 算法的应用。在这些应用研究中,文献的作者都或多或少的进行了  $A^*$ 算法的改进,改进的思路主要有以下两个方向:(1) 提高规划效果,(2) 提升算法速度。规划效果提升方面,文献[77]考虑  $A^*$ 算法产生的路径通常为次优,将  $A^*$ 算法搜索方向的个数由 8 个扩展到 24 个,以获得更优的规划路径。文献[78]中,更是将搜索方向的个数由 8 个扩展为无限个,使算法可以朝任意方向进行搜索,并在真实环境下验证的改进的效果。算法速度降低方面的研究更加的丰富,文献[75]提出用一种二叉堆的数据结构来存储 OPEN 表,有效的提高了算法的搜索速度,尤其在搜索区域较大时,有着明显的效果提升。文献[49]同样是针对 OPEN 表的查找速度问题,提出了一种新的 Cheap 表,用于提升算法速度。

本文航路规划不仅用于任务预先分配中,也将在重分配过程中航路重规划时使用,需要具有计算快速、算法易于实现同时可以保证全局最优解的收敛性的特点。因此,选用  $A^*$ 算法进行编队的航路规划,并提出了改进方法进一步提升了  $A^*$ 算法的计算速度,通过仿真证明了改进的合理性。

### 4.2 航路规划研究对象

本章主要研究各编队前往执行任务的过程中的航路规划。如图 4.1 所示,整个系统的航路规划是由系统中每个编队的航路所组成,而每个编队的航路规划可以具体再细分为各编队任务点至任务点的航路规划。因此,只要规划出各编队任务点之间的航路,再将航路组合,就可以完成整个航路规划的过程。本文中由于各类型无人机除载荷不同外,其他属性都一致,



假定执行任务过程中无人机编队内各机飞行高度一致且不变，一定程度上将问题简化为一个二维航路规划。

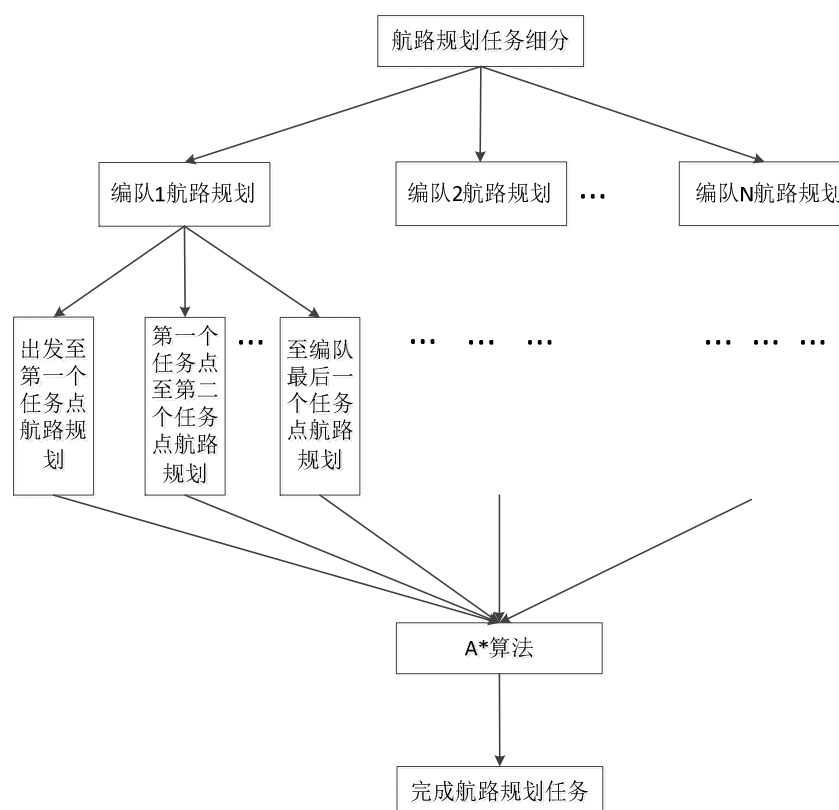


图 4.1 航路规划细分过程示意图

## 4.3 A\*算法

### 4.3.1 算法原理

A\*(A-Star)算法是一种常见的最短路径搜索方法，目前已被大量运用在工程实践和路径规划类软件中。A\*算法评估公式为：

$$f(n) = h(n) + g(n) \quad (4.1)$$

其中  $f(n)$  是从起始点抵达目标点并且经节点  $n$  时的估计代价函数；

$g(n)$  是从起始点到节点  $n$  的实际代价值；

$h(n)$  是从节点  $n$  出发抵达目标点的预估代价值，又称启发函数。

在二维地图中，每个节点有可能有 8 个相邻节点，当从初始点向目标点移动时，算法将权衡  $g(n)$  和  $h(n)$ 。每次进行主循环时，它检查  $f(n)$  最小的节点  $n$ ，并将其作为扩展的后继节点。如此往复，直至搜寻到目标节点为止。

算法具体步骤如下<sup>[75]</sup>：

Step1: 新建 *OpenList* 与 *ClosedList*，确定  $S$  为起始点， $E$  为目标点，打开 *OpenList*，

关闭 *ClosedList* ;

Step2: 将起始节点 *S* 加入 *OpenList* 中, 查看与起始节点 *S* 相邻的网格点, 把其中可到达的网格点加入 *OpenList*, 不可到达的, 如地图中需要规避的障碍, 不加入 *OpenList*。从 *OpenList* 中移除 *S*, 并将 *S* 放入 *ClosedList* 中, *ClosedList* 中的节点都是不在需要关注的;

Step3: 当前如果 *OpenList* 为空, 则退出算法;

Step4: 如果 *OpenList* 不为空, 从 *OpenList* 中选取 *f* 值最小的节点 *n*;

Step5: 将节点 *n* 从 *OpenList* 中移除, 将其加入 *ClosedList* 中;

Step6: 判断节点 *n* 是否是目标节点 *E*。如果是, 则找到最优路径; 如果不是, 就转到下一步;

Step7: 对节点 *n* 进行子结点扩展, 并对所有子结点执行下述步骤, 设一个子结点为 *m*;

(1) 如果 *OpenList* 中和 *ClosedList* 中均不存在节点 *m*, 就将 *m* 加入 *OpenList*。并给 *m* 赋予一个指向它的父节点 *n* 的指针, 航路生成时可根据这一指针回溯, 找到一条完整的路径。

(2) 如果 *OpenList* 中或者 *ClosedList* 中已有节点 *m*, 将节点 *m* 的 *f* 新值与其 *f* 旧值进行对比。如果此时的新值小于旧值, 以新值替换旧值, 并修改 *m* 的父指针, 使其指向节点 *n*, 如果此时 *m* 在 *ClosedList* 中, 则将其从 *ClosedList* 中放入 *OpenList*。

Step8: 跳转至 Step3, 进入下一轮路径寻找。

#### 4.3.2 启发函数

启发函数  $h(n)$  的选择是影响 A\*算法效果的一个重要函数, 对于估计代价函数  $f(n)$  而言,  $g(n)$  是一个已知的值, 此时  $f(n)$  直接受到  $h(n)$  的影响。启发函数  $h(n)$  对算法的影响具体总结如下:

(1) 若  $h(n)$  值小于节点 *n* 到达目标节点的实际距离, 此时算法一定可以找到最优解, 当  $h(n)$  值相对实际距离越小, 算法的运行速度越慢;

(2) 若  $h(n)$  值等于节点 *n* 到达目标节点的实际距离, 在这种最理想的情况下, 也一定可以找到最优解, 同时, 算法效率最高;

(3) 若  $h(n)$  值大于节点 *n* 到达目标节点的实际距离, 此时算法效率也相对较高, 但是并不能一定找到最优解。

A\*算法的启发函数一般来说有三种选择, 其一是曼哈顿距离, 其二是对角线距离, 其三是欧氏距离。曼哈顿距离为两个点在直角坐标下向两个坐标轴投影的距离之和, 又被称为出租车距离。

如图 4.2 所示, 点 *n* 到点 *E* 的曼哈顿距离为:

$$h_m(n) = (|x(n) - x(E)| + |y(n) - y(E)|) \quad (4.2)$$

本章所设定的场景中允许进行对角线移动，对角线距离定义为：

$$h\_diag(n) = \sqrt{2} * \min(|x(n) - x(E)|, |y(n) - y(E)|) \quad (4.3)$$

现将曼哈顿距离与对角线距离相结合，可以得到一种新的启发函数，即：

$$h\_md(n) = h\_diag(n) + (h\_m(n) - \sqrt{2} * h\_diag(n)) \quad (4.4)$$

此时  $n$  到目标点  $E$  之间的启发函数  $h\_md(n)$  的值在图 4.2 中被表示出，可以发现，此时的启发函数值大于使用对角线距离情况下的值，小于使用曼哈顿距离情况下的值，更加接近真实的距离。

欧氏距离，也称为欧几里德距离，欧式距离表示当前点  $n$  到目标点  $E$  之间的直线距离，经常使用于允许进行对角线移动的场景中：

$$h\_o(n) = \sqrt{(x(n) - x(E))^2 + (y(n) - y(E))^2} \quad (4.5)$$

在二维坐标系中，A\*算法所产生的路径不可能严格按照欧式的直线距离行进，也就是实际距离可能比启发函数  $h\_o(n)$  大，这种情况下可以保证搜索到最短路径。

以上四种启发函数的示意图如下图 4.2 所示：

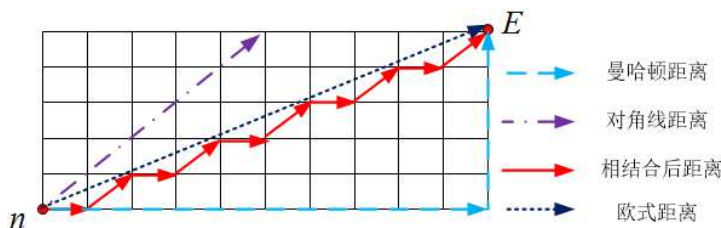


图 4.2 多种启发函数效果示意图

#### 4.4 A\*算法的改进

上述经典的 A\*算法的精度与速度在实际的应用中不是非常理想，有较大的改进空间。本章通过改进 *ClosedList* 的查找方式对 A\*算法进行改进，以提升其运算速度。

时间复杂度是描述算法运行时间的一个概念，当一个算法中一段操作需要重复执行，并且其重复的次数与问题大小  $n$  有着函数关系  $F(n)$ ，则算法的运行时间直接与  $n$  相关，其增长率与  $F(n)$  的增长率相同，记为  $O(F(n))$ 。

在 A\*算法的规划中，需要将整个任务地图空间进行栅格化处理，实现寻路地图的离散化。地图栅格需要选取合适的尺寸，一般而言，尺寸应该选取的比较小，但如果选取尺寸过小，虽然可以提升结果的精度，但是算法需要搜索更多的点，导致运算速度下降，算法的实时性降低，因此需要选取合适的地图栅格化尺寸。在完成地图的栅格化之后，将得到一个地图矩阵，对于地图中的障碍区赋予值 1，可自由通行区赋值 0，在此基础上可以进行 A\*算法的规划。

本章引言中提到的大部分文献通过改进启发函数来提升算法速度，算法速度也可以从

*ClosedList* 入手考虑。从上文的 A\* 算法原理中可以看出，每搜寻一个节点时都要遍历一遍 *ClosedList*，而且 *ClosedList* 的规模是随着算法运行不断增加的。因此，如何有效地加快 *ClosedList* 中的节点查找速度是解决算法效率问题的一个十分重要的手段。

在 4.3 节中详细介绍的 A\* 寻找路径的算法流程中，可以发现，*ClosedList* 被多次用于：

- (1) 查找某个点是否在其中；
- (2) 将某个点加入 *ClosedList*；
- (3) 将某个点从 *ClosedList* 中取出。

其中最耗时的为(1)和(3)，这两个过程中都涉及 *ClosedList* 中点的查找，设当前 *ClosedList* 有  $N$  个点，则查找一次点的时间复杂度为  $O(N)$ 。

现设计一种新的地图矩阵内元素类型，建立如下结构体，命名为 *point*：

```
struct point{
    bool InClosedList;
    bool IsBlock;
};
```

分别介绍 *point* 结构体内部的两个数据成员 *InClosedList* 以及 *IsBlock*。*IsBlock* 与原先的地图矩阵中元素的作用相同，用于标示当前地图栅格中此处是障碍区还是通行区，为障碍区时，*IsBlock* 值为 1，为通行区设为 0。*InClosedList* 是在原地图矩阵元素基础上扩展的一个数据成员，用于标示当前所指向的地图矩阵元素是否在 *ClosedList* 中，如果在则值为 1，如果不在则值为 0。

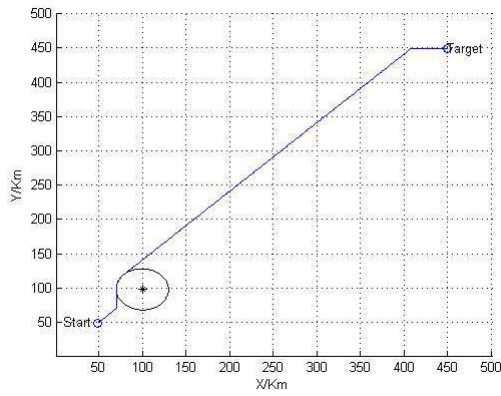
通过此处的改进，判断某个点是否在 *ClosedList* 中，只需要查看此点在地图矩阵中对应的 *InClosedList* 值是否为 1 即可，此时在 *ClosedList* 中查找一次点的时间复杂度由  $O(N)$  降至  $O(1)$ ，可以有效提高 A\* 算法的运行速度。

## 4.5 仿真算例

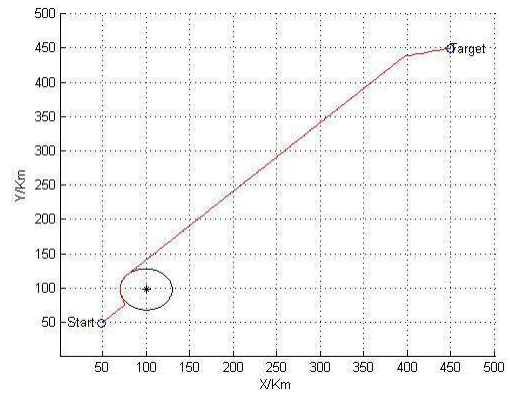
本节对不同启发函数在实际程序中的表现进行仿真，以选取适合的启发函数用于不同的场景需求。同时对 *ClosedList* 的改进效果进行仿真，证明改进对于 A\* 算法速度带来的提升。程序运行平台为 win10 系统 MATLAB R2014a，i5 酷睿 CPU，主频 3.2GHz。本小节的仿真中威胁区域半径统一为 30Km，起始点坐标 (50,50)，终点坐标 (450,450)，将 500\*500Km 的地图离散尺寸设为 1Km。

### 4.5.1 不同启发函数的影响

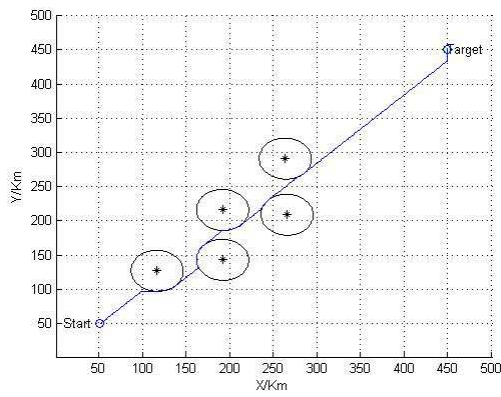
本节在比较启发函数时，主要选取了曼哈顿距离与对角线距离相结合的启发函数  $h\_md(n)$  与欧氏距离启发函数  $h\_o(n)$  相比较，图 4.3 中左侧为  $h\_md(n)$  规划效果图，右侧为  $h\_o(n)$  规划效果图。



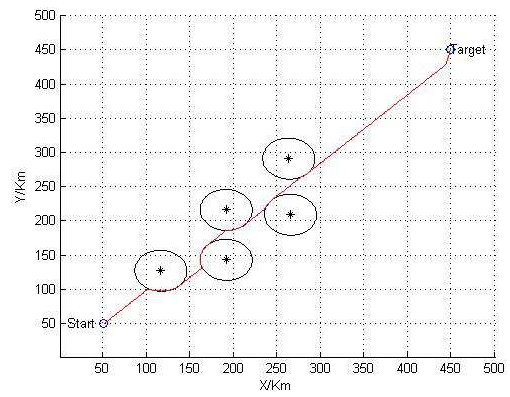
(a)  $h_{md}(n)$  在 1 个威胁区域下效果



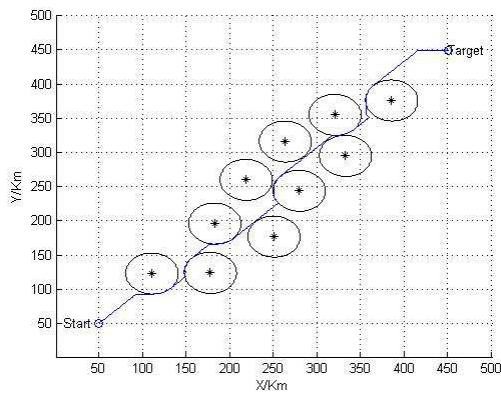
(b)  $h_o(n)$  在 1 个威胁区域下效果



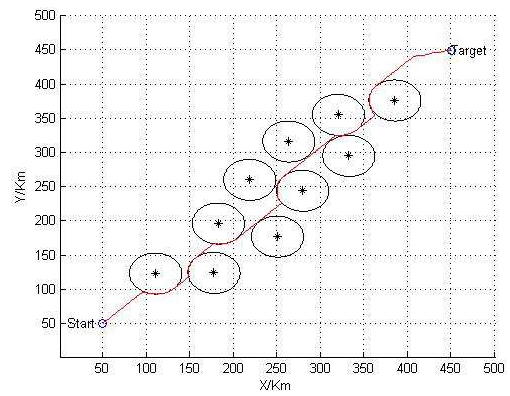
(c)  $h_{md}(n)$  在 5 个威胁区域下效果



(d)  $h_o(n)$  在 5 个威胁区域下效果



(e)  $h_{md}(n)$  在 10 个威胁区域下效果



(f)  $h_o(n)$  在 10 个威胁区域下效果

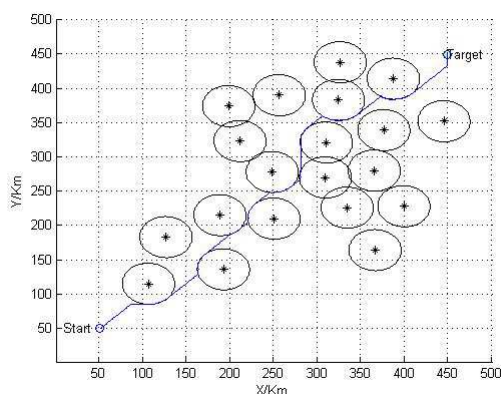
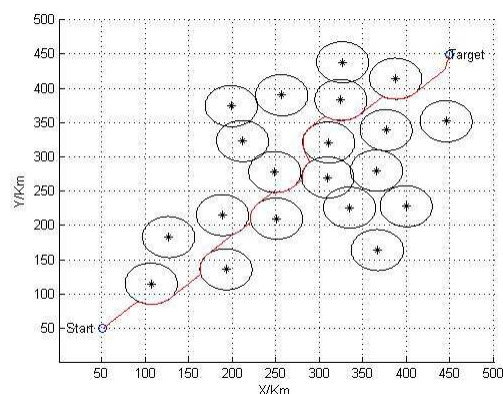
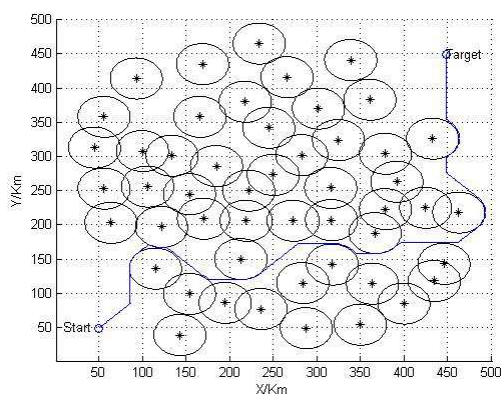
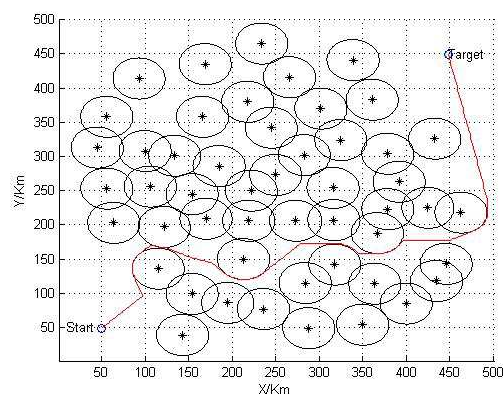
(g)  $h\_md(n)$  在 20 个威胁区域下效果(h)  $h\_o(n)$  在 20 个威胁区域下效果(i)  $h\_md(n)$  在 50 个威胁区域下效果(j)  $h\_o(n)$  在 50 个威胁区域下效果

图 4.3 不同启发函数下 A\*算法效果图

对照上图, 表 4.1 中分别为  $h\_o(n)$  与  $h\_md(n)$  分别在威胁区域个数为 1、5、10、20、50 的情况下所规划的路径的路程与规划时间的对比。可以发现,  $h\_md(n)$  所规划的路径路程相对  $h\_o(n)$  所规划的路程较长, 同时由于  $h\_md(n)$  公式的计算较  $h\_o(n)$  复杂, 因此在计算过程中相对更加耗时, 因此本文采用欧氏距离作为航路规划的启发函数。

表 4.1  $h\_o(n)$  与  $h\_md(n)$  在不同威胁区域数下对比表

威胁数	$h\_md(n)$ 路程(Km)	$h\_o(n)$ 路程(Km)	$h\_md(n)$ 时间(s)	$h\_o(n)$ 时间(s)
1	595.8448	589.7027	2.158801	2.083916
5	599.5311	594.3890	2.841581	2.404212
10	627.8915	626.7494	9.688085	5.809275
20	653.1341	648.1636	28.764816	21.499149
50	880.4062	874.1219	153.584944	132.760876



### 4.5.2 ClosedList 改进结果

为了验证所提出的 *ClosedList* 查找速度的改进, 现进行仿真对比。启发函数统一选取为  $h_o(n)$ , 同样在上一节的地图与威胁区域基础上, 分别对威胁区域个数为 1、5、10、20、50 的情况进行改进前与改进后的仿真, 改进前后由于搜索算法相同, 因此所产生的规划效果完全相同, 运行时间的差别如表 4.2 所示。

表 4.2 改进前后运行时间对比表

威胁数	改进前运行时间(s)	改进后运行时间(s)
1	2.083916	1.944084
5	2.404212	2.179431
10	5.809275	3.956052
20	21.499149	11.407443
50	132.760876	48.189557

通过对比发现, 威胁区域数量较少时, 算法运行速度的提升在 10% 以内, 随着威胁区域个数的增加, 改进后算法运行速度的提升愈加明显, 尤其是当几乎整个地图都充满了威胁区域时, 运行速度将提升 2 倍以上。

### 4.5.3 改进 A\*算法用于多编队航路规划

利用改进的 A\*对第三章 3.4 节中的任务分配结果进行多编队航路规划, 所选用的启发函数为欧式距离启发函数。规划的效果如图 4.4 所示, 图中任务点、任务点产生的动态威胁区域、静态威胁都已标出, 编队  $F_1$ 、 $F_2$ 、 $F_3$  的航路在图中已进行规划, 可以发现规划的航路有效的避开了威胁区域, 同时航路的距离也较短。

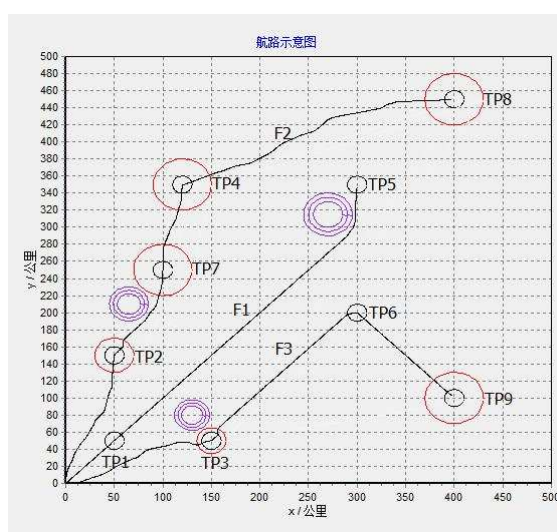


图 4.4 多编队航路规划效果

## 4.6 本章小结

本章首先分析了本文所设计的协同任务规划系统中，航路规划部分所需要完成的任务，并将整个航路规划过程进行细分，提出了不同使用情况下对航路规划算法的不同要求。阐述了 A\*算法在路径规划领域的特点与地位，将 A\*算法使用于无人机的航路规划中。对于不同的启发函数进行合适的选择，针对 A\*算法速度提升，提出了改进 *ClosedList* 的查找方式，通过仿真证明了改进的有效性，最后将改进后 A\*算法使用在多编队的航路规划中。



## 第五章 无人机任务重分配技术

### 5.1 引言

信息是现代化的战场中最重要的因素,战前侦察与及时情报获取可以帮助我方知己知彼,打有准备之仗,而战场上想获得敌方完整的信息绝非易事,在现代化战场环境下,常用侦查手段进行战前侦察已经很难确保所获得的敌方情报的完整性与正确性,仅仅依靠战前的侦察信息进行任务规划也是不明智的。面对这种情况,需要提高规划系统应对战前信息与实时战场信息相冲突的应变能力,能够及时处理新任务加入、事先未侦察到的威胁区域、敌方的虚假目标等等情况的出现。这些功能的具体实现都需要在任务重分配中实现。

与任务预分配不同的是,任务的重分配侧重实时性,要求能够立刻解决当前的突发情况,因此所得到的解更倾向于“可行”而非“最优”。任务重分配过程应满足以下三个原则<sup>[40]</sup>:

- (1) 相对于预规划,重分配更强调实时性,因此相应的算法需要能在短时间内完成重分配,重分配过程中,依旧以原先的规划方案为基准,在原方案基础上进行一些局部调整。
- (2) 新增任务或者任务集内的某些任务需要进行重分配时,若存在多个满足接收条件的编队,应由执行代价最小的编队接收此任务。
- (3) 若可执行待重分配任务的编队中,任务执行代价相同,则应该优先将待重分配任务重分配给当前任务较少的编队执行,使得编队之间的任务数量均衡。

### 5.2 基于合同网协议的任务重分配

#### 5.2.1 合同网协议

Reid G.Smith 于 1980 年在其论文“合同网络协议:高层次、分布式的通信和控制问题解决者”中首次提出合同网协议,并将其使用在分布式系统通信问题的解决中<sup>[79]</sup>。近年来合同网协议理论经过不停的发展改进,已运用到车间调度、无人机任务分配、火力分配等多个领域。算法的具体过程大致描述如下:

**Step1:** 任务公布,当系统中一个个体产生一个任务时,该个体作为任务发布者,通过任务声明将任务的相关信息公布给系统中的其他节点;

**Step2:** 招标,系统中个体在受到任务的信息后,将自身情况与任务类型进行匹配,如果具有执行能力,则向任务发布者进行任务投标;

**Step3:** 中标处理,任务发布者对投标进行综合评价并排序,选择其中最令自己“满意”的标书作为中标对象,并通知所有个体中标情况;

Step4: 合同执行, 成功中标的个体将任务加入自己的任务队列中, 按新的任务队列继续执行任务。

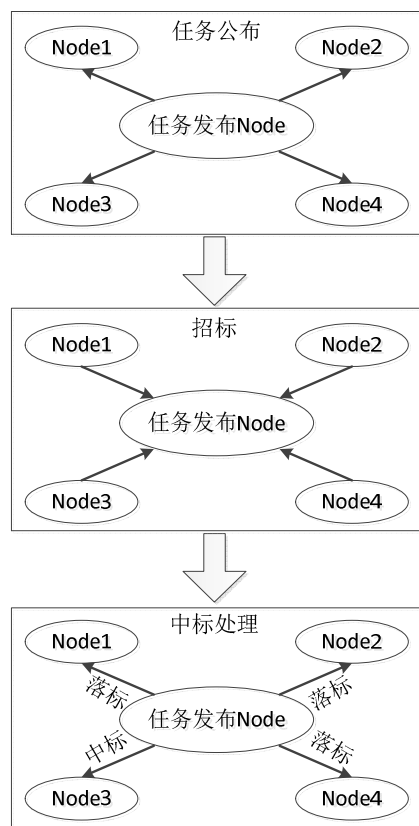


图 5.1 基于合同网的算法流程图

图 5.1 给出了网络中 5 个个体 *Node* 之间进行一次基于合同网的交易过程。

### 5.2.2 基于改进合同网算法的编队任务重分配

#### (1) 合同网协议的改进

传统的合同网模型允许接到招标通知的所有个体参与投标, 但是最终只会有一个优胜者中标, 过多的标书会增加发布者的决策负担, 浪费系统资源。因此, 在个体进行投标时, 与自己相邻的投标个体进行标书对比, 如果自己出价较低, 则主动退出招标, 这样的初步筛选会一定程度上减少发布者中标处理过程中的负担。

#### (2) 任务重分配设计

无人机的任务重分配考虑到战场中的典型突发情况, 这些情况需要事先从以往战场经验中总结得到。本章总结的重分配情况主要为下文中的五种。

##### 1) 任务点增加

突然出现的未被事先侦察到的任务点。新增任务点信息由某一编队侦察到或者由指挥中心提出, 并由指挥中心向各个编队统一发布信息, 各编队将执行如下步骤:

Step1: 接收由指挥系统发出的新增任务点的信息;

Step2: 根据当前编队内无人机的组成情况, 判断编队是否有执行任务点上任务的能力, 如果有, 执行 Step3, 否则不发起投标;

Step3: 判断此任务点加入编队的任务点子集后, 新的任务集是否是可执行的, 如果是, 执行 Step4, 否则不发起投标;

Step4: 与自己相邻的投标个体进行标书对比, 获胜后发起投标。指挥系统在收到各编队的投标信息后, 选择执行代价最小的编队作为中标对象。

## 2) 任务点取消

由于事先情报错误、敌方虚假目标等情况, 有些任务点在任务执行过程中不需要被执行, 即被取消, 不在此类目标上浪费我方的编队的火力与资源。任务取消是一种相对简单的情况, 各编队将执行如下过程:

Step1: 接收由指挥系统发出的取消任务的信息;

Step2: 如果被取消的任务点存在于当前编队任务点子集中, 执行 Step3, 否则继续执行任务;

Step3: 如果此任务点的任务正在被执行或者尚未被该编队执行, 则放弃此任务点, 将其从编队任务点子集中剔除, 并继续执行其他任务。

## 3) 新增威胁区域

编队执行任务的过程中, 可能会发现一些未被事先侦察到的威胁区域, 导致一些编队原本的航路受到阻碍, 这样, 编队按原计划继续执行任务的代价陡增, 此时需要进行适当的任务调整, 降低执行任务的代价。新增的威胁区域将作为一个静态威胁存在于地图中, 将影响到编队预规划好的航路, 部分编队会因为航程的增加无法及时执行一些任务点的任务, 这些任务点将交给指挥系统进行重分配处理, 重分配过程与新增任务点过程类似。

## 4) 无人机战损

编队中的无人机在执行任务的过程中, 可能因为被敌击落、自身故障等因素, 无法继续作为编队的一员执行接下来的任务。此时对于任务点子集中仍未完成任务的任务点而言, 有一部分需要编队中有此无人机才可以被执行, 对于这样的任务点需要交由指挥系统重分配, 指挥系统将执行如下步骤:

Step1: 判断基地内是否有同类别无人机补充, 如果有, 执行 Step2; 否则执行 Step3;

Step2: 判断补充无人机立刻起飞是否能及时追上编队, 如果能, 则对编队进行无人机的补充, 使其可以继续执行原计划的任务, 结束判断; 否则, 执行 Step3;

Step3: 找出该编队中因为缺损此架无人机而无法执行的任务点, 对其进行重分配处理, 每个任务点的被重分配过程与任务点增加的过程类似。

### 5) 人为任务点调整

在任务预分配后, 执行任务的过程中, 考虑进行人为任务点调整, 临时指派给特定的无人机编队去执行特定的任务点, 此时临时调整的任务点优先级别较高, 需要接受临时任务点的编队及时进行相应的任务点调整, 编队的预规划任务点子集中可能会存在与临时调整的任务点之间执行时间的冲突, 即出现并行关系, 此时将触发无人机任务的重分配, 将预规划任务点子集中的冲突任务点进行重分配。对于在人为调整任务点中被指派新任务点的编队而言, 将执行以下步骤:

**Step1:** 如果被指派的编队内无人机类型不满足所指派任务点内目标的类型约束要求, 则编队无法执行任务点, 向指挥系统提示编队类型不满足指派任务要求, 并继续执行原先任务, 否则执行 Step2;

**Step2:** 如果编队立即从当前位置出发或者执行完当前正在执行的任务后立刻去执行被指派的任务点, 无法在任务点的开始执行时间之前到达, 则向指挥系统提示编队无法及时执行指派任务点, 并继续执行原先任务, 否则执行 Step3;

**Step3:** 被指派的任务点加入编队任务点子集后, 编队需判断从当前仍未执行的任务点子集中, 最多执行完哪个任务点后可以及时执行被指派的任务点, 以及执行完指派的任务点后, 可以继续去完成尚未完成的哪些任务点。在这个过程中, 编队任务点子集中可能存在为了执行所指派的任务点而被舍弃的一些任务点, 将这些任务点交由指挥系统进行重分配处理, 重分配过程与新增任务点过程类似。

整个重分配的模型图如图 5.2 所示, 当系统触发了重分配后, 根据当前的重分配情况, 需要判断是否有任务点需要被重分配, 如果有, 则将此任务点发布, 否则继续执行任务。

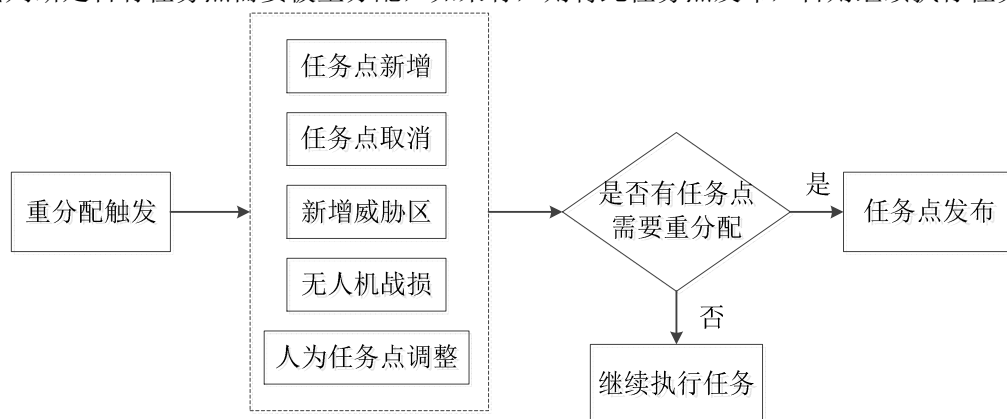


图 5.2 任务重分配模型图

## 5.3 仿真算例

所假设的场景如图 5.3 所示, 在所假设的场景中, 我方一共有 4 个编队  $F_1$ ,  $F_2$ ,  $F_3$ ,  $F_4$ , 有 8 个任务点以及两个威胁区域。各编队处于前往执行任务的途中, 图中的三角代表各编队

此时所在的位置,同心圆表示威胁区域,任务点所在位置在图中也用圆圈指示出。假设编队 $F_1$ ,  $F_2$ 在类型约束上可执行任务点 $TP_1$ 、 $TP_2$ 、 $TP_5$ 、 $TP_6$ ,编队 $F_3$ ,  $F_4$ 在类型约束上可执行任务点 $TP_3$ 、 $TP_4$ 、 $TP_7$ 、 $TP_8$ ,经过初步任务分配后 $F_1$ 执行 $\{TP_1, TP_5\}$ ,  $F_2$ 执行 $\{TP_2, TP_6\}$ ,  $F_3$ 执行 $\{TP_3, TP_7\}$ ,  $F_4$ 执行 $\{TP_4, TP_8\}$ 。

任务点增加:图 5.4(a)中新增了任务点 $TP_9$ ,当前编队 $F_1$ ,  $F_2$ 满足 $TP_9$ 的类型约束, $TP_9$ 放入两个编队的任务点子集后仍满足任务点间的时间约束。此时编队 $F_1$ ,  $F_2$ 进行任务代价对比,编队 $F_1$ 在执行 $TP_9$ 的任务过程中需要绕过一个威胁区域,付出更高的航程代价,因此 $F_2$ 在对比中获胜,向指挥中心发起投标,最终作为一个投标者中标,重分配后将任务点加入编队序列,此时编队 $F_2$ 新序列为 $\{TP_2, TP_9, TP_6\}$ 。

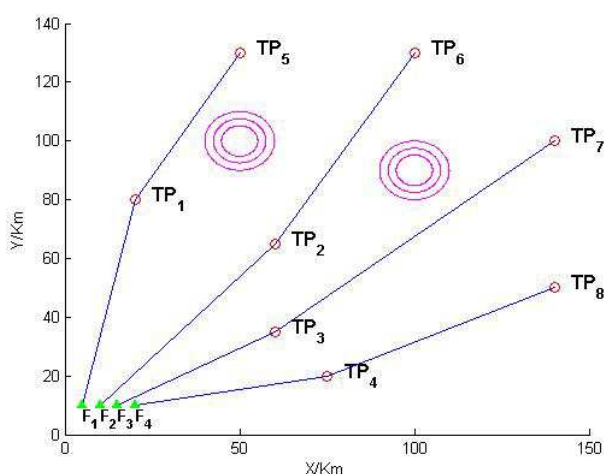


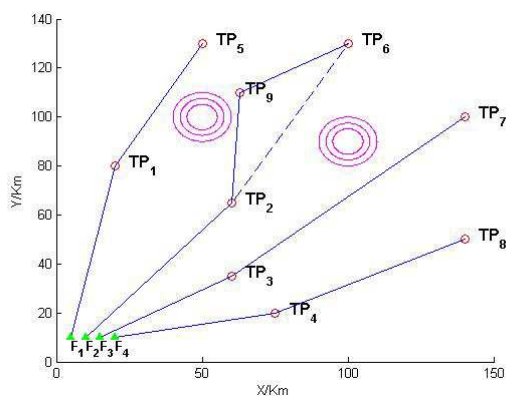
图 5.3 任务场景示意图

任务点取消:图 5.4(b)中当前编队 $F_2$ 的任务点序列为 $\{TP_2, TP_9, TP_6\}$ ,当编队前往执行任务点 $TP_2$ 上的任务时,任务点 $TP_6$ 上的任务被取消,系统重分配后,编队新序列为 $\{TP_2, TP_9\}$ 。

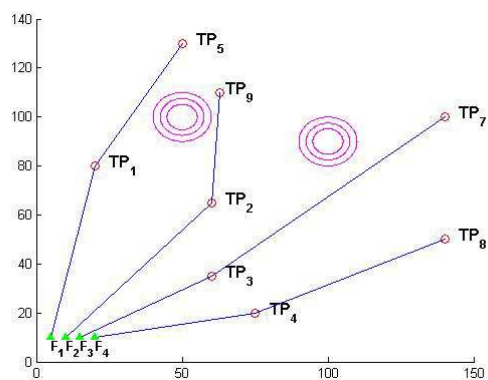
无人机战损:图 5.4(c)中编队 $F_1$ 在执行任务的过程中,编队中某一架无人机故障,使编队不满足 $TP_5$ 的任务类型约束,同时基地无无人机补充,将 $TP_5$ 进行重分配,由于只有编队 $F_2$ 满足接收此任务点后满足时间约束和类型约束,因此 $TP_5$ 被重分配给编队 $F_2$ 。

人为任务调整:图 5.4(d)中编队 $F_3$ 前往执行任务 $TP_3$ 的过程中,人为的向 $F_3$ 的任务集中加入 $TP_8$ ,由于 $TP_7$ 与 $TP_8$ 之间的任务执行时间存在重合,无法存在于同一编队任务集中,因此将其调整给编队 $F_4$ 。

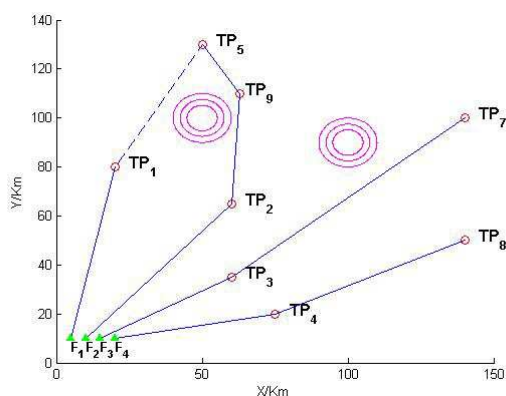
新增威胁区域:图 5.4(e)中右下角新增事先未被发觉的威胁区域,它的出现使得原序列为 $\{TP_3, TP_8\}$ 的编队 $F_3$ 在由前往执行任务 $TP_8$ 时,需要花费较大的代价,在进行航路重规划后发现, $F_3$ 无论如何也无法在满足执行时间约束的情况下执行 $TP_8$ ,将 $TP_8$ 交给指挥中心重分配,由于此时无编队满足执行 $TP_8$ 的条件,任务点 $TP_8$ 被放弃执行。



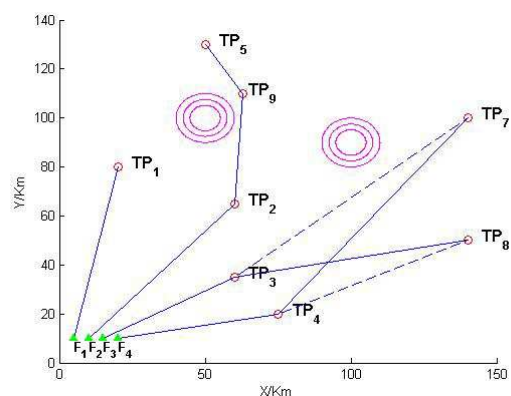
(a) 新增任务点后重分配



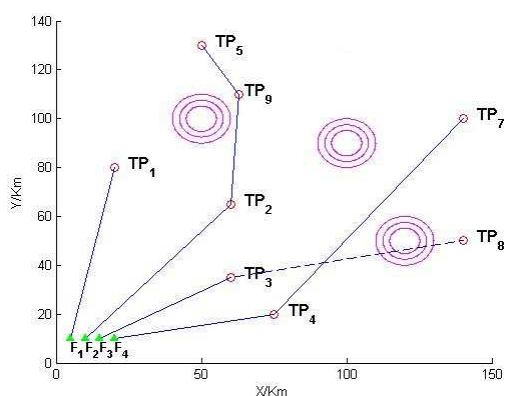
(b) 任务点取消后重分配



(c) 无人机战损后重分配



(d) 人为任务调整后重分配



(e) 新增威胁区域后重分配

图 5.4 多种情况下重分配仿真示意图

## 5.4 本章小结

在任务预规划的基础上，为了提高协同规划系统应对突发情况的应变能力，总结出了 5

种典型的重分配情况，介绍了合同网协议的内容，并将其改进，使用到任务的重分配中。通过对所提出的 5 种情况的仿真，展示出在不同情况下系统的重分配过程与效果。最终仿真结果证明，合同网协议在任务重分配中是一种有效可行的方法。

## 第六章 非完备信息下的空中目标意图识别与攻击决策

### 6.1 引言

本章开始以无人机对空作战为背景,针对第二章中所提及的无人机攻击决策系统,设计了在信息化战场下利用不完备信息的敌机意图识别与攻击决策的方法。在具有不确定性和不完备性特点的现代化战场信息环境下,充分利用各种战场信息、发挥多源信息效能与优势、探索无人作战飞机战场决策与战场态势以及敌方意图之间的耦合机理。

目前,意图识别研究问题已存在一些经典的方法,例如贝叶斯网络法,D-S(Dempster-Shafer)证据理论等等。文献[80]中提出了一种基于多实体贝叶斯网络法的意图识别方法,此方法需要获取大量的先验知识以保证可靠性。文献[81]中,针对反导作战中的弹道导弹意图预测问题,依照径向速度、航路捷径、落点预报等要素,研究了一种基于D-S证据理论的目标意图预测方法。诸如此类的文献数量庞大,所用的方法也不尽相同,然而涉及所获信息非完备条件下的意图预测的文献数量却十分有限。软集理论和粗糙集理论是用于处理不完备信息的两种经典的方法,其中粗糙集理论作为一种在数据挖掘领域的强大的工具,可用于不完备的信息的处理,相关的文献研究也比较丰富<sup>[82][83][84][85]</sup>。

本章首先进行战场态势信息不完备下的敌机意图识别,在2.4节空战态势与威胁因子定义的基础上,利用基于相似关系的粗糙集理论对已有的具有不确定性的空情信息转化为非完备区间数决策信息系统,并对其进行处理与挖掘,找出其中所蕴含的规则,再参考文献[87]的思路,将用区间数表示的当前的待识别空情与所获规则转化为联系数,利用联系数之间的贴进度来找出与当前空情最接近的规则,由此获得当前空情下的目标意图。

在完成目标意图识别的基础上,进行相应的攻击决策,匈牙利算法是求解指派问题一个经典而又有效的方法,在文献[86]中已成功应用于无人机的火力分配中,属于一种较为成熟的方法。但考虑到获得信息存在不完备性的特点,传统的匈牙利算法显然已不再适用。本章给出了改进的方法,使得匈牙利算法可以应用于存在信息不完备条件下的攻击决策中,并将敌方意图作为分配时的考虑因素之一,实现了一种基于意图的多机攻击决策。

### 6.2 非完备信息下的空中目标意图识别

#### 6.2.1 区间数与联系数

本章引言中提到将利用区间数转化为联系数的方法进行意图识别,首先介绍区间数以及联系数的一些相关概念。



## (1) 区间数的相关计算与定义

**定义 6.1<sup>[87]</sup>**: 有区间数  $[x^-, x^+]$  和  $[y^-, y^+]$ , 当  $x^- = y^-$  且  $x^+ = y^+$ , 则  $[x^-, x^+] = [y^-, y^+]$ 。

有区间数加法:

$$[x^-, x^+] + [y^-, y^+] = [x^- + y^-, x^+ + y^+] \quad (6.1)$$

区间数减法( $x^- \geq y^+$  时):

$$[x^-, x^+] - [y^-, y^+] = [x^- - y^+, x^+ - y^-] \quad (6.2)$$

**定义 6.2**: 有区间数  $[x^-, x^+]$  和常数  $y$ , 则有区间数加法:

$$[x^-, x^+] + y = [x^- + y, x^+ + y] \quad (6.3)$$

区间数减法:

$$[x^-, x^+] - y = [x^- - y, x^+ - y] \quad (6.4)$$

$$y - [x^-, x^+] = [y - x^+, y - x^-] \quad (6.5)$$

**定义 6.3<sup>[87]</sup>**: 对于区间数  $[x^-, x^+]$ , 设有  $l_x$ ,  $l_x = x^+ - x^-$ , 则称  $l_x$  为区间数  $[x^-, x^+]$  的长度。

**定义 6.4**: 对于区间数  $[x^-, x^+]$  和  $[y^-, y^+]$ , 大小关系用概率来表示:

$$P([x^-, x^+] > [y^-, y^+]) = \frac{\min\{l_x + l_y, \max(x^+ - y^-, 0)\}}{l_x + l_y} \quad (6.6)$$

若  $x^+ \leq y^-$ , 有  $P([x^-, x^+] > [y^-, y^+]) = 0$ , 即此时  $[x^-, x^+] \leq [y^-, y^+]$ 。

## (2) 二元联系数定义

**定义 6.5<sup>[87]</sup>**: 设  $A, B \in R$ ,  $i \in [-1, 1]$ , 则称  $U = A + Bi$  为二元联系数。

当  $A + B = N$ ,  $u = U/N$ ,  $a = A/N$ ,  $b = B/N$ , 则称  $u = a + bi$  为二元联系度。

**定义 6.6<sup>[89]</sup>**: 有两个联系数  $u_1 = a_1 + b_1 i$ ,  $u_2 = a_2 + b_2 i$ , 当  $r_1 = \sqrt{(a_1^2 + b_1^2)}$ ,  $r_2 = \sqrt{(a_2^2 + b_2^2)}$ , 称  $r_1$  和  $r_2$  为  $u_1$  和  $u_2$  的模。

**定义 6.7<sup>[89]</sup>**: 有两个联系数  $u_1 = a_1 + b_1 i$ ,  $u_2 = a_2 + b_2 i$  的模分别为  $r_1$  和  $r_2$ , 则

$$\Re(u_1, u_2) = \frac{\min(r_1, r_2)}{\max(r_1, r_2)} \quad (6.7)$$

称为  $u_1$  与  $u_2$  的贴近度。

(3) 区间数向联系数的转换<sup>[89]</sup>

对于区间数  $[x^-, x^+]$ , 令  $A = \frac{x^- + x^+}{2}$ ,  $B = x^+ - \frac{x^- + x^+}{2}$ , 根据联系数形式, 可写成

$$U = A + Bi, \text{ 即 } U = \frac{x^- + x^+}{2} + (x^+ - \frac{x^- + x^+}{2})i, i \in [-1, 1]。$$

### 6.2.2 粗糙集相关理论

本章利用了基于相似关系的粗糙集理论对获取的空情信息进行规则提取，首先需要介绍一下粗糙集中信息系统相关的概念。

**定义 6.8<sup>[82]</sup>:** 一个信息系统为  $(U, AT, F)$ ，其中：

$U = \{x_1, x_2, \dots, x_n\}$  称为论域，是所有对象的集合， $U$  中的  $x_i$  称为一个对象。

$AT = \{a_1, a_2, \dots, a_m\}$  称为属性集，是所有属性的集合， $AT$  中的每个  $a_i$  称为一个属性。

$F$  称为  $U$  与  $AT$  的关系集， $\forall a_i \in AT, x_j \in U, F(x_j, a_i)$  表示对象  $x_j$  在属性  $a_i$  上的属性值， $F(x_j, a_i) \in Va_i, Va_i$  为  $a_i$  的值域。

**定义 6.9<sup>[82]</sup>:** 对于一个决策信息系统  $(U, AT \cup D, F)$ ， $AT$  称为条件属性集， $D$  称为决策属性集， $AT \cap D = \emptyset$ 。

**定义 6.10<sup>[90]</sup>:** 在一个信息决策系统  $(U, AT \cup D, F)$  中，如果属性  $\forall a_i, a_i \in AT$  的值域  $Va_i$  为区间数，则称  $(U, AT \cup D, F)$  为区间数决策信息系统。

**定义 6.11<sup>[90]</sup>:** 当  $(U, AT \cup D, F)$  为一个区间数决策信息系统时，其中至少存在一个属性  $a_i, a_i \in AT$ ，其属性值的上界或者下界值缺失，缺失值记为“\*”，则称  $(U, AT \cup D, F)$  为非完备区间数决策信息系统。

**定义 6.12<sup>[90]</sup>:** 当  $(U, AT \cup D, F)$  为一个区间数决策信息系统时，可以进行规则提取

$$\bigcap \{(a_i, F(x, a_i)) \mid a_i \in AT\} \rightarrow \bigcap \{(d_i) \mid d_i \in D\} \quad (6.8)$$

$\cup$  表示对元素求析取，类似逻辑或； $\cap$  表示对元素求合取，类似逻辑与。规则代表从决策信息系统中获得的一个确定知识，对规则进行约简可以获得更加精简的规则。

### 6.2.3 非完备区间数决策信息系统中的相似关系定义

在非完备区间数决策信息系统  $(U, AT \cup D, F)$  中，对于条件属性集中的任意一个属性值，设其值为  $[f^-, f^+]$ ，有 4 种可能的情况出现，“\*”为缺失值：

- (1)  $f^- \neq *$  且  $f^+ \neq *$ ，称为完整型；
- (2)  $f^- = *$ ， $f^+ \neq *$ ，称为下界缺失型；
- (3)  $f^- \neq *$ ， $f^+ = *$ ，称为上界缺失型；
- (4)  $f^- = *$  且  $f^+ = *$ ，称为全缺失型。

相似关系相关定义如下：

**定义 6.13<sup>[90]</sup>:** 有对象  $x_i, x_j \in U$ ，则当  $x_i$  与  $x_j$  有相似关系的充要条件是： $x_i$  与  $x_j$  上的每个条件属性  $\forall a_k, a_k \in AT$  上都具有相似关系。

设对象  $x_i$  在属性  $a_k$  上的属性值为  $[a_k^-(x_i), a_k^+(x_i)]$ ， $x_j$  的为  $[a_k^-(x_j), a_k^+(x_j)]$ 。单个条件属性上的相似关系判定过程，属性  $a_k$  上的区间数平均长度记为  $Lavg(a_k)$ ：

(1) 当 $[a_k^-(x_i), a_k^+(x_i)]$ 与 $[a_k^-(x_j), a_k^+(x_j)]$ 都为完整型时。

若 $a_k^+(x_i) < a_k^-(x_j)$ 或者 $a_k^-(x_i) > a_k^+(x_j)$ 时, 两者在属性 $a_k$ 上不满足相似关系, 否则满足相似关系。

(2) 当 $[a_k^-(x_i), a_k^+(x_i)]$ 为下缺失型,  $[a_k^-(x_j), a_k^+(x_j)]$ 为完整型时。

若 $a_k^+(x_i) - \text{Lavg}(a_k) > a_k^+(x_j)$ 或者 $a_k^+(x_i) < a_k^-(x_j)$ 时, 两者在属性 $a_k$ 上不满足相似关系, 否则满足相似关系。

(3) 当 $[a_k^-(x_i), a_k^+(x_i)]$ 为上缺失型,  $[a_k^-(x_j), a_k^+(x_j)]$ 为完整型时。

若 $a_k^-(x_i) > a_k^+(x_j)$ 或者 $a_k^-(x_i) + \text{Lavg}(a_k) < a_k^-(x_j)$ 时, 两者在属性 $a_k$ 上不满足相似关系, 否则满足相似关系。

(4) 当 $[a_k^-(x_i), a_k^+(x_i)]$ 为上缺失型,  $[a_k^-(x_j), a_k^+(x_j)]$ 为下缺失型时。

若 $a_k^-(x_i) > a_k^+(x_j)$ 或者 $a_k^-(x_i) + \text{Lavg}(a_k) < a_k^+(x_j) - \text{Lavg}(a_k)$ 时, 两者在属性 $a_k$ 上不满足相似关系, 否则满足相似关系。

(5) 当 $[a_k^-(x_i), a_k^+(x_i)]$ 为上缺失型,  $[a_k^-(x_j), a_k^+(x_j)]$ 为上缺失型时。

若 $a_k^-(x_i) > a_k^-(x_j) + \text{Lavg}(a_k)$ 或者 $a_k^-(x_i) + \text{Lavg}(a_k) < a_k^-(x_j)$ 时, 两者在属性 $a_k$ 上不满足相似关系, 否则满足相似关系。

(6) 当 $[a_k^-(x_i), a_k^+(x_i)]$ 为下缺失型,  $[a_k^-(x_j), a_k^+(x_j)]$ 为下缺失型时。

若 $a_k^+(x_i) - \text{Lavg}(a_k) > a_k^+(x_j)$ 或者 $a_k^+(x_i) < a_k^+(x_j) - \text{Lavg}(a_k)$ 时, 两者在属性 $a_k$ 上不满足相似关系, 否则满足相似关系。

(7) 当 $[a_k^-(x_i), a_k^+(x_i)]$ 或 $[a_k^-(x_j), a_k^+(x_j)]$ 存在全缺失型时。

一个对象的全缺失型属性值与其他对象在此属性上都满足相似关系。

**定义 6.14<sup>[90]</sup>**: 对于对象 $x_i, x_j \in U$ , 其所有的相似类集合定义为

$$S(x_i) = \{x_j \in U \mid x_i \text{与} x_j \text{具有相似关系}\} \quad (6.9)$$

**定义 6.15<sup>[90]</sup>**: 对于对象 $x_i, x_j \in U$ , 定义其约简的分辨函数如下:

$$\Delta(x_i) = \bigcap_{x_j \in \{x \in U \mid D(x) \not\subseteq \partial(x_i)\}} \bigcup \alpha(x_i, x_j) \quad (6.10)$$

其中 $\partial(x_i) = \{F(x, D) \mid x \in S(x_i)\}$ 称为广义决策集,  $D(x)$ 为对象 $x$ 的决策属性。

$\alpha(x_i, x_j) = \{a \in AT \mid x_i \text{与} x_j \text{在属性} a \text{上不满足相似关系}\}$ 为 $x_i, x_j$ 之间的可分辨属性集。

## 6.2.4 不完备信息决策表中的规则获取

将可能存在一定不完备性的原始空战数据整理成一张非完备区间数决策信息表, 2.4 节中定义的五种威胁因子: 角度、速度、距离、高度和空战能力组成了信息表中的条件属性, 同时定义敌机意图为决策属性。本章将敌机的意图总结为 5 种: 进攻, 佯攻, 侦察, 防御与规避。表 6.1 为我方已有的一张非完备区间数决策信息表。

表 6.1 非完备区间数决策信息表

对象	能力威胁 $E$	角度威胁 $Ta$	速度威胁 $Tv$	距离威胁 $Td$	高度威胁 $Th$	意图
$x_1$	[0.72,0.79]	[0.25,*]	[0.53,0.7]	[0.32,0.4]	[0.25,0.35]	侦察
$x_2$	[0.84,0.99]	[0.4,0.49]	[0.72,1]	[0.4,0.58]	[0.36,0.47]	攻击
$x_3$	[0.68,0.77]	[0.3,0.48]	[0.52,0.7]	[*,*]	[0.34,0.49]	侦察
$x_4$	[0.78,0.93]	[0.49,0.7]	[0.71,0.95]	[0.44,0.69]	[0.37,0.47]	攻击
$x_5$	[0.28,0.37]	[0.14,0.21]	[* ,0.39]	[0.19,0.26]	[0.17,0.23]	规避
$x_6$	[* ,0.39]	[0.14,0.19]	[0.38,0.43]	[0.21,0.3]	[* ,0.23]	规避
$x_7$	[0.55,0.69]	[0.47,0.62]	[0.6,0.73]	[0.56,0.71]	[0.48,0.61]	佯攻
$x_8$	[0.4,0.57]	[0.6,0.69]	[0.43,0.55]	[0.68,0.72]	[0.52,*]	佯攻

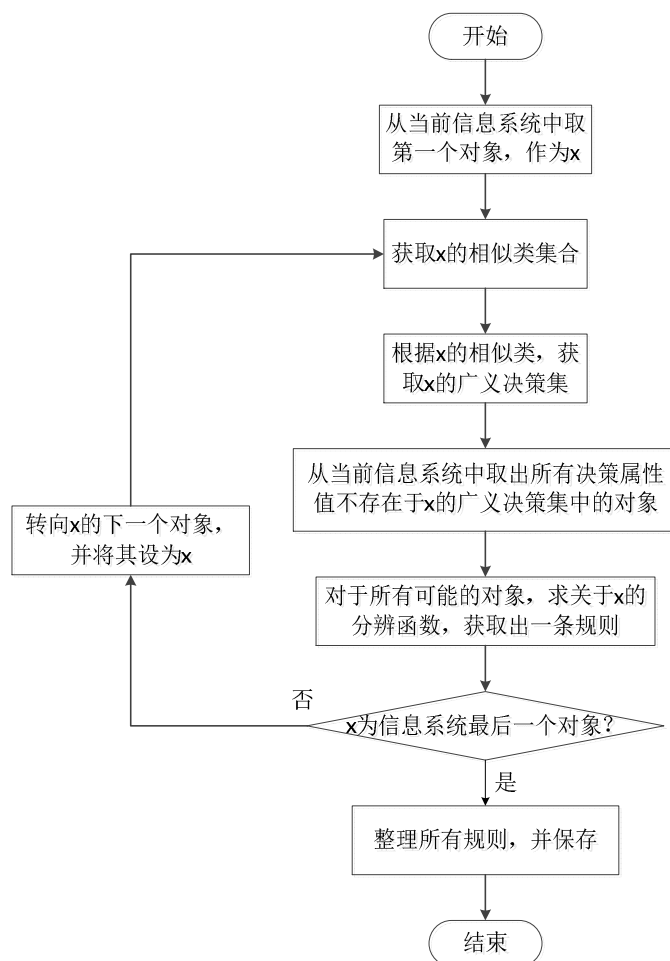


图 6.1 非完备决策信息表获取规则流程图

从表 6.1 中的决策信息表中获取规则的流程如图 6.1，具体描述如下：

Step1: 选取当前信息表第一个对象，设为  $x$ ；

Step2: 查找出与  $x$  满足相似关系的所有对象，组成其相似类集合  $S(x)$ ；

Step3: 根据  $x$  的相似类集合  $S(x)$ ，可以获得  $S(x)$  所包含的所有对象的意图，即  $x$  的广义决策集  $\partial(x)$ ；

Step4: 从当前信息表中的所有对象中，找出意图未在  $\partial(x)$  中出现的对象；

Step5: 对于上述所得的所有对象，根据定义，求出关于  $x$  的分辨函数，由此得到一条由对象  $x$  产生的规则；

Step6: 判断  $x$  是否为当前信息表中最后一个对象，若不是，转向  $x$  的下一个对象，并将其设为  $x$ ，返回 Step2，重复上述过程；若是，整理所有规则，删除冲突规则，并保存规则。

现以  $x_1$  为例，进行上述流程并获取一条规则： $S(x_1) = \{x_1, x_3\}$ ，此时可以获得  $x_1$  的广义决策集  $\partial(x_1) = \{\text{侦察}\}$ ，即广义决策集内只有一个意图：侦察；意图未在  $\partial(x_1)$  中出现的对象集合为  $\{x_2, x_4, x_5, x_6, x_7, x_8\}$ ，此时

$$\Delta(x_1) = \{(E \cup Ta \cup Tv \cup Th) \cap (Ta \cup Tv \cup Td \cup Th) \cap (E \cup Ta \cup Tv \cup Td \cup Th) \cap (E \cup Ta \cup Tv \cup Td \cup Th) \cap (E \cup Ta \cup Td \cup Th) \cap (E \cup Ta \cup Td \cup Th) \cap (E \cup Ta \cup Td \cup Th)\}$$

简化后得  $\Delta(x_1) = \{Ta, Th\}$ ，此时可得规则：若某架敌机对我机的角度威胁  $Ta \in [0.25, 0.37]$  同时高度威胁  $Th \in [0.25, 0.35]$  时，此敌机意图为侦察，规则的条件缺失值已用该属性的平均长度补充。整个决策信息表表 6.1 可获得的规则结果如表 6.2。

表 6.2 非完备决策信息表获取规则表

序号	规则条件	规则意图
1	$Ta \in [0.25, 0.37] \&\& Th \in [0.25, 0.35]$	侦察
2	$E \in [0.84, 0.99]$	攻击
3	$E \in [0.68, 0.77] \&\& Tv \in [0.52, 0.7]$	侦察
4	$E \in [0.28, 0.37] \&\& Ta \in [0.14, 0.21] \&\& Tv \in [0.23, 0.39] \&\& Th \in [0.17, 0.23]$	规避
5	$E \in [0.27, 0.39] \&\& Ta \in [0.14, 0.19] \&\& Th \in [0.1, 0.23]$	规避
6	$E \in [0.55, 0.69] \&\& Th \in [0.48, 0.61]$	佯攻
7	$E \in [0.4, 0.57] \&\& Th \in [0.52, 0.62]$	佯攻

### 6.2.5 意图识别的流程

当完成对一张非完备区间数决策信息表的规则获取后，进行下一步的目标意图识别，识

别过程如图 6.2 所示。首先利用 6.2.4 小节中的规则获取方法，规则区可实现从非完备区间数决策信息表中获取空战意图规则，并将规则保存。识别区则读取规则区所保存的规则，并将此时需要识别意图的威胁因子信息与规则进行匹配，识别出当前威胁因子信息下敌机的意图。识别的具体过程如下：

Step1: 将规则表中的各规则条件由区间数转化为联系数，同时将待识别威胁因子信息也由区间数转化为联系数；

Step2: 求待识别威胁因子信息与规则表中每条规则的规则条件的贴适度，构成贴适度表；

Step3: 对贴适度表，对各个威胁因子进行权重分配；

Step4: 根据权重可获得待识别威胁因子信息与每条规则的综合贴适度，取综合贴适度最高的规则的意图作为待识别威胁因子信息所携带的意图输出。

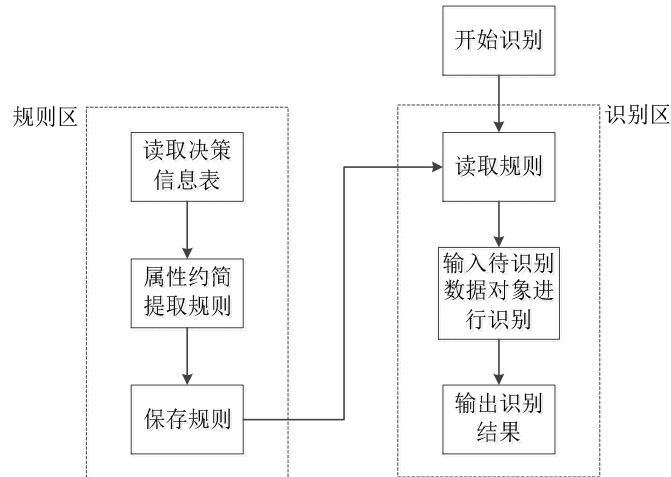


图 6.2 目标意图识别流程图

## 6.2.6 仿真算例

对表 6.3 中的威胁因子信息进行意图识别，读取表 6.2 作为规则：

表 6.3 待识别的威胁因子信息

能力威胁 $E$	角度威胁 $Ta$	速度威胁 $Tv$	距离威胁 $Td$	高度威胁 $Th$
$[0.65, 0.69]$	$[0.45, 0.55]$	$[0.61, 0.72]$	$[0.63, 0.71]$	$[0.50, 0.61]$

将表 6.3 中威胁因子由区间数转化为联系数，得到下表 6.4。

表 6.4 转化为联系数的威胁因子信息

能力威胁 $E$	角度威胁 $Ta$	速度威胁 $Tv$	距离威胁 $Td$	高度威胁 $Th$
$0.67 + 0.02i$	$0.5 + 0.05i$	$0.665 + 0.055i$	$0.67 + 0.04i$	$0.555 + 0.055i$

求表 6.4 中威胁因子信息与规则表中各规则的规则条件联系数形式下的贴适度，得到贴

近度表表 6.5。

表 6.5 威胁因子信息与规则的贴近度

规则编号	能力威胁贴 近度	角度威胁贴 近度	速度威胁贴 近度	距离威胁贴 近度	高度威胁贴 近度
1	Inf	0.6284	Inf	Inf	0.5453
2	0.7301	Inf	Inf	Inf	Inf
3	0.9228	Inf	0.9241	Inf	Inf
4	0.4895	0.3552	0.4760	Inf	0.3626
5	0.4987	0.3321	Inf	Inf	0.3299
6	0.9308	Inf	Inf	Inf	0.9841
7	0.7346	Inf	Inf	Inf	0.9671

表中 Inf 表示规则中的规则条件不包含此威胁因子，因此与待识别态势信息在此威胁因子上不存在贴近度。

为每个威胁因子赋权值，权值的选取可以采用主观赋值、专家赋值以及信息熵法。本章采用文献[88]中的信息熵的方法，得到的权重结果如下表 6.6:

表 6.6 各威胁因子权重分配

能力威胁 $E$	角度威胁 $Ta$	速度威胁 $Tv$	距离威胁 $Td$	高度威胁 $Th$
0.0657	0.3164	0.4628	0	0.1551

对于表 6.5 每行规则，假设有效属性(贴近度值不为 Inf)贴近度为  $C_1, C_2, \dots, C_k$ ，则待识

别态势信息与其综合贴近度为  $\frac{\sum_{i=1}^k w_i * C_i}{\sum_{i=1}^k w_i}$ ，其中  $w_i$  表示此规则中第  $i$  个有效属性的权重。获

得待识别态势与所有规则的综合贴近度如下表 6.7。

表 6.7 威胁因子信息与所有规则的综合贴近度

规则编号	1	2	3	4	5	6	7
综合贴近度	0.6284	0.7301	0.9239	0.4318	0.3608	0.9308	0.7346

表 6.7 中待识别的威胁因子信息与规则 6 的综合贴近度最大，此时推断出威胁因子信息对应的意图为规则 6 对应的意图：佯攻。

## 6.3 存在信息不完备条件下的多目标分配决策

### 6.3.1 问题描述

设有  $m$  架我机遭遇  $n$  架敌机，现需要制定对敌机进行合理的打击分配方案，使我方能够在一定的战场优势下，优先打击对我方意图威胁较大的目标。对于  $m$  架我机与  $n$  架敌机，可以建立一个  $m \times n$  的评价矩阵，其中第  $i$  行第  $j$  列的元素代表第  $i$  架我机对第  $j$  架敌机的评价价值，用于表述我机打击敌机的收益评价。

评价值由我机对敌机的态势优势、空战效能优势和敌机对我机的意图威胁组成。具体定义如下：

令矩阵  $Q$  为评价矩阵， $Q_{ij}$  表示第  $i$  架我机相对第  $j$  架敌机的评价值。

$$Q_{ij} = \omega_1 \times (1 - TS_{ij}) + \omega_2 \times (1 - TI_{ij}) + \omega_3 \times (1 - E_{ji}) \quad (6.11)$$

其中：

$TS_{ij}$  为第  $i$  架我机对第  $j$  架敌机的态势优势值；

$TI_{ij}$  为第  $j$  架敌机对第  $i$  架我机的意图威胁值；

$E_{ji}$  为第  $i$  架我机对第  $j$  架敌机的空战能力威胁值，或称为空战能力优势值。

$\omega_1$ 、 $\omega_2$ 、 $\omega_3$  为大于 0 的权值因子， $\omega_1 + \omega_2 + \omega_3 = 1$ 。

式中态势优势、意图威胁的定义具体如下。

#### (1) 态势优势定义

$$TS_{ij} = \alpha_1 \times Ta_{ji} + \alpha_2 \times Tv_{ji} + \alpha_3 \times Td_{ji} + \alpha_4 \times Th_{ji} \quad (6.12)$$

其中， $Ta_{ji}$ 、 $Tv_{ji}$ 、 $Td_{ji}$ 、 $Th_{ji}$  分别为我机  $i$  对敌机  $j$  的角度，速度，距离，高度威胁，

$\alpha_1$ 、 $\alpha_2$ 、 $\alpha_3$ 、 $\alpha_4$  为大于零的权值，且  $\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 1$ 。

#### (2) 意图威胁定义

传统的空战评估模型侧重于对作战飞机性能和战场态势的评估，极少考虑目标意图威胁。本章将目标意图威胁加入评价模型中，有利于我方作战飞机发现潜在威胁，提高决策准确性，利用 6.2 节中的方法，使我方能够事先识别敌方各架飞机对我方各架飞机的意图。敌机  $j$  对我机  $i$  每种意图对应的意图威胁值如表 6.8，敌机对我机的意图越危险，对应的意图威胁值越大。

表 6.8 意图威胁值表

意图	攻击	佯攻	侦查	防御	规避
意图威胁值	0.9	0.7	0.5	0.3	0.1



### 6.3.2 目标平衡分配问题的数学模型

多目标的分配决策问题可分为平衡分配问题和非平衡分配问题，这里首先考虑平衡分配问题，即我机数量  $m$  等于敌机数量  $n$ 。针对平衡分配问题，做出如下前提假设：

**假设 6.1<sup>[86]</sup>**：(1) 每架敌机仅受一架我机攻击；(2) 每架我机仅对一架敌机进行攻击。

为每一对我机/敌机定义  $W_{ij}$

$$W_{ij} = \begin{cases} 1 & \text{第 } i \text{ 架我机攻击第 } j \text{ 架敌机} \\ 0 & \text{第 } i \text{ 架我机不攻击第 } j \text{ 架敌机} \end{cases} \quad (6.13)$$

满足  $\sum_{i=1}^m W_{ij} = 1, j=1, \dots, n$ ，同时  $\sum_{j=1}^n W_{ij} = 1, i=1, \dots, m$ 。

平衡分配后，要实现分配结果总评价值之和最小。这样既能保证我机拥有较高的战场优势，也可以优先消灭对我方有较大意图威胁的目标。即目标函数为：

$$J = \min \sum_{i=1}^m \sum_{j=1}^n Q_{ij} W_{ij} \quad (6.14)$$

### 6.3.3 匈牙利算法的改进

在机载设备能力限制，战场环境复杂，敌方无人机可能具有隐身性和电子干扰能力等情况普遍存在的前提下，我方在获知敌机速度、距离、角度、高度信息方面以及双方的空战效能都可能存在不确定的特征，这些不确定特征将进一步表现在评价矩阵  $Q$  中，当我机  $i$  对敌机  $j$  的评价值  $Q_{ij}$  存在不确定时，将其视为一个区间数。

匈牙利算法由数学家康尼格率先提出，1955 年数学家库恩在解决指派问题时，对理论进行了引用与改进，由于提出该算法的两位学者都来自匈牙利，所以被称作匈牙利算法。匈牙利算法可以快速可靠地解决指派问题，具有理论简洁、求解精确等优点。经典的匈牙利算法主要分为三步<sup>[86]</sup>：

**Step1**：对评价矩阵进行变换，使各行各列中都出现 0 元素；

**Step2**：进行指派，寻找最优分配；

**Step3**：如果评价矩阵中独立的 0 元素不够，无法完成指派，做最少的直线覆盖现有的 0 元素，从没有被直线覆盖的部分获得一个子评价矩阵，找出子评价矩阵中的最小元素，即可在保证原有 0 元素个数不变的情况下，得到新的独立 0 元素，若产生了足够的独立 0 元素，则可以继续寻找最优分配，否则，重复执行此步骤。

对于本章存在区间数的评价矩阵而言，常规的匈牙利算法将无法对其进行目标分配，下面

对匈牙利算法进行适当的拓展,使其可以处理存在信息不确定时的目标分配问题。从上述匈牙利算法描述可以看出,完成整个指派过程,Step1 和 Step3 都需要确认某一行或者某一列或者评价矩阵中的某些元素组成的子矩阵中的最小元素,从而获得独立的 0 元素用于分配。当行或者列或者子矩阵中存在区间数导致最小值不确定时,此时考虑不同的最小值情况,将评价矩阵进行拆分讨论,得到确定的最小值。针对改进匈牙利算法制定的具体规则如下:

对于平衡分配下的评价矩阵  $Q$ 。假设某行(列)或者子矩阵需要进行减去最小值操作,以行为例:

(1) 此行中不存在区间数,元素都为常数,设此行最小值为  $c$ ,那么此行中其他值减去  $c$  即可获得 0 元素。

(2) 此行中存在唯一区间数  $[a, b]$ , 设此行常数最小值为  $c$ 。那么如果:

①  $c \leq a$ , 则此行确定的最小值为  $c$ , 此行各值减去最小值可获得 0 元素, 区间数减去常数的计算过程参考式(6.4);

②  $c \geq b$ , 则此行确定的最小值为  $[a, b]$ , 此行各值减去最小值可获得 0 元素, 常数减去区间数的计算过程参考式(6.5), 区间数减去区间数的计算过程参考式(6.2), 最小值减去自身将得到 0 元素;

③  $a < c < b$ , 则此行最小值不确定, 需要对区间数  $[a, b]$  进行分割讨论, 将评价矩阵分为针对区间数  $[a, b]$  的  $[a, c]$  和  $[c, b]$  两种情况, 对每种情况下评价, 再查找规则。

(3) 此行中存在  $k(2 \leq k < n)$  个区间数  $[a_1, b_1], [a_2, b_2] \dots [a_k, b_k]$ , 其中  $\min(\{a_1, a_2, \dots, a_k\}) = a_f$ , 在所有的  $k$  个区间数中存在一个或多个区间数  $[a_{f1}, b_{f2}], [a_{f2}, b_{f2}] \dots [a_{ft}, b_{ft}] (1 \leq t < k)$ ,  $a_{f1}, a_{f2}, \dots, a_{ft}$  的值都等于  $a_f$ , 且都按上界值从小到大排列, 即  $b_{f1} \leq b_{f2} \leq \dots \leq b_{ft}$ 。  $\min(\{a_1, a_2, \dots, a_k\} / \{a_{f1}, a_{f2}, \dots, a_{ft}\}) = a_s$  (文中  $(A/B)(B \in A)$  表示集合  $A$  除去集合  $B$  中元素的剩余部分), 在此行所有区间数中, 存在至少一个区间数的下界值等于  $a_s$ 。设此行常数中最小值为  $c$ 。那么如果:

①  $c \leq a_f$ , 则此行确定最小值为  $c$ , 此行各值减去最小值可获得 0 元素;

②  $b_{f1} > a_s$  且  $a_f < c \leq a_s$ , 此行最小值不确定, 从  $[a_{f1}, b_{f1}]$  开始, 将评价矩阵拆分成此行为  $[a_{f1}, c]$  和  $[c, b_{f1}]$  两种情况, 对于拆分后得到的两个评价矩阵, 分别对两个评价矩阵根据此行的  $[a_{f2}, b_{f2}]$ , 再分为  $[a_{f2}, c]$  和  $[c, b_{f2}]$  两种情况下的评价矩阵, 得到四种情况的评价矩阵, 每种情况下再对  $[a_{f3}, b_{f3}]$  进行拆分, 如此重复, 最终对  $[a_{ft}, b_{ft}]$  完成拆分。拆分后得到所有情况下的评价矩阵中此行若存在值为  $[a_f, c]$ , 则确定其为此行最小值, 认为所有拆分得到的  $[a_f, c]$  相减为 0; 若此行不存在值为  $[a_f, c]$ , 则  $c$  为此种情况下此行确定最小值;

③  $b_{f1} > a_s$  且  $c > a_s$ , 此行最小值不确定, 从  $[a_{f1}, b_{f1}]$  开始, 将评价矩阵拆分成此行为  $[a_{f1}, a_s]$  和  $[a_s, b_{f1}]$  两种情况, 对于拆分后得到的两个评价矩阵, 分别对两个评价矩阵根据此

行的 $[a_{f2}, b_{f2}]$ ，再分为 $[a_{f2}, a_s]$ 和 $[a_s, b_{f2}]$ 两种情况下的评价矩阵，得到四种情况的评价矩阵，每种情况下再对 $[a_{f3}, b_{f3}]$ 进行拆分，如此重复，最终对 $[a_{fi}, b_{fi}]$ 完成拆分。拆分后得到所有情况下的评价矩阵中此行若存在值为 $[a_f, a_s]$ ，则确定其为此行确定的最小值，认为所有拆分得到的 $[a_f, a_s]$ 相减为0；若此行不存在值为 $[a_f, a_s]$ ，则对此情况下的评价矩阵此行再进行查找改进规则；

④  $b_{f1} \leq a_s$  且  $a_{f1} < c < b_{f1}$ ，此行最小值不确定，从 $[a_{f1}, b_{f1}]$ 开始，将评价矩阵拆分成此行为 $[a_{f1}, c]$ 和 $[c, b_{f1}]$ 两种情况，对于拆分后得到的两个评价矩阵，分别对两个评价矩阵根据此行的 $[a_{f2}, b_{f2}]$ ，再分为 $[a_{f2}, c]$ 和 $[c, b_{f2}]$ 两种情况下的评价矩阵，得到四种情况的评价矩阵，每种情况下再对 $[a_{f3}, b_{f3}]$ 进行拆分，如此重复，最终对 $[a_{fi}, b_{fi}]$ 完成拆分。拆分后得到所有情况下的评价矩阵中此行若存在值为 $[a_f, c]$ ，则确定其为此行最小值，认为所有拆分得到的 $[a_f, c]$ 相减为0；若此行不存在 $[a_f, c]$ ，则 $c$ 为此种情况下此行确定最小值；

⑤  $b_{f1} \leq a_s$  且  $b_{f1} \leq c$ ，此行最小值不确定，从 $[a_{f2}, b_{f2}]$ 开始，将评价矩阵拆分成此行为 $[a_{f2}, b_{f1}]$ 和 $[b_{f1}, b_{f2}]$ 两种情况，对于拆分后得到的两个评价矩阵，分别对两个评价矩阵根据此行的 $[a_{f3}, b_{f3}]$ ，再分为 $[a_{f3}, b_{f1}]$ 和 $[b_{f1}, b_{f3}]$ 两种情况下的评价矩阵，得到四种情况的评价矩阵，每种情况下再对 $[a_{f4}, b_{f4}]$ 进行拆分，如此重复，最终对 $[a_{fi}, b_{fi}]$ 完成拆分。拆分后得到所有情况下的评价矩阵中此行都存在值为 $[a_f, b_{f1}]$ ，其为此行确定最小值，认为所有拆分得到的 $[a_f, b_{f1}]$ 相减为0。

(4) 此行中存在  $n$  个区间数 $[a_1, b_1], [a_2, b_2] \dots [a_n, b_n]$ ，即此行所有元素都为区间数。其中  $\min(\{a_1, a_2, \dots, a_n\}) = a_f$ ，在所有的  $n$  个区间数中存在一些区间数 $[a_{f1}, b_{f2}]$ ， $[a_{f2}, b_{f2}] \dots [a_{fi}, b_{fi}]$  ( $1 \leq t < n$ )，其中  $a_{f1}, a_{f2}, \dots, a_{fi}$  的值都等于  $a_f$ ，且所有区间数都按上界值从小到大排列，即  $b_{f1} \leq b_{f2} \leq \dots \leq b_{fi}$ 。  $\min(\{a_1, a_2, \dots, a_n\} / \{a_{f1}, a_{f2}, \dots, a_{fi}\}) = a_s$ ，在此行所有区间数中，存在至少一个区间数的下界值等于  $a_s$ 。那么如果：

①  $b_{f1} > a_s$ ，处理过程与(3)中的③过程相同；

②  $b_{f1} \leq a_s$ ，处理过程与(3)中的⑤过程相同。

(5) 此行中存在  $k$  ( $2 \leq k \leq n$ ) 个区间数  $[a_1, b_1], [a_2, b_2] \dots [a_k, b_k]$ ，其中  $a_1 = a_2 = \dots = a_k = a_f$ ，且所有区间数都按上界值从小到大排列，即  $b_1 \leq b_2 \leq \dots \leq b_k$ 。那么如果：

① 当  $k = n$  时，即此行全部为区间数且各值下界相同，从 $[a_2, b_2]$ 开始，将评价矩阵拆分成此行为 $[a_2, b_1]$ 和 $[b_1, b_2]$ 两种情况，对于拆分后得到的两个评价矩阵，分别对两个评价矩阵根据此行的 $[a_3, b_3]$ ，再分为 $[a_3, b_1]$ 和 $[b_1, b_3]$ 两种情况下的评价矩阵，得到四种情况的评价矩阵，每种情况下再对 $[a_4, b_4]$ 进行拆分，如此重复，最终对 $[a_k, b_k]$ 完成拆分。拆分后得到所有情况下的评价矩阵此行中都存在值为 $[a_f, b_1]$ ，其为此行确定最小值，认为所有拆分得到的 $[a_f, b_1]$ 相减为0；

②当  $k < n$  时, 设此行常数中最小值为  $c$ 。若  $c \leq a_f$ , 则此行确定最小值为  $c$ ; 若  $a_1 < c < b_1$ , 处理过程与(3)中的④过程相同; 若  $b_1 \leq c$ , 处理过程与(5)中①的过程相同。

### 6.3.4 不完备信息下的目标平衡分配流程

改进匈牙利算法用于存在不完备情形下的目标平衡分配时, 具体算法流程如图 6.3 所示, 其具体描述过程如下:

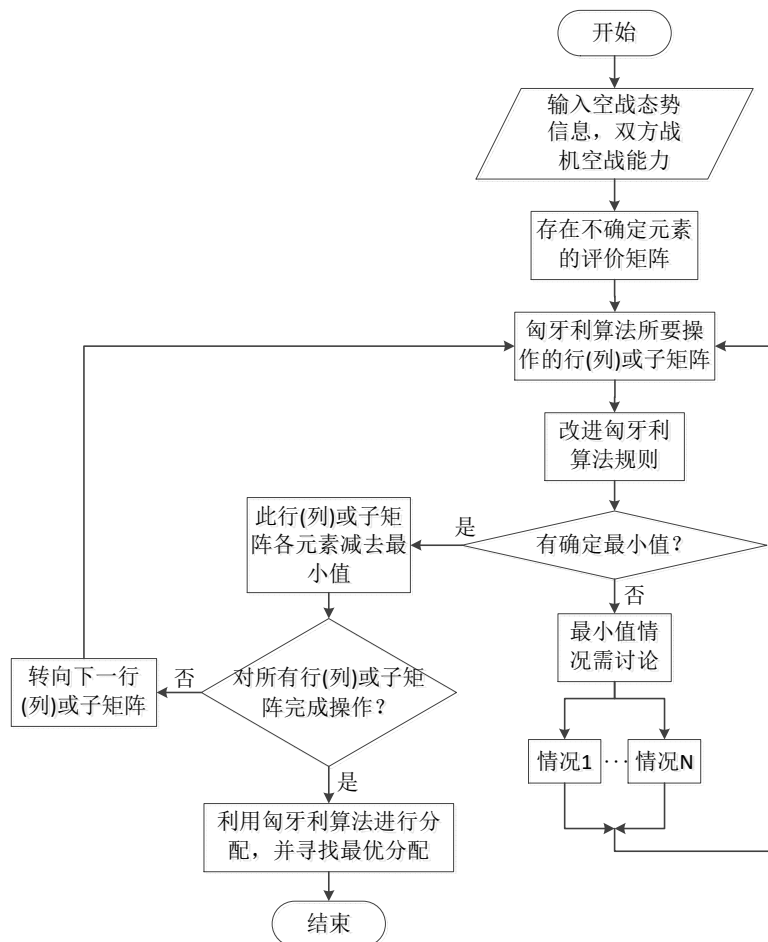


图 6.3 不完备信息下的目标平衡分配算法流程

**Step1:** 通过获取战场态势以及我机与敌机的空战效能, 可以得到一个存在不完备元素的评价矩阵用于目标分配;

**Step2:** 在按照匈牙利算法进行处理的过程中, 第一步和第三步都会需要带有不确定元素的评价矩阵中的最小元素, 以产生 0 元素。当需要对某种情况下的评价矩阵中某行(列)或子评价矩阵进行取最小元素操作时, 如确定最小元素转 Step3; 如不存在确定的最小元素时, 对照改进的匈牙利算法规则将评价矩阵进行拆分讨论, 直到对于每一种情况下的评价矩阵的此行(列)或子评价矩阵可确定最小元素, 并将此时的评价矩阵转 Step3;

Step3: 当评价矩阵中此行(列)或子评价矩阵确定了最小元素后, 此行(列)减去最小元素获得 0 元素, 子评价矩阵进行增加独立 0 元素操作。如果完成了对所有行和列以及子评价矩阵的取 0 操作, 可以利用原匈牙利算法中的第二步进行指派, 寻找最优分配; 如果仍有行(列)以及子评价矩阵需要完成找最小值的操作, 转向此行(列)以及子评价矩阵, 返回 Step2;

Step4: 针对多种情况下的已使用匈牙利算法进行分配的评价矩阵, 总结并比较所有的分配方式, 选择总评价值最小的情况作为最终的分配方案。

### 6.3.5 仿真算例

现考虑一种简单的情况, 我机和敌机数量都为 3, 评价矩阵如下:

$$\begin{bmatrix} 0.2 & 0.6 & 0.2 \\ [0,0.2] & [0.1,0.3] & 0.1 \\ 0.2 & [0.1,0.3] & 0.3 \end{bmatrix}$$

(1)根据经典的匈牙利算法, 第一行可以直接取最小值 0.2, 得到矩阵:

$$\begin{bmatrix} 0 & 0.4 & 0 \\ [0,0.2] & [0.1,0.3] & 0.1 \\ 0.2 & [0.1,0.3] & 0.3 \end{bmatrix};$$

(2)对第二行使用改进的匈牙利算法, 由于没有明确的最小值, 可以根据改进规则(3)中的

$$\textcircled{2}, \text{将矩阵拆分为} \begin{bmatrix} 0 & 0.4 & 0 \\ [0,0.1] & [0.1,0.3] & 0.1 \\ 0.2 & [0.1,0.3] & 0.3 \end{bmatrix} \text{和} \begin{bmatrix} 0 & 0.4 & 0 \\ [0.1,0.2] & [0.1,0.3] & 0.1 \\ 0.2 & [1,0.3] & 0.3 \end{bmatrix} \text{两种情况, 此时}$$

可以确定第二行最小值分别为 [0,0.1] 和 0.1, 分别减去最小值可被转换为

$$\begin{bmatrix} 0 & 0.4 & 0 \\ 0 & [0,0.3] & [0,0.1] \\ 0.2 & [0.1,0.3] & 0.3 \end{bmatrix} \text{和} \begin{bmatrix} 0 & 0.4 & 0 \\ [0,0.1] & [0,0.2] & 0 \\ 0.2 & [0.1,0.3] & 0.3 \end{bmatrix};$$

(3) 对第三行同样使用改进的匈牙利算法, 根据改进规则(2)中的③, 上述的两种情况的评价矩阵可得到 4 种情况:

$$\begin{bmatrix} 0 & 0.4 & 0 \\ 0 & [0,0.3] & [0,0.1] \\ [0,0.1] & 0 & [0.1,0.2] \end{bmatrix}, \begin{bmatrix} 0 & 0.4 & 0 \\ 0 & [0,0.3] & [0,0.1] \\ 0 & [0,0.1] & 0.1 \end{bmatrix}, \begin{bmatrix} 0 & 0.4 & 0 \\ [0,0.1] & [0,0.2] & 0 \\ [0,0.1] & 0 & [0.1,0.2] \end{bmatrix},$$

$$\begin{bmatrix} 0 & 0.4 & 0 \\ [0,0.1] & [0,0.2] & 0 \\ 0 & [0,0.1] & 0.1 \end{bmatrix};$$

(4)对每列元素减去该列的最小元素的操作,过程与上述行操作相同,最终原评价矩阵总共被拆分为6种情况:

$$\begin{bmatrix} 0 & 0.4 & 0 \\ 0 & [0,0.3] & [0,0.1] \\ [0,0.1] & 0 & [0.1,0.2] \end{bmatrix}, \begin{bmatrix} 0 & [0.3,0.4] & 0 \\ 0 & 0 & [0,0.1] \\ 0 & 0 & 0.1 \end{bmatrix}, \begin{bmatrix} 0 & [0.3,0.4] & 0 \\ 0 & [0,0.3] & [0,0.1] \\ 0 & 0 & 0.1 \end{bmatrix},$$

$$\begin{bmatrix} 0 & 0.4 & 0 \\ [0,0.1] & [0,0.2] & 0 \\ [0,0.1] & 0 & [0.1,0.2] \end{bmatrix}, \begin{bmatrix} 0 & [0.3,0.4] & 0 \\ [0,0.1] & 0 & 0 \\ 0 & 0 & 0.1 \end{bmatrix}, \begin{bmatrix} 0 & [0.3,0.4] & 0 \\ [0,0.1] & [0,0.2] & 0 \\ 0 & 0 & 0.1 \end{bmatrix}。$$

(5)对上述6种情况进行匈牙利算法分配步骤,得到如下三种分配方式:

$$\textcircled{1} \begin{bmatrix} * & * & \otimes \\ \otimes & * & * \\ * & \otimes & * \end{bmatrix}, \textcircled{2} \begin{bmatrix} * & * & \otimes \\ * & \otimes & * \\ \otimes & * & * \end{bmatrix}, \textcircled{3} \begin{bmatrix} \otimes & * & * \\ * & * & \otimes \\ * & \otimes & * \end{bmatrix}, \otimes \text{表示此列所代表的敌机分配给}$$

此行所代表的我机,\*代表一个可为任意值的区间值或常数,计算三种模式分配下的总评价值分别为:  $[0.3,0.7]$ ,  $[0.5,0.7]$ ,  $[0.4,0.6]$ 。

根据式(6.6),  $P([0.3,0.7] > [0.4,0.6]) = P([0.4,0.6] > [0.3,0.7]) = 0.5$ ,因此认为  $[0.3,0.7]$  与  $[0.4,0.6]$  之间效果相等,同时  $P([0.5,0.7] > [0.4,0.6]) > 0.5$ ,  $P([0.5,0.7] > [0.3,0.7]) > 0.5$ ,则  $[0.3,0.7]$  和  $[0.4,0.6]$  同为最小评价值区间,因此①和③是最好的两个分配方式。

### 6.3.6 改进匈牙利算法应用于非平衡分配

上述方法都是建立在平衡分配的基础上,现考虑将其拓展到非平衡分配的问题中,即  $m \neq n$ 。

(1)我方有  $m$  架飞机,敌方有  $n$  架飞机,  $n > m$ ,每一架我机都需要攻击一架敌机,既满足约束条件:

$$\sum_{j=1}^n W_{ij} = 1, i=1, \dots, m \quad (6.15)$$

考虑将评价矩阵进行补齐,在评价矩阵中增加  $n-m$  行填充元素  $\oplus$  (此元素没有实际价

值),使得矩阵成为一个方阵,这样可以使用平衡分配情况下的改进匈牙利算法。例如:两架

我机,四架敌机,评价矩阵为 $\begin{bmatrix} 0.2 & [0,0.2] & 0.2 & 0.3 \\ 0.6 & [0.2,0.3] & [0.1,0.2] & 0.4 \end{bmatrix}$ ,进行补齐后获得新评价矩

阵为 $\begin{bmatrix} 0.2 & [0,0.2] & 0.2 & 0.3 \\ 0.6 & [0.2,0.3] & [0.1,0.2] & 0.4 \\ \oplus & \oplus & \oplus & \oplus \\ \oplus & \oplus & \oplus & \oplus \end{bmatrix}$ ,进行分配可以获得分配方式 $\begin{bmatrix} * & \otimes & * & * \\ * & * & \otimes & * \\ \oplus & \oplus & \oplus & \oplus \\ \oplus & \oplus & \oplus & \oplus \end{bmatrix}$ 。

(2) 我方有  $m$  架飞机,敌方有  $n$  架飞机,  $m > n$ , 每一架敌机都需要一架我机攻击,既满足约束条件:

$$\sum_{i=1}^m W_{ij} = 1, j=1, \dots, n \quad (6.16)$$

思路与(1)相同,通过在评价矩阵中补充  $m-n$  列填充元素  $\oplus$ ,使矩阵变成一个方阵后,再使用平衡分配情况下的改进匈牙利算法。

## 6.4 本章小结

无人机在军用领域的发展,改变了现代战场的作战方式,对无人机的研究也成为一大热点。本章首先针对信息不完备时对敌方无人机意图预测的问题,根据敌机信息获得不完备信息决策表,将基于相似关系的粗糙集方法和联系数的贴近度相结合,实现了对敌方无人机的意图识别,仿真验证了方法的可行性。提出了一种改进的匈牙利算法,解决一种信息不完备条件下的,考虑敌机意图的无人机攻击决策问题。提出的改进匈牙利算法由经典匈牙利算法出发,通过定义存在区间数的评价矩阵并建立规则,使得所研究的行(列)以及子评价矩阵中呈现最小值,实现平衡以及非平衡问题下的多目标分配。

## 第七章 无人机协同任务规划系统仿真设计

### 7.1 引言

无人机协同任务规划系统的核心是,根据一定的战场环境和任务要求,给定无人机种类和数量,分析出完成任务所需要的无人机编队数量、编队组成等信息,将合适的任务在合适的时间分配给合适的无人机编队,从而使无人机编队拥有一组有序的任务,综合考虑各无人机编队平台的性能和分配的任务计划进行航路规划,在各编队依照计划沿航路执行任务的过程中,面对突发情况能够以最小的代价进行任务的合理调整,保证任务的执行率,提高系统的鲁棒性。

本章针对上文中所讨论的无人机协同任务规划系统,将无人机任务分配,航路规划,重分配等技术,在 Microsoft Visual Studio 2015 平台上,利用 C/C++ 语言分别进行程序设计与实现。并将几个部分相融合,利用 MFC 技术搭建信息输入与结果输出的人机界面,实现了一个协同任务规划系统仿真软件,软件中使用了 TeeChart 控件进行无人机编队航路规划与任务执行模拟的显示。

### 7.2 协同任务规划系统软件架构

#### 7.2.1 软件整体结构

整体的软件结构如下图 7.1 所示,简要叙述一下各模块之间的关系:

(1) 信息输入模块负责任务信息、地图信息以及我方无人机信息的输入,这一块是整个软件主要的信息输入部分,后续的各模块运算均依赖于此信息;

(2) 任务分配与编队分配模块,该部分首先对所输入的任务进行合理的分配,将整个任务点集划分为各个子任务点集方便编队执行。针对式(3.6)中任务分配的目标函数  $Z$ ,调节其中的参数  $k_1$ 、 $k_2$ 、 $a_1$ 、 $a_2$  的大小,使得产生:1、无人机数量最少优先、2、航程代价最小优先、3、任务收益最高优先三种不同侧重的任务分配目标,任务分配模块依此产生三种分配结果,编队分配则依照任务分配出的分配结果,对执行任务的每个编队分配执行任务所需的无人机;

(3) 分配评估模块将统计不同分配结果的各编队任务目标覆盖率、任务收益、任务代价,并据此对分配结果进行综合评估;

(4) 航路规划与任务过程模拟模块,航路规划负责依照任务分配的结果,产生编队执行任务过程中的航路规划结果,任务过程模拟则将结合任务分配与航路规划的结果,模拟各编



队执行任务的过程，实时显示编队工作状态，所在位置等信息。

(5) 在编队模拟执行任务的过程中，允许变更信息的输入，此时重分配模块将根据输入的变更信息进行任务的重分配，并依据重分配结果调整无人机编队的航路；

(6) 航路评估模块负责在上述模拟过程完成后，对所规划的航路的效果进行评估。

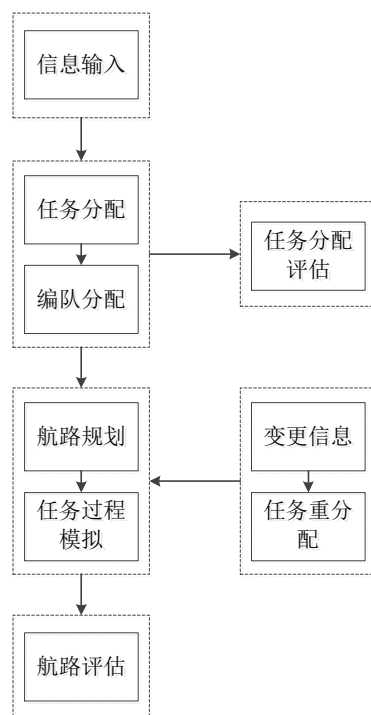


图 7.1 整体软件结构图

本章所设计的软件中，使用一种算法与存储区域相分离的结构，如下图 7.2 所示，信息输入完成后，所有的输入信息都将存储在系统的存储区中。每个模块直接从存储区读取自己所需要的数据，并将模块运行所得的结果同样保存在存储区中，这样的设计使得存储区成为联系各模块传递数据的媒介，不仅降低了各模块之间因为数据传输造成的耦合性，也使得各模块相互之间更加独立，不需要依赖其他模块的运行与否，直接从存储区读取信息即可实现本模块的单独运行，增强了模块的内聚性。

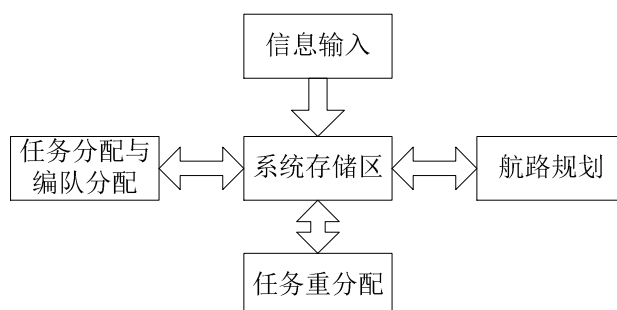


图 7.2 各模块与系统存储区关系图

## 7.2.2 各模块介绍

任务规划软件的各模块介绍如下：

### (1) 信息输入模块

信息输入分为两种方式：手动输入与文件载入。手动输入模块需要人为手动输入信息，文件载入方式则可以载入计算机中先前所保存的信息。所需要输入的内容主要为：任务点信息，任务点内目标的信息，地图中的威胁区域输入，我方无人机信息以及一些辅助信息的输入，具体所需输入内容如图 7.3 所示。

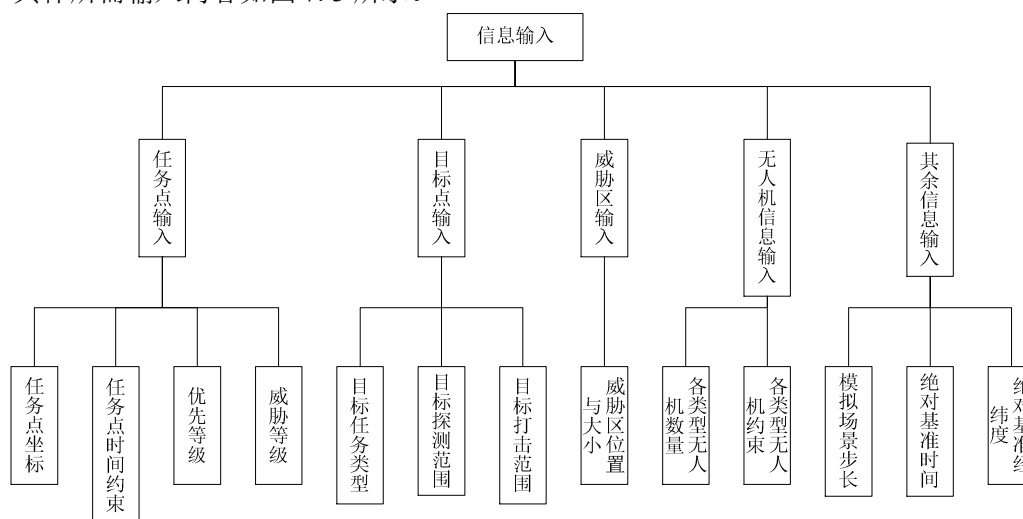


图 7.3 信息输入内容

### (2) 任务分配与编队分配模块

在模块读取任务分配所需的相关信息后，依照三种不同侧重下任务分配目标进行任务分配，并获得三种任务分配结果。任务分配中需要排除因开始执行时间设置的过早无法及时赶到的任务点和因编队航程限制而无法到达的任务点，对剩余的任务点进行任务分配和编队的分配，具体逻辑如图 7.4。

### (3) 任务分配评估模块

任务分配中有着不同侧重目标函数的选择，本模块对于所得到的不同任务分配结果进行评估，主要评估以下几个方面：

#### 1) 目标覆盖程度

目标覆盖程度为得到分配的任务点中所有目标点数量与所有的任务点中的目标数量的比值。

#### 2) 完成任务的收益

主要与得到分配的任务点本身的价值以及被编队成功执行的概率有关。

#### 3) 完成任务的代价

通常与燃油消耗，无人机弹药消耗，以及飞行过程中遇到的威胁区域等有关。

获得上述三个方面的评估值后，综合评价采用模糊多属性决策法，将三个方面的评估值进行综合评价，并给出分配评估结果。

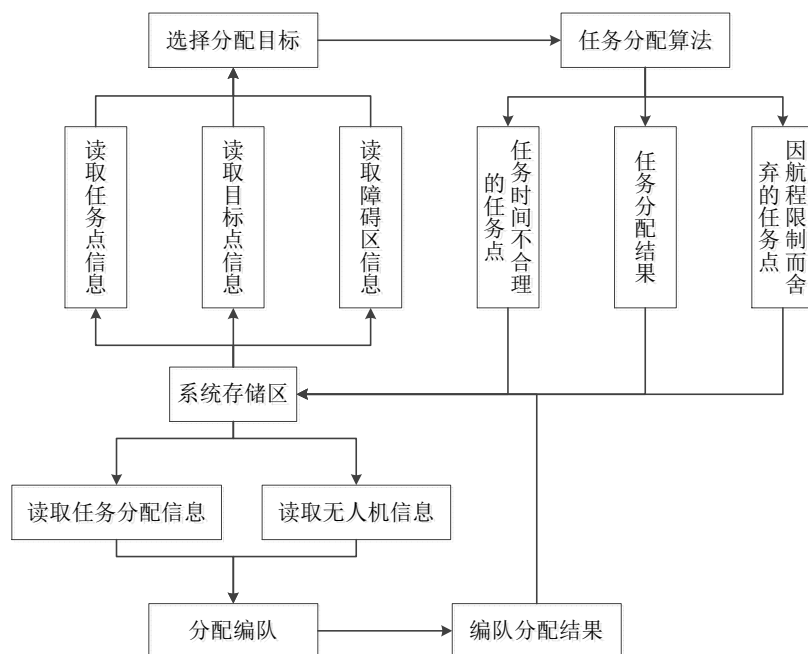


图 7.4 任务分配与编队分配模块工作流程

#### (4) 航路规划与任务过程模拟模块

航路规划部分使用了第四章所设计的 A\*算法进行航路规划，规划完成后将航路结果存储至系统存储区。在得到航路规划轨迹后，各编队在界面中开始进行任务过程的模拟展示，模拟展示利用现实中的 1 秒时间表示仿真模拟中的一个步长时间，如 5 分钟，并且每秒刷新一次界面显示，此时需要及时更新各编队的位置以及任务状态。

在实际的无人机飞行中，需要对无人机航路规划结果进行平滑处理，使其更符合无人机实际飞行。

#### (5) 重分配模块

上述的任务模拟过程中可以实现模拟过程的暂停，此时进行重分配的信息输入与重分配执行，模拟的场景将根据重分配结果进行调整，并继续任务过程的模拟。重分配完成后任务分配结果与航路规划结果会被改变，需要更新系统存储区相应的值。

#### (6) 航路评估模块

在任务过程模拟结束后，对存储区中航路规划结果进行评估。任务航程是重要评估指标，无人机完成任务所经过的航程直接决定了无人机执行任务的效率，同时，无人机在威胁区域飞行时间越长，意味着被击毁的概率越大。因此，航路评估模块从任务航程以及威胁区域飞

行时间对规划的航路进行评估。

### 7.3 协同任务规划系统仿真软件设计

图 7.5 为软件启动后的主界面，界面主要由以下几个区域组成：

图 7.5 中区域 1 为信息输入以及仿真进度控制按键，点击“手动输入”按钮后，将弹出图 7.6 所示信息输入界面，在此界面可以手动依次输入各信息。点击“载入文件”按钮，可以直接读取计算机中的已有文件作为信息的输入。仿真采用一键运行的模式，即完成信息输入后，点击“仿真开始”按钮，系统开始自动执行任务分配、分配评估、航路规划与动态演示。“仿真暂停”按钮用于动态演示过程中的重分配，点击此按钮后，演示过程将暂停，此时输入重分配信息。“仿真复位”按钮用于初始化当前软件，使仿真可以重新执行。

区域 2 为输入信息的展示界面，用于显示输入信息的内容，包括任务点位置，任务点执行时间约束，任务点内目标信息，我方无人机信息以及威胁区域信息，方便仿真过程中进行查看。

区域 3 为功能选项卡，每一个选项卡对应一个特定的模块输入与结果展示，选项卡中主要的选项为：作战信息、任务分配结果、分配评估结果、航路模块结果、任务重规划页等。

区域 4 用于显示软件此时工作状态与仿真进度。

区域 5 为 TeeChart 画图控件，主要用于航路规划显示与任务过程的模拟展示。

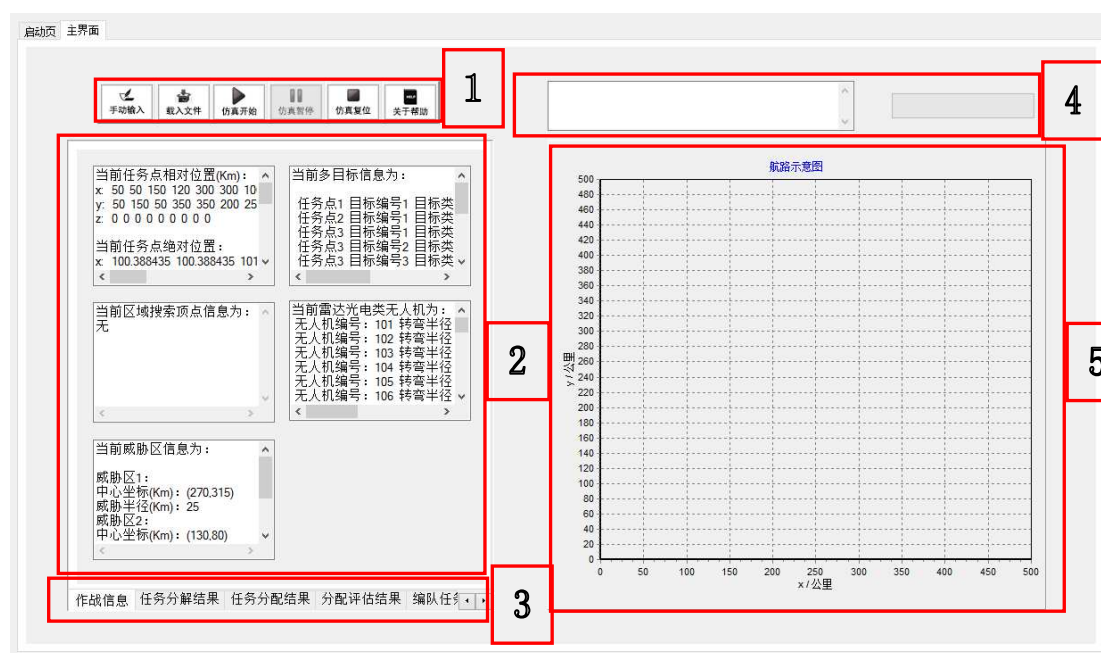


图 7.5 仿真软件主界面

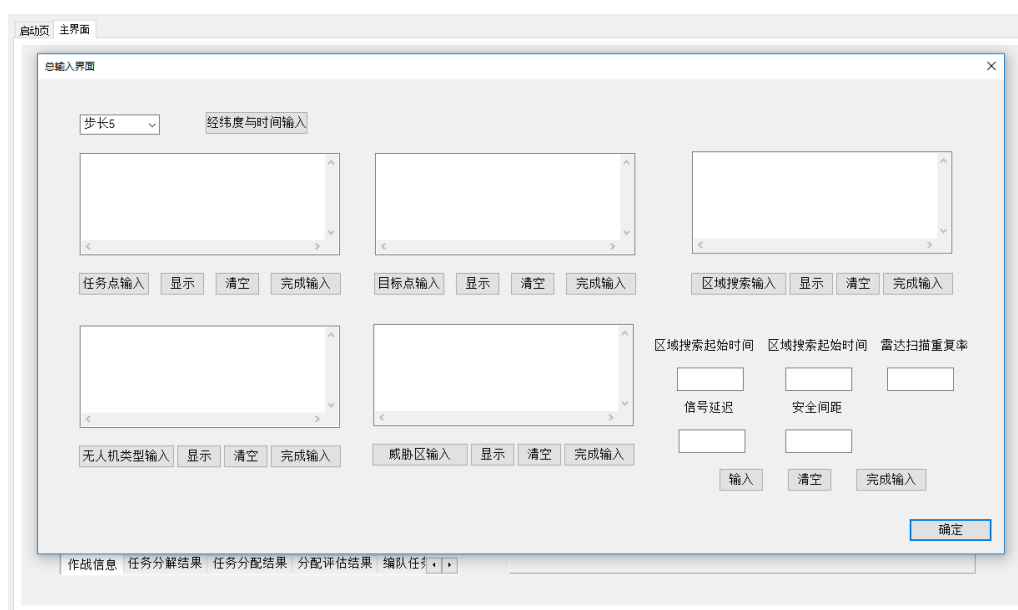


图 7.6 信息输入界面

现利用仿真软件对第三章中 3.4 节的任务场景进行仿真，仿真的步骤以及效果说明如下：

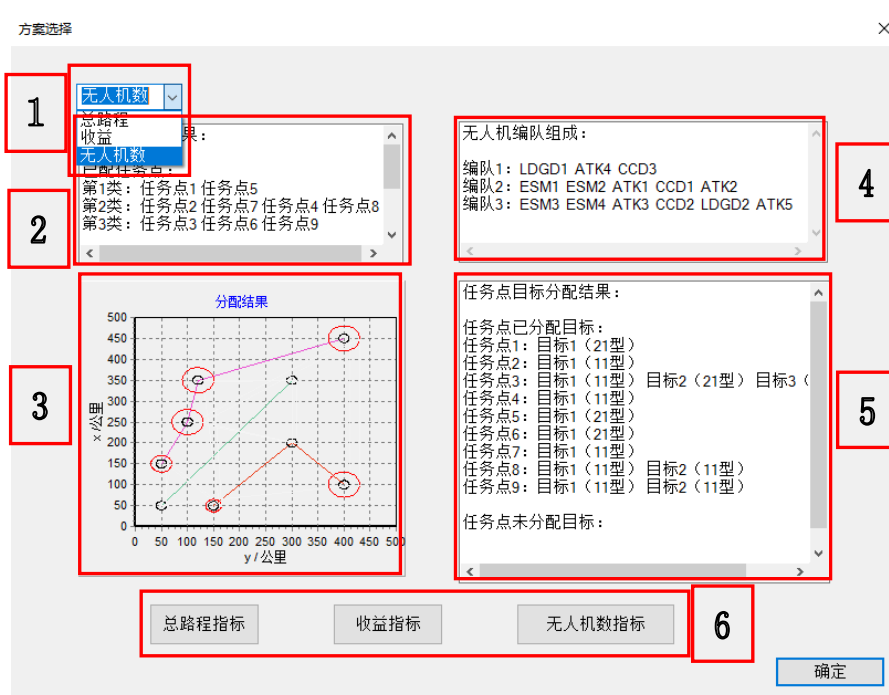


图 7.7 任务分配结果方案选择界面

参照 3.4 节的任务场景，从图 7.6 中的信息输入界面输入任务场景的信息。点击“仿真开始”按钮，首先进行任务分配，任务分配完成后，将显示图 7.7 所示的方案选择界面，可以在图 7.7 中区域 1 的下拉框选择不同的分配结果，界面中显示的内容为：区域 2 显示任务点分配的结果，将一共的 9 个任务点分配至 3 个编队中。区域 4 则显示了每个编队的编队组

成分, 其中 LDGD 代表侦察探测类无人机, ESM 代表电子战类无人机, ATK 代表攻击类无人机, CCD 代表毁伤评估类无人机。区域 5 为任务点内各目标被覆盖的情况。区域 3 中的显示控件中, 将划分到同一编队的任务点进行连线, 更加形象的显示分配结果。区域 6 为分配结果选择按钮, 在参考完所有的任务分配结果后, 选择一种分配方式, 再点击右下角“确定”按钮, 将开始航路规划, 如图 7.8 所示。

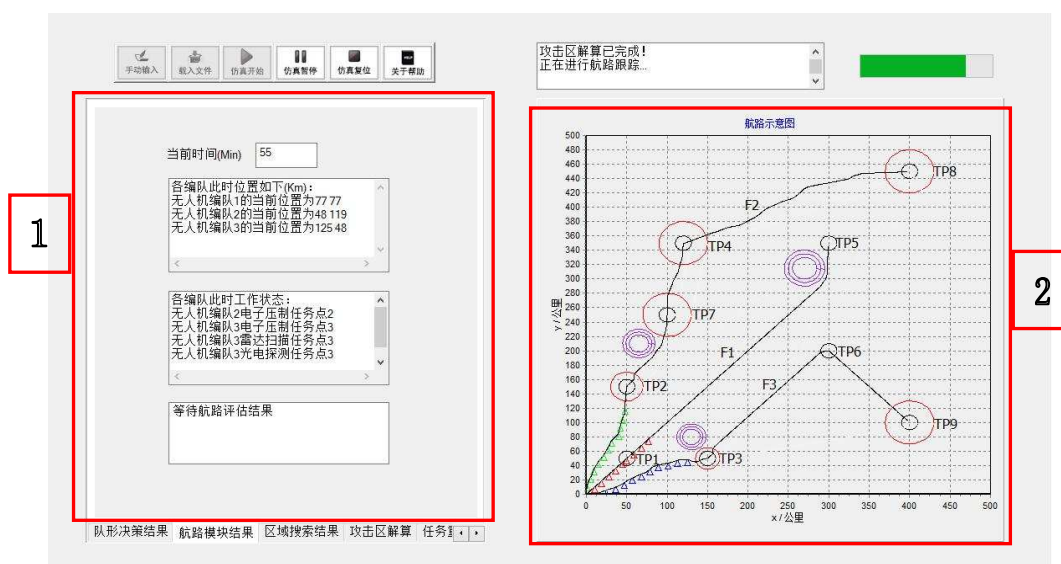


图 7.8 航路规划与任务过程模拟显示界面

图 7.8 显示的是航路规划与任务过程的模拟。其中区域 2 为航路显示图, 航路规划的结果在航路显示区中已显示, 同心圆为地图中的威胁区域, 任务点位置在图中已标识, 每个任务点内部的圈用于标识此任务点在地图中的位置, 外围的圈为其雷达或攻击的最大范围, 编队代号  $F_1$ 、 $F_2$ 、 $F_3$  分别指出了每个编队的航路, 航路上的三角分别代表各编队此时已经过的航路。区域 1 为航路模块结果选项卡, 页面上显示当前任务时间以及各编队此时的位置与工作状态。

仿真软件中图 7.8 的界面显示的是一个任务动态执行的过程, 各编队位置和工作状态随着当前任务时间的增加而随之变化。若某时刻需要重分配, 如新增任务点时, 点击“仿真暂停”, 切换至图 7.9 任务重规划选项卡, 输入新增任务点及其目标信息后, 点击“任务新增输入”按钮, 新增的任务点得到了分配, 图中新增了一个任务点  $TP_{10}$ , 此任务点被分配给了编队 1。此时再点击“仿真继续”按钮, 航路信息被重新规划, 如图 7.10 所示。

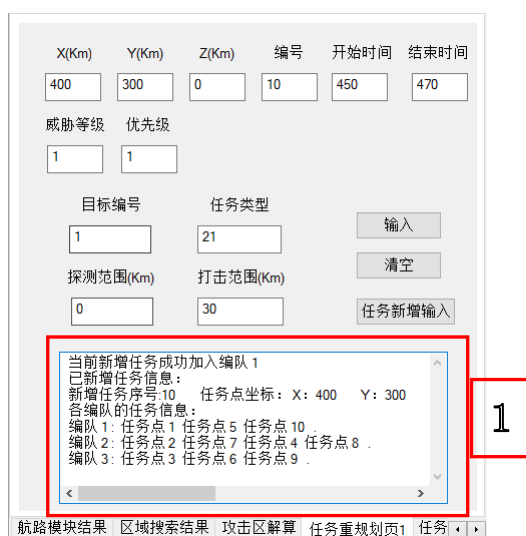


图 7.9 新增任务点重分配输入界面

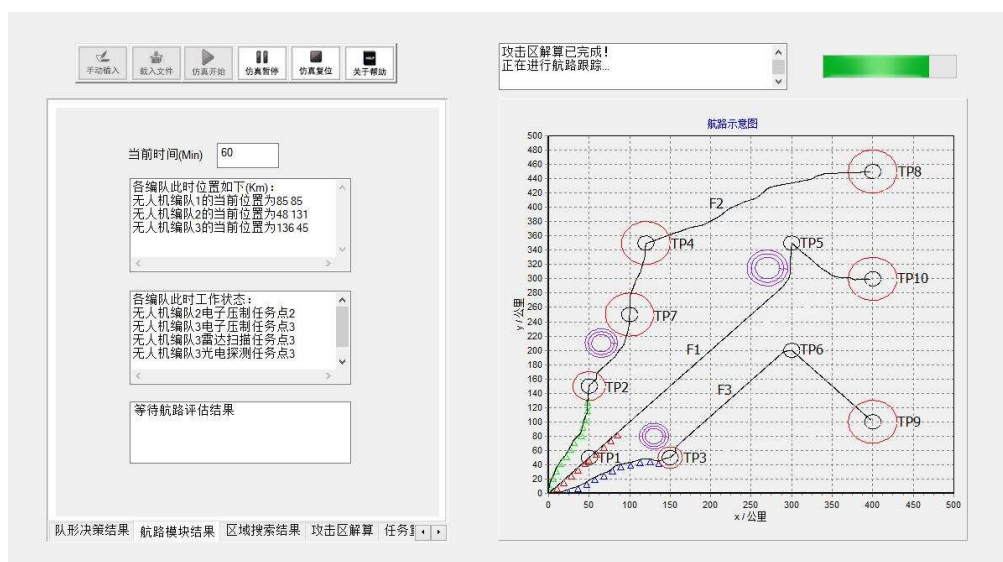


图 7.10 任务重分配后航路重规划结果

重分配还包括任务点取消，无人机故障，人为指定任务以及新增威胁区域等情况，这些重分配的输入在图 7.11 所示任务重规划页 2 选项卡中。图 7.12 为分配评估结果的显示，同样具有下拉菜单可以查看不同分配结果的评估结果。



图 7.11 任务重分配信息输入界面



图 7.12 任务分配结果评估界面

## 7.4 本章小结

本章主要在 Microsoft Visual Studio 2015 平台上利用 MFC 技术将无人机协同任务规划系统进行设计与实现，先给出了软件整体结构以及各组成模块在软件中的作用，接着展示了所设计的软件界面与效果，完成了协同任务规划系统设计。



## 第八章 总结与展望

### 8.1 本文工作总结

随着当今无人机技术的发展,军事领域中无人机协同作战已成为一个必然趋势,本文围绕无人机协同作战展开研究,主要完成了以下的工作:

(1) 提出一种无人机协同作战系统,将其划分为协同任务规划系统以及攻击决策系统,并对两者的组成进行了进一步的划分。对协同任务规划系统场景中的任务点约束与地图约束进行建模,同时设计了一种航路预估方法。

(2) 设计了协同任务规划系统中的任务分配模块。在上一步的场景中的约束以及航路预估的基础上,使用遗传算法进行任务预分配。设计的过程中,首先定义了目标函数,染色体编码方式参照了静态链表的结构,并对染色体,设计了独特的交叉、变异过程,完成任务的分配。

(3) 设计了协同任务规划系统中的航路规划模块。简单介绍了 A\* 算法的算法内容,总结了 A\* 算法在不同启发函数下的效果,并针对 A\* 算法中的 *ClosedList* 查找速度进行改进,通过仿真证明了改进的有效性。

(4) 设计了协同任务规划系统中的任务重分配模块,分析了并总结出了作战过程中需要重分配的情况。介绍了合同网协议的内容,改进其投标过程,通过对所提出的 5 种重分配情况的仿真,展示出在不同情况下系统的重分配过程与效果,说明了改进合同网协议在任务重分配中的有效性。

(5) 设计了非完备信息下的目标意图识别模块与空战决策模块。第一部分给出了与非完备相关的区间数和联系数的相关定义,以及粗糙集的相关定义,详细介绍了非完备区间数决策信息系统中的相似关系。建立了空战场景下的五种威胁因子,并以此建立非完备区间数决策信息表,进行空战规则的获取。提出利用联系数贴近度的方式进行意图的识别,通过仿真进行效果的展示。第二部分结合态势优势、敌方意图、空战能力优势三个方面,建立了用于攻击分配的评价矩阵,给出了平衡分配下的数学模型,对匈牙利算法进行改进,使其可以应用在基于存在信息非完备的评价矩阵的攻击决策中,同时也给出了改进匈牙利算法在非平衡问题下的分配方法,通过仿真证明了改进的有效性。

(6) 设计了协同任务规划系统仿真软件。基于 Visual Studio 平台,使用 MFC 搭建界面,实现了任务分配、航路规划、任务重分配等相关功能,对所设计的软件界面与效果进行了展示。

## 8.2 不足与展望

无人机在军事中的应用是一个十分复杂的实际问题，本文虽然对无人机协同作战进行了一定的研究，但是由于时间和精力限制，尚有许多内容需要继续研究，同时本文也存在一定的不足之处：

(1) 无人机协同任务规划系统中假设的场景中，虽然考虑了一定的约束条件，但是和实际的场景仍有着非常大的差距，需要进一步总结实际中出现的约束，使得场景更贴近现实，提高协同任务规划系统的实用性；

(2) 本文假设编队内无人机在执行任务时飞行高度统一，实际中的编队内无人机可能工作在不同的高度，需要为每架无人机进行单独的航路规划，同时，无人机在飞行时高度也可以变化，下一步需要将二维的航路规划拓展到三维的航路规划；

(3) 非完备信息下的目标意图识别与攻击决策的研究十分具有现实价值，本文提出的方法对进一步的研究具有一定的参考意义，但是相关的理论目前尚不成熟，还需要进行更进一步的深入研究。

## 参考文献

- [1] Slater G L. Cooperation between UAVs in a search and destroy mission. AIAA Guidance, Navigation, and Control Conference and Exhibit, AIAA. 2003:11-14.
- [2] Chandler P R. Cooperative control of a team of UAVs for tactical missions. Proceedings of the AIAA 1st Intelligent Systems Technical Conference. 2004:20-22.
- [3] Subbarao K, Muralidhar P. Cooperative control of UAVs in dynamic pursuit subject to communication delays. Proceedings of the AIAA Guidance, Navigation, and Control Conference. 2005: 15-18.
- [4] 嵇亮亮. 无人机的导引及协同编队飞行控制技术研究, [硕士学位论文]. 南京航空航天大学, 2008.
- [5] 蓝伟华, 喻蓉. 多机编队协同空战的概念及关键技术. 电光与控制, 2005, 12(6): 12-15.
- [6] 王建华. 炮兵防空兵侦察技术导论. 北京: 国防工业出版社, 2006.
- [7] Rasmussen S J, Chandler P R. MultiUAV: a multiple UAV simulation for investigation of cooperative control. Conference on Winter Simulation: Exploring New Frontiers. 2002: 869-877.
- [8] 刘晓宁, 崔彬, 尹勇, 等. 空空多目标截获与攻击综合半实物仿真系统的设计与研制. 测控技术, 2007, 26(1): 79-81.
- [9] 余振翔. 基于模糊动态贝叶斯网络的空战敌方作战企图识别方法, [硕士学位论文]. 合肥工业大学, 2013.
- [10] 刘同明. 数据挖掘技术及其应用. 北京: 国防工业出版社, 2001.
- [11] 张在美. 一种基于粗糙集的不完备信息处理方法研究, [硕士学位论文]. 湖南大学, 2007.
- [12] 吴国强. 无人机航迹跟踪控制方法研究, [硕士学位论文]. 哈尔滨工程大学, 2010.
- [13] 邹湘伏, 何清华, 贺继林. 无人机发展现状及相关技术. 飞航导弹, 2006, No.10: 9-14.
- [14] Werrell K P. The evolution of the cruise missile. Washington: Air University Press, 1985.
- [15] Shima T, Rasmussen S. UAV cooperative decision and control: challenges and practical approaches. Philadelphia: Society for Industrial and Applied Mathematics, 2008.
- [16] Wise K A. Unmanned aircraft: the future in military aviation. American Control Conference, 2004, 37(13): 46-57.
- [17] 张立珍. 无人机自主飞行控制系统的设计, [硕士学位论文]. 南京航空航天大学, 2011.
- [18] 李伟, 梁兵兵, 石敏超. 直接作战无人机的分类与发展. 飞航导弹, 2013, No.12:

27-31+35.

- [19] 张有. 空中“敢死队员”——以色列“哈洛普”自杀式无人机. 坦克装甲车辆, 2016, No.2: 40-43.
- [20] 冯云松, 张俊. 国外无人侦察机的现状及其发展. 航空科学技术, 2005, No.2: 24-27.
- [21] 包强, 姜为学, 刘小松, 张伍. 全球鹰无人机导航系统分析. 飞航导弹, 2009 No.11: 60-63.
- [22] 安永旺, 杨继何, 王倩. 美军无人侦察机发展. 国防科技, 2009, 30(2): 81-84.
- [23] 李会明. 无人机系统协同电子对抗模型及效果评估技术研究, [硕士学位论文]. 吉林大学, 2015.
- [24] 沈文亮, 张卓鸿. 无人机在电子对抗中的应用研究. 舰船电子对抗, 2013, 36(6): 14-18.
- [25] 任翔宇, 刘丽, 马燕. 美军电子攻击型无人机的发展. 航天电子对抗, 2014, 30(6): 53-56.
- [26] 沈亮, 欧阳平. 捕食者系列无人机特点及发展经验. 飞航导弹, 2012, No.12: 33-36.
- [27] 何凡, 吴文海, 曲建岭. 美国海军无人机系统的发展现状、趋势及关键技术. 飞机设计, 2007, 27(1): 62-64+77.
- [28] 沈娟, 王蒙. 美国的多无人机自主协同作战项目. 飞航导弹, 2008, No. 8: 39-42.
- [29] Ollero A, Lacroix S, Merino L, et al. Multiple eyes in the skies: architecture and perception issues in the COMETS unmanned air vehicles project. Robotics & Automation Magazine IEEE, 2005, 12(2): 46-57.
- [30] 李小磊. 无人机任务规划软件设计与实现, [硕士学位论文]. 电子科技大学, 2014.
- [31] Darrah M, Niland W, Stolarik B. Multiple UAV dynamic task allocation using mixed integer linear programming in a SEAD mission. Infotech, 2006: 26-29.
- [32] Schumacher C, Chandler P, Pachter M, et al. Constrained optimization for UAV task assignment. AIAA Guidance, Navigation, and Control Conference and Exhibit. 2006:354-362.
- [33] Edison E, Shima T. Integrated task assignment and path optimization for cooperating uninhabited aerial vehicles using genetic algorithms. Computers & Operations Research, 2011, 38(1): 340-356.
- [34] Shima T, Rasmussen S J, Sparks A G, et al. Multiple task assignments for cooperating uninhabited aerial vehicles using genetic algorithms. Computers & Operations Research, 2006, 33(11): 3252-3269.
- [35] Gerkey B P, Matarić M J. Murdoch: publish/subscribe task allocation for heterogeneous agents. Proceedings of Autonomous Agents, 2000: 203-204.
- [36] Sujit P B, Sinha A, Ghose D. Multiple UAV task allocation using negotiation. International

- Joint Conference on Autonomous Agents & Multiagent Systems. 2006: 471-478.
- [37] Nygard K E, Chandler P R, Pachter M. Dynamic network flow optimization models for air vehicle resource allocation. American Control Conference, 2001, No.3: 1853-1858.
- [38] Sujit P B, Sinha A, Ghose D. Multi-UAV task allocation using team theory. Decision and Control, 2005 and 2005 European Control Conference. 2006: 1497-1502.
- [39] 龙国庆, 祝小平, 董世友. 基于聚类算法的多无人机系统任务分配(英文). 火力与指挥控制, 2011, 36(12): 54-59.
- [40] 韩攀. 多无人机协同任务规划技术, [硕士学位论文]. 南京航空航天大学, 2013.
- [41] 程聪. 无人机协同作战任务分配与攻击效能评估技术, [硕士学位论文]. 南京航空航天大学, 2013.
- [42] 邓启波. 多无人机协同任务规划技术研究, [博士学位论文]. 北京理工大学, 2014.
- [43] 国博, 王社伟, 陶军. 基于改进粒子群算法的多无人机任务分配研究. 计算机仿真, 2009, 26(7): 62-64+153.
- [44] 苏菲, 陈岩, 沈林成. 基于蚁群算法的无人机协同多任务分配. 航空学报, 2008, S1: 184-191.
- [45] Tisdale J, Kim Z, Hedrick J K. Autonomous UAV path planning and estimation. IEEE Robotics & Automation Magazine, 2009, 16(2): 35-42.
- [46] 孙炜, 吕云峰, 唐宏伟, 等. 基于一种改进 A\*算法的移动机器人路径规划. 湖南大学学报(自然科学版), 2017, 44(4): 94-101.
- [47] Tisdale J, Durrant-Whyte H, Hedrick J K. Path planning for cooperative sensing using unmanned vehicles. ASME 2007 International Mechanical Engineering Congress and Exposition. 2007: 715-723.
- [48] 王光磊, 魏巍, 丁轶. 运输机航路规划算法. 指挥信息系统与技术, 2016, 7(2): 43-46.
- [49] 穆中林, 鲁艺, 任波, 等. 基于改进 A\*算法的无人机航路规划方法研究. 弹箭与制导学报, 2007, 27(1): 297-300.
- [50] Guo-Hui H E, Chen J Q. Research on algorithm of intelligent path finding in game development. Computer Engineering & Design, 2006, 27(13): 2334-2337.
- [51] Bonabeau E, Dorigo M, Theraulaz G. Swarm intelligence: From natural to artificial systems. New York: Oxford University Press, 1999.
- [52] 徐啟云, 王洁, 郝文渊, 等. 国外无人战斗机发展历程和趋势. 飞航导弹, 2016, No.3: 28-32.
- [53] 姚庆锴, 柳少军, 贺筱媛, 等. 战场目标作战意图识别问题研究与展望. 指挥与控制学

- 报, 2017, 3(2): 127-131.
- [54] 欧微, 柳少军, 贺筱媛, 等. 基于时序特征编码的目标战术意图识别算法. 指挥控制与仿真, 2016, 38(6): 36-41.
- [55] Austin F, Carbone G, Hinz H, et al. Game theory for automated maneuvering during air-to-air combat. *Journal of Guidance Control & Dynamics*, 1990, 13(6): 1143-1149.
- [56] Rosenberger J M, Hwang H S, Pallerla R P, et al. The generalized weapon target assignment problem. Texas Univ at Arlington, 2005.
- [57] Liao M, Chen Z. Coordinated target assignment in multi-UAV based on satisficing decision theory. *Journal of Beijing University of Aeronautics & Astronautics*, 2007, 33(1): 81-85.
- [58] 黄捷, 陈谋, 姜长生. 无人机空对地多目标攻击的满意分配决策技术. 电光与控制, 2014, 21(7): 10-13+30.
- [59] 孟光磊, 罗元强, 梁宵, 徐一民. 基于动态贝叶斯网络的空战决策方法. 指挥控制与仿真, 2017, 39(3): 49-54.
- [60] 黄家成, 谢奇峰. 基于遗传算法的协同多目标攻击空战决策方法. 火力与指挥控制, 2004(01): 49-52.
- [61] 朱艳萍, 姚敏, 赵敏. 敌对环境多无人机协同空战决策. 火力与指挥控制, 2013, 38(3): 12-16.
- [62] 陈侠, 魏晓明, 徐光延. 多无人机模糊态势的分布式协同空战决策. 上海交通大学学报, 2014, 48(7): 907-913+921.
- [63] 陈侠, 刘敏, 胡永新. 基于不确定信息的无人机攻防博弈策略研究. 兵工学报, 2012, 33(12): 1510-1515.
- [64] 陈侠, 唐婷. 不确定环境下多无人机动态任务分配方法. 火力与指挥控制, 2013, 38(1): 45-49+53.
- [65] 陈侠, 胡永新. 不确定环境下基于 PSO 算法的多无人机任务分配方法. 兵工自动化, 2013, 32(6): 11-16.
- [66] 杨尚君, 王社伟, 陶军, 等. 动态环境下的多 UCAV 协同任务分配研究. 电光与控制, 2012, 19(7): 32-36+79.
- [67] 缪永飞. 军用飞行器航迹规划与显示技术研究, [硕士学位论文]. 国防科学技术大学, 2006.
- [68] Xia P, Chen M, Zou J, et al. Prediction of air target intention utilizing incomplete information. *Proceedings of 2016 Chinese Intelligent Systems Conference*, 2016, No.1: 395-403.
- [69] 童奇, 李建勋, 童中翔, 等. 基于机动识别的空战意图威胁建模与仿真. 现代防御技术,

2014, 42(4): 174-184.

- [70] 朱艳萍. 多无人机协同攻击策略研究, [硕士学位论文]. 南京航空航天大学, 2012.
- [71] 张永芳, 张安, 张志禹, 等. 战术飞行路径规划算法. 交通运输工程学报, 2006, 6(4): 84-87.
- [72] 吴明珠, 史婷婷. 解决旅行商问题的改进算法. 微计算机信息, 2011, 27(9): 189-191.
- [73] 姜长生, 王从庆, 魏海坤, 等. 智能控制与应用. 北京: 科学出版社, 2007.
- [74] 严蔚敏, 吴伟民. 数据结构. 北京: 清华大学出版社, 2006.
- [75] 周小镜. 基于改进 A\*算法的游戏地图寻径的研究, [硕士学位论文]. 西南大学, 2011.
- [76] 李志建, 郑新奇, 王淑晴, 等. 改进 A~\*算法及其在 GIS 路径搜索中的应用. 系统仿真学报, 2009, 21(10): 3116-3119.
- [77] 王淼弛. 基于 A\*算法的移动机器人路径规划, [硕士学位论文]. 沈阳工业大学, 2017.
- [78] 辛煜, 梁华为, 杜明博, 等. 一种可搜索无限个邻域的改进 A\*算法. 机器人, 2014, 36(05): 627-633.
- [79] Smith R G. The contract net protocol: high-level communication and control in a distributed problem solver. IEEE Computer Society, 1980.
- [80] 王昊冉. 基于多实体贝叶斯网络的空中目标意图识别方法研究, [硕士学位论文]. 国防科学技术大学, 2011.
- [81] 周志强, 钱建刚, 尹康银, 等. 一种基于 DS 证据理论的目标意图预测方法. 空军预警学院学报, 2014, 28(2): 116-118.
- [82] 梁美莲. 不完备信息系统中数据挖掘的粗糙集方法, [硕士学位论文]. 广西大学, 2005.
- [83] Qu M, SuihuaiYu, Chen D, et al. An improved rough set approach to evaluate auto-mobility systems. Journal of Intelligent & Fuzzy Systems, 2017, 33(5): 1-15.
- [84] Yang Q, Du P A, Wang Y, et al. Developing a rough set based approach for group decision making based on determining weights of decision makers with interval numbers. Operational Research, 2017, No.5: 1-23.
- [85] Azar A T, Inbarani H H, Devi K R. Improved dominance rough set-based classification system. Neural Computing & Applications, 2017, 28(8): 2231-2246.
- [86] 柳毅, 佟明安. 匈牙利算法在多目标分配中的应用. 火力与指挥控制, 2002, 27(4): 34-37.
- [87] 刘秀梅, 赵克勤. 区间数决策集对分析. 北京: 科学出版社, 2014.
- [88] 张涛, 周中良, 苟新禹, 等. 基于信息熵和 TOPSIS 法的目标威胁评估及排序. 电光与控制, 2012, 19(11): 35-38.
- [89] 刘秀梅, 赵克勤. 基于联系数的不确定空情意图识别. 智能系统学报, 2012, 7(5):

450-456.

- [90] 刘祥锐. 不完备区间值信息系统的知识约简与规则提取, [硕士学位论文]. 青岛理工大学, 2013.
- [91] Shima T, Rasmussen S, Gross D. Assigning micro UAVs to task tours in an urban terrain. *IEEE Transactions on Control Systems Technology*, 2007, 15(4): 601-612.
- [92] Brumitt B, Stentz A, Hebert M, et al. Autonomous driving with concurrent goals and multiple vehicles: mission planning and architecture. *Autonomous Robots*, 2001, 11(2): 103-115.
- [93] Eun Y, Bang H. Cooperative Task assignment/path planning of multiple unmanned aerial vehicles using genetic algorithm. *Journal of Aircraft*, 2009, 46(1): 338-343.
- [94] Juhasz Z, Paul P. Scalability analysis of the contract net protocol. *IEEE/ACM International Symposium on CLUSTER Computing and the Grid*. IEEE Computer Society, 2002: 346.
- [95] Parunak H V, Purcell M, O'Connell R. Digital pheromones for autonomous coordination of swarming UAV's. *Uav Conference*. 2002: 2002-3446.
- [96] Choi J, Oh S, Horowitz R. Distributed learning and cooperative control for multi-agent systems. *Automatica*, 2009, 45(12): 2802-2814.



## 致 谢

研究生阶段的学习和生活转眼就要接近尾声，回首两年多的时光，颇有感慨。我要感谢两年多来老师们对我知识的传授、同学们对我无私的帮助，感谢这段时间所有的经历对我的磨练，使我能自信而坚强地面对以后的工作和生活。

首先，我必须要感谢我的导师陈谋教授，感谢他在学习及生活中给与我的指引与帮助，从我的论文选题、开题、撰写、到论文的定稿，这一路走来，每一步都离不开老师的谆谆教诲和悉心关怀。陈老师治学严谨、知识渊博、谦虚平和，所讲授的课程，是我研究生生涯宝贵的知识收获。在这两年多的时间里，陈老师教会我的不仅是做学问的态度，更重要的是为人处事的方式，我将带着这些在接下来的工作和生活中，继续努力，不断锤炼自己、超越自己，让自己变得更加优秀。感谢吴庆宪教授在组会讨论中所提的建议，这些宝贵的建议在我的学术发展中有着重要的指引作用。

感谢402实验室的师兄师姐们，你们所营造的浓厚的学术氛围，对我的学术科研起到了非常积极的影响。感谢瞿友杰、孙诚、高欢、张鹏、米培超等同届的同学，衷心祝愿各位在以后的工作和生活中一切顺利，感谢胡鲲、丁晟辉、李轶锟、陈文辉、郑文超等师弟，和你们在一起度过了这段值得回忆的时光。

感谢我的家人为我作出的无私的支持与帮助，你们对我的信任是我前进的动力。

最后，我还要向在百忙之中评阅论文和参加答辩的各位专家、教授致以真诚的谢意！

夏鹏程

二零一八年一月于南航

## 在学期间的研究成果及发表的学术论文

### 攻读硕士期间发表(录用)论文情况

1. Xia P, Chen M, Zou J, et al. Prediction of Air Target Intention Utilizing Incomplete Information. Proceedings of 2016 Chinese Intelligent Systems Conference. Xiameng, China, 2016: 395-403.(EI 收录)

### 攻读硕士学位期间参加科研项目情况

1. 参加中航工业无线电研究所项目“无人机任务分配技术”。