学校代码	10699
分类号	TP391.9
密　级	公开
学　号	2021262279

西北工業大學
NORTHWESTERN POLYTECHNICAL UNIVERSITY

硕士学位论文

专业学位研究生

题目　编队协同智能空战战术决策模型研究

作者　周星宇

专业领域　电子信息

指导教师　张建东 刘洁凌

培养单位　电子信息学院

申请日期　2024 年 3 月

西 北 工 业 大 学

硕 士 学 位 论 文

题目：编队协同智能空战战术决策模型研究

专业领域：电子信息

作　　者：周星宇

指导教师：张建东

2024 年 03 月

Title: Research on Tactical Decision-Making Models for Coordinated Squadron-Based Intelligent Aerial Combat

By
Zhou Xingyu
Under the Supervision of Associate Professor
Zhang Jiandong

A Dissertation Submitted to
Northwestern Polytechnical University

In Partial Fulfillment of the Requirement
For The Degree of
Master of Electronic Information

Xi'an P. R. China
March/2024

学位论文评阅人和答辩委员会名单

学位论文评阅人名单

姓名	职称	工作单位
全盲评阅	无	无
全盲评阅	无	无

答辩委员会名单

答辩日期			2024 年 3 月 11 日
答辩委员会	**姓名**	**职称**	**工作单位**
主席	宋海浪	研究员	中国飞行试验研究院
委员	吴勇	教授	西北工业大学 电子信息学院
委员	史国庆	副教授	西北工业大学 电子信息学院
秘书	杨啟明	助理研究员	西北工业大学 电子信息学院

摘 要

随着无人机相关领域研究的不断深入，多无人机协同规划成为当下无人机研究的主流方向。本文面向无人机空战战术决策展开研究，主要包括进入作战区域前的航路规划引导和进入作战区域后的协同战术决策。在进入作战区域前，对多无人机进行航路规划引导，在进入作战区域后，对无人机空战机动决策和编队协同策略模型分别展开研究。基于以上内容，本文主要从以下三个方面展开研究：

(1) 针对无人机协同航路规划问题，本文提出一种基于改进灰狼优化算法的无人机协同航路规划模型，实现了抵达作战区域前的多机引导与航路规划。首先，构建了无人机协同航路规划模型，包括协同约束分析、单机评价函数和多机协同航迹评价函数等；其次，针对灰狼算法（Grey Wolf Optimizer, GWO）收敛速度慢和易陷入局部最优等问题进行改进，利用非线性函数优化收敛系数同时根据不同灰狼的适应度设计权重比例因子，建立了改进的灰狼优化算法（NI-GWO）模型；最后，建立多无人机航路规划的仿真平台，完成算法的验证，并对算法的鲁棒性进行分析。仿真结果表明，与传统算法相比，本文提出的 NI-GWO 模型在解决多机航路规划问题中拥有着卓越的表现，同时算法在迭代次数和作战场景上有较强鲁棒性。

(2) 针对空战机动决策问题，本文提出了一种基于 E-SAC 算法的无人机空战决策模型，实现了空战机动决策中对目标的追踪与打击。首先，建立导弹攻击区解算的数学模型，包括导弹运动与导引建模、攻击区拟合；其次，针对 SAC 算法在解决机动决策问题时收敛速度慢和易陷入局部最优等问题进行了改进，利用 Expert Actor 增强智能体的探索效率，建立了 E-SAC 算法模型；最后，基于 MaCA 平台实现无人机一对一场景下的空战机动决策仿真，并对改进算法的有效性进行分析。仿真结果表明，本文所提的 E-SAC 算法模型可以解决无人机机动决策收敛速度慢和易陷入局部最优的问题，同时该算法相较于传统 SAC 算法在收敛速度和机动决策方面有显著的提升。

(3) 针对无人机编队协同空战决策问题，本文提出了一种基于分层强化学习的空战策略模型，实现了多无人机编队协同空战策略模型的构建。首先，建立了无人机编队协同空战元策略模型，包括雷达开关、主动干扰、主动探测、队形转换、机动决策以及元策略模型的封装；其次，针对空战战术决策中传统的 Option-Critic 模型训练不易收敛的问题，建立了基于改进分层强化学习的无人机全流程空战策略模型；最后，基于 MaCA 平台实现无人机四对四场景下的编队协同空战战术决策仿真，并对模型的有效性进行分析。仿真结果表明，本文所提的改进分层强化学习模型可以对无人机编队做出有效的协同空战决策，同时，相较于传统的 Option-Critic 模型，所提模型展现了有效的收敛性能，并能够有效地对抗敌机。

关键词：无人机编队；战术决策；航路规划；机动决策；分层强化学习

Abstract

With the constant deepening of research in UAV-related fields, multi-UAV collaboration has emerged as a central focus in current UAV research. This paper focuses on tactical decision-making for UAV air combat, encompassing both pre-combat route planning guidance and in-combat collaborative tactical decisions. It addresses the guidance of multi-UAVs through route planning before combat and explores air combat maneuver decision-making and formation coordination strategies within combat scenarios. Specifically, the study is structured around three main aspects:

(1) In response to the complexities involved in the coordinated flight path design for Unmanned Aerial Vehicles (UAVs), this study introduces a cooperative route planning framework for UAVs utilizing an enhanced version of the grey wolf optimization algorithm. Designed to streamline multi-UAV navigation and strategy formulation before entering combat zones, the framework comprises a UAV cooperative route planning model. Firstly, a UAV cooperative route planning model is constructed, including cooperative constraint analysis, single aircraft evaluation function and multi-aircraft cooperative trajectory evaluation function, etc. Second, to address the slow convergence and propensity for local optima characteristic of the conventional Grey Wolf Optimizer (GWO), this study introduces modifications that optimize the convergence coefficient using a non-linear function and adjust the algorithm's weighting based on the adaptability of each Grey Wolf. The improved grey wolf optimization algorithm (NI-GWO) model is established by using the nonlinear function to optimize the convergence coefficient while designing the weight proportion according to different grey wolf adaptations. Finally, a dedicated simulation environment for the route planning of multiple UAVs has been developed, serving to confirm the efficacy of the algorithm and assess its resilience. Simulation results demonstrate that the NI-GWO model surpasses traditional algorithms in tackling the complexities of multi-UAV route planning and exhibits considerable robustness across various combat scenarios and iterations.

(2) Addressing the challenge of air combat maneuver decision-making, this research formulates a UAV air combat maneuver model leveraging the E-SAC algorithm to facilitate target tracking and engagement. Firstly, it constructs a mathematical model for determining the missile attack

zone, incorporating aspects of missile dynamics, guidance systems, and attack zone estimation. Secondly, to mitigate the slow convergence and propensity for local optima in maneuver decision-making, the SAC algorithm is refined with an Expert Actor, leading to the E-SAC algorithm that enhances the exploratory efficiency. Finally, this study implements and validates a tailored UAV air combat decision-making model for one-on-one engagements on the MaCA platform. The simulations demonstrate that the E-SAC algorithm outperforms the conventional SAC algorithm by significantly improving convergence speed and maneuver decision-making efficiency, effectively addressing the challenges of sluggish convergence and susceptibility to local optima.

(3) Aiming at enhancing UAV formation and cooperative air combat decision-making, this paper explores a strategy model based on hierarchical reinforcement learning. This approach facilitates the development of a multi-UAV formation cooperative air combat strategy. Initially, a coordinated air combat meta-strategy model for UAV formations is developed, encompassing elements such as radar operation, active jamming, active detection, formation transition, and maneuver decision-making. This includes the encapsulation of the meta-strategy model for streamlined strategy application. To address challenges in convergence difficulties in the traditional Option-Critic model training within aerial combat tactical decision-making, an improved full-process UAV air combat strategy model employing hierarchical reinforcement learning is proposed. Moreover, a specific UAV four-to-four air combat strategy model is executed using the MaCA platform, focusing on formation coordination in a four-to-four scenario. The model's effectiveness is validated through simulation, illustrating that the advanced hierarchical reinforcement learning model facilitates effective cooperative aerial combat decisions. The model introduced not only surpasses the conventional Option-Critic framework in achieving efficient convergence but also demonstrates superior capability in effectively countering adversarial aircraft.

Key words: UAV formation; Tactical decision-making; Route planning; Maneuver decision-making; Hierarchical reinforcement learning.

目 录

图索引

表索引

第 1 章　绪论

本章首先讨论编队协同智能空战战术决策模型的研究背景与意义。在此基础上，分别对无人机协同航路规划、无人机空战机动决策和无人机编队协同空战战术决策的国内外研究现状进行总结和分析，最后，凝练出本文的研究内容和创新点。

1.1 研究背景及意义

在军事领域，无人机（Unmanned Aerial Vehicle, UAV）技术的飞速发展[1]正彰显其日益重要的战略地位。无人机因其卓越的机动性和灵活性而作为未来先进武器系统新的发展方向[2]。伴随着计算机、电子信息技术的突飞猛进，无人机技术已取得显著的进步[3]，从而极大地推动了军事任务向多样化和自主化的演进。

在现代战争中，无人机的单机作战已经扩展到多机编队协同作战[4]。这种转变不仅体现了技术的进步，更标志着战术和战略层面的创新。无人机编队协同作战能力的提高，意味着能更有效地应对快速变化的战场环境，实现更高的作战效率和适应性。因此，深入探索和优化无人机编队的协同决策机制变得尤为重要，这不仅推动了无人机技术的发展，也为未来的空中作战策略提供了新的视角。目前，人工智能算法在多无人机协同作战决策算法中扮演着重要的角色。

自 21 世纪以来，人工智能进入了一个快速发展的阶段，它的核心是通过模拟人类的认知过程，使得机器能够执行原本需要人类智能才能完成的复杂任务[5]。随着技术的不断进步，人工智能已经超越了简单任务的自动化，扩展到需要复杂决策和认知能力的领域。强化学习作为人工智能的一个关键分支，通过奖励机制引导机器学习策略，已在围棋[6]和电子游戏[7]等领域展现出超越人类的能力。如今这种基于强化学习的人工智能技术也正逐渐应用于军事领域[8]，特别是在编队协同空战战术决策中[9]，通过深入研究智能空战战术决策模型，人工智能可以实现更加高效、自主和智能化的协同作战策略，显著提升战斗机任务系统的性能和适应能力，也为现代战争的未来发展提供了新的视角和可能性。

在此背景下，采用智能化策略以实现无人机编队协同空战战术决策，不仅能够丰富未来军事战略和战术的理论基础，且具有实践意义。论文的研究重点为进入作战区域前对无人机的航路规划引导和进入作战区域后无人机的空战机动决策以及编队协同战术决策策略的模型构建。此研究使得无人机编队在执行攻击任务时能够独立于人工操控，根据战场上的态势变化，自主做出精准的攻击决策，以提高无人机编队在作战中的自治能力和智能化水平，推动未来无人作战系统的发展。

1.2 国内外研究现状

多无人机战术决策研究包含了无人机空战的多个流程，可以将无人机战术决策的研究划分为抵达作战区域前的航路规划和抵达作战区域后的战术决策。在进入作战区域之前，多无人机需基于飞行区域内的潜在威胁区域等要素进行多无人机航路规划；一旦进入作战区域，便需要依据实时战场情势，选取适宜的战术行动方案与敌机进行对抗，完成整个空战流程。其中空战战术决策包含顶层编队战术策略和底层单无人机机动策略。基于此，本文的三个关键技术分别为：无人机协同航路规划、无人机空战机动决策和无人机编队协同策略模型构建。

1.2.1 无人机协同航路规划研究现状

多无人机航路规划涉及为每架无人机根据其任务需求制定一条从起始位置到目标地点的航向，目的是满足特定的性能指标并实现最优或近似最优的结果[10]。

相较于单一无人机的航迹规划设计，多无人机的航迹规划面临更高的复杂度，这主要表现在几个方面：首先，规划所需考虑的空间更广阔且更加复杂，需要应对多样的空间障碍和威胁因素；其次，所需满足的限制条件增多，不仅要确保所规划的航线能够符合无人机的动力学与运动学特征，还需确保时间与空间上的协调；最后，航迹规划必须能够适应战场环境的动态变化，并能实时调整航线。

鉴于多无人机系统在航迹规划方面的高度复杂性，传统的算法方法不再能够有效应对如此大规模的规划需求。目前，国内外研究学者提出了不同类型的元启发式算法来解决无人机自主轨迹规划问题。例如人工势场方法[11]、模拟退火算法[12]。文献[13]介绍了一种结合遗传算法与进化策略的混合方法，旨在解决电磁优化难题。该算法经过调整，以更好地适应灾难情形。文献[14]提出了一种自适应约束差分进化算法，该算法通过考量个体的适应度和约束条件来挑选个体。文献[15]提出了离散杜鹃搜索和模拟退火算法来解决连续优化问题。启发式优化算法具有快速规划速度、良好的并行性和易于操作的优势，但该类算法易陷入局部最优解，并且非常依赖启发式信息，这使得算法复杂性、普适性以及收敛速度难以确定。

另一种是群体智能优化算法，越来越多的研究致力于实现算法的收敛性和搜索能力之间的平衡[16-18]。群体智能优化算法基于对生物群体行为的研究，生物群体通过合作、信息交换和群体之间的互动来实现优化目标。常见的群智能算法如粒子群优化算法[19]、蚁群优化算法[20]。文献[21]提出了一种改进的粒子群优化算法，通过智能控制粒子数量以优化电机设计。实验结果表明，该算法能有效减少函数调用次数，成功降低了背电势的总谐波失真（THD）。蚁群优化算法[22][23]常用于解决单一K-Means算法的过拟合问题。改进的鲸鱼优化算法[24]通过结合逻辑映射技术，显著增强了在参数识别方面的应用效果，

不同于众多经典优化方法，此算法基于个体通过自然选择和无意识的适应性行为进行生存状况的优化，从而适应多样化的任务场景。尽管上述提及的群智能优化算法在许多应用场景中表现出色，但它们常面临陷入局部最优解的挑战。

为了解决这一问题，Mirjalili 在受到灰狼捕猎策略的启发后，提出了灰狼算法(Grey Wolf Optimizer, GWO)[25]。此算法基于灰狼群体的多层社会结构，并利用狼群的社会行为来寻找和围捕猎物，以确定围捕猎物的最佳位置，与其他元启发式算法相比，GWO 算法由于存在全局搜索和局部搜索的机制，因而提高了解算全局最优解时的效率和准确性，通常能够在较少的迭代次数内找到解决方案，这有助于减少计算时间，并且实现相对简单，不需要复杂的参数调整，易于使用和理解。该算法模拟了灰狼的社会行为和领导层次结构。GWO 算法的仿生结构使其具有在实际应用中相对容易实施和在处理各种类型的优化问题时表现出较好的灵活性[26]。GWO 算法目前已经在众多工程和控制领域得到应用，包括但不限于负载频率控制[27]、特征筛选[28]以及无人机航迹规划[29]等问题的解决上。文献[30]提出了一种启发式灰狼优化器来解决数值优化问题。文献[31]提出了一种受天体物理学启发的灰狼算法来解决工程设计问题。文献[32]提出了一种探索增强型灰狼优化器来解决高维数值优化。

根据对现有文献的分析，可以观察到群体智能优化算法在处理多无人机航路规划问题时显示出一定的优势。然而，也有观点指出，传统的群体智能优化算法在某些情况下可能会面临一些挑战，如有时可能会陷入局部最优以及收敛速度可能相对较慢[33]。因此需要对算法进一步改进和优化。借助文献[34-39]中的先进优化思想，本文对 GWO 算法进行改进，通过非线性函数优化收敛系数并为不同适应度的灰狼分配不同的权重比例因子，以增强 GWO 算法的探索和开发能力并提升传统 GWO 算法的收敛速度，当存在较大的收敛系数和比例权重时，将有助于算法提高收敛速度，并更快地收敛到更优解的附近。

1.2.2 无人机空战机动决策研究现状

在执行作战任务期间，无人机的独立机动决策能力指其能够基于空战环境和局势等多重因素，自行作出机动策略决定[40]。面对空中作战环境的复杂化以及任务需求的多元化，基于地面站控制的无人机空战机动决策已经无法满足复杂的空战需求。基于此，国内外相关学者对无人机机动决策领域的研究主要分为：基于经典算法和基于人工智能的决策方法。

近年来，经典算法在解决无人机空战机动决策问题时主要以博弈论等算法为代表，如文献[41]中，研究者开发了一种基于博弈矩阵的方法，可以通过比较不同机动组合的方向、距离、速度和地形间隙得分，生成丘陵地形中一对一空战的低空飞行飞机的智能机动决策。文献[42]则设计了一种基于微分博弈理论的生成算法，用于近距离空对空战

斗的无人机，它采用层次决策结构和评分函数矩阵来评估无人机的空中优势。文献[43]介绍了一种用于模拟一对一空战中飞行员机动决策的多阶段影响图博弈方法，这一方法不仅以图形化的方式展示了决策过程和飞机动力学模型，还考虑了飞行员在不确定性条件下的偏好，为最优空战机动策略分析和空战模拟器中战斗机动选择的自动化决策系统开发提供了新的思路。文献[44]深入探讨了一对一空中战斗的微分博弈模型，并分析了双目标博弈公式及其解决方案的复杂性。文献[45]则通过分析三维轨迹、相对角度的历史和占有率表格，研究了基于微分博弈的一对一近视距离空对空战斗机动生成算法，该算法结合评分函数矩阵和敌方机动分析来确定最优战术动作。文献[46]介绍了一种处理一对一空战机动中水平飞行定速问题的近似动态规划方法，该方法能够快速适应不断变化的战术环境，并提供了长期规划视角和卓越性能，无需对空战策略进行具体编码。文献[47]介绍了一种适用于未知环境中的改进人工势场法，结合 Tangent Bug 算法进行无人机的避障操作，并在有威胁区的环境下利用 A*算法生成预规划路径。

在机动决策的研究领域中，智能算法已成为推动技术创新的主要驱动力。这些算法，特别是基于人工智能的方法，在处理复杂决策环境和动态适应新情况方面表现出卓越的能力。文献[48]在构建超视距空中对抗训练环境的基础上，提出了启发式 Q-Network 方法，结合专家经验和强化学习，有效地提升了空中对抗机动策略的学习效率和实战应用性能。文献[49]则开发了基于端到端强化学习的深度 Q 网络人工智能策略，它能够直接从高维感官输入中学习，并在一系列游戏测试中达到与专业人类玩家相当的性能。文献[50]利用飞机简化运动学方程、改进的三自由度控制指令，以及基于轨迹采样、特征提取和策略学习的方法，开发了一种近似动态规划方法，显著提升了无人机在模拟空战训练中的智能决策能力。文献[51]开发了一种利用深度强化学习技术的无人机近距离空战控制策略模型，通过锁定参数网络重启机制优化深度确定性策略梯度算法，显著提高了无人机在复杂战斗机动控制方面的性能和效率。文献[52]设计了包含能量状态的奖励函数，并将其应用与强化学习算法之中，实现了空战智能决策，经验证，该文献所提出的模型可以有效提升无人机空战决策的能力。文献[53]设计了基于粒子群优化径向基函数的预测方法，结合模拟操作命令和最终奖励值，提出了一种深度确定性策略梯度算法，能够在一定时间内将最终奖励值反馈到之前的奖励值进行离线训练，有效提高了算法的收敛速度。文献[54]采用基于邻近策略优化的深度强化学习方法，专注于自主近距空战策略的学习。通过端到端的方式从观察中提炼策略，并在设计的连续动作空间中实施，显著提高了学习效率，并能以大约 97%的胜率成功击败极小极大策略的对手。文献[55]提出了改进的深度确定性策略梯度(Deep Deterministic Policy Gradient, DDPG)算法，修改了无人机模型，并基于向量距离处理了经验数据，以提高计算效率。基于此，研究团队进一步在 DDPG 算法的框架下，引入了改进型算法——双延迟深度确定性策略梯度

(Twin Delayed Deep Deterministic Policy Gradient, TD3)算法[56]，优化了 DDPG 算法中值函数过估计的问题。在文献[57]中，基于 TD3 的无人机导航算法被提出，允许无人机在有多个威胁区的动态环境中执行导航任务。虽然该方法可以通过在先前任务中获得的知识或经验减少在各种应用场景中策略的训练成本，但所需的先验知识仍需要长时间训练。由于空中战斗过程中的状态空间和动作空间较为庞大，深度强化学习算法在解决无人机在空战中的机动决策问题时，对策略的探索将会变得较为困难。与确定性策略算法不同，软演员批评家(Soft Actor-Critic, SAC)算法[58]旨在最大化通过熵正则化的累积奖励，而不仅仅是累积奖励。SAC 算法可以增加行动的随机性，鼓励智能体在训练过程中进行探索，从而加快后续学习速度[59]。许多基准研究已经证明了其最先进的性能。然而，该算法仍有一些局限性，在训练过程中需要大量样本来学习良好的无人机机动决策策略，这导致训练时间长和训练期间样本利用率低。

对上述文献分析可得，基于经典算法的无人机机动决策方法在某些情况下可能表现出一定的局限性，例如在灵活性、实时性以及在处理复杂问题方面。同时，基于学习的方法中的确定性策略算法在训练时间和收敛速度方面也存在一些挑战。这些为无人机机动决策的进一步研究提供了改进的空间和潜在的研究方向。因此，论文采用了人工智能算法对无人机机动决策问题进行研究，并针对其样本利用率低的缺陷进行改进，提出一种 Expert-Actor Critic 算法，通过 Expert Actor 采样环境来获取 Expert 经验样本，防止算法陷入局部最优，并加速其训练过程。

1.2.3 无人机编队协同空战战术决策的研究现状

随着空战环境的日益复杂化和任务需求的多样化，单无人机在面对复杂战场情况时，作战能力会产生明显的劣势。因此，多无人机系统协同执行作战任务是未来无人机发展的主流趋势[60]，相关学者已开始广泛研究多无人机系统的协同决策问题。目前主流的多机空战战术决策方法有：基于规则的方法[61][62]、基于搜索的方法[63][64]和基于学习的方法[65][66]。

基于规则的方法依赖于专业知识，通过抽象出具有 IF-THEN 结构的问题特定逻辑树来进行决策。文献[61]研究了无人机群体的空中作战战术，构建了一个简单的规则集，并在仿真分析中对智能体进行设计并建立了相关空战模型，以模拟多无人机协同战术。文献[62]开发了一个基于遗传模糊树的二对四的空中作战决策模型。它在模拟空中作战中击败了专业人类飞行员。然而，由于硬编码规则的局限，这种方法不够灵活，无法应对超出专业知识的复杂情景。

基于搜索的方法为使用一些并行搜索和迭代机制来找到明确目标函数的次优解。文献[63]使用多目标遗传算法找到多无人机任务规划问题的帕累托前沿解。文献[64]提出了一种智能自组织蚂蚁群优化算法，用于多无人机协同搜索和攻击任务规划。然而，这

种方法需要明确的目标函数，并执行在线搜索。因此，基于搜索的方法的实时性较差。随着问题变得更加复杂，找到最佳解也变得更加困难。

基于学习的方法主要以强化学习(Reinforcement Learning , RL)算法为核心，该方法主要通过智能体反复试验学习以获得最优策略，其中神经网络用于映射状态和行动。文献[65]提出了一种基于多智能体强化学习的三对三空中作战方法，该算法在群体空中作战环境中训练了多个智能体，并通过采用场景迁移训练及自我博弈策略，进一步增强了多智能体强化学习过程中的效率提升。文献[66]开发了一种依托于强化学习的策略，用于优化无人机群体内的任务分配，目标是提升群体间的协作效率。此算法赋予无人机根据即时数据动态调整其任务执行策略的能力，有效处理动态环境中的任务执行，并通过解决信道分配等关键问题，显著提高了无人机集群的任务性能效率和协作能力。文献[67]研究了深度强化学习算法在多无人机协同控制中的应用，并提出了一种改进的多智能体联合邻近策略优化算法，该算法通过集中学习和分散执行以及移动窗口平均法，有效提高了多无人机系统的协作性和奖励值总和。文献[68]介绍了一个基于 F-16 Fighting Falcon 飞机的仿真模型，用于验证飞机在进行地面碰撞避免和其他自主机动时的性能。该模型包含 16 个连续变量和分段非线性微分方程，使用 MATLAB 和 Python 进行模拟。通过内外环控制器和有限状态机实现自主机动，该模型旨在作为分析航空航天系统行为的基准测试和起点。文献[69]提出了一种新型的自主空战算法，专注于视距内枪炮交战，并基于基本战斗机动进行高性能、实时计算。

在利用强化学习解决全流程空战决策问题时，需要克服在空战环境中奖励稀疏的问题，因此奖励函数的设计的复杂性使得将强化学习应用于空战决策之中成为了挑战[70][71]。文献[72]构建了一个高保真度的空战模拟环境，并提出了一个关键的空战事件奖励塑形机制来减少情节性的胜负信号，使训练过程快速收敛。实施结果表明，强化学习可以在超视距条件下产生多种有价值的空战战术行为。文献[73]研究了在超视距空战中，利用深度强化学习算法提升多架无人机的合作决策和智能优化能力。实验验证了深度强化学习可以提高超视距空战中多飞机协同决策的智能化水平。文献[74]基于原始深度强化学习方法设计了一种奖励函数，奖励的设计维度包括机动带来的实时收益以及最终结果的收益。对于奖励稀疏的空战机动决策问题，文献[75]应用了一种基于课程的学习方法来设计一个角度、距离和混合的决策课程，与原始方法相比，提高了训练的速度和稳定性，能够处理来自不同方向的目标。文献[76]提出将课程设计视为马尔可夫决策过程，对智能体与任务互动时知识的积累进行建模，并提出了一种近似执行最优策略的方法，以生成适用于每个智能体的个性化课程。文献[77]通过采用多智能体邻近策略优化算法，成功开发了针对超视距空战的无人作战飞行器协同决策方法。文献[78]提出了一种多智能体分层策略梯度算法，用于解决空对空对抗中的作战策略选择问题。该算法通过对抗性

自我博弈学习以掌握多元化策略，并借助分层决策网络高效处理复合性和多样化动作。

由以上研究可以看出，目前对多无人机战术决策问题的研究主要聚焦在特定场景中引入强化学习训练后决策逻辑的合理性[79][80]，但是各种独立状态的组合形成了一个非常大的决策空间，这会导致状态维度的爆炸[81]的问题。于是研究人员提出了一个基于分层强化学习(Hierarchical Reinforcement Learning, HRL)的空战框架[82]以降低强化学习决策方法的训练难度，这一框架将决策过程分为顶层策略和底层策略两部分，其中顶层策略根据敌机的相对位置和我机的状态，向底层策略发送宏观指令，底层策略负责直接操控无人机以响应这些宏观指令。目前采用分层强化学习来解决编队协同智能空战战术决策问题的相关研究较少。主要研究包括，文献[83]采用了结合分层架构和最大熵强化学习的方法，通过 Soft Critic-Actor 算法[84]训练顶层策略和元策略，并且该方法整合了专家知识和策略模块化。文献[85]对多无人机近距离自主空中战斗决策的问题进行研究，文章设计了一种改进的分层强化学习架构，使其能够处理不同规模编队的空战场景，其结合了循环 Soft Critic-Actor 算法和竞争性自我对弈来学习子策略。文献[86]提出了一个分层多智能体强化学习框架，用于解决多个智能体的空对空战斗决策问题。该框架将决策过程分为两个层次，低层次策略控制个体单位的行动，高层指挥策略根据整体任务目标发布宏观指令。文献[87]引入了一种改进的分层机动决策架构并将其应用于双飞机近距离空中战斗场景中。文章结合了 Soft Critic-Actor 算法和竞争性自我对弈，以整合子策略的知识，并使用强化学习技术实现了近似的纳什均衡主策略。

通过对文献的深入分析，发现在处理多无人机的战术决策问题时，传统的算法可能会遇到计算需求大和实时响应缓慢的问题。强化学习方法在某些方面表现出色，但它也面临着奖励稀疏和维度爆炸等问题。因此，在利用强化学习对无人机编队协同智能空战战术决策进行研究时，仍有许多关键问题需要进一步解决。为了解决分层强化学习中模型的不敏感性，终止函数设置不合理导致算法最终无法收敛到稳定状态的问题。本论文根据不同的任务设计了相应的元策略模型，并通过将该模型封装并输入环境状态返还动作，利用 Option-Critic 网络训练终止函数，以完成无人机编队在空战场景中的作战任务需求，增强多无人机协同战术决策的智能化。

1.3 论文研究内容与创新点

1.3.1 论文研究内容

随着人工智能技术的迅速进步和无人机在军事上的广泛部署，研究多无人机协同智能战术决策变得尤为重要。智能化和自主化的无人机编队不仅能提升单机作战的效率，还能有效优化整体作战策略。因此，如何将人工智能技术有效融入多无人机协同空战战术决策以增强其自主决策和协同作战能力，已成为一个研究热点。

本文根据无人机编队协同战术决策的需求，将研究内容分为三个部分。首先是抵达

作战区域的航路规划问题，本文采用改进的灰狼算法来规划无人机编队的自主航路。其次是在作战区域内的无人机机动决策问题，运用深度强化学习的方法来制定无人机的空战机动决策。最后是在作战区域内的无人机编队协同战术决策问题，本文基于分层强化学习算法完成无人机编队协同空战战术决策。本文的研究内容如图 1-1 所示。

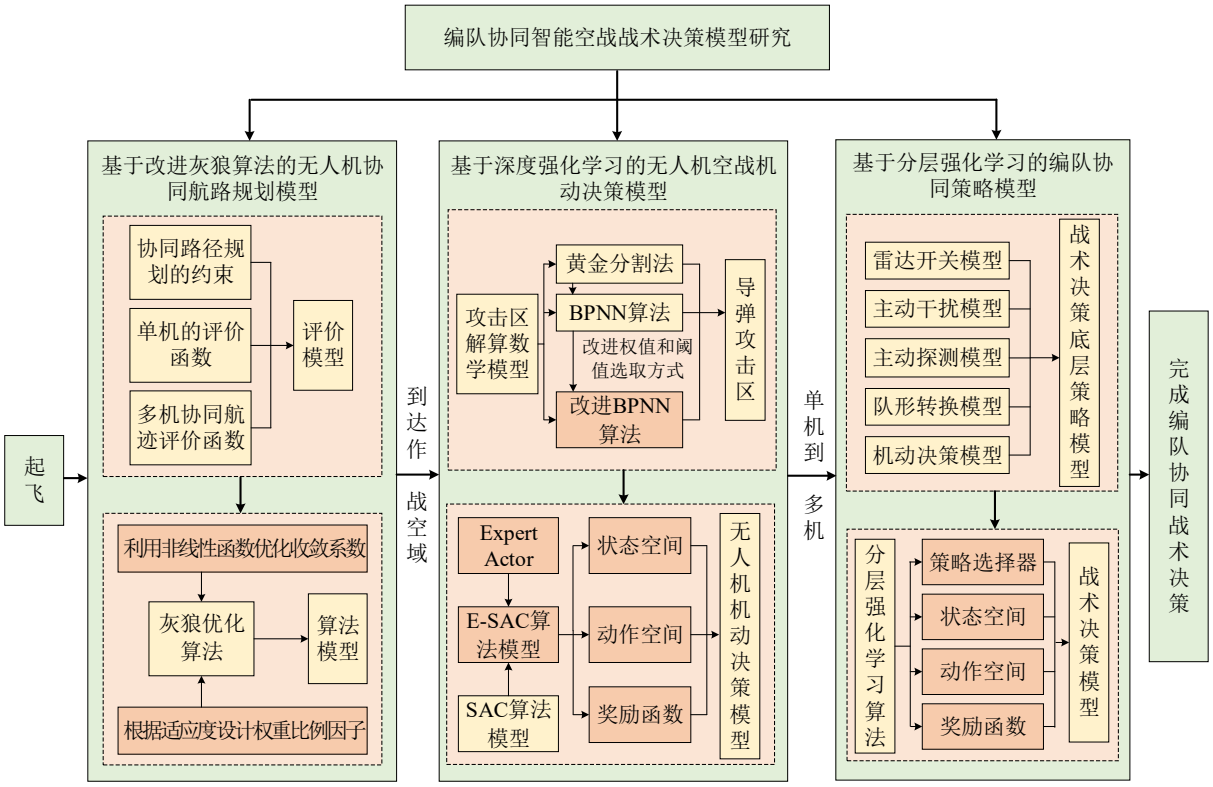


图 1-1 论文研究内容

论文研究内容如下：

(1) 基于改进灰狼算法的无人机协同航路规划模型

在抵达作战区域前，首先需要进行自主航路规划。针对灰狼算法在多无人机航路规划的不足，提出了一种面向多无人机的改进灰狼优化算法用于优化多机的航路规划问题。首先，对灰狼优化算法在多机航路规划的不足进行改进，利用非线性函数优化灰狼算法的收敛系数并根据不同灰狼的适应度设计权重比例因子，提出面向多机的改进灰狼优化算法；其次，基于 MATLAB 设计多无人机航路规划的仿真平台，模拟多无人机躲避威胁区并抵达任务目标点的作战任务；最后，将本文所提的 NI-GWO 算法代入仿真平台进行验证，并将其与传统的 GWO 算法和 MP-GWO 算法的多机航路规划性能进行比较并分析算法的鲁棒性。

(2) 基于深度强化学习算法的无人机空战机动决策模型

在抵达作战区域后，每架无人机必须根据目前的空中战斗情况来做出独立的机动决

策。针对传统 SAC 算法机动决策状态空间大、训练不易收敛等问题，介绍了一种名为 E-SAC 的算法，旨在为无人机空中战斗环境中提供机动决策策略。首先，对导弹攻击区进行解算，基于改进 BP 神经网络实现导弹攻击区的快速拟合；其次建立无人机空战机动决策模型，提出一种专家演员经验嵌入的 E-SAC 算法。最后，基于 MaCA 仿真平台完成无人机机动决策策略的仿真，并将本文所提的 E-SAC 算法与传统的 SAC 算法进行对比，以验证算法性能。

(3) 基于分层强化学习的编队协同空战战术决策模型

在到达作战区域后，无人机编队需要根据当前的空战态势做出编队协同战术决策。针对多机空战中编队协同决策响应速度慢的问题，基于分层强化学习算法，提出了一种无人机编队协同空战策略模型。首先，建立无人机编队协同决策的元策略模型，包括基于专家经验的雷达开关策略、主动干扰策略、主动探测策略以及队形转换策略，完成多维元策略模型的封装。其次基于 Option-Critic 算法的策略选择器进行改进，提出改进 Option-Critic 算法的策略选择器来训练无人机编队协同战术决策模型，完成顶层策略选择器的构建；最后，基于 MaCA 仿真平台对本文构建的基于分层强化学习无人机编队协同策略模型进行验证，并将本文所提改进 Option-Critic 算法与传统的 Option-Critic 算法进行对比验证算法的有效性。

1.3.2 论文创新点

论文从基于灰狼优化算法的无人机协同航路规划模型、基于深度强化学习的无人机机动决策模型和基于分层强化学习的编队协同空战战术决策模型三个方面进行研究，创新点如下：

(1) 提出了一种基于改进灰狼优化算法的多无人机航路规划数学模型，解决了多无人机协同航路规划问题。

为克服在多无人机协同航路规划的过程中，传统灰狼优化算法易于落入局部最优解的难题，本论文构建了一种改进灰狼优化算法的多无人机航路规划模型。通过在灰狼算法的基础上引入非线性函数调整其收敛参数，并利用灰狼适应度值来更新算法中的位置参数，从而增强了其在未知环境探索中的性能。研究结果表明，该改进算法提升了多无人机航路规划的效率，显示出较传统方法更优的学习及收敛速度。

(2) 提出了一种利用深度强化学习技术的无人机机动决策策略，该策略有效应对了在特定攻击区域限制条件下的无人机独立机动决策挑战。

针对受导弹攻击区域终端约束的无人机自主机动的挑战，本文构建了一个创新的深度强化学习架构，以优化空中战斗机动决策过程。本论文基于 SAC 算法，针对无人机自主机动决策进行了研究，设计了基于态势评估的奖励函数，在此基础之上，提出 E-SAC 算法对自主机动决策进行研究，并利用解算得出的导弹攻击区域作为模型的限制条

件。由仿真结果可以看出，所提出的策略在处理无人机自主机动决策的导弹攻击区域终端约束问题方面是有效的，并且通过改进的方法，无人机的机动决策训练速度得到了提高，同时避免陷入局部最优。

(3) 提出了基于分层强化学习的无人机编队协同策略模型，解决了多无人机空战协同战术决策问题。

针对无人机编队协同空战战术决策问题，提出了基于分层强化学习的无人机编队协同策略模型。论文将空战任务进行分解，利用专家经验模型设计了雷达开关、主动干扰、主动探测和队形转换的元策略模型，结合(2)中设计的机动决策模型，采用分层强化学习模型，将所设计的五种不同的元策略在空战场景中进行交互，顶层策略负责学习与分析元策略的互动结果，从而使顶层策略选择器能够识别并选择出最适合的元策略。分层强化学习模型可以将这些不同的元策略结合起来，从而实现空中作战的全流程仿真模拟。结果表明，所提方法在空战对抗中表现出了高度的灵活性和适应性。

1.4 论文组织结构

基于上述研究主题与创新要素，本论文将划分为六个章节进行阐述，每章的构成如图 1-2 所示：

第一章 绪论

介绍了本文研究背景及意义，对多无人机协同航路规划、无人机空战机动决策和多无人机协同空战战术决策方法的研究现状进行了分析，同时对论文的研究主题、结构布局和创新进行了阐释。

第二章 无人机空战战术决策相关技术

引入了无人机数学模型，接着介绍了强化学习算法基本原理，其中包括了马尔可夫决策和半马尔可夫决策过程、基于价值学习和基于策略学习的强化学习方法、Actor-Critic 算法，最后分类介绍了分层强化学习算法的相关理论。

第三章 基于改进灰狼算法的无人机协同航路规划模型

首先建立了多机协同航路规划评价模型，其次基于灰狼优化算法，对其收敛系数和比例权重进行改进，最后进行仿真验证。

第四章 基于深度强化学习的无人机空战机动决策模型

首先利用 TBPNN 算法对导弹攻击区域的数学模型进行了迅速拟合，其次设计了基于 E-SAC 算法的具有终端约束条件的机动决策模型，最后通过模拟实验进行了验证。

第五章 基于分层强化学习的编队协同策略模型

首先根据无人机编队协同全流程空战任务，建立了基于分层强化学习的元策略模型，其次基于分层强化学习算法提出改进的元策略选择器，最后在 MaCA 平台上对我机的

战术决策模型进行了训练，并将其与敌机的算法进行对抗，证明了分层强化学习体系结构在实际应用中的优越性和可行性。

第六章 总结与展望

本论文综合了关于无人机编队在智能空战战术决策中的研究，并展望了论文研究的未来方向。

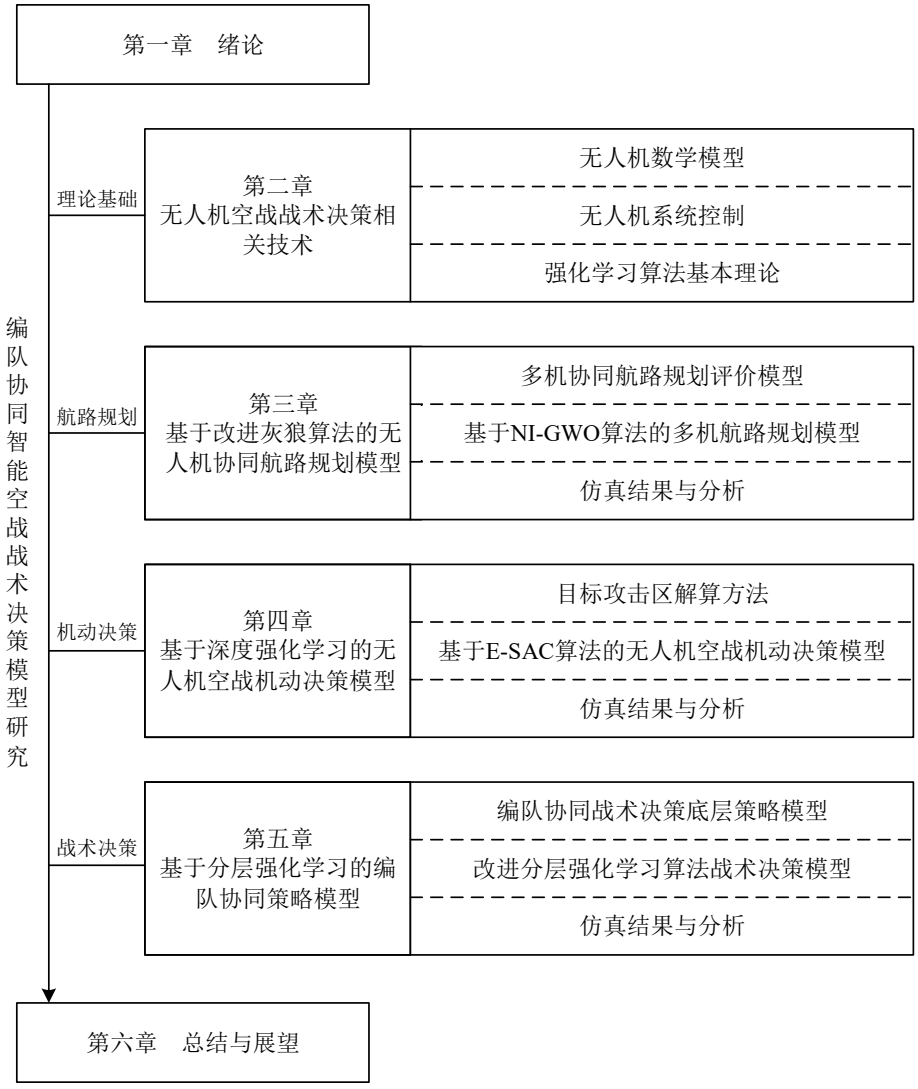


图 1-2 论文组织结构图

1.5 本章小结

本章首先介绍了无人机编队协同智能空战战术决策的研究背景并分析其研究意义；其次分别对多无人机协同航路规划、无人机空战机动决策和无人机编队协同空战战术决策的国内外研究现状进行了总结和分析；最后论文概述了其核心研究主题、创新贡献以及结构安排。

第 2 章 无人机空战战术决策相关技术

本章首先介绍了无人机动力学模型和运动学模型，其次概述了强化学习算法基本原理，包括了马尔可夫决策和半马尔可夫决策过程、基于价值学习和基于策略学习的强化学习方法和 Actor-Critic 方法，最后讨论了分层强化学习算法的基本思想，为后面章节奠定理论基础。

2.1 无人机数学模型

构建无人机数学模型时，考虑到实际飞行中系统的复杂性和多变因素，采取了一系列简化的假设以便于分析[88]：

1) 不考虑地球自转和曲率效应，忽视离心加速度和科里奥利加速度的作用，从而简化地球物理环境对模型的复杂性；

2) 假定重力加速度在无人机的不同飞行高度上保持恒定，避免了高度变化带来的计算复杂度；

3) 将无人机视作一个质量和质心位置在燃料消耗过程中保持不变的刚体，简化了动力系统和能源消耗的影响；

4) 忽略大气气流等气象条件对无人机飞行状态的影响，使模型更加专注于无人机本身的运动特性。

由此绘制三自由度的无人机质点模型如图 2-1 所示。

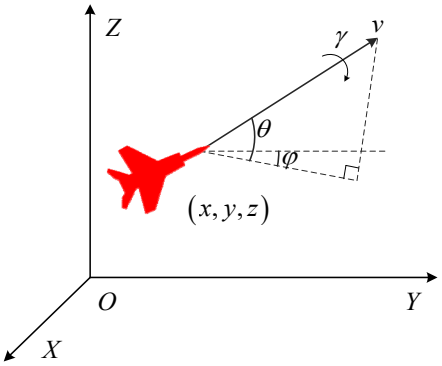


图 2-1 三自由度的无人机质点模型示意图

如图 2-1 所示，该方法在简化现实飞行条件的同时，仍然能够有效地捕捉无人机的关键运动特性，为理论分析和应用开发提供了坚实的基础。

2.1.1 无人机动力学模型

无人机动力学模型是通过数学描述无人机动力学特性的工具，其建立基于 2.1 的一系列假设，如式(2-1)所示。

$$\begin{cases} \dfrac{dv}{dt} = g(n_x - \sin\theta) \\ \dfrac{d\theta}{dt} = \dfrac{g}{v}(n_z \cos\gamma - \cos\theta) \\ \dfrac{d\varphi}{dt} = -\dfrac{n_z g \sin\gamma}{v\cos\theta} \end{cases} \tag{2-1}$$

其中，v为无人机在参考坐标系中的速度；g 为当地重力加速度；θ为无人机的俯仰角，表示无人机速度与参考坐标系水平面之间夹角；φ为航向角，指的是无人机在参考坐标系中的速度投影到水平面上与 Y 轴之间的夹角；γ为滚转角，可以用来控制无人机航向角φ；n_x和n_z分别代表无人机切向与法向过载，它们各自负责调整无人机的飞行速度v和飞行轨迹倾角度θ。

2.1.2 无人机运动学模型

无人机运动学模型通过数学方法对无人机的运动特性进行表述。在之前所述假设的基础上，推导出了一套表征无人机三自由度运动的运动学方程组：

$$\begin{cases} v_x = \dfrac{dx}{dt} = v\cos\theta\cos\varphi \\ v_y = \dfrac{dy}{dt} = v\cos\theta\sin\varphi \\ v_z = \dfrac{dz}{dt} = v\sin\theta \end{cases} \tag{2-2}$$

其中，x、y、z分别为无人机在东北天惯性坐标系中 X 轴、Y 轴、Z 轴的坐标。

2.2 强化学习理论基础

强化学习（Reinforcement Learning, RL）经历了长期的研究和发展，如今已在多领域取得广泛的应用，包括但不限于智能推荐[89-98]、游戏攻略[99-106]、生物医药[107-110]、网络安全[111]、自动驾驶[112]、机器人技术[113-117]、自然语言处理[118-124]等广泛领域。在智能推荐领域中，强化学习的应用使得网络平台可以根据每个用户的特殊需求，按顺序推荐个性化学习路径；在游戏攻略领域中，强化学习算法可以让智能体通过对游戏规则的学习，从而提升其推理能力和决策能力，从而达成游戏所需目标；在生物医药领域中，强化学习算法可以帮助人类发现新的药物或材料，提升疾病预测的准确度；在网络上安全领域中，强化学习算法可以作为检测病毒或者恶意攻击软件的入侵的关键技术，降低潜在的网络威胁对人们做出的影响；在自动驾驶领域中，强化学习的应用赋予自动驾驶汽车互动的能力，使其能够不断优化决策策略，从而在与环境互动中提高行为选择的效力；在自然语言处理领域中，强化学习算法辅助计算机实现有效的决策，从而生成连贯的文字以及准确地回复内容，提升了人机交互的能力，实现人机的无障碍沟通等。总之，强

化学习的进步，推动着不同领域的发展，也促使人类具有更好的解决问题的方法[125-126]。

强化学习的核心理念在于通过智能体与环境的互动来寻求最佳的决策策略。它依赖反馈信息来选择有效且有益的做法，同时降低那些无效和有害行为的影响，它的目标是掌握最优策略，以达成既定目标。其原理图如图 2-2 所示。

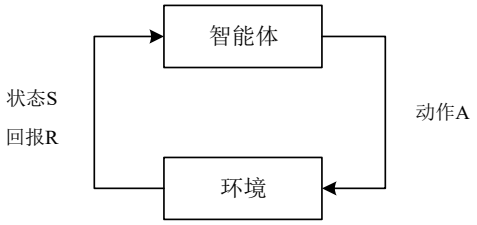


图 2-2 强化学习中智能体与环境之间的互动示意图

如图 2-2 所示，强化学习中主要涵盖四个核心要素：

策略(Policy)：智能体在当前环境下可采取的行为的映射；

回报信号(Reward Signal)：智能体在选择一个行为后所能获得的反馈，不同状态下执行相同的行为也可能会产生不同的奖励回报值，而强化学习的目标就是找到累积奖励值最大的方式，该值的大小可以显示出某个决策当下的优劣；

值函数(Value Function)：某个状态的价值，即从当前状态开始，经过一系列动作，到达中止状态所累计获得的奖励，该值的大小可以显示出某个决策长期的优劣；

环境模型(Environment Model)：智能体可以利用环境模型来预测环境所带来的响应结果。

2.2.1 MDP 与 SMDP

(1) 马尔可夫决策过程

马尔可夫决策过程（Markov Decision Process, MDP）是现实中抽象出来的一个概念，也是强化学习的数学模型，目前被广泛应用于各个领域[127]，如决策设计[128-133]、目标跟踪[134-140]、智能控制[141-144]、路况识别[145-148]、网络通信[149-151]、航路规划[152-154]、目标识别[155-161]、声呐通信[162-167]、导航定位[168-169]。

1957 年贝尔曼[170]提出了马尔可夫决策过程，即在智能体与环境的互动中学习并实现目标的过程，该过程是一个无记忆的随机过程。在强化学习中，智能体根据所处的环境状态作出决策，并执行相应的动作；环境随后对这些动作作出反应，导致智能体状态的变化。同时，环境提供奖励信号，表示智能体通过选择动作来追求最大化的数值奖励。这种连续的互动过程定义了马尔可夫决策过程。智能体通过不断调整其策略以最大化长期奖励来学习如何在特定环境中做出决策。因此，MDP 提供了一种强大的框架，可用于建模和解决各种决策问题。如图 2-3 所示。

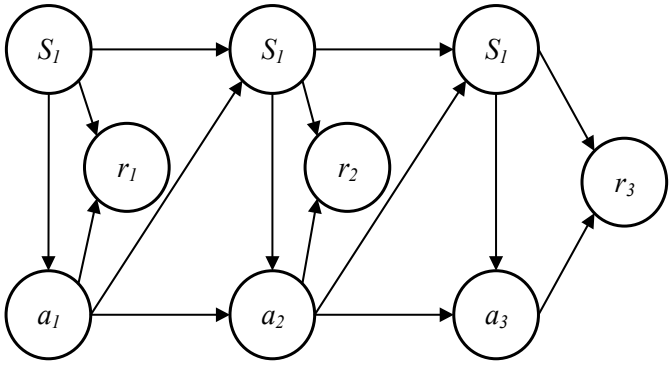


图 2-3 马尔可夫决策过程表示图

在马尔可夫决策中主要关注智能体的状态(State)、行为(Action)、最终所获得的奖励(Reward)和概率(Probability)，可用五元组$< S, A, R, P, \gamma >$进行描述，各元素的具体含义如下：

S (State Space)表示智能体所观察到的所有环境特征数量，即状态空间，通常用$S=\{s_1, s_2, \cdots s_m\}$来表示，智能体在环境中遇到的任何情形都会有一个状态与之对应。

A (Action Space)表示智能体进行决策可选择执行的所有动作，即动作空间，通常用$A=\{a_1, a_2, \cdots a_n\}$来表示，它的作用往往是改变状态。

R (Reward Function)表示在某个状态下智能体与环境互动所获得的反馈，即回报函数，如果奖励值为正，代表当前行为是被鼓励的；奖励如果为负，代表当前行为应当被惩罚。R 在状态s下，当智能体决定执行动作a，并随后切换到另一个状态s'，那么环境为这一行为所提供的实时奖励可以表示为：

$$R_{ss'}^{a} = E\left[r_{t+1} \middle| S_t = s, A_t = a, S_{t+1} = s' \right] \tag{2-1}$$

其中，$R_{ss'}^{a}$代表智能体在状态s执行动作a之后，会进入到下一个状态s'的奖励函数；r_{t+1}代表智能体在特定 $t+1$ 时刻所能得到的实时奖励，在这个复杂的智能体系统中，奖励函数的设计至关重要；A_t代表智能体在 t 时刻所进行的动作；S_t 和S_{t+1}分别代表智能体在特定 t 和 $t+1$ 时刻的状态。这一过程体现了智能体学习与决策的机制，通过评估不同动作的奖励，指导未来的行为选择。

P (Transition Probability)代表了一个状态转移的可能性，当智能体在特定的状态s中执行某一特定的动作a时，其状态会转变为另一个状态s'的可能性，如式所示：

$$P_{ss'}^{a} = P\left[S_{t+1} = s' \middle| S_t = s, A_t = a \right] \tag{2-2}$$

其中，$P_{ss'}^{a}$代表智能体在状态s采取动作a之后转移到下一状态s'时所对应的奖励函数；S_{t+1}是智能体在时间点$t+1$获得的即时奖励；A_t为智能体在时间点t所执行的动作；S_t为智能体在时间点t所处的状态。

γ为折扣因子，其范围为$0 < \gamma < 1$。在马尔可夫决策过程中，折扣因子的存在可以让

优化函数收敛于有限值，从而避免了无限循环。通常由于远期收益其不确定性较大，因此，对于越早获得的奖励，将被认为越重要，越晚获得的奖励，其衰减越严重，这导致智能体在做出决策时更加侧重于近期的回报，$\gamma = 0$ 表示只关注最近奖励，$\gamma = 1$ 表示会关注所有的价值。

每个时刻智能体执行一个动作并获得相应奖励，奖励函数和状态转移矩阵共同构成了环境，而 P 和 R 则定义了模型。马尔可夫决策过程大致可以描述为：首先，在 $t = 0$ 时刻，初始状态 s_0 以概率 $P(s_0)$ 进行随机初始化，随后，智能体根据当前状态 s_t 选择行动 a_t，环境响应智能体的动作，产生奖励 r_t，条件概率为 $R(\cdot \mid s_t, a_t)$，同时，环境根据条件概率 $P(\cdot \mid s_t, a_t)$ 确定下一时刻的状态 s_{t+1}，智能体接收奖励 r_t 和状态 s_{t+1}。这一过程不断循环进行。

(2) 半马尔可夫决策过程

在 MDP 框架下，状态转换及奖励获取均在执行动作之后的下一个时间步内发生。该假设简化了决策过程模型，但对于许多实际应用场景来说，这种简化可能不足以准确捕捉动作的持续影响，因而引入了半马尔可夫决策过程(Semi-Markov Decision Process, SMDP)作为 MDP 的一个扩展。在 SMDP 框架中，决策点不再是按固定时间间隔出现，而是可以基于更复杂的时间模型，这意味着在两次连续的决策之间，可以有不同的时间跨度，这更贴近那些动作结果需要时间发展的现实情况。两者的状态区别如图 2-4 所示。

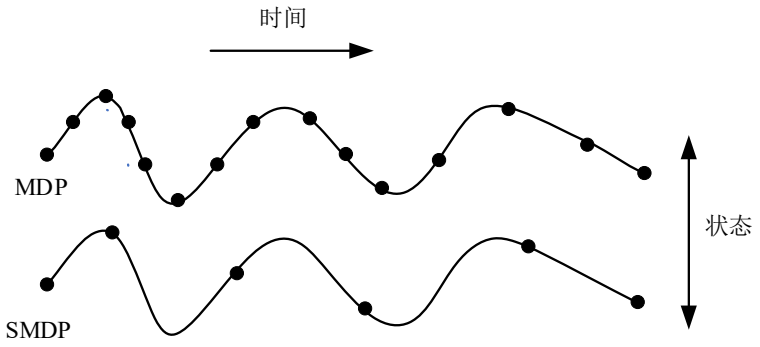


图 2-4 MDP 与 SMDP 状态比较

可以看出 SMDP 是 MDP 的一般化过程，SMDP 是将标准 MDP 中的状态 s 执行动作 a 到达状态 s' 的过程中固定单位时间步长变为可变时间步长。设置时间步长为 N，MDP 的概率转移函数 $P(s,a,s')$ 和期望奖励 $R(s,a,s')$ 可以相应的扩展到 SMDP 中，为 $P(s', N \mid s, a)$ 和 $R(s', N \mid s, a)$，根据贝尔曼方程，确定性的策略 π 下的状态值函数 $V^{\pi}(s)$ 可以定义为执行策略 π 在状态 s 下所得到的奖励和未来预期奖励的加权总和，即：

$$V^{\pi}(s) = \bar{R}(s, \pi(s)) + \sum_{s', N} P(s', N \mid s, \pi(s)) \gamma^{N} V^{\pi}(s') \tag{2-3}$$

其中，$\bar{R}(s, \pi(s))$ 表示状态 s 下执行策略 $\pi(s)$ 后的平均即时奖励，该奖励与转移概率

P 和奖励函数 R 密切相关。

SMDP 的这种特性使其成为模拟和解决那些涉及延迟效应或长期规划决策的理想工具。例如，在智能空战决策的影响可能不会立即显现，而是随着时间的推移逐渐展现。通过引入动作的持续时间和复杂的时间依赖性，SMDP 提供了一个更为细腻和适应性强的框架，用于捕捉和优化这些类型的顺序决策问题。本论文的顶层策略采用了半马尔可夫决策的思想进行设计与训练。

2.2.2 基于价值的学习

在价值学习中通常需要关注动作价值函数 $Q_\pi(s, a)$ 和最优动作价值函数 $Q_*(s, a)$[171]。

定义 U_t 为从 t 时刻开始计算的所有奖励累计和，即：

$$U_t = \sum_{x=t}^{H} \gamma^x r_x \tag{2-4}$$

动作价值函数定义为：

$$Q_\pi\left(s_t, a_t\right) = E\left[U_t \middle| S_t = s_t, A_t = a_t\right] \tag{2-5}$$

最优行为价值函数定义如下：

$$Q_*\left(s_t, a_t\right) = \max_\pi Q_\pi\left(s_t, a_t\right) \tag{2-6}$$

最优动作价值函数代表在最优策略下的预期奖励值，即：

$$\pi^* = \arg\max_\pi Q_\pi\left(s_t, a_t\right) \tag{2-7}$$

通过为当前状态 s_t 下的所有可能动作打分，也就是估算每个动作的期望总奖励，可以确定最佳动作选择[172]。

一般而言，值函数的更新通过时间差分方法（Temporal Difference, TD）实现：

$$Q_*\left(s_t, a_t\right) = E_{S_{t+1}\sim T(s_t, a_t)}\left[R_t + \gamma \cdot \max_{A\in\mathcal{A}} Q_*\left(S_{t+1}, A\right) \middle| S_t = s_t, A_t = a_t\right] \tag{2-8}$$

将式(2-8)通过蒙特卡洛方法进行近似处理，可得：

$$Q_*\left(s_t, a_t\right) \approx r_t + \gamma \cdot \max_{a\in\mathcal{A}} Q_*\left(s_{t+1}, a\right) \tag{2-9}$$

通过使用神经网络 $q\left(s_t, a_t; \theta\right)$ 来逼近最优动作价值函数，TD 误差的表达式为：

$$\delta_t = q\left(s_t, a_t; \theta\right) - \left(r_t + \gamma \cdot \max_{a\in\mathcal{A}} q\left(s_{t+1}, a; \theta\right)\right) \tag{2-10}$$

其中，θ 表示神经网络的参数。设定目标函数为 TD 误差的均方值：

$$J(\omega) = \frac{1}{2}\delta_t^2 \tag{2-11}$$

最后，使用梯度下降策略对神经网络的参数 $q\left(s,a;\theta\right)$ 进行更新。

2.2.3 基于策略的学习

不同于价值学习，策略学习的目标是直接解决优化问题，以获得最优策略函数或其

近似，例如通过策略网络，并采用策略梯度方法更新模型[173]。对于公式(2-5)其状态价值函数为：

$$V_\pi(s_t)=E_{A_t\sim\pi(\cdot|s_t;\omega)}\left[Q_\pi(s_t,A_t)\right] \tag{2-12}$$

其中，ω代表策略网络的参数，$V_\pi(s_t)$估当前状态s_t的效益，$V_\pi(s_t)$的值越高，意味着累积奖励U_t越大，从而策略π更优，表明参数θ更理想，进而$V_\pi(s_t)$的值也将增大。

因此，将状态价值函数$V_\pi(S)$的期望定义为优化目标函数：

$$J(\omega)=E_S\left[V_\pi(S)\right] \tag{2-13}$$

从而，策略学习的目标如下式所示：

$$\max_\theta J(\omega) \tag{2-14}$$

策略梯度的计算可以表示为：

$$\frac{\partial J(\omega)}{\partial \omega}=E_S\left[E_{A\sim\pi(\cdot|\ S;\omega)}\left[\frac{\partial \ln\pi(A|\ S;\omega)}{\partial\omega}\cdot Q_\pi(S,A)\right]\right] \tag{2-15}$$

最后，采用梯度上升法更新策略网络的参数。

2.2.4 Actor-Critic 算法

“演员-评论家”算法（Actor-Critic, AC）[174]采用了两种神经网络——一种负责动作生成（Actor）和另一种负责评价动作价值（Critic）。此方法通过不断改善这两个网络的性能，以增强决策的准确性并优化学习效率。该方法的基本学习过程展示如图 2-5 所示。

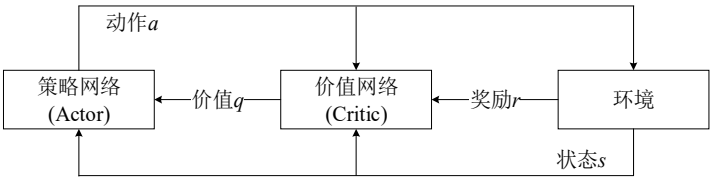


图 2-5 Actor-Critic 算法基本流程

(1) 价值网络更新

价值网络更新过程采用基于 SARSA 算法的贝尔曼方程，具体表述见式(2-16)。

$$Q_\pi(S_t,A_t)=E_{S_{t+1},A_{t+1}}\left[R_t+\gamma\cdot Q_\pi(S_{t+1},A_{t+1})|\ S_t=s_t,A_t=at\right] \tag{2-16}$$

在时间点t，网络预测的动作价值为$q(s_t,a_t;\theta)$，而在时间点$t+1$，观测到的实际值为r、s_{t+1}、a_{t+1}，此时网络预测的动作价值为$q(s_{t+1},a_{t+1};\theta)$。通过这些观测值，我们可以计算出 TD 误差：

$$\delta_t=q(s_t,a_t;\theta)-\left(r_t+\gamma\cdot q(s_{t+1},a_{t+1};\theta)\right) \tag{2-17}$$

因此，定义的价值网络的目标更新函数如式(2-18)所示，并通过梯度下降法调整网络参数，如式(2-19)所示。

$$J(\theta)=\frac{1}{2}\delta_t^2 \tag{2-18}$$

$$\theta\leftarrow\theta-\alpha\cdot\delta_t\cdot\nabla_\omega q(s_t,a_t;\theta) \tag{2-19}$$

其中，α代表学习率。

(2) 策略网络更新

策略网络$\pi(a|\ s;\omega)$的更新则是基于策略梯度法，如式(2-15)所示。在给定时间点t，策略网络根据当前状态s_t来选择动作a_t，并利用价值函数网络的评估结果$q(s,a;\theta)$作为式(2-15)中$Q_\pi(S,A)$的估计，从而得到策略梯度的近似表达，见式(2-20)。策略网络参数的更新遵循梯度上升法，表达式见(2-21)。

$$\mathbf{g}(s_t,a_t;\omega)=\nabla_\theta\ln\pi(a_t|\ s_t;\omega)\cdot q(s_t,a_t;\theta) \tag{2-20}$$

$$\omega\leftarrow\omega+\beta\cdot g(s_t,a_t;\omega) \tag{2-21}$$

其中，β表示策略网络的学习率。

2.2.5 分层强化学习

分层强化学习，作为强化学习领域的一个重要子领域，旨在通过与环境的互动和试错过程来逐步改进决策策略。然而，在传统强化学习中，一个显著的问题是维度灾难的现象，即随着系统状态维度的提升，所需训练的参数数量也会以指数形式增加，导致计算和存储资源的大量消耗。

为应对这一挑战，分层强化学习策略被提出，其基本思路是将一个复杂的问题划分为多个更易管理的子问题，具体分解策略主要有两种：一是将一个复杂的任务分割为多个简单的子任务，并分别对它们进行求解；二是在解决一个子问题的基础上，将得到的策略作为解决下一个子问题时策略学习的一部分。

分层强化学习的核心理念在于通过设计算法结构来对策略和价值函数实施约束，或采用能够自然引入这些约束的算法。这一领域的常见方法主要包括三种：基于选项、基于分层局部策略以及子任务。下面将对这三类方法进行简要的介绍。

(1) 基于选项的分层强化学习

选项[175]框架是强化学习中一种用于分层决策的方法，它由三个基本元素组成的元组构成：$<I,\pi,\beta>$。此处，$\mathrm{I}\subseteq\mathrm{S}$表示选项的初始状态集合；$\pi:S\times A\rightarrow[0,1]$表示策略，即一个涵盖状态空间与动作空间的概率密度函数；$\beta:S\rightarrow[0,1]$是终止条件，用于确定何时应停止执行当前选项。

在基于选项的学习过程中，智能体从一个初始状态开始，选择并执行一个选项。根据该选项的策略π，智能体选择并执行动作或者转向另一个选项，这一过程持续进行，直到满足某一选项的终止条件。在每个状态s，智能体可以选择的选项由集合$O(s)$表示。

选项框架中的 Q-learning 更新公式是强化学习的关键组成部分，它帮助智能体学习

在给定状态下选择最优选项或动作的策略，以最大化预期奖励。该方程考虑了当前状态、选取的动作或选项、结果状态以及由此产生的奖励，通过迭代更新 Q 值，逐渐逼近最优策略。基于选项的 Q-learning 更新方程如式所示：

$$Q(s,o) \leftarrow Q(s,o) + \alpha \left[r + \gamma^k \max_{o' \in O_{s'}} Q\left(s', o'\right) \quad Q(s,o) \right] \tag{2-22}$$

(2) 基于分层局部策略的学习

通过应用分层抽象机制（Hierarchical Abstract Machines, HAMs）[176]的学习策略，可以为特定场景下的策略选择提供一种结构化的框架。HAMs 构建于一系列限定的抽象机集合上，记作 H，其中每个元素 $h \subseteq H$ 代表一个独立的 HAM 抽象机。这些实体由三个关键组成部分构成：抽象机的状态、状态之间的转换逻辑以及初始状态的设置。

具体来说，在 HAM 框架中，状态会触发预定动作的实施；调用状态允许系统转向另一 HAM 进行处理；选择状态提供了一个决策点，从而非确定性地选择下一步行动；终止状态标志着当前 HAM 的结束。对于 HAM Q-learning 更新方程，它扩展了传统 Q-learning 算法，以适应 HAMs 的结构。HAM Q-learning 的更新方程如式所示：

$$Q\left(\left[s_c, m_c\right], a\right) \leftarrow Q\left(\left[s_c, m_c\right], a\right) + \alpha \left[r_c + \beta_c V([t,n]) - Q\left(\left[s_c, m_c\right], a\right) \right] \tag{2-23}$$

在这种扩展中，Q 值 $\mathrm{Q}([\mathrm{s},\mathrm{m}],\mathrm{a})$ 考虑了环境状态 s 、抽象机状态 m ，以及在选择 m 时采取的动作 a 。下标 c 用于标识选择点，这反映了分层决策过程中的一个特定决策点。

(3) 基于子任务的学习

基于子任务的学习方法通过 MAXQ(MAXQ value function decomposition)价值函数分解来实现，是由 Dietterich[177]在 2000 年引入的一种技术。这种方法的核心在于将整个马可夫决策过程（M）分解为若干个较小、更易管理的子任务，以此方式简化了整个决策过程的复杂性。通过独立解决每一个子任务，最终能够解决原始的、更为复杂的问题。

在基于子任务的学习框架中，每个子任务被定义有其特定的终止状态 T 和子动作空间 A。子任务的主要目标是，从给定的初始状态出发，通过连续执行一系列动作，驱动系统状态的转换，直到达到预定的终止状态。这种方法不仅有助于明确子任务间的关系和层次结构，也使得问题的解决过程更加模块化，每个子任务的成功解决可以为解决更大规模问题提供了基石。

2.3 本章小结

本章主要介绍了无人机编队协同智能空战战术决策问题的相关技术。首先，介绍了无人机的数学模型；然后，阐述了强化学习算法的基本原理，包括马尔可夫决策和半马尔可夫决策过程、基于价值学习和基于策略学习的方法和 Actor-Critic 方法；最后，本章探讨了分层强化学习算法的核心思想，为后续章节中的算法应用奠定了坚实的理论基础。

第 3 章 基于改进灰狼算法的无人机协同航路规划模型

无人机编队的航路规划是完成与敌机对抗任务的初始步骤，无人机编队在执行战术决策时，需要根据预先规划好的航线飞抵作战空域。本章以灰狼算法作为研究理论工具，研究了在具有大量威胁区的环境下无人机编队航路规划的方法。首先，介绍了多无人机航路规划的模型；其次，对灰狼算法进行改进，提出了 NI-GWO 算法；最终，通过设计一系列仿真测试来验证该算法的有效性。

3.1 问题分析

在执行自主机动决策之前，无人机编队必须从起始点起飞并飞行至作战区域。此过程中，它们必须绕开敌方的威胁区并制定飞行计划以抵达指定目的地。这一飞行计划的制定，是在不需要操作人员干预的情况下，根据周围环境和潜在威胁，指引无人机编队成功到达任务目标点的过程。无人机编队航路规划是无人机进入作战区域前的关键一环。

针对无人机编队航路规划问题，本文将无人机在进入作战区域前需要规避的敌方威胁区简化为火控威胁区和雷达威胁区，将无人机编队从起始点到目标点的航路规划作为求解目标。本章通过采用改进的灰狼优化算法实现了多无人机协同航路规划的作战任务，本章具体研究内容如图 3-1 所示。

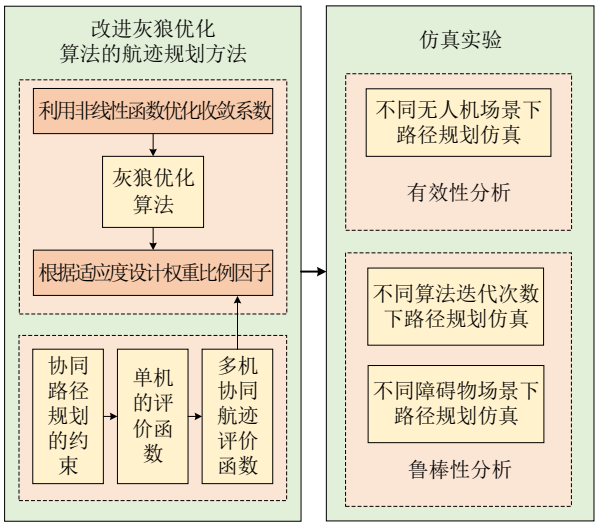


图 3-1 基于改进灰狼算法的无人机协同航路规划模型内容框图

3.2 多机协同航路规划模型

设 $V=\{V_i,i=1,2,\cdots,N_v\}$ 为执行任务的多无人机集合，$T=\{T_i,i=1,2,\cdots,N_v\}$ 为各无人机单元相应目标构成的集合，$M=\{M_j,j=1,2,\cdots,N_M\}$ 表示敌机威胁集合，任务区 E 设定为 N_r 行 N_c 列的栅格化环境。对单架无人机而言，从起点到终点的路径是由多个航迹点构成的序列，通过顺序连接起始点和目标点之间的航点，可以形成无人机的具体飞行

路径。假定无人机V_i的最大飞行距离为$L_{\max}^{(i)}$，最大转向角度为$\theta_{\max}^{(i)}$，无人机速度范围定在$\left[v_{\min}^{(i)},v_{\max}^{(i)}\right]$，航路规划任务起点为$S_i$，航路规划终点为$G_i$，飞行路径中的节点标记为$P_{i-k}(k=1,2,\cdots,n-1)$，因此无人机$V_i$的飞行航迹可以用$S_i \overset{\Gamma(q)}{\rightarrow} P_{i-1}\cdots P_{i-(n-1)} \overset{\Gamma(q)}{\rightarrow} G_i$来表示，其中$\Gamma(q)$代表航路规划中的约束条件，$q$表示航迹的约束参数。

3.2.1 无人机协同航路规划约束分析

无人机协同航迹规划的约束条件确保每架无人机在遵循各自约束的同时，整个编队能够有效地执行任务。

(1) 无人机航路规划空间协同约束

空间协同约束即避免碰撞的约束[178]，确保多架无人作战飞机间的距离不低于规定的最小安全飞行间隔，即：

$$d_{i-j} \geqslant d_s \tag{3-1}$$

式中，d_{i-j}表示飞行过程中第i个无人机与第j个无人机的航迹点之间的最小距离，d_s表示多无人机之间最小安全飞行距离。

(2) 无人机航路规划时间协同约束

根据文献[179]和[180]，时间协同约束的定义和求解过程涉及较高的计算复杂度，这对于实现实时的协同航迹规划构成了障碍。因此，本文为多无人机到达目标点设定了预定时间，并计算它们到达终点的时间窗口，随后通过比较这两个时间参数来设定时耗成本函数。

将多无人机到达目标点的预定时间定为t_c，利用此时间指令来确保无人机群体在时间维度上的协调一致性；基于无人机的速度限制和飞行路径长度，可以计算出它们实际到达目的地的时间区间$\left[t_{\min^i},t_{\max^i}\right]$，则：

$$t_{\min^i}=\frac{L_i}{v_{\max^i}},t_{\max^i}=\frac{L_i}{v_{\min^i}} \tag{3-2}$$

其中，$v_{\max^i}$与$v_{\min^i}$分别为多无人机的最大航行速度和最小航行速度；L_i为第i个多无人机的航迹长度。

若$t_{\min^i} \le t_c \le t_{\max^i}$，则第$i$个多无人机能按照规定的时间完成任务；否则不能完成任务。

3.2.2 单无人机航迹评价函数

无人机的航迹成本函数通常用作航迹评估的标准，主要涵盖了燃料成本和威胁成本[181-183]。单架无人机的航迹成本可以通过以下公式进行计算：

$$J=\sum_{i=1}^{n}\left(w_1 f_O\left(l_i\right)+w_2 f_{T-i}\right) \tag{3-3}$$

其中，l_i表示第i段无人机航迹长度，f_O表示燃料消耗成本函数，f_{T-i}为第i段航迹中的风险成本，其主要取决于威胁区域的位置；w_1和w_2为相应的权系数，且$w_1+w_2=1$。

3.2.3 多无人机机协同航迹评价函数

多机协同航迹评价函数基于单无人机评价标准，进一步整合了多无人机在时间和空间上的协作约束。本文研究重点放在时间成本和空间成本上。多无人机协同航迹的评价函数公式如下式所定义：

$$J=\sum_{i=1}^{m}\left(w_1 f_{o-i}+w_2 f_{T-i}+w_3 f_{m-i}+w_4 f_{C-i}\right) \tag{3-4}$$

其中，m为多无人作战飞机的数量；f_{o-i}、f_{T-i}、f_{m-i}和f_{C-i}表示第i个无人机的燃料消耗成本、风险成本、时耗成本和碰撞成本；w_i为上述各个成本的权重系数，其中$w_1+w_2+w_3+w_4=1$。通过 3.2.1 节中无人机的时间协同约束分析，多无人机协同的时耗成本的表示如式所示：

$$f_{m-i}=\begin{cases}0 & ,\ t_{\min^i}\leqslant t_c\leqslant t_{\max^i}\\ \left|t_i-t_c\right| & ,\ t_{\min^i}>t_c\ \text{or}\ t_{\max^i}<t_c\end{cases} \tag{3-5}$$

其中，t_i为第i个无人机到达目标点的时间。

在计算多无人机间碰撞成本时，以无人机i为例，首步是确定该无人机在每个航点的时间位置及与先前$i-1$架无人机的间距。若任两架无人机间的距离低于预定的最小安全飞行间隔，则视为一次碰撞事件。通过对航迹上所有航点的碰撞事件进行累加，得出总碰撞次数的计算方法如下：

$$f_{c-i}=\begin{cases}1 & ,\ d_{i-j}<d_s\\ 0 & ,\ d_{i-j}\geqslant d_s\end{cases} \tag{3-6}$$

其中，$j\leqslant i-1$，指第i个多无人机与前$i-1$比较。

3.3 基于改进灰狼优化算法的多无人机航路规划模型

3.3.1 灰狼优化算法

灰狼优化算法(Grey Wolf Optimizer, GWO) 是群体智能优化策略中的一个重要方法，其灵感来源于灰狼的社会组织结构与集体狩猎行为。该算法借鉴了灰狼分级制度及其群体狩猎技巧来寻求问题的最佳解决方案。在算法实现中，模拟了狼群内部的等级分布，确保了狼群成员能够根据自然界的等级制度协作，以有效追求优化目标。如图 3-2 所示。

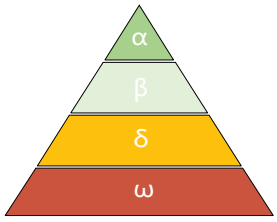


图 3-2 灰狼种群等级制度

在灰狼的社会结构顶层位于部落的最高领导地位的是 Alpha 狼，承担着狼群管理及对狩猎活动的整体策划与决策责任。紧随其后的第二级别是 Beta 狼，作为狼群中的副领袖，其主要职责是辅助 Alpha 狼进行决策和策划，在狼群的权力结构中排名第二。第三级别的 Delta 狼，也称作侦察兵，负责执行侦查、警戒和看护等任务。社会等级中最底层的是 Omega 狼，虽然地位最低，但对于维护狼群内部和谐关系发挥着至关重要的作用。在灰狼优化算法中，这三个上层等级相当于最佳、次佳和第三佳的解，它们各自向 Omega 狼展示了如何向目标进行探索。在优化的过程中，狼群将持续调整这些 Alpha、Beta、Delta 和 Omega 狼的位置，目的是实现最优化目标。

在狼群进行捕食的过程中，该活动分为四个基本步骤：包围、狩猎、攻击以及搜寻。捕食行为开始于追踪并精确定位猎物，随后，在 Alpha、Beta、Delta 狼这三位领导者的引导下，其余狼群成员会以这些领导狼的位置为参照点，不断调整自己的位置。这一策略使得狼群能够逐步缩小与猎物之间的距离，直到最终实施捕捉。该捕食策略的详细过程如图 3-3 所示。

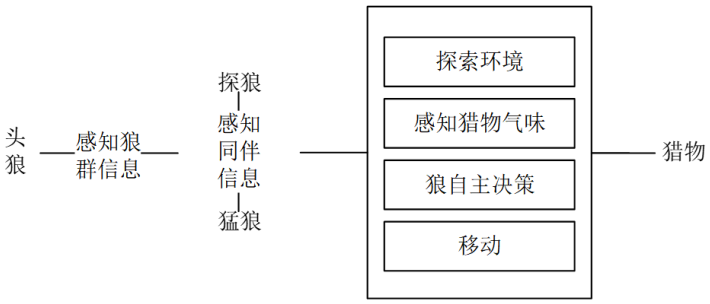


图 3-3 狼群捕猎模型

(1) 包围

在进行狩猎时，灰狼展现出了一种协同的战术行为，即通过集体行动包围其猎物。为了定量地模拟灰狼的包围行为，其包围可以表示为以下方程：

$$\vec{D}=\left|\vec{C}\cdot\vec{X}_p(i)-\vec{X}(i)\right| \tag{3-7}$$

$$\vec{X}(i+1)=\vec{X}_p(t)-\vec{A}\cdot\vec{D} \tag{3-8}$$

其中，i 代表当前的迭代次数，$\vec{X}_p$ 为狼群相对猎物的位置向量，$\vec{X}$ 为狼群的位置向量。$\vec{A}$和$\vec{C}$为系数向量，其具体计算方式如下：

$$\vec{A}=2\vec{a}\cdot\vec{r}_1-\vec{a} \tag{3-9}$$

$$\vec{C}=2\cdot\vec{r}_2 \tag{3-10}$$

在该迭代过程中，收敛因子$\vec{a}$从 2 开始逐渐减少至 0，采取线性递减的方式，而r_1和r_2分别是在[0,1]区间内生成的随机向量。更新参数a的方程如下：

$$\vec{a}=2-i\left(\frac{2}{I}\right) \tag{3-11}$$

其中，i 代表最新的迭代次数，I 是迭代总数。

在灰狼算法中，二维位置向量和多个目标的潜在位置如图 3-4 所示。灰狼可以根据猎物的位置 $X_{(i)}$ 更新对应狼群位置。由于随机向量$\vec{r_1}$和$\vec{r_2}$，狼群可以到达如图 3-4 所示中各点之间的所有位置。因此，狼群可以使用方程式(3-7)和(3-8)在猎物周围区域的任何位置更新对应狼群位置。

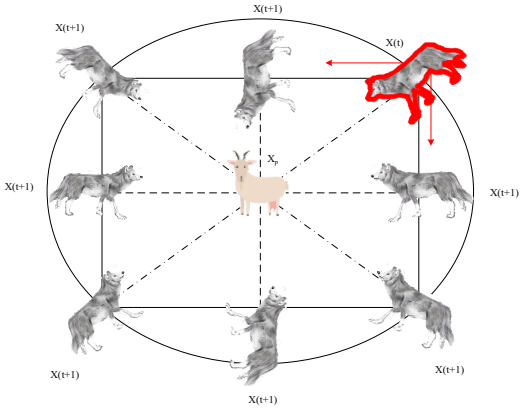


图 3-4 二维位置向量和潜在的未来位置

(2) 狩猎

在野外定位并追踪其猎物时灰狼展现出了卓越的能力，通常这一行为由群体中的领导者 Alpha 狼来指挥。然而，在某些情况下，狩猎的领导权可能会落在 Beta 或 Delta 狼身上。在自然界的复杂环境中，最佳猎物的具体位置往往是未知的。为了数学化地模拟灰狼的狩猎策略，本模型假定 Alpha、Beta 和 Delta 狼拥有相对更多的信息关于猎物可能的所在位置。这种假设允许模型更加精确地描绘狼群如何在不确定的环境中通过其社会结构的优势来优化狩猎行为。因此，总是使用前三名狼的搜索结果。所有其他搜索智能体，包括 Omega 狼，都被指示重新排列。在狩猎中，灰狼算法的公式如下所示：

$$\begin{cases}\vec{D}_{\alpha}=\left|\vec{C}_1\cdot\vec{X}_{\alpha}-\vec{X}\right| \\ \vec{D}_{\beta}=\left|\vec{C}_2\cdot\vec{X}_{\beta}-\vec{X}\right| \\ \vec{D}_{\delta}=\left|\vec{C}_3\cdot\vec{X}_{\delta}-\vec{X}\right|\end{cases} \tag{3-12}$$

$$\begin{cases}\vec{X}_1=\vec{X}_{\alpha}-\vec{A}_1\cdot\left(\vec{D}_{\alpha}\right) \\ \vec{X}_2=\vec{X}_{\beta}-\vec{A}_2\cdot\left(\vec{D}_{\beta}\right) \\ \vec{X}_3=\vec{X}_{\delta}-\vec{A}_3\cdot\left(\vec{D}_{\delta}\right)\end{cases} \tag{3-13}$$

$$\vec{X}(i+1)=\frac{\vec{X}_1+\vec{X}_2+\vec{X}_3}{3} \tag{3-14}$$

在二维搜索空间中，智能搜索智能体的位置更新机制受到 Alpha、Beta 和 Delta 狼位置的直接影响，如图 3-5 所示。这些领导狼确定了猎物的大致位置，而其他狼则根据这些关键参考点的位置，通过随机过程调整自己的位置。具体来说，智能体的最终定位是一个随机过程，该过程以 Alpha、Beta 和 Delta 狼的当前位置为基础来计算。这意味着，尽管猎物的确切位置由领导狼界定，但其他狼在其周围的具体位置更新则是随机发生的，从而模拟了灰狼狩猎时的动态调整和适应性行为。

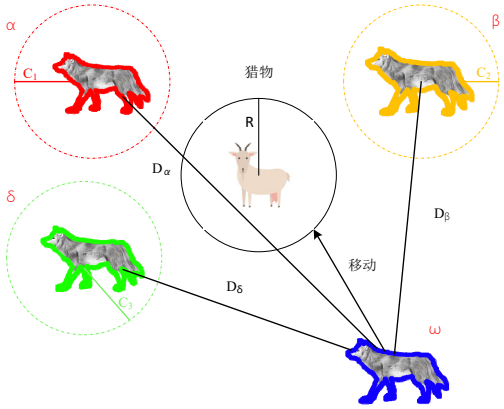


图 3-5 灰狼算法中的位置更新

(3) 攻击

灰狼发起攻击并结束狩猎的时机是在猎物停止移动时。为了在数学模型中模拟这一行为，对向猎物靠近的策略进行了调整，通过降低参数 a 的值实现。这个调整不仅减少了搜索智能体向猎物靠近的步长，同时也缩小了参数 $\vec{A}$ 的波动范围。这样的变化模拟了灰狼在最终攻击阶段减少移动幅度，提高攻击精准度的自然行为，从而更加精确地捕捉到猎物的位置并成功完成狩猎。特别是，参数 a 在迭代过程中从 2 降到 0，并且在 $[-2a,2a]$ 范围内随机地递减。当 $\vec{A}$ 的随机值落在 $[-1,1]$ 区间内时，灰狼的未来位置具备覆盖从其当前所在处至猎物所在处之间所有潜在位置的能力。当达到 $|A|<1$ 时，狼群将朝向猎物位置发起攻击，如图 3-6 所示。

在灰狼优化算法框架下，灰狼根据领头者 Alpha、Beta 和 Delta 狼的定位信息，灵活地调整自己的位置以达成最优化。因此，它们可以利用建议的运算符朝向猎物展开进攻。但在应用这些操作的过程中，灰狼优化算法往往容易遭遇陷入到局部最优的问题。

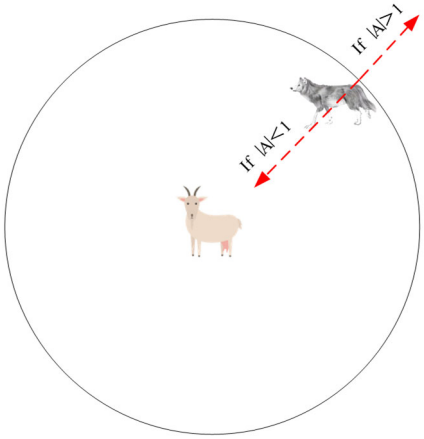


图 3-6 捕猎猎物和搜索猎物

(4) 搜寻

为了有效地指导搜索过程，灰狼算法依赖于 Alpha、Beta 和 Delta 狼的位置信息，这些位置代表了狼群中信息共享的层次结构。这些领导狼独立地搜寻猎物，并基于各自的搜索结果指导整个狼群重新聚焦并协同攻击猎物。为模拟这种探索行为的扩散特征，灰狼算法采用了一种方法，通过赋予参数随机值，使其大于 1 或小于-1，以促使搜索狼群向远离猎物的方向移动。这种方法模拟了灰狼在探索环境和搜寻猎物时的行动，允许算法在广阔的搜索空间内进行有效探索，避免过早聚焦于局部最优解，从而增强了算法的全局搜索能力和适应性。如图 3-6 所示展示了当$|A|>1$时，灰狼如何从它们的猎物偏离以寻找更适合的猎物。

GWO 的另一个值得探究的方面是$\vec{C}$向量，它具有$[0,2]$的随机值。此部分通过分配随机权重给猎物，实现了对猎物影响距离作用的随机增强$(C>1)$或削弱$(C<1)$。因此，GWO 可能在优化中表现得更随机，同时避免局部最优陷阱。与 A 不同，C 的下降不是线性的。

最后，通过 GWO 算法产生一个随机的灰狼种群。Alpha、Beta 和 Delta 狼通过迭代计算猎物的潜在位置，每个可能的响应都改变了它与猎物之间的距离，a 的值从 2 降到 0。当满足结束标准时，GWO 算法完成整个流程。

3.3.2 改进灰狼优化算法

GWO 算法利用位置更新方程来找到最优解，但其存在以下缺陷：GWO 算法将狼分为四个等级，这些狼基于式(3-14)调整它们的位置以捕捉猎物，主要依靠 Alpha 狼、Beta 狼和 Delta 狼的信息来进行搜索。在解决多模态问题时，可能会导致算法停滞于局部最优解处[184]。在迭代过程中，因为 Delta 狼的解较差，在 Alpha 狼根据三只狼的信息同时调整其位置，可能引导 Alpha 狼走向错误的方向，同样，Beta 狼也可能获得错误的引导。因此，式(3-14)给予 Alpha 狼、Beta 狼和 Delta 狼同样的权重 1/3，这是不科学的[185]。

混合方法可以有效地结合算法的优势，同时摒弃算法劣势。大量研究表明，与单一算法相比，混合算法在解决各种优化问题时表现更为出色[186][187]。这种理念已广泛应用于传统的进化算法中，但在群体智能算法中，尤其是 GWO 算法领域，其应用相对较少。为了填补这一研究空白并扩展算法的应用范围，本文首先加入非线性控制机制来优化收敛因子，然后根据狼在层次结构中的不同社会地位，对式(3-14)的位置更新方程进行了改进，以提高算法的探索能力，使其跳出局部最优，即利用非线性因子改进灰狼优化算法(Nonlinear Improved Grey Wolf Optimizer Algorithm，NI-GWO)。具体的改进方式如下：

(1) 改进的适应度函数模型

在多无人机协同航路规划任务中，综合航迹评价函数是 NI-GWO 算法中灰狼适应度函数，基于无人机协同航路评价函数、战场环境设置以及协同任务的需求，根据 3.2 节多无人机协同航路规划模型及各个变量之间的相互作用，可对适应度值函数 $f\left(\vec{X}\right)$ 进行优化，改进方法如(3-15)所示，同时该式也是多无人机协同航路规划的目标函数。

$$f\left(\vec{X}\right)=\min\sum_{i=1}^{m}\left(w_1 f_{o-i}(l)+w_2 f_{T-i}(p)+w_3 f_{m-i}(t)+w_4 f_{C-i}(X)\right) \tag{3-15}$$

其中，f_{o-i}、f_{T-i}、f_{m-i} 和 f_{C-i} 分别表示第 i 个无人机的燃料消耗成本、风险成本、时耗成本和碰撞成本；w_i 为各个成本的权重，其中 $w_1+w_2+w_3+w_4=1$。

(2) 非线性优化算法的收敛因子机制

随着智能算法的发展，灰狼算法被广泛应用于解决无人机的航路规划问题。但常用灰狼算法的建模方式是收敛因子取值方式较为随机，并且缺少对无人机编队整体适应度对灰狼位置更新的影响，无法达到算法在航路规划时的高效性和鲁棒性，于是对收敛因子的取值方式进行改进。文献[188]介绍了一种非线性控制收敛因子，由实验证明该 s 型函数变化的收敛因子可以提升算法的搜索能力，于是本章设计了非线性控制机制来优化收敛因子的取值，然后通过引入适应度值的权重来优化位置更新方式，并根据狼的适应度值对狼群算法中的位置更新方法进行改进。

由于其强大的搜索能力，群体智能优化算法被广泛用于不同类型的离散优化问题。因为全局搜索能力和局部搜索能力之间的不平衡，群体智能算法通常会导致收敛精度和优化性能的下降。在 GWO 算法中，参数 A 是算法寻找最优解的扰动因子，GWO 将根据变换值 A 决定算法的开发和探索能力。从式(3-9)和式(3-11)中可知，参数 $\vec{A}$ 的转换值将由收敛因子 $\vec{a}$ 决定，随着迭代次数的增加，$\vec{a}$ 的值将从 2 线性减少至 0。

对于 GWO 算法而言，迭代开始时较大的 $\vec{a}$ 值提高了其对局部区域的搜索能力，并加速了算法的收敛过程。在迭代即将结束的阶段，较低的 $\vec{a}$ 值将促进狼群缩短其搜索范围，增强其在特定区域的搜寻效果，并促进算法更迅速地达到收敛。鉴于搜索策略不总是线性变化，收敛因子的线性降低并不能充分反映算法在应对多峰环境下的复杂搜索动

态。因此，本文为了提升搜索能力，在 GWO 中引入了以下非线性控制机制：

$$\vec{a}=\vec{a}_{\text{inital}}\bullet\frac{1}{1+e^{\frac{10i-5I}{\text{I}}}} \tag{3-16}$$

其中，$\vec{a}_{\text{inital}}$ 是收敛因子 $\vec{a}$ 的初始值，i 表示当前迭代次数，I 表示最大迭代次数。其变化曲线图如图 3-7 所示。

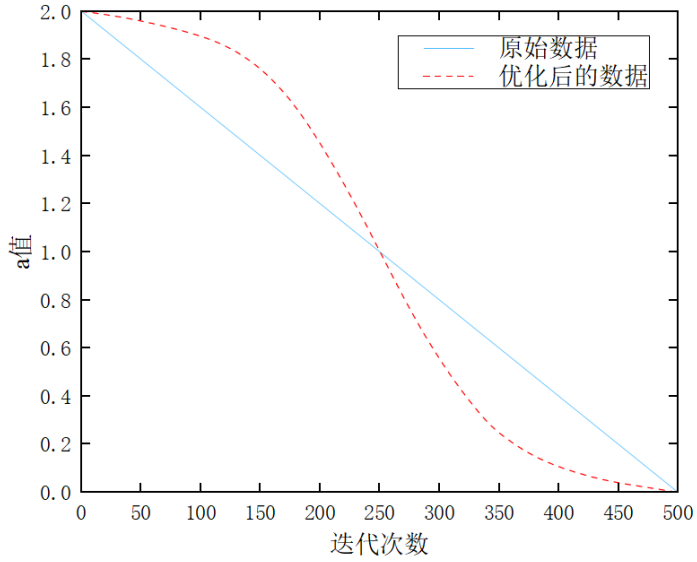


图 3-7 收敛因子变化曲线

如图 3-7 所示可以看出，通过 Sigmoid 函数调整的变化轨迹在迭代初期逐步降低，可以让收敛因子 $\vec{a}$ 能够在较长的周期内维持一个较高的水平，因此，增强了算法在初期的全局探索效能。由于 $\vec{a}$ 的值迅速下降，扰动因子 A 在迭代后期也可以保持较低的值，这一策略增强了算法在进行局部探索时的准确度，同时也实现了全局搜寻与局部搜寻能力之间的有效均衡。

(3) 改进 GWO 的位置更新方程

在 GWO 中，前三名狼有不同的社会地位。为了更好地利用领头狼的信息，Beta 和 Delta 狼围绕 Alpha 狼进行搜索，Alpha 狼自行探索，根据狼的适应度值计算权重。为了增强算法的开发能力，给最佳狼分配更多的权重，如下所示：

$$\begin{cases}\vec{X}_{\alpha}=\vec{X}_{\alpha}+2\times\vec{a}\times\ \text{rand}\ \times\left(\vec{X}_{r_1}-\vec{X}_{r_2}\right)\\ \vec{X}_{\beta}=\vec{X}_{\beta}+2\times\vec{a}\times\ \text{rand}\ \times\left(\vec{X}_{\alpha}-\vec{X}_{\beta}\right)\\ \vec{X}_{\delta}=\vec{X}_{\delta}+2\times\vec{a}\times\ \text{rand}\ \times\left(\vec{X}_{\alpha}-\vec{X}_{\delta}\right)\end{cases} \tag{3-17}$$

其中，$\vec{X}_{\alpha}$、$\vec{X}_{\beta}$ 和 $\vec{X}_{\delta}$ 分别代表着 Alpha、Beta 和 Delta 狼的位置，$\vec{a}$ 以线性的方式从 2 减少到 0，r_1 和 r_2 是范围在 1 到狼群数量之间的不同整数 $\left(r_1\neq r_2\right)$，$rand\in[0,1]$ 是一个随机数。对于其他的狼，它们根据位置更新公式进行移动。然而，将相同的权重赋予

$\vec{X}_\alpha$、$\vec{X}_\beta$ 和 $\vec{X}_\delta$ 是不公平的。权重取决于它们的适应度值。狼的适应度越好，它拥有的信息就越多，应该给予更多的权重。权重可以基于狼的适应度值来计算，通常，优化问题是为了找到最小化。权重的计算方式如下：

$$\left(w_\alpha, w_\beta, w_\delta\right)=\left(f\left(\vec{X}_\alpha\right)+f\left(\vec{X}_\beta\right)+f\left(\vec{X}_\delta\right)\right)\times\left(\frac{1}{f\left(\vec{X}_\alpha\right)}, \frac{1}{f\left(\vec{X}_\beta\right)}, \frac{1}{f\left(\vec{X}_\delta\right)}\right) \tag{3-18}$$

$$\overrightarrow{X}(t+1)=\frac{w_\alpha \times \vec{X}_1+w_\beta \times \vec{X}_2+w_\delta \times \vec{X}_3}{w_\alpha+w_\beta+w_\delta} \tag{3-19}$$

其中，$\vec{X}_\alpha$、$\vec{X}_\beta$ 和 $\vec{X}_\delta$ 分别是 Alpha、Beta 和 Delta 狼的位置，$f\left(\vec{X}_\alpha\right)$、$f\left(\vec{X}_\beta\right)$ 和 $f\left(\vec{X}_\delta\right)$ 是它们的适应度值，w_α、w_β 和 w_δ 是它们的权重（$w_\alpha+w_\beta+w_\delta=1$），三个向量 $\vec{X}_1$、$\vec{X}_2$ 和 $\vec{X}_3$ 来自式(3-13)，式(3-18)和式(3-19)负责根据它们的适应度计算分配给前三名狼的权重。本次改进通过给最优的狼分配更多的权重(而不是传统 GWO 中的相同权重 1/3)来增强算法的开发利用率。由于 Alpha 狼对猎物位置的了解最多，因此设置最高的权重，Beta 狼排在第二位，Delta 狼的权重最低。

本论文设计的 NI-GWO 算法的具体改进内容如图 3-8 所示。

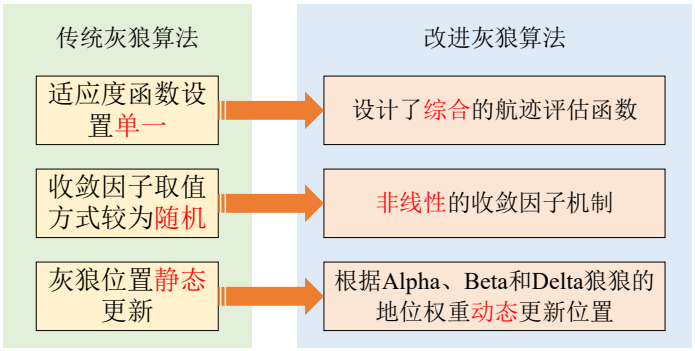


图 3-8 NI-GWO 算法的具体改进内容

利用所提出的 NI-GWO 算法求解第 3.2 节中的模型，具体过程如下：

步骤 1：设置每架无人机的起点和终点。本文实验将在二维场景中展开，起点和目标点坐标都为二维坐标。

步骤 2：建立了一个考虑了目标函数和两个约束条件的多无人机协同航路规划模型。

步骤 3：随机生成一定数量的灰狼，每只灰狼的位置代表一个可能的解，设置最大迭代次数、非线性收敛因子和其他相关参数。

步骤 4：根据多无人机协同航迹规划的综合评价函数，对每个灰狼进行适应度评估，以识别群中的 Alpha、Beta 和 Delta 狼。

步骤 5：根据式(3-15)中提到的 Sigmoid 函数调整收敛因子。收敛因子在迭代开始时较高，以增强全局搜索能力，随着迭代进行逐渐降低，以提高局部搜索精度。

步骤 6：根据式(3-16)和式(3-17)、(3-18)，结合 Alpha、Beta 和 Delta 狼的位置以及它们的适应度值，更新群体中其他灰狼的位置。权重的分配基于各狼的适应度值，以增强算法的开发利用率。

步骤 7：确保所有灰狼的位置都在预定义的搜索空间内。式(3-15)用来评价该模型。

步骤 8：不断重复步骤 3 到步骤 7 过程，直至完成预定的最大迭代轮数。

步骤 9：算法结束时，Alpha 狼的位置被认为是问题的最优解，无人机编队可以根据这个解进行引导。

3.4 仿真结果与分析

3.4.1 实验环境设定

为检验本文算法的性能，选取 GWO 和文献[189]所提出的 MP-GWO 算法作为对比算法。本实验中 GWO，MP-GWO 和 NI-GWO 三种算法均采用相同软件和硬件平台，操作系统为 Windows 10 系统，编程环境为 MATLAB R2020a。地图尺寸为 20×20 km，每张图中不同颜色的线段分别对应于相同环境约束和算法下不同初始位置和目标位置的无人机的航路规划结果。其中航路规划模型仿真参数如表 3-1 所示。

表 3-1 航路规划模型仿真参数表

参数名称	参数大小
狼群数量	80
最大探索步数	145
速度约束	(0.3Ma ~ 0.7Ma)
偏角约束	(-60° ~ 60°)
倾角约束	(-45° ~ 45°)
位置 x 约束	(0 ~ 20km)
位置 y 约束	(0 ~ 20km)
安全距离	500m
权重 (w_1,w_2,w_3,w_4)	$(0.05,0.15,0.70,0.10)$

3.4.2 仿真结果与分析

实验一：算法有效性分析

本次实验分别对两架、三架和四架无人机进行航路规划仿真。通过改变无人机的数量以验证所改进的算法在处理不同数量无人机时的有效性，并展现算法在这三种无人机数量场景中的适用性。

(1) 两架无人机航路规划

本次实验设定了 2 架无人机和 2 个目标点，使用灰狼算法进行多机航路规划时当最

大迭代次数达到 145 时航路规划效果趋于稳定，因此本文设置迭代次数为 145 次。每架无人机从设置好的初始位置通过 GWO、MP-GWO 和 NI-GWO 算法进行航路规划以到达相应的目标点，无人机位置信息和目标信息如表 3-2 所示，同时本次实验还再地图上设置了两个雷达威胁区和七个火炮威胁区如表 3-3 所示。

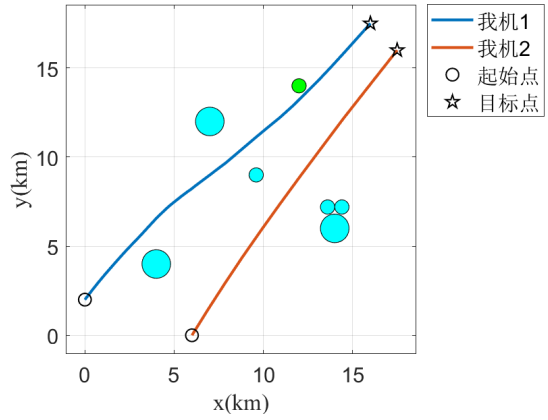
表 3-2 初始位置和目标位置设置表

参数名称	我机 1	我机 2
初始位置	(0 km,2 km)	(6 km,0 km)
目标位置	(16 km,17.5 km)	(17.5 km,16 km)

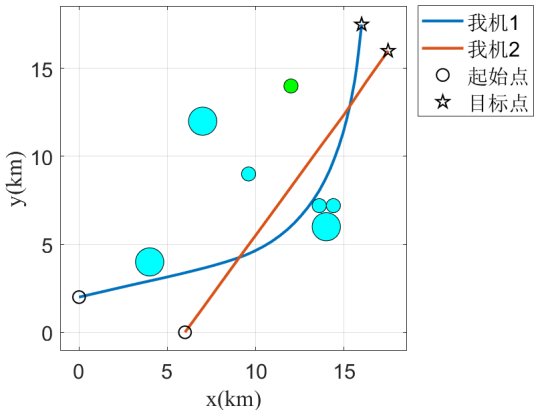
表 3-3 威胁区设置表

威胁区中心	雷达威胁区 1	雷达威胁区 2	火炮威胁区 1	火炮威胁区 2	火炮威胁区 3	火炮威胁区 4	火炮威胁区 5	火炮威胁区 6
横坐标 (km)	4	12	4	7	9.6	14	14.4	13.6
纵坐标 (km)	4	14	4	12	9	6	7.2	7.2
半径 (km)	0.4	0.4	0.8	0.8	0.4	0.8	0.4	0.4

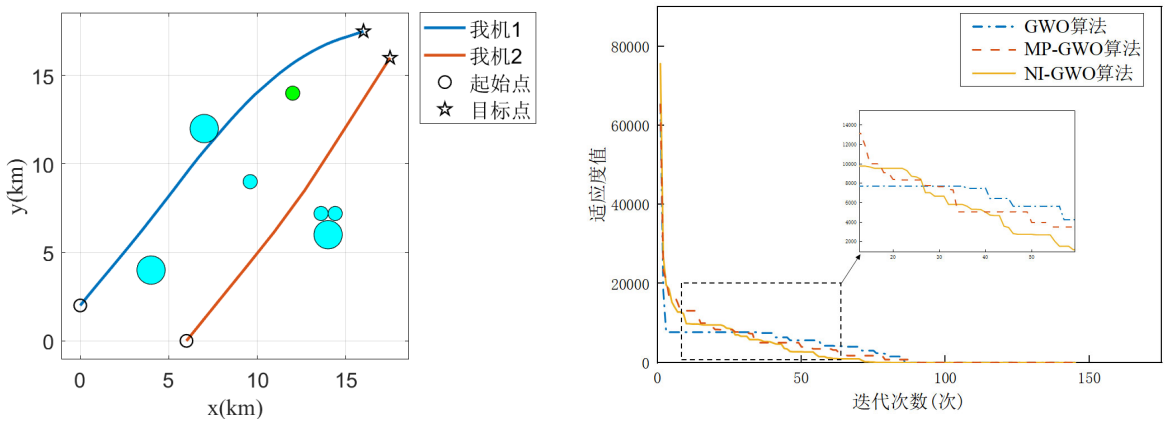
分别利用三种算法对 2 架无人机在起始点到目标点进行航路规划，三种算法航路规划结果对比如图 3-9 所示。



(a) GWO 算法航路规划图



(b) MP-GWO 算法航路规划图



(c) NI-GWO 算法航路规划图　　(d) 适应度值结果图

图 3-9 两架无人机在不同算法下的仿真结果图

如图 3-9 所示，三种算法都顺利完成了对 2 架无人机的路径规划。在我机 2 的航路规划中，三种算法的区别不大，而在我机 1 的航路规划中，三种算法在航行距离上产生了较大的差异。相较于 MP-GWO 算法，本文所提的 NI-GWO 算法路径规划的航行距离更短，且与 GWO 算法路径规划的结果相似。为了进一步分析无人机路径规划效果，重复 500 次仿真实验，对实验过程中的数据进行统计如图 3-10 所示。

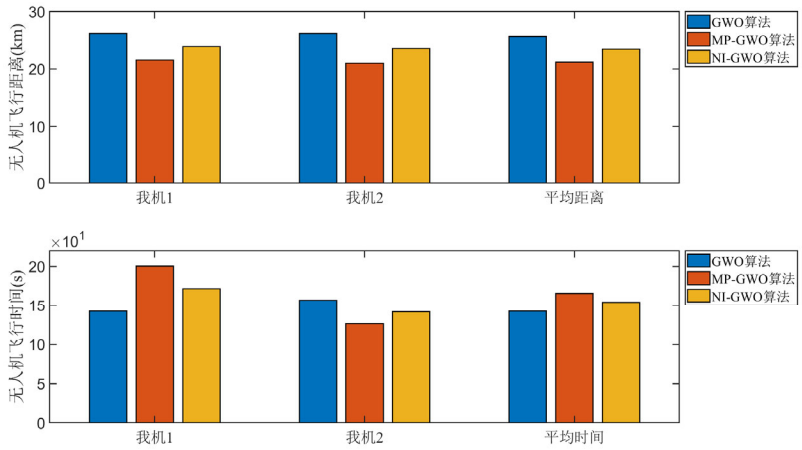


图 3-10 两架无人机的航路规划飞行数据

如图 3-10 所示，三种算法在规划两架无人机飞行过程中航行距离和航行时间相差不大，且三种算法的适应度结果接近。这表明三种算法在无人机编队路径规划过程中在无人机数量较少时都具有良好的性能。

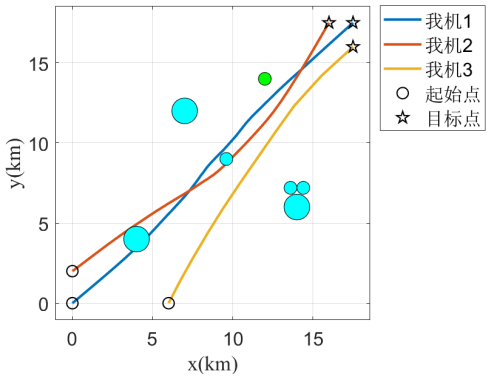
(2) 三架无人机航路规划

在验证算法对 2 架无人机飞行路径规划的有效性后，本小节设置无人机飞行路径规划中无人机数量为 3 架。在本次仿真中，设置算法迭代次数与之前相同（145 次），飞行地图的中的威胁区信息参数不变。无人机位置信息和目标信息如表 3-4 所示。

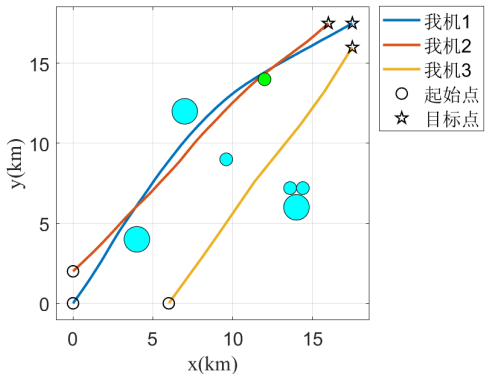
表 3-4 初始位置和目标位置设置表

参数名称	我机 1	我机 2	我机 3
初始位置	(0 km,0 km)	(0 km,2 km)	(6 km,0 km)
目标位置	(17.5 km,17.5 km)	(16 km,17.5 km)	(17.5 km,16 km)

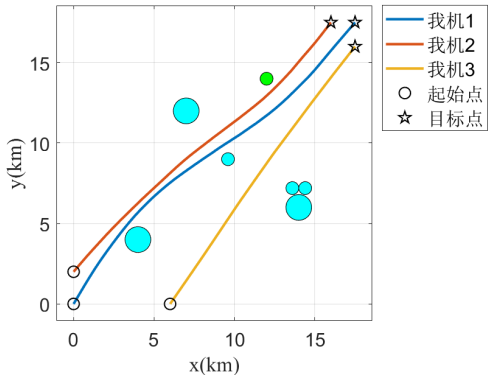
如表 3-4 所示设置仿真场景，随后对三架无人机进行航路规划，航路规划如图 3-11 所示。



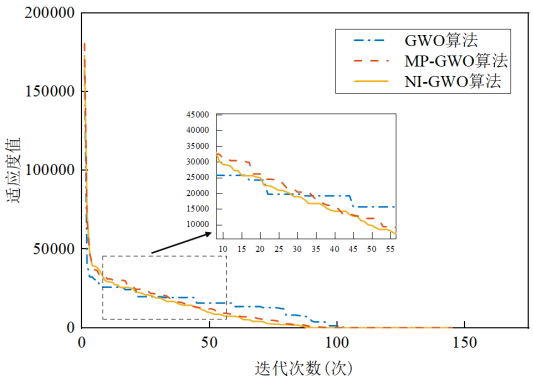
(a) GWO 算法航路规划图



(b) MP-GWO 算法航路规划图



(c) NI-GWO 算法航路规划图



(d) 适应度值结果图

图 3-11 三架无人机在不同算法下的仿真结果图

如图 3-11 所示，三种算法对 3 架无人机航路规划都存在可行性。然而，本文所提的 NI-GWO 算法对威胁区的避障效果优于 GWO 算法和 MP-GWO 算法且规划路径距离最短。为了进一步分析无人机路径规划效果，重复 500 次仿真实验，对实验过程中的数据进行统计如图 3-12 所示。

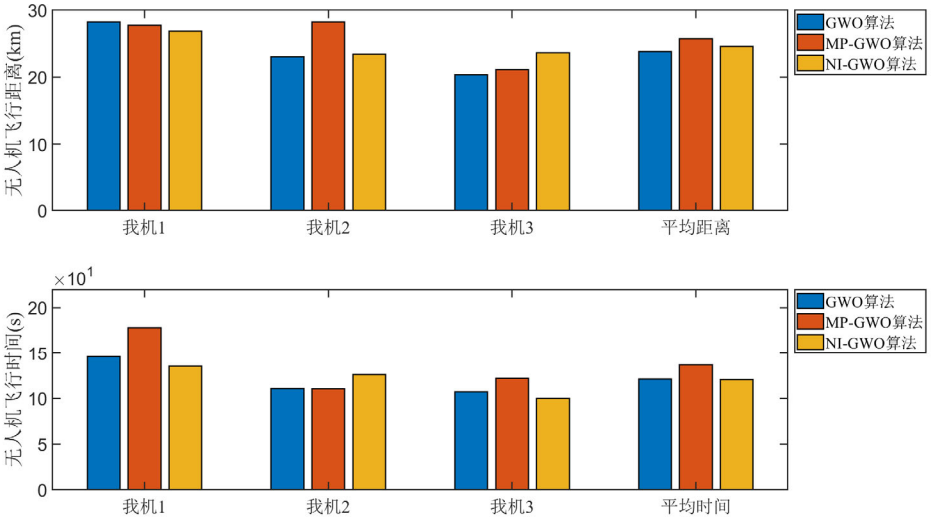


图 3-12 三架无人机的航路规划飞行数据

如图 3-12 所示，分析了三架无人机在三种航路规划算法中的飞行时间和距离，分别统计三种算法在该环境中运行的相关特征（最大航行距离、最小航行距离、最大航行时间、最小航行时间和适应度收敛值），如图 3-13 所示。

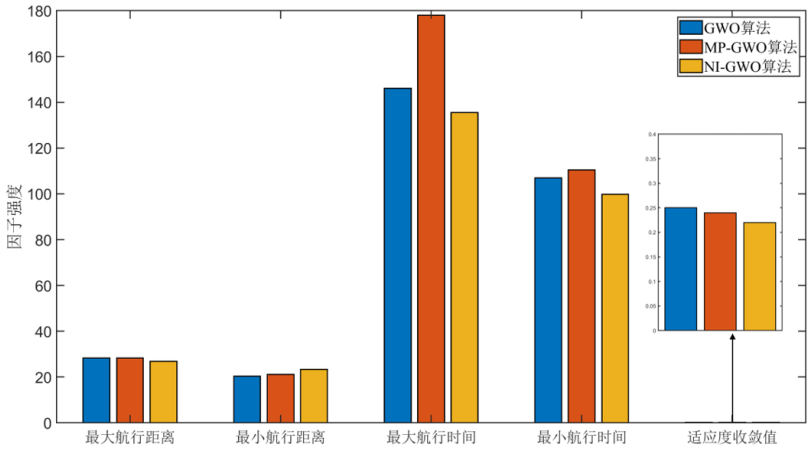


图 3-13 三架无人机的因子强度图

如图 3-13 所示，可以看出三种算法在对 3 架无人机进行航路规划过程中，NI-GWO 算法在最大航行距离、最小航行距离、最大航行时间、最小航行时间和适应度收敛值的表现上均优于两种传统算法。仿真结果表明，本文所提的 NI-GWO 算法在三架无人机路径规划效果上也优于传统的 GWO 算法和 MP-GWO 算法。同时，相比于对 2 架无人机航路规划过程中三种算法的相差无几，在 3 架无人机航路规划中本文所提的 NI-GWO 算法航路规划的效果逐渐提高，这表明本文所提的算法在面对更复杂问题中具有相对优势。

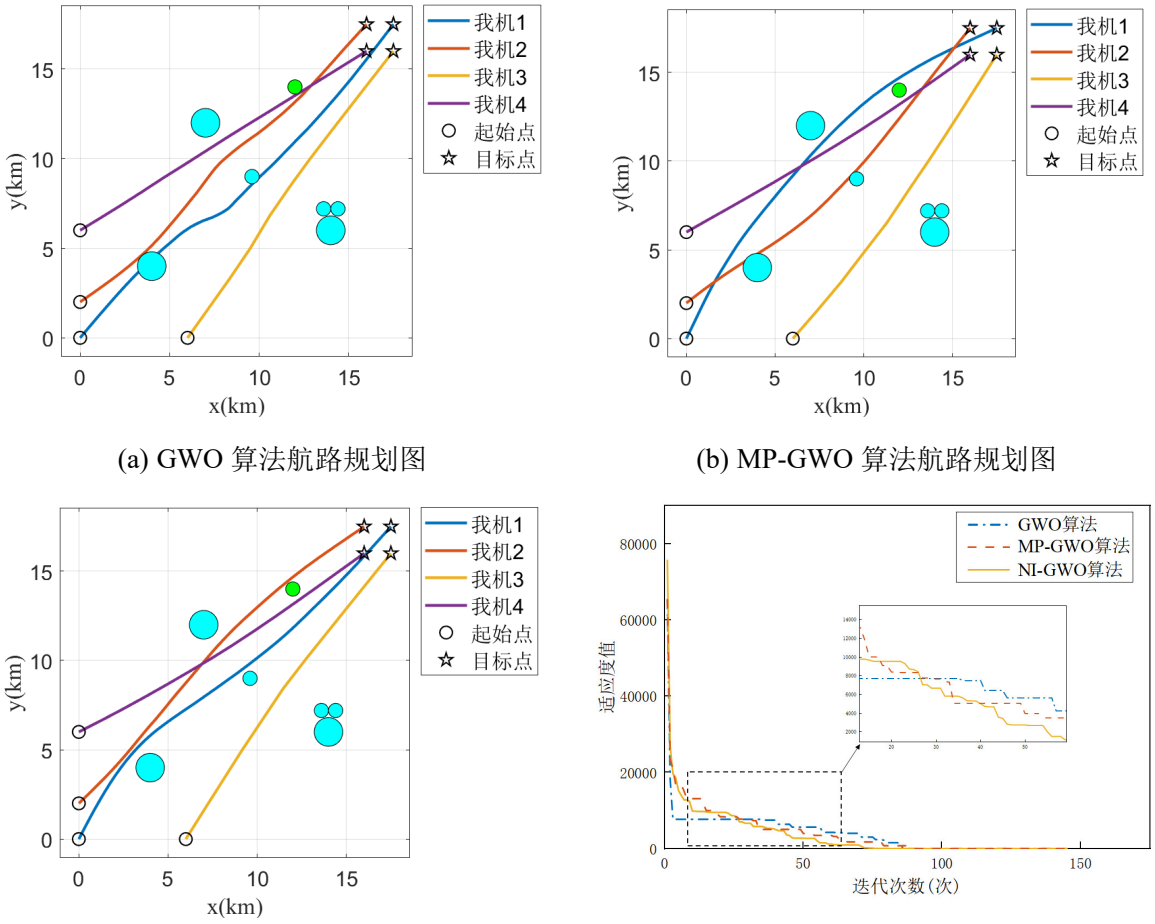
(3) 四架无人机航路规划

最后，对四架无人机进行路径规划，按照如表 3-1 所示设置算法迭代次数与威胁区信息参数后对四架无人机进行路径规划仿真。无人机位置信息和目标信息如表 3-5 所示。

表 3-5 初始位置和目标位置设置表

参数名称	我机 1	我机 2	我机 3	我机 4
初始位置	(0 km,0km)	(0km,2km)	(6km,0km)	(0km,6km)
目标位置	(17.5km,17.5km)	(16km,17.5km)	(17.5km,16km)	(16km,16km)

不同算法下的航路规划仿真结果图如图 3-14 所示。



(a) GWO 算法航路规划图

(b) MP-GWO 算法航路规划图

(c) NI-GWO 算法航路规划图

(d) 适应度值结果图

图 3-14 四架无人机在不同算法下的仿真结果图

如图 3-14 所示，GWO 算法路径规划效果最差，MP-GWO 算法路径规划效果次之，NI-GWO 算法路径规划效果最好。本文所提的 NI-GWO 算法对威胁区的避障效果最佳且规划路径距离最短。为了进一步分析无人机路径规划效果，重复 500 次仿真实验，对实验过程中的数据进行统计如图 3-15 所示。

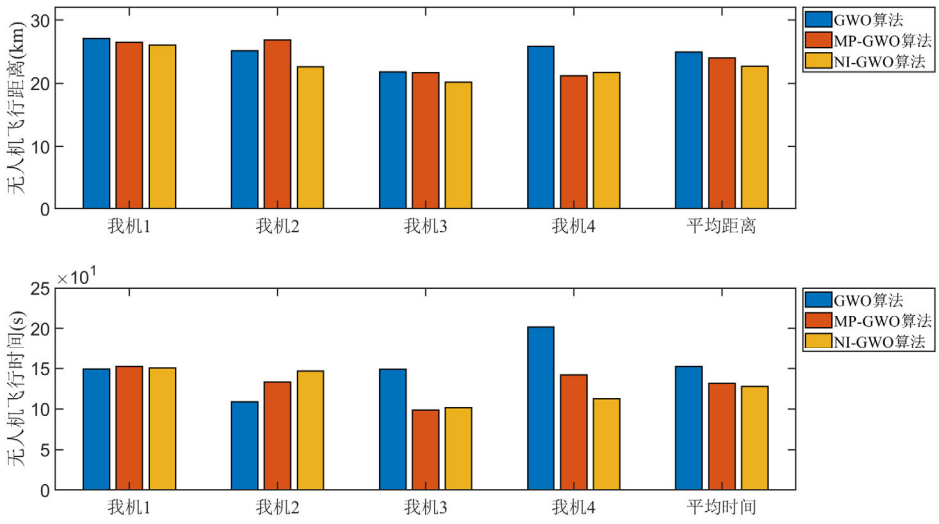


图 3-15 四架无人机的航路规划飞行数据

如图 3-15 所示，分析了四架无人机在三种航路规划算法中的飞行时间和距离，分别统计三种算法在该环境中运行的相关特征（最大航行距离、最小航行距离、最大航行时间、最小航行时间和适应度收敛值）如图 3-16 所示。

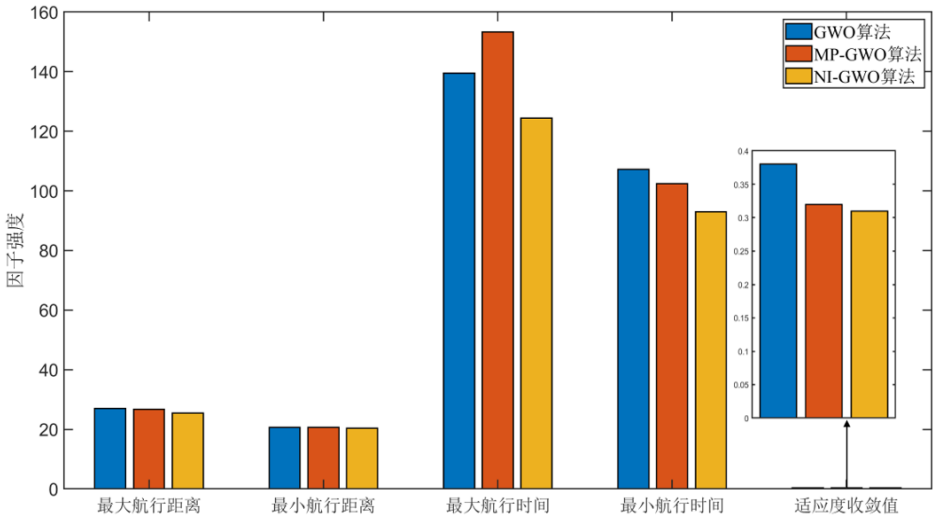


图 3-16 四架无人机背景下的因子强度图

如图 3-16 所示，可以看出本文所提的 NI-GWO 算法对 4 架无人机航路规划过程中在航行距离和航行时间上短于传统的 GWO 算法和 MP-GWO 算法，表明该算法在多机航路规划过程中可以尽可能的减少航行距离和航行时间。

综上分析，本文所提的 NI-GWO 算法在 2 架无人机航路规划中与传统算法航路规划能力相似，然而其在多机航路规划中表现出了相比传统算法更优的路径规划能力。这表明，本文所提的 NI-GWO 算法在多机航路规划上的效果上优于传统的 GWO 算法和 MP-GWO 算法，可以作为多机航路规划引导提供可靠的路径规划方法。即通过上述实验表明本文所提的 NI-GWO 算法在面对更为复杂的航路规划任务时有更好的航路规划

表现。

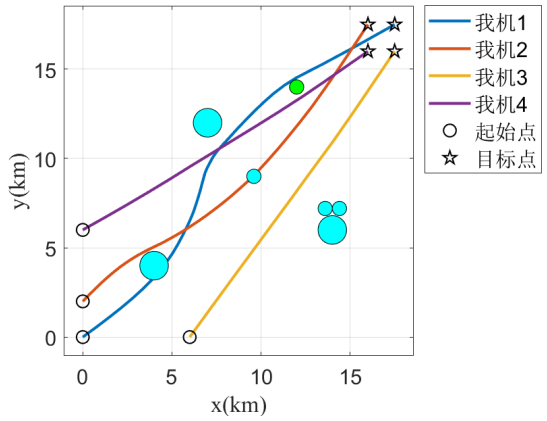
实验二：算法鲁棒性分析

为了检验算法的鲁棒性，本文设置了两组分析对照实验（算法迭代次数更改和路径规划场景更改）。(1) 算法迭代次数更改：为了验证模型在迭代次数降低时的表现，本实验分别设置了 100 次和 60 次迭代次数以分析三种算法在迭代次数不足时的表现。(2) 路径规划场景更改：本实验分别增加威胁区和减少威胁区，以分析三种算法对不同威胁区场景的表现。

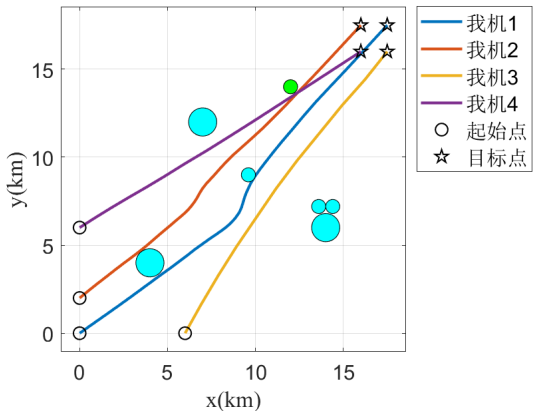
(1) 算法迭代次数更改

本次实验为了验证在迭代次数减少时 GWO 算法、MP-GWO 算法和 NI-GWO 算法的路径规划能力，首先将迭代次数降低至 100 次，障碍物设置与实验一相同，进行了四架无人机的仿真测试，以收集和分析该编队协同航路规划实验的仿真结果，具体地图场景设置如表 3-5 和表 3-3 所示。

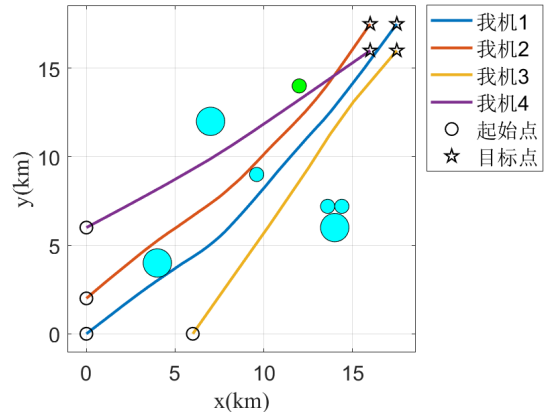
设置迭代次数为 100 次进行仿真，对应无人机路径规划仿真图如图 3-17 所示。



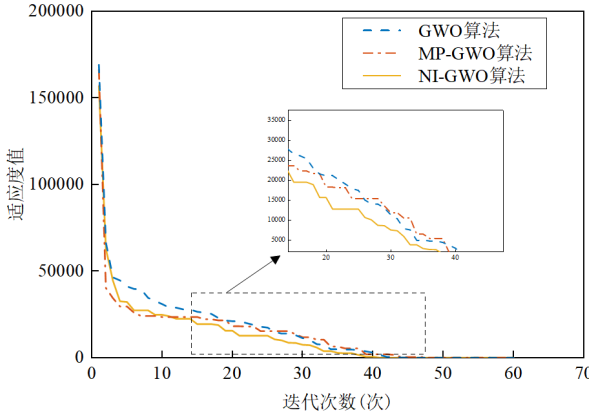
(a) GWO 算法航路规划图



(b) MP-GWO 算法航路规划图



(c) NI-GWO 算法航路规划图



(d) 适应度值结果图

图 3-17 迭代次数为 100 次的仿真结果图

如图 3-17 所示，在迭代次数降低至 100 次时，GWO 算法穿越威胁区域次数迅速提高，MP-GWO 算法路径规划效果次之，NI-GWO 算法路径规划效果大致稳定。由上可以看出，相较于 GWO 算法而言，本文所提的 NI-GWO 算法对迭代次数不敏感，可以容许一定的训练次数不足。为了进一步分析无人机路径规划效果，重复 500 次仿真实验，对实验过程中的数据进行统计如图 3-18 所示。

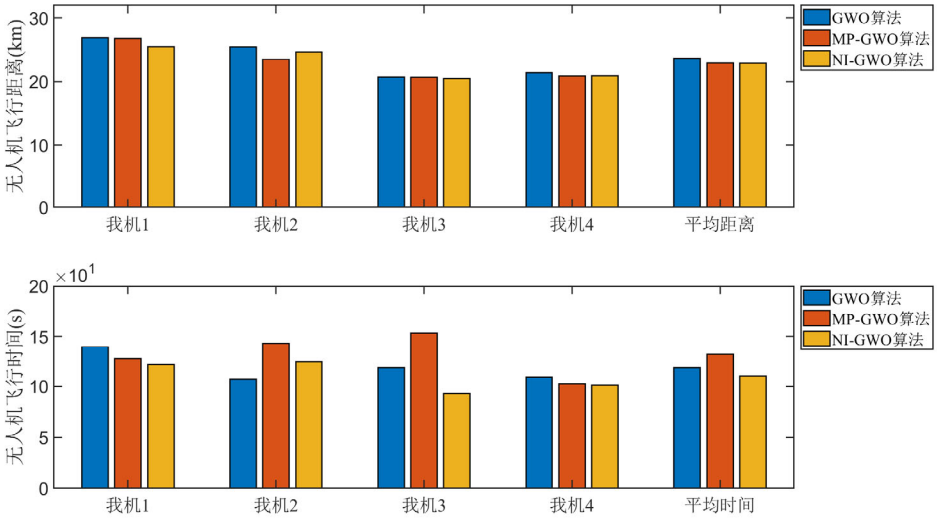


图 3-18 迭代次数为 100 次的航路规划飞行数据

如图 3-18 所示，分析了迭代次数为 100 次时各无人机在三种航路规划算法中的飞行时间和距离，分别统计三种算法在该环境中运行的相关特征（最大航行距离、最小航行距离、最大航行时间、最小航行时间和适应度收敛值），如图 3-19 所示。

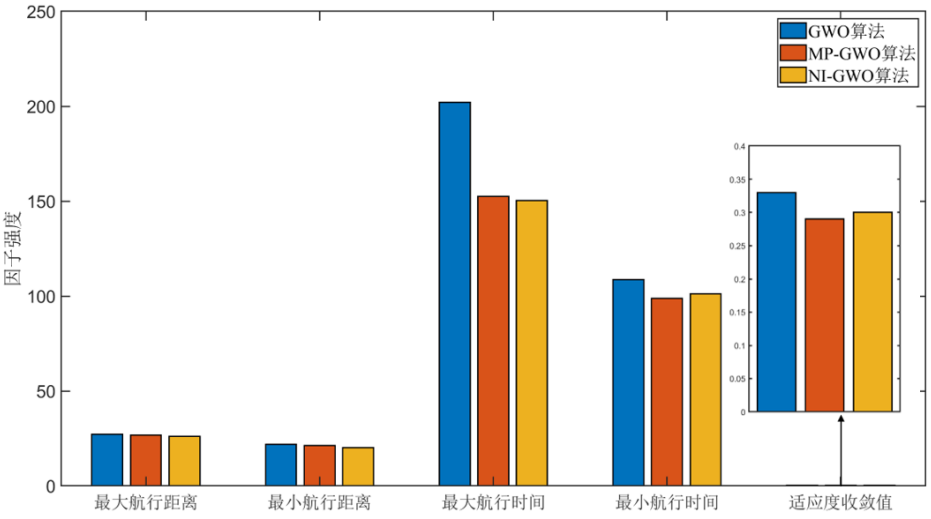
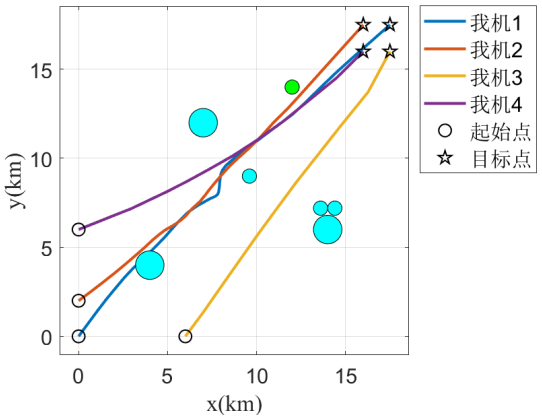


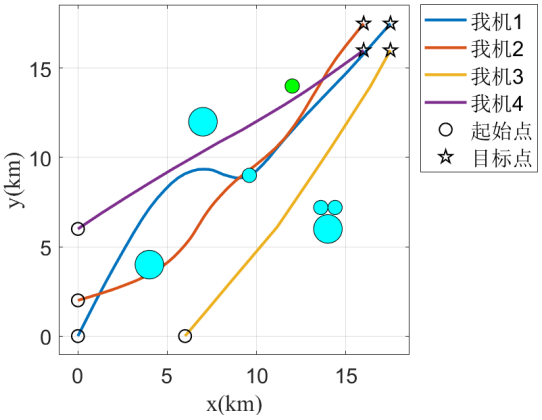
图 3-19 迭代次数为 100 次的因子强度图

从仿真统计数据可以看出，迭代次数的降低导致了 GWO 和 MP-GWO 算法的性能迅速降低（包括迅速增长的航行距离和航行时间），相比之下，本文所提的 NI-GWO 算法在遇到迭代次数降低时展现出了较优的效果。

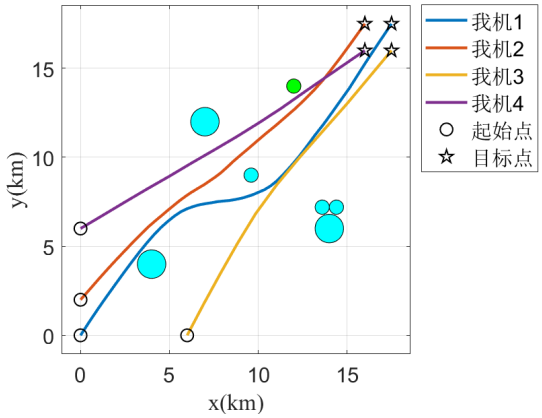
随后将迭代次数减少至 60 次研究无人机编队的飞行性能。其仿真结果如图 3-20 所示。



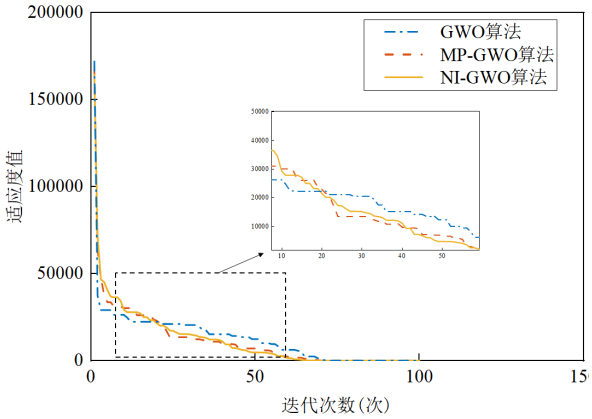
(a) GWO 算法航路规划图



(b) MP-GWO 算法航路规划图



(c) NI-GWO 算法航路规划图



(d) 适应度值结果图

图 3-20 迭代次数为 60 次的仿真结果图

如图 3-20 所示，在迭代次数降低至 60 次时，GWO 和 MP-GWO 算法路径规划效果迅速降低（多次穿越威胁区），而 NI-GWO 算法路径规划效果大致稳定。以上分析可以看出，相较于 GWO 和 MP-GWO 算法而言，本文所提的 NI-GWO 算法对迭代次数不敏感，可以容许一定的训练次数不足。为了进一步分析无人机路径规划效果，重复 500 次仿真实验，对实验过程中的数据进行统计如图 3-21 所示。

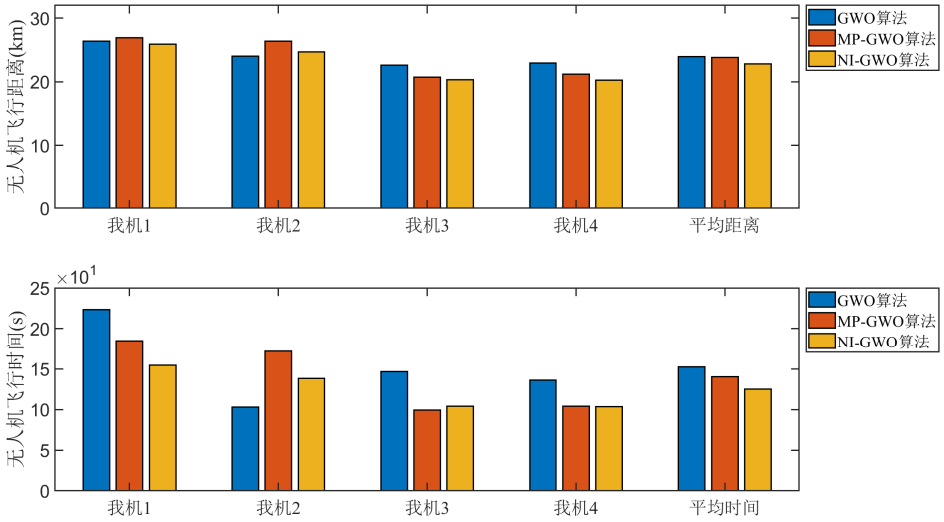


图 3-21 迭代次数为 60 次的航路规划飞行数据

如图 3-21 所示，从 60 次迭代的结果来看，三种算法的路径规划结果均有降低，可见迭代次数的降低会极大的影响无人机的路径规划能力。分别统计三种算法在该环境中运行的相关特征（最大航行距离、最小航行距离、最大航行时间、最小航行时间和适应度收敛值），如图 3-22 所示。

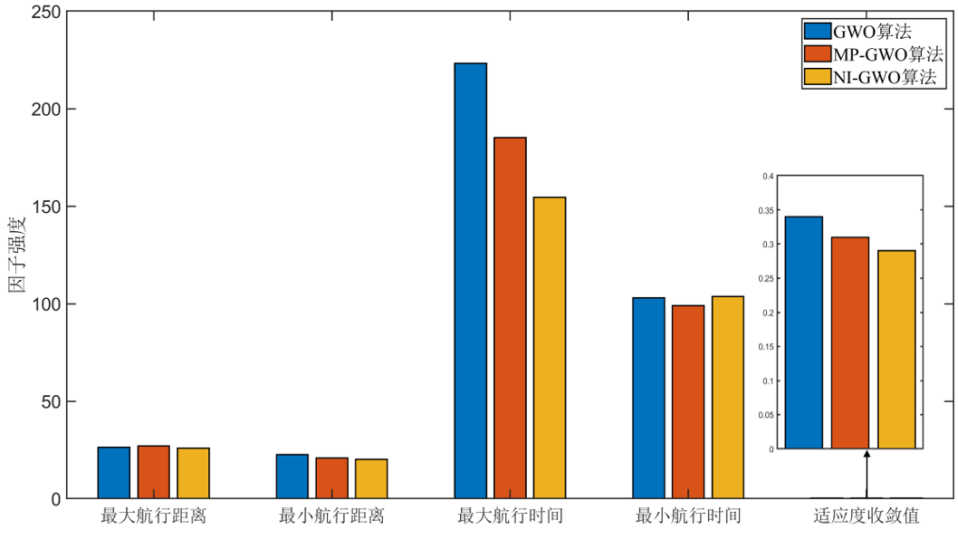


图 3-22 迭代次数为 60 次的因子强度图

从仿真统计数据可以看出，迭代次数的降低为 60 次导致了 GWO 和 MP-GWO 算法的航路规划能力迅速降低（包括迅速增长的航行距离和航行时间），相比之下，本文所提的 NI-GWO 算法在遇到迭代次数降低时展现出了较优的效果。由此可见 NI-GWO 算法在大幅度降低迭代次数的情况下并未受到较大的波动，具有较强的鲁棒性。

(2) 航路规划场景更改实验

为了验证在航路规划场景改变时 GWO 算法、MP-GWO 算法和 NI-GWO 算法的航路规划能力，本实验对基础的航路规划场景中的威胁区个数进行增减，迭代次数保持不

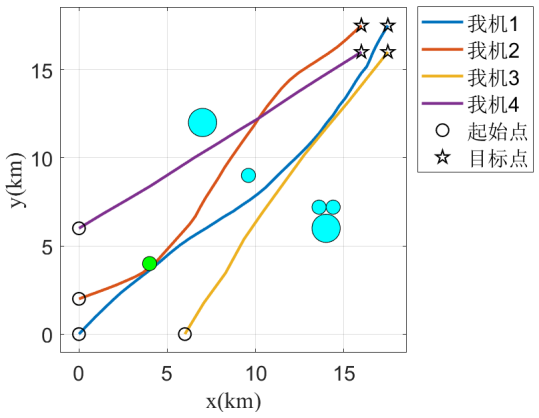
变，开展四架无人机的模拟实验，记录并分析编队协作路径规划的效果，以验证该算法在不同场景下的鲁棒性。

首先，基于如表 3-3 所示的威胁区，本实验将在减少部分威胁区的情况下进行仿真，以验证不同算法的鲁棒性，其中更新后的威胁区信息如表 3-6 所示。

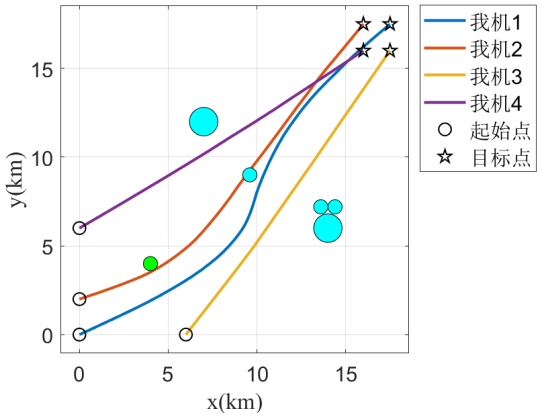
表 3-6 威胁区设置表

威胁区中心	雷达威胁区 1	火炮威胁区 1	火炮威胁区 2	火炮威胁区 3	火炮威胁区 4	火炮威胁区 5
横坐标(km)	4	7	9.6	14	14.4	13.6
纵坐标(km)	4	12	9	6	7.2	7.2
半径(km)	0.4	0.8	0.4	0.8	0.4	0.4

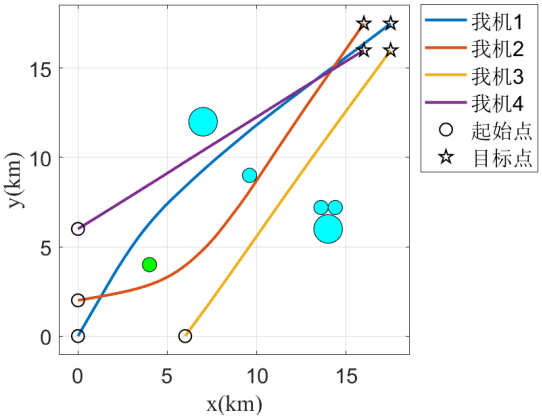
不同算法下的航路规划仿真结果如图 3-23 所示。



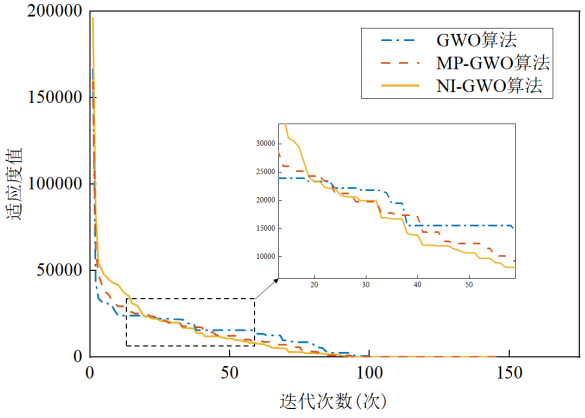
(a) GWO 算法航路规划图



(b) MP-GWO 算法航路规划图



(c) NI-GWO 算法航路规划图



(d) 适应度值结果图

图 3-23 减少威胁区数量后的仿真结果图

如图 3-23 所示，在威胁区域发生变化后，GWO 和 MP-GWO 算法路径规划能力有所降低（多次穿越威胁区），而 NI-GWO 算法路径规划大致稳定。由上可以看出，相较

于 GWO 和 MP-GWO 算法而言，本文所提的 NI-GWO 算法对威胁区场景的变化不敏感，可以适应复杂多变战场环境下的航路规划。为了进一步分析无人机路径规划效果，重复 500 次仿真实验，对实验过程中的数据进行统计如图 3-24 所示。

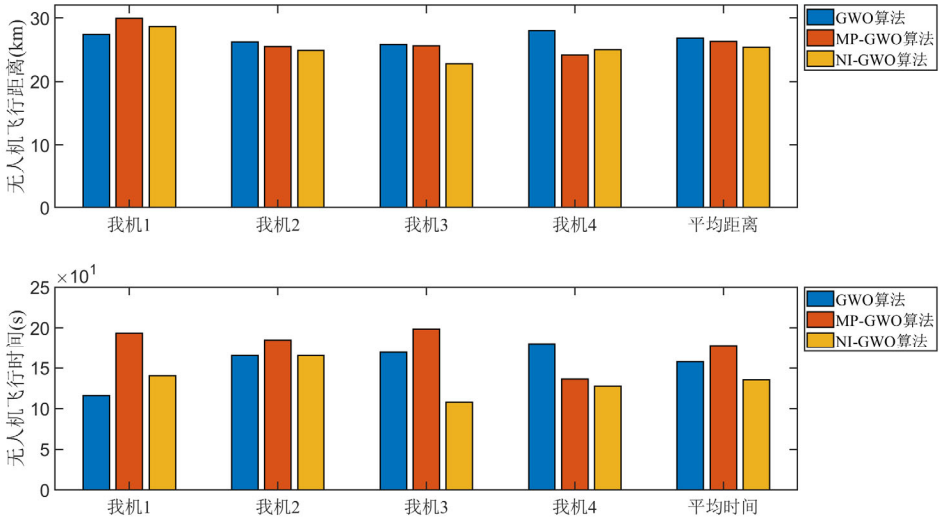


图 3-24 减少威胁区后航路规划飞行数据

如图 3-24 所示，减少威胁区数量后，三种算法的路径规划结果有所提升，但 GWO 算法和 MP-GWO 算法在路径规划中会存在穿越威胁区的情况，相对而言，NI-GWO 算法的稳定性能较高。分别统计三种算法在新环境中运行的相关特征（最大航行距离、最小航行距离、最大航行时间、最小航行时间和适应度收敛值），如图 3-25 所示。

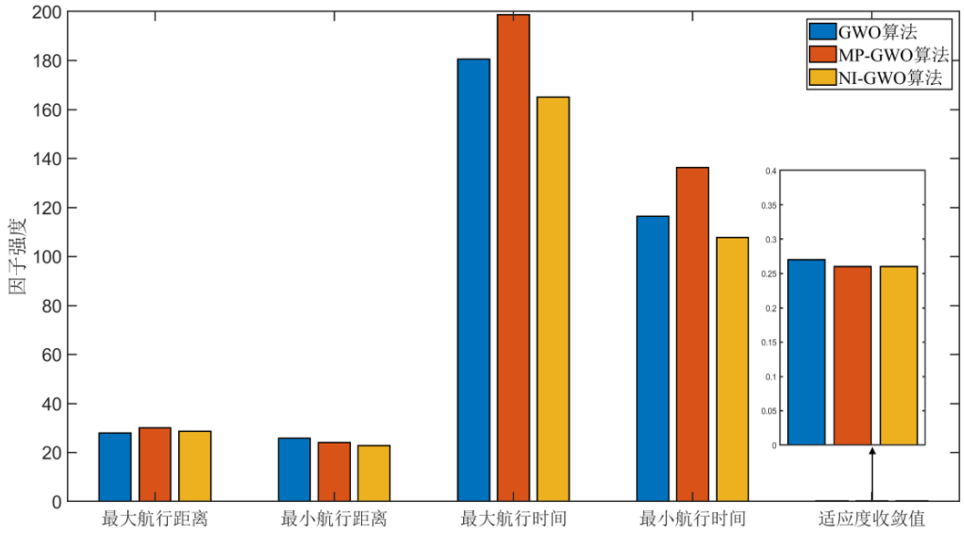


图 3-25 减少威胁区后因子强度图

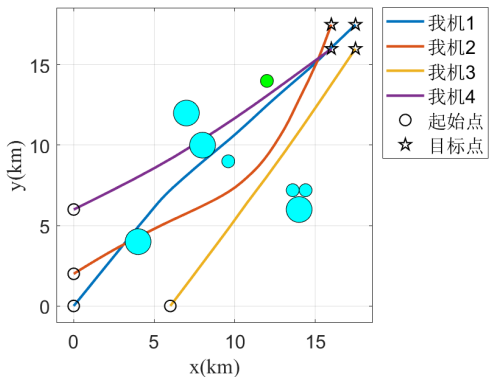
从仿真统计数据可以看出，在减少威胁区后 MP-GWO 算法的航行时间为三种算法中最长的；GWO 算法的最大航行距离与 NI-GWO 算法接近，但是最小航行距离最大；NI-GWO 算法的最小航行距离最小，且航行时间和航行速度均有着最优的效果。相比之下，本文所提的 NI-GWO 算法在所设置的新场景中的航路规划能力更优。

随后验证增加威胁区数量时三种算法对不同威胁区场景的鲁棒性。在如表 3-3 所示的威胁区基础上增加了 3 个火炮威胁区，增加后的威胁区信息如表 3-7 所示。

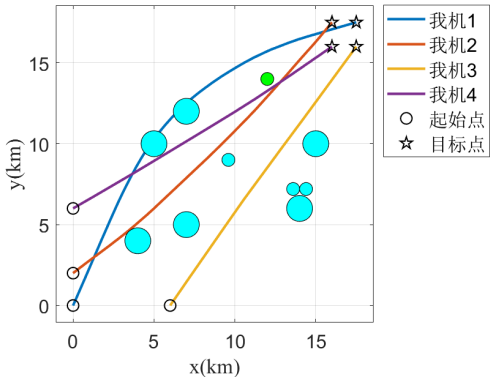
表 3-7 增加后的威胁区设置表

威胁区中心	雷达威胁区 1	雷达威胁区 2	火炮威胁区 1	火炮威胁区 2	火炮威胁区 3	火炮威胁区 4	火炮威胁区 5	火炮威胁区 6	火炮威胁区 7	火炮威胁区 8	火炮威胁区 9
横坐标 (km)	4	12	4	7	9.6	14	14.4	13.6	5	7	15
纵坐标 (km)	4	14	4	12	9	6	7.2	7.2	10	5	10
半径 (km)	0.4	0.4	0.8	0.8	0.4	0.8	0.4	0.4	0.8	0.8	0.8

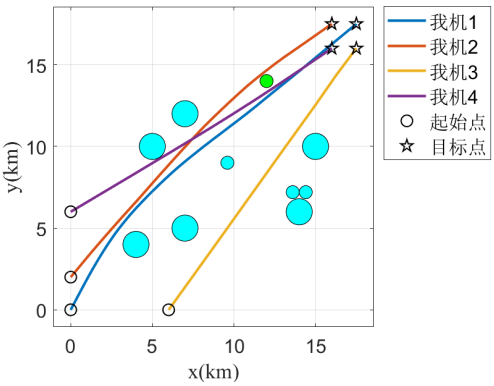
不同算法下的路径规划仿真结果如图 3-26 所示。



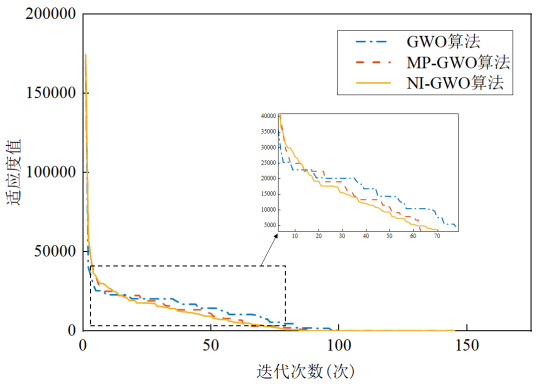
(a) GWO 算法航路规划图



(b) MP-GWO 算法航路规划图



(c) NI-GWO 算法航路规划图



(d) 适应度值结果图

图 3-26 增加威胁区数量的仿真结果图

在图 3-26 中可以看出，新增威胁区域后，GWO 和 MP-GWO 算出现多次穿越威胁区的情况，而 NI-GWO 算法没有出现穿越威胁区的情况。由上可以看出，相较于 GWO 和 MP-GWO 算法而言，本文所提的 NI-GWO 算法对更为复杂的作战场景有着相对优秀的路径规划能力。为了进一步分析无人机路径规划效果，重复 500 次仿真实验，对实验过程中的数据进行统计如图 3-27 所示。

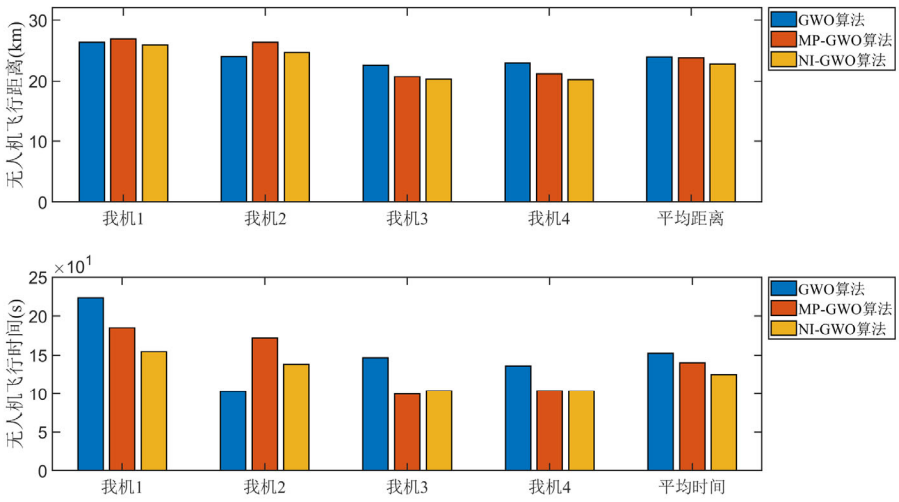


图 3-27 增加威胁区后航路规划飞行数据

如图 3-27 所示，在威胁区数量增加后，三种算法的路径规划结果均有所降低（航行距离增大、航行时间延长）。GWO 算法和 MP-GWO 算法在路径规划中会多次出现穿越威胁区的情况，而 NI-GWO 算法避免了穿越威胁区的情况，保证了路径规划的可靠性。最后，分别统计三种算法在新环境中运行的相关特征（最大航行距离、最小航行距离、最大航行时间、最小航行时间和适应度收敛值），其结果如图 3-28 所示。

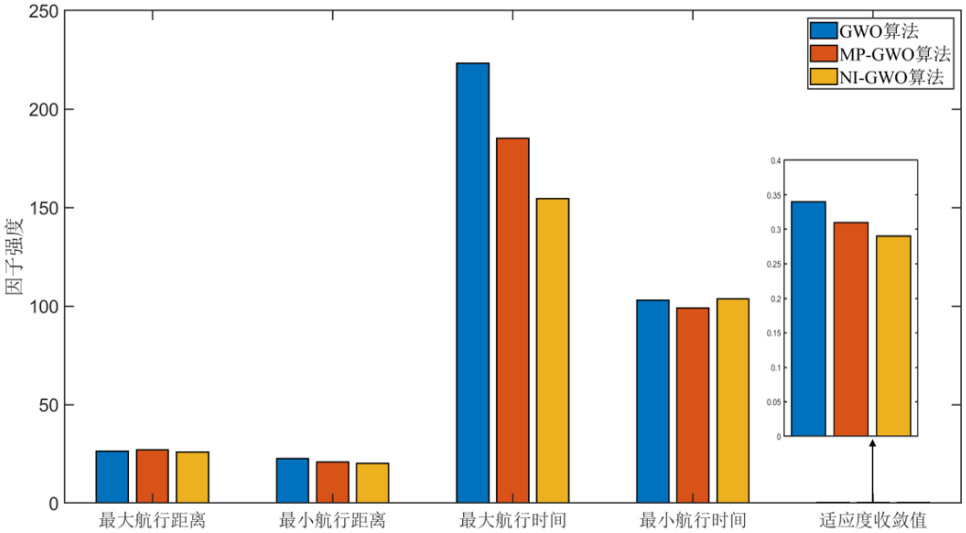


图 3-28 增加威胁区后因子强度图

如图 3-28 所示，在增加威胁区后 MP-GWO 算法的航行距离为三种算法中最长的；GWO 算法的航行时间是三种算法中最长的；相较于两种传统算法而言，尽管本文所提的 NI-GWO 算法并未做到所有性能均为最优，但其在多个性能评价指标中进行了均衡且在新场景下的路径规划中未穿越威胁区。相比于两种传统算法，本文所提的 NI-GWO 算法在全新的复杂场景下的路规划能力更优。

通过本文设置的三种仿真实验（不同架无人机航路规划分析、迭代次数鲁棒性分析以及新增威胁区域后航路规划的鲁棒性分析）分析结果表明，传统算法 GWO 算法和 MP-GWO 算法由于适应度函数设置单一、收敛因子取值方式较为随机和位置更新方式为静态更新，从而导致算法在规避威胁区域方面的能力不足。与之相比，本章所提的 NI-GWO 算法通过构建综合的航迹评估函数，并采用非线性的收敛因子机制，同时兼顾了 Alpha、Beta 和 Delta 狼的地位权重动态更新位置，显著提升了在多无人机航线规划问题上的性能。因此，该算法在迭代次数和适应不同作战场景的能力上展现了较强的鲁棒性。

3.5 本章小结

本章针对无人机编队航路规划问题，提出一种基于 NI-GWO 算法的多无人机航路规划方法。首先，根据多无人机协同路径规划问题，建立了多机协同航路规划模型；其次，对 GWO 算法的收敛因子和位置更新方式进行了改进，提出了 NI-GWO 算法模型；随后，分别设置两架、三架和四架无人机仿真场景，以验证 NI-GWO 算法对多无人机航路规划的有效性；最后，通过更改迭代次数和增减威胁区域以验证 NI-GWO 算法对多无人机航路规划的鲁棒性。

第 4 章 基于深度强化学习的无人机空战机动决策模型

在无人机航路规划引导进入作战区域后，各无人机需要完成相应的机动决策从而实现指定的战术目标。本章针对空战机动决策进行建模，从而为下一章中无人机编队协同策略模型提供机动决策元策略模型。首先，根据攻击区的解算模型，采用 BP 神经网络对导弹攻击区进行解算与改进；其次，针对 SAC 算法的收敛速度缓慢，可能会陷入局部最优的不足，设计了 E-SAC 算法，并根据算法模型和导弹攻击区终端约束选择了合适的状态空间和动作空间，建立了相应的奖励函数；最后，在不同近距空战场景的仿真中验证算法的有效性。

4.1 问题分析

进入作战区域后，各无人机需要完成对应的机动决策以实现作战任务。无人机机动决策应先解算出导弹的攻击区，随后引导无人机完成对应的机动决策，实现对目标敌机的追踪与打击任务。

针对无人机编队的机动策略模型问题，本章主要研究了导弹攻击区域的解算及机动策略的设计。首先，依据攻击区的计算模型，制定了相应的攻击区解决方案，通过黄金分割法解算出导弹攻击区数据完成 BP 神经网络对攻击区的拟合，再到对传统的 BP 神经网络进行改进以提升拟合的准确度，完成了对导弹攻击区解算的设计；其次，根据解算出的导弹攻击区终端约束条件制定奖励函数，并建立无人机空战机动决策的状态空间和动作空间，设计了基于 E-SAC 算法的无人机编队作战的机动决策模型；最后，基于 MaCA 平台完成了近距空战机动决策的仿真验证。本章具体研究内容如图 4-1 所示。

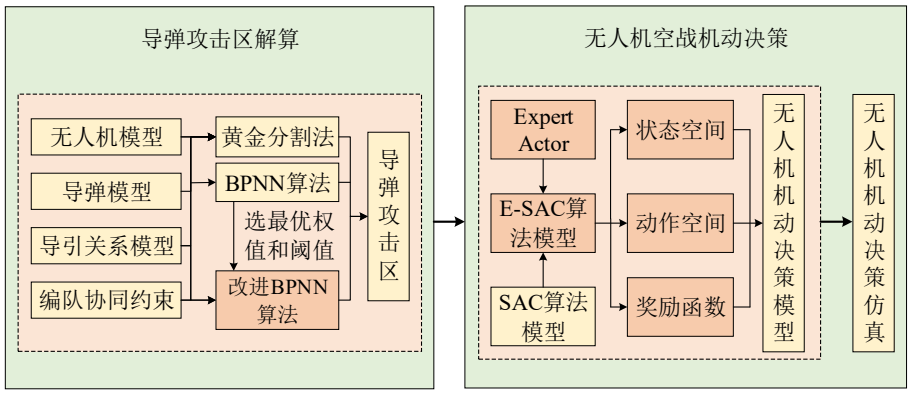


图 4-1 基于深度强化学习的无人机空战机动决策模型内容框图

4.2 导弹攻击区解算方法

4.2.1 导弹数学模型

根据 2.1 节中所述假设，可以建立以下模型。

(1) 导弹动力学模型

导弹三自由度的动力学方程组[190]：

$$\begin{cases} \dfrac{dv_m}{dt} = \dfrac{F_m - F_a}{m_m} - g\sin\theta_m \\ \dfrac{d\theta_m}{dt} = \dfrac{g(n_{my} - \cos\theta_m)}{v_m} \\ \dfrac{d\varphi_m}{dt} = -\dfrac{n_{mz}g}{v_m\cos\theta_m} \end{cases} \tag{4-1}$$

其中，v_m、θ_m、φ_m依次代表空对空导弹的速度、攻角和侧滑角；F_m和F_a分别对应于导弹的推力和遭遇的气动阻力；n_{my}与n_{mz}各自指向导弹的横向和纵向过载力。

(2) 导弹运动学模型

导弹三自由度的运动学方程组[190]：

$$\begin{cases} \dfrac{dx_m}{dt} = v_m\cos\theta_m\cos\varphi_m \\ \dfrac{dy_m}{dt} = v_m\sin\theta_m \\ \dfrac{dz_m}{dt} = -v_m\cos\theta_m\sin\varphi_m \end{cases} \tag{4-2}$$

其中，x_m、y_m和z_m分别表示在X轴、Y轴和Z轴正方向上的导弹位置坐标。

4.2.2 导弹运动与导引模型

我机发射的导弹与敌机的弹目相对运动关系如图 4-2 所示。

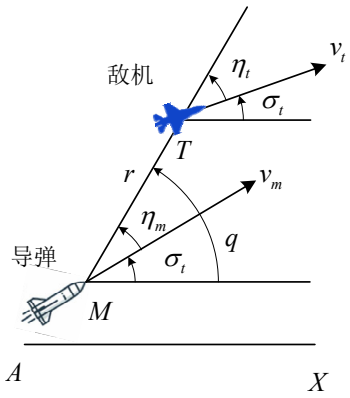


图 4-2 导弹与目标相对运动关系示意图

如图 4-2 所示，其中M代表导弹，而T表示目标飞机的具体位置；导弹与目标飞机的间隔距离用r表示，此距离在导弹成功命中目标时减至零；MT用于表示导弹与目标之间的连线，即追踪线，而AX代表攻击平面上的水平基线；导弹的弹道角度由σ_m表示，目标的航迹角度由σ_t表示，它们反映了相对于水平基线的速度向量的角度；v_m与v_t分别描述导弹和目标的速度向量，显示了其方向和速度；q角度描绘了追踪线与水平基线的

交角，而η_m与η_t分别表示导弹和目标速度向量相对于追踪线的前置角，这表明了它们的移动方向与追踪线的角度差。

由此，得到的导弹与目标之间的相对运动模型的公式表示如下[190]：

$$\begin{cases} \dfrac{dr}{dt} = v_t \cos\eta_t - v_m \cos\eta_m \\ r\dfrac{dq}{dt} = v_m \sin\eta_m - v_T \sin\eta_T \\ q = \sigma_m + \eta_m \\ q = \sigma_T + \eta_T \\ \eta_1 = 0 \end{cases} \tag{4-3}$$

其中，$\eta_1 = 0$表示导引关系控制方程由方程组。

导弹的主要制导技术包括追踪法、平行法和比例导引法。其中，比例导引法能够有效地利用导弹的机动性，被广泛用于多种类型的导弹制导系统中。该方法的核心原理如式(4-4)所示[190]。

$$\eta_1 = \frac{d\sigma_m}{dt} - K\frac{dq}{dt} = 0 \tag{4-4}$$

其中，K表示比例导引中使用的比例常数。将上式结合起来，形成了比例导引法的数学模型[190]：

$$\begin{cases} \dfrac{dr}{dt} = v_t \cos\eta_t - v_m \cos\eta_m \\ r\dfrac{dq}{dt} = v_m \sin\eta_m - v_T \sin\eta_T \\ q = \sigma_m + \eta_m \\ q = \sigma_T + \eta_T \\ \dfrac{d\sigma_m}{dt} = K\dfrac{dq}{dt} \end{cases} \tag{4-5}$$

(1) 空空导弹最大攻击区

导弹的最大攻击范围指的是那些有一定几率成功击中目标的导弹的最大发射区域。导弹的最大攻击区计算考虑到了导弹的飞行轨迹、目标的位置、环境因素等多种因素，需要精确的数学模型支持。在其计算过程中，主要的约束条件可以从以下四个维度来考虑:

1) 在导弹的最大攻击范围内，雷达的截获几率$P_{dr_\max}$；

2) 导弹所需的能源运行时段t_{energy}；

3) 导弹在成功击中目标之前的最低飞行速度$v_{m_\min}$；

4) 击中目标时，弹目之间的最低相对速度$v_{r_\min}$。

综合上述条件可知，导弹的最终攻击范围限制条件如下式所定义[190]：

$$\begin{cases} P_{dr} \geq P_{dr\_\max} \\ t_e \leq t_{energy} \\ v_m \geq v_{m\_\min} \\ v_r \geq v_{r\_\min} \end{cases} \tag{4-6}$$

其中，P_{dr}表示载机的雷达截获概率；t_e表示空空导弹的实际飞行时长；v_m表示空对空导弹在被击中之前的飞行速率；v_r表示命中目标时的弹目相对速度。

(2) 空空导弹最小攻击区

导弹的最小攻击区域定义为无人机在保持最低安全发射距离的前提下，仍能够以一定成功率命中目标的范围[190]。在解算的过程中，主要的约束条件可以从以下三个维度来考虑：

1) 空空导弹的最大过载$n_{m_\max}$；

2) 空空导弹引信解除保险时间t_{fuze}；

3) 无人机最小安全距离d_{safe}。

综合以上三个维度，空对空导弹最小攻击区的约束如式所示：

$$\begin{cases} n_m \leq n_{m\_\min} \\ t_m \geq t_{fuze} \\ d_u \geq d_{safe} \end{cases} \tag{4-7}$$

其中，n_m代表空空导弹机动过载；t_m表示空空导弹达到目标所需飞行时间；d_u指的是无人机与目标之间的距离。

(3) 空空导弹不可逃逸攻击区

空对空导弹的“不可逃逸攻击区”定义为一个发射区域，在该区域内，无论目标进行何种机动操作，导弹都有能力摧毁该目标。因此，在空空导弹的不可逃逸攻击区内，目标的任何移动都注定无法摆脱导弹的锁定。空对空导弹的不可逃逸攻击区的约束条件与空空导弹最大攻击区的约束条件所述相同，即如式所示：

$$\begin{cases} P_{dr} \geq P_{dr\_ine} \\ t_e \leq t_{energy} \\ v_m \geq v_{m\_\min} \\ v_r \geq v_{r\_\min} \end{cases} \tag{4-8}$$

其中，P_{dr_ine}表示在不可逃逸攻击区内无人机雷达侦测到目标的概率。

4.2.3 传统 BP 神经网络的导弹攻击区拟合方法

(1) 解算流程

黄金分割法是一种传统的导弹的攻击区拟合方法，然而其计算速度缓慢，尤其是在弹道仿真中需要解算高阶微分方程，因此该方法难以适应空战场景中迅速变化的战况。鉴于无人机和目标在空战中的高速移动和复杂多变的战场态势，需要一种能够迅速响应的解算方法。

为此，本节采用一种快速拟合攻击区的技术——反向传播神经网络(Back Propagation Neural Network, BPNN)，即 BP 神经网络[191]，该算法能够有效处理复杂的数据模式，并且不需要对其内部的构造和参数进行深入的模型构建。因此，在这一节中，使用黄金分割法来计算导弹的攻击范围，并将其作为 BP 神经网络的训练样本。通过合适配置的神经网络进行数据拟合，旨在实现对攻击区域的精准预测，同时也满足无人机在空中战斗中的实时决策需求。其网络结构图如图 4-3 所示。

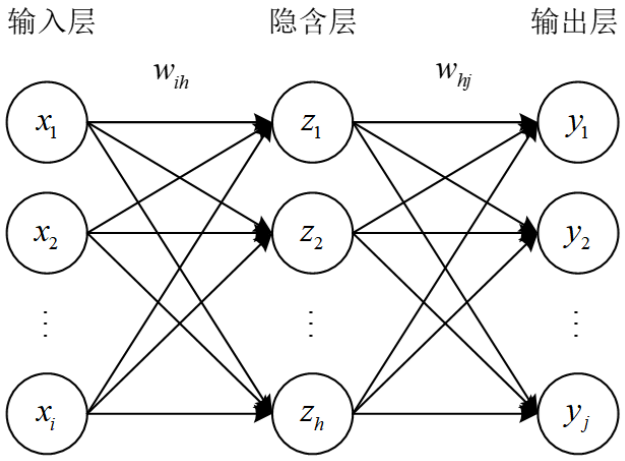


图 4-3 BP 神经网络结构图

在 BP 神经网络的前向传播阶段，数据按顺序经过网络各层，每一层对数据执行加权和偏置调整，随后通过激活函数进行处理，以传递给下一层的输入。当数据到达输出层时，网络产生一个预测结果。在反向传播阶段，预测结果与实际值比较，计算误差，然后这个误差沿着网络向后传播，途中根据误差的梯度调整每层的权重和阈值。通过这种方式，BP 神经网络在多次迭代中不断优化，以减少输出结果和实际值之间的差异。其具体流程图如图 4-4 所示。

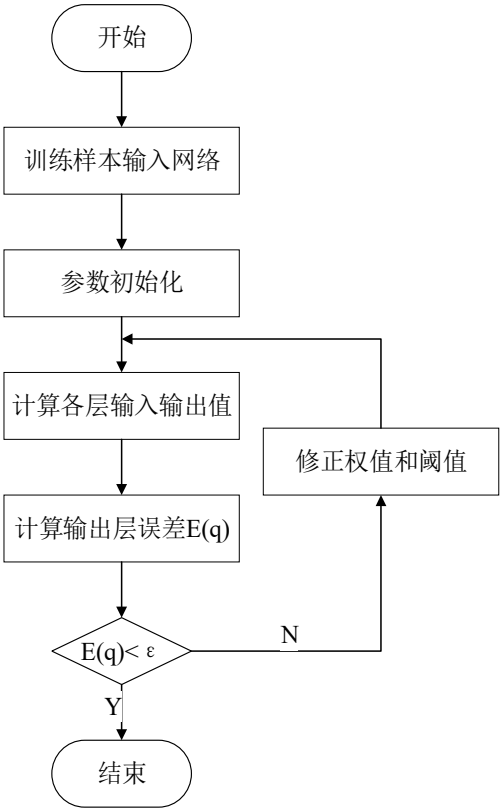


图 4-4 BP 神经网络算法流程图

为了提高 BP 神经网络的拟合精度并加快其训练速度，在这一节中根据目标线的方位角大小将数据集划分为 18 个不同的子集（设定方位角以每 20 度为一个区间划分数据集），通过对这些分组数据集分别使用 BP 神经网络进行训练，构建一个较为快速、准确的神经网络模型，满足实时性和精确度的需求。

在这个模型中，输入网络的训练集涵盖要素有：我机的飞行速度 v_u 、我机的飞行高度 h_u 、目标的航速与飞行的高度分别为 v_t 和 h_t ，输出是攻击区的边界距离。

(2) 仿真结果与分析

本节首先设定了不同的初始条件并采用黄金分割法[192]模拟出理论导弹攻击区数据，在仿真设置中，我机飞行角度和目标进入角度的变化范围为 $[0°,360°]$ ，我机和目标的飞行高度的变化范围为 $2000m$ 到 $4000m$ ，我机和目标飞行速度变化范围为 $100m/s$ 到 $450m/s$ 。总共生成了 253400 条数据作为神经网络的训练集。

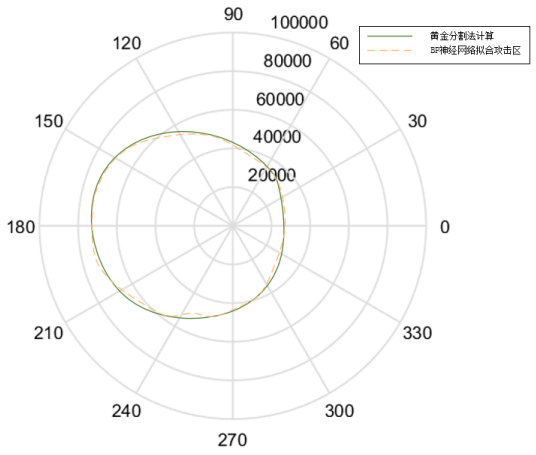
随后利用 MATLAB 的神经网络工具箱构建 18 个结构一致的神经网络。这些网络根据各自的 18 份训练集进行独立训练，以确保各自的准确性和效率。网络的具体参数详如表 4-1 所示，基于 4.2.3 节的方法进行仿真。

表 4-1 BP 神经网络参数

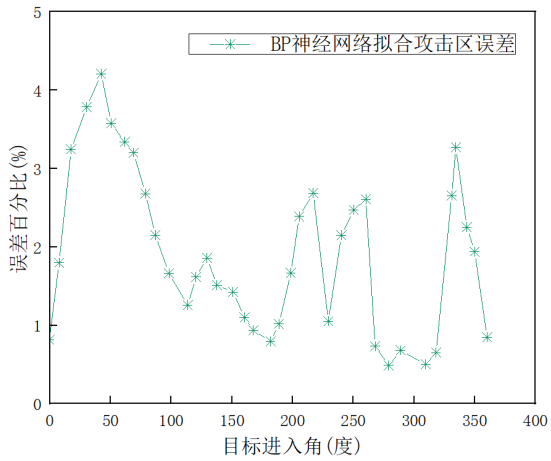
参数名称	参数大小
输入层神经元个数	4
隐含层个数	1
隐含层神经元个数	16
隐含层传递函数	tansig
输出层神经元个数	1
输出层传递函数	purelin
最大学习轮回	40000

完成训练后，通过 BP 神经网络得到的空空导弹攻击区域结果与经典的黄金分割法得出的结果进行比较。

以最大攻击区为例，当本次实验的初始条件为：我机的飞行速度 $v_u=150m/s$ 、飞行高度 $h_u=2000m$，目标速度 $v_t=200m/s$ 、高度 $h_t=2000m$ 时，两种算法拟合出的攻击区结果如图 4-5(a)所示，BP 神经网络模型计算的攻击区所需时间为 0.57 秒，其与黄金分割法的攻击区计算结果的误差分布如图 4-5(b)所示。



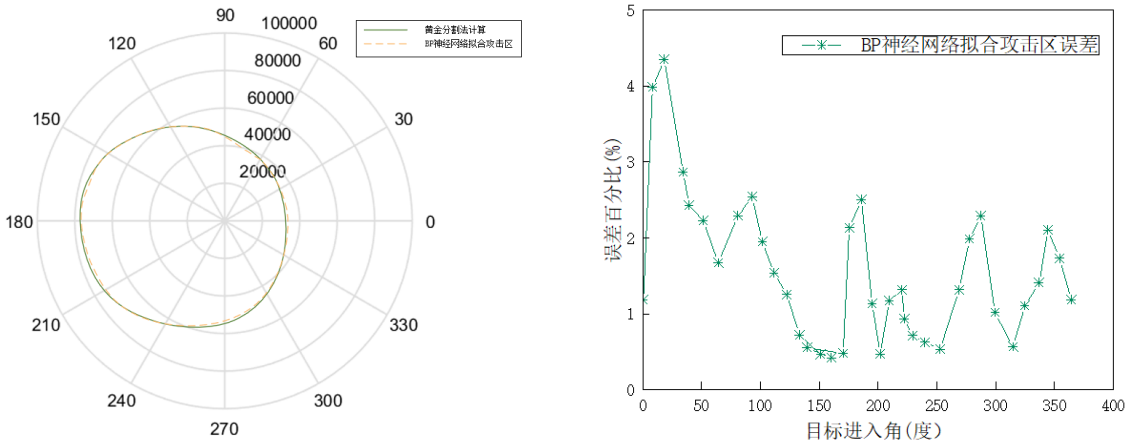
(a) 空空导弹攻击区仿真结果对比图(1)



(b) 神经网络拟合攻击区误差曲线图(1)

图 4-5 BP 神经网络仿真结果图(1)

当我机飞行速度 $v_u=300m/s$ 、飞行高度 $h_u=4000m$，目标速度 $v_t=250m/s$ 、高度 $h_t=3500m$ 时，两种算法拟合出的导弹攻击区如图 4-6(a)所示，BP 神经网络计算出的攻击区所需时间为 0.62 秒，其与黄金分割法攻击区计算结果的误差分布如图 4-6(b)所示。



(a) 空空导弹攻击区仿真结果对比图(2)　　(b) 神经网络拟合攻击区的误差曲线图(2)

图 4-6 BP 神经网络仿真结果图(2)

由仿真结果可知，采用 BP 神经网路拟合得到的导弹攻击区与使用黄金分割法计算得到的攻击区之间的误差百分比保持在 5%内，同时采用 BP 神经网络进行拟合的时间均小于 1 秒，远小于黄金分割法的时间，因此该算法的实时性能具有明显优势，同时也能基本满足所需的准确度，但是从图中可以看出，其效果仍有改进的空间。

4.2.4 基于改进神经网络的攻击区拟合方法

(1) 解算流程

空战背景中对无人机解算能力的准确度的要求较高，因此无人机必须准确地解算出攻击区以适应战场环境。然而，传统的空空导弹攻击区拟合方法在精度方面还存在提升空间，为了提升无人机在高速动态目标跟踪和精确打击方面的能力，本节提出了一种基于改进神经网络(Tuning of Backpropagation Neural Networks, TBPNN)的攻击区拟合方法，旨在通过对 BP 神经网络权重和阈值的优化，以提升计算精度，增强无人机的作战效率[193]。

1) 基于 TBPNN 算法的攻击区拟合步骤

首先输入 TBPNN 算法的权重和阈值作为配置参数，再由 BP 神经网络拟合出的攻击区预测输出值与黄金分割法所得到的期望输出值之间的误差绝对值和作为权重和阈值数据的评估分数，计算式如式所示：

$$F = k\left(\sum_{i=1}^{n} abs\left(y_i - o_i\right)\right) \tag{4-9}$$

其中，k指的是系数，n是指神经网络输出层中节点的数量，y_i代表网络中第i个节点所期望的输出值；而o_i是第i个节点上预测的输出值。

接下来对配置参数进行重组操作，随机生成一系列配置参数，某个配置参数被选中

进行重组的概率是基于其评估分数占所有配置参数评估分数总和的比例。这种方法确保了拥有较高评估分值的配置参数有更大的概率被选中参与下一次迭代。可以用如式所示描述该过程：

$$f_i = \frac{k}{F_i} \tag{4-10}$$

$$p_i = \frac{f_i}{\sum_{i=1}^{N} f_i} \tag{4-11}$$

其中，k表示系数；F_i表示当前权重和阈值的评价指标，该值越小代表配置参数的评估分数越高，因此在配置参数进行选择判断的时候将评估分数放入分母部分；p_i表示第i个体的选择概率；N表示所有配置参数数量。

将配置参数采用实数来进行编码，因此可以选择实数重组法增加配置参数的多样性。具体而言，所有配置参数中选取了第k个配置参数a_k与第l个配置参数a_l在j位进行重组，以产生新的配置参数，可以表示为如式所示：

$$\begin{cases} a_{kj} = a_{kj}(1-b) + a_{lj}b \\ a_{lj} = a_{lj}(1-b) + a_{kj}b \end{cases} \tag{4-12}$$

其中，b为随机数且$b \in [0,1]$。

随后，对第i个配置参数中的第j位a_{ij}进行改造操作，具体的操作方式如式所示：

$$a_{ij} = \begin{cases} a_{ij} + \left(a_{ij} - a_{\max}\right) * f(g) & r > 0.5 \\ a_{ij} + \left(a_{\min} - a_{ij}\right) * f(g) & r \le 0.5 \end{cases} \tag{4-13}$$

$$f(g) = r_2\left(1 - \frac{g}{G_{\max}}\right)^2 \tag{4-14}$$

其中，$a_{ij} \in [a_{\min}, a_{\max}]$，$r_2$和$r$均为随机生成的数值且$r \in [0,1]$，$g$表示当前的迭代次数，$G_{\max}$为迭代的最大限度。

本节中，首先，通过反向传播算法调整输入层参数，构建 BP 神经网络算法的基本结构；接着，将 BP 神经网络算法中的权重和阈值输入到的 TBPNN 方法中计算其评估分数；其次，通过选择、重组和改造操作来选出最适合的 BP 神经网络初始权重和阈值；最后，对导弹攻击区进行预测计算，并通过训练输出结果。整个算法流程详如图 4-7 所示。

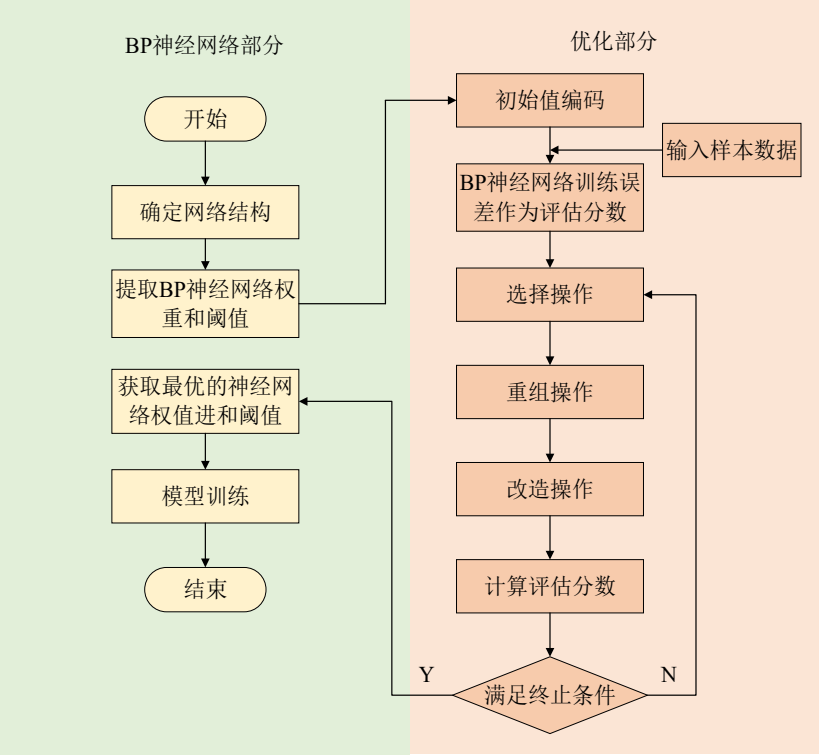


图 4-7 TBPNN 算法流程图

2) 基于 TBPNN 算法的无人机攻击区拟合模型

结合前述的 TBPNN 算法流程，可知该模型输入无人机和目标的实时状态信息，输出预测的最佳攻击区域，同时优化神经网络参数以提高预测的准确性和效率。其中优化过程分为评估分数、优化主循环两个主要模块。

a) 评估分数模块

该模块的主要用于评估每个输入进来的 BP 神经网络的阈值和权值在解算无人机攻击区精度方面的分数。

首先，输入一组神经网络的权重和阈值在 BP 神经网络中进行前向传播；然后，将神经网络的输出与实际的目标或期望值进行比较，计算误差，基于该误差确定评估分数，通常误差越小，评估分数越高，即神经网络预测的准确性更高；最后，输出每个权重和阈值的评估分数，以便用于后续的算法操作。其伪代码如表 4-2 所示。

表 4-2 评估分数模块伪代码

模块: 评估分数
Function score = evaluateScore(configuration, *y*, *o*, *k*) 　　ErrorSum = 0; 　　For *i* = 1:length(*y*) 　　ErrorSum = errorSum + abs(*y*(*i*) - o(*i*)); 　　End For 　　score = *k* / (1 + errorSum);　% 误差越小，评估分数越高 Return score

b) 优化主循环模块

该模块输入配置参数及其评估分数、迭代次数、重组率和改造率，输出是经过一系列优化操作后形成的新配置参数，将用于下一轮迭代。其伪代码如表 4-3 所示。

表 4-3 优化主循环模块伪代码

模块: 优化主循环
输入：配置参数 *i*、配置参数 *j* 和重组系数 *b*
输出：新配置参数
For 每个参数 = 1:length do:
生成随机数 *r*;
If *r* < 0.5
新配置参数 *i*= 配置参数 *i** (1 - b) +配置参数 *j** b;
新配置参数 *j*=配置参数 *j** (1 - b) +配置参数 *i** b;
End If
End For
Returen 新配置参数 *i*, 新配置参数 *j*
输入：新配置参数 *i*, 当前迭代次数 iteration, 最大迭代次数 max_iterations, 最小值 a_min 和最大值 a_max
输出：经过改造的新配置参数 *i*
For 每个参数 = 1:length do:
生成随机数 *r*, *f*, *g*
If *r* < 0.5
新配置参数 *i* =新配置参数 *i*+ (新配置参数 *i*- a_min) * *f* * (1 - iteration/max_iterations)^2;
Else
新配置参数 *i* =新配置参数 *i*+ (a_max -新配置参数 *i*) * *f* * (1 - iteration/max_iterations)^2;
End If
End For
For 每次迭代 do:
For *i* = 1:length(selected_configurations)/2
配置参数 *i* = 被选择的配置参数 (*i**2-1);
配置参数 *j* = 被选择的配置参数 (*j**2);
b = rand(); % 用于重组的随机数 *b*
If rand() < real_crossover_rate
[新配置参数 *i*, 新配置参数 *j*] = performRealCrossover(新配置参数 *i*, 配置参数 *j*, *b*);
更新配置参数= [更新配置参数, 新配置参数 *i*, 新配置参数 *j*];
End If
End For
End For
For *j* = 1:length(new_generation)
新配置参数= 更新配置参 *j*;
If rand() < real_mutation_rate
新配置参数= performRealMutation(新配置参数, iteration, max_iterations, a_min, a_max);
更新配置参 *j* = 新配置参数;
End If
End For

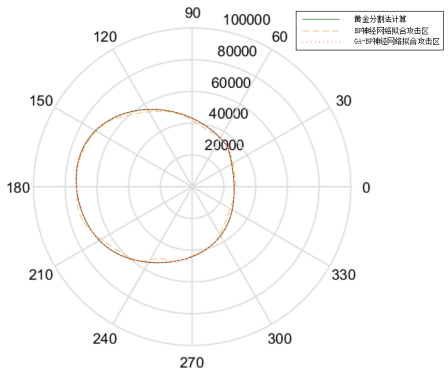
(2) 仿真结果与分析

本节同样设定了不同的初始条件，采用黄金分割法模拟出理论导弹攻击区数据，在

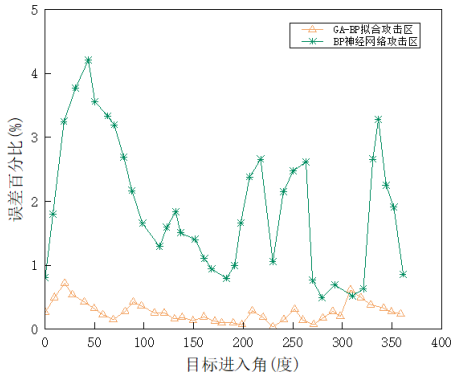
仿真设置中，我机飞行角度和目标进入角度的变化范围为$[0°,360°]$，我机和目标的飞行高度的变化范围为$2000m$到$4000m$，我机和目标飞行速度变化范围为$100m/s$到$450m/s$。总共生成了 232400 条数据作为训练集。

设置 BP 神经网络的训练次数不超过 800 次，学习速率为 0.01，训练过程中误差不大于 0.0001，初始参数规模$N=10$，最大迭代数$G_{\max}\le 30$，重组概率和改造概率分别为 0.8 和 0.2，基于 4.2.4 节的模型进行仿真。

以最大攻击区为例，当本次实验的初始条件为：我机飞行速度$v_u=150m/s$、飞行高度$h_u=2000m$，目标速度$v_t=200m/s$、高度$h_t=2000m$时，两种算法拟合出的导弹攻击区如图 4-8(a)所示，TBPNN 算法计算出的攻击区域所需时间为 0.18 秒，其与黄金分割法的结果的误差分布如图 4-8(b)所示。



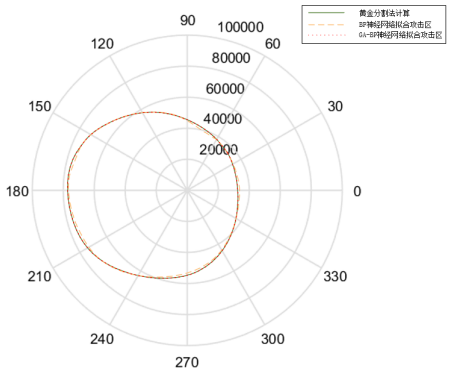
(a) TBPNN 拟合攻击区仿真对比图(1)



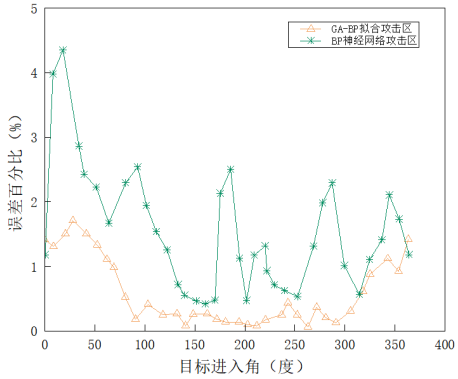
(b) 不同算法拟合出的误差曲线(1)

图 4-8 TBPNN 仿真结果图(1)

当我机飞行速度$v_u=300m/s$、飞行高度$h_u=4000m$，目标速度$v_t=250m/s$、高度$h_t=3500m$时，两种算法拟合出的导弹攻击区如图 4-9(a)所示，TBPNN 算法计算出的攻击区域所需时间为 0.23 秒，其与黄金分割法的结果的误差分布如图 4-9(b)所示。



(a) TBPNN 拟合攻击区仿真对比图(2)



(b) 不同算法拟合出的误差曲线(2)

图 4-9 TBPNN 仿真结果图(2)

仿真结果表明，TBPNN 拟经网络更接近期望值，其误差百分比均在 2%以内，准确

度显著高于 BP 神经网络；在解算时间方面，优化后的 BP 神经网络的解算时间均低于 0.3 秒，大幅提升了计算效率。因此，利用 TBPNN 来拟合攻击区，能够有效地满足未来无人机编队在空中协同作战战术决策中对实时性和准确性的需求。

4.3 基于改进 SAC 算法的无人机空战机动决策方法

4.3.1 状态空间

无人机机动决策模型的状态空间用于描述空中战斗的全部情况，该情况在任何时候都可以通过无人机状态来确定，无人机状态包含位置矢量 $\mathbf{R}=[x,y]$、速度 v、航向角 φ、相对方位角 q 和距离 d 中的相对信息。为了统一每个变量的范围并提高网络学习的效率，每个状态变量都被归一化为 $[0,1]$。状态空间定义为一个 6 元组 $S=[x,y,v,\varphi,d,q]$。具体空战态势示意图如图 4-10 所示。

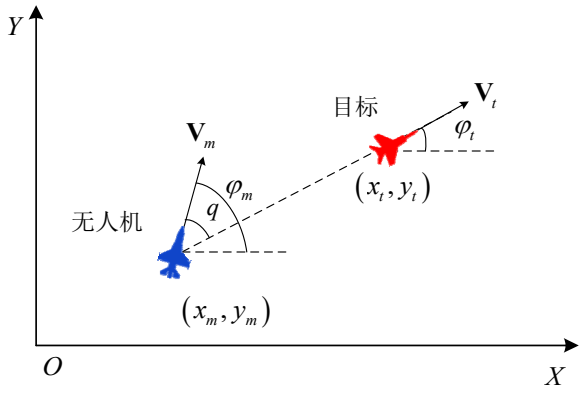


图 4-10 空战态势图

4.3.2 动作空间

从无人机的数学模型可以看出，在适当的积分步长 dt 内定义单位速度 dv 和单位航向角 $d\varphi$，能够使无人机在二维空间执行连续的机动动作，由此，可以构建其动作空间：$A=[dv,d\varphi]$。

4.3.3 奖励函数

在空中战斗中，无人机通过不断估计奖励完成相应的动作，以使敌方无人机进入导弹的攻击区域。战斗中的双方在 OXY 坐标系中建模，设定我机的速度矢量为 $\mathbf{V}_m=(v_{xm},v_{ym})$、位置矢量为 $\mathbf{L}_m=(x_m,y_m)$。敌机的速度矢量为 $\mathbf{V}_t=(v_{xt},v_{yt})$、位置矢量为 $\mathbf{L}_t=(x_t,y_t)$。双方的相对位置由 $\mathbf{D}=\mathbf{L_m}-\mathbf{L_t}$ 表示，双方的距离为 $d=\|\mathbf{D}\|_2$。$\mathbf{V_m}$ 和 $\mathbf{D}$ 之间的角度为相对方位角 q，目标速度矢量与方向矢量的夹角为 q_t。无人机空战示意图如图 4-11 所示。

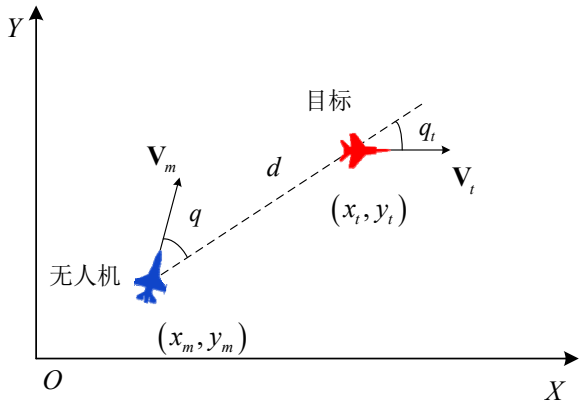


图 4-11 无人机空战示意图

机动决策的目标是无人机在空战中进行机动，使敌方无人机进入导弹的攻击区域。为了成功攻击敌人，我机应该尽量朝向敌人飞行，其相对方位角 q 应该最小化。随后定义无人机在空中战斗中的角度优势为：

$$\eta_A = -q/180^{\circ} \tag{4-15}$$

其中，$q = \arccos\left(\left(\mathbf{D}\times\mathbf{V}_m\right)/\left(\|\mathbf{D}\|_2 \cdot \|\mathbf{V}_m\|_2\right)\right)$。$\eta_A$ 越小表示无人机在空中战斗中的优势越大。

距离在空中战斗中也是一个重要因素。导弹无法攻击超出其攻击区，因此在解算出最大攻击距离 $D_{\max}$ 和最小攻击距离 $D_{\min}$ 后无人机在空中战斗中的距离优势定义为：

$$\eta_D = -\frac{d}{\left(5 \cdot D_{\max}\right)} + \delta \tag{4-16}$$

$$\delta = \begin{cases} -1, & if\ d < D_{\min} \\ 0, & else \end{cases} \tag{4-17}$$

结合角度优势和距离优势，无人机在空中战斗中的优势评估函数定义为：

$$\eta = \eta_A + \eta_D \tag{4-18}$$

本论文基于无人机在空战中的优势评估函数 η 设计了奖励函数，以无人机相对于敌机的战术位置和能力优势为依据，通过连续的负反馈激励无人机保持或提升其战术地位。在无人机处于明显优势的情况下，奖励函数会提供更大的正向激励，鼓励无人机继续执行和优化当前的机动决策。同时，为避免不必要的战术风险，当无人机越界时，会施加相应的惩罚。奖励函数是对智能体在当前空战情况下行动的实时评估，它是基于无人机在空战中的优势评估函数设计的，该函数是一个负的连续函数，同时，当我机在空战中处于较大优势时，它会获得更大的激励以保留已经探索的策略。设 $q_{\max}$ 为最大方位角，定义机动决策函数如下：

$$\eta_{ai} = \begin{cases} 3, & |q| < q_{\max} \text{ 或 } D_{\min} < |d| < D_{\max} \\ 0, & \text{其他} \end{cases} \tag{4-19}$$

设置我机超越环境边界时给予惩罚奖励：

$$r_{m,b}=\begin{cases}-5, & \text{我机超越边界}\\ 0, & \text{我机未超越边界}\end{cases} \tag{4-20}$$

则总奖励为：

$$r=\eta+\eta_{ai}+r_{m,b} \tag{4-21}$$

其中，η为无人机在空中战斗中的优势评估函数见式(4-18)。

发射空对空导弹之前需要锁定目标的一定时间$t_{\max}$。假设敌人连续处于导弹的攻击区域内的时间为t_{in}，当满足方程(4-22)时，便发射导弹。

$$\begin{cases}D_{\min}<|d|<D_{\max}\\ q<q_{\max}\\ t_{in}>t_{\max}\end{cases} \tag{4-22}$$

4.3.4 E-SAC 算法

SAC 算法(Soft Actor-Critic, SAC)[194]是建立在 Actor-Critic 架构上的一种前沿的强化学习方法。“Actor”负责根据当前的环境状态选择动作；“Critic”负责评估在给定状态下采取特定动作的好坏，这种架构使模型能够同时学习最佳策略（即应采取哪些动作）和评估这些策略的效果（即这些动作的效果如何）。

SAC 算法作为一种深度强化学习算法，在处理连续动作空间和高维状态空间的问题方面表现出色，但也存在一些局限性，如训练收敛速度慢和对超参数的高敏感性。为了解决这些问题，本文提出了一种改进的 SAC 算法，称为 E-SAC(Expert-Soft Actor-Critic)算法。E-SAC 算法通过引入 Expert Actor 来增强智能体的探索效率，并优化学习过程。

E-SAC 算法可以对空战中的无人机自主机动决策模型进行构建。它是一个带有策略网络π_θ、两个 soft-Q 网络Q_{φ_1}和Q_{φ_2}以及两个目标 soft-Q 网络Q'_{φ_1}和Q'_{φ_2}的演员-评论家架构，其中θ，φ_1，φ_2，φ'_1和φ'_2分别对应网络的参数。

在 E-SAC 算法中，策略$\pi_\theta(\cdot|s_t)$遵循特定的分布，策略的随机性由策略的熵$H\left(\pi_\vartheta(\cdot|s_t)\right)$进行评估。为了平衡最优策略的探索和策略探索性的最大化，策略熵融入于回报函数之中。期望的累积回报定义如下：

$$J(\pi)=\sum_{t=0}^{T}E_{(s_t,a_t)\sim\rho_\pi}\left[r(s_t,a_t)+\alpha H\left(\pi(\cdot|s_t)\right)\right] \tag{4-23}$$

其中，α是熵正则化系数，表示回报中熵的比例，控制策略的随机性，$r(s_t,a_t)$为状态s_t下动作a_t的奖励值。

动作由策略网络π_θ和噪声生成。噪声τ从标准正态分布中采样。因此，动作a_t的生成过程如方程(4-24)至(4-25)所示。

$$\mu,\sigma=\pi_\theta(s_t) \tag{4-24}$$

$$a_t=\tanh(\mu+\tau*\exp(\sigma)) \tag{4-25}$$

其中，μ和σ分别代表策略网络π_θ输出动作的平均值和标准差。

Soft-Q 网络的输入是状态s_t、动作a_t、折扣率为γ，输出是Q_{soft}。Soft-Q 函数定义如下：

$$\begin{aligned}Q_{\text{soft}}(s_t,a_t)&=E_{(s_t,a_t)}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)+\alpha\sum_{t=1}^{\infty}\gamma^t H\left(\pi\left(\cdot|\ s_t\right)\right)\right]\\&=r(s_t,a_t)+\gamma E_{(s_{t+1},a_{t+1})}\times\left[Q_{\text{soft}}(s_{t+1},a_{t+1})-\alpha\log\left(\pi\left(a_{t+1}|\ s_{t+1}\right)\right)\right]\end{aligned} \tag{4-26}$$

该算法首先通过改变初始的空战场景设置不同的 Expert Actor 采样环境，以获取 Expert 经验样本。在某个环境中，无人机依据当前状态使用 Expert Actor 做出策略决策，获得相应的行动a_t，并通过执行该行动获取奖励r_t以及更新后的状态s_{t+1}，此过程持续直到成功击败敌机。在不同的空战环境中重复此过程，并将收集到的 Expert 经验样本(s_t,r_t,s_{t+1},a_t)储存于 Expert 经验回放池中。从经验池中随机抽取样本对策略和 Soft-Q 网络进行训练。

策略网络是根据损失函数$J_\pi(\theta)$进行更新的，如式(4-27)所示：

$$J_\pi(\theta)=E_{s_t\sim R,a_t\sim\pi_\theta}\left[\log\pi_\theta\left(a_t|\ s_t\right)-Q_\varphi(s_t,a_t)\right] \tag{4-27}$$

Sotf-Q 网络是根据损失函数$J_Q(\varphi_i)$进行更新的，如式(4-28)所示：

$$\begin{aligned}J_Q(\varphi_i)=E_{(s_t,a_t,s_{t+1})\sim R,a_{t+1}\sim\pi_\theta}\Bigg[&\frac{1}{2}Q_{\varphi_i}(s_t,a_t)-\Big(r(s_t,a_t)\\&+\gamma\left(Q_{\varphi'}(s_{t+1},a_{t+1})-\alpha\log\pi_\theta\left(a_{t+1}|\ s_{t+1}\right)\right)\Big)^2\Bigg]\end{aligned} \tag{4-28}$$

其中，$Q_{\varphi'}(s_{t+1},a_{t+1})=\min\left(Q_{\varphi_1'}(s_{t+1},a_{t+1}),Q_{\varphi_2'}(s_{t+1},a_{t+1})\right)$。

根据训练过程动态调整α，具体如下：

$$J(\alpha)=E\left[-\alpha\log\pi_t\left(a_t|\ \pi_t\right)-\alpha H_0\right] \tag{4-29}$$

其中，H_0是目标熵。

E-SAC 算法在训练初期通过将一定比例的 Expert Actor 经验样本混合到算法的经验回放缓冲区中，丰富了样本多样性，有助于指导智能体进行更全面的环境探索。为了减少专家演员在智能体后期决策过程中可能产生的不利影响，Expert Actor 经验样本与通过标准 SAC 算法探索获得的样本之间的比例会动态调整。通过这种方式，E-SAC 算法不仅了智能体的深度探索，同时加快了训练进程。E-SAC 算法的结构如图 4-12 所示。

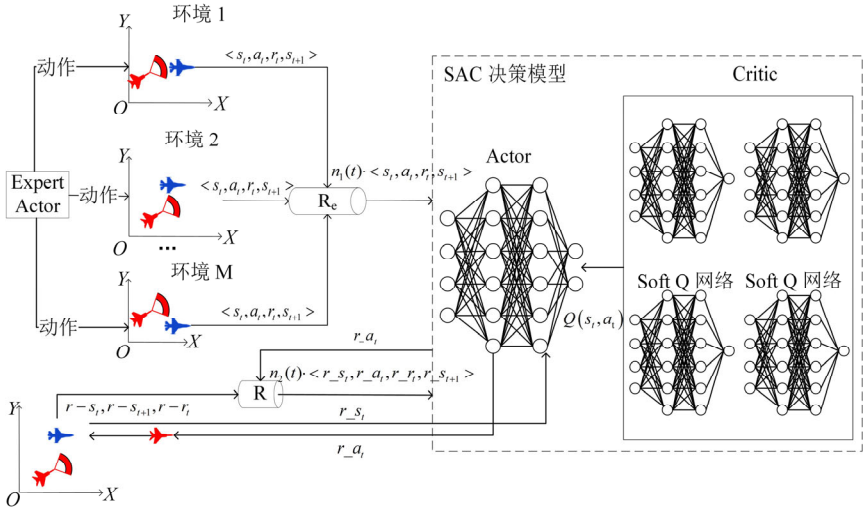


图 4-12 E-SAC 算法示意图

E-SAC 算法通过提高训练样本的多样性，不仅引导了智能体的深度探索，还有效预防了算法早熟收敛于局部最优解，同时显著加快了训练进程。该算法的训练流程详如表 4-4 所示。

表 4-4 基于 E-SAC 的无人机自主机动决策算法伪代码

算法: E-SAC	
1:	初始化 E-SAC 算法的相关网络参数、空战环境、回放缓冲区 R 、batch 大小(batch_size)、以 E-SAC 算法开始做出决策的步骤数(start_size)
2:	For episode = 1, N do
3:	获取无人机的初始状态 s_t
4:	For t = 1, T do:
5:	If 放入缓冲区 R 的长度小于 start_size
6:	随机选择一个动作 a_t
7:	Else 使用策略 $\pi_\theta(s_{a_t})$ 来选择动作
8:	无人机执行动作 a_t ，获取奖励和新状态 s_{t+1}
9:	将 (s_t, r_t, s_{t+1}, a_t) 存放到缓冲区 R 中
10:	If 回放缓冲区 R 的长度大于 batch_size：
11:	If 回放缓冲区 R 的长度小于 start_size
12:	从专家经验回放缓冲区 R 中取出 n 组样本
13:	Else 从回放缓冲区 R 中取出 n_2 组样本与 n_1 合并
14:	End If
15:	If 回放缓冲区 R 的长度是偶数
16:	$n_1 = n_1 - 1$
17:	End If
18:	更新 E-SAC 算法的神经网络
19:	End If
20:	End If
21:	End For
22:	End For

关于 Expert-Actor 模型，其建立方法如下：当我机发射导弹攻击敌机时，相对方位角 q 和相对距离 d 应同时满足发射条件。当采用 Expert 方法控制无人机运动时，首先可

以将q减少到一定范围内，以使我机处于面对敌机的正面或尾部攻击位置，然后，我机增加速度以迎接敌人，直到敌人进入之前所计算好的导弹攻击区范围内。基于我机运动模型的 Expert-Actor 的数学模型如式(4-30)-(4-34)所示：

$$\begin{cases}\Delta X = x_t - x_m \\ \Delta Y = y_t - y_m\end{cases} \tag{4-30}$$

$$D_\varphi = \begin{cases}180^\circ + \arctan\left(\dfrac{\Delta X}{\Delta Y}\right)/\pi * 180^\circ, & \Delta Y < 0, \Delta X > 0 \\ \arctan\left(\dfrac{\Delta X}{\Delta Y}\right)/\pi * 180^\circ - 180^\circ, & \Delta Y < 0, \Delta X < 0 \\ \arctan\left(\dfrac{\Delta X}{\Delta Y}\right)/\pi * 180^\circ, & \text{else}\end{cases} \tag{4-31}$$

其中，ΔX、ΔY 分别代表敌人相对于无人机的位置向量分量，D_φ代表两边相对位置向量**D**与 X 轴之间的角度。

设$\Delta v = v_{t+1} - v_t$、$\Delta\varphi = D_\varphi - \varphi$，设我机速度和航向角变化需要在以下范围内进行控制。

$$\begin{cases}-\Delta v_0 \le \Delta v \le \Delta v_0 \\ -\Delta\varphi_0 \le \Delta\varphi \le \Delta\varphi_0\end{cases} \tag{4-32}$$

无人机速度变化dv和航向角变化$d\varphi$分别可以通过式(4-33)和式(4-34)算出。

$$dv = \begin{cases}0, & q > 30^\circ \\ \Delta v_0, & q < 30^\circ \cap d > D_{\max} \cap v < v_{\max} \\ v_{\max} - v, & q < 30^\circ \cap d > D_{\max} \cap v > v_{\max} \\ -\Delta v_0, & q < 30^\circ \cap d < D_{\max} \cap v < -\Delta v_0 \\ \Delta v & q < 30^\circ \cap d < D_{\max} \cap -\Delta v_0 < v < \Delta v_0 \\ \Delta v_0 & q < 30^\circ \cap d < D_{\max} \cap v > \Delta v_0\end{cases} \tag{4-33}$$

$$d\varphi = \begin{cases}-\Delta\varphi_0, & \Delta\varphi < -\Delta\varphi_0 \\ \Delta\varphi, & -\Delta\varphi_0 < \Delta\varphi < \Delta\varphi_0 \\ \Delta\varphi_0, & \Delta\varphi > \Delta\varphi_0\end{cases} \tag{4-34}$$

其中，$v_{\max}$是我机的最大速度。

4.4 仿真结果与分析

4.4.1 实验环境设定

本次仿真采用 Python 语言，开发环境为 PyCharm，训练环境采取 1v1 方式。输入状态维度为 2，动作维度为 1，操作系统为 Windows 10 系统，仿真环境基于 MaCA 平台。

本文基于 MaCA 环境实现无人机智能作战算法的仿真，搭建仿真环境逻辑和运行流程如图 4-13 所示。

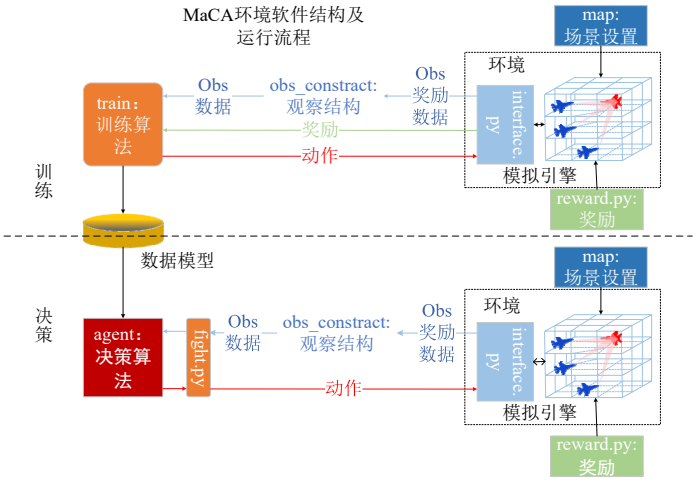


图 4-13 MaCA 环境软件架构及运行流程

设置地图尺寸12000×12000米，敌机策略采用 MaCA 平台中的 fix_rule 固有策略，我机采用本文建立的 E-SAC 算法，具体参数配置如表 4-5 所示。

表 4-5 无人机空战机动决策仿真参数表

参数名称	参数大小
最大离轴发射角 $q_{\max}$	30°
目标锁定时间 $t_{\max}$	$2s$
无人机最大速度 $v_{\max}$	$260\,m/s$
无人机最大变化速度 Δv_0	$20\,m/s$
网络隐含层个数	2
网络隐含层神经元个数	256
学习速率	0.0003
累积回报折扣因子	0.99
目标熵 H_0	–3
熵的正则化系数初始值	1
目标网络更新间隔步数	300
经验池容量	100,000
经验池提取样本数量	200
激活函数	$ReLU$
优化器 Adam 软更新系数	0.005

利用 4.2 节的方法对无人机空战攻击区实时解算并将解算结果传输至机动决策模型。为检验 E-SAC 算法在自主完成机动决策任务方面的有效性，本节设计了 4 种不同的模拟环境，以研究我机与敌机的初始距离和初始相对方位角的改变对不同算法训练过程的

影响。其中，在环境 1 和环境 2 中，我机与敌机有相同的初始距离，但初始相对方位角不同，在环境 1 和环境 3 中，我机与敌机有不同的初始距离，但初始相对方位角相同，在环境 3 和环境 4 中，我机与敌机有不同的初始距离，但初始相对方位角相同。基于设设置的环境 1-4 进行分析，对比不同初始距离和不同初始方位角场景下对近距无人机机动决策的影响。具体模拟环境的空战初始态势如表 4-6 所示。

表 4-6 空战初始态势

环境编号	初始距离 (m)	初始相对方位角 (°)
1	6000	30
2	6000	60
3	9000	30
4	9000	60

根据 4.3 节的模型，在 4 个环境中分别采用了 SAC 算法和增加了专家演员的 E-SAC 算法对无人机机动决策进行训练。在 E-SAC 算法的应用中，Expert Actor 为 SAC 算法提供了从第 1 到 256 个训练回合的 Expert 样本，并逐渐降低了这些样本在总样本中的比例，直到在 256 个训练回合后完全停止使用 Expert 样本。

4.4.2 训练结果与分析

为了更直观地对比两种算法的性能，本文设计了四种不同的环境，让我机（红色无人机）在环境 1 到环境 4 中进行训练，并详细记录了双方的战斗过程。作为对照，敌机（蓝色无人机）遵循 MaCA 平台中封装好的 fix_rule 算法策略进行机动，以提供一致的比较基准。通过这些实验设计和记录，分析和评估各种算法在复杂空战环境中的性能和适应性。

基于如式(4-20)-(4-21)所示设置的奖励函数，分别利用 SAC 和 E-SAC 算法训练无人机机动决策的测试结果如表 4-7 至表 4-10 所示。

表 4-7 环境 1 训练结果

环境 1	SAC	E-SAC
最大奖励	518.36	584.34
奖励收敛回合数	384	365

表 4-8 环境 2 训练结果

环境 2	SAC	E-SAC
最大奖励	392.33	498.78
奖励收敛回合数	466	406

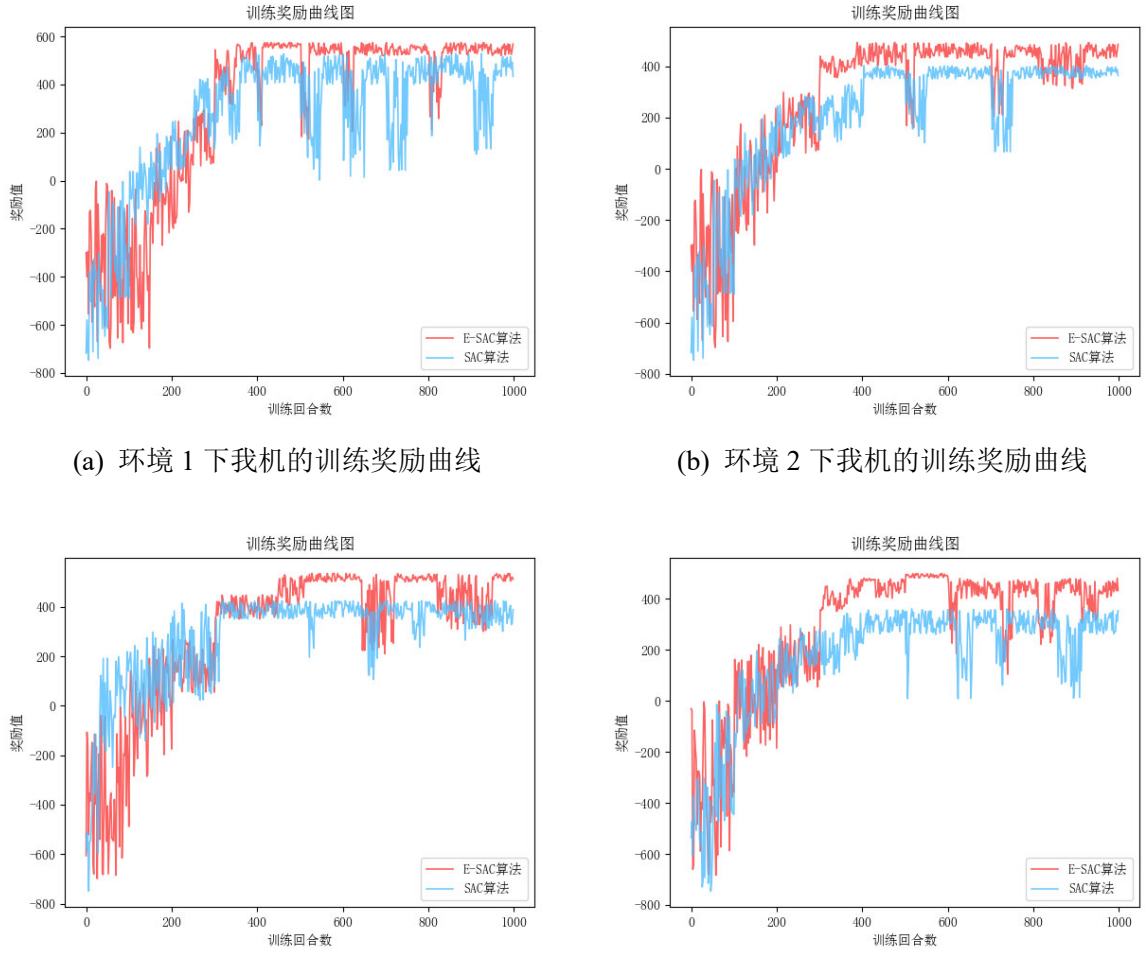
表 4-9 环境 3 训练结果

环境 3	SAC	E-SAC
最大奖励	422.25	522.39
奖励收敛回合数	319	531

表 4-10 环境 4 训练结果

环境 4	SAC	E-SAC
最大奖励	329.18	479.01
奖励收敛回合数	358.1	404

两种算法的训练奖励曲线图如图 4-14 所示。



(a) 环境 1 下我机的训练奖励曲线　　(b) 环境 2 下我机的训练奖励曲线

(c) 环境 3 下我机的训练奖励曲线　　(d)环境 4 下我机的训练奖励曲线

图 4-14 训练奖励曲线

由实验结果可知，两种算法均可以在环境 1、环境 2、环境 3 和环境 4 中完成近距空战中对目标的追踪与打击的机动决策训练任务。

将两种算法的训练结果在不同的环境下进行对比，在环境 1 实验结果中，两种算法

训练的奖励值较接近且具有相同的收敛趋势，而相较之传统的 SAC 算法，本文所提的 E-SAC 算法训练速度较快，提前了 19 个回合收敛，且 SAC 算法训练奖励函数曲线波动较大，表明了算法的不稳定性；在环境 2 中，本文所提的 E-SAC 算法最大奖励值优于传统的 SAC 算法，且 E-SAC 算法收敛更快，提前 60 个回合就达到了最大回报；环境 3 中的实验结果表明 SAC 算法在该环境下陷入局部最优解，只获得了 422.25 的奖励，而 E-SAC 算法获得了 522.39 的奖励；环境 4 中的我机与敌机的初始距离和相对方位角较大，SAC 算法只获得了 329.18 的奖励，而 E-SAC 算法在经过 404 个回合的训练后，获得了 479.01 的奖励。

综上分析可知，E-SAC 算法不仅可以获得更高的奖励，还能够用更少的步骤完成任务，即本文所提的 E-SAC 算法针对无人机机动决策在训练效率、收敛速度和奖励方面比传统的 SAC 算法有显著提升。

下面对环境 1 和 4 中的仿真结果进行展示。首先，在环境 1 中，红方与蓝方的初始相对方位角为 30 度，初始距离为 6000 米。提取 MaCA 平台中的无人机的轨迹数据，将环境 1 的近距空战过程中无人机机动决策过程绘制到地图中，其示意图如图 4-15 所示。

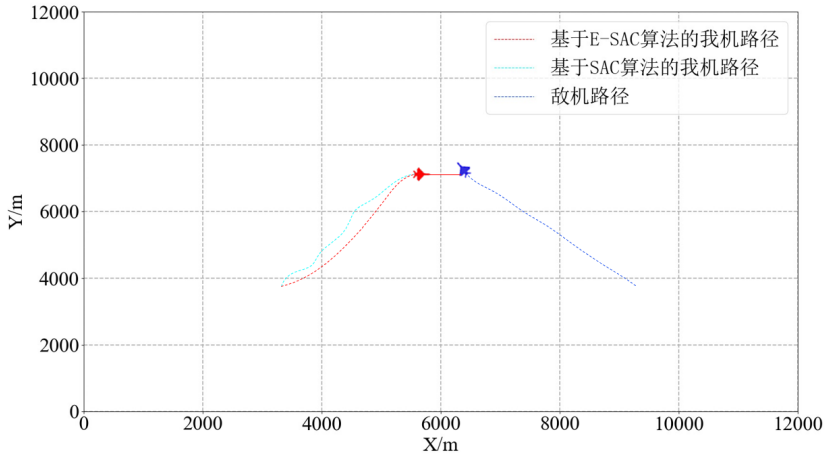


图 4-15 环境 1 中无人机一对一机动决策训练仿真过程

如图 4-15 所示，E-SAC 算法和 SAC 算法在决策过程中展现出相近的机动决策过程，可以看出我方无人机一开始便迅速调整航向，降低与敌机的相对方位角，以尾追攻击姿态使得我机进入之前所计算出的导弹攻击区，然后，迅速调整速度以缩短与敌机的距离。E-SAC 算法在减小相对方位角方面表现出了更高的稳定性和效率，显著优于传统的 SAC 算法，实验结果证明了 E-SAC 算法在训练机动决策时的有效性。

在环境 4 中，设置敌我双方的初始距离增加为 9000 米，角度增加为 60 度。提取 MaCA 平台中的无人机的轨迹数据，环境 4 的近距空战过程中无人机机动决策示意图如图 4-16 所示。

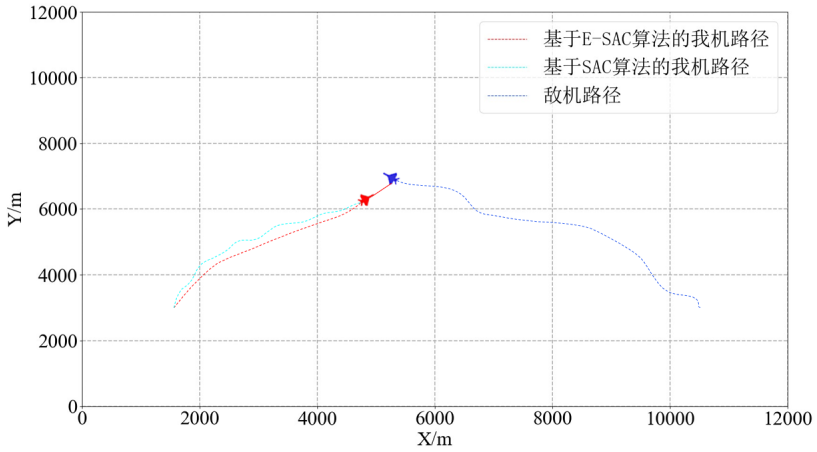


图 4-16 环境 4 中无人机一对一机动决策训练仿真过程

如图 4-16 所示，E-SAC 算法和 SAC 算法的决策过程均是首先改变方向，减少相对方位角，然后缩短距离，并最终使到达攻击区范围内。仿真结果表明，两种算法均能进行对敌机追踪与打击的机动决策。相对而言，本文所提的 E-SAC 算法的无人机相对距离和方位角的变化较为稳定。

综上所述，E-SAC 算法可以增强探索过程，并有效处理 SAC 算法可能面临的局部收敛问题，从而快速获得无人机在空中作战过程中的最佳机动决策策略。

4.5 本章小结

本章针对无人机进入作战区域后的机动决策进行研究，解决了无人机近距空战机动决策问题，为下一章无人机编队协同策略模型提供可靠的机动决策元策略模型。首先，根据攻击区解算的相关模型，采用 BP 神经网络对攻击区进行解算；其次，将解算出的导弹攻击区作为导弹终端约束，设计了基于 E-SAC 算法的无人机编队自主机动决策方法；最后，仿真结果表明，本文所提出的 E-SAC 算法能够很好地引导无人机编队进行近距空战机动决策以追踪和击打敌机。

第 5 章 基于分层强化学习的编队协同策略模型

本章以分层强化学习为研究理论工具，研究了无人机编队在整个空战过程中的战术决策方法。首先，设计了编队协同空战过程元策略模型；其次，建立了基于分层强化学习算法的编队协同策略模型；最后，对无人机编队的空战战术协同策略模型进行仿真，验证算法的有效性。

5.1 问题分析

无人机编队协同策略是指无人机编队在空中作战时的决策选择和机动协调，需要设计合理的探测与攻击策略以实现高效的编队协同作战能力，从而提高无人机编队的任务执行效率和生存能力。

本章采用了分层强化学习算法来研究多无人机空中战斗中无人机编队的协同策略模型。首先，对编队协同决策元策略模型进行设计，将第四章所设计的机动决策模型嵌入为编队协同策略模型中的机动决策元策略模型，并设计基于专家经验建立了雷达开关、主动干扰、主动探测和队形转换模型；其次，对分层强化学习中的 Option-Critic 算法进行改进，并建立合适的状态空间和动作空间；最后，对整个基于分层强化学习的编队协同智能空战策略模型进行仿真验证。本章研究内容如图 5-1 所示。

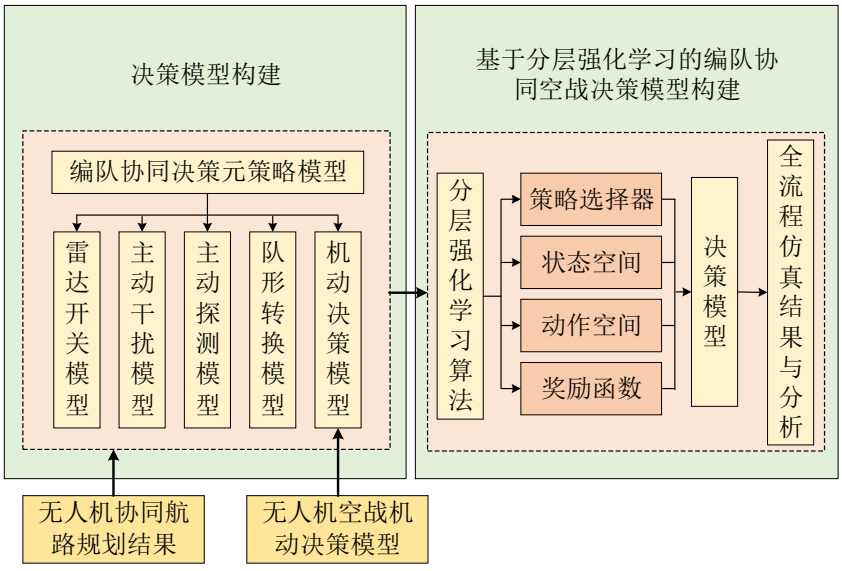


图 5-1 基于分层强化学习的编队协同策略模型研究内容

5.2 编队协同决策元策略模型

本文将无人机编队协同策略模型中的元策略定义为雷达开关策略、主动干扰策略、主动探测策略、队形转换策略和机动决策策略这五个部分，无人机编队协同空战策略模型的分层结构如图 5-2 所示。

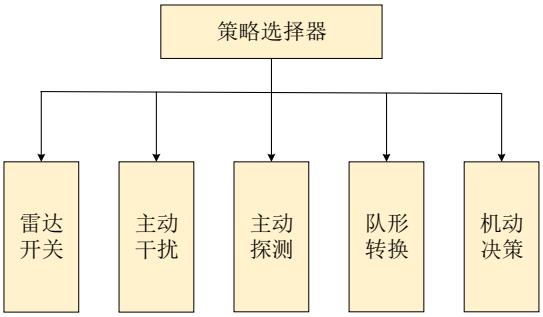


图 5-2 无人机编队协同空战策略模型的分层结构

如图 5-2 所示，无人机编队的协同空战策略模型按层级组织，其中策略选择器层扮演着决策选择的角色，其主要职责在于筛选出适宜的元策略。该层在编队建立之初和需要重新组织编队时，负责选取队形转换策略以调整战场中我机编队的队形；当雷达还未侦测到任何目标时，实施主动探测策略以进行分布式搜寻；在搜寻阶段，选择雷达开关策略以实现雷达资源的高效配置；目标一经侦测到，便通过机动决策策略追踪目标，确保在敌机进入攻击范围时实施打击，并在追踪过程中使用主动干扰策略来扰乱敌机雷达。

在分层强化学习的框架下，每个层级的策略可以独立学习并优化。顶层模型可以在更广泛的时间尺度上进行决策，而底层元策略模型则聚焦于更短时间尺度的决策。

5.2.1 雷达开关模型

雷达开关策略是无人机空战中雷达的使用策略[195]，关键在于决定开启雷达进行主动探测以及关闭雷达以保持隐蔽的时机。这一策略的核心目的是在确保有效地进行探测和信息收集的同时，最大限度地降低被敌方电子侦察系统发现的风险。

为了优化雷达的使用并减少资源消耗，同时降低飞机因雷达活动而被敌方探测到的潜在风险。本文提出了一种雷达开关策略模型，该模型重点分析雷达探测的重叠区域，以此为依据来决定雷达的开启或关闭。如图 5-3 所示，其中 θ 代表雷达的探测半角，r 为雷达探测距离。

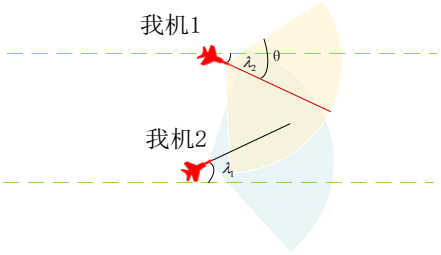


图 5-3 雷达探测重叠区域分析

当两架飞机之间的距离 d 以及航向角 λ_1 和 λ_2 满足如式(5-1)所示的标准时，它们被认为已进入所定义的“决策区”：

$$\begin{cases} d \leqslant r\sin\theta \\ \lambda_1 - \lambda_2 \leqslant \theta \end{cases} \tag{5-1}$$

在这种情况下，其中一架飞机的雷达会被关闭，以减少电磁信号的重叠和降低位置暴露风险。

5.2.2 主动干扰模型

主动干扰策略[196]在于调整我机的干扰设备，使其工作频点与敌方雷达的频点相同或接近。这种策略能够有效干扰敌方雷达的正常运作，从而影响其探测和追踪能力。为了在电子战中有效地对敌机实施干扰，本文构建了一个主动干扰模型，目的是提升干扰的精度和效果。模型实施的具体步骤包括：

首先，在干扰操作开始前，需要确定被干扰目标的雷达频点(r_t)，并将此频点作为模型的输入参数。

目标雷达开机频点的观测信息需具备以下两个前提：

1) 目标必须位于我机雷达的有效探测范围之内。如果目标超出此范围，我机雷达则无法捕获到目标雷达的信息。经 MaCA 平台中主动干扰区域的测量结果可知，干扰的最大有效距离达到 300m，干扰的角度范围限定在-5°到 5°。

2) 目标雷达不应受到外部干扰。如果目标雷达受到电子对抗等干扰，那么我机雷达捕获的频点信息可能会不准确或无法获取。

在获取必要信息后，我机设置对应干扰频点。这一过程可用以下等式表示：

$$r_t = r_j \tag{5-2}$$

其中，r_j 是我机的干扰频点。

模型根据是否成功探测到目标雷达的频点，将产生不同的奖励值。如果未探测到任何目标雷达频点，奖励值设为 0；如果探测到 n 个目标的干扰频点，奖励值则设置为 n。

该模型的输入参数为敌机雷达的频点，输出为我机实施干扰的雷达频点。

5.2.3 主动探测模型

雷达启动后进入目标搜索阶段，在此阶段，雷达的主要任务是探测潜在目标。一旦探测到目标，雷达系统便进入目标确认过程，即对探测到的目标进行持续监测以确保其真实性[197]。目标一旦被确认，雷达随即启动对目标的持续跟踪模式，持续监视直至完成对目标的打击任务。这个过程标志着雷达对目标的完整作战周期，如图 5-4 所示，从初始探测到最终的目标打击，展现了雷达系统在空战中的关键作用和战术价值。本文的主动探测策略主要针对发现敌机的过程，之后会开启对目标的持续锁定，即使切换到其他策略，雷达依然可以获取到目标的位置。

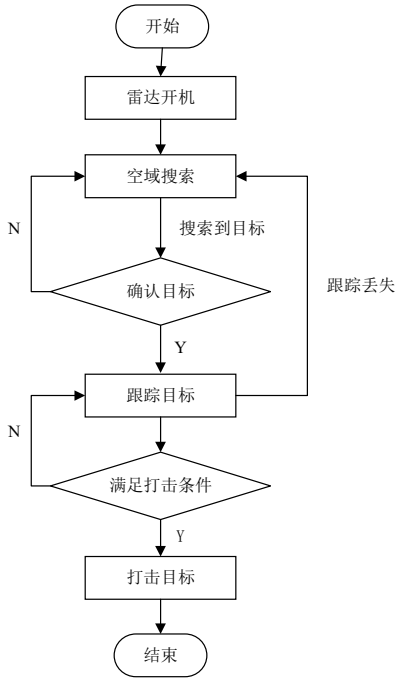


图 5-4 机载雷达作战过程示意图

开启主动探测策略以发现目标，设置时间步长τ_1实现最大搜索面积。为了提升无人机编队分布式搜索的效率，引入了人工势场法以维持我机之间的安全间距，如式(5-3)所示。定义$\rho(q)$为我机q到其他友机q'边界$\mathcal{QO}$的距离：

$$\rho(q)=\min_{q'\in\partial\mathcal{QO}}\|q-q'\| \tag{5-3}$$

其中，$\partial\mathcal{QO}$代表作战区的边界，ρ_0表示威胁区域的影响范围。若我机q与障碍物的距离超过ρ_0，则不会受到排斥力的影响。势场函数的具体形式如式(5-4)所示：

$$U_{rep}(q)=\begin{cases}\dfrac{1}{2}\eta\left(\dfrac{1}{\rho(q)}-\dfrac{1}{\rho_0}\right), & \rho(q)\le\rho_0\\ 0, & \rho(q)>\rho_0\end{cases} \tag{5-4}$$

其中，η表示势函数的比例系数，而斥力则是U_{rep}的负梯度值。当满足条件$\rho(q)\le\rho_0$时，斥力的表达式如式所示：

$$F_{req}(q)=\eta\left(\frac{1}{\rho(q)}-\frac{1}{\rho_0}\right)\frac{1}{\rho^2(q)}\nabla\rho(q) \tag{5-5}$$

当$\mathcal{QO}$是凸函数时，b是$\mathcal{QO}$边界上最接近q的点，则$\rho(q)$如式所示：

$$\rho(q)=\|q-b\| \tag{5-6}$$

其中，$\rho(q)$的梯度如式所示：

$$\nabla\rho(q)=\frac{q-b}{\|q-b\|} \tag{5-7}$$

其中，单位向量从b指向q。

5.2.4 队形转换模型

在初始阶段，我机以两架一组的长机-僚机编队形式进行。长机承担搜索和攻击职责，而僚机负责执行侦查和干扰操作。在长机被击落的情况下，僚机将立即被提升为下一任领航机。长机的标识符设为id_l，僚机的标识符设为id_f。据此，$\{[id_l, id_f]...\}$为建立编队列表和全局编队管理字典，以便在战斗过程中，一旦编队结构遭受破坏，能够迅速进行编队重组。

编队重组的过程包括检查编队列表，以识别和筛选出因战损而不完整的编队。这是通过查找列表中的负标识符来实现的，这些负值表示相应的飞机已被击毁。不完整的编队数量用N表示，而成功重组的编队数量用T标记，即：

$$T = N\%2 \tag{5-8}$$

若存在无法通过重组恢复的编队，其数量则用L来表示，即：

$$L = N - 2T \tag{5-9}$$

根据遍历结果，重组后的编队将重新分配长机或僚机的角色，而那些无法重组的单机则需单独执行战斗任务。

5.2.5 机动决策模型

机动决策模型的训练方法采用第四章基于 E-SAC 算法的无人机空战机动决策的模型，状态空间、动作空间和奖励函数均见 4.3 节。

5.2.6 元策略模型封装

综合上述 5 种编队协同元策略构成了空战全流程作战决策模型，元策略封装流程如图 5-5 所示。

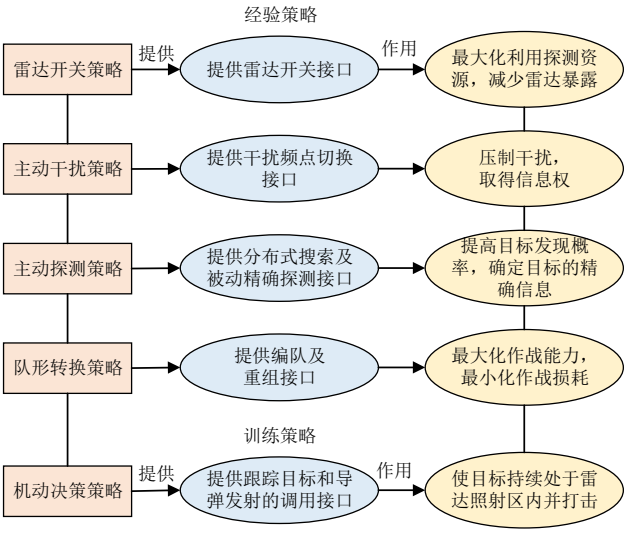


图 5-5 元策略封装结构图

如图 5-5 所示，元策略封装包括雷达开关策略、主动干扰策略、主动探测策略、队形转换策略和机动决策策略 5 种策略。5 种策略的封装内容及功能如下：

雷达开关策略定义一个接口来调控无人机编队中每架机载雷达的激活与关闭，确保无人机能够有效分配主动探测策略的应用，从而降低我机被敌机探测到的风险。此接口应在主动探测阶段被激活。

主动干扰策略通过封装一个干扰频点跟踪的接口，允许系统依据雷达侦测的目标频率自动变换干扰频率。该接口应当在目标被发现后立即调用。

主动探测策略通过设计的接口，使得在空战中能够同步友机资源进行高效搜寻，以获得目标的精确信息。该接口应在空战中编队组建后调用。

队形转换策略通过其封装接口提供了机群编队及编队重建的功能。此接口应在作战的初始阶段及编队受损时被激活。

机动决策策略集成了导弹发射和目标追踪的接口，实现了在雷达探测到目标后自动追踪的能力。当目标被追踪并引导至攻击区域后，将触发导弹发射接口，以完成对目标的打击。

本节中所提的 5 种基本元策略模型在空战全流程中的逻辑关系如图 5-6 所示。

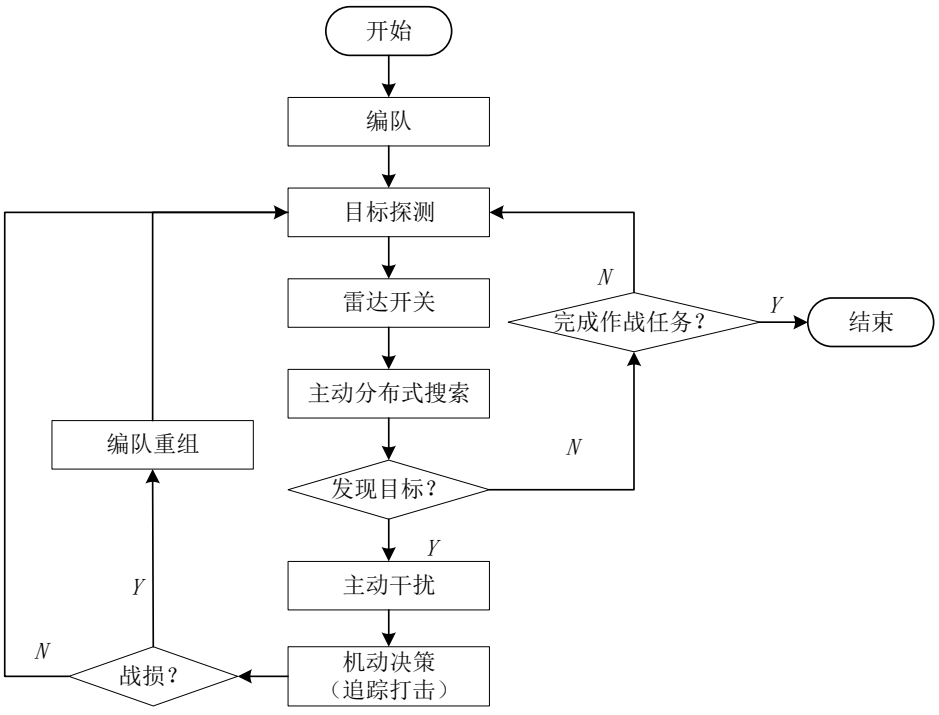


图 5-6 元策略与空战全流程的联系

5.3 改进分层强化学习算法战术决策模型

在分层强化学习算法中，Options 是一个重要的概念。作为一种分层强化学习策略，Options 的核心目的在于引入时间层次和行为层次，以此来应对复杂的决策过程。Options

通过整合单个动作为复杂的操作序列，在处理无人机编队协同空战战术决策时展现出其优势。此外，Options 的引入实现了在时间和空间维度上的抽象化，从而加速了学习过程，并促进了策略在不同任务间的迁移。基于此背景提出了 Option-Critic 架构，该架构在 Options 的基础上进一步发展，提供了一种用于学习和优化 Options 的机制。

5.3.1 元策略决策网络模型

由于经典的 Option-Critic 方法训练得到的元策略往往难以将行动序列细化成符合宏观逻辑与实际经验的策略，并且容易产生对特定行动的过度拟合，从而忽视了行动序列的整体重要性，使得作战结果最终难以达到全局最优，本文提出了改进 Option-Critic 方法，该模型在训练阶段融入了元策略模型，以对终止函数进行训练为主，策略函数则由论文中所设计的专家经验模型和训练模型提供，无需深入探讨底层 Options 策略的训练机制。该模型能够基于给定的环境状态产出相应的行动，避免了采用传统 Option-Critic 方法训练速度慢和难以收敛的不足。

顶层策略选取一个 Option，定义其为$\omega \in \Omega$，其中综合了三个要素：首先，策略$\pi_{\omega}(a|s)$界定了在 Option 中的行动方案；其次，终止条件β规定了在状态s时，退出该 Option 的可能性为$\beta_{\omega}(s)$；最后，初始集I_{ω}明确了 Option 激活所需的起始状态集合。

神经网络能够近似模拟所有 Option 的行动策略及其终止函数，其中行动策略表示为$\pi_{\omega,\theta}(a|s)$，终止函数表示为$\beta_{\omega,\vartheta}(s)$，若$\beta_{\omega,\vartheta}(s)=0$，则控制权仍由当前 Option 保持；若$\beta_{\omega,\vartheta}(s)=1$，则表明该 Option 的任务在当前时刻已经完成，随后控制权将返回至顶层策略。在顶层策略的决定框架中，状态s下采纳某一 Option 的可能性由$\pi_{\Omega}(\omega|s)$量化。据此，界定了在给定状态下选取某一 Option 所能获得的综合效益，包括在状态s选择某 Option 并采取行动后的累计收益，以及在实施某 Option 并达到特定状态后所获得的总收益。

Option 的内部更新主要涉及到各个 Option 的策略$\pi_{\omega,\theta}(a|s)$以及它们的终止函数$\beta_{\omega,\vartheta}(s)$，如式(5-10)所示。

$$\begin{aligned}\mu_{\Omega}\left(s,\omega|\ s_0,\omega_0\right)=&\sum_{t=0}^{\infty}\gamma^t P\left(s_t=s,\omega_t=\omega|\ s_0,\omega_0\right)\\&-\sum_{s',\omega}\mu_{\Omega}\left(s',\omega|\ s_1,\omega_0\right)\sum_{a}\frac{\partial\beta_{\omega,\theta}\left(s'\right)}{\partial\vartheta}A_{\Omega}\left(s',\omega\right)\end{aligned}\tag{5-10}$$

其中，$\mu_{\Omega}(s',\omega|s_1,\omega_0)$表示一个 State-Option 对在轨迹折现中的加权因子，起始于(s_1,ω_0)，$\mu_{\Omega}(s,\omega|s_1,\omega_0)=\sum_{t=0}^{\infty}\gamma^t P(s_{t+1}=s,\omega_t=\omega|s_1,\omega_0)$。首先针对 Critic 中的$Q_U$和$A_{\Omega}$进行学习，随后调整 Option 的参数以促进选项的学习过程。顶层策略运用贪婪策略，主要由$\pi_{\Omega}(\omega|s)$进行描述，通过评估所有 Option 中奖励函数最大的 Option 来选择，即$\max_{\omega}\sum_{a}\pi_{\omega,\theta}(a|s_{t+1})Q_U(s_{t+1},\omega,a)$。

基于 Option-Critic 的无人机战术决策算法伪代码如表 5-1 所示。

表 5-1 基于 Option-Critic 的算法伪代码

算法: Option-Critic 算法伪代码

初始化 $s \leftarrow s_0$

根据选项上的 ε -软策略 $\pi_{\Omega}(s)$ 选择选项 ω

For episode = 1, M do

  根据选项 ω 的策略 $\pi_{\omega,\theta}(a \mid s)$ 选择动作 a

  在状态 s 中采取动作 a，观察新状态 s' 和奖励 r

选项评估

$\delta \leftarrow r - Q_U(s,\omega,a)$

If s' 不是终止状态 Then

  $\delta \leftarrow \delta + \gamma(1-\beta_{\omega,\vartheta}(s'))Q_{\Omega}(s',\omega) + \gamma\beta_{\omega,\vartheta}(s')\max_{\bar{\omega}} Q_{\Omega}(s',\bar{\omega})$

End If

$Q_U(s,\omega,a) \leftarrow Q_U(s,\omega,a) + \alpha\delta$

选项改进

$$\theta \leftarrow \theta + \alpha_{\theta}\frac{\partial \log \pi_{\omega,\theta}(a \mid s)}{\partial \theta} Q_U(s,\omega,a)$$

$$\vartheta \leftarrow \vartheta - \alpha_{\vartheta}\frac{\partial \beta_{\omega,\vartheta}(s')}{\partial \vartheta}(Q_{\Omega}(s',\omega) - V_{\Omega}(s'))$$

If $\beta_{\omega,\vartheta}$ 在 s' 终止 then

  根据 $\epsilon - soft(\pi_{\Omega}(s'))$ 选择新的 ω

  $s \leftarrow s'$

直到 s' 是终止状态

End If

End For

Option-Critic 算法的结构如图 5-7 所示。

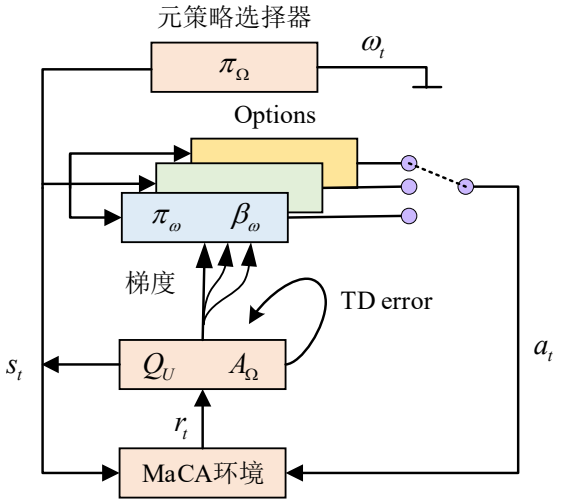


图 5-7 Option-Critic 结构图

该模型的构建过程如图 5-8 所示。

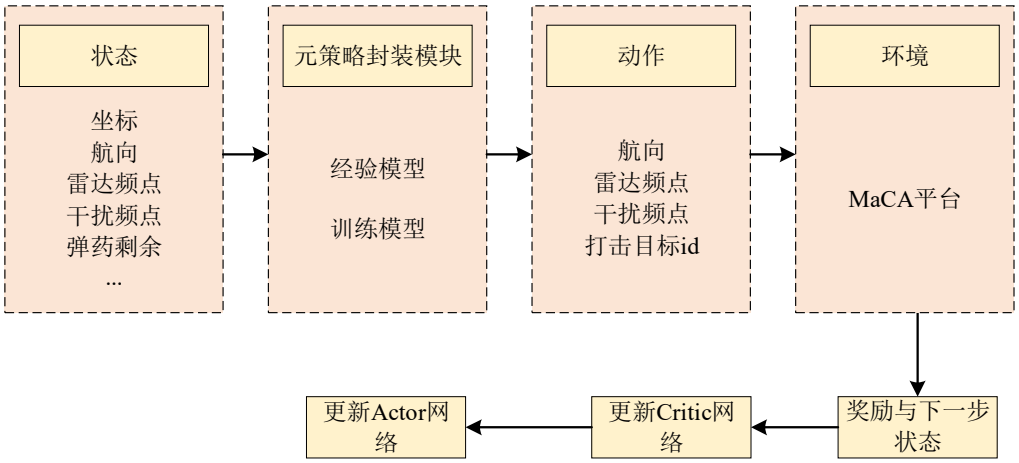


图 5-8 Options 模型构建及训练流程

在该策略框架内，探测到的敌机坐标主要来源于两个渠道：一是友机的侦测数据，二是我机的侦测信息。

Options 通过终止条件 $\beta(s)$ 以判断是否终止。单步离线策略更新目标 $g_t^{(1)}$ 如式所示：

$$\begin{aligned} g_t^{(1)} = r_{t+1} + \gamma\Big((1-\beta_{\omega_t,\vartheta}(s_{t+1}))\sum_a \pi_{\omega_t,\theta}(a\,|\,s_{t+1})Q_U(s_{t+1},\omega_t,a) \\ + \beta_{\omega_t,\vartheta}(s_{t+1})\max_{\omega}\sum_a \pi_{\omega,\theta}(a\,|\,s_{t+1})Q_U(s_{t+1},\omega,a)\Big) \end{aligned} \tag{5-11}$$

整体作战策略包括一个可训练的策略，即机动决策策略，以及由四部分组成的固定策略集：雷达开关、主动干扰、主动探测和队形转换策略。在采用基于 E-SAC 算法框架的训练策略中，分别建立了策略执行网络和策略评估网络，此外，对状态空间、动作空间以及奖励函数进行了定义，具体如下所述。

(1) 状态空间

无人机机动决策模型的状态空间用于描述空中战斗的全部情况，该情况在任何时候都可以通过无人机状态来确定，无人机状态包含位置矢量 $\mathbf{R}=[x,y]$、速度 v、航向角 φ 和距离 d 中的相对信息。为了统一每个变量的范围并提高网络学习的效率，每个状态变量都被归一化为 $[0,1]$。状态空间定义为一个 6 元组 $S=[x,y,v,\varphi,d,q]$。

(2) 动作空间

与 4.3.2 节的动作空间相同，通过在合适的积分步长 dt 内定义单位速度 dv 和单位航向角 $d\varphi$，其动作空间为 $A=[dv,d\varphi]$。

(3) 奖励函数

为了确保空战模型的有效性，对于每个元策略，需要设计合适的奖励机制以激励最佳行为，并设定恰当的终止函数以在适当的时机结束当前策略，从而提高整体模型的性能。5 种元策略的奖励设置方案与顶层奖励函数如表 5-2 所示：

表 5-2 奖励设置

事件	奖励或惩罚	说明
雷达开关	+20	完成雷达开关切换动作
主动干扰	+100	我机成功设定干扰频点
主动探测	+10	我机成功发现目标
队形转换	+30	我机完成队形转换
机动决策	+100	我机追踪并打击目标
摧毁敌机	+500	我机成功击杀对手
被敌机摧毁	-500	我机被敌机击杀
超界	-10	我机超出战场边界

5.3.2 顶层策略网络模型

顶层策略网络主要负责根据当前环境和状态选择合适的元策略，其中，策略选择器是实现这一目标的具体机制，本章策略选择器的构建采用了基于贪婪策略的传统策略模型，即在每一状态下挑选那个预计带来最大回报的元策略。具体来说，这些元策略模型涵盖了雷达开关、主动干扰、主动探测、队形转换以及机动决策模型。策略选择器的模型构建如图 5-9 所示。

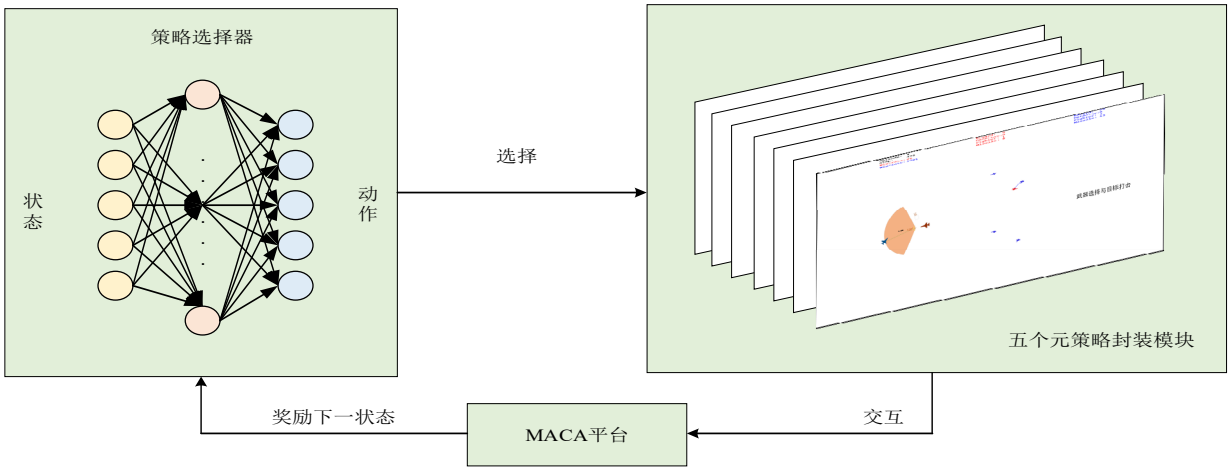


图 5-9 策略选择器模型构建

在这一框架中，策略选择器将上述五个元策略视为五种可选动作，并依据贪婪策略在给定的环境状态中选取元策略，以此方式获取奖励并过渡到下一个时间点的状态。通过不断重复这一过程，策略选择器得以训练，直至能够准确选择适宜的策略动作以应对顶层策略的需求。

在整个作战流程中，策略选择器需在不同状态下展现出以下能力：

(1) 先敌发现

初始阶段，我机采取队形转换策略以重组编队；随后，执行主动探测策略以侦察目标；在执行分布式搜寻时，应用雷达开关策略，而在其他情形下则依据“决策区”标准做出适当调整；在战斗操作中，通过将干扰频点设定为已侦测的敌机雷达频点，实行主动干扰策略。

(2) 先敌打击

一旦发现目标，为防止目标逃逸，应采用机动决策策略，锁定目标位置并将其引导至攻击区域。

顶层策略选择器的训练过程包括以下步骤：

1) 初始化设置：初始化元策略训练网络及元策略选择网络的参数；

2) 底层动作产生：利用 Actor 网络以及预设的专家经验模型策略函数来输出动作值；

3) 元策略动作决定：创建遵循均匀分布的随机动作选择；

4) 与环境互动：执行生成的动作集在模拟环境中，收集获得的奖励集、下一状态集以及任务完成标识集；

5) 经验保存：将获得的经验数据分别储存于相应的元策略经验池中；

6) 测试与评估：若达到预定标准则终止训练，否则回到步骤 1 并重复执行。

5.4 仿真结果与分析

5.4.1 实验环境设定

设置敌我双方无人机各四架，每架携带八枚导弹，设定地图的尺寸为$150\times150km$，并使用 MaCA 环境中的 fix_rule 黑盒算法作为敌机的控制算法，我机则采用本文所提出的改进分层强化学习算法。其具体参数如表 5-3 所示。

作战结果判定标准如下。

1) 胜利：当双方的导弹库存耗尽，拥有更多存活作战单元的一方获胜，或者当战斗持续到最大步数限制时，剩余的作战单元数量较多的一方获胜；

2) 平局：若双方作战单位全部被击毁，或者双方均用尽导弹且剩余作战单位数相同，或当战斗进行到步数上限而双方存活单位数仍旧持平；

2) 战败：若一方的作战单位全部被击毁、在导弹耗尽时拥有较少存活作战单元，或战斗达到步数上限时剩余作战单元数较少。

表 5-3 基于分层强化学习的编队协同智能空战战术决策模型仿真参数表

参数名称	参数大小
Actor 策略网络学习率	3×10^{-4}
Critic 策略网络学习率	3×10^{-3}
温度参数	3×10^{-4}
网络隐含层个数	2
网络隐含层神经元个数	512
学习速率	0.0003
累积回报折扣因子	0.99
目标熵 H_0	–3
熵的正则化系数初始值	1
目标网络更新间隔步数	300
经验池容量	100,000
经验池提取样本数量	200
激活函数	*ReLU*
优化器 Adam 软更新系数	0.005

5.4.2 全流程仿真结果与分析

(1) 雷达开关策略有效性验证

雷达开关策略可以帮助我机降低暴露于敌机探测区域的风险。为了验证雷达开关策略的有效性，本章首先不考虑雷达开关策略，保持雷达常开的状态，统计我机被敌机被动探测到的次数；在对照仿真中，启用雷达开关策略以模拟整个的空战过程，记录我机被敌机探测到的次数。一共进行 2000 个回合的空战对抗实验，空战结果如表 5-4 所示。

表 5-4 雷达开关策略有效性验证

	关闭雷达开关策略	使用雷达开关策略
平均被发现次数(次)	21.5	15.7

由实验结果可知，关闭雷达开关策略其平均每局被动探测 21.5 次，而开启雷达开关策略其平均每局被动探测 15.7 次，两者的结果对比可知，雷达策略可以使得被敌机发现的次数要平均少 5.8 次，可见使用雷达开关策略可以降低我机因被动探测而暴露位置的风险。该实验验证了分层强化学习空战智能决策中雷达开关策略的有效性。

(2) 主动干扰策略有效性验证

主动干扰的核心目标是通过发送干扰信号来对敌方飞机的雷达功能产生影响，从而增强我方的攻击效能。为了验证主动干扰策略的有效性，采用基于改进分层强化学习算法对全流程空战进行模拟，首先关闭主动干扰策略，记录我机被敌机探测到的次数，然后开启主动干扰策略对全流程的空战进行模拟，记录我机被敌机探测到的次数。一共进

行 2000 个回合的空战对抗实验，空战结果如表 5-5 所示。

表 5-5 主动干扰策略有效性验证

	关闭主动干扰策略	使用主动干扰策略
敌机平均探测到的我机数量(架)	2.3	1.1

由实验结果可知，该实验验证了分层强化学习空战智能决策中雷达开关策略的有效性。开启主动干扰策略后，敌机探测到我机的次数有所减少，从平均探测到 2.3 架变为平均探测到 1.1 架，这表明主动干扰策略能够有效地降低了敌机的探测能力。

(3) 主动探测策略有效性验证

在主动探测策略框架下，采用的分布式搜索技术结合了人工势场理论的斥力场机制，以保持友机之间的适当间隔，防止对相同区域的反复搜索。提取我机在进行分布式搜索策略时的无人机轨迹数据，并将其绘制到地图中，其示意图如图 5-10 所示，可以看到该策略能够实现我机在彼此靠近时自动执行避让动作，以达到避撞的效果。

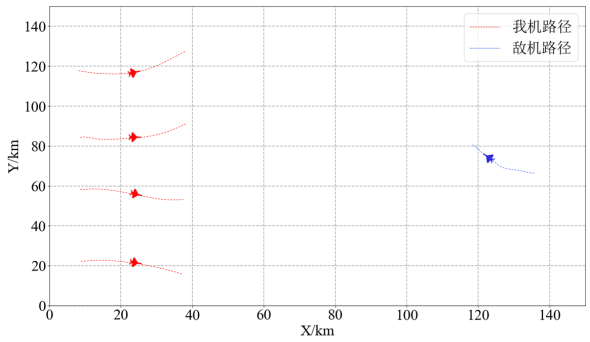


图 5-10 我机分布式搜索策略

(4) 队形转换策略有效性验证

空战场景初始化设置为长机-僚机编队形式，如图 5-11 所示。

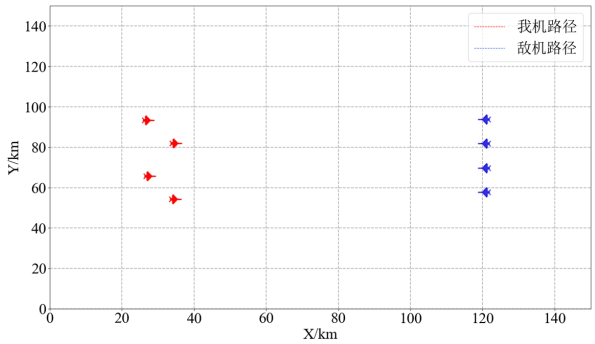


图 5-11 初始编队示意图

队形转换策略使得我机队形在遭受破坏的情形下也能够重新组织我机编队。此重组后的编队能够对目标施加压力和干扰，从而有效地规避了孤立作战的风险。为了验证队形转换策略的有效性，在进行回合数为 2000 的 4v4 空战全流程仿真时关闭队形转换策略，即初始的长机-僚机编队中有飞机战损，则编队中的无人机以单机的方式对探测目标

并对其进行攻击，统计实验中 1000 步内击败敌机的数量；然后，同样进行回合数为 2000 的 4v4 空战全流程仿真，此时启用队形转换策略，同样统计 1000 步内击败敌机的数量。具体实验结果如表 5-6 所示。

表 5-6 队形转换策略有效性验证

	关闭队形转换策略	使用队形转换策略
1000 步内打击敌机数量(架)	0.5	1.1

由实验结果可知，在关闭队形转换策略时 2000 个回合中 1000 步内平均打击敌机的数量为 0.5 架，但开启队形转换策略后其数量上升至 1.1 架，可证明队形转换策略的有效性。同时，保持长-僚机编队的策略能够有效地确保优先取得信息优势，本策略旨在对敌机实施有效的压制和干扰，在此策略下，我机能够在不受损失的情况下消灭敌机，并且从作战的初始阶段就确保了数量上的优势。

(5) 无人机编队全流程空战仿真

首先采用第四章所提的 E-SAC 算法对 4v4 场景下的机动决策元策略进行训练，奖励函数设置如表 5-2 所示，其训练奖励曲线图如图 5-12 所示。

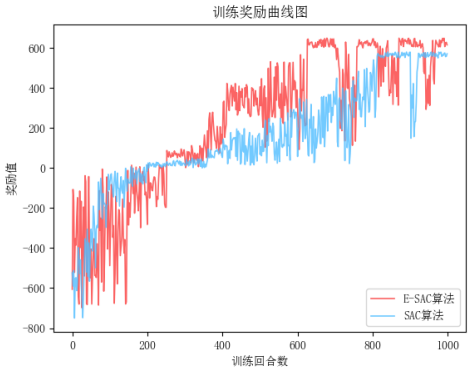
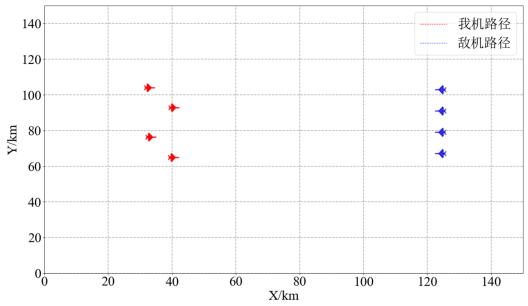
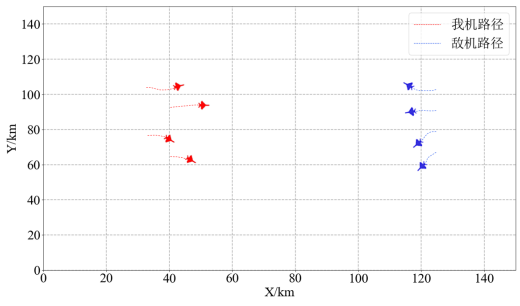


图 5-12 机动决策策略训练奖励曲线图

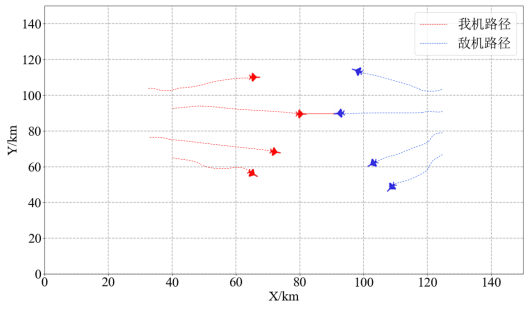
训练结束后，提取 MaCA 平台中我机与敌机对抗时的仿真结果轨迹数据，并将其绘制到地图中，具体仿真过程如图 5-13 所示。



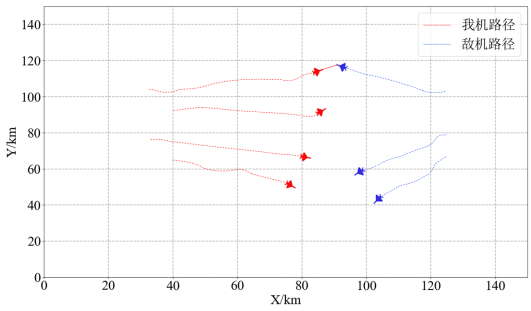
(a) 初始态势



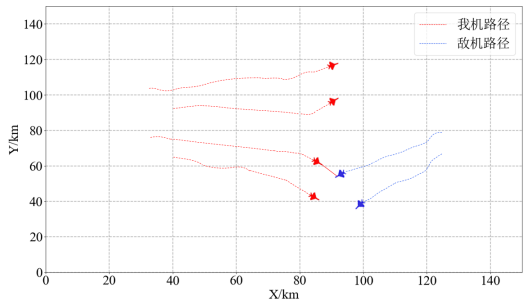
(b) 追踪敌机



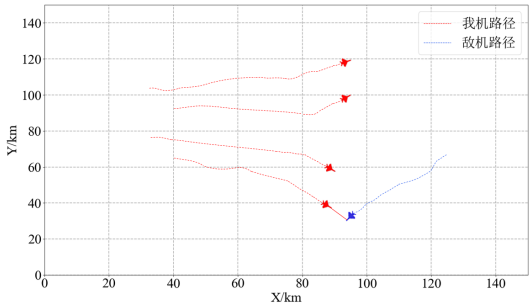
(c) 打击敌机(1)



(d) 打击敌机(2)



(e) 打击敌机(3)



(f) 打击敌机(4)

图 5-13 无人机编队四对四机动决策训练仿真过程

然后将训练完成的策略与敌机进行对抗仿真，采用本章所提的改进 Option-Critic 算法和传统 Option-Critic 算法对 4v4 场景下的全流程空战进行仿真，其具体的训练奖励曲线如图 5-14 所示。

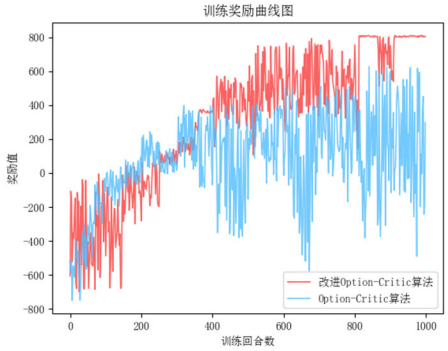


图 5-14 全流程空战训练曲线图

具体的训练结果数据如表 5-7 所示。

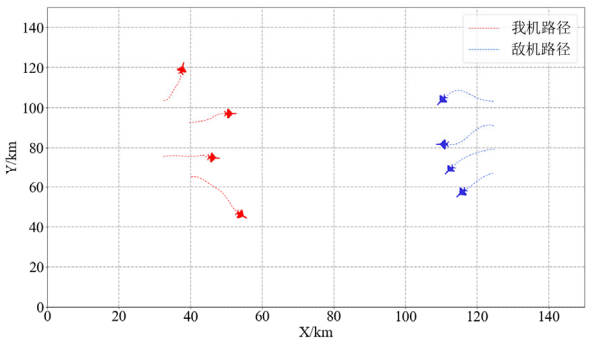
表 5-7 分层强化学习训练结果

	Option-Critic	改进 Option-Critic
最大奖励	-	808
奖励收敛回合数	-	811

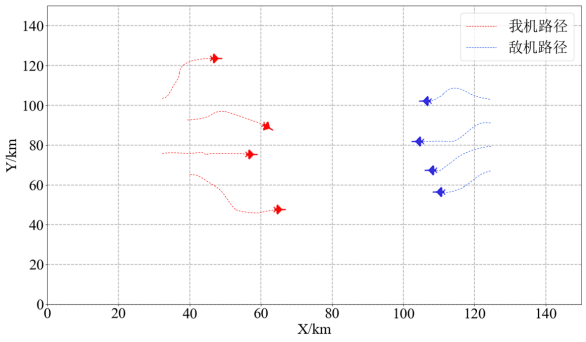
从训练过程中的奖励曲线图可以看出，在 1000 个回合数内，传统 Option-Critic 算

法出现难以收敛的情况；而本文所提的改进 Option-Critic 算法能够完成全流程空战任务，并最终获得 808 的奖励值，因此本文所提的算法具有有效性。

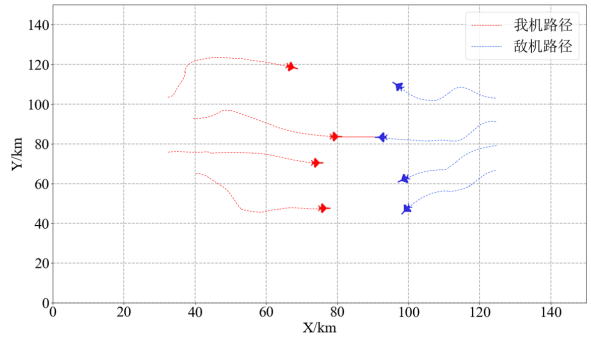
利用本文所提的改进分层强化学习对无人机编队协同战术决策仿真，提取我机与敌机对抗时的仿真结果轨迹数据，并将其绘制于地图中以获得仿真过程示意图，如图 5-15 所示。



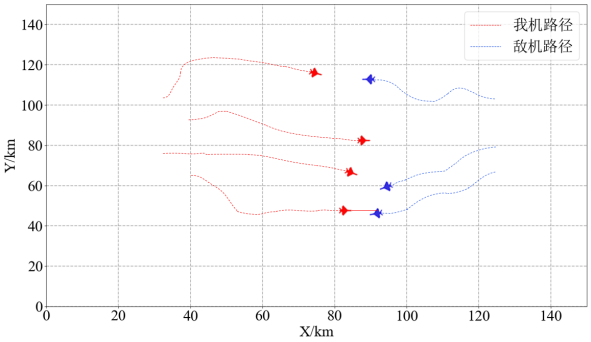
(a) 我机主动探测



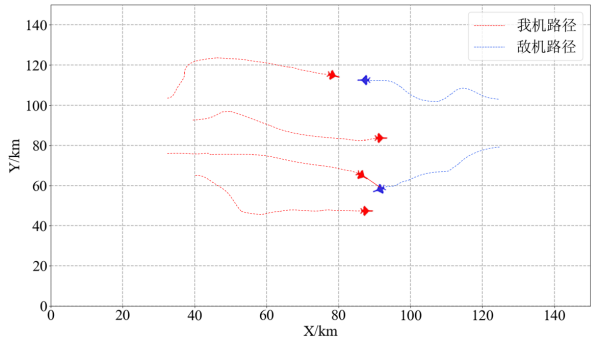
(b) 我机主动干扰



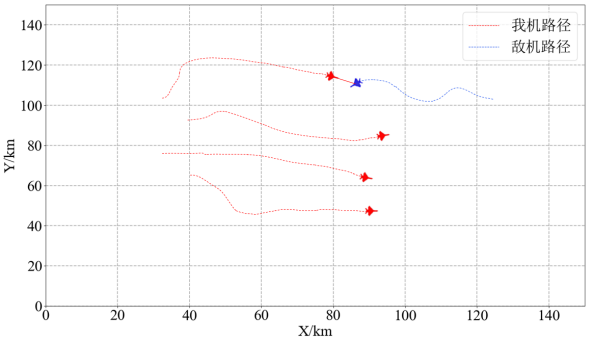
(c) 打击敌机(1)



(d) 打击敌机(2)



(e) 打击敌机(3)



(f) 打击敌机(4)

图 5-15 无人机编队四对四战术决策作战仿真过程

由仿真结果表明，我机能够在空战中进行有效的战术决策。在空战 4v4 场景中，我方无人机首先采用了主动探测策略对作战环境进行分布式搜索，在我机探测到敌机后选择了主动干扰策略对敌机进行干扰，随后采用了机动决策策略引导我机朝着敌机航行，

并在到达导弹攻击区后对敌机进行攻击，最终将敌机全部歼灭，验证了本文所设计的无人机编队空战战术决策方法的有效性。

为进一步评估本文设计的编队协同分层强化学习空战决策模型的稳定性与可靠性，引进了“平均战损”这一度量指标，该指标定义为每次空战模拟中无人机平均损失的数量。本次仿真对 50 场作战结果进行统计，统计作战胜率和平均战损，如表 5-8 所示。

表 5-8 分层强化学习仿真结果

	蓝方(敌机)	红方(我机)
胜率	2%	98%
平均战损	5.7	1.4

由实验结果可知，在 50 场空战对抗中，应用本文的编队协同智能空战战术决策方法平均可以实现 98%的胜率，其战损仅为 1.4 架战机，蓝方飞机胜率为 2%且平均每局损失高达 5.7 架战机，可见本论文提出的编队协同智能空战战术决策算法显著提升了无人机编队的作战能力，具备卓越的作战效能。

5.5 本章小结

本章针对无人机编队协同空战中的战术决策问题，提出了一种基于改进Option-Critic算法的编队协同策略模型。首先，构建了一个元策略模型，该模型基于分层强化学习算法，用于制定无人机编队协同空战的战术决策；其次，基于元策略选择器对分层强化学习战术决策模型进行改进；最后，设置仿真实验验证本文所提的分层强化学习算法在空战战术决策中的有效性，并将改进前后的算法进行对比。

第 6 章 总结与展望

6.1 论文总结

随着无人机技术的飞速发展和现代战争形态的不断演变，空中作战环境变得日益复杂和难以预测。在这样的背景下，提高无人机编队在战术决策上的自主性和效率，尤其是其智能化作战能力，成为了军事科技进步的重要方向。论文针对无人机编队的协同战术决策问题，从无人机的协同航路规划、机动决策以及编队协同策略模型三个维度进行了深入研究。以下是该论文的核心研究内容：

(1) 针对多无人机航路规划问题，提出了一种改进灰狼算法的多无人机协同航路规划模型，提高了多无人机航路规划的能力和效率。

首先，构建了多无人机协同航路规划的模型；其次，针对灰狼算法收敛速度缓慢和倾向于落入局部最优解的问题，提出了通过非线性函数优化收敛系数的方法，并根据不同灰狼的适应度设计权重比例因子；最后，将传统灰狼优化算法与改进后的灰狼算法进行仿真对比，同时分析了算法的有效性。

(2) 针对无人机空战机动决策问题，提出了一种基于 E-SAC(Expert-Soft ActorCritic, E-SAC)算法的机动决策模型，增强了智能体的探索效率以及优化学习过程。

首先，针对黄金分割法在实时性方面的局限性以及传统 BP 神经网络在拟合攻击区精度方面的不足，提出一种基于调优反向传播神经网络(Tuning of Backpropagation Neural Networks, TBPNN)算法的导弹攻击区的拟合方法；其次，根据计算出的攻击区，设计了一个基于深度强化学习算法的无人机机动决策模型；最后，对 SAC（Soft Actor Critic SAC）算法进行了改进，并提出了一种基于 E-SAC 算法的无人机自主机动决策方法，有效解决了 SAC 算法在处理无人机空战机动决策问题时收敛慢和容易停留在局部最优解的难题。仿真实验结果显示，本文提出的 E-SAC 方法在处理无人机机动决策问题上显示出了高效性和实用性，并且在收敛速度上显著超过了传统的 SAC 算法。

(3) 针对无人机编队在协同空中作战的整体决策问题，提出了一种基于改进的分层强化学习算法的编队协作策略模型，实现了多无人机编队协同空战策略模型的构建。

首先，根据无人机编队空战战术决策任务需求，建立了无人机编队协同空战元策略模型；其次，基于分层强化学习算法框架提出了一种无人机编队协同策略模型，解决了空战战术决策中传统 Option-Critic 模型训练不易收敛的问题；最后，基于 MaCA 平台实现无人机编队协同策略的仿真。仿真结果表明，本文所提的改进分层强化学习模型可以对无人机编队做出有效的协同空战战术决策。

6.2 后续展望

本文针对无人机编队协同智能空战战术决策问题，按照不同的作战任务进行划分，包括进入作战区域前和进入作战区域后的无人机编队协同场景进行研究。根据仿真的数据分析，本文实现了所设定的研究目标。考虑到论文的研究主题以及实际空中战斗对无人机的具体需求，这篇论文还可以在几个关键领域进行更深入的探讨：

(1) 论文中雷达使用相关的元策略的研究未涉及被动探测策略。

考虑到被动探测作为一种有效降低敌机发现我方飞行器概率的策略，在实际空战中扮演着至关重要的角色。因此，未来的研究应当考虑纳入被动探测策略，这不仅能增强无人机在复杂环境中的生存能力，还能提升其战术上的灵活性和适应性。

(2) 在论文中探讨无人机空战全流程决策对抗仿真的过程中，鉴于多智能体系统固有的复杂性，未来的工作将着重于探索更高效的协同协议和通信机制。

有效的协同协议对于确保多无人机系统在复杂空战环境中的高效协作至关重要。这包括但不限于，发展先进的任务分配算法、协同航路规划和战术决策同步机制。同时，鉴于通信延迟和中断可能对协同操作产生显著影响，改进通信机制，比如实现低延迟、高可靠性的数据传输，将是提高整体系统性能的关键。

参考文献

[1] 蒯伟. 无人机集群自主控制发展综述[J]. 信息化研究, 2023, 49(05): 1-5+25.

[2] 李军, 陈士超. 无人机蜂群关键技术发展综述[J]. 兵工学报, 2023, 44(09): 2533-2545.

[3] 朱超磊, 袁成, 杨佳会等. 2021 年国外军用无人机装备技术发展综述[J]. 战术导弹技术, 2022(01): 38-45.

[4] 魏齐. 里程碑!中国大型无人机编队飞行[N]. 环球时报, 2023-06-30(008).

[5] Sun W, Jiang N, Krishnamurthy A, et al. Model-based rl in contextual decision processes: Pac bounds and exponential improvements over model-free approaches[C]. Conference on learning theory. PMLR, 2019: 2898-2933.

[6] Lee C S, Wang M H, Chaslot G, et al. The computational intelligence of MoGo revealed in Taiwan's computer Go tournaments[J]. IEEE Transactions on Computational Intelligence and AI in games, 2009, 1(1): 73-89.

[7] Lu F, Yamamoto K, Nomura L H, et al. Fighting game artificial intelligence competition platform[C]. 2013 IEEE 2nd Global Conference on Consumer Electronics (GCCE). IEEE, 2013: 320-323.

[8] Li Y, Han W, Wang Y. Deep reinforcement learning with application to air confrontation intelligent decision-making of manned/unmanned aerial vehicle cooperative system[J]. IEEE Access, 2020, 8: 67887-67898.

[9] Yue L, Yang R, Zhang Y, et al. Deep reinforcement learning for UAV intelligent mission planning[J]. Complexity, 2022, 2022.

[10] Wu Z S, Fu W P. A review of path planning method for mobile robot[J]. Advanced Materials Research, 2014, 1030: 1588-1591.

[11] Zhu Z, Yin Y, Lyu H. Automatic collision avoidance algorithm based on route-plan-guided artificial potential field method[J]. Ocean Engineering, 2023, 271: 113737.

[12] Bertsimas D, Tsitsiklis J. Simulated annealing[J]. Statistical science, 1993, 8(1): 10-15.

[13] Choi K, Jang D H, Kang S I, et al. Hybrid algorithm combing genetic algorithm with evolution strategy for antenna design[J]. IEEE transactions on Magnetics, 2015, 52(3): 1-4.

[14] Yu X, Li C, Zhou J F. A constrained differential evolution algorithm to solve UAV path planning in disaster scenarios[J]. Knowledge-Based Systems, 2020, 204: 106209.

[15] Alkhateeb F, Abed-alguni B H, Al-rousan M H. Discrete hybrid cuckoo search and simulated annealing algorithm for solving the job shop scheduling problem[J]. The Journal of Supercomputing, 2022: 1-28.

[16] Khan M S A, Santhosh R. Task scheduling in cloud computing using hybrid optimization algorithm[J]. Soft Computing, 2022, 26(23): 13069-13079.

[17] Praveen Kumar R, Raj J S, Smys S. Performance analysis of hybrid optimization algorithm for virtual head selection in wireless sensor networks[J]. Wireless Personal Communications, 2021: 1-16.

[18] Chandrasekhar U, Khare N. An intelligent tutoring system for new student model using fuzzy soft set-based hybrid optimization algorithm[J]. Soft Computing, 2021, 25: 14979-14992.

[19] Alkhateeb F, Abed-alguni B H, Al-rousan M H. Discrete hybrid cuckoo search and simulated annealing algorithm for solving the job shop scheduling problem[J]. The Journal of Supercomputing, 2022: 1-28.

[20] Yan F. Gauss interference ant colony algorithm-based optimization of UAV mission planning[J]. The Journal of Supercomputing, 2020, 76(2): 1170-1179.

[21] Lee J H, Song J Y, Kim D W, et al. Particle swarm optimization algorithm with intelligent particle number control for optimal design of electric machines[J]. IEEE Transactions on Industrial Electronics, 2017, 65(2): 1791-1798.

[22] Li H, Baoyin H. Optimization of multiple debris removal missions using an evolving elitist club algorithm[J]. IEEE Transactions on Aerospace and Electronic Systems, 2019, 56(1): 773-784.

[23] Cheng R, Song Y, Chen D, et al. Intelligent positioning approach for high speed trains based on ant colony optimization and machine learning algorithms[J]. IEEE Transactions on Intelligent Transportation Systems, 2018, 20(10): 3737-3746.

[24] Wang Y, Zhang Y, Xu D, et al. Improved whale optimization-based parameter identification algorithm for dynamic deformation of large ships[J]. Ocean Engineering, 2022, 245: 110392.

[25] Mirjalili S, Mirjalili S M, Lewis A. Grey wolf optimizer[J]. Advances in engineering software, 2014, 69: 46-61.

[26] Daniel E, Anitha J, Gnanaraj J. Optimum laplacian wavelet mask based medical image using hybrid cuckoo search–grey wolf optimization algorithm[J]. Knowledge-Based Systems, 2017, 131: 58-69.

[27] Guha D, Roy P K, Banerjee S. Load frequency control of interconnected power system using grey wolf optimization[J]. Swarm and Evolutionary computation, 2016, 27: 97-115.

[28] Emary E, Zawbaa H M, Hassanien A E. Binary grey wolf optimization approaches for feature selection[J]. Neurocomputing, 2016, 172: 371-381.

[29] Radmanesh M, Kumar M, Sarim M. Grey wolf optimization based sense and avoid algorithm in a Bayesian framework for multiple UAV path planning in an uncertain environment[J]. Aerospace Science and Technology, 2018, 77: 168-179.

[30] Long W, Jiao J, Liang X, et al. Inspired grey wolf optimizer for solving large-scale function optimization problems[J]. Applied Mathematical Modelling, 2018, 60: 112-126.

[31] Kumar V, Kumar D. An astrophysics-inspired grey wolf algorithm for numerical optimization and its application to engineering design problems[J]. Advances in Engineering Software, 2017, 112: 231-254.

[32] Long W, Jiao J, Liang X, et al. An exploration-enhanced grey wolf optimizer to solve high-dimensional numerical optimization[J]. Engineering Applications of Artificial Intelligence, 2018, 68: 63-80.

[33] Zhang X, Wang X, Kang Q, et al. Hybrid grey wolf optimizer with artificial bee colony and its application to clustering optimization[J]. Acta Electonica Sinica, 2018, 46(10): 2430.

[34] Lu F, Feng W, Gao M, et al. The fourth-party logistics routing problem using ant colony system-improved grey wolf optimization[J]. Journal of Advanced Transportation, 2020, 2020: 1-15.

[35] Wu Y, Sun X, Dai B, et al. A transformer fault diagnosis method based on hybrid improved grey wolf optimization and least squares-support vector machine[J]. IET generation, transmission & distribution, 2022, 16(10): 1950-1963.

[36] Khan I, Basuli K, Maiti M K. Multi-objective covering salesman problem: a decomposition approach using grey wolf optimization[J]. Knowledge and Information Systems, 2023, 65(1): 281-339.

[37] Wang Y L, Wang T, Yao C. Gray wolf optimization algorithm based on Kent mapping and adaptive weight[J]. Application Research of Computers, 2020, 37(2): 37-40.

[38] Ou Y, Yin P, Mo L. An Improved Grey Wolf Optimizer and Its Application in Robot Path Planning[J]. Biomimetics, 2023, 8(1): 84.

[39] Liu Y, Jiang Y, Zhang X, et al. An improved grey wolf optimizer algorithm for

identification and location of gas emission[J]. Journal of Loss Prevention in the Process Industries, 2023, 82: 105003.

[40] 张超然, 吕余海. 基于置信度神经网络的机载武器攻击区拟合算法[J]. 弹箭与制导学报, 2021, 41(04): 120-124.

[41] Austin F, Carbone G, Falco M, et al. Game theory for automated maneuvering during air-to-air combat[J]. Journal of Guidance, Control, and Dynamics, 1990, 13(6): 1143-1149.

[42] Park H, Lee B Y, Tahk M J, et al. Differential game based air combat maneuver generation using scoring function matrix[J]. International Journal of Aeronautical and Space Sciences, 2016, 17(2): 204-213.

[43] Virtanen K, Karelahti J, Raivio T. Modeling air combat by a moving horizon influence diagram game[J]. Journal of guidance, control, and dynamics, 2006, 29(5): 1080-1091.

[44] Grimm W, Well K H. Modelling air combat as differential game recent approaches and future requirements[C].Differential Games—Developments in Modelling and Computation: Proceedings of the Fourth International Symposium on Differential Games and Applications August 9–10, 1990, Helsinki University of Technology, Finland. Springer Berlin Heidelberg, 1991: 1-13.

[45] Lee B Y, Han S, Park H J, et al. One-versus-one air-to-air combat maneuver generation based on the differential game[C].Proceedings of the 2016 Congress of the International Council of the Aeronautical Sciences. 2016: 1-7.

[46] McGrew J S, How J P, Williams B, et al. Air-combat strategy using approximate dynamic programming[J]. Journal of guidance, control, and dynamics, 2010, 33(5): 1641-1654.

[47] 袁广进. 多无人机协同搜索算法研究与实现[D]. 南京邮电大学, 2021.

[48] Zhang X, Liu G, Yang C, et al. Research on air confrontation maneuver decision-making method based on reinforcement learning[J]. Electronics, 2018, 7(11): 279.

[49] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. nature, 2015, 518(7540): 529-533.

[50] Fang J, Zhang L, Fang W and Xu T, "Approximate dynamic programming for CGF air combat maneuvering decision," 2016 2nd IEEE International Conference on Computer and Communications (ICCC), Chengdu, 2016, pp. 1386-1390.

[51] Wang Y, Long Q, Wu M, Chen Y and Tuhinur R, "Research on Maneuvering Control Algorithm of Short-Range UAV Air Combat Based on Deep Reinforcement Learning," 2023 2nd International Conference on Machine Learning, Cloud Computing and Intelligent Mining (MLCCIM), Jiuzhaigou, China, 2023, pp. 83-90.

[52] Isci H, Koyuncu E. Reinforcement Learning Based Autonomous Air Combat with Energy Budgets[C].AIAA SCITECH 2022 Forum. 2022: 0786.

[53] Li Y, Lyu Y, Shi J, et al. Autonomous Maneuver Decision of Air Combat Based on Simulated Operation Command and FRV-DDPG Algorithm[J]. Aerospace, 2022, 9(11): 658.

[54] Li L, Zhou Z, Chai J, et al. Learning continuous 3-DOF air-to-air close-in combat strategy using proximal policy optimization[C]//2022 IEEE Conference on Games (CoG). IEEE, 2022: 616-619.

[55] Jing X, Hou M, Wu G, et al. Research on maneuvering decision algorithm based on improved deep deterministic policy gradient[J]. IEEE Access, 2022, 10: 92426-92445.

[56] Fujimoto S, Hoof H, Meger D. Addressing function approximation error in actor-critic methods[C].International conference on machine learning. PMLR, 2018: 1587-1596.

[57] Zhang S, Li Y, Dong Q. Autonomous navigation of UAV in multi-obstacle environments based on a Deep Reinforcement Learning approach[J]. Applied Soft Computing, 2022, 115: 108194.

[58] Haarnoja T, Zhou A, Abbeel P, et al. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor[C].International conference on machine learning. PMLR, 2018: 1861-1870.

[59] Li B, Huang J, Bai S, et al. Autonomous air combat decision-making of UAV based on parallel self-play reinforcement learning[J]. CAAI Transactions on Intelligence Technology, 2023, 8(1): 64-81.

[60] 刘承相, 熊俊杰, 魏秀玲, 孙多. 多无人机协同控制现状及发展前景探究[J]. 电子测试, 2022(16): 110-112.

[61] Gaerther U, Chung T H. UAV swarm tactics: an agent-based simulation and Markov process analysis[M]. Monterey, California: Naval Postgraduate School, 2013.

[62] Ernest N, Carroll D, Schumacher C, et al. Genetic fuzzy based artificial intelligence for unmanned combat aerial vehicle control in simulated air combat missions[J]. Journal of Defense Management, 2016, 6(1): 2167-0374.

[63] Ramirez-Atencia C, Bello-Orgaz G, R-Moreno M D, et al. Solving complex multi-UAV mission planning problems using multi-objective genetic algorithms[J]. Soft Computing, 2017, 21: 4883-4900.

[64] Zhen Z, Xing D, Gao C. Cooperative search-attack mission planning for multi-UAV based on intelligent self-organized algorithm[J]. Aerospace Science and Technology, 2018, 76: 402-411.

[65] Zhang G, Li Y, Xu X, et al. Multiagent reinforcement learning for swarm confrontation environments[C].Intelligent Robotics and Applications: 12th International Conference, ICIRA 2019, Shenyang, China, August 8–11, 2019, Proceedings, Part III 12. Springer International Publishing, 2019: 533-543.

[66] Sun Z, Piao H, Yang Z, et al. Multi-agent hierarchical policy gradient for air combat tactics emergence via self-play[J]. Engineering Applications of Artificial Intelligence, 2021, 98: 104112.

[67] Zhao W, Chu H, Miao X, et al. Research on the multiagent joint proximal policy optimization algorithm controlling cooperative fixed-wing UAV obstacle avoidance[J]. Sensors, 2020, 20(16): 4546.

[68] Heidlauf P, Collins A, Bolender M, et al. Verification Challenges in F-16 Ground Collision Avoidance and Other Automated Maneuvers[C].ARCH@ ADHS. 2018: 208-217.

[69] Wang M, Wang L, Yue T, et al. Influence of unmanned combat aerial vehicle agility on short-range aerial combat effectiveness[J]. Aerospace Science and Technology, 2020, 96: 105534.

[70] Wei Y J, Zhang H P, Huang C Q. Maneuver Decision-Making For Autonomous Air Combat Through Curriculum Learning And Reinforcement Learning With Sparse Rewards[J]. arXiv preprint arXiv:2302.05838, 2023.

[71] Hu Y, Wang W, Jia H, et al. Learning to utilize shaping rewards: A new approach of reward shaping[J]. Advances in Neural Information Processing Systems, 2020, 33: 15931-15941.

[72] Piao H, Sun Z, Meng G, et al. Beyond-visual-range air combat tactics auto-generation by reinforcement learning[C].2020 international joint conference on neural networks (IJCNN). IEEE, 2020: 1-8.

[73] Wang A, Zhao S, Shi Z, et al. Over-the-Horizon Air Combat Environment Modeling and Deep Reinforcement Learning Application[C].2022 4th International Conference on Data-driven Optimization of Complex Systems (DOCS). IEEE, 2022: 1-6.

[74] Hu J, Wang L, Hu T, et al. Autonomous maneuver decision making of dual-UAV cooperative air combat based on deep reinforcement learning[J]. Electronics, 2022, 11(3): 467.

[75] Zhan G, Zhang X, Li Z, et al. Multiple-uav reinforcement learning algorithm based on improved ppo in ray framework[J]. Drones, 2022, 6(7): 166.

[76] Narvekar S, Sinapov J, Stone P. Autonomous Task Sequencing for Customized Curriculum Design in Reinforcement Learning[C].IJCAI. 2017: 2536-2542.

[77] Liu X, Yin Y, Su Y, et al. A Multi-UCAV Cooperative Decision-Making Method Based on an MAPPO Algorithm for Beyond-Visual-Range Air Combat[J]. Aerospace, 2022, 9(10): 563.

[78] Sun Z, Piao H, Yang Z, et al. Multi-agent hierarchical policy gradient for air combat tactics emergence via self-play[J]. Engineering Applications of Artificial Intelligence, 2021, 98: 104112.

[79] Tobin J, Fong R, Ray A, et al. Domain randomization for transferring deep neural networks from simulation to the real world[C].2017 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE, 2017: 23-30.

[80] Comanici G, Precup D. Optimal policy switching algorithms for reinforcement learning[C].Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1. 2010: 709-714.

[81] Rane S. Learning with curricula for sparse-reward tasks in deep reinforcement learning[D]. Massachusetts Institute of Technology, 2020.

[82] Barto A G, Mahadevan S. Recent advances in hierarchical reinforcement learning[J]. Discrete event dynamic systems, 2003, 13(1-2): 41-77.

[83] Pope A P, Ide J S, Mićović D, et al. Hierarchical reinforcement learning for air-to-air combat[C].2021 international conference on unmanned aircraft systems (ICUAS). IEEE, 2021: 275-284.

[84] Haarnoja T, Zhou A, Abbeel P, et al. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor[C].International conference on machine learning. PMLR, 2018: 1861-1870.

[85] Kong W, Zhou D, Du Y, et al. Hierarchical multi-agent reinforcement learning for multi-aircraft close-range air combat[J]. IET Control Theory & Applications, 2023, 17(13): 1840-1862.

[86] Selmonaj A, Szehr O, Del Rio G, et al. Hierarchical Multi-Agent Reinforcement Learning for Air Combat Maneuvering[J]. arXiv preprint arXiv:2309.11247, 2023.

[87] Kong W, Zhou D, Zhou Y, et al. Hierarchical reinforcement learning from competitive self-play for dual-aircraft formation air combat[J]. Journal of Computational Design and Engineering, 2023, 10(2): 830-859.

[88] 高晓光. 航空军用飞行器导论[M]. 西北工业大学出版社, 2004, 12.

[89] Afsar M M, Crump T, Far B. Reinforcement learning based recommender systems: A survey[J]. ACM Computing Surveys, 2022, 55(7): 1-38.

[90] Song W, Duan Z, Yang Z, et al. Explainable knowledge graph-based recommendation via deep reinforcement learning[J]. arXiv, et al. "Explainable knowledge graph-based recommendation via deep reinforcement learning." arXiv preprint arXiv:1906.09506 13 (2019).

[91] Wang X, Xu Y, He X, et al. Reinforced negative sampling over knowledge graph for recommendation[C].Proceedings of the web conference 2020. 2020: 99-109.

[92] Xian Y, Fu Z, Muthukrishnan S, et al. Reinforcement knowledge graph reasoning for explainable recommendation[C].Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval. 2019: 285-294.

[93] Zhao K, Wang X, Zhang Y, et al. Leveraging demonstrations for reinforcement recommendation reasoning over knowledge graphs[C].Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval. 2020: 239-248.

[94] Liu Q, Tong S, Liu C, et al. Exploiting cognitive structure for adaptive learning[C].Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2019: 627-635.

[95] Wang P，Fan Y，Xia L, et al. KERL: A knowledge-guided reinforcement learning model for sequential recommendation[C].Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. 2020: 209-218.

[96] Zhou S, Dai X, Chen H, et al. Interactive recommender system via knowledge graph-enhanced reinforcement learning[C].Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval. 2020: 179-188.

[97] Lei W, Zhang G, He X, et al. Interactive path reasoning on graph for conversational recommendation[C].Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining. 2020: 2073-2083.

[98] Deng Y, Li Y, Sun F, et al. Unified conversational recommendation policy learning via graph-based reinforcement learning[C].Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2021: 1431-1441.

[99] Ammanabrolu P, Riedl M O. Playing text-adventure games with graph-based deep reinforcement learning[J]. arXiv preprint arXiv:1812.01628, 2018.

[100] Hausknecht M, Ammanabrolu P, Côté M A, et al. Interactive fiction games: A colossal adventure[C].Proceedings of the AAAI Conference on Artificial Intelligence. 2020, 34(05): 7903-7910.

[101] Xu Y, Fang M, Chen L, et al. Deep reinforcement learning with stacked hierarchical attention for text-based games[J]. Advances in Neural Information Processing Systems, 2020,

33: 16495-16507.

[102] Adhikari A, Yuan X, Côté M A, et al. Learning dynamic belief graphs to generalize on text-based games[J]. Advances in Neural Information Processing Systems, 2020, 33: 3045-3057.

[103] Ammanabrolu P, Riedl M O. Transfer in deep reinforcement learning using knowledge graphs[J]. arXiv preprint arXiv:1908.06556, 2019.

[104] Murugesan K, Atzeni M, Shukla P, et al. Enhancing text-based reinforcement learning agents with commonsense knowledge[J]. arXiv preprint arXiv:2005.00811, 2020.

[105] Vinyals O, Babuschkin I, Czarnecki W M, et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning[J]. Nature, 2019, 575(7782): 350-354.

[106] Reis S, Reis L P, Lau N. Game adaptation by using reinforcement learning over meta games[J]. Group Decision and Negotiation, 2021, 30(2): 321-340.

[107] You J, Liu B, Ying Z, et al. Graph convolutional policy network for goal-directed molecular graph generation[J]. Advances in neural information processing systems, 2018, 31.

[108] Do K, Tran T, Venkatesh S. Graph transformation policy network for chemical reaction prediction[C].Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. 2019: 750-760.

[109] Wang S, Ren P, Chen Z, et al. Order-free medicine combination prediction with graph convolutional reinforcement learning[C].Proceedings of the 28th ACM international conference on information and knowledge management. 2019: 1623-1632.

[110] Sun Z, Dong W, Shi J, et al. Interpretable disease prediction based on reinforcement path reasoning over knowledge graphs[J]. arXiv preprint arXiv:2010.08300, 2020.

[111] Piplai A, Ranade P, Kotal A, et al. Using knowledge graphs and reinforcement learning for malware analysis[C].2020 IEEE International Conference on Big Data (Big Data). IEEE, 2020: 2626-2633.

[112] Kiran B R, Sobh I, Talpaert V, et al. Deep reinforcement learning for autonomous driving: A survey[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 23(6): 4909-4926.

[113] Kober J, Bagnell J A, Peters J. Reinforcement learning in robotics: A survey[J]. The International Journal of Robotics Research, 2013, 32(11): 1238-1274.

[114] Cui Y, Matsubara T, Sugimoto K. Pneumatic artificial muscle-driven robot control using local update reinforcement learning[J]. Advanced Robotics, 2017, 31(8): 397-412.

[115] 许雅筑, 武辉, 游科友等. 强化学习方法在自主水下机器人控制任务中的应用

[J]. 中国科学:信息科学, 2020, 50(12): 1798-1816.

[116] Yahya A, Li A, Kalakrishnan M, et al. Collective robot reinforcement learning with distributed asynchronous guided policy search[C].2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2017: 79-86.

[117] Akila V, Christaline J A, Mani A J, et al. Reinforcement Learning For Walking Robot[C].IOP Conference Series: Materials Science and Engineering. IOP Publishing, 2021, 1070(1): 012075.

[118] Uc-Cetina V, Navarro-Guerrero N, Martin-Gonzalez A, et al. Survey on reinforcement learning for language processing[J]. Artificial Intelligence Review, 2023, 56(2): 1543-1575.

[119] 徐聪. 基于深度学习和强化学习的对话模型研究[D]. 北京科技大学, 2020.

[120] Kaiser M, Saha Roy R, Weikum G. Reinforcement learning from reformulations in conversational question answering over knowledge graphs[C].Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2021: 459-469.

[121] Christmann P, Saha Roy R, Abujabal A, et al. Look before you hop: Conversational question answering over knowledge graphs using judicious context expansion[C].Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2019: 729-738.

[122] Kasim M F. Playing the game of Congklak with reinforcement learning[C].2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE). IEEE, 2016: 1-5.

[123] Hu H, et al. "Rationalizing translation elongation by reinforcement learning." bioRxiv (2018): 463976.

[124] Mahadevan S, Marchalleck N, Das T K, et al. Self-improving factory simulation using continuous-time average-reward reinforcement learning[C].MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-. MORGAN KAUFMANN PUBLISHERS, INC., 1997: 202-210.

[125] 张磊. 基于强化学习的多无人机协同控制算法研究[D]. 中国科学院大学(中国科学院长春光学精密机械与物理研究所), 2023.

[126] 马昂, 于艳华, 杨胜利等. 基于强化学习的知识图谱综述[J]. 计算机研究与发展, 2022, 59(08): 1694-1722.

[127] 严浙平, 杨泽文, 王璐等. 马尔可夫理论在无人系统中的研究现状[J]. 中国舰

船研究, 2018, 13(06): 9-18.

[128] Ji X, Li J. Online motion planning for UAV under uncertain environment[C].2015 8th International Symposium on Computational Intelligence and Design (ISCID). IEEE, 2015, 2: 514-517.

[129] Ure N K, Chowdhary G, How J P, et al. Health aware planning under uncertainty for UAV missions with heterogeneous teams[C].2013 European Control Conference (ECC). IEEE, 2013: 3312-3319.

[130] Kiam J J, Schulte A. Multilateral quality mission planning for solar-powered long-endurance UAV[C].2017 IEEE Aerospace Conference. IEEE, 2017: 1-10.

[131] 陈少飞. 无人机集群系统侦察监视任务规划方法[D]. 国防科学技术大学, 2017.

[132] 李月娟, 吕永健, 常迁臻等. 基于 MMDP 的无人作战飞机任务分配模型研究[J]. 计算机应用与软件, 2013, 30(07): 276-279+286.

[133] Jeong B M, Ha J S, Choi H L. MDP-based mission planning for multi-UAV persistent surveillance[C].2014 14th International Conference on Control, Automation and Systems (ICCAS 2014). IEEE, 2014: 831-834.

[134] 霍璐. 旋翼无人机对地目标检测与态势估计方法研究[D]. 中国民航大学, 2017.

[135] Baek S S, Kwon H, Yoder J A, et al. Optimal path planning of a target-following fixed-wing UAV using sequential decision processes[C].2013 IEEE/RSJ International conference on intelligent robots and systems. IEEE, 2013: 2955-2962.

[136] Ragi S, Chong E K P. UAV path planning in a dynamic environment via partially observable Markov decision process[J]. IEEE Transactions on Aerospace and Electronic Systems, 2013, 49(4): 2397-2412.

[137] Vanegas F, Campbell D, Eich M, et al. UAV based target finding and tracking in GPS-denied and cluttered environments[C].2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2016: 2307-2313.

[138] Vanegas F, Campbell D, Roy N, et al. UAV tracking and following a ground target under motion and localisation uncertainty[C].2017 IEEE Aerospace Conference. IEEE, 2017: 1-10.

[139] Vanegas F, Gonzalez F. Uncertainty based online planning for UAV target finding in cluttered and GPS-denied environments[C].2016 IEEE Aerospace Conference. IEEE, 2016: 1-9.

[140] 郭军, 朱凡, 刘远飞. 基于马尔可夫链预测的多无人机协同搜索控制[J]. 弹箭与制导学报, 2007(05): 315-318.

[141] 左青海, 潘卫军, 卓星宇等. 无人机感知与避撞方法研究[J]. 装备制造技术, 2016(11): 65-69.

[142] Fu Y, Yu X, Zhang Y. Sense and collision avoidance of unmanned aerial vehicles using Markov decision process and flatness approach[C].2015 IEEE International Conference on Information and Automation. IEEE, 2015: 714-719.

[143] Krishnamoorthy K, Pachter M, Chandler P, et al. UAV perimeter patrol operations optimization using efficient dynamic programming[C].Proceedings of the 2011 American Control Conference. IEEE, 2011: 462-467.

[144] 陈占海, 祝小平. 无人作战飞机数据链延时对攻击决策的影响及其补偿[J]. 科学技术与工程, 2011, 11(09): 2043-2047.

[145] Vaca F E. National Highway Traffic Safety Administration (NHTSA) notes[J]. Annals of Emergency Medicine, 2004, 43(2): 275-277.

[146] 侯海晶. 高速公路驾驶人换道意图识别方法研究[D]. 长春: 吉林大学, 2013.

[147] 高振海. 驾驶员无意识车道偏离识别方法[J]. 吉林大学学报 (工学版), 2017, 47(3): 709-716. Zeng X, Fu H, Dai B, et al. A new approach for dense depth image recovery[C].2015 7th International Conference on Intelligent Human-Machine Systems and Cybernetics. IEEE, 2015, 2: 491-496.

[148] 谷明琴. 复杂环境中交通标识识别与状态跟踪估计算法研究[D]. 长沙: 中南大学, 2013.

[149] Cuenca Á, Ojha U, Salt J, et al. A non-uniform multi-rate control strategy for a Markov chain-driven Networked Control System[J]. Information Sciences, 2015, 321: 31-47.

[150] Thulasiraman P, Sagir Y. Dynamic bandwidth provisioning using Markov chain based RSVP for unmanned ground networks[C].2014 IEEE International Inter-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA). IEEE, 2014: 130-136.

[151] Thulasiraman P, Clark G A, Beach T M. Mobility estimation using an extended Kalman filter for unmanned ground vehicle networks[C].2014 IEEE International Inter-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA). IEEE, 2014: 223-229.

[152] Kawano H. Real-time obstacle avoidance for underactuated autonomous underwater vehicles in unknown vortex sea flow by the mdp approach[C].2006 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2006: 3024-3031.

[153] Teck T Y, Chitre M. Single beacon cooperative path planning using cross-entropy

method[C].OCEANS'11 MTS/IEEE KONA. IEEE, 2011: 1-6.

[154] Tan Y T, Gao R, Chitre M. Cooperative path planning for range-only localization using a single moving beacon[J]. IEEE Journal of Oceanic Engineering, 2014, 39(2): 371-385.

[155] 温涛, 许枫, 杨娟, 等. 连续隐马尔可夫模型在多基地目标识别中的应用[J]. 应用声学, 2017, 36(6): 512-520.

[156] 吴小勇. 反潜体系的搜索能力优化方法研究[D]. 长沙: 国防科学技术大学, 2012.

[157] Myers V, Williams D P. A POMDP for multi-view target classification with an autonomous underwater vehicle[C].OCEANS 2010 MTS/IEEE SEATTLE. IEEE, 2010: 1-5.

[158] Myers V, Williams D P. Adaptive multiview target classification in synthetic aperture sonar images using a partially observable Markov decision process[J]. IEEE Journal of Oceanic Engineering, 2011, 37(1): 45-55.

[159] 詹艳梅, 孙进才, 胡友峰. 纯方位目标运动分析的自适应算法[J]. 西北工业大学学报, 2002, 20(4): 647-650.

[160] 王彪, 曾庆军, 夏捷. 一种有效的水下目标运动分析算法与仿真研究[J]. 系统仿真学报, 2012, 24(11): 2410-2413.

[161] 陈钊, 刘郑国, 王海燕, 等. 基于 RJMCMC 方法的水下被动目标声源数和方位联合估计[J]. 鱼雷技术, 2011, 19(6): 415-422.

[162] 梁庆卫, 孙天元, 蒋姗姗. 基于马尔可夫链的水下移动网络可靠性研究[J]. 鱼雷技术, 2014, 22(3): 165-168.

[163] 刘友永. 水声差分跳频通信关键技术研究[D]. 哈尔滨: 哈尔滨工程大学, 2009.

[164] 徐君锋. 无线传感器网络中通信可靠性和能量有效性的机制研究[D]. 大连: 大连理工大学, 2011.

[165] 钟贞魁. 基于优先级 QoS 选择策略的水下网络中继算法[J]. 计算机仿真, 2016 (1): 317-320.

[166] 苏毅珊, 金志刚, 窦飞. 高性能水下传感器网络多信道 MAC 协议[J]. 哈尔滨工程大学学报, 2015, 36(7): 987-991.

[167] 张艳莉, 黄军辉. 一种基于马尔可夫链的水下频谱预测方法[J]. 广东轻工职业技术学院学报, 2012, 11(4): 15-19.

[168] 迟凤阳. 水下航行器组合导航定位技术研究[D]. 哈尔滨: 哈尔滨工程大学, 2015.

[169] 陈鹏云. 多传感器条件下的 AUV 海底地形匹配导航研究[D]. 哈尔滨: 哈尔滨工程大学, 2016.

[170] Bellman R E. A Markov decision progress[J]. Journal of Mathematical Mechanics, 1957, 6(5):679-684.

[171] 张忠铝. 基于SAC算法的AUV航路规划与跟踪方法研究[D]. 山东大学, 2023.

[172] Doya K. Reinforcement learning in continuous time and space[J]. Neural computation, 2000, 12(1): 219-245.

[173] Sutton R S, McAllester D, Singh S, et al. Policy gradient methods for reinforcement learning with function approximation[J]. Advances in neural information processing systems, 1999, 12.

[174] Konda V, Tsitsiklis J. Actor-critic algorithms[J]. Advances in neural information processing systems, 1999, 12.

[175] Sutton, R. S., Precup, D., & Singh, S. (1999). Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. Artificial intelligence, 112(1-2), 181-211.

[176] Parr, R., & Russell, S. J. (1998). Reinforcement learning with hierarchies of machines. In Advances in neural information processing systems (pp. 1043-1049).

[177] Dietterich, T. G. (2000). Hierarchical reinforcement learning with the MAXQ value function decomposition. J. Artif. Intell. Res.(JAIR), 13, 227-303.

[178] 周欢, 赵辉, 韩统等. 基于规则的无人机集群飞行与规避协同控制[J]. 系统工程与电子技术, 2016, 38(06): 1374-1382.

[179] 王健. 多架无人机攻击多目标的协同航迹规划算法研究[D]. 西北工业大学, 2004.

[180] Bellingham J, Tillerson M, Richards A, et al. Multi-task allocation and path planning for cooperating UAVs[J]. Cooperative control: models, applications and algorithms, 2003: 23-41.

[181] 胡中华, 赵敏, 姚敏等. 一种改进蚂蚁算法的无人机多目标三维航迹规划[J]. 沈阳工业大学学报, 2011, 33(05): 570-575.

[182] Swartzentruber L, Foo J L, Winer E. Multi-objective UAV path planning with refined reconnaissance and threat formulations[C].51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 18th AIAA/ASME/AHS Adaptive Structures Conference 12th. 2010: 2758.

[183] 史志富. 基于贝叶斯网络的 UCAV 编队对地攻击智能决策研究[D]. 西北工业大学, 2007.

[184] Gupta S, Deep K. A novel random walk grey wolf optimizer[J]. Swarm and

evolutionary computation, 2019, 44: 101-112.

[185] Rodríguez L, Castillo O, Soria J, et al. A fuzzy hierarchical operator in the grey wolf optimizer algorithm[J]. Applied Soft Computing, 2017, 57: 315-328.

[186] Daniel E. Optimum wavelet-based homomorphic medical image fusion using hybrid genetic–grey wolf optimization algorithm[J]. IEEE Sensors Journal, 2018, 18(16): 6804-6811.

[187] Jayabarathi T, Raghunathan T, Adarsh B R, et al. Economic dispatch using hybrid grey wolf optimizer[J]. Energy, 2016, 111: 630-641.

[188] Wang N, He Q. Seagull optimization algorithm combining golden sine and sigmoid continuity[J]. Appl. Res. Comput, 2022, 39: 157-162.

[189] 周瑞, 黄长强, 魏政磊等. MP-GWO 算法在多 UCAV 协同航迹规划中的应用[J]. 空军工程大学学报(自然科学版), 2017, 18(05): 24-29.

[190] 郑志伟. 空空导弹系统概论[M].北京:兵器工业出版社, 1997.

[191] Dai H, MacBeth C. Effects of learning parameters on learning procedure and performance of a BPNN[J]. Neural networks, 1997, 10(8): 1505-1521.

[192] 徐啟云, 王洁, 罗畅等. 无人机对地自主攻击方法仿真与研究[J]. 计算机仿真, 2015, 32(12): 55-59+106.

[193] 罗广彬, 洪成雨, 程志良等. 基于 BP 和 TBPNN 神经网络的混凝土抗压强度预测研究[J]. 混凝土, 2023(03): 37-41.

[194] Ibarz, J., Tan, J., Finn, C., Kalakrishnan, M., Pastor, P., & Levine, S. (2021). How to train your robot with deep reinforcement learning: lessons we have learned. The International Journal of Robotics Research, 40(4-5), 698-721.

[195] 蒲小勃, 缪炜星. 超视距空战中机载雷达的使用策略研究[J]. 电光与控制, 2012, 19(6): 1-4.

[196] 张建东, 王鼎涵, 杨啟明等. 基于分层强化学习的无人机空战多维决策[J]. 兵工学报, 2023, 44(6): 1547.

[197] 王翔. 猝发通信技术在超短波通信系统中的应用[J]. 电子制作, 2013(11):116.

致 谢

岁月流转，时光飞逝，不觉间我就要迎来属于我的研究生毕业季了，在西北工业大学的硕士学习生活中，我收获颇丰，不仅深入学习和研究了相关专业知识，还拓宽了我的眼界，提升了我的认知，增长了我的见识，在这个过程中，我学到了许多宝贵的人生经验和道理。此刻，我衷心想向每一位在我成长道路上给予我帮助和关心的人表达我深深的谢意。

一朝沐杏雨，一生念师恩。感谢我的导师张建东副教授。回想研究生初期，张建东老师的关怀和支持给予了我巨大的帮助。在整个研究生阶段，张老师以其深厚的学识储备指导我解决学术问题，用其严谨的态度向我传授做科研方法，帮助我不断进步。尤其是在我迷茫困惑之时，张老师总是用温暖的话语为我指明了前行的方向，用无尽的耐心引领我成长。此外，感谢吴勇教授，在我申请博士需要推荐信时，尽管吴勇教授工作繁忙，仍然会不遗余力地抽出时间，为我撰写至关重要的信件，并确保及时递交到我心仪的学校，吴老师的这种关怀让我深受感动。感谢史国庆副教授，在我遇到困难时史老师总是会用细腻的关怀和智慧的话语安抚我的不安，并帮助我找到解决问题的途径。这种关心不仅体现在学术指导上，更是一种精神上的支撑，让我感受到了教师这一职业的温暖和伟大。感谢张耀中副教授，在我撰写论文的过程中，老师不厌其烦地为我的论文提出修改建议，张老师一丝不苟的态度以及对学术严谨的坚持都让我受益匪浅。此外，我还要向杨啟明老师表达特别的感激之情，在我的学术旅程中，杨老师一直是我的引路人，带领我不断前行。杨老师温和的性格和鼓舞人心的言语，经常为我注入力量，并激励着我不断提升自己，鼓励我坚定地朝着目标前行。感谢老师们的赋予了我追逐梦想的勇气。师恩重如山巍，令人敬仰；恩情深似海渊，浩渺无垠，难以衡量。在此，我衷心地祝愿老师们身体健康、工作顺利。

感谢硕士期间与我一起成长的朋友们。感谢 430 教研室的每一位同学，是你们让教研室总是洋溢着欢声笑语。我们一起玩乐、一起吃喝、一起打闹、一起通宵.....是你们与我日夜相伴，相互扶持，让我感受到研究生生活也可以过得很有意思。感谢 616A 寝室的室友们——侯佳萍、冯越、张金玲，她们的关爱和包容，总能带给我温馨与力量，让我感受到家一般的温暖和安心，是她们的存在为我的研究生生活增添了无数温暖和美好的记忆。我们一起笑，一起闹，尽管我们的相识不到三年，但我们之间的友谊却宛如经过了十几年的深厚沉淀，仿佛天生就有一种难以言喻的默契。想到不久的将来我们即将分别，我的心中泛起了深深的不舍。感谢我的挚友白双霞，一次偶然的机遇，让共同准备雅思的我们结缘，没想到这成为了我们友谊的开端，将两条原本毫无交集的人生轨

迹紧密相连。我永远无法忘怀我们一起回宿舍的路上开怀大笑的时光；无法忘怀申博期间你对我无微不至的支持与陪伴；无法忘怀在我遇到困难时积极为我思考解决办法并给予我温暖的拥抱。能够与你成为朋友，我感到无比幸运。此外，感谢一直与我相伴的好友董开心，虽然我们相隔遥远，但对彼此的关心却永远不缺席。在我情绪低沉的日子里，是你那无数条关心我的消息点亮了我的世界。如今，看着你在职场上的飞速进步，无论面对何种挑战都显得游刃有余，我真心为你的成就感到欣喜。感谢硕士生涯中结识的每一位朋友，正是你们的加入，让我的研究生之旅绽放出绚烂多姿的色彩。在此，愿各位云程发轫，万里可期。

感谢我的父母，他们为我人生之舟织就温暖的风帆。他们见证了我的每一步成长，他们教导我以正直、善良、真诚之心待人，他们始终不言辛苦地陪伴在我身边，给予我深厚的情感依托和心灵的慰藉，他们是我人生旅途中最可靠的港湾。在漫长的学习旅程中，每次与他们的通电话都如同温暖的阳光，照亮我前行的道路，给我带来无尽的爱和温暖。我衷心希望自己能成为一个让他们引以为傲的女儿，用我的努力和成就来回报他们无私的爱和付出。愿未来的我能够牵着他们的手，共同探索世界的无限风光。

我想向那个平凡又努力的自己道一声感谢，谢谢你的勇敢让我有机会在西北工业大学的硕士论文中书写心声，谢谢你在面对挑战和逆境的时刻从未放弃希望，毅然决然地向前奔跑，谢谢你能够温柔地拥抱并治愈那个偶尔会流泪的自己。我期盼未来的你可以过得更加鲜活和从容，变得更热爱这个世界，也要更爱自己。学着去拥抱自己所有的情绪，接纳不完美的自我。世间的安排，莫非天命，那些得与失，无需过分挂怀。花开花落，人来人去，皆是自然之序。若力不能及，便让一切随风，顺其自然，继续成长！

最后，我要向西北工业大学表达我的深深感谢。在这里我参与过丰富多彩的活动、见证了校园四季的美景：春日的樱花，夏夜的星空，秋季的银杏，以及冬日的飘雪。这些美好的回忆将永远珍藏在我的心中。我衷心祝愿学校未来发展得更加繁荣昌盛。

行文至此，感慨万分。虽纸张有限，但对于各位的感激无限。希望以上每一位，在未来的岁月里幸福、顺遂。

在学期间发表的学术成果和参加科研情况

发表学术论文：

[1] Zhou Xingyu. Evaluation of Autonomous Capability of Ground Attack UAV Based on Hierarchical Analysis Method[C]. International Conference on Autonomous Unmanned Systems. Singapore: Springer Nature Singapore, 2022, 9:1072-1081. (一作，EI Index：20231313822517)

[2] Liu Y, Zhang J, Zhang K, et al. Research on Threat Assessment Method of Formation Cooperative Combat in a Complex Environment[C]//2023 8th International Conference on Computational Intelligence and Applications (ICCIA). IEEE, 2023: 59-63. (本人第四作者 20240715546893)

发明专利：

[1] 张建东，杨啟明，杜梓冰，周星宇等. 无人机自主对地攻击能力评价方法[P].中国专利.申请号： ZL202205096301

参与科研项目：

[1] 中国试飞院，无人机作战自主对地攻击评价体系研究。2021.05-2021.12.

[2] 中航 601 所，所综合通信识别仿真系统。2022.11-2023.04.

[3] 中航 611 所，编队协同智能战术决策模型和互操作设计。2023.04-2023.09.

所获荣誉奖项：

[1] 西北工业大学研究生二等奖学金，2021 年；

[2] 西北工业大学校级优秀共青团员荣誉称号，2021 年；

[3] 西北工业大学研究生二等奖学金，2022 年；

[4] 西北工业大学“微拓”优秀主讲人，2022 年；

[5] 西北工业大学“翱翔杯”研究生电子设计竞赛二等奖，2022 年；

[6] 航天三院“智信杯”大学生创新创业大赛技术攻关类三等奖，2022 年；

[7] 西北工业大学优秀研究生荣誉称号，2022 年；

[8] 西北工业大学研究生录取通知书设计大赛三等奖，2022 年；

[9] 西北工业大学研究生二等奖学金，2023 年；

[10] 西北工业大学校级优秀共青团员荣誉称号，2023 年；

[11] 西北工业大学优秀研究生荣誉称号，2023 年；

[12] 西北工业大学优秀毕业生荣誉称号，2024 年。

西北工业大学

学位论文知识产权声明书

本人完全了解学校有关保护知识产权的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属于西北工业大学。学校有权保留并向国家有关部门或机构送交论文的复印件和电子版。本人允许论文被查阅和借阅。学校可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。同时本人保证，毕业后结合学位论文研究课题再撰写的文章一律注明作者单位为西北工业大学。

本学位论文属于（在以下方框内打“√”）：

□ 保密论文，保密期（　　年　　月　　日至　　年　　月　　日）。

☑ 公开论文。

学位论文作者签名：周星宇　　指导教师签名：张建东

2024 年 3月 13日　　2024 年 3 月 13日

西北工业大学

学位论文原创性声明

秉承学校严谨的学风和优良的科学道德，本人郑重声明：所呈交的学位论文，是本人在导师的指导下进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容和致谢的地方外，本论文不包含任何其他个人或集体已经公开发表或撰写过的研究成果，不包含本人或其他已申请学位或其他用途使用过的成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式表明。

本人学位论文与资料若有不实，愿意承担一切相关的法律责任。

学位论文作者签名：周星宇

2024 年 3月 13日

西北工业大学学位办制