

分类号：TP391
学号：21208013

单位代码：11510
密级：



山东交通学院

专业学位硕士学位论文

(2024 届)

基于群体智能算法的多无人机协同运行规划算法研究
Research on Multi-UAV Collaborative Operation Planning
Algorithm Based on Swarm Intelligence Algorithm

论文作者：	何宇来
专业学位类别：	机械
专业领域：	机器人工程
指导教师：	武华 李艳秀 李鹏
所在学院：	信息科学与电气工程学院 (人工智能学院)

2024 年 5 月 30 日

分 类 号 : TP391

单 位 代 码 : 11510

学 号 : 21208013

密 级 :

山东交通学院

专业学位硕士学位论文

基于群体智能算法的多无人机协同运行规划算法研究

Research on Multi-UAV Collaborative Operation Planning
Algorithm Based on Swarm Intelligence Algorithm

论 文 作 者 : 何宇来

专业学位类别 : 机械

专 业 领 域 : 机器人工程

指 导 教 师 : 武华 李艳秀 李鹏

所 在 学 院 : 信息科学与电气工程学院
(人工智能学院)

2024 年 5 月 30 日

目 录

1 绪论.....	1
1.1 研究背景及意义.....	1
1.2 国内外研究现状.....	3
1.2.1 无人机任务分配研究现状.....	3
1.2.2 群体智能算法研究现状.....	4
1.2.3 其他算法研究现状.....	7
1.2.4 研究现状总结.....	8
1.3 论文主要工作及章节安排	9
1.3.1 论文主要工作.....	9
1.3.2 章节安排.....	10
2 考虑动态优先级的多无人机协同任务分配模型	12
2.1 多无人机协同任务分配问题分析	12
2.1.1 任务分配算法分析.....	12
2.1.2 任务分配模型分析.....	14
2.2 动态优先级问题模型描述	15
2.2.1 动态优先级问题模型.....	15
2.2.2 目标函数设定.....	17
2.3 本章小结.....	19
3 动态优先级确定性变化下多无人机任务分配算法研究	21
3.1 动态优先级确定性变化下问题描述	21
3.2 贪婪算法应用设计.....	22
3.2.1 贪婪算法概述.....	22
3.2.2 贪婪算法在本问题中的应用设计.....	23
3.3 改进单亲遗传-贪婪结合算法设计	25
3.3.1 遗传算法概述.....	25
3.3.2 改进单亲遗传-贪婪结合算法在本问题中的应用设计	28
3.4 仿真实验与结果分析.....	31
3.4.1 关键参数分析实验.....	32
3.4.2 算法对比实验.....	35
3.5 本章小结.....	39
4 动态优先级不确定性变化下多无人机任务分配算法研究	40
4.1 不确定性问题分析.....	40
4.2 动态优先级不确定性变化下问题描述	42
4.3 基于一致性的捆绑拍卖算法应用设计	43

4.3.1 基于一致性的捆绑拍卖算法概述.....	43
4.3.2 基于一致性的捆绑拍卖算法在本问题中的应用设计	44
4.4 自适应搜索模式切换粒子群算法设计	46
4.4.1 粒子群算法概述.....	46
4.4.2 自适应搜索模式切换粒子群算法在本问题中的应用设计 ..	47
4.5 仿真实验与结果分析.....	53
4.5.1 关键参数分析实验.....	54
4.5.2 算法对比实验.....	56
4.6 本章小结.....	60
5 总结与展望.....	61
5.1 本文工作总结.....	61
5.2 未来工作展望.....	62
参考文献.....	63

摘 要

随着无人机在诸多领域中得到了广泛应用,与无人机相关的无人机协同运行规划技术不断发展,已经成为国内外研究的热点问题。多个无人机组成集群可以完成更复杂的任务,通过合理的协同控制达到整体效能最优,在实际场景中展现出它的特点和优势。一直以来,在特殊环境或应急条件下的多无人机协同任务分配问题是一个广泛研究的经典问题,已有相关研究或方法普遍对多无人机任务分配过程设置固定的任务优先级(任务序列)。但是,在现实无人机执行任务的过程中,可能会出现由于环境的变化,人为因素等导致任务优先级的变化(比如气温变低的地方对物资的急需程度变高,突发疫情状态下对药品等物资的急需程度变高等)的情况,此时原有的规划方法会变得不适用。所以,本文面向这些场景,基于群体智能算法对多无人机协同运行规划算法进行了深入研究。

首先,本文建立了一种在特殊环境下任务优先级发生动态变化的多无人机协同任务分配问题模型。为了增强本文所建模型的可扩展性和普适性,本文分别研究了确定动态优先级变化情况下的多无人机任务分配问题和不确定动态优先级变化情况下的多无人机协同任务分配问题。

其次,本文构建了动态优先级确定性变化情况下的多无人机协同任务分配问题模型,以安全库存模型作为参考生成需求状态变化函数,在考虑目标任务优先级的基础上把总奖励值最大化作为任务的主要目标。本文对遗传算法进行改进,提出了改进单亲遗传-贪婪结合算法。本文用贪婪算法,改进单亲遗传算法和改进单亲遗传-贪婪结合算法进行了对比仿真实验,验证了改进单亲遗传-贪婪结合算法在处理动态优先级确定性变化情况下的多无人机协同任务分配问题的有效性,提高了解决问题的效率。

最后,本文构建了动态优先级不确定性变化情况下的多无人机协同任务分配问题模型,用泊松分布来生成不确定性变化情况下需求状态变化函数。本文用基于一致性的捆绑拍卖算法对不确定性变化情况下的问题进行了应用设计。本文对粒子群算法进行改进,提出自适应搜索模式切换粒子群算法,并用基于一致性的捆绑拍卖算法和离散粒子群算法作为对比算法进行了对比实验。实验验证了自适应搜索模式切换粒子群算法能够提高种群的多样性和全局搜索能力,可以更高效

稳定地解决不确定动态优先级变化情况下的多无人机协同任务分配问题。

关键词：多无人机；任务分配；任务优先级；遗传算法；粒子群算法

ABSTRACT

As UAVs are widely used in many fields, UAV collaborative operation planning technology related to UAVs continues to develop and has become a hot issue in domestic and foreign research. The Multi-UAV forming a cluster can complete more complex tasks and achieve optimal overall performance through reasonable collaborative control, showing its characteristics and advantages in actual scenarios. For a long time, the distribution of materials by UAV under special environments or emergency conditions has been a classic problem of extensive research, and relevant studies or methods have generally set fixed task priorities (task sequences) in task assignment. However, in the process of performing a mission by a real UAV, there may be changes in the priority of the mission objectives due to changes in the environment and human factors (such as a higher demand for materials in places where the temperature becomes lower, and a higher demand for medicines and other materials in a sudden epidemic situation), and the original planning method will become inapplicable. Therefore, in this thesis for these scenarios, this thesis makes an in-depth study on the Multi-UAV collaborative operation planning algorithm based on the group intelligence algorithm.

Firstly, this thesis develop a problem model in collaborative task assignment in a special environment. In order to verify the effectiveness of the model established in this thesis, the task assignment problem of Multi-UAV under the condition of determining dynamic priority change and the cooperative task assignment problem of multiple UAV under the condition of uncertain dynamic priority change are studied respectively.

Secondly, this thesis establishes a model of Multi-UAV collaborative task assignment problem with dynamic changes in task priority in special environments. In order to enhance the scalability and universality of the model built in this thesis, this thesis studied the Multi-UAV task assignment problem in the case of determining dynamic priority change and the Multi-UAV collaborative task assignment in the case of uncertain dynamic priority change.

Then, a Multi-UAV collaborative task assignment problem model is constructed under the condition of deterministic change of dynamic priority, and the safety stock model is used as a reference to generate the demand state change function, and the total reward value is maximized as the main goal of the task on the basis of considering the

priority of the target task. In this thesis, the genetic algorithm of the swarm intelligence representative algorithm is improved, and the improved partheno genetic-greedy combination algorithm is proposed. Comparative simulation experiments were carried out with the greedy algorithm, the improved partheno genetic algorithm and the improved partheno genetic-greedy combination algorithm, which verified the effectiveness of the improved partheno genetic-greedy combination algorithm in dealing with the Multi-UAV collaborative task assignment problem under the condition of dynamic priority deterministic change, and improved the efficiency of problem solving.

Finally, a Multi-UAV collaborative task assignment problem model is constructed in the case of dynamic priority uncertainty change, and the Poisson distribution is used to generate the demand state change function in the case of dynamic uncertainty change, and the total reward value continues to maximize as the main goal of the task. In this thesis, the consensus-based bundle algorithm is used to design the problem in the change of uncertainty. This thesis improved the particle swarm algorithm, propose the adaptive search mode switching algorithm, and perform simulation experiments by using consensus-based bundle auction algorithm and discrete particle swarm algorithm as comparison algorithm. The improved algorithm can improve the population diversity and global search ability, which can more efficiently solve the Multi-UAV collaborative task assignment in the case of uncertain dynamic priority change.

Key words: Multi-UAV; Task assignment; Task priority; Genetic algorithm; Particle swarm optimization

1 绪论

1.1 研究背景及意义

无人机(Unmanned Aerial Vehicle, UAV)作为飞行器具有远程控制或自主飞行能力的特点,它主要是通过地面系统的无线电遥控和机载的计算机系统来控制飞行。无人机可以携带多种设备在海、陆、空中执行各种指定的任务,并且可以一次或多次使用。相比较于传统的有人驾驶飞机,无人机具有尺寸小、价格低、生存能力强、方便灵活、机动性强等优点^[1],在复杂的任务环境中发挥了极大地作用,使航天技术进入到了一个新颖的时代。随着无人机相关技术与研究的快速发展,无人机的种类也越来越多。最近几年多无人机协同执行任务技术发展迅速,使得对无人机的任务需求不断发生变化,同时根据具体需求环境不同,无人机在各种形态的环境扮演了不同的角色,应用前景将非常广阔。



图 1.1 无人机执行运输、农药喷洒、灭火、侦察任务示例图

Figure 1.1 Example diagram of UAV performing transportation, pesticide spraying, fire fighting and reconnaissance missions

在执行军事作战、救援侦察和物资配送等任务时,单个无人机存在承载能力低、侦察面积小和单体作战能力不足等问题,导致其在执行复杂任务时难以完成^[2]。相对于单个无人机,多无人机的集群协同系统可以做到多架单体无人机并行执行任务,通过各个无人机的协同作战,不仅可以增加侦察面积,而且可以达到一加

一大于二的高效率任务执行效果。无人机系统虽然能够弥补一些载人飞机无法完成任务的空白，但未来仍存在无人机系统任务环境越来越严峻的问题。近些年无人机产业迅速崛起，同时芯片行业的快速迭代发展也更加快了无人机的智能化进程，无人机执行的任务也变得更加复杂和多样。如图 1.1 所示，无人机系统在军事、商业和农业等多个应用场景发挥了举足轻重的作用，如军事作战任务目标侦察、对敌打击、迂回作战等^[3]；商业物资的运输、商业电影的场景、宣传视频广告的拍摄等；农业救灾抢险、环境监测、农药喷洒^[4]以及地质勘探等。无人机系统的应用还包括材料运输、测绘、电力线检查操作、后勤和无线通信等^[5]。因此，近年来多无人机的协同运行规划相关问题备受关注，成为未来无人机发展的重要方向之一。

群体智能(Swarm Intelligence, SI)是多无人机协同运行的核心技术之一，指的是在人工智能的控制下，无人机集群能够自主地完成任务，以提高无人机集群的整体作战能力^[6]。群体智能是指由相互作用的简单独立自组织个体涌现出的求解某一类问题的智能集体行为，是受自然界生物个体的自发地组织行为的启发^[7]。在多无人机协同合作运行系统的配置中，多无人机使用分布式协同分配，通过个体行为完成群体行为。群体智能研究自然界中的社会性群体行为，从独立个体的自发性组织行为至完成群体社会性行为。在生物界中，大量生物个体在集群活动的过程中总会形成一种协调有序的神奇场景^[8]，比如各种生物的群体活动：海洋中以集群为单位活动的鱼，杂乱却有序的随着洋流行进，遇到威胁的时候又尽可能散开以保证安全；鸟类在不同时期、不同季节的迁徙也是如此，大规模的鸟群集体翱翔，成千上万只鸟一起飞翔行动；蚁群、狼群为了种族的繁衍生息，需要整个集体外出觅食，往往需要分工合作来完成觅食；微观上细菌、动物的各种细胞也都存在类似的集群运动^[9]。实际上生物集群运动无处不在，无论是生命体还是非生命体所处的场景中，而这些群体都呈现出了各个个体之间的分工、协作、沟通、智能涌现的特征。在生物进化进行自然界“优胜劣汰”的自然选择过程中，生物个体的自主能力逐步增加，增加其他个体之间的信息交流、增强自身生存能力，经过时间和环境不断演变，个体的进步会促使整个生物种群展现出极强的自主智能性，协调合作能力以及对于周围环境变化的高敏感度与适应性^[10]。

多无人机协同任务分配技术是多无人机协同运行规划的基础和核心组成部分，本文将对多无人机协同任务分配技术展开重点研究。国内外学者对特殊环境或应急条件下的多无人机协同任务分配问题进行了广泛研究，已有相关研究或方

法普遍对多无人机任务分配过程设置固定的任务优先级(任务序列),却忽略了任务优先级的变化。这种做法会在实际紧急救助等状况下受到任务周期限制、新任务的无序派遣等情况下受到未知因素的干扰,导致无人机产生多余的能耗和路径规划,影响无人机任务完成程度,降低成功概率。为了解决这一问题,本文提出了一种新的多无人机协同任务分配模型,该模型考虑了在特殊环境下任务优先级的变化。通过构建这种模型,无人机可以更好地应对实际场景中出现的状况,并有效解决在特殊环境下任务优先级变化带来的问题。因此,本文对考虑动态优先级的多无人机协同任务分配问题及解决该问题所使用的群体智能算法进行全面研究,对多无人机协同运行、任务分配等相关问题的研究具有一定的意义。

1.2 国内外研究现状

随着无人机相关软件和硬件技术的迅速发展,无人机的应用急剧增加,对无人机执行任务的难度也越来越高,这更加要求了无人机系统实现智能化、高效化的应用,从而匹配人们在民用邻域和军事领域中不断增加的使用需求。多无人机协同任务分配问题作为多无人机协同运行的重要组成部分,被国内外学者们广泛研究。

1.2.1 无人机任务分配研究现状

目前,在多无人机系统中使用任务分配技术的应用,主要包括四类:移动边缘计算(Mobile Edge Computing, MEC)应用、军事应用、搜索和救援(Search And Rescue, SAR)任务应用,以及民用应用,下面具体说明每类应用。

(1) 移动边缘计算应用

近几年来,无人机在 MEC 领域得到了广泛的研究,无人机可以在基站覆盖范围较小的地区扮演移动基站的角色,它们可以用于将计算负载转移到 MEC 服务器或云和频谱资源分配等。MEC 的应用范围非常广泛,大多数 MEC 的研究都集中在最小化所使用的无人机的能量消耗,有些还伴随着计算延迟^[11]。MEC 还有其他涉及任务分配的应用,如空天地一体化网络(Space-Air-Ground Integrated Network, SAGIN)^[12]、MEC 中的安全性和移动云计算^[13]。

(2) 军事应用

军事应用包括攻击地面或空中目标、情报监视和侦察(Intelligence Surveillance and Reconnaissance, ISR)、压制敌人防空系统(Suppression of Enemy Air Defenses,

SEAD)和摧毁敌人防空系统(DEAD),近年来,许多研究工作集中于多无人机的ISR相关的任务分配问题,ISR任务包括一般侦察、针对不同类型无人机和目标的侦察^[14]、协作侦察、任务战略侦察,以及监视侦察^[15]。此外,ISR任务应用还包括具有截获概率的监视和具有任务不确定性的监视^[16]。其他军事应用包括SEAD任务,攻击目标和空战任务^[17]。军事应用中一个非常广泛的类别是ISR任务和攻击的结合,在大多数研究中无人机同时具有ISR和攻击能力,使用同时具有ISR和攻击能力的异构无人机,其中一些无人机能力的性能值不同^[18]。

(3) 搜索和救援任务应用

搜索和救援任务是无人机任务分配技术的一个常见使用领域,其主要目标包括:最大限度地增加幸存者的数量、最大限度地减少终止搜救任务所需的时间和最大限度地减少无人机飞行的距离。参与搜救任务的无人机可以同时具有搜索和救援能力,也可以存在两种不同类型的无人机,一种专注于搜索幸存者,另一种专注于救援任务。其他应用的计划包括使用两种不同类型的无人机,一种用于向幸存者提供食物,另一种用于提供药品^[19],以及无人机和无人驾驶地面车辆在山区困难地形地区的搜救任务中的合作^[20]。

(4) 民用应用

最近几年,无人机民用应用发展迅速,民用应用同样需要无人机的任务分配技术。在农业邻域内,无人机可用于农作物保护、喷洒杀虫剂^[4]和化肥以及湿度监测等^[21]。民用应用还包括城市中的监控和跟踪任务,这些任务需要无人机之间的高度合作,特别是当无人机应用在复杂城市环境中的导航任务时,例如高密度建筑或工厂。在这些环境中,无人机要执行电网检查、产品转移和物流、空中结构组装、人群感应和野地火灾处理等任务^[22-25]。

1.2.2 群体智能算法研究现状

对于多无人机协同任务分配问题,传统的确定性搜索算法可以获得低维问题的局部最优解,但当问题的复杂性和计算量增加时,传统的确定性搜索算法不再能够有效快速地找到问题的可行解,利用随机操作和元启发式算法可以得到可行解,但不一定能得到最优解。群体智能算法最近几年已广泛应用于多无人机系统的任务分配相关问题,群体智能算法的灵感来自具有社会行为的动物,包括昆虫群落、鱼群和鸟群等。这些动物表现出高效的分工,其专门成员为提高群体的整

体效率做出了贡献^[26]。虽然单个智能体可能并不复杂，但它们可以通过合作集体完成复杂任务，从而产生稳健、高效且计算成本不高的解决方案^[27]。不过，这些算法有时可能会把不需要的任务分配给智能体，从而导致冲突，而且对环境变化的全局反应也很慢^[28]。元启发式方法的两大类是基于阈值的方法和概率方法。基于阈值的方法，如响应阈值法^[29]，根据一些监测值和一个固定或可变的阈值来决定智能体的行动。智能体可能只有关于这些值的局部或全局信息。在概率方法中，任务是利用从环境观测或历史数据中导出的概率分布随机改变的。此外，这类方法还可以使用环境刺激，以刺激值作为选择特定任务的标准^[30]。

在军事应用中，大多数基于群体智能算法的技术都集中在 ISR 和目标攻击应用的结合上，如 Wang 等人^[31]研究了攻击和侦察应用中协同无人机之间的任务分配问题，将甲虫天线搜索(Beetle Antennae Search, BAS)与遗传算法相结合，从而增强了遗传算法的多样性和搜索能力，该算法比基准遗传算法和基于粒子群算法的基准变异算法具有更快的收敛速度和更好的性能。Wu 等人^[32]研究了多无人机侦察攻击应用中的任务分配问题，提出了一种改进的模拟退火法融合遗传算法，与模拟退火法相比，改进了解的接受准则，改进后的算法比基准模拟退火算法法具有更短的求解时间和更高的效率。Duan 等人^[33]对多无人机联合攻击任务规划问题进行了研究，提出了基于动态离散鸽群启发式优化算法，仿真结果表明了该算法的有效性，并且该算法的性能优于粒子群算法、鸽群算法、离散粒子群算法。Chen 等人^[34]研究了多无人机侦察攻击应用中的任务分配问题，提出了采用增强随机分形搜索的模糊狼群算法。与经典狼群算法和另外两种改进的狼群算法方法相比，该方法具有更好的全局搜索能力和整体性能。Han 等人^[35]研究了 ISR 和攻击应用中的多无人机任务分配和路径规划问题，利用 Dubins 曲线，提出了一种特殊粒子群算法，算法比经典粒子群算法具有更好的收敛速度和更好的解。Kolar 等人^[15]使用分布式、自下而上的动态蚁群分工方法，研究了用于 ISR 和攻击应用中无人机群的动态任务分配，与狼群算法相比，该方法具有更好的任务分配、更快的威胁响应速度、鲁棒性和灵活性。

此外，还有大量针对多无人机的 ISR 或攻击问题，如 Zhang 等人^[36]研究了多个无人机的任务分配问题，重点关注在任务分配过程中无人机信号被拦截的低概率，利用标准粒子群算法，提出了一种基于低截获概率的任务分配算法，仿真实

验证了该算法的有效性。Gao 等人^[14]研究了多无人机侦察任务分配问题, 根据目标的几何特征对其进行划分, 提出了基于蚁群算法改进的分组蚁群算法, 该算法比基准蚁群算法具有更好的搜索能力和更低的代价。对于仅攻击的应用, Huang 等人^[37]讨论了多无人机协同作战任务的任务规划问题, 将问题分解为多约束、多目标、耦合的整数优化问题和线性整数优化的资源分配问题, 对这两个问题分别采用了基于遗传算法和模拟退火法的求解方法。通过仿真验证了算法的有效性, 实现了快速的任务分配, 使其适用于复杂的约束环境。Xia 等人^[38]研究了多无人机攻击地面目标的任务分配和航迹规划问题, 提出了一种新的基于自适应参数调整和双向搜索的蚁群算法用于协作路径规划, 以及一种新的基于制导机制的改进粒子群算法来处理移动目标问题, 分别与基准蚁群算法、动态反馈蚁群算法、遗传算法、粒子群算法和多目标粒子群算法进行了比较, 证明了所提出算法的有效性。Zhang 等人^[39]针对目标攻击等作战无人机应用, 研究了具有资源约束和优先约束的多无人机协同任务分配问题, 提出了完全自适应交叉熵算法, 并与交叉熵算法和粒子群算法进行了比较, 在较大规模的问题中, 该方法比粒子群优化算法和并行进化算法具有更好的性能和更快的收敛速度, 而在小规模应用中, 其收敛速度略慢于粒子群优化算法和进化算法。

群体智能算法在其他应用上也被国内外众多学者研究。Zheng 等人^[40]通过同时处理传感器分配、任务分配和无碰撞路径规划, 研究了异类无人机的任务规划问题, 提出了两级自适应邻域搜索算法, 其中第一级用于传感器分配计划, 第二级用于路径规划, 与非自适应算法相比, 该算法具有更好的性能, 但计算量稍大。Miao 等人^[41]利用基于免疫多智能体网络框架的分布式免疫多智能体算法研究了多无人机的动态任务分配问题, 仿真结果表明, 该算法收敛速度快, 通信损失小, 鲁棒性好, 但静态任务分配的收敛速度较慢。Yang 等人^[42]研究了多无人机协同任务分配问题, 提出了一种改进的量子遗传算法, 采用了混合蛙跳算法和模拟退火法的分组优化策略, 与多粒度遗传算法和增量学习算法相比, 该方法具有更好的局部最优解和更大的多样性, 但存在计算复杂度较高的缺点。Ompusunggu 等人^[21]研究了无人机在农业活动中的使用, 重点研究了无人机与任务之间的距离以及所携带的资源, 提出了能力值敏感的集合路径蚁群优化方法 (Capability Value Sensitive-Collection Path Ant Colony Optimization, CVS-CPACO), 与集合路径蚁群算法相比具有更好的解和更少的资源浪费。Jin 等人^[43]研究了复杂化工厂和城市环

境中的无人机任务分配问题，这些环境中存在密集的建筑和不可忽视的环境风，提出了一种基于双胞胎排斥机制、分层目标支配算子和分段基因编码的技术，对第三代非支配排序遗传算法(Non-dominated Sorting Genetic Algorithm-III, NSGA-III)进行了改进，与 NSGA-III 算法相比，该算法具有更好的效率和收敛特性。

1.2.3 其他算法研究现状

最近几年，很多新的多无人机任务分配方法被提出，多无人机任务分配中常用的一类算法是基于拍卖的算法。这类算法利用较少的通信和计算资源，具有极大的灵活性和鲁棒性。Choi 等人^[44]提出了一种基于一致性的捆绑拍卖算法，即无人机对任务出价，出价最高者赢得任务，并通过共识程序最终确定中标者。性能影响(Performance Impact, PI)算法是解决对时间敏感的任务分配问题的一种实用方法^[45]，它采用启发式优化原则来优化总体目标，从而以较低的时间成本在不同任务间分配资源^[46]。Turner 等人^[47]在 2017 年的研究提出先使用“PI-最大化分配”后再使用 PI 方法，重点关注严格的时间限制和多次重新分配，为尚未分配的任务分配时隙，解决了具有复杂时间窗口约束的多无人机任务分配问题。Cui 等人^[48]在 2022 年对 PI 算法进行了修改。其目标是在三个阶段(即任务包含、冲突解决和任务重新分配)中最小化平均任务完成时间。Qamar 等人^[49]在 2023 年将 PI 算法扩展到最大分配问题，以最大化任务分配并改进任务分配。这种方法利用 PI 算法来分配任务，并在无人机之间重新分配任务，为未分配的任务建立时隙，最大化任务分配进程。

强化学习是一种常用的机器学习方法，近年来被多次用于研究多无人机任务分配问题，这种方法的主要思想是让无人机根据自己的经验学习如何在环境的各种状态下行动^[50]。在强化学习方法中，无人机通常在被建模为马尔可夫决策过程的环境中优化成本或者回报函数，Q-learning 是常用的无模型强化学习方法之一，它帮助无人机在马尔可夫决策过程中找到最优解^[51]。强化学习方法具有以下优点：它能够处理环境不确定性、能对训练好的网络进行实时实施以及具有处理不同任务的灵活性，然而在大规模复杂多无人机系统中，大多数强化学习算法需要大量的计算能力^[52]。大多数基于强化学习的方法的应用都集中在 MEC 应用上，Liu 等人^[53]研究了支持无人机的边缘计算问题，无人机可以从地面本地设备和其他邻近的无人机承担计算任务，提出了一种基于深度 Q 网络方法的深度强化学习框

架,用于集中式和分布式网络拓扑结构的协作卸载和资源分配过程。Xu 等人^[54]利用基于强化学习的方法研究了海上网络中基于无人机的 MEC 网络联合资源分配问题,通过仿真实验验证了该方法的收敛性和性能。

1.2.4 研究现状总结

当前多无人机协同任务分配问题研究还存在很多问题,对于任务优先级、动态环境和不确定性问题等相关方面值得更深入的研究和讨论。现阶段对群体智能算法的研究都由于算法本身的特性而存在不同方面的不足之处,对于不同应用场景的应用上还有改进和提升的空间。

考虑任务优先级是无人机任务分配问题确保有效处理所有任务的一个重要过程,任务优先级会在其他因素影响下,根据任务具体情况决定。Luo 等人^[55]在对多无人机系统进行任务规划时,设计多目标随机分散在环境空间的情况,打击不同目标产生的资源消耗也不同,所以要根据任务优先级进行主次分配,结果还需要满足多无人机性能方面约束,无人机飞行线路最优等条件。任务优先级被设置为:在无人机执行打击任务时,目标价值与距离近作为最高优先级;在时间约束要求程度较高时,目标任务执行时间最少成为最高优先级。Zhang 等人^[56]在研究多无人机协同执行对敌攻击任务的目标分配过程中设置任务优先级约束,重要程度高的目标优先被攻击,其优先级高于其他待攻击目标,若分配方案中出现某架无人机攻击次序未能满足任务优先级约束,设置一定的惩罚。还有很多研究为无人机设定了任务优先级,但是基本上对多无人机任务分配过程设置固定的任务优先级(任务序列),没有考虑到任务优先级变化的情况。

多无人机协同运行规划问题在实际场景中都是复杂的动态环境,算法在处理任务或环境的在线变更时需要表现出更强的弹性,从而产生更高的鲁棒性。动态环境中的多无人机系统经常面临在不确定时间和位置变化的任务,这种系统的例子包括将货物递送到接收者当前位置的无人机递送系统或攻击移动目标的无人机系统等。在这些系统中,任务点可以与移动实体(例如人或车辆)相关,动态环境中的多无人机系统具有有限的知识来预测其移动。为了提高其性能,在任务执行之前,动态环境中的多无人机系统不能依赖于一次性任务分配或调度,因为当动态实体移动时,原始分配或调度结果的优势可能很快降级,所以需要动态地调整其任务分配和调度结果^[57]。

多无人机协同任务分配在实际应用中，考虑不确定性对于高效、可靠的任务分配是至关重要的。许多现有的方法，特别是分布式方法，在处理不确定性方面的能力有限，而且往往依赖于对环境的过于简化的假设。不确定性可能由传感器不准确、试剂故障、环境干扰等引起。现有的研究表明，预先考虑可靠性是至关重要的，因为忽略故障的可能性可能会导致性能恶化，比如基于异步共识的捆绑算法在通信过程中具有不确定性的环境中会产生低效的任务分配，特别是对于大量的无人机系统^[58]。当使用优化技术时，很难在时间关键型任务分配任务中包含不确定性，因为不确定性需要在系统中表示，并且通常一些不确定性不能用解析的方式表示。另一方面，即使在可能的情况下，问题的维度也增加了很多，计算成本也增加了。一般来说，在上文讨论的许多任务分配技术中，不确定性是一个通常被忽略的问题。

1.3 论文主要工作及章节安排

1.3.1 论文主要工作

本文介绍了多无人机系统相关技术和群体智能算法的研究背景，以及多无人机协同任务分配方法在四种应用场景的研究现状，并重点介绍了用于解决不同优化问题的群体智能算法和其他算法的研究现状。本文对多无人机协同任务分配算法和多无人机协同任务分配模型进行了分析，针对相关问题和模型的不足之处，提出新的多无人机协同任务分配模型。本文的主要研究工作的结构关系如图 1.2 所示。

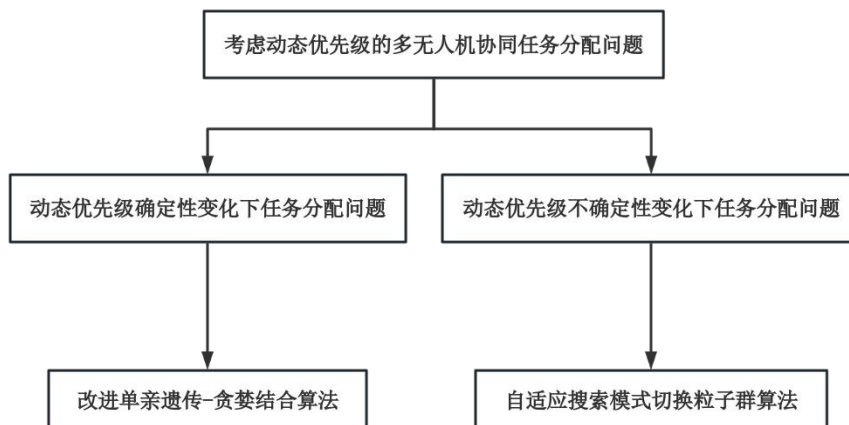


图 1.2 主要研究内容结构关系图

Figure 1.2 The main research content structure relationship diagram

本文建立了一种在特殊环境下任务优先级发生动态变化的多无人机协同任务分配模型，模型主要考虑了无人机执行任务过程中出现的任务优先级动态变化的问题。首先，本文根据区域差异化的思想，将环境划分成不同区域，在每个区域设定若干任务目标点。然后，本文通过用函数表示不同子区域的需求状态的动态变化，从而产生不同的任务优先级(A,B,C)。最后，本文构建多无人机协同执行物资分配任务的约束条件和目标函数，以最大化总奖励值作为多无人机协同执行任务完成程度的评价标准。

针对动态优先级确定性变化下的多无人机协同任务分配问题模型，本文提出改进单亲遗传-贪婪结合算法来对问题进行更高效地求解。本文用贪婪算法，改进单亲遗传算法和改进单亲遗传-贪婪结合算法来对问题进行仿真实验，实验验证了改进单亲遗传-贪婪结合算法具有优秀的搜索能力和搜索效率。

针对动态优先级不确定性变化下的多无人机协同任务分配问题模型，本文提出自适应搜索模式切换粒子群算法，对粒子群算法进行改进，设定不同搜索模式决定粒子的加速系数和学习对象，解决了粒子群算法容易陷入局部最优解的问题。本文用基于一致性的捆绑拍卖算法，自适应搜索模式切换粒子群算法和离散粒子群算法进行对比实验，实验验证了本文提出的自适应搜索模式切换粒子群算法具有更强的全局搜索能力，可以更高效地解决动态优先级不确定性变化情况下的多无人机协同任务分配问题。

本文在最后对全文研究内容进行了总结，分析并展望了可以继续深入研究的方向和思路。

1.3.2 章节安排

第一章：本章首先说明了多无人机技术的发展和群体智能算法的研究背景，其次介绍了多无人机任务分配方法在四种应用场景的研究现状，然后重点介绍了用于解决不同优化问题的各种群体智能算法和其他算法，最后对本文的主要研究内容和章节安排进行了阐述。

第二章：本章首先对多无人机协同任务分配问题相关研究进行了分析，对集中式算法和分布式算法两种任务分配算法和四种任务分配模型进行了介绍，得出已有相关研究或方法普遍在任务分配问题上设置固定的任务优先级(任务序列)并且欠缺对不确定性问题的研究的结论。然后，本章建立了一种在特殊环境下任务

优先级发生动态变化的多无人机协同任务分配问题模型，通过用需求状态动态变化函数来产生不同的任务优先程度(A,B,C)，以此产生动态优先级。最后，构建多无人机协同执行物资分配任务的目标函数和约束条件，以最大化奖励值为多无人机协同执行任务完成程度的评价标准。

第三章：本章研究了多无人机协同任务分配在动态优先级确定性变化下问题的求解和遗传相关算法的改进方式。首先构建了动态优先级确定性变化的多无人机协同任务分配模型，描述了不同子区域的需求变化，采用周期性函数来表示子区域 G_z 里接收点的需求状态。其次用贪婪算法求解此问题，设计了贪婪算法解决此问题的流程，通过递进的循环和选择得到一个解。然后分析了群体智能算法中最具有代表性的遗传算法，通过改进单亲遗传算法和改进单亲遗传-贪婪结合算法来对问题进行仿真实验。实验结果表明，本文提出的改进单亲遗传-贪婪结合算法具有优秀的搜索能力和搜索效率，在处理多无人机协同任务分配在动态优先级确定性变化下，具有很高的适应性。

第四章：本章主要研究了多无人机协同任务分配在动态优先级不确定性变化下问题的求解和粒子群相关算法的改进方式。首先，本章对不确定性问题进行分析，分析了基于拍卖式算法，基于启发式算法，基于强化学习式算法等处理不确定性问题时的特点。其次，本章构建了动态优先级不确定性变化情况下的多无人机协同任务分配问题模型，用泊松分布来生成不确定性变化情况下需求状态变化函数。然后，本章进行了基于一致性的捆绑拍卖算法对本章问题的应用设计，通过任务包构建阶段和冲突消解阶段求出多无人机任务分配方案。然后，本章提出自适应搜索模式切换粒子群算法，设定不同搜索模式决定粒子的加速系数和学习对象，解决了离散粒子群算法容易陷入局部最优解的问题。最后，本章对新提出的算法和基准算法进行了对比仿真实验，实验结果表明，本章提出的自适应搜索模式切换粒子群算法能够提高种群的多样性和全局搜索能力，可以更高效地解决动态优先级不确定性变化情况下的多无人机协同任务分配问题。

第五章：本章首先对全文研究内容进行了总结，然后提出了未来多无人机协同运行相关的技术应用发展前景，分析了论文中仍需要进一步研究的几种情形，并且提出了未来研究的方向。

2 考虑动态优先级的多无人机协同任务分配模型

2.1 多无人机协同任务分配问题分析

多无人机协同任务分配指的是根据静态已知威胁信息和地理的环境信息，以及无人机机体的设备参数和设备性能，制定多无人机的分配任务，同时能够随着环境的动态变化对任务方案变动进行协商和实时调整。为充分发挥各架无人机的优势，提升整个系统的执行效能，使无人机集群系统达到最理想的作战状态，多无人机系统需在执行任务之前做好准备工作。与计算机系统中传统的任务分配问题相比较，多无人机协同任务分配需要考虑不同的影响因素，例如有限的能耗与资源、无人机之间的通信限制等。本章通过任务分配算法和任务分配模型两大方面面对多无人机协同任务分配问题进行分析。

2.1.1 任务分配算法分析

解决多无人机协同任务分配问题的任务分配算法基于通信方式(是否存在中央服务器)分为两类：集中式算法和分布式算法^[59]。

(1) 集中式算法

在集中式算法中，存在一个中央服务器，它可以结合来自其他无人机的信息并判断，它与所有其他无人机有通信通道连接，如图 2.1 所示。中央服务器会确定最适合执行任务的无人机，按照使用的算法要求将任务分配给其他无人机。

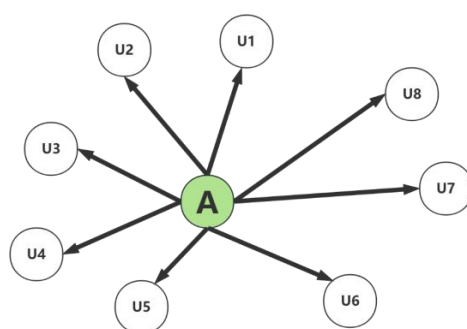


图 2.1 集中式算法示意图

Figure 2.1 A Schematic diagram of the centralized algorithm

集中式分配方法主要适用于解决小型规模的多无人机协同任务分配问题，在以减少系统资源或降低实施成本为目标的情况下，可以优先使用这种任务分配算法，与分布式算法不同，大多数计算能力仅在中央服务器上需要。集中式算法通

常不能有效地应用于存在大量无人机的系统，因为此时系统大幅度增加了计算成本。此外，集中式算法可伸缩性较低的另一个原因是所有无人机都必须与中央服务器通信，这使得无人机的通信负荷非常高。集中式算法通过多无人机任务的集中分配有助于避免冲突，有时不需要存在一致性判断阶段，在某些情况下也可以提前找到最佳解决方案^[60]。

（2）分布式算法

分布式算法能够解决集中式算法的一些局限性，算法在没有负责分配任务的中央服务器的情况下运行，如图 2.2 所示。每个无人机将其任务执行适应度与其邻居的计算拟合度进行比较，然后自行分配最合适的任务^[61]。无人机使用它们的局部情况观测和对环境的感知，并且如果所使用的算法需要的话，无人机还需要具有相互通信和协调的能力，这要求每个无人机以分布式方式在局部范围内做出任务分配决策，每架无人机也可以具有其本身的效用函数^[62]。

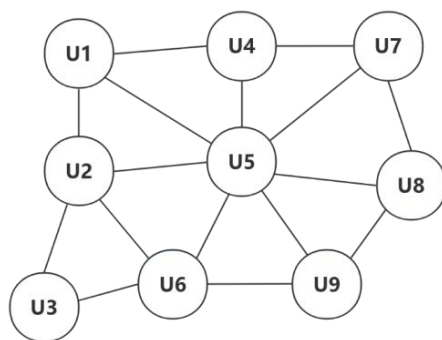


图 2.2 分布式算法示意图

Figure 2.2 A Schematic diagram of the distributed algorithm

分布式算法的优点主要为面对无人机故障方面的稳健性，出现故障时对整体性能影响很小。此外，分布式算法具有高度的可扩展性，因为通信要求通常低于集中式算法，不需要频繁地与中央服务器通信，在某些情况下，其他无人机甚至可以被视为环境的一部分。与集中式方法相比，分布式算法的计算成本相对较低，这使得分布式算法非常适合用于大规模系统。然而分布式算法同样存在缺点，这种算法可能只为任务分配问题提供次优或近似的解决方案，由于缺乏中央服务器，在一些任务分配方法中，可能需要一种基于共识的算法来解决可能由本地任务分配引起的冲突^[63]。分布式算法除了常见的完全分布式算法之外，近年来产生了两种新型的分布式算法，分别是部门分布式算法和智能体分布式算法。

部门分布式算法对无人机设定等级进行控制，设置中心服务器，算法将任务拆分后，分成多个单一任务，将任务集合交由长机，长机将多个单一任务分配给无人机群，无人机群完成各自任务避免了矛盾冲突，这种算法解决了集中式算法中央服务器计算量过大的问题。

智能体分布式算法将每架无人机设定为智能体，每个智能体能够完成指定类型任务，按照任务为智能体进行分组。无人机对分组内信息进行分析并进行协调，避免组内无人机之间的冲突。无人机具有很强的主动性以及群体性，所有个体集合组成集群，所有个体完成各自的任务从而完成集群的整体目标^[62]。

2.1.2 任务分配模型分析

多无人机协同任务分配问题模型主要包括多旅行商问题(Multiple Travelling Salesman Problem, MTSP)模型、多车辆路径问题(Multiple Vehicle Routing Problem, MVRP)模型、混合整数线性规划(Mixed Integer Linear Programming, MILP)模型和网络流模型(Network Flow Model, NFM)四种模型。

(1) 多旅行商问题模型

多旅行商问题模型是旅行商问题模型的拓展，假设场景中有若干个目的地，多架无人机从相同的初始城市出发，每个无人机要求自己的旅行路线与其余无人机者不同，且需要保证每个城市有且能够容纳一个无人机路过，要求求解出整体总路程最短的旅行方式。多无人机协同任务分配问题模型用于分配任务的标准参数之一是行驶距离，此时，任务分配过程可以表述为多旅行商问题，以选择使任务分配问题最大化的最优分配向量^[64]。

(2) 多车辆路径问题模型

多车辆路径问题模型是由车辆路径问题拓展而来。假设有若干无人机，从同一个基地出发，为不同的地方运送物资，每个地方需求的物资数量不同，距离出发点的距离也不同，如何保证全部物资运送到达且距离最短是该问题模型求解的问题。多车辆路径问题模型在其无人机的参数诸如最大载重、最大行驶距离、固定成本以及变化成本等方面不能够完全相同，还有些多车辆的路径问题中引入了多个停车场的概念，增加了对车辆的调整和控制问题^[65]。

(3) 混合整数线性规划模型

混合整数线性规划模型在线性规划的单纯形间算法基础上增加了一个整数约

束, 用来表示机器数量或人的个数情况, 改进后更可以处理更多实际问题^[66]。混合整数线性规划模型描述简单直接, 方便无人机进行全局约束, 提高了完成任务的效率和生存能力, 在无法得到最优解的情况下, 能很快的求解出次优的解^[37]。

(4) 网络流模型

网络流模型是无人机在指定的物流网络环境下, 将执行任务起点和目标当做物流过程中的供货方和收货方, 将无人机任务分配问题转换成物流成本和利润的运算问题。网络流模型减少了求解难度, 使用目标数据集进行学习, 选择并优化相应的组合性特征。利用网络流模型对多无人机搜索任务分配问题进行研究, 能够实现多无人机协同搜索任务, 降低了多无人机网络流的总代价^[67]。

根据对多无人机协同任务分配问题的分析, 本章设计了一种在特殊环境下任务优先级发生动态变化的多无人机协同任务分配问题模型。为了增强本章所建模型的可扩展性和普适性, 将不确定性加入研究, 分别研究了确定动态优先级变化情况下的多无人机任务分配问题和不确定动态优先级变化情况下的多无人机协同任务分配问题。

2.2 动态优先级问题模型描述

2.2.1 动态优先级问题模型

已知物资配送任务区域内共有 M 个接收点, 接收点的详细位置坐标信息已提前通过侦察系统或者雷达等手段获取到。物资配送任务区域被分为 z 个子区域 $G=\{G_1, G_2, \dots, G_z\}$, 其中每个子区域中存在着若干个接收点, 用 $\{M_{z1}, M_{z2}, \dots, M_{zm}\} \in G_z$ 进行表示, 如图 2.3 所示, 用不同形状的符号表示了不同子区域中的接收点, 黑色方形表示派送中心。

已知有 $N(N < M)$ 架无人机 $U=\{u_1, u_2, \dots, u_N\}$ 在派送中心中, 它们的性能是相同的。这些无人机需要通过配送物资救助需要物资的接收点, 通过往返于接收点之间和派送中心执行任务。多架无人机的主要目标是在限定轮次内并且满足各种约束条件下, 能够有效地执行这所有 M 个接收点的配送任务, 本文主要的研究问题是求解多无人机最佳协同任务分配方案。

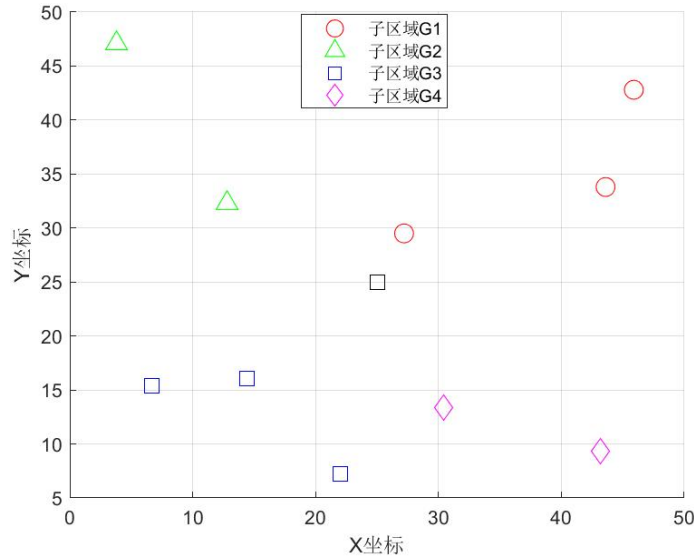


图 2.3 不同子区域划分图

Figure 2.3 Different sub-region division diagram

在无人机执行配送任务的过程中，不同子区域对物资的需求量和急需程度会发生变化。当子区域的需求变化能够确定时，变化是可预知的，有规律性的。此时，采用周期性函数来表示子区域 G_z 里接收点的需求状态。本文为了避免问题函数设计的过于复杂，周期内的函数使用简单并且具有代表作用的函数来设计。本文采用近年来被多次用于研究的安全库存模型作为参考^[68]，在原文献的基础上进行了一定的简化和修改，修改后的子区域 G_z 的需求状态随时间变化的函数：

$$\begin{cases} W_{G_z}(t) = c\sqrt{\xi_z t^2} & t \in [0, T] \\ W_{G_z}(t+T) = W_{G_z}(t) \end{cases} \quad (2.1)$$

其中 $W_{G_z}(t)$ 表示 t 时刻子区域 G_z 的需求状态值， ξ_z 单位/(h)表示子区域 G_z 需求状态的变化率， c 表示物资消耗水平系数，函数的周期为 T 。

当子区域的需求变化不能够确定时，变化是不可预知的，无规律的情况。此时用动态需求变化问题经常被使用的方法：泊松分布应用设定子区域 G_z 的需求状态随时间变化的函数，函数中 λ 为泊松分布参数，函数如公式(2.2)所示：

$$W_{G_z}(t) = c\sqrt{\xi_z T^2} \cdot \frac{\lambda^t}{t!} e^{-\lambda} \quad (2.2)$$

为了设定体现任务需求紧急度的指标，通过任务动态优先级程度集合 $P=\{A,B,C\}$ 来进行表示，其中 A,B,C 分别表示不同的任务优先程度，A 表示紧缺状态，B 表示一般紧缺状态，C 表示不紧缺状态。

当子区域 G_z 的需求状态周期函数值每次上升的过程中出现函数值等于界限值 ($W_{G_z}(t_n)=W_0$) 的情况之后, 区域 G_z 中的所有接收点变成状态 A。子区域 G_z 的每次转换到状态 A 的时刻用集合 $T_z=\{t_1, t_2, t_3, \dots, t_n\}$ 表示。因为需求状态函数为周期函数, 可以确定:

$$t_{n+1} = t_n + T \quad (2.3)$$

对于每个子区域的接收点默认初始状态为 B, 此时若从无人机接收到物资后状态变为 C。如果没有接收到物资, 等到 t_1 时刻, 子区域 G_z 会从状态 B 变化为状态 A。子区域 G_z 在状态 C 度过一段时间后, 等到下一次区域状态转换时刻后会再次变成状态 A, 状态 A 在从无人机接收到物资后状态再变为 C, 状态 A 和状态 C 之后形成循环。按照上述模式绘出区域状态转换模式图如图 2.4 所示, 可以更直观地展示出状态之间的转换模式。

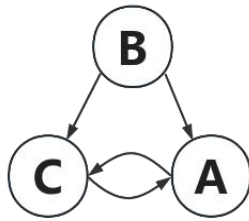


图 2.4 状态转换模式图

Figure 2.4 State transition pattern diagram

2.2.2 目标函数设定

从现实角度进行分析, 在紧缺状态和一般紧缺状态下, 无人机在越少的时间内完成配送任务会获得更高的收益, 如果超过了一定的时间范围将会产生惩罚。根据上文的表述, 设定了奖励函数作为多无人机协同任务分配的目标函数。

在构建多无人机协同任务分配问题目标函数的过程中, 无人机相关参数设定如表 2.1 所示。

除表 2.1 外, 还有其他约束条件:

(1) 每架无人机的最大载重量: $L_{max}=kL_0$ 。

(2) 每架无人机单次飞行最大飞行路程限制: $\sum_{u=1}^n \sum_{z=1}^Z \sum_{l=1}^m D_{z2m2}^{z1m1} x_{z2m2}^{z1m1} + D_{zm}^P x_{zm}^P \leq R_{max}$

(3) 不同子区域 G_z 的需求状态的变化率不同: $\xi_1 \neq \xi_2 \neq \xi_3 \dots \neq \xi_z$ 。

表 2.1 参数设定

Table 2.1 Parameter settings

代表符号	参数意义
L_0	每个接收点单次接收固定重量物资
T_{unload}	每架无人机在接收点的卸货时间
T_{load}	无人机返回派送中心进行装货和补充燃料过程的时间
V_{AVG}	每架无人机运行状态的平均速度
D_{z2m2}^{z1m1}	目标接收点 M_{Z1m1} 与目标任务点 M_{Z2m2} 之间的直线距离
P	派送中心位置
D_{zm}^P	目标接收点 M_{Zm} 与派送中心的直线距离
$x_{z2m2}^{z1m1}=1$, 否则 $x_{z2m2}^{z1m1}=0$	(0-1)变量, 若无人机 u_n 从目标接收点 M_{Z1m1} 移动到目标任务点 M_{Z2m2}
$x_{zm}^p=1$, 否则 $x_{zm}^p=0$	(0-1)变量, 若无人机 u_n 从派送中心移动到目标任务点 M_{Zm}
d	单位距离运输成本

无人机第 i 次在子区域 G_Z 的第 m 个接收点 M_{Zm} 完成配送任务的奖励值为 $P_{Zm}(t_i)$, 通过公式(2.4)求得。

$$P_{Zm}(t_i) = \begin{cases} P_1 \times [1 - \frac{2(t_i - t_{Astart})}{T}], & (t_i \in T_A) \\ P_2 \times [1 - \frac{2(t_i - t_{Bstart})}{T}], & (t_i \in T_B) \end{cases} \quad (2.4)$$

当超过紧缺状态开始时间 $\frac{T}{2}$ 小时(需求状态函数的周期的 $\frac{1}{2}$)和超过一般紧缺状态开始时间 $\frac{T}{2}$ 后, 奖励函数 $P_{Zm}(t)$ 的值小于 0, 此时产生负收益(惩罚), 到达时间越晚, 函数值越低, 则惩罚越多。公式(2.4)中 t_i 为无人机 i 到达接收点 M_{Zm} 的时刻, $t_i \in (0, T_{max}]$, T_{max} 为最晚接收时间。 P_1 和 P_2 分别表示紧缺状态和一般紧缺状态的每次配送任务起始奖励值。为了表示出不同任务优先级的奖励设置差异, 紧缺状态 A 的初始奖励值应多于一般紧缺状态 B, 所以设定要求 $P_1 > P_2$ 。 T_A 和 T_B 分别表示子区域 G_Z 的紧缺状态时期和一般紧缺状态时期, t_{Astart} 表示 t_i 时刻之前最近一次转换为状态 A 的时刻, t_{Bstart} 同理。

成本计算过程主要分为两个部分：固定成本和运输成本。

(1) 固定成本

无人机配送过程中的固定成本指的是无人机在任务过程中总额不受工作量的增减而保持不变的成木，包括检修维护费及管理保险费等。 F_n 是每架无人机的固定成本。固定成本用 F_{cost} 表示，固定成本函数为：

$$F_{cost}=N \times F_n \quad (2.5)$$

(2) 运输成本

运输成本的计算是用单位距离内运输成本乘以运输总距离，计算过程忽略无人机起飞和降落过程的时间和距离，以模型中坐标点之间的距离作为运输距离。 i_{max} 表示所有无人机中，单架无人机出现执行同一个接收点任务的最多次数。运输成本用 T_{cost} 表示，运输成本函数为：

$$T_{cost} = d \times (\sum_{l=1}^{i_{max}} \sum_{m=1}^n \sum_{j=1}^Z \sum_{k=1}^m D_{z2m2}^{z1m1} x_{z2m2}^{z1m1} + D_{zm}^P x_{zm}^P) \quad (2.6)$$

为了评价多无人机协同任务分配的完成情况，以最大化总奖励值作为目标函数的设置条件。因此，目标函数设定为在限制时间 T_{max} 内，所有无人机所获得的奖励值减去所有无人机的成本得到的最后的收益值，目标函数如公式(2.7)所示。

$$MaxP = [\sum_{l=1}^{i_{max}} \sum_{m=1}^n P_{Zm}(t_i)] - F_{cost} - T_{cost} \quad (2.7)$$

2.3 本章小结

本章首先对多无人机协同任务分配问题相关研究进行了分析，对集中式算法和分布式算法两种任务分配算法进行了介绍，指出了两种算法各自的特点和缺陷。其次，对多无人机协同任务分配问题主要包括的多旅行商问题模型、多车辆路径问题模型、混合整数线性规划模型和网络流模型四种模型进行分析，指出各种问题模型的主要应用场景，对多无人机任务优先级方面和不确定性方面相关研究问题进行分析得出已有相关研究或方法普遍在任务分配问题上设置固定的任务优先级(任务序列)并且欠缺对不确定性问题的研究的结论。本章建立了一种多无人机协同任务分配模型，模型主要考虑无人机配送任务过程中的接收点的任务优先级动态变化的问题。先对接收点进行区域化分，构建子区域需求状态函数，用需求状

态动态变化函数来产生不同的任务优先程度(A,B,C)，以此产生不同接收点的动态优先级。最后，本章构建多无人机协同执行物资分配任务的目标函数和约束条件，以最大化奖励值作为多无人机协同执行任务完成程度的评价标准。

3 动态优先级确定性变化下多无人机任务分配算法研究

本文提出的问题模型同时拥有离散和连续两种性质而被定义为不确定的多项式时间(Non-deterministic Polynomial-time hard, NP-hard)问题,传统算法很难完全满足系统对于求解效率、精度及稳定性等方面的要求,群体智能算法作为自然启发式算法,是解决相关类似问题的有效工具。对于动态优先级确定性变化下的多无人机任务分配问题,本章选择贪婪算法作为基准算法,用贪婪算法先确定可行解的范围。本章将遗传算法作为研究的基础,选择近些年新提出的改进单亲遗传算法作为基准算法,在该算法的基础上进行改进,提出改进单亲遗传-贪婪结合算法,设计适合本文问题模型的编码方式,改进初始种群的产生方式,将贪婪算法得到的可行解添加进初始种群,引入自适应交叉算子和变异算子产生公式。最后,本章通过仿真实验验证了改进后的算法的有效性。

3.1 动态优先级确定性变化下问题描述

为了提高确定性变化问题的普适度,本章采用周期性函数来表示子区域 G_z 里接收点的需求状态,周期内的函数用任何一个满足限制条件的函数都能来定义。本章采用近年来被多次用于研究的安全库存模型作为参考,在原文献的基础上进行了一定的简化和修改,修改后的子区域 G_z 的需求状态随时间变化的函数见公式(2.1)。安全库存是物流中的一个概念,是为物资供应和需求的不确定因素而准备的缓冲库存,企业使用安全库存以确保生产活动不会受到不确定因素影响从而导致停产情况的发生^[68],用安全库存模型来表示疫情突发时对物资的动态需求。安全库存模型可以用于表达存货率的要求,同时用来表达物资的急需程度,与本文构建模型的动态优先级变化相契合。

根据表 3.1 设定不同子区域需求状态变化率。

表 3.1 不同子区域需求状态变化率

Table 3.1 Rate of demand state change in different sub-regions

ξ_1	ξ_2	ξ_3	ξ_4
0.4	0.8	1.3	1.6

设定周期 $T=4$ ，物资消耗水平系数 $c=2$ ，界限值 $W_0=3$ 等参数后，此时根据公式(2.1)可以得到如图 3.1 所示的不同子区域需求状态函数图。

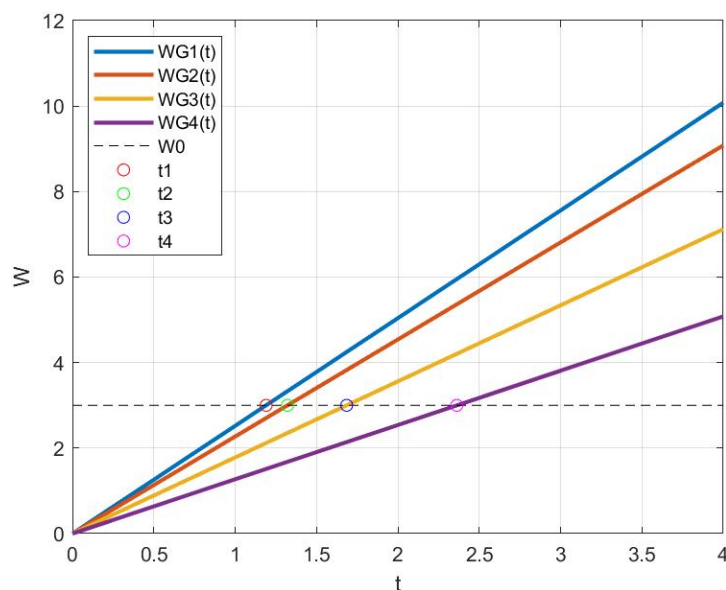


图 3.1 不同子区域需求状态函数

Figure 3.1 Different sub-regions requirement state functions

当子区域 G_z 的需求状态周期函数值上升的过程中出现函数值等于界限值 ($W_{G_z}(t_1)=W_0$) 的情况之后，子区域 G_z 中的所有接收点变成状态 A。图 3.1 中 t_1, t_2, t_3, t_4 四个点分别表示四个子区域在第一个周期内变化成状态 A 的时刻。需求状态变化后，奖励函数也会改变，根据公式(2.4)进行计算。

为了评价多无人机协同任务分配的完成情况，用总奖励值作为任务完成情况的评价标准，总奖励值函数见公式(2.7)。通过用贪婪算法和群体智能算法求解出能使总奖励值最大化的多无人机任务分配方案是本章的主要研究内容。

3.2 贪婪算法应用设计

3.2.1 贪婪算法概述

贪婪算法(Greedy Algorithm)，又称贪心算法，是一种能够得到某种度量意义下的最优解的分级处理方法，它总是做出在当前看来是最优的选择，其并不是从整体上进行考虑，所做出的选择只是在某种意义上的局部最优解算法，然后通过局部最优解求解得到全局最优解^[69]。

贪婪算法具有无后效性。算法的基本思路是从问题的某一个初始解开始，逐步地进行，按照设定优化测度，每步都确保能获得局部最优的情况，每步只考虑

一个数据，然后向下进行，且不能回溯。如果下一个数据与部分最优解连接在一起不再形成可行解，那么就不将该数据添加到部分解中，直到枚举完所有数据或无法继续添加为止，算法随之停止。

对于问题求解，必须确定一个构成解的潜在候选对象集合，该集合能够优化目标函数，这一过程涉及逐步实施贪婪算法。首先，算法选择的候选对象集合为空。然后在逐步计算过程中，按照函数结果进行选择，在集合剩余的对象中选取最优秀的构成对象。如果该对象加入集合后出现集合错误，那么该对象会被迅速丢弃，不再列入考量范围，否则，该对象将被添加到集合中，随着集合不断扩充，对集合可行性进行检查，整个算法在正确运行的时候初始解即为当前最优。

3.2.2 贪婪算法在本问题中的应用设计

贪婪算法思路虽然是一种通用思路，但是具体用于新问题时，需要根据条件进行一定的细化设计。本文基于贪婪算法思路对用于解决动态优先级确定性变化多无人机任务分配问题的贪婪算法进行了如下设计，具体过程步骤为：

步骤 1：对无人机位置和接收点位置初始化，设定场景。

步骤 2：设定任务总共有 k 轮。

步骤 3：遍历所有没有被分配的接收点，根据每架无人机当前位置和执行完前面已分配任务后到达这个接收点时的时间计算把当前轮到任务分配给当前轮到无人机的预期奖励值。

步骤 4：把所有奖励值进行排序，得到当前最大奖励值，记录此时奖励值最大的无人机和任务组合。

步骤 5：对无人机位置和已分配的任务进行更新，若此时所有任务分配出去执行步骤 6，否则返回步骤 3。

步骤 6：所有无人机返回派送中心，计算本轮的总奖励值。

步骤 7：对每轮获得的奖励值进行累加，然后进行任务总共进行的轮次进行判断，若已进行 k 轮，输出总奖励值，若不足 k 轮，返回步骤 3。

对应用在本问题中的贪婪算法的时间复杂度进行分析，时间复杂度为：

$$O(T) = k \bullet n \bullet O((m - m_{um}) + (m - m_{um} + 1) + \dots + (m - 1)) \quad (3.1)$$

其中， $O(T)$ 表示应用在本问题中的贪婪算法的时间复杂度， n 表示无人机个数， m 表示接收点总数， m_{um} 表示未分配的接收点个数，设任务分配共进行 k 轮。

用流程图来展示贪婪算法在动态优先级确定性变化下多无人机协同任务分配问题中的具体流程，如图 3.2 所示。

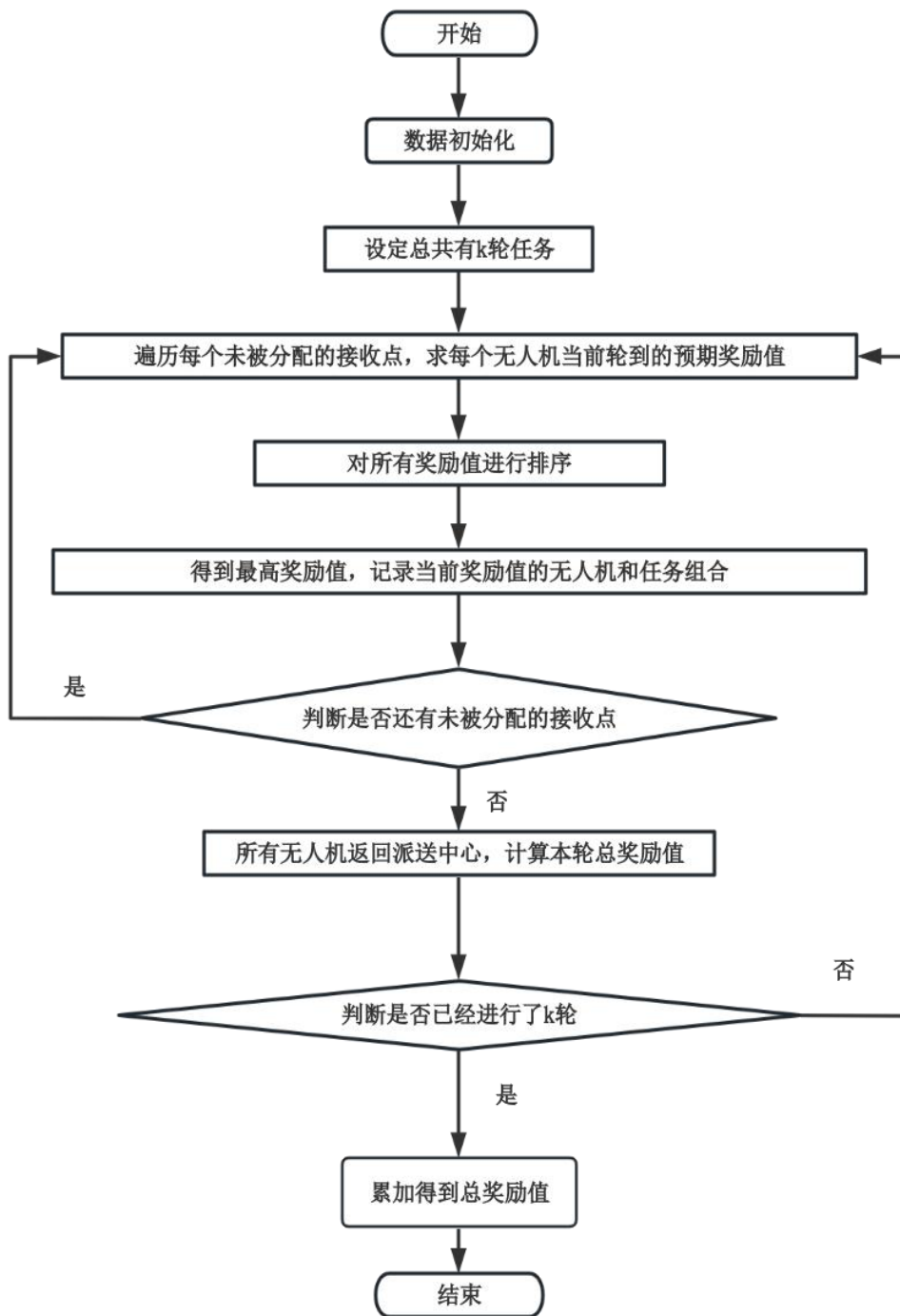


图 3.2 贪婪算法在本问题中的应用流程图

Figure 3.2 Application flow chart of the greedy algorithm in this problem

3.3 改进单亲遗传-贪婪结合算法设计

3.3.1 遗传算法概述

遗传算法(Genetic Algorithm, GA)是一类通用问题求解方法,一般建立在自然选择和群体遗传学规则基础上,具有普遍的适用性^[70]。遗传算法以进化中的优胜劣汰思想作为依据,对染色体执行复制、交叉、突变等生物基因进化操作以产生更能适应环境的个体。相比枚举法等传统优化算法,遗传算法以生物进化为原型,随机性较强,并且全局寻优能力良好,因此被人们广泛地应用于组合优化、数据库查询和信号处理等领域。

通过对问题解决需求的提升,组合优化问题的规模随需求的增加呈现出指数级增长,导致问题求解的搜索空间显著扩大。在应对这类问题时,关注点主要集中在获得令人满意的解决方案上,而遗传算法在这方面具有显著的优势。大量研究实际案例体现出遗传算法在解决旅行商问题、车辆路径问题、和图着色问题等各种 NP-hard 问题上的发挥了重要作用。

遗传算法求解组合优化问题的基本过程为:

步骤 1: 将问题参数编码;

步骤 2: 产生初始种群;

步骤 3: 计算种群上各个体的适应度函数值;

步骤 4: 按选择规则选择将要进入下一代的个体;

步骤 5: 按交叉概率 P_c 进行交叉操作;

步骤 6: 按变异概率 P_m 进行变异操作;

步骤 7: 如果不满足停止要求,则转到步骤 3,否则进入下一步;

步骤 8: 将最优解替换成适应度函数值最高的个体,然后输出结果。

编码过程通过将具体求解的表述转换为算法空间解的表示,即使用字符串或位串组成基因,其中二进制串被通常使用。解码操作则是染色体编码操作的相反操作,经常使用的方式有单染色体编码和双染色体编码^[71]。

单染色体编码是指将 n 个接收点, m 个无人机编码成一条长度为 $m+n-1$ 的染色体,用 $m-1$ 个负整数将 n 个接收点分为 m 段,分别代表 m 个无人机所要遍历的接收点,单染色体编码方式的解空间为 $(m+n-1)!$,具体表示如图 3.3 所示,即将 10 个接收点分配给 4 个无人机进行遍历,则 4 个无人机分别要遍历的接收点为(4,6),

(9,7,5), (10,3,8,2), (1)。

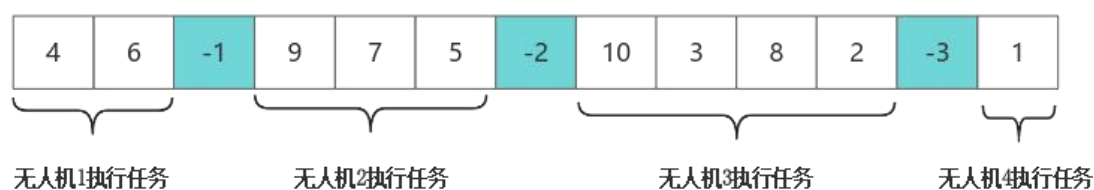


图 3.3 单染色体编码示例

Figure 3.3 Example of the single-chromosome encoding

双染色体编码是指将 n 个接收点编为一条染色体, 用各个接收点对应的无人机编码另一条染色体, 双染色体编码方式的解空间为 $n!m^n$, 具体表示如图 3.4 所示, 4 个无人机分别要遍历的接收点为(5,3), (4,2,8), (6,1,7), (9,10)。

接收点	4	6	1	9	7	5	2	10	3	8
无人机	2	3	3	4	3	1	2	4	1	2

图 3.4 双染色体编码示例

Figure 3.4 Example of the dual-chromosome encoding

单亲遗传算法(Partheno Genetic Algorithm, PGA)是李茂军等人提出的一种特殊遗传算法^[72], 该算法在求解组合优化问题时放弃了传统遗传算法中的交叉操作, 不使用传统遗传算法中的特殊交叉算子(两点交叉、循环交叉、均等交叉和顺序交叉等), 通过只在单条染色体上执行基因倒位、基因移位等操作来实现基因重组的过程。

PGA 算法中核心是个体基于父代进行单亲变异, 采用的变异操作有基因移位、基因倒转、基因左滑、基因右滑这四种变异操作与随机插入思想相结合, 形成以下四种单亲变异操作:

1.基因移位操作: 随机产生两个交换点, 两个交换点之间进行基因交换, 然后将介于这两个交换点之间的片段整体剪切并插入到片段外随机两个相邻基因点之间。

2.基因倒转操作: 随机产生两个倒转点, 将这两个点之间的基因片段进行倒序排列, 然后将新的片段剪切并插入到片段外随机两个相邻基因点之间。

3.基因左滑操作: 随机产生两个左滑点, 将这两个左滑点内的每个基因依次向

左移动一个点，然后将新的基因片段剪切并插入到片段外随机两个相邻基因点之间。

4.基因右滑操作：随机产生两个右滑点，将这两个右滑点内的每个基因依次向右移动一个点，然后将新的基因片段剪切并插入到片段外随机两个相邻基因点之间。

与传统遗传算法相比，PGA 算法具有执行简单，搜索效率高等特点，PGA 算法的流程如图 3.5 所示：

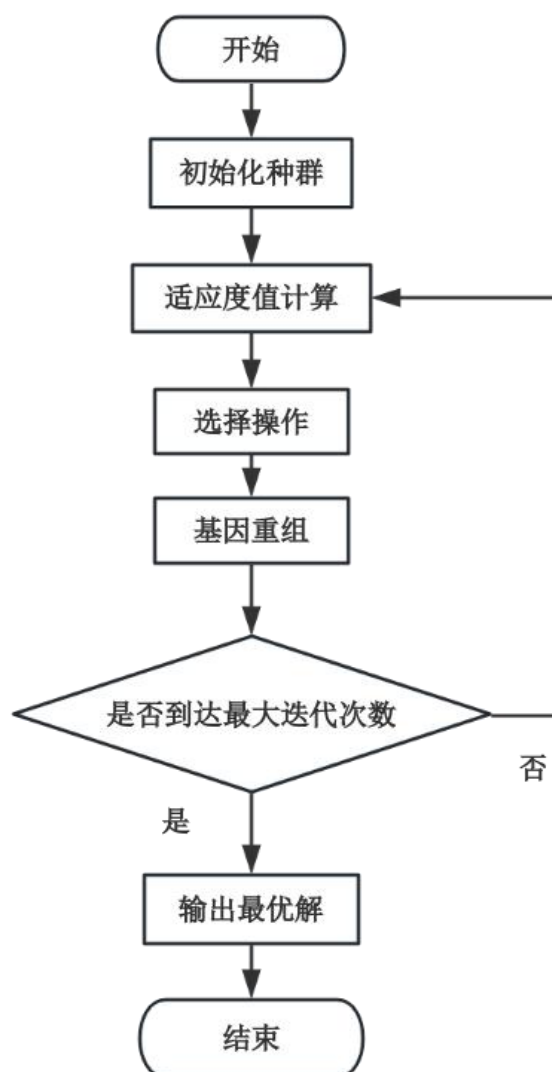


图 3.5 PGA 算法流程图

Figure 3.5 Flow chart of the PGA algorithm

3.3.2 改进单亲遗传-贪婪结合算法在本问题中的应用设计

对于本章提出的多无人机协同任务分配问题，需要把任务序列染色体切割为若干个片段，表示无人机各自执行任务访问接收点路线方案。Zhou 等人提出的 IPGA(Improved Partheno Genetic Algorithms, IPGA)算法使用两段式染色体编码方式来解决多旅行商问题^[73]，该问题与本文提出的模型相似，本文提出的改进单亲遗传-贪婪结合算法(Improved Partheno Genetic Algorithms-Greedy combination algorithm, IPGA-G)在原编码方式上进行修改，并且增加约束条件以符合前文提出的模型所要求的每架无人机执行最少接收点任务数目。

两段式染色体编码是由前后两部分组成，如图 3.6 所示，第一部分为 n 个物资接收点的随机排列，第二部分表示间断点编码段，将第一部分划分为 m 个片段，这 m 个片段即是 m 个无人机各自需要执行的任务序列。记无人机的起点派送中心为 0，则第一个无人机的本轮任务访问接收点路线为(0-4-6-0)，第二个无人机的本轮任务访问接收点路线为(0-2-9-0)，第三个无人机的本轮任务访问接收点路线为(0-7-5-1-0)，第四个无人机的本轮任务访问接收点路线为(0-10-3-8-0)。第一部分有 $n!$ 种排列，第二部分有 C_{n-1}^{m-1} 种排列，两段式染色体的解空间为 $n! * C_{n-1}^{m-1}$ 。

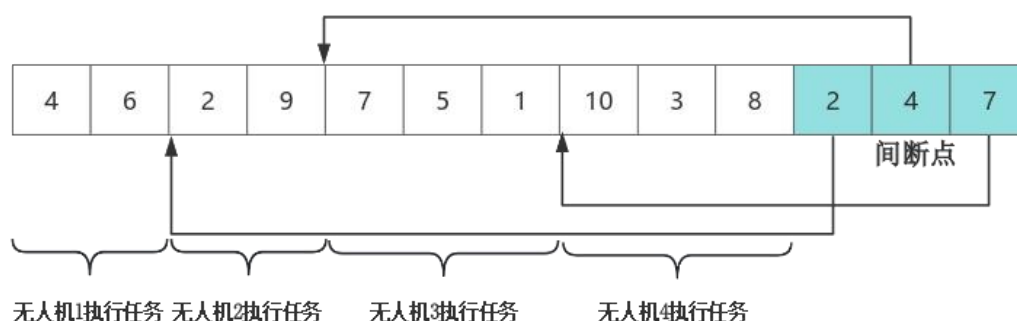


图 3.6 两段式染色体编码示例

Figure 3.6 Example of a two-segment chromosome encoding

在两段式染色体初始化编码的过程中，原始路线种群中的每个个体都应该是 1 到 n 的整数的随机排列。原始间断点染色体中的每个个体长度为 $m-1$ 。虽然 $m-1$ 个间断点是随机选择的，但为了满足本文提出的模型中假设每个无人机必须走过的最少接收点数目等于 N_{min} 的要求， $m-1$ 个数的每个组合都必须满足以下限制条件：

- (1) 数字应按递增顺序排列。
- (2) 每两个相邻数字之差不能小于 N_{min} 。

(3)第一个数字不能小于 N_{min} 。

(4) n 与最后一个数字之差不能小于 N_{min} 。

遗传算法的初始种群表达的是解空间的信息，初始种群覆盖的范围越广，则包含全局最优解的可能性就越大，因此初始种群在一定程度上影响了遗传算法的求解能力。IPGA 算法的初始种群使用随机性设置方法有一定的缺点，初始种群中的整体适应度值不会太优秀，导致 IPGA 算法的收敛过程效率不高。贪婪算法的核心思想是得到某种程度上最优的可行解，将基于贪婪算法得到的可行解加入到遗传算法的初始种群在一定程度上可以提高算法搜索效率。通过对贪婪算法得到的可行解进行基因重组，变异后的解大概率比随机产生的染色体具有更高的适应度值，更大概率被选择继续进行下一轮的单亲繁殖。

综上所述，本文提出 IPGA-G 算法对 IPGA 算法的初始种群获取过程进行了修改，优化染色体编码的默认随机产生初始化方式，将基于贪婪算法得到的解加入初始化种群。相比于 IPGA 算法，改进单亲遗传-贪婪结合算法能够在更少的迭代次数内完成对最优解的搜索，可以减少运算时间。

为了让子代种群更加多元化，扩大子代种群的寻优范围，IPGA-G 算法修改了 IPGA 算法的变异方式，对于选择过程后形成的包含 10 个个体的种群进行基因重组，在原算法 4 种变异算子的基础上扩展为 10 种基因重组方式：

- (1) 不对第一个个体做任何操作。
- (2) 第二个个体执行基因移位操作。
- (3) 第三个人执行基因倒转操作。
- (4) 第四个个体执行基因左滑操作。
- (5) 第五个个体执行基因右滑操作。
- (6) 第六个个体执行修改间断点操作。
- (7) 第七个个体执行基因移位操作和修改间断点操作。
- (8) 第八个个体执行基因倒转操作和修改间断点操作。
- (9) 第九个个体执行基因左滑操作和修改间断点操作。
- (10) 第十个个体执行基因右滑操作和修改间断点操作。

通过对单个染色体随机执行 10 种基因重组方式其中之一的操作，来实现算子的单亲繁殖与变异过程，实现子代种群更加多元化，扩大子代种群的寻优范围，

同时加速了进化过程。修改间断点的操作有利于扩大种群的搜索范围，防止陷入局部最优解。

本章在 Hu 等人^[74]提出的自适应遗传算法基础上，设计能够提高 IPGA-G 算法速度的自适应交叉算子。根据不同的适应度值设置不同的交叉概率，用自适应交叉算子 P_c 来表示基因重组过程中执行基因移位，基因倒转，基因左滑，基因右滑四种操作的概率，IPGA-G 算法的自适应交叉算子计算公式为：

$$P_c = \begin{cases} k_1 - \frac{(k_1 - k_2) \times |F - F_{avg}|}{F_{max} - F_{avg}}, & F \geq F_{avg} \\ k_1, & F < F_{avg} \end{cases} \quad (3.2)$$

其中， F_{max} 为当前适应度最大值； F_{avg} 为本代平均适应度值， k_1 和 k_2 是交叉算子系数，满足 $k_1 > k_2$ 。

在对染色体进行变异操作时，用变异算子来表示基因重组过程中执行操作(6)-操作(10)的概率，和自适应交叉算子设置类似， k_3 和 k_4 是变异算子系数并且 $k_3 > k_4$ ，自适应变异算子计算公式为：

$$P_m = \begin{cases} k_3 - \frac{(k_3 - k_4) \times |F - F_{avg}|}{F_{max} - F_{avg}}, & F \geq F_{avg} \\ k_3, & F < F_{avg} \end{cases} \quad (3.3)$$

为了使 IPGA-G 算法生成子代的选择过程能最大程度保留优秀的基因，对较差基因中的有用信息也尽量保存，IPGA-G 算法参考入侵野草优化算法(Invasive Weed Optimization, IWO)中的繁殖机制来选择产生子代种群^[75]。根据每个父代个体的适应度，按照公式(3.4)产生子代种群。

$$S(i) = \frac{(S_{max} - S_{min}) \times (F_{min} - F(i))}{F_{min} - F_{max}} + S_{min} \quad (3.4)$$

IPGA-G 算法能确保发现的最佳个体一定会留在后代群体中，最佳个体将通过几种不同的方式发生变异，产生新的个体。在初始搜索过程中，避免了一些潜在模型被淘汰，被选中并用于变异产生后代的个体正是一个相对较好的个体。基于上述原因，IPGA-G 的全局搜索能力和局部搜索能力更加出色，更容易找到最优解，具有更高的效率和更好的收敛能力。

改进单亲遗传-贪婪结合算法的整个流程如图 3.7 所示。

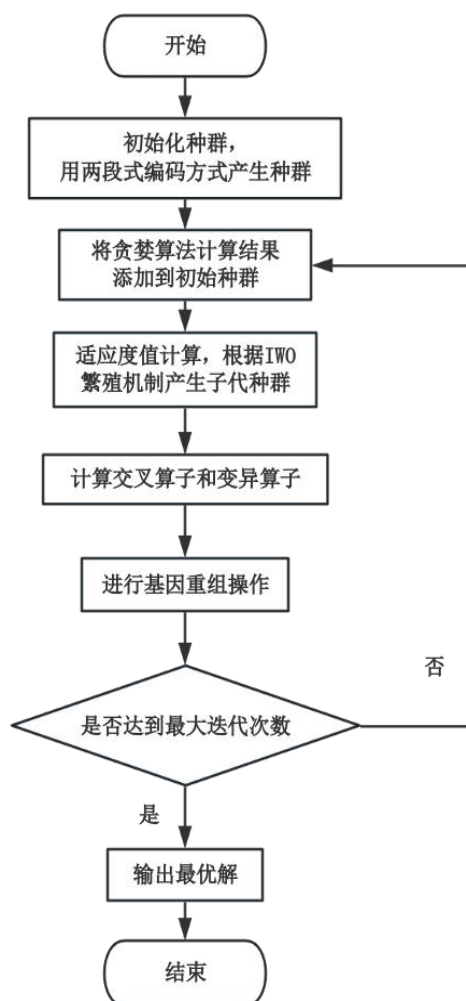


图 3.7 IPGA-G 算法流程图

Figure 3.7 IPGA-G algorithm flow chart

3.4 仿真实验与结果分析

本章为了验证 IPGA-G 算法对动态优先级确定性变化下多无人机协同任务分配问题的有效性, 选择贪婪算法和 IPGA 算法进行对比实验。在仿真实验的硬件环境中, 计算机处理器为 Intel(R) Corei5, 主频为 2.50GHz, 仿真实验使用软件 Matlab R2019b 编译。

实验设定为一个 50×50 平方千米的区域, 用坐标来表示各个接收点位置, 派送中心坐标为(25, 25)。实验区域按照子区域 $G_1(x > 25, y > 25)$, 子区域 $G_2(x > 25, y < 25)$, 子区域 $G_3(x < 25, y < 25)$, 子区域 $G_4(x < 25, y > 25)$ 进行子区域划分。为了模拟各种不同情况, 避免接收点设置的过于集中或者过于分散, 每组实验在实验区域内随机生成 10 个接收点, 根据接收点的坐标确定每个接收点所在的子区域, 每个子区域设固定的需求状态的变化率 ξ 值, 数据如表 3.1 所示。

设 4 架无人机 $U_N = \{u_1, u_2, u_3, u_4\}$ 从派送中心出发执行物资配送任务, 每架无人机平均飞行速度 $V_{AVG} = 11\text{km/h}$, 最大飞行距离 $R_{\max} = 120\text{km}$ 。设装货时间 $T_{load} = 0.5\text{h}$, 卸货时间 $T_{unload} = 0.1\text{h}$, 每个接收点单次接收固定重量物资 $L_0 = 2.5\text{kg}$, 设物资消耗水平系数 $c = 2$, 任务奖励函数参数 $P_1 = 200$, $P_2 = 180$, 单位距离运输成本 $d = 3.5$, 每架无人机的固定成本 $F_n = 80$, 本章所有实验设置无人机总共进行 4 轮任务分配。

3.4.1 关键参数分析实验

本节对决定需求状态变化时间最关键的参数周期 T 和界限值 W_0 进行对比测试, 找出最合适的周期和界限值来更好地体现出不同算法完成动态优先级确定性变化问题的性能差异。设置周期 $T = 4$, 界限值 $W_0 = 2$ 为第一组参数。根据每个子区域的需求状态的变化率, 可以求出子区域 G_z 在每轮周期的需求状态变化的时间, 如表 3.2 所示。

表 3.2 不同子区域需求状态变化时间表($T=4, W_0=2$)

Table 3.2 Schedule of demand status change in different sub-regions ($T=4, W_0=2$)

子区域	第一轮	第二轮	第三轮	第四轮
G_1	1.58	5.58	9.58	13.58
G_2	1.12	5.12	9.12	13.12
G_3	0.88	4.88	8.88	12.88
G_4	0.79	4.79	8.79	12.79

设置周期 $T = 4$, 界限值 $W_0 = 3$ 为第二组参数, 此时不同子区域需求状态变化时间如表 3.3 所示。

表 3.3 不同子区域需求状态变化时间表($T=4, W_0=3$)

Table 3.3 Schedule of demand status change in different sub-regions ($T=4, W_0=3$)

子区域	第一轮	第二轮	第三轮	第四轮
G_1	2.37	6.37	10.37	14.37
G_2	1.68	5.68	9.68	13.68
G_3	1.32	5.32	9.32	13.32
G_4	1.19	5.19	9.19	13.19

设置周期 $T=4$ ，界限值 $W_0=4$ 为第三组参数，此时不同子区域需求状态变化时间如表 3.4 所示。

表 3.4 不同子区域需求状态变化时间表($T=4, W_0=4$)

Table 3.4 Schedule of demand status change in different sub-regions ($T=4, W_0=4$)

子区域	第一轮	第二轮	第三轮	第四轮
G_1	3.16	7.16	11.16	15.16
G_2	2.24	6.24	10.24	14.24
G_3	1.75	5.75	9.75	13.75
G_4	1.58	5.58	9.58	13.58

由三组参数产生的需求状态变化可知，在周期相同的时候， $W_0=2$ 组每个子区域之间的平均状态变化时间差为 0.26 为 3 组最小， $W_0=4$ 组每个子区域之间的平均状态变化时间差为 0.53 为 3 组最大。 $W_0=2$ 组四个子区域平均需求状态变化时间为每个周期开始后的 1.09h 为三组中最小， $W_0=4$ 组四个子区域平均需求状态变化时间为每个周期开始后的 2.18h 为三组中最大。

用贪婪算法测试三组参数的实验结果分布情况，设置 100 组实验，每组实验随机生成接收点，用这些接收点对三组参数进行测试，记录每组实验三种参数对应的实验结果，生成不同界限值 W_0 的 100 组实验结果箱形图，如图 3.8 所示。

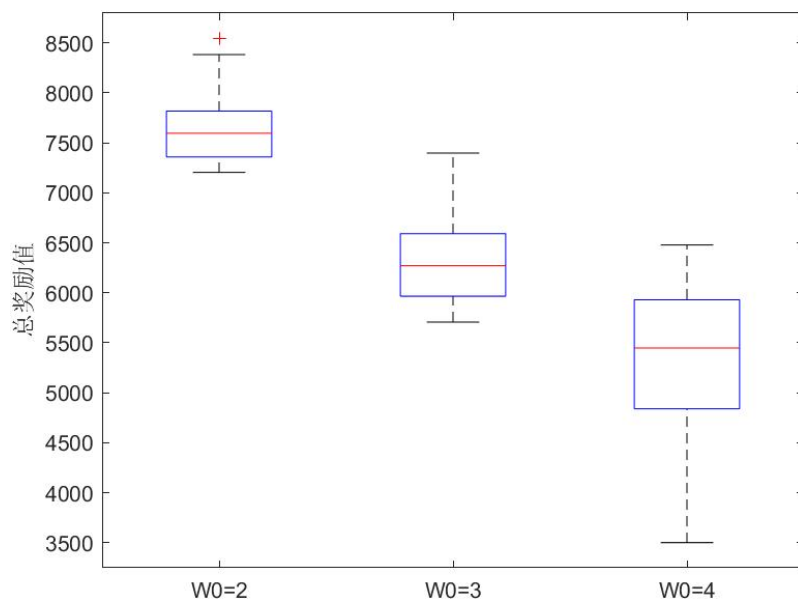


图 3.8 不同界限值 W_0 实验结果对比箱型图

Figure 3.8 Box plots of different boundary values W_0

由不同界限值 W_0 进行 100 组实验结果生成的箱型图可知, 在相同周期 $T=4$ 条件下, $W_0=2$ 时总奖励值相对过于集中, 且奖励值过高, $W_0=4$ 时总奖励值相对过于分散, 同时出现奖励值过低的情况。 $W_0=2$ 时因为四个子区域平均状态改变时间差较小体现不出动态优先级变化的影响, 所以会产生结果相对过于集中且总奖励值偏高的情况。 $W_0=4$ 时四个子区域平均需求状态变化过晚会造成总奖励值出现过低的情况。根据上述分析, $W_0=3$ 适合作为符合实验条件的固定界限值, 其实验结果上下限均衡, 结果分布相对适中。

同理, 对周期的参数值进行实验测试, 在 $W_0=3$ 的情况下分别用 $T=3.5$, $T=4$, $T=4.5$ 三个周期进行 100 组实验, 每组实验随机生成接收点。记录 100 组实验三种参数对应的实验结果, 生成不同周期 T 的实验结果箱形图, 如图 3.9 所示。

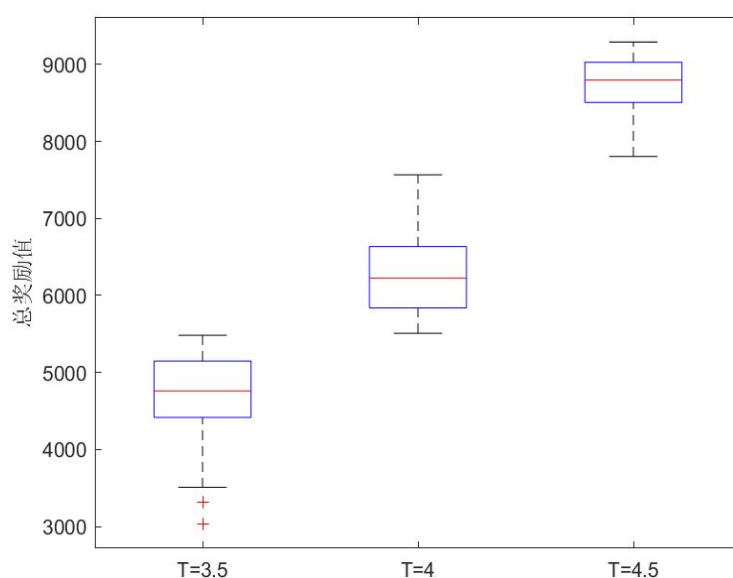


图 3.9 不同周期 T 实验结果对比箱型图

Figure 3.9 Box plots of results from different period T

根据 100 组实验的结果来看, $T=3.5$ 时总奖励值普遍过低, $T=4.5$ 时总奖励值普遍较高, 相比于另外两个参数 $T=4$ 时的实验结果相对稳定且总奖励值处在合适的范围之内。由于多无人机任务分配总共进行四轮, 不同周期下无人机每轮获得的奖励值的差值会进行累加, 造成不同周期下的总奖励值相差较大的情况出现。

综合上述实验结果和结果分析, $T=4$, $W_0=3$ 是动态优先级确定性变化下的问题模型最合适的周期和界限值, 可以作为决定不同子区域需求状态变化函数的固定参数, 能得到更合理的总奖励值, 更好地进行后续实验来体现出不同算法完成动态优先级确定性变化下多无人机协同任务分配问题的性能差异。

3.4.2 算法对比实验

本节进行相同场景下不同算法能力对比实验，在区域内随机产生一组接收点作为这个场景的固定设置，接收点详细信息如表 3.5 所示。

表 3.5 接收点信息表

Table 3.5 Receive point information sheet

接收点序号	1	2	3	4	5	6	7	8	9	10
x 轴坐标	43.6	27.2	39.9	12.8	3.8	6.7	22.0	14.4	43.2	30.4
y 轴坐标	33.8	22.5	42.8	32.3	47.1	15.4	7.2	16.1	9.3	13.4
子区域编号	1	1	1	2	2	3	3	3	4	4

首先通过贪婪算法求得一种可行解，表 3.6 将贪婪算法整个求解过程表示了出来，不同数字代表不同的接收点序号，数字 0 用来表示派送中心。无人机在执行完全部 10 个接收点的任务后，从派送中心出发进行新一轮任务分配，通过贪婪算法对无人机进行了四轮分配过后得到的总奖励值为 5876.17。

表 3.6 贪婪算法求解过程

Table 3.6 Greedy algorithm solving procedure

第一轮										
UAV1	2	2	2	2	10	7	7	8	6	6
UAV2	0	1	1	1	1	1	1	1	1	9
UAV3	0	0	4	4	4	4	5	5	5	5
UAV4	0	0	0	3	3	3	3	3	3	3
第二轮										
UAV1	0	0	0	0	0	10	10	10	7	7
UAV2	0	0	0	0	8	8	8	6	6	6
UAV3	0	0	4	4	4	4	5	5	5	5
UAV4	2	1	1	3	3	3	3	3	3	9
第三轮										
UAV1	0	0	0	4	4	4	5	5	5	5
UAV2	0	1	1	1	3	3	3	3	3	3
UAV3	2	2	10	10	10	7	7	8	6	6
UAV4	0	0	0	0	0	0	0	0	0	9

续表 3.6 贪婪算法求解过程

Table 3.6 Greedy algorithm solving procedure(continued)

第四轮										
UAV1	0	0	4	4	4	4	5	5	5	5
UAV2	2	1	1	3	3	3	3	3	3	3
UAV3	0	0	0	0	8	8	8	6	6	6
UAV4	0	0	0	0	0	10	10	10	7	9

将 IPGA-G 算法与 IPGA 算法分别用于此场景下进行求解,对实验结果总奖励值进行比较。设置 IPGA-G 算法参数 $k_1=0.7$, $k_2=0.4$, $k_3=0.1$, $k_4=0.05$, IPGA-G 算法根据这些参数进行自适应变异算子和自适应交叉算子的更新。第一组实验设置两种算法初始种群大小都为 50,迭代次数为 100,重复进行 50 轮实验用来测试算法的稳定性,记录 50 轮实验的实验结果,生成展示实验结果分布情况的箱式图,如图 3.10 所示。

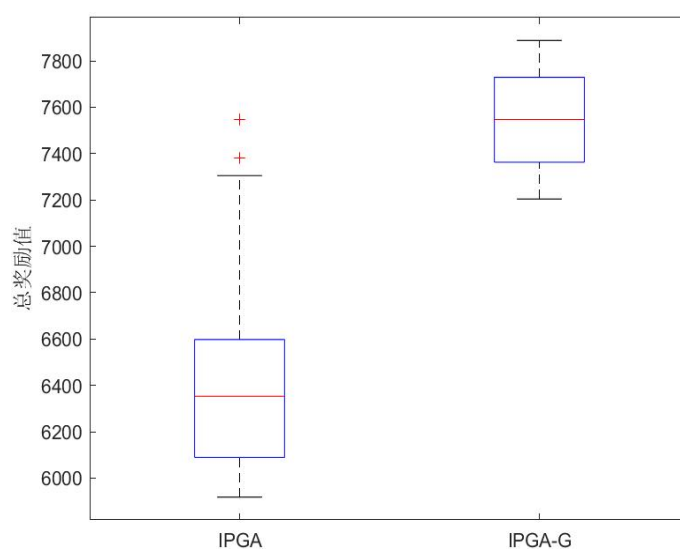


图 3.10 种群大小为 50 时实验结果对比箱型图

Figure 3.10 Experimental results comparing box plots when the population size is 50

为了更好地分析两种算法的能力,研究种群大小对两种算法的影响,使用不同种群大小进行多组实验。第二组实验设置两种算法初始种群大小都为 100,迭代次数为 100,重复进行 50 轮实验用来测试算法的稳定性,记录 50 轮实验的实验结果,生成展示实验结果分布情况的箱式图,如图 3.11 所示。

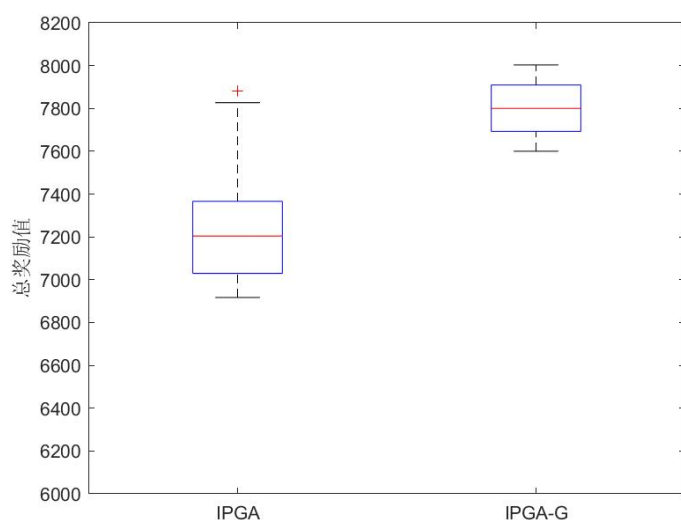


图 3.11 种群大小为 100 时实验结果对比箱型图

Figure 3.11 Experimental results comparing box plots when the population size is 100

根据两组实验结果表明,相同场景中 IPGA-G 算法求出的平均总奖励值明显高于 IPGA 算法的平均总奖励值,初始种群越小时平均总奖励值差距越大,并且 IPGA-G 算法的实验结果的分布相比于 IPGA 算法更为集中,上下限差距更小。

选取两个算法在取得中位数总奖励值时的对应实验作为参考,得到两种算法的收敛曲线对比图,如图 3.12 所示。

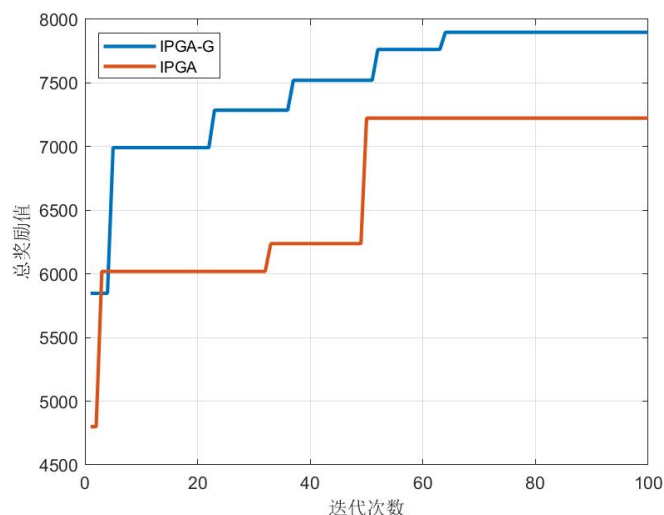


图 3.12 收敛曲线对比图

Figure 3.12 The comparison diagram of the convergence curve

根据收敛曲线对比图可知,IPGA-G 算法收敛速度明显快于 IPGA 算法,在算法迭代过程中后期仍然具有较强的搜索能力。然后对两个算法的计算时间进行分析,实验设置与第二组设置相同,记录 50 轮每个算法每次迭代过程求解得到或超过奖励值 6904.57(IPGA 算法的最差结果)时所花费的时间,求出 50 轮实验的平均

值，实验结果如图 3.13 所示。

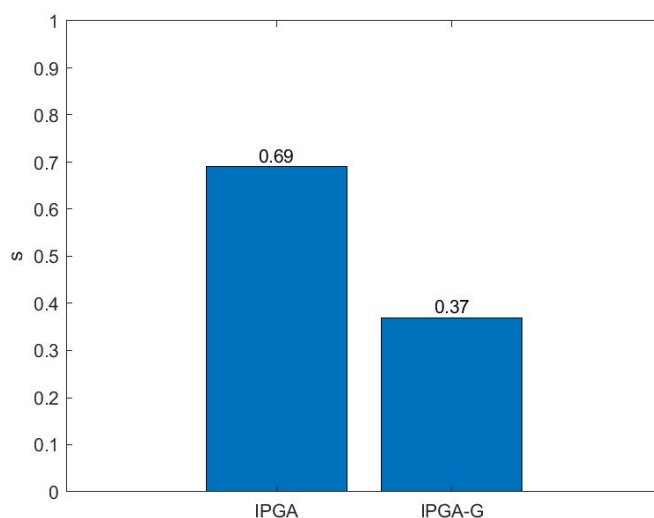


图 3.13 不同算法平均计算时间

Figure 3.13 Average calculation time across the different algorithms

实验结果表明，相比于 IPGA 算法，IPGA-G 算法能在更短的时间内取得相同的奖励值，IPGA-G 能够在更少的时间内完成达到设定目标总奖励值(6904.57)的收敛。

最后测试每种算法在不同场景下的实验结果情况，设置 50 组实验，每组实验在实验设定区域内随机生成 10 个接收点的坐标，用这组接收点对三种算法进行测试，设置 IPGA 和 IPGA-G 算法初始种群大小都为 100，迭代次数为 100 次，实验结果如图 3.14 所示。

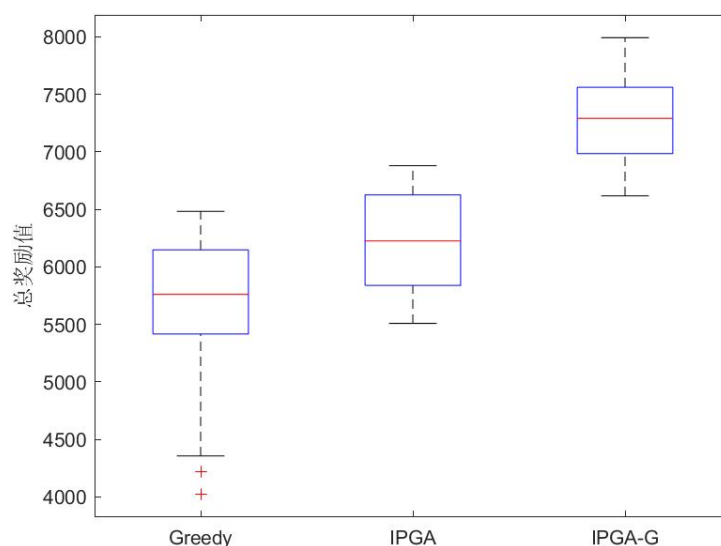


图 3.14 50 组不同场景下实验结果箱型图

Figure 3.14 Boxplots of experimental results in 50 sets of different scenarios

根据仿真实验所得结果得出结论, IPGA-G 算法在不同场景下解得最优解平均总奖励值远超过 IPGA 算法和贪婪算法的最优解平均值, 说明 IPGA-G 算法不论接收点的分布情况如何, 都能搜索出高质量的最优解, 具有很强的适应性。

综上所述, 本文提出的 IPGA-G 算法能够有效地处理多无人机在动态优先级确定性变化时的协同任务分配问题, 并且在收敛速度、全局搜索能力和运算稳定性方面具有明显优势。

3.5 本章小结

本章研究了多无人机协同任务分配在动态优先级确定性变化下问题的求解和遗传相关算法的改进方式。首先, 基于第二章对动态优先级的描述构建了动态优先级确定性变化的多无人机协同任务分配模型, 用函数描述了急需程度的变化, 采用周期性函数来表示子区域 G_z 里接收点的需求状态。其次, 本章设计了贪婪算法解决此问题的流程, 通过递进的循环和选择得到一个可行解。然后, 本章分析了遗传算法, 提出适用于解决此问题的改进单亲遗传-贪婪结合算法, 设计符合问题模型的编码方式, 添加无人机相应约束, 设计自适应变异算子和自适应交叉算子, 将贪婪算法结合到算法中。最后进行仿真实验, 对界限值和周期参数设定进行分析, 然后进行算法的重复多次对比实验。实验结果表明, 本文提出的改进单亲遗传-贪婪结合算法具有优秀的搜索能力和搜索效率, 在处理多无人机协同任务分配在动态优先级确定性变化问题时, 具有很高的稳定性, 能够迅速高效地找到更高质量的最优解。

4 动态优先级不确定性变化下多无人机任务分配算法研究

本章主要针对解决动态优先级不确定性变化下多无人机任务分配问题,对群体智能算法中的粒子群算法进行了展开研究,提出自适应搜索模式切换粒子群算法用于寻找动态优先级不确定性变化问题的最优解。本章首先对不确定性问题进行分析,针对目前相关研究出现的不足设计合理的模型,用泊松分布来进行描述需求状态的不确定性变化,对不同的子区域的急需程度进行表示。其次,本章设计了用于解决动态优先级不确定性变化下多无人机任务分配问题的基于一致性的捆绑拍卖算法。然后,本章提出符合本章问题模型的自适应搜索模式切换粒子群算法,设计与问题对应编码和解码方式,通过一种新的粒子速度和位置更新机制(搜索模式切换学习策略)在每次迭代中根据搜索模式自适应地选择加速系数和更新速度模型,从而有助于对问题空间的彻底搜索。最后,进行仿真实验对比测试不同算法在处理不确定性变化问题时的性能,选择前文中解决确定性变化问题的算法作为对比算法,与本章提出的算法进行对比。实验结果表明,本章提出的自适应搜索模式切换粒子群算法具有较强的最优解搜索能力,能够跳出局部最优持续进行对最优解的搜索,能更有效地解决多无人机在动态优先级不确定性变化下的多无人机协同任务分配问题。

4.1 不确定性问题分析

在实际多无人机协同运行过程中,考虑不确定性对于高效可靠地分配任务至关重要。然而,许多现有的方法,尤其是分布式方法在处理不确定性方面能力有限,而且往往依赖于过于简单的环境假设。不确定性可能来自传感器不准确、智能体故障^[59]、环境干扰等^[45]。以往的研究表明,事先考虑可靠性至关重要,因为忽视故障的可能性会导致性能下降^[76]。例如,基于异步共识的捆绑算法在通信过程不确定的环境中,尤其是在大量智能体执行任务的情况下,任务分配效率相对较低,这也是该算法的理论性能与更现实的场景相比存在差异的原因^[67]。在使用智能优化技术时,很难将不确定性问题纳入与时间相关联的任务分配任务中,因为不确定性需要在系统中表示,而通常有些不确定性无法通过分析来表达,问题的维度也会增加很多,因此计算成本也会增加^[59]。由于无人机任务分配的复杂性,特别是在考虑到现实生活实际场景的情况下,不确定性问题是复杂多变的,此时

在解决各种不确定性问题的时候就需要用到不同的算法。

在基于拍卖的相关算法中，PI 算法对于截止期较紧的概率任务分配似乎比 CBBA(Consensus-Based Bundle Algorithm, CBBA)算法有更好的性能，但算法的鲁棒性，特别是 PI 算法的鲁棒性暂时还没有得到很好的研究。通常情况下，鲁棒性对于时间要求苛刻相关问题的应用是必不可少的，如合成孔径雷达或对多无人机执行任务失败率要求较低的军事方面应用^[59]。根据 PI 算法提出了不同的鲁棒性模型，使用期望值和最坏情况度量的组合来处理任务成本的不确定性，使用概率抽样，假设任务分配过程的不确定变量服从高斯分布，Whitbrook 等人^[59]得到的结论是，基准 CBBA 比基准 PI 更稳健，因为基准 PI 算法不能很好地处理不确定性，当加入鲁棒性方案时，两种算法都具有更好的性能，其中 PI 的性能优于 CBBA，而解的质量不受鲁棒性模块的影响。此外，在处理不确定性方面，带有鲁棒性模块的 CBBA 比 Ponda 等人^[77]的基准稳健 CBBA 更好地处理不确定性。然而，可伸缩性仍然是一个问题，因为鲁棒的 PI 比基准版本具有更高的计算成本。Liu 等人^[78]假定无人机执行任务的持续时间是未知的，因为在实际环境中，每个任务持续时间通常可以被认为是一个随机过程，因此提出了一种基于马尔可夫决策过程的稳健 CBBA 扩展，通过结合马尔可夫决策过程的期望值来变换奖励函数，在保持收敛的同时提高了稳健性。该方法还具有比基准 CBBA 算法更好的性能，比基准稳健 CBBA 有更好的性能和更低的计算代价，但由于 MDP 的构建模型计算量大，不适合于像稳健基准 CBBA 这样的大规模系统，所以该方法的计算代价高于基准 CBBA 算法。

在无人机辅助的真实移动边缘计算环境中，由于任务的随机性和移动性，任务具有不确定性。Fu 等人^[79]提出了一种基于可能环境的贝叶斯联盟博弈，该博弈具有信念更新机制，该策略获得与任务不确定性相关的环境的概率，由于任务的不确定性与环境分类有关，如果增加环境分类中的细节，这将导致更好的优化联盟结构，但同时也存在增加复杂性的缺点。

为了研究多架无人机攻击地面目标的任务分配和航迹规划相关问题，当目标在移动时，存在不确定性，使得战场环境更加复杂。Xia 等人^[38]提出了一种用于协同路径规划的自适应参数调整与双向搜索算法和一种新的基于制导机制的改进粒子群优化算法(Guidance Mechanism Particle Swarm Optimization, GMPSO)，并提出

了一种针对时敏不确定性的在线任务重分配方法,保证了当运动目标状态发生变化时,无人机的航迹被重新规划并重新完成任务分配。

在环境不确定的情况下,Zhao 等人^[80]研究了无人机对目标的监视和攻击问题,如当地风速和降雨量不确定的天气。Zhao 等人提出了一种基于强化学习 Q-learning 方法的任务分配算法,该算法在不确定条件下比差分进化-离散粒子群-郭涛-模拟退火融合算法具有更好的性能。Gregory 等人^[76]的研究调查了不确定性对多智能体系统的影响(根据任务分配组合的失败情况),利用非马尔科夫状态和启发式方法。通过研究发现,简化假设的时候,例如使用马尔可夫状态,可能会导致系统性能的不表示不准确。除此之外,还证明了更复杂的启发式方法可以更好地描述物理环境和不确定性,从而提高某些问题类别的性能。

综上所述,目前考虑不确定性的研究数量较多,国内外学者研究方式多是通过动态规划、模糊规划、随机规划或鲁棒性优化等方法解决不确定性问题,也有一部分研究将不确定问题具体情况通过罚函数和客户满意度等形式进行表现。在近年来相关的不确定性任务分配研究普遍存在两点问题:

(1) 分析问题对象单一化,比如仅对问题对象数据进行分析,缺少对某一类事件进行分析和预测。相关多智能体不确定性任务分配问题常采用及时配送的问题形式,任务种类多样但数量有限,如果按单一问题对象的需求进行分配任务可能不能够满足现实场景急需程度变化,在现实场景出现物品的需求量增多或减少变化的情况会产生更多的消耗,甚至导致任务失败,达不到所提出的要求。

(2) 在现有研究的求解过程中,往往根据数据等信息的变动情况来预测未来,然而在这一阶段,预测会产生偏差,对减少这种预测产生偏差的影响研究较少。如日常生活中出现一些特殊情况,比如在雨天会增加人们对雨伞的急需程度,对雨伞急需程度的变动就存在不规则的变化规律。

因此,构建不确定性问题模型需要提高许多算法性能的要求,并在效率、鲁棒性和收敛时间之间需要权衡。所以在设计不确定性问题的过程中必须充分考虑这种平衡,同时考虑可用算法计算能力和每个分析对象的具体要求。

4.2 动态优先级不确定性变化下问题描述

在实际配送任务中,客户急需程度具有不确定性,无人机在派出之前不可预知即将配送客户的急需程度,因此要建立需求预测模型。本文建立的任务分配模

型有一定的应用范围，是因为多无人机任务分配问题本身存在无人机性能约束，适用于急需程度的变动有一定的规律且需求程度变化量较小的情况，这样在满足约束的条件下，使任务分配有意义也提高了总奖励值。

本文采用动态需求变化问题相关研究经常使用的泊松分布来进行模拟^[81]。本文根据情况，以泊松分布模拟急需程度的变化情况对动态优先级不确定性任务进行分配，减少这种预测偏差产生的影响，设计了一种动态优先级不确定性变化下多无人机协同任务分配模型。在当子区域的需求变化不能够确定时：泊松分布应用设定子区域 G_z 的需求状态随时间变化的函数见公式(2.2)。

设定函数中的 $\lambda=5$ ，物资消耗水平系数 $c=2$ ，得到的函数图像如图 4.1 所示。

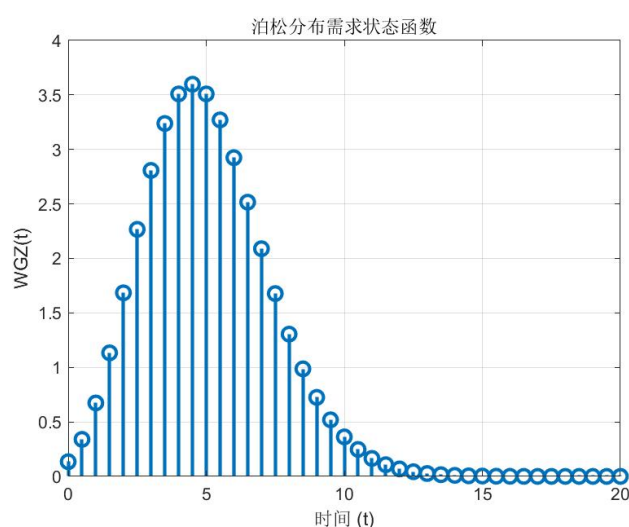


图 4.1 泊松分布需求状态函数

Figure 4.1 Poisson-distributed demand state function

4.3 基于一致性的捆绑拍卖算法应用设计

4.3.1 基于一致性的捆绑拍卖算法概述

基于一致性的捆绑拍卖算法(Consensus-Based Bundle Algorithm, CBBA)是 Choi 等人提出的一个用于解决任务分配相关问题的算法，由相互分离的两个阶段组成：任务包构建阶段和冲突消解阶段^[44]。CBBA 算法近年来多次被用于研究多无人机任务分配相关问题，可以通过与相邻无人机通信来消除冲突，能够适应多无人机系统，算法具有快速高效和系统通信量低等优点^[78]。在解决多无人机任务分配问题时 CBBA 算法的任务包构建阶段和冲突消解阶段大致过程如下：

(1) 任务包构建阶段：每架无人机先贪婪地生成其任务包和任务执行序列，直到所有任务均被分配完毕，各架无人机首先计算所有未添加进自身任务包的任

务能够产生的奖励，从中选出奖励最大的任务，然后与当前优胜列表中该任务的标价进行比较，当其收益大于此时优胜标价时将该任务添加到自身任务包。

(2) 冲突消解阶段：各架无人机与相邻无人机进行通信，利用优胜标价列表、优胜无人机列表，以及时间戳向量来消除冲突。优胜标价列表和优胜无人机列表的更新使各架无人机的任务包和任务执行序列产生变更。

算法在任务包构建阶段和冲突消解阶段循环迭代直到所有无人机就非冲突的任务分配方案达成一致，迭代就会结束。

4.3.2 基于一致性的捆绑拍卖算法在本问题中的应用设计

CBBA 算法已经广泛用在无人机任务分配相关问题研究，但是具体用于新问题，需要根据条件进行一定的细化设计，本章对用于解决动态优先级不确定性变化下多无人机任务分配问题的 CBBA 算法进行了如下设计。将 CBBA 算法用到的相关符号加以说明，在多无人机协同任务分配过程中无人机 u_i 需要存储和更新信息如下：

(1) 无人机任务包：无人机任务包定义为 $B_i = \{b_{i1}, b_{i2}, \dots, b_{i(n_b)}\}$ ，其中 n_b 表示当前竞拍得到的任务数， b_{in} 表示无人机 u_i 竞拍得到的第 n 个任务，如 $b_{21}=4$ 表示无人机 u_2 竞拍得到的第 1 个任务是序号为 4 号接收点的任务。无人机 u_i 当前被分配任务包的长度为 n_b ，设无人机 u_i 能够执行的最多任务数为 n_{ik} ，满足 $n_b \leq n_{ik}$ 。任务包表示无人机 u_i 通过依次竞拍得到的任务序列， $b_i = \phi$ 表示无人机 u_i 目前未竞拍得到任务。

(2) 无人机路径集合：无人机路径集合为 $P_i = \{p_{i1}, p_{i2}, \dots, p_{i(n_b)}\}$ ， n_b 表示当前竞拍得到的任务数，无人机路径集合 P_i 是无人机 u_i 根据任务包 b_i 得到的满足最大奖励的任务执行顺序，执行顺序为： $p_{i1} \rightarrow p_{i2} \rightarrow \dots \rightarrow p_{i(n_b)}$ 。

(3) 优胜无人机集合：优胜无人机集合定义为： $Z_i = \{z_{i1}, z_{i2}, \dots, z_{in}\}$ ， z_{in} 表示无人机 u_i 和相邻无人机通过信息交换得到的前往接收点 M_n 执行任务的优胜无人机信息，如 z_{31} 表示无人机 u_3 通过信息交换得到无人机 u_1 在竞拍中获得前往接收点 M_n 的信息， $z_{in} = \phi$ 表示无人机 u_i 已知前往接收点 M_n 的任务未被竞拍。

(4) 优胜标价集合：优胜标价集合定义为 $Y_i = \{y_{i1}, y_{i2}, \dots, y_{in}\}$ ， y_{in} 表示当前各架无人机对前往接收点 M_n 的最大标价值，如果当前前往接收点 M_n 的任务没有被任何无人机竞拍到，此时 $y_{in}=0$ 。

(5) 时间戳集合: 时间戳集合定义为 $S_i = \{s_{i1}, s_{i2}, \dots, s_{in}\}$, s_{in} 表示无人机 u_i 最近一次从相邻无人机获得信息交换的时间, 时间戳向量是冲突消解的重要参数, 表示了无人机 u_i 获取信息的新旧程度。

无人机 u_i 按照路径集合 P_i 中的任务序列执行任务获得的奖励值表示为 $R_i^{P_i}$, 当无人机任务包 B_i 内插入新任务 j 时, $R_i^{P_i \oplus_x \{j\}}$ 表示无人机 u_i 将任务 j 插入到任务路径集合 P_i 的第 x 个位置后无人机 u_i 得到的总奖励值。 $C_{ij}[B_i]$ 表示无人机 u_i 添加任务 j 到任务包 B_i 产生的奖励值提高量, 如公式 4.3 所示。

$$C_{ij}[B_i] = \begin{cases} \max_{n \leq |P_i|+1} R_i^{P_i \oplus_x \{j\}}, & j \notin B_i \\ 0, & j \in B_i \end{cases} \quad (4.3)$$

第一阶段: 任务包构建阶段

CBBA 算法在解决本章问题时任务包构建阶段的步骤如下:

步骤 1: 每架无人机首先按照一定的顺序生成一个本地的任务包, $i=j=n=1$ 。

步骤 2: 检查无人机 u_i 的任务包是否满足条件 $n_b \leq n_{ik}$, 若满足执行步骤 3, 若不满足, 则返回步骤 1。

步骤 3: 检查当前无人机路径集合 P_i 是否包含了接收点 M_n 的任务, 如果包含则检查下一个接收点的任务执行步骤 7, 如果不包含则转入步骤 4。

步骤 4: 对于每一个插入位置 x 计算无人机 u_i 添加任务 j 到任务包 B_i 产生的奖励值提高量 $C_{ij}[B_i]$ 。

步骤 5: 此插入位置是否可行进行判定, 如果可行则确认此位置信息执行步骤 6, 否则 $j=j+1$ 然后返回步骤 4。

步骤 6: 比较插入路径集合 P_i 各位置的奖励值提高量, 找出提高量最大的插入位置, 记录位置序号, 奖励值提高量和按当前路径集合 P_i 执行任务的总奖励值。

步骤 7: 更新执行接收点 M_n 任务的序号, $n=n+1$, 如果全部接收点都被检查, 执行步骤 8, 否则返回步骤 3。

步骤 8: 进行无人机 u_i 的信息更新, 更新共享信息, 任务包 B_i 、任务路径 P_i , 优胜标价集合 Y_i 和优胜无人机集合 Z_i 。

步骤 9: 更新无人机序号 $i=i+1$, 如果达到无人机总数执行步骤 10, 否则返回步骤 2。

步骤 10: 结束任务包构建阶段。

第二阶段：冲突消解阶段

在任务包构建完成后，进行无人机冲突消解阶段。无人机 u_i 与其相邻无人机 u_j 交换优胜标价集合 Y_i ，优胜无人机集合 Z_i 和时间戳向量 S_i 信息。在得到信息后，无人机 u_i 进行比较并判断能否优胜中标，如果无人机 u_i 奖励更高，将执行接收点 M_n 的任务。其他在任务包集中存在接收点 M_n 的任务的无人机将会放弃该任务，路径集合中在接收点 M_n 的任务之后的任务也会被放弃，在下一次任务包构建过程进行重新分配。当相邻无人机将相关信息传递给无人机 u_i 之后，无人机 u_i 对信息进行比较，并按照任务更新机制进行更新，任务更新机制包括更新、保留和重置三种操作^[44]。

(1) 更新操作： $y_{in}=y_{jn}$ ， $z_{in}=z_{jn}$ 。

(2) 保留操作： $y_{in}=0$ ， $z_{in}=\phi$ 。

(3) 重置操作： $y_{jn}=y_{jn}$ ， $z_{jn}=z_{jn}$ 。

如果标价集合在冲突消解阶段被修改，每架无人机判断更新或重置的任务是否在其任务包里，若在任务包里，在任务包内删除该任务及其后面的任务，因为后面任务的奖励值发生了变化。若 n_0 是无人机 u_i 任务包 B_i 中第一次出现删除的任务所在的位置，则满足条件 $n \geq n_0$ 的任务 b_{in} 对应的 Y_i ， Z_i 重置， $y_{in}=0$ ， $z_{in}=\phi$ ， $b_{in}=\phi$ 。冲突消解解决在分配过程任务冲突后，返回任务包构建阶段继续添加任务，将在冲突消解阶段删掉的任务进行重新分配。当优胜标价集合 Y_i ，优胜无人机集合 Z_i 不在发生变化时，冲突消解过程结束，各架无人机分配了互相无冲突的任务包集合，完成对各接收点执行任务的任务分配。

4.4 自适应搜索模式切换粒子群算法设计

4.4.1 粒子群算法概述

粒子群优化算法(Particle Swarm Optimization, PSO)是一种模拟鸟类群体捕食活动来解决优化问题的群体智能算法^[82]。根据个体对环境的适应程度，群体中的个体向更好适应的区域方向移动。它使用粒子来表示每个优化问题的解决方案，每个粒子的方向和距离由位置矢量和速度矢量决定，每个粒子的适应度函数值由相应的适应度函数决定。粒子通过搜索得到的局部最优粒子和全局最优粒子不断更新迭代，在解空间中进行迭代搜索，以寻找最优解。如图 4.2 所示，图中描述了粒子运动的示意图，粒子自身速度用 v 表示，当前个体最优用 Pb 表示，当前全局

最优用 Gb 表示。粒子在此刻迭代中以自身速度、向个体最优解运动速度和向全局最优解运动速度三个矢量和可以得到粒子在本次迭代后的速度 v' 。

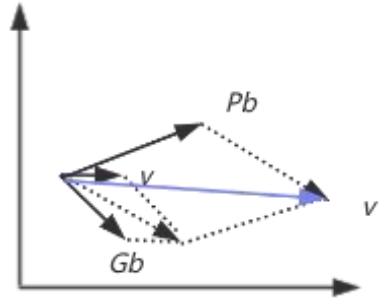


图 4.2 粒子运动示意图

Figure 4.2 Schematic representation of the particle motion

假设 D 维空间中有 i 个粒子, 每个粒子的位置为 $X_i = (x_i^1, x_i^2, \dots, x_i^D)$, 粒子的速度为 $V_i = (v_i^1, v_i^2, \dots, v_i^D)$, 在算法迭代过程中, 这两个向量分别按公式(4.2)和公式(4.3)进行更新。

$$V_i(t+1) = \omega \cdot V_i(t) + c_1 r_1 (pbest_i(t) - X_i(t)) + c_2 r_2 (gbest(t) - X_i(t)) \quad (4.2)$$

$$X_i(t+1) = X_i(t) + V_i(t) \quad (4.3)$$

其中, $pbest_i(t)$ 是第 i 个粒子搜索到的个体最佳, 而 $gbest_i(t)$ 是整个种群搜索到的迄今为止的全局最佳。 c_1 和 c_2 是个体学习因子和全局学习因子, 两个加速系数协调保持个体经验和社会经验之间的权衡。 r_1 和 r_2 是两个在 $(0, 1)$ 范围内的随机数, ω 是惯性权重。

粒子群优化算法具有搜索质量高、搜索速度快、参数可调整等优点, 但在处理离散问题时容易陷入局部最优解问题。现实中的许多优化问题都涉及连续和离散决策变量, 这类优化问题统称为混合变量优化问题(Mixed Variable Optimization Problem, MVOP)。MVOP 的混合决策变量增加了搜索空间的复杂性, 使其难以求解。许多现有的粒子群优化算法变体已经被提出来解决连续或离散优化问题, 这使得解决 MVOP 变得可行且有前景, 这些粒子群算法的变体可以采用自适应参数调整策略和约束处理方法来提高算法的整体效率。

4.4.2 自适应搜索模式切换粒子群算法在本问题中的应用设计

本章提出一种自适应搜索模式切换粒子群算法 (Adaptive Search-Mode Switching Particle Swarm Optimization, ASSPSO), 以提高粒子群算法的搜索性能,

本算法设计了一种新的速度更新机制，如公式(4.4)所示，迭代进行了 k 次时粒子 i 的速度和位置根据公式(4.4)进行更新。

$$\begin{aligned} v_{ij}(k+1) &= \omega(k)v_{ij}(k) + c_1(\varphi(k))r_1(p_{best}^j(\varphi(k)) - x_{ij}(k)) + c_2(\varphi(k))r_2(g_{best}^j(\varphi(k)) - x_{ij}(k)) \\ x_{ij}(k+1) &= x_{ij}(k) + v_{ij}(k+1) \end{aligned} \quad (4.4)$$

其中 j 表示种群空间的维度， $\omega(k)$ 表示惯性权重， $c_1(\varphi(k))$ 和 $c_2(\varphi(k))$ 分别表示粒子根据四种搜索模式 $\varphi(k)$ ($k \in 1, 2, 3, 4$) 确定的个体加速系数和全局加速系数。

$p_{best}^j(\varphi(k))$ 和 $g_{best}^j(\varphi(k))$ 表示根据搜索模式 $\varphi(k)$ 确定的个体学习粒子位置和全局学习粒子位置。

为了更高效地找到最优解，ASSPSO 算法对适应度函数进行改进，针对本章的动态优先级不确定性变化下的多无人机协同任务分配问题，将无人机任务的均衡度加到适应度函数中，加入均衡度能够让每架无人机执行任务路线尽可能保持均衡，避免出现其中某一条任务路线过长且任务负载过重或者路线过短任务负载过轻的情况。在无人机总数量为 N 架的情况下，均衡度函数如公式(4.5)所示。

$$Q(i) = \frac{\max\{F_x - F_y\}}{\max\{F_x\}}, \quad x, y \in [1, N], x \neq y \quad (4.5)$$

其中， $Q(i)$ 表示均衡度值， $\max\{F_x\}$ 表示获得最大奖励值的无人机的奖励值， $\max\{F_x - F_y\}$ 表示两架无人机之间获得奖励值的最大差值。均衡度值 $Q(i)$ 越接近于 0 说明无人机之间任务分配均衡度高，路线之间差距不大；均衡度值 $Q(i)$ 越接近于 1 说明无人机之间任务分配均衡度低，存在路线差距过大的情况。

用 $Fit(i)$ 表示粒子 i 当前适应度值，粒子 i 在通过公式(2.7)求得的总奖励值为 P ，根据总奖励值粒子计算得到当前适应度值，如公式(4.6)所示。

$$Fit(i) = P \times [1 - \psi Q(i)] \quad (4.6)$$

其中， ψ 表示均衡度权重系数。当总奖励值确定时，无人机任务分配不均衡时均衡度值越大适应度函数值越小，同理无人机任务分配均衡时均衡度值越小适应度函数值越大。对于 ASSPSO 算法来说，加入均衡度之后的适应度函数求解方式比直接采用总奖励值函数作为适应度值更高效，有利于加快收敛速度，提高解的质量。

对于本文提出的动态优先级不确定性变化下多无人机任务分配问题，涉及到

连续和离散决策变量，因此本问题属于混合变量优化问题，需要先对粒子的位置和速度进行离散化编码的同时需要求解粒子群算法中粒子连续性的位置和速度。

本章设计一种针对于本章多无人机任务分配问题的编码和解码过程，使 ASSPSO 算法能在连续的解空间内求得离散化的解，编码方式如下：

采用实数编码方式，令无人机总数量为 N 架，粒子群的维数 $j=M$ ， M 为总接收点的数量，粒子 $p(i)$ 的位置 $X_j[k]$ 和速度 $V_j[k]$ 都是 j 维实数向量，每个位置元素 $X_j[k]$ 限制在 1 到 $N+1$ 之间但不等于 $N+1$ ，即 $X_j[k] \in [1, N+1)$ ，每个速度元素 $V_j[k]$ 限制在 $[-N, N]$ ， N 为无人机个数。通过编码过程可以用位置元素 $X_j[k]$ 来表示一个任务分配方案。在解空间编码时考虑连续值和离散值的转换关系，通过采用数值和序值进行相互转换的形式完成解码，解码方式如下：

令 $\text{Int}(N)$ 为实数 N 的整数部分， $\text{Dec}(N)$ 为实数 N 的小数部分，那么此时 $\text{Int}(X_j[k])=s$ 表示将接收点 M_s 的物资派送任务交给了无人机 u_s 。当出现具有相同整数部分的位置元素时，此时将这些任务分给同一架无人机，用小数部分按大小排序后的由小到大顺序对应此无人机执行这些接收点的任务的先后顺序。用 $\text{Seq}[i]$ 来表示用位置元素解码出来的无人机 u_i 的任务序列。下面对实际编码和解码操作进行举例，若将 10 个任务 $M_1 \sim M_{10}$ 分配给 3 架无人机 $u_1 \sim u_3$ ，此时 ASSPSO 算法的解空间是一个 10 维空间，设定 $[-3, 3]$ 为粒子的速度元素范围， $[1, 4)$ 为位置元素范围，表 4.1 给出一个示例粒子的速度元素、位置元素和对应的 $\text{Int}(N)$ 和 $\text{Dec}(N)$ 。

表 4.1 示例粒子速度元素、位置元素和对应的整数部分和小数部分

Table 4.1 Example particle velocity and position element, corresponding integer and decimal part

接收点	$X_j[k]$	$\text{Int}(N)$	$\text{Dec}(N)$	$V_j[k]$
M_1	3.34	3	0.34	0.78
M_2	2.02	2	0.02	2.45
M_3	2.31	2	0.31	-2.21
M_4	1.53	1	0.53	-2.06
M_5	3.26	3	0.26	1.81
M_6	1.95	1	0.95	-0.98
M_7	3.21	3	0.21	0.45
M_8	1.45	1	0.45	1.75
M_9	2.43	2	0.43	-1.32
M_{10}	1.78	1	0.78	1.56

从表 4.1 中可以看出接收点 M_4, M_6, M_8, M_{10} 的任务分配给无人机 u_1 。根据 $\text{Dec}(X_j[8]) < \text{Dec}(X_j[4]) < \text{Dec}(X_j[10]) < \text{Dec}(X_j[6])$ 可解码出分配给无人机 u_1 的任务序列 $\text{seq}[1]=[M_8, M_4, M_{10}, M_6]$ 。同理，能够解码得到无人机 u_2 的任务序列 $\text{seq}[2]=[M_2, M_3, M_9]$ ，无人机 u_3 的任务序列 $\text{seq}[3]=[M_7, M_5, M_1]$ 。

为了提高粒子群算法的搜索性能，本章提出的 ASSPSO 算法将粒子搜索过程分为四种搜索模式，每个粒子的当前搜索模式由进化因子 E_f 决定， E_f 是描述粒子群分布特性的指数，进化因子 E_f 的计算过程如公式(4.5)所示。

$$E_f(i) = \frac{d_i - d_{\min}}{d_{\max} - d_{\min}} \quad (4.5)$$

其中， d_i 是粒子 i 与其他粒子欧式距离平均值， d_{\max} 和 d_{\min} 分别为 d_i 之间的最大和最小欧式距离。 d_i 的计算公式如公式(4.6)所示。

$$d_i = \frac{1}{N-1} \sum_{j=1, j \neq i}^N \sqrt{\sum_{k=1}^D (x_i^k - x_j^k)^2} \quad (4.6)$$

其中， N 和 D 分别代表种群的维度大小和迭代的总次数。

速度和更新公式(4.4)中的惯性权重 $\omega(k)$ 由进化因子 E_f 决定，如公式(4.7)所示。

$$\omega(k) = \omega_{\max} - E_f(\omega_{\max} - \omega_{\min}) \quad (4.7)$$

其中 ω_{\max} 表示设定的惯性权重最大值， ω_{\min} 为最小值。

进化因子 E_f 代表种群多样性。 E_f 越大说明种群的多样性越差。因此根据进化因子的大小，来对搜索模式进行选择，在多样性最差的时候进行跳出，在多样性最好的时候进行收敛，四种不同搜索模式的切换规则如下：

$$\begin{cases} \text{收敛: } \varphi(k) = 1, & 0 \leq E_f < 0.25 \\ \text{开发: } \varphi(k) = 2, & 0.25 \leq E_f < 0.5 \\ \text{探索: } \varphi(k) = 3, & 0.5 \leq E_f < 0.75 \\ \text{跳出: } \varphi(k) = 4, & 0.75 \leq E_f \leq 1 \end{cases}$$

本章提出的 ASSPSO 算法使用一种利用邻域信息的粒子学习策略，根据不同的搜索模式确定不同的学习对象。ASSPSO 算法通过充分利用粒子群邻域的信息让更多有信息的粒子用于搜索过程，有助于缓解过早收敛问题。参考 Zeng 等人^[83]的邻域生成方法，通过对粒子 $p(i)$ 和其他粒子之间的欧氏距离的计算，将与粒子 $p(i)$ 的解欧式距离最小的 q 个粒子构成邻域，在算法整个迭代过程中，会对粒子邻

域进行更新。

在种群中随机选择除粒子 $p(i)$ 之外的粒子 p_a ，从 p_a 对应个体最优 $pbest$ 的邻域中进行粒子 $p(i)$ 的 $pbest$ 部分的速度学习。粒子 $p(i)$ 的全局最优 $gbest$ 部分的速度学习方式为在 $gbest$ 的邻域中随机抽选除去 $gbest$ 之外的一个粒子进行学习。通过这种在 $gbest$ 附近学习方式，ASSPSO 算法在当前搜索模式下解决陷入局部最优的麻烦。当邻域粒子数目 $q=3$ 时，基于欧式距离开发出的邻域的粒子学习策略的示意图如图 4.3 所示。

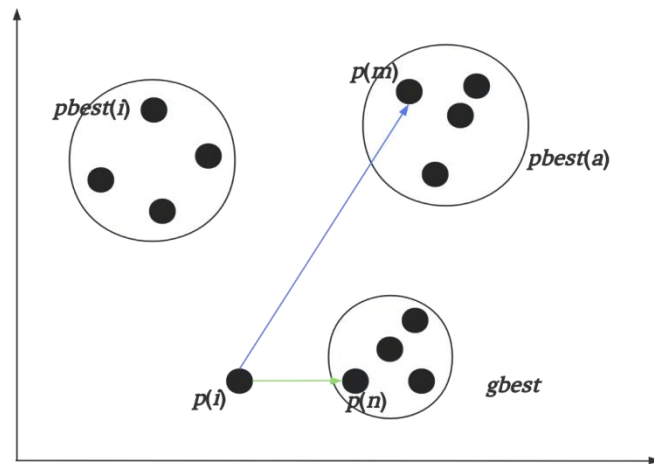


图 4.3 基于开发邻域的粒子学习策略示意图

Figure 4.3 Schematic figure of the particle learning strategy based on the develop neighborhood

对于粒子种群中的所有粒子，基于欧式距离开发出的邻域的粒子学习策略的完整过程如算法 4.1 所示。

算法 4.1 基于开发邻域的粒子学习策略

Algorithm 4.1 Particle learning strategy based on developed neighborhood

算法 4.1：基于开发邻域的粒子学习策略

For $i = 1, 2, \dots, N$

根据粒子之间欧氏距离的计算，开发出 $pbest(i)$ 粒子的邻域(即距离 $pbest(i)$ 最小的 q 个粒子)

根据粒子之间欧氏距离的计算，开发出 $gbest$ 的邻域(即距离 $gbest$ 最小的 q 个粒子)

For $j = 1, 2, \dots, M$

从种群中随机选择一个粒子(第 i 个粒子除外)，称为 p_a 。

从 $pbest(a)$ 的邻域中随机抽选一个粒子 $p(m)$ 。

续算法 4.1 基于开发邻域的粒子学习策略

Algorithm 4.1 Particle learning strategy based on developed neighborhood(continued)

 从 $gbest$ 的邻域中随机抽选一个粒子 $p(n)$ 。

 从粒子 $p(m)$ 和粒子 $p(n)$ 中进行学习。

End

End

根据粒子速度和位置的更新公式，对四种搜索模式加速系数 c_1 和 c_2 值进行设定。开发和探索模式的核心目标为更全面探索解空间，以防止粒子陷入局部最优解，因此开发和探索模式的 c_2 值均被设定较小以减轻 $gbest$ 的影响。在跳出模式下为了让粒子离开当前最佳位置，设定较大的 c_2 值和较小的 c_1 值。根据上述描述设计加速系数和 $pbest^i$ 、 $gbest^i$ 根据搜索模式的变换策略，如表 4.2 所示。

表 4.2 加速系数和 $pbest^i$ 、 $gbest^i$ 根据搜索模式的变换策略Table 4.2 Transformation strategy according to the search pattern of acceleration coefficient and $pbest^i$ 、 $gbest^i$

模式	模式	c_1	c_2	$pbest^i$	$gbest^i$
收敛	$\varphi(k) = 1$	1.8	1.8	$p_{i,j}(k)$	$p_{g,j}(k)$
开发	$\varphi(k) = 2$	1.9	1.7	$p_{m,j}(k)$	$p_{g,j}(k)$
探索	$\varphi(k) = 3$	2.0	1.6	$p_{i,j}(k)$	$p_{n,j}(k)$
跳出	$\varphi(k) = 4$	1.6	2.1	$p_{m,j}(k)$	$p_{n,j}(k)$

对四种搜索模式下 $pbest^i$ 、 $gbest^i$ 变换的更全面描述如下：

在收敛模式下，粒子进行常规搜索，每个粒子都根据粒子本身当前的当代最优 $p_{i,j}(k)$ 和全局最优 $p_{g,j}(k)$ 进行速度变化。

在开发模式下，粒子要增加个体范围的空间搜索， $pbest^i$ 使用从 $pbest(a)$ 的邻域中随机抽选的粒子 $p(m)$ 的 $p_{m,j}(k)$ ， $gbest^i$ 为粒子本身全局最优 $p_{g,j}(k)$ 。

在探索模式下，粒子要倾向于在最优解区域相对更远处搜索，摆脱陷入局部最优的局面。 $pbest^i$ 使用粒子本身当前的当代最优 $p_{i,j}(k)$ ， $gbest^i$ 使用在 $gbest$ 邻域中抽选出的粒子 $p(n)$ 的 $p_{n,j}(k)$ 。

在跳出模式下，粒子要倾向于离开当前的局部最优， $pbest^i$ 使用从 $pbest(a)$ 的邻域中随机抽选的粒子 $p(m)$ 的 $p_{m,j}(k)$ ， $gbest^i$ 为粒子本身全局最优 $p_{g,j}(k)$ 。

图 4.4 展示了 ASSPSO 算法在解决动态优先级不确定性变化下多无人机任务分配问题的完整流程。

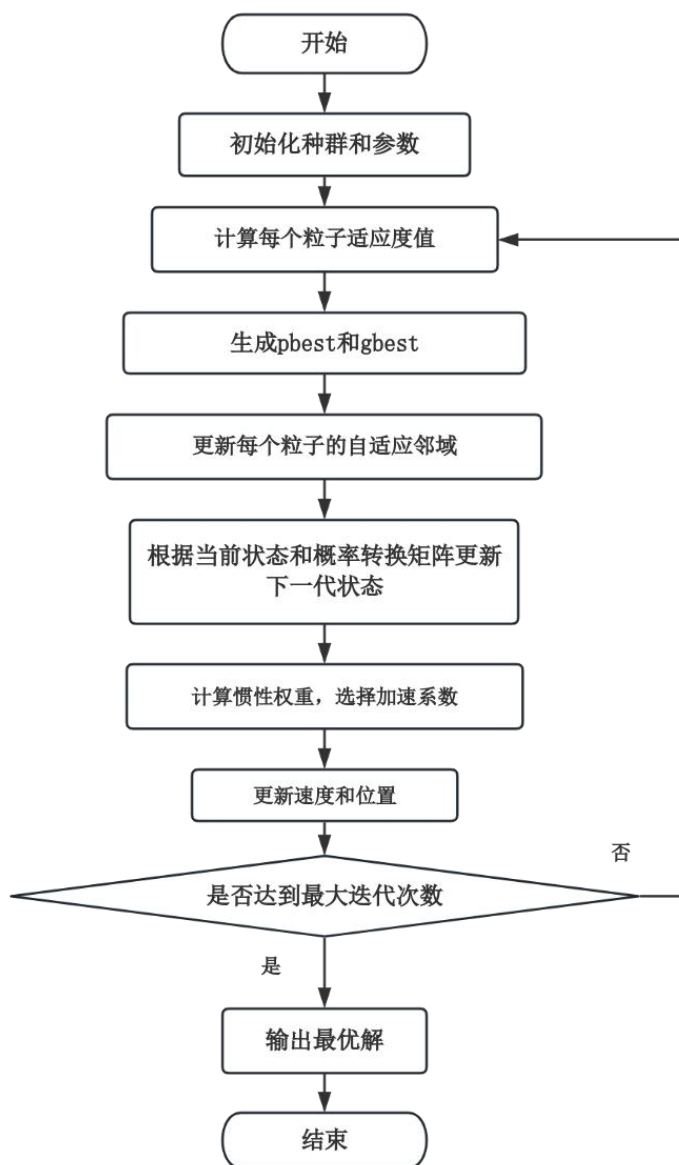


图 4.4 ASSPSO 算法流程图

Figure 4.4 ASSPSO algorithm flow chart

4.5 仿真实验与结果分析

为了验证 ASSPSO 算法对动态优先级不确定性变化下多无人机任务分配问题的有效性, 本章使用 CBBA 算法、近年来被多次用于无人机任务分配相关问题的离散粒子群算法(Discrete Particle Swarm Optimization, DPSO)^[84]、本文第三章解决确定性变化问题的贪婪算法和 IPGA-G 算法作为 ASSPSO 算法的对比算法进行实验。在仿真实验的硬件环境中, 计算机处理器为 Intel(R) Corei5, 主频为 2.50GHz, 仿真实验使用软件 Matlab R2019b 编译。

仿真实验子区域设置方法同 3.4 节, 设 4 架无人机 $U = \{u_1, u_2, u_3, u_4\}$ 从派送中心出发执行物资配送任务, 每架无人机平均飞行速度 $V_{AVG} = 11\text{km/h}$ 。设置装货时间 $T_{load}=0.5\text{h}$, 卸货时间 $T_{unload}=0.1\text{h}$, 每个接收点单次接收固定重量物资 $L_0=2.5\text{kg}$, 设物资消耗水平系数 $c=2$, 奖励函数参数 $P_1=200, P_2=180$, 单位距离运输成本 $d=3.5$, 每架无人机的固定成本 $F_n=80$, 无人机最大单次飞行最大飞行路程限制 $R_{\max}=160\text{km}$, 本章所有实验设置无人机进行 1 轮任务分配。

用于本章实验的贪婪算法和 3.2 节用于确定性变化问题的贪婪算法有所不同, 贪婪算法在处理本章不确定性问题时由于不能获得全局的确定信息, 修改成每当出现子区域需求状态变化时进行重分配的方式进行计算。

4.5.1 关键参数分析实验

进行算法对比实验之前, 先使用 ASSPSO 算法测试不确定性变化问题的界限值 W_0 和周期 T 对于实验结果的影响。设置 50 组实验, 每组实验在 50×50 平方千米的仿真区域内随机产生 10 个接收点, 设置固定周期 $T=8$, 界限值分别设为 $W_0=1$ 、 $W_0=1.5$ 和 $W_0=2$, 比较不同界限值对实验结果的影响, 实验结果如图 4.5 所示。

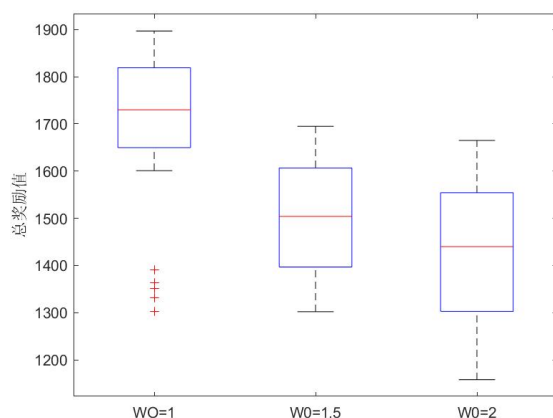


图 4.5 不同界限值 W_0 实验结果对比箱式图

Figure 4.5 Box results of results of different boundary values W_0

根据实验结果, $W_0=1.5$ 实验结果相对集中且稳定, 所以选择 $W_0=1.5$ 作为本章后续实验的界限值参数。设置界限值 $W_0=1.5$, 设置 50 组实验, 周期分别设为 $T=7$, $T=8$, $T=9$, 比较不同周期对实验结果的影响, 实验结果如图 4.6 所示。

根据实验结果, $T=7$ 时总奖励值上下限差距较大, $T=9$ 时总奖励值普遍偏高且上下限差距较小, $T=8$ 时上下限差距适中更适合作为实验模型的固定条件, 所以根据实验结果设定 $T=8$, $W_0=1.5$ 作为本章后续实验的固定参数。在 $T=8$, $W_0=1.5$ 时不同子区域的需求状态变化时间如表 4.3 所示。

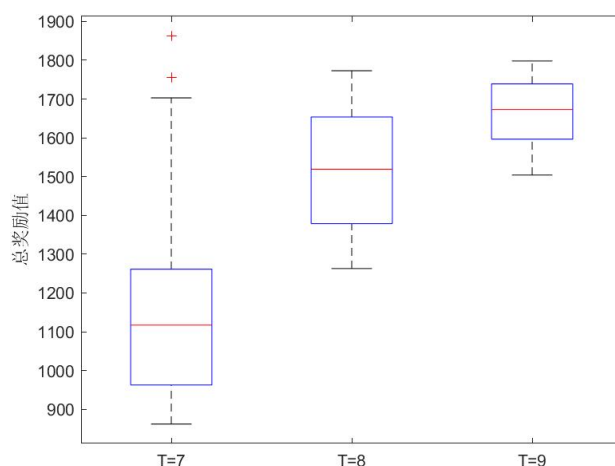


图 4.6 不同周期 T 实验结果对比箱式图

Figure 4.6 Box plots of results of different period T

表 4.3 不同子区域需求状态变化时间表($T=8, W_0=1.5$)

Table 4.3 Schedule of demand status changes in different sub-regions ($T=8, W_0=1.5$)

子区域	第一次变化	第二次变化
G1	2.26	9.77
G2	1.88	10.12
G3	1.23	10.76
G4	1.07	11.33

邻域大小 q 对 ASSPSO 算法性能会产生一定的影响, 设定邻域大小时要根据求解问题的特质和群体大小来共同决定, 下面进行测试用于本章不确定性问题的 ASSPSO 算法的最合适的邻域大小 q 。设 ASSPSO 算法的粒子群种群大小 60, 迭代次数为 80 次, 对 $q=3$, $q=4$, $q=5$ 三种情况分别进行测试。设置 100 组实验, 每组实验随机产生 10 个接收点, 实验结果如图 4.7 所示。

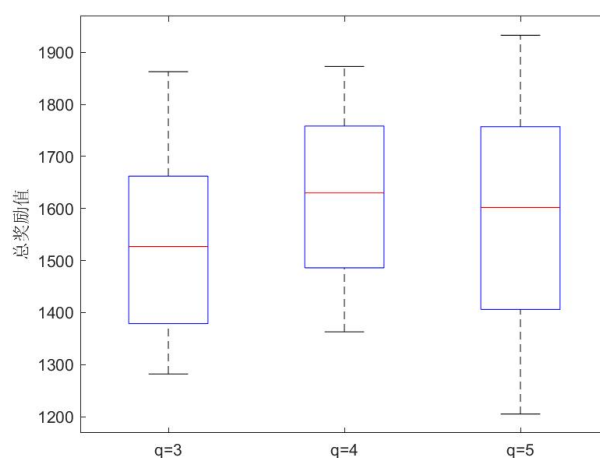


图 4.7 不同邻域大小 q 实验结果对比箱式图

Figure 4.7 Box plots of experimental results with different neighborhood sizes q

根据图 4.7 所示, $q=5$ 时 ASSPSO 算法能得到更高的总奖励值上限高于 $q=4$ 的上限, 但是下限和平均值比 $q=4$ 低; $q=3$ 时的平均值不如 $q=4$ 。根据实验结果可以得出结论, $q=4$ 更适合用于种群大小为 60 的 ASSPSO 算法。在使用 ASSPSO 算法求解本章提出的动态优先级不确定性变化下多无人机协同任务分配问题时, 邻域大小设置为粒子群种群大小的 1/15 左右最合适。

4.5.2 算法对比实验

为了验证 ASSPSO 算法对动态优先级不确定性变化下多无人机任务分配问题的解决能力, 进行对比实验, 用贪婪算法、IPGA-G 算法、CBBA 算法、DPSO 算法和 ASSPSO 算法进行对比实验, 不同算法的参数设定如表 4.4 所示。

表 4.4 不同算法参数设定表

Table 4.4 Parameters setting of different algorithms

算法名称	种群大小	迭代次数	其他参数
IPGA-G	100	100	$k_1=0.7, k_2=0.4, k_3=0.1, k_4=0.05$
DPSO	100	80	$\omega=0.8, c_1=0.6, c_2=0.6$
ASSPSO	60	80	$d_{\max}=9, d_{\min}=0, \psi=0.2, \omega_{\max}=0.9, \omega_{\min}=0.3$

设置 100 组实验, 每组实验随机产生 10 个接收点, 记录下各算法解得最优结果, 将 100 组实验的结果绘制箱式图, 如图 4.8 所示。

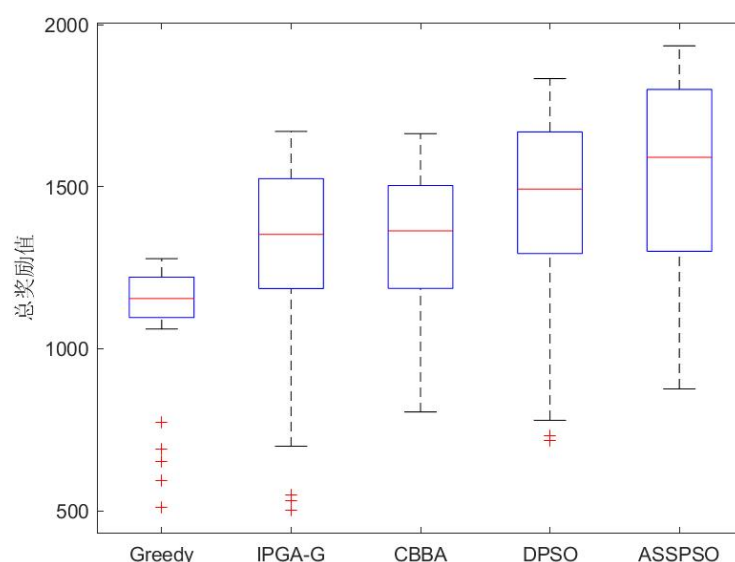


图 4.8 不同算法实验结果对比箱式图

Figure 4.8 Box-plots of experimental results of the different algorithms

根据 100 组实验的实验结果可以看出贪婪算法和 IPGA-G 算法在求解动态优先级不确定性变化问题时会出现总奖励值过低的情况, 不如本章提出的 ASSPSO 算

法，说明了用于解决动态优先级确定性变化下的问题的算法不适用于动态优先级不确定变化下的问题。本章提出的 ASSPSO 算法解的上限较高，对最优解求解具有一定优势。

将 DPSO 算法解得 100 组实验结果的中位数所在实验组的实验结果作为示例。在这一组实验中，不同算法在相同的一组接收点坐标下进行无人机任务分配，可以展示出不同算法的能力。CBBA 算法、DPSO 算法和 ASSPSO 算法的实验结果如表 4.5 所示。

表 4.5 ASSPSO 算法和 DPSO 算法的分配结果

Table 4.5 Allocation results of ASSPSO and DPSO

算法名称	总奖励值	Seq[1]	Seq[2]	Seq[3]	Seq[4]
CBBA	1454.84	$[M_2, M_{10}, M_9]$	$[M_4, M_5, M_7]$	$[M_8, M_6, M_3]$	$[M_1]$
DPSO	1496.58	$[M_8, M_4, M_6, M_1]$	$[M_2, M_{10}]$	$[M_7, M_5, M_9]$	$[M_3]$
ASSPSO	1624.56	$[M_8, M_6, M_7]$	$[M_2, M_1, M_3]$	$[M_4, M_5]$	$[M_{10}, M_9]$

根据表 4.5 可知，ASSPSO 算法的无人机任务分配更加均衡，并且获得了更高的奖励值，实验证明了在 ASSPSO 算法的适应度计算过程加入均衡度的计算能够对搜索最优解的过程产生帮助。

为了比较两个粒子群衍生算法的性能，在实验开始前，对 ASSPSO 算法和 DPSO 算法设置相同的迭代次数 80 次，将 ASSPSO 算法和 DPSO 算法两种算法在这组实验过程中的收敛曲线进行对比，生成收敛曲线对比图，如图 4.9 所示。

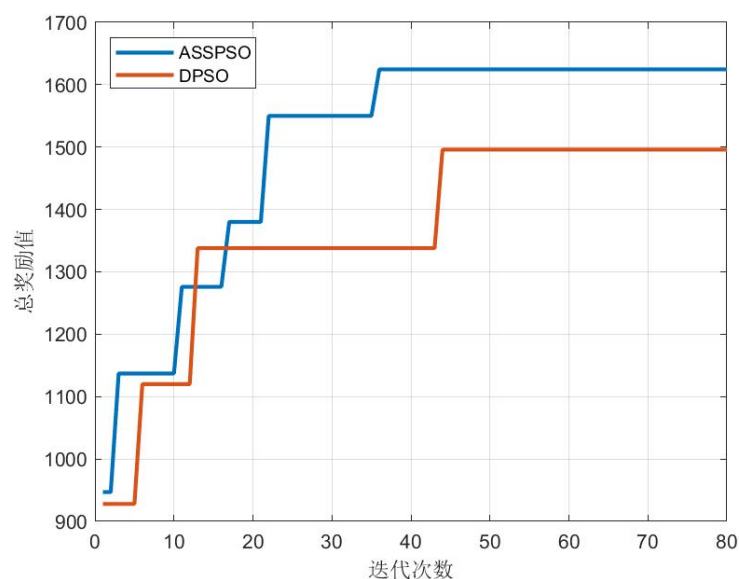


图 4.9 收敛曲线对比图

Figure 4.9 The comparison diagram of the convergence curve

根据收敛曲线对比图可知, DPSO 算法虽然在前期收敛速度较快,但在中期陷入局部收敛,没能继续搜索到更高的结果; ASSPSO 算法具有更强的持续搜索能力的同时具有很快的收敛速度。

在这组接收点的仿真实验场景下,进行 50 轮的重复实验用于测试算法的稳定性,对 DPSO 算法和 ASSPSO 算法的 50 轮实验结果进行记录,将实验结果进行记录,如表 4.6 所示。

表 4.6 DPSO 算法和 ASSPSO 算法 50 轮实验结果

Table 4.6 Results of 50 rounds of experiments with DPSO and ASSPSO

轮次	DPSO 算法	ASSPSO 算法	轮次	DPSO 算法	ASSPSO 算法
1	1454.84	1624.56	26	1551.43	1594.98
2	1594.98	1624.56	27	1440.72	1624.56
3	1551.43	1624.56	28	1496.58	1496.58
4	1496.58	1462.77	29	1496.58	1462.77
5	1454.84	1624.56	30	1409.34	1624.56
6	1431.06	1594.98	31	1551.43	1624.56
7	1411.97	1624.56	32	1434.28	1594.98
8	1580.98	1624.56	33	1413.04	1624.56
9	1496.58	1551.43	34	1436.61	1624.56
10	1551.43	1624.56	35	1594.98	1496.58
11	1404.17	1624.56	36	1551.43	1624.56
12	1594.98	1551.43	37	1462.77	1551.43
13	1551.43	1496.58	38	1419.20	1624.56
14	1442.19	1624.56	39	1551.43	1551.43
15	1436.61	1551.43	40	1488.38	1624.56
16	1594.98	1624.56	41	1424.55	1624.56
17	1462.77	1551.43	42	1496.58	1462.77
18	1496.58	1551.43	43	1406.02	1594.98
19	1486.31	1624.56	44	1580.38	1624.56
20	1454.84	1624.56	45	1454.84	1551.43
21	1551.43	1551.43	46	1496.58	1624.56
22	1428.21	1624.56	47	1462.77	1496.58
23	1454.84	1624.56	48	1504.39	1462.77
24	1462.77	1624.56	49	1496.58	1624.56
25	1496.58	1496.58	50	1437.20	1551.43

将 DPSO 算法和 ASSPSO 算法的 50 轮实验结果生成箱型图，如图 4.10 所示。

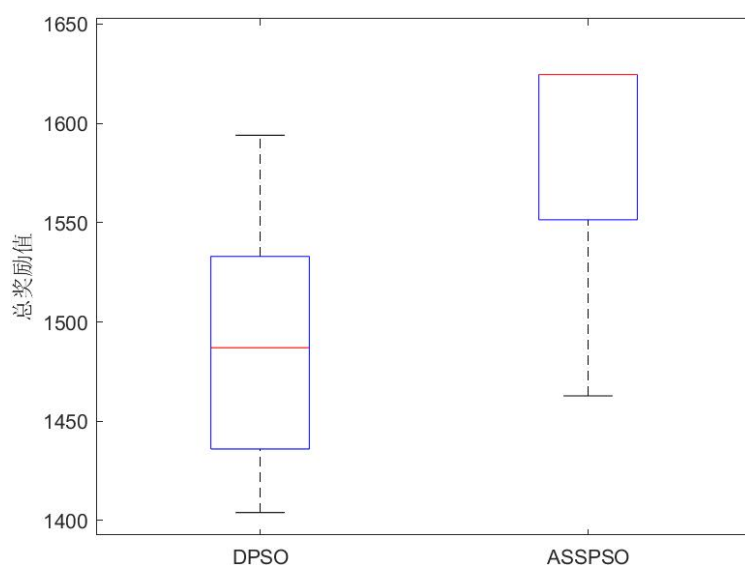


图 4.10 不同算法重复实验结果箱型图

Figure 4.10 Box-plots of repeated experimental results for different algorithms

根据图 4.10 可以看出，DPSO 算法的实验结果比较分散，ASSPSO 算法的实验结果主要集中在最高值。根据 50 轮重复实验的实验结果，ASSPSO 算法在其中 27 次达到最高总奖励值 1624.56，有超过一半的实验轮次达到了当前场景的最优解，说明 ASSPSO 算法具有很强的稳定性。

为了比较算法搜索速度，记录 CBBA 算法、DPSO 算法和 ASSPSO 算法在这 50 轮实验过程中每轮实验搜索得到最优解超过总奖励值 1400 的时间，计算 50 轮时间的平均值，如图 4.11 所示。

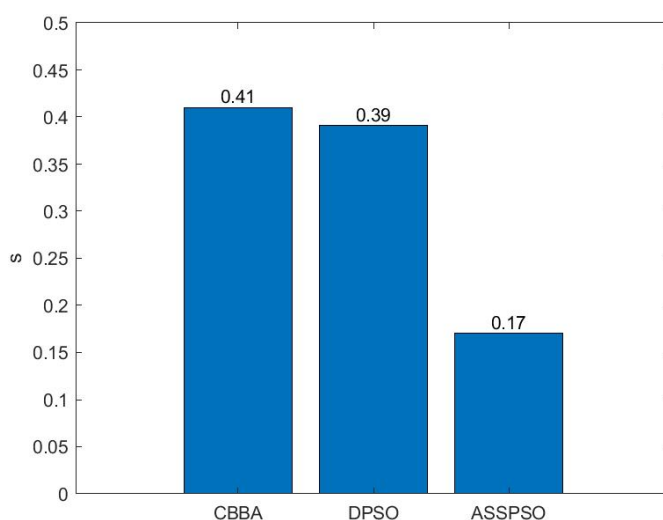


图 4.11 三种算法运算平均时间

Figure 4.11 The operation average time of the three algorithms

根据三种算法平均运算时间可以看出, ASSPSO 算法搜索得到超过总奖励值 1400 的最优解的平均时间为 0.17s, 是 DPSO 算法平均时间的 44%, 是 CBBA 算法的 41%。实验结果说明本章提出的 ASSPSO 算法在搜索速度上具有明显优势, 能够在更短的时间得到优秀的解。

综上所述, 根据上述实验结果, 可以得出相应的结论, ASSPSO 算法具有更强的搜索能力, 能在更短的时间内完成对最优解的搜索, 能够搜索得到更加稳定的最优解实验结果, 可以更高效地解决动态优先级不确定性变化情况下的多无人机协同任务分配问题, 求解出更合理且收益更高的多无人机协同任务分配方案。

4.6 本章小结

本章采用动态需求变化问题相关研究经常使用的泊松分布来生成不确定性变化情况下需求状态变化函数, 将总奖励值最大化继续作为任务的主要目标。本章设计了基于一致性的捆绑拍卖算法在解决本章问题的具体流程。本章提出自适应搜索模式切换粒子群算法用于解决不确定性变化问题, 算法设计了一种新的粒子速度和位置更新机制, 根据四种搜索模式切换学习策略和加速系数, 在每次迭代中根据搜索模式自适应地选择加速系数和更新速度模型, 从而有助于对问题空间的彻底搜索, 解决了离散粒子群算法容易陷入局部最优解的问题。为了更好地适应本章提出的问题模型, ASSPSO 算法设计了适用于本章问题模型的编码和解码方式, 将无人机均衡度加入适应度值的计算, 提高了算法的寻优能力。本章最后进行了关键参数分析实验, 用 ASSPSO 算法和基准算法进行对比实验。实验验证 ASSPSO 算法能够提高种群的多样性和全局搜索能力, 可以更高效地解决动态优先级不确定性变化情况下的多无人机协同任务分配问题, 得到更优秀的多无人机协同任务分配方案。

5 总结与展望

5.1 本文工作总结

本文建立了一种在特殊环境下任务优先级发生动态变化的多无人机协同运行规划模型，模型主要是针对于在现实无人机执行任务的过程中，由于环境的变化，人为因素等导致任务目标优先级的变化的情况。在此模型下，本文基于群体智能算法对多无人机协同运行规划算法进行了深入研究。本文的主要工作如下：

(1) 本文对任务目标优先级发生动态变化问题进行了分析，定义任务优先程度(A,B,C)来作为体现任务需求紧急度的指标。通过区域差异化的思想，将区域划分为四个子区域，然后构建子区域的需求状态函数来对子区域的急需程度进行表示，通过需求状态函数可以得到某一时刻子区域的任务优先程度。根据任务优先程度设定奖励函数，构造用来表示多无人机协同任务分配完成情况总奖励值函数。在约束条件下，以最大化奖励值作为多无人机协同执行任务完成程度的评价标准。

(2) 本文采用安全库存模型作为参考构建了多无人机协同任务分配在动态优先级确定性变化下问题模型，采用周期性函数来表示子区域中接收点的需求状态。首先，本文用贪婪算法求解此问题，设计了用贪婪算法解决此问题的具体应用流程；然后分析群体智能算法中最具有代表性的遗传算法，用对应问题的编码方式进行编码，构建改进单亲遗传算法和改进单亲遗传-贪婪结合算法来对问题进行仿真实验。实验结果表明，本文提出的改进单亲遗传-贪婪结合算法具有优秀的搜索能力和搜索效率，在处理多无人机协同任务分配在动态优先级确定性变化问题时，具有很高的适应性。

(3) 本文采用动态需求变化问题相关研究经常使用的泊松分布来生成不确定性变化情况下需求状态变化函数，将总奖励值最大化继续作为多无人机协同任务分配的主要目标。本文用 CBBA 算法求解此问题，设计了用 CBBA 算法解决此问题的具体应用流程。本文提出自适应搜索模式切换粒子群算法用于解决不确定性变化问题，引入了搜索模式来增强种群中的交流，解决了粒子群算法容易陷入局部最优解的问题。本文用 ASSPSO 算法和基准算法进行了对比实验，实验结果表明，ASSPSO 算法能够提高种群的多样性和全局搜索能力，可以更高效地解决动态优先级不确定性变化情况下的多无人机协同任务分配问题。

5.2 未来工作展望

本文在研究过程中，为了简化问题规模，降低计算难度，忽略了很多因素。因此，目前的研究成果距离应用到实际环境中还有很大的差距。基于现有成果，今后的工作可以从以下几个方面进行更深入的研究：

(1) 在实现多无人机协同任务分配的过程中，并未考虑完全集中式控制的方式中无人机的通信限制与计算能力限制，在未来多无人机协同运行规划过程中，这两点是必不可忽略的因素。

(2) 本文构建的模型是在二维平面上进行，即忽略了无人机的飞行高度，现实环境中需要考虑飞行高度对问题的影响。在模型区域内还忽略了一些障碍物、通信障碍区域等因素，它们会影响无人机的路径，从而影响了整体任务分配的过程。在后续的研究中，可以使用一些方法将飞行高度对问题的影响加入到目标函数的构建中。

(3) 在构建动态环境下多无人机集群协同任务分配模型的过程中，可能会出现动态优先级影响约束条件的情况。在实际环境下，约束条件比本文描述的要多得多，在动态优先级变化的过程中，约束条件也可能会随着发生变化。

(4) 在解决动态环境下多无人机协同任务分配的过程中，由于仿真环境的局限性、数学建模的局限性，本文只进行了较小规模的区域规模构建。在实际环境中，无人机集群的规模将会是较为庞大的，所以更大规模的动态环境模型的值得进行更深入的研究。

参考文献

- [1] 庞强伟, 胡永江, 李文广等. 多无人机协同侦察任务规划方法研究综述[J]. 电讯技术, 2019, 59(6): 741-748.
- [2] Zhou S, Yin G, Wu Q. UAV cooperative multiple task assignment based on discrete particle swarm optimization[C]//2015 7th International Conference on Intelligent Human-Machine Systems and Cybernetics. IEEE, 2015, 2: 81-86.
- [3] Zheng Y J, Du Y C, Ling H F, et al. Evolutionary collaborative human-UAV search for escaped criminals[J]. IEEE Transactions on Evolutionary Computation, 2020, 24(2): 217-231.
- [4] Huang J, Luo Y, Quan Q, et al. An autonomous task assignment and decision-making method for coverage path planning of multiple pesticide spraying UAVs[J]. Computers and Electronics in Agriculture, 2023, 212: 108128.
- [5] 吴琦. 基于计算机仿真技术的农业无人机通信系统设计[J]. 农机化研究, 2024, 46(8): 129-133.
- [6] Sharma A, Shoval S, Sharma A, et al. Path planning for multiple targets interception by the swarm of UAVs based on swarm intelligence algorithms: A review[J]. IETE Technical Review, 2022, 39(3): 675-697.
- [7] Sharma A, Sharma A, Pandey J K, et al. Swarm intelligence: foundation, principles, and engineering applications[M]. CRC Press, 2022.
- [8] Duan H, Zhang X. Phase transition of vortexlike self-propelled particles induced by a hostile particle[J]. Physical Review E, 2015, 92(1): 012701.
- [9] Li S, Chen H, Wang M, et al. Slime mould algorithm: A new method for stochastic optimization[J]. Future Generation Computer Systems, 2020, 111: 300-323.
- [10] 段海滨, 张祥银, 徐春芳. 仿生智能计算[M]. 北京: 科学出版社, 2011.
- [11] Seid A M, Boateng G O, Mareri B, et al. Multi-agent DRL for task offloading and resource allocation in multi-UAV enabled IoT edge network[J]. IEEE Transactions on Network and Service Management, 2021, 18(4): 4531-4547.
- [12] Liao H, Zhou Z, Zhao X, et al. Learning-based queue-aware task offloading and resource allocation for space-air-ground-integrated power IoT[J]. IEEE Internet of Things Journal, 2021, 8(7): 5250-5263.
- [13] Gao A, Geng T, Hu Y, et al. Decentralized continuous game for task offloading in UAV cloud[C]//2020 29th Wireless and Optical Communications Conference (WOCC). IEEE, 2020: 1-6.

- [14] Gao S, Wu J, Ai J. Multi-UAV reconnaissance task allocation for heterogeneous targets using grouping ant colony optimization algorithm[J]. *Soft Computing*, 2021, 25(10): 7155-7167.
- [15] Kolar P. Coupling consensus based tasks with subsumption architecture for UAS swarm based intelligence surveillance and reconnaissance operations[C]//2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC). IEEE, 2020: 1-11.
- [16] Ye F, Chen J, Sun Q, et al. Decentralized task allocation for heterogeneous multi-UAV system with task coupling constraints[J]. *The Journal of supercomputing*, 2021, 77: 111-132.
- [17] Zhang Z, Wu J, Dai J, et al. Cooperative Tactical Planning Method for UAV Formation[C]//2020 39th Chinese Control Conference (CCC). IEEE, 2020: 1542-1547.
- [18] Wei T, Yongjiang H, Yuefei Z, et al. Multi-UAV task allocation based on type mamdani fuzzy logic[C]//2021 7th International Symposium on Mechatronics and Industrial Informatics (ISMII). IEEE, 2021: 184-187.
- [19] Chen X, Zhang P, Li F, et al. A cluster first strategy for distributed multi-robot task allocation problem with time constraints[C]//2018 WRC symposium on advanced robotics and automation (WRC SARA). IEEE, 2018: 102-107.
- [20] Rodríguez M, Al-Kaff A, Madridano Á, et al. Wilderness search and rescue with heterogeneous multi-robot systems[C]//2020 International Conference on Unmanned Aircraft Systems (ICUAS). IEEE, 2020: 110-116.
- [21] Ompusunggu V M M O, Hardhienata M K D, Priandana K. Application of ant colony optimization for the selection of multi-UAV coalition in agriculture[C]//2020 International Conference on Computer Science and Its Application in Agriculture (ICOSICA). IEEE, 2020: 1-8.
- [22] Zheng H, Hongxing W, Tianpei Z, et al. The Collaborative Power Inspection Task Allocation Method of "Unmanned Aerial Vehicle and Operating Vehicle"[J]. *IEEE Access*, 2021, 9: 62926-62934.
- [23] Fang Z, Hong-Hai Z. A method for "last mile" distribution demand for drones[C]//2020 IEEE 5th International Conference on Intelligent Transportation Engineering (ICITE). IEEE, 2020: 561-564.
- [24] Caraballo L E, Díaz-Báñez J M, Maza I, et al. The block-information-sharing strategy for task allocation: A case study for structure assembly with aerial robots[J]. *European Journal of Operational Research*, 2017, 260(2): 725-738.
- [25] Akhloufi M A, Couturier A, Castro N A. Unmanned aerial vehicles for wildland fires: Sensing, perception, cooperation and assistance[J]. *Drones*, 2021, 5(1): 15.
- [26] Tkach I, Jevtić A, Nof S Y, et al. A modified distributed bees algorithm for multi-sensor task

- allocation[J]. *Sensors*, 2018, 18(3): 759.
- [27] Bonabeau E, Theraulaz G, Deneubourg J L. Fixed response thresholds and the regulation of division of labor in insect societies[J]. *Bulletin of Mathematical Biology*, 1998, 60(4): 753-807.
- [28] Tan Y, Zheng Z. Research advance in swarm robotics[J]. *Defence Technology*, 2013, 9(1): 18-39.
- [29] Theraulaz G, Bonabeau E, Deneubourg J N. Response threshold reinforcements and division of labour in insect societies[J]. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 1998, 265(1393): 327-332.
- [30] Bayındır L. A review of swarm robotics tasks[J]. *Neurocomputing*, 2016, 172: 292-321.
- [31] Wang Z, Wang B, Wei Y, et al. Cooperative multi-task assignment of multiple UAVs with improved genetic algorithm based on beetle antennae search[C]//2020 39th Chinese Control Conference (CCC). IEEE, 2020: 1605-1610.
- [32] Wu X, Yin Y, Xu L, et al. Multi-UAV task allocation based on improved genetic algorithm[J]. *IEEE Access*, 2021, 9: 100369-100379.
- [33] Duan H, Zhao J, Deng Y, et al. Dynamic discrete pigeon-inspired optimization for multi-UAV cooperative search-attack mission planning[J]. *IEEE Transactions on Aerospace and Electronic Systems*, 2020, 57(1): 706-720.
- [34] Chen H, Xu J, Wu C. Multi-UAV task assignment based on improved Wolf Pack Algorithm[C]//Proceedings of the 2020 International Conference on Cyberspace Innovation of Advanced Technologies. 2020: 109-115.
- [35] Han Q T. Research on cooperate search path planning of multiple UAVs using Dubins curve[C]//2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA). IEEE, 2021: 584-588.
- [36] Zhang W, Shi C, Zhou J. LPI-Based Searching Task Allocation for Multi-UAVs System[C]//2020 3rd International Conference on Unmanned Systems (ICUS). IEEE, 2020: 873-877.
- [37] Huang T, Wang Y, Cao X, et al. Multi-UAV mission planning method[C]//2020 3rd International Conference on Unmanned Systems (ICUS). IEEE, 2020: 325-330.
- [38] Xia C, Yongtai L, Liyuan Y, et al. Cooperative task assignment and track planning for multi-UAV attack mobile targets[J]. *Journal of Intelligent & Robotic Systems*, 2020, 100: 1383-1400.
- [39] Zhang X, Wang K, Dai W. Multi-UAVs task assignment based on fully adaptive cross-entropy algorithm[C]//2021 11th International Conference on Information Science and Technology (ICIST). IEEE, 2021: 286-291.

- [40] Zheng H, Yuan J. An Integrated Mission Planning Framework for Sensor Allocation and Path Planning of Heterogeneous Multi-UAV Systems[J]. *Sensors*, 2021, 21(10): 3557.
- [41] Miao Y, Zhong L, Yin Y, et al. Research on dynamic task allocation for multiple unmanned aerial vehicles[J]. *Transactions of the Institute of Measurement and Control*, 2017, 39(4): 466-474.
- [42] Yang W Z, Yan X. Multi-UAV Task Assignment Based on Quantum Genetic Algorithm[C]//*Journal of Physics: Conference Series*. IOP Publishing, 2021, 1824(1).
- [43] Jin Y, Feng J, Zhang W. UAV Task Allocation for Hierarchical Multiobjective Optimization in Complex Conditions Using Modified NSGA-III with Segmented Encoding[J]. *Journal of Shanghai Jiaotong University (Science)*, 2021(004):026.
- [44] Choi H L, Brunet L, How J P. Consensus-based decentralized auctions for robust task allocation[J]. *IEEE transactions on robotics*, 2009, 25(4): 912-926.
- [45] Whitbrook A, Meng Q, Chung P W H. A robust, distributed task allocation algorithm for time-critical, multi agent systems operating in uncertain environments[C]//*Advances in Artificial Intelligence: From Theory to Practice: 30th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2017, Arras, France, June 27-30, 2017, Proceedings, Part II 30*. Springer International Publishing, 2017: 55-64.
- [46] Zhao W, Meng Q, Chung P W H. A heuristic distributed task allocation method for multivehicle multitask problems and its application to search and rescue scenario[J]. *IEEE transactions on cybernetics*, 2015, 46(4): 902-915.
- [47] Turner J, Meng Q, Schaefer G, et al. Distributed task rescheduling with time constraints for the optimization of total task allocations in a multirobot system[J]. *IEEE transactions on cybernetics*, 2017, 48(9): 2583-2597.
- [48] Cui W, Li R, Feng Y, et al. Distributed task allocation for a multi-UAV system with time window constraints[J]. *Drones*, 2022, 6(9): 226.
- [49] Qamar R A, Sarfraz M, Ghauri S A, et al. TRMaxAlloc: Maximum task allocation using reassignment algorithm in multi-UAV system[J]. *Computer Communications*, 2023, 206: 110-123.
- [50] Tian Y T, Yang M, Qi X Y, et al. Multi-robot task allocation for fire-disaster response based on reinforcement learning[C]//*2009 International Conference on Machine Learning and Cybernetics*. IEEE, 2009, 4: 2312-2317.
- [51] Noureddine D B, Gharbi A, Ahmed S B. Multi-agent Deep Reinforcement Learning for Task Allocation in Dynamic Environment[C]//*ICSOFT*. 2017: 17-26.
- [52] Majkowska A, Zydek D, Koszałka L. Task allocation in distributed mesh-connected machine

- learning system: Simplified busy list algorithm with q-learning based queuing[C]//Proceedings of the 8th International Conference on Computer Recognition Systems CORES 2013. Springer International Publishing, 2013: 763-772.
- [53] Liu Y, Xie S, Zhang Y. Cooperative offloading and resource management for UAV-enabled mobile edge computing in power IoT system[J]. IEEE Transactions on Vehicular Technology, 2020, 69(10): 12229-12239.
- [54] Xu F, Yang F, Zhao C, et al. Deep reinforcement learning based joint edge resource management in maritime network[J]. China Communications, 2020, 17(5): 211-222.
- [55] Luo Q, Li Z, Du W, et al. Uav task allocation based on behavioral mechanisms of wolf-pack hunting strategies[C]//2021 7th International Conference on Control, Automation and Robotics (ICCAR). IEEE, 2021: 179-183.
- [56] Zhang Y, Wang P, Yang L, et al. Novel swarm intelligence algorithm for global optimization and multi-UAVs cooperative path planning: Anas platyrhynchos optimizer[J]. Applied Sciences, 2020, 10(14): 4821.
- [57] Seenu N, RM K C, Ramya M M, et al. Review on state-of-the-art dynamic task allocation strategies for multiple-robot systems[J]. Industrial Robot: the international journal of robotics research and application, 2020, 47(6): 929-942.
- [58] Rantanen M, Modares J, Mastronarde N, et al. Performance of the asynchronous consensus based bundle algorithm in lossy network environments[C]//2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM). IEEE, 2018: 311-315.
- [59] Johnson L B, Choi H L, How J P. The role of information assumptions in decentralized task allocation: A tutorial[J]. IEEE Control Systems Magazine, 2016, 36(4): 45-58.
- [60] Wu H, Li H, Xiao R, et al. Modeling and simulation of dynamic ant colony's labor division for task allocation of UAV swarm[J]. Physica A: Statistical Mechanics and its Applications, 2018, 491: 127-141.
- [61] Lerman K, Jones C, Galstyan A, et al. Analysis of dynamic task allocation in multi-robot systems[J]. The International Journal of Robotics Research, 2006, 25(3): 225-241.
- [62] Xie B , Chen J , Shen L .Cooperation Algorithms in Multi-Agent Systems for Dynamic Task Allocation: A Brief Overview[C]//2018 37th Chinese Control Conference (CCC). 2018.
- [63] Whitbrook A, Meng Q, Chung P W H. Addressing robustness in time-critical, distributed, task allocation algorithms[J]. Applied intelligence, 2019, 49(1): 1-15.
- [64] Li J, Chen R, Peng T. A distributed task rescheduling method for UAV swarms using local task reordering and deadlock-free task exchange[J]. Drones, 2022, 6(11): 322.
- [65] 张亚明, 李娜等. 基于精英单亲遗传算法的冷链物流 VRP 模型优化研究[J]. 数学的实践与

- 认识, 2016(04): 87-96.
- [66] 王俊淑, 殷明浩, 谷文祥等. 基于整数线性规划的智能规划研究[J]. 世界科技研究与发展, 2008, 030(005): 577-584.
- [67] Schumacher C, Chandler P R, Rasmussen S R. Task allocation for wide area search munitions[C]//Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301). IEEE, 2002, 3: 1917-1922.
- [68] 庄伟卿, 吴琮. 突发疫情下考虑优先级和动态需求的应急物资配送[J]. 福州大学学报: 哲学社会科学版, 2022, 36(4): 54-62.
- [69] 常友渠, 肖贵元, 曾敏. 贪心算法的探讨与研究[J]. 重庆电力高等专科学校学报, 2008, 13(3): 40-42.
- [70] 李敏强. 遗传算法的基本理论与应用[M]. 北京:科学出版社, 2002.
- [71] 赵新超, 郭赛. 遗传算法求解多旅行商问题的相对解空间分析[J]. 智能系统学报, 2018, 13(5):760-768.
- [72] 李茂军, 童调生. 单亲遗传算法及其应用研究[J]. 湖南大学学报: 自然科学版, 1998, 25(6): 56-59.
- [73] Zhou H, Song M, Pedrycz W. A comparative study of improved GA and PSO in solving multiple traveling salesmen problem[J]. Applied Soft Computing, 2018, 64: 564-580.
- [74] 胡杰, 鲍帆, 石潇竹. 基于贪婪-遗传算法的机场登机口分配策略[J]. 系统工程与电子技术, 2023, 45(11): 3555-3564.
- [75] Mehrabian A R, Lucas C. A novel numerical optimization algorithm inspired from weed colonization[J]. Ecological Informatics, 2006, 1(4): 355-366.
- [76] Gregory J M, Al-Hussaini S, Gupta S K. Heuristics-based multi-agent task allocation for resilient operations[C]//2019 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR). IEEE, 2019: 1-8.
- [77] Ponda S S. Robust distributed planning strategies for autonomous multi-agent teams[D]. America: Massachusetts Institute of Technology, 2012.
- [78] Liu R, Seo M, Yan B, et al. Decentralized task allocation for multiple UAVs with task execution uncertainties[C]//2020 International Conference on Unmanned Aircraft Systems (ICUAS). IEEE, 2020: 271-278.
- [79] Fu X, Zhang J, Zhang L, et al. Coalition formation among unmanned aerial vehicles for uncertain task allocation[J]. Wireless Networks, 2019, 25: 367-377.
- [80] Zhao X, Zong Q, Tian B, et al. Fast task allocation for heterogeneous unmanned aerial vehicles through reinforcement learning[J]. Aerospace Science and Technology, 2019, 92: 588-594.
- [81] 张玉州, 张子为. 考虑动态客户需求的物资配送问题求解方法[J]. 西安交通大学学报,

2020, 54(8): 124-131.

- [82] Shi Y, Eberhart R. A modified particle swarm optimizer[C]//1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence (Cat. No. 98TH8360). IEEE, 1998: 69-73.
- [83] Zeng N, Wang Z, Liu W, et al. A dynamic neighborhood-based switching particle swarm optimization algorithm[J]. IEEE Transactions on Cybernetics, 2020, 52(9): 9290-9301.
- [84] 吴立志, 崔彦琛, 朱红伟, 等. 基于离散粒子群算法多旋翼无人机任务分配研究[J]. 消防科学与技术, 2020, 39(5): 662-665.