中国科技期刊卓越行动计划项目入选期刊

控制与决策

CONTROL AND DECISION

基于任务分解与强化学习的多平台协同火力分配方法

伍国华, 李冰洁, 袁于斐, 陆志沣

引用本文:
伍国华,李冰洁,袁于斐,陆志沣. 基于任务分解与强化学习的多平台协同火力分配方法[J]. 控制与决策, 2024, 39(5): 1727−1735.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2022.0624

基于任务分解与强化学习的多平台协同火力分配方法

伍国华[1], 李冰洁[1], 袁于斐[1], 陆志沣[2†]

(1. 中南大学 交通运输工程学院，长沙 410075；2. 上海机电工程研究所，上海 201109)

摘 要: 为了有效求解多平台协同火力分配问题,根据“分而治之”的思想,基于任务分解策略将复杂的决策任务分解为子目标平台选择和子平台火力分配两个阶段,通过融合启发式算法和强化学习模型,提出一种新的强化学习求解方法(HARL),并以多平台联合火力打击为作战背景进行实验仿真. 子目标平台选择层根据当前状态,基于强化学习策略选择攻击当前子目标最适合的火力平台;而子平台火力分配层则使用启发式算法为执行攻击任务的平台规划最优的火力分配方案. 实验结果表明,融合启发式算法和强化学习的HARL方法相比于传统的强化学习算法武器消耗量减少15%以上,相比于经典的启发式算法求解时效性提升20%以上,表明该研究成果可为未来求解复杂作战决策问题提供有力的技术支持.

关键词: 多平台协同火力分配；强化学习；任务分解；迭代优化

中图分类号: E837 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2022.0624

引用格式: 伍国华, 李冰洁, 袁于斐, 等. 基于任务分解与强化学习的多平台协同火力分配方法[J]. 控制与决策, 2024, 39(5): 1727-1735.

Multi-platform collaborative firepower allocation method based on task decomposition and reinforcement learning

WU Guo-hua[1], *LI Bing-jie*[1], *YUAN Yu-fei*[1], *LU Zhi-feng*[2†]

(1. College of Information Science and Engineering，Central South University，Changsha 410075，China；2. Shanghai Institute of Mechanical and Electrical Engineering，Shanghai 201109，China)

Abstract: In order to effectively solve the multi-platform collaborative fire allocation problem, this paper decomposes complex decision-making tasks according to the divide conquer frame and task decomposition technology. The paper proposes a novel combination approach of a heuristic algorithm and reinforcement learning (HARL), and carries out simulation experiments on the background of multi-fire platform joint attack. The sub-target platform allocation layer will select the platforms that are most suitable for attacking the current sub-target based on the reinforcement learning model, and the sub-platform fire allocation layer plans the optimal fire allocation plan for the platform executing the attack task based on the heuristic algorithm. Experimental results of simulation examples show that the reinforcement learning algorithm that combines heuristic operators outperforms traditional reinforcement learning algorithms by less than 15%, and improves the solving time by 20% compared with the classical heuristic algorithm. The research results may provide powerful technology support to solve more complex decision problems in the future.

Keywords: multi-platform collaborative fire allocation；reinforcement learning；task decomposition；iterative optimization

0 引 言

协同作战作为现代战争的主要作战模式,需要根据作战任务合理地进行火力资源分配[1],研究复杂作战环境中的协同火力分配问题,对于提高整体作战效能具有重要的现实意义. 火力分配问题是以进攻成本最小化为目标将m种武器最优分配给n个打击点,该问题可视为一个非线性整数规划问题[2-3],同时也是NP-hard问题[4]. 已有许多文献对火力分配问题进行了研究. Olwell等[5]分析了信息在火力分配问题中的重要性; Eckler等[6]给出了火力分配问题的数学模型; Murphey[2]对火力分配问题的有关文献进行了完整的综述; Ahuja等[3]对已有的精确算法和启发式

收稿日期: 2022-04-15；**录用日期:** 2022-11-28.

基金项目: 国家自然科学基金面上项目(62073341)；中南大学研究生科研创新项目(2022ZZTS0750).

责任编委: 刘宝碇.

†通讯作者. E-mail: luzhifeng424@163.com.

算法两类求解算法分别进行了回顾分析; Cai等[7]针对动态火力分配问题进行了文献综述.

传统的火力分配方法主要有精确算法和启发式方法. 精确算法主要适用于一些特定的火力分配问题,如只有一种火力单元[8]或一个目标至多只能由一种武器进行攻击[9]. 精确算法虽然易理解,但是,对于大规模火力分配问题的求解时效性难以保障,因此,现有文献主要使用启发式算法求解大规模火力分配. Khosla[10]使用混合遗传算法求解带时间窗的动态火力分配问题; 于博文等[11]针对多阶段武器协同火力分配问题,引入基于优势度矩阵的非支配排序改进了非支配排序遗传算法III (non-dominated sorting genetic algorithm III, NSGA-III); 孙海文等[12]构建了一个有关目标-火力节点-制导节点的匹配规则库,可快速生成火力分配方案. 虽然使用启发式算法和精确算法进行火力分配已取得了一系列研究成果,但是,这两类方法主要依托于对作战环境信息的提取和挖掘,要求决策人员掌握专业的领域知识来构建数学模型和设计搜索规则,且需要充足的计算时间才能找到较优解. 随着军事智能技术的广泛应用,战争形态正由机械化战争、信息化战争向智能化战争快速演变[13],启发式算法和精确算法已难以应对瞬息万变的战场环境.

为了解决在大规模、信息不完全、多约束战场背景下,传统的火力分配算法无法在短时间内快速决策和动态响应的不足,基于学习的优化方法开始应用于作战决策. 基于学习的优化方法利用离线训练的优势,通过在训练集上学习到一个表征输入数据与解间映射关系的模型,可在新的测试案例上快速给出火力分配方案[14]. 自主学习的特点使得基于学习的优化方法不需要充足的领域知识进行人为的推理设计和调参,可在信息不完全的环境中更加灵活地建模. 因此,这种方法已广泛应用于人类生产生活的各领域,如视频游戏[15-16]、机器人控制[17]以及图像识别[18]等. 按照在训练集上学习方式的不同,基于学习的优化方法可分为监督学习、无监督学习和强化学习3种[19-20]. 监督学习是指模型从带有标签的数据集上进行学习; 无监督学习与监督学习相反,模型是从没有标签的数据集上进行学习; 强化学习则通过模型与环境的自主互动和试错产生相应的数据集用于学习. 3类学习方法中,监督学习和无监督学习相比于强化学习存在泛化性较差的技术瓶颈[21-22],难以适应多变未知的战场环境,因此,强化学习算法更加广泛应用于作战决策. 黄亭飞等[23]使用深度Q网络求解多类型拦截装备复合式反无人机的任务分配问题,并通过实验验证基于学习的优化算法训练的智能体表现更为优异; Luo等[24]基于数据驱动的深度强化学习方法求解导弹的火力分配问题,不同规模的仿真实验结果均表明强化学习算法可产生令人满意的解.

虽然使用强化学习算法进行火力资源分配已被验证可快速有效地求解问题,但是从现有的文献[23, 25]可以看出,当前应用强化学习方法求解的火力分配问题多集中于对单个火力平台的资源分配,不涉及多火力平台间的协同作战. 协同作战作为典型的多智能体学习问题,由于需要对战场态势信息进行综合考虑而导致动作空间巨大,直接应用强化学习求解存在奖励稀疏、训练难收敛等问题. 于博文等[26]针对这一难题,采用“分而治之”的策略,将整个战场的作战决策分为宏观的战役决策和微观的战术决策,有效解决了复杂装备体系下的作战决策问题. 本文根据文献[26]“分而治之”的思想,针对多平台协同火力分配问题提出一种融合启发式算法与强化学习的求解方法——HARL (a combination approach of heuristic algorithm and reinforcement learning). 该方法基于任务分解策略将复杂的决策问题分解为两个求解阶段: 首先利用强化学习模型选择攻击当前目标点的火力平台,然后使用启发式算法对所选择的火力平台进行火力资源分配. 以海空联合协同火力打击为应用场景的多组实验结果表明,所提出HARL方法既能够改善传统启发式方法求解时间长的缺陷,又能够加快强化学习模型的收敛速度,为未来求解多约束、结构复杂的决策优化问题提供新思路.

1 多平台协同火力分配问题描述

多平台协同火力分配问题是一种对装载不同武器单元的多个火力平台进行作战资源协同分配的优化问题,要求在完成对所有目标点攻击的同时使得总武器消耗量最少. 多平台协同火力分配问题需要考虑不同武器单元的射程、对不同目标点的杀伤率以及平台携带的武器数量限制等约束,是一个复杂的多约束优化问题.

本文以海空联合多平台火力打击为作战背景,假设作战单元拥有M个火力攻击平台$\{w_1, w_2, \ldots, w_M\}$,其中变量$w_i$为第$i$个火力平台. 每个平台会携带$W$个火力单元$\{m_1, m_2, \ldots, m_k, \ldots, m_W\}$,不同的火力单元$m_k$具有不同的弹药量约束,$V_{i,k}$为火力平台$w_i$搭载的火力单元$m_k$的最大弹药量. 假设打击区域共有$N$个攻击目标点,分别用$T_j(j = 1, 2, \ldots, N)$表示. 二元决策变量$x_{j,i,k}$表示火力平台$w_i$是否选择

其搭载的火力单元m_k打击第j个目标,若选择攻击,则$x_{j,i,k}=1$;否则为0. $p_{j,i,k}$为火力单元m_k对目标点j造成的损伤率,$v_{j,i,k}$为m_k用于攻击目标j消耗的弹药量. 该问题的数学模型可描述如下:

$$\text{minimize} \sum_{j=1}^{N}\sum_{i=1}^{M}\sum_{k=1}^{W} x_{j,i,k}\times v_{j,i,k}. \tag{1}$$

$$\sum_{i=1}^{M}\sum_{k=1}^{W} x_{j,i,k}\times p_{j,i,k}\geqslant 1,\ \forall j\in[1,N]; \tag{2}$$

$$\sum_{j=1}^{N} x_{j,i,k}\times v_{j,i,k}\leqslant V_{i,k}; \tag{3}$$

$$L_0\leqslant\sqrt{(X_i-X_j)^2+(Y_i-Y_j)^2}\leqslant L_f; \tag{4}$$

$$x_{j,i,k}\in[0,1]. \tag{5}$$

其中:式(1)为该问题的目标函数,即最小化攻击所有目标点所需的武器数量;式(2)规定了目标点的损伤率总和大于等于1时,该目标点才能被击毁;式(3)表示每个平台上所搭载的火力单元均具有各自的容量约束,即该种火力单元攻击所有目标的武器消耗量总和不能超过其最大装载量;式(4)表示每种火力单元m_i具有最近射程L_0和最远射程L_f约束. 各火力单元的位置坐标记为(x_i,y_i),目标点位置记为(x_j,y_j),目标点与火力单元的距离必须在这个射程范围内才能有效攻击. 为了减少变量数目,假设每个武器单元每次只射出1枚弹药,只要在有效射程范围内目标击中率为100 %,同时,不同平台发射的多种火力间不存在冲突约束.

2　基于任务分解的多平台协同火力分配问题求解模型构建

由第1节中的问题描述可知,多平台协同火力分配问题具有决策空间大、约束多、结构复杂的特点. 传统的火力分配问题研究多集中于单个火力平台的资源分配,使用的方法主要为启发式算法. 经典的启发式算法依赖于专家知识设计有效的搜索规则,在求解结构复杂的决策问题时需要足够的计算时间搜索可行解,在实际作战时难以高效地作出决策. 近年来引起广泛关注的强化学习模型通过离线训练,可在应用时快速地给出决策方案,但是,该方法在大规模作战场景下易遇到学习效率低、收敛难的技术瓶颈. 为了弥补两种算法各自的不足,本文分析多平台协同火力分配问题的内在逻辑顺序,基于任务分解策略将强化学习与启发式算法有效结合,提出一种新的强化学习求解方法——HARL方法,该求解方法的数学模型如下式所示:

$$\text{Num}=\sum_{j=1}^{N}\sum_{i=1}^{M} t_{j,i}\sum_{k=1}^{W} u_{j,i,k}\times v_{j,i,k}; \tag{6}$$

$$F(S)\rightarrow\{t_{j,0},t_{j,1},\ldots,t_{j,M}\},\ \forall j\in[1,N]; \tag{7}$$

$$\sum_{i=1}^{M} t_{j,i}\sum_{k=1}^{W} u_{j,i,k}\times p_{j,i,k}\geqslant 1,\ \forall j\in[1,N]; \tag{8}$$

$$\min\Big(\sum_{i=1}^{M} t_{j,i}\sum_{k=1}^{W} u_{j,i,k}\times v_{j,i,k}\Big),\ \forall j\in[1,N]; \tag{9}$$

$$0\leqslant v_{j,i,k}\leqslant V_{j,i,k}; \tag{10}$$

$$V_{j,i,k}=\begin{cases} V_{i,k}^{\max},\ j=0;\\ V_{j-1,i,k}-v_{j-1,i,k},\ j\neq 0;\end{cases} \tag{11}$$

$$t_{j,i}\in[0,1],\ u_{j,i,k}\in[0,1]. \tag{12}$$

其中:式(6)用于计算攻击所有目标点消耗的武器数量,$t_{j,i}$表示是否选择火力平台w_i攻击当前目标点j,该变量是一个二元变量,在子目标平台分配层由强化学习模型确定,如式(7)所示;F为强化学习模型;S为当前环境的状态,包括各火力平台的位置、弹药剩余情况以及当前目标点的位置;二元变量$u_{j,i,k}$表示平台w_i是否选择火力单元m_k攻击j,变量$v_{j,i,k}$为m_k用于攻击的弹药数量,这两个变量均由子平台火力分配层在式(8)的约束下,由式(9)的优化目标进行确定,$p_{j,i,k}$为火力单元m_k对目标点j的杀伤概率;每种火力单元在进行火力分配时均会受到式(10)的约束,当前时刻的最大容量$V_{j,i,k}$由式(11)进行更新;在初始时刻,每种火力单元有自己的最大弹药量$V_{i,k}^{\max}$,后续随着对目标点的攻击,最大弹药量随之下降;$V_{j-1,i,k}$为攻击上一目标点$j-1$时的最大弹药量;$v_{j-1,i,k}$为攻击$j-1$消耗的武器量.

HARL方法通过将复杂的火力分配问题划分为两个子任务层:子目标平台分配层和子平台火力分配层,首先使用强化学习算法在子目标平台分配层为攻击的目标选择火力平台,然后使用启发式算法为所选择的平台进行火力资源的分配,HARL的求解框架如图1所示.

3　HARL算法设计

3.1　算法介绍

3.1.1　强化学习算法概述

本文使用强化学习算法中的DQN算法(deep Q network, DQN)为每个攻击目标选择最佳的火力平台,DQN算法是一种基于值函数近似法的强化学习方法,通过调整参数θ使得预测网络预测的动作值函数$Q(s,a,\theta)$逼近实际动作值函数$Q_{\text{target}}(s,a,\theta')$.

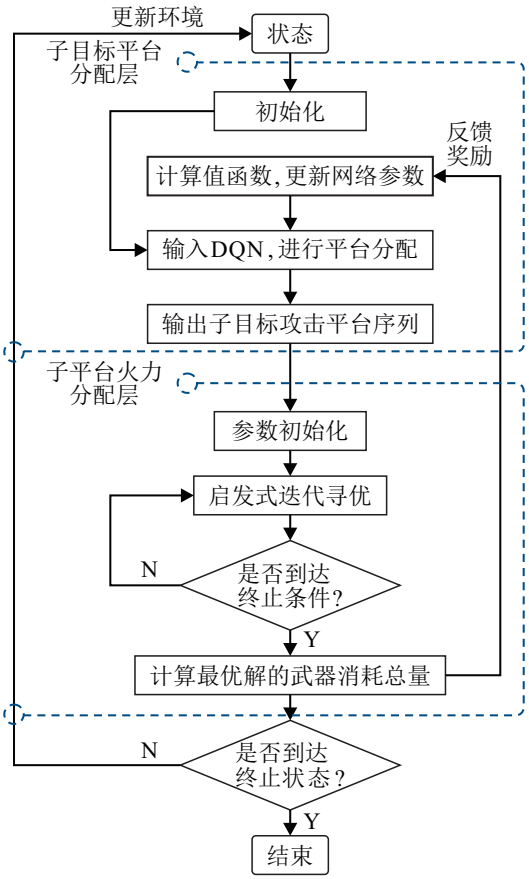

图 1 基于任务分解的多平台协同火力分配问题求解框架

DQN 利用深度学习可自行提取数据特征的优势,基于如下深度神经网络拟合动作的值函数,并选择值函数最大的动作作为输出:

$$Q^*(s,a)\leftarrow$$
$$Q(s,a)+\alpha[r+\gamma\max_{a'}Q(s',a')-Q(s,a)]. \quad (13)$$

DQN 算法作为最早的深度强化学习算法之一,通过融合深度神经网络和 Q-learning 策略,充分结合了深度学习的表征能力和强化学习的决策优势,使得模型展现出良好的学习能力[27],也已被验证可有效求解离散优化问题.

由式(13),DQN 的损失函数可定义为

$$L(\theta)=E[(Q_{\text{target}}-Q(s,a,\theta))^2]. \quad (14)$$

其中:θ 为预测网络的权重参数,θ' 为目标网络的参数,目标 Q 值为

$$Q_{\text{target}}=r+\gamma\max_{a}Q(s',a',\theta'). \quad (15)$$

在获得损失函数后,可直接采用梯度下降算法对神经网络模型损失函数 $L(\theta)$ 的权重参数 θ 进行求解.

本文使用两个全连接的神经网络搭建强化学习模型:用于评估当前的状态-动作值函数 $Q(s,a,\theta)$ 的网络称为预测网络,另一个用于产生目标 Q 值 Q_{target} 的网络称为目标网络. 目标网络输出当前状态的目标 Q 值 Q_{target},预测网络则根据式(14)和 Q_{target} 更新自身参数 θ,为了避免由于目标网络参数变化导致模型的震荡和发散,在多步迭代后才会将预测网络的参数 θ 赋值给目标网络,以增强模型的稳定性.

3.1.2 启发式算法概述

根据搜索方法的不同,可用于火力分配的启发式算法包括 3 类[28]: 1) 基于贪婪准则的启发式算法,这类算法结构简单,求解速度快,但是易陷入局部最优. 2) 基于种群进化的元启发式算法,如遗传算法、差分进化算法等. 进化算法通过模拟生物在自然界中的进化,具有自适应的特点,保留求解质量好的可行解作为父代产生子代,具有高鲁棒性和强探索性的特点. 3) 基于单点随机搜索的元启发式算法,包括禁忌搜索、大规模邻域搜索算法等. 这类算法的搜索规则具有策略性,增加了对搜索算子效果的评估和考量. 本文从以上 3 类启发式算法中,选取具有代表性的贪婪算法[29]、大规模邻域搜索算法[30] 以及遗传算法[31] 用于子平台火力分配层.

1) 贪心算法.

贪心算法(greedy algorithm, G)是最简单的启发式算法之一. 贪心算法从初始解开始,基于迭代的方法,从一代到下一代寻求更好的解决方案[29]. 贪心算法包含破坏、重建以及最优性评判 3 个步骤. 首先,贪心算法会随机构造初始解 $S_0=\{v_0,v_1,\dots,v_k\}$,随机删除 S_0 中的几个元素,得到不可行解 S_0';然后,从删除的元素中重新选择元素插入 S_0',直至 S_0' 变为新的可行解 S';最后,基于接受准则,进行最优性判断,若 $F(S_0)>F(S')$,则令 $S_0\leftarrow S'$.

2) 遗传算法.

遗传算法(genetic algorithm, GA)模仿了适者生存的方式,是一种基于自然选择和遗传机制的启发式搜索方法[32],在每次迭代优化时会根据适应度函数的大小选择优良父代解产生新的邻域解集. 适应度函数是指通过指定一个与种群生存能力成比例的数值适应度值来评估当前种群子代的优良[31],适应度函数值 f_i 一般是基于其目标函数值 f 设定,在多平台协同火力分配问题的应用背景下遗传算法的目标函数定义为 $f=\min\sum_{i=1}^{M}t_{j,i}\sum_{k=1}^{W}u_{j,i,k}\times v_{j,i,k}$. 其中:$v_{j,i,k}$ 为当前分配方案中每种火力单元的消耗数量,并规定适应度函数 $f_i=1/f$. 除了目标函数和适应度函数两个基本函数,遗传算法还包括选择、交叉和突变 3 个操作算子. 其中:选择操作通过比较子代的适应度大小,按照一定的规则和方法选择当前种群中最优势的一些个体遗传到下一代种群中,如最常见的“轮盘

赌”[33]选择方法,利用每个个体适应度大小决定被选择的可能性p_i,有

$$p_i = f_i \Big/ \sum_{i=1}^{\mathrm{NP}} f_i. \tag{16}$$

3) 大规模邻域搜索算法.

大规模邻域搜索算法(large-scale neighborhood search algorithm, LNS)是Shaw[34]于1997年提出的一种新的邻域搜索算法. 传统的邻域搜索算法通常只探索邻近区域,这些区域可相对较快地完成解的寻找,但是易陷入局部最优解. 与这些算法相比,大规模邻域搜索算法基于目标函数最大化原则[35],可探索更广阔的解空间. 大规模邻域搜索算法主要包括“破坏”和“修复”两个操作,“破坏”算子通过随机移除部分元素对当前解进行破坏,产生部分不可行解;而“修复”算子则通过插入新的元素使得不可行解变成可行解.

3类启发式算法通过综合考虑各火力单元的射程约束、数量约束以及对不同目标点的杀伤率3个准则设计搜索规则,并以完成对目标点的完全攻击(各火力单元杀伤率的累计和达到1)为基本要求,按照火力单元的总消耗数量构造评价函数.

3.2　HARL算法流程

HARL算法基于任务分解策略,将复杂的多火力平台协同作战问题按照“先分配再执行”的原则进行分阶段处理. HARL算法的构建包含状态、动作、动作策略以及奖励4个要素,该4要素的定义和作用流程如下.

1) 状态: 首先将当前环境信息作为强化学习模型的状态输入,该信息包含目标点的位置、M个火力平台的位置以及各火力资源的剩余弹药量和对当前目标点的毁伤率,即

$$\begin{aligned} S_j = \{&(x_j, y_j), [(x_1, y_1), (x_2, y_2), \ldots, (x_M, y_M)], \\ &(p_{j,i,1}, p_{j,i,2}, \ldots, p_{j,i,W}), [V_{j,i,1}, V_{j,i,2}, \ldots, V_{j,i,W}]\}. \end{aligned} \tag{17}$$

其中: (x_j, y_j)为目标点j所处的地理位置,$[(x_1, y_1), (x_2, y_2), \ldots, (x_M, y_M)]$为各火力平台的位置,$(p_{j,i,1}, p_{j,i,2}, \ldots, p_{j,i,W})$为不同武器单元对该目标点的毁伤率,$[V_{j,i,1}, V_{j,i,2}, \ldots, V_{j,i,W}]$为不同武器单元当前的最大弹药余量.

2) 动作: 在输入状态后,模型会根据动作选择策略$\pi(a|s)$输出动作a_j,a_j被定义为选择哪几个平台对当前目标点j进行攻击. 为了便于编码,本文使用one-hot的2进制编码方式: 该数位为1表示选择该平台进行攻击,为0则表示不选择,有

$$a_j = \{t_{j,0}, t_{j,1}, \ldots, t_{j,M}\},\ t_{j,i} \in [0, 1]. \tag{18}$$

3) 动作策略: 为了平衡“探索”与“利用”,增强模型搜索过程中的随机性,采用ϵ-greedy动作策略,有

$$\pi(a|s) = \begin{cases} \arg\max\limits_{a} Q^{\pi}(S, a),\ \text{概率为} 1-\epsilon; \\ a,\ \text{概率为} \dfrac{\epsilon}{|A|}. \end{cases} \tag{19}$$

其中: $Q^{\pi}(S, a)$为神经网络计算出当前时刻状态下每个动作对应的值函数,有$1-\epsilon$的概率选择其中值函数最大的动作;同时为了学到新的知识,智能体也有$\dfrac{\epsilon}{|A|}$的可能性随机从动作空间中挑选动作a,ϵ可根据动作空间$|A|$的大小进行调整,动作空间越大,该参数越大.

4) 奖励: 智能体根据第2节的数学模型,以式(9)为目标函数,在式(8)和(10)的约束下,为所选择的子平台调用启发式算法分配火力资源,并将消灭目标点j所需要的武器数量的负值定义为执行动作a_j后的奖励r_j,有

$$r_j = -\sum_{i=1}^{M} t_{j,i} \sum_{k=1}^{W} u_{j,i,k} \times v_{j,i,k}. \tag{20}$$

由第2节数学模型可知,模型的目标函数是使得最后消耗的弹药数量最少,因此,以消耗弹药量的负值作为奖励会促使智能体选择弹药数量消耗少的分配方案,从而不断优化模型参数,提升模型所规划火力分配方案的优化性.

为了便于实验中进行对比分析,本文根据子目标平台分配层使用的强化学习算法以及子平台火力分配层使用的启发式算法,对用于实验的3种HARL求解方法分别命名为HDQNG(a hybrid of deep-Q network with greedy algorithm)、HDQNGA(a hybrid of deep-Q network with genetic algorithm)以及HDQNLNS(a hybrid of deep-Q network with large neighborhood search algorithm).

3.3　算法复杂性分析

在线测试时,HARL方法中的子目标平台分配层通过离线训练,可直接选出攻击子目标的火力平台,N个目标点的计算复杂度为$O(N)$. 子平台火力分配层需要为子目标平台分配层选择的攻击平台进行火力分配,假设作战单元共有M个火力平台,每个火力平台搭载有W个火力单元,则子平台火力分配层进行分配的火力单元总数为$m \times W$. 其中: m表示子目标平台分配层选择了m个平台攻击当前目标点,$0 \leqslant m \leqslant M$. 此外,不同启发式算法根据搜索规则的不同,计算复杂度存在差异. 如在HDQNG方法中,

贪婪算法求解时计算复杂度只受到要分配的火力单元总数的影响,N个目标点的计算复杂度为$O(N \times m \times W)$;HDQNGA中的遗传算法除了受到火力单元总数的影响,还会受到种群大小G、进化代数P的影响,可得到计算复杂度为$O(N \times G \times P \times m \times W)$;大规模邻域搜索算法的计算复杂度取决于迭代次数Num、每次迭代中修复算子和破坏算子执行的次数Pu以及火力单元总数$m \times W$,因此,HDQNLNS的计算复杂度为$O(N \times \text{Num} \times \text{Pu} \times m \times W)$. 若使用单一的启发式算法,如蚁群算法,则计算时间复杂度取决于蚂蚁个数A_{num}、迭代次数Num以及要分配的火力单元总数. 值得注意的是,单一启发式算法每次迭代求解时需要分配的是所有火力平台的火力单元总数$M \times W$,其中$M \gg m$,最终的计算复杂度为$O(N \times M \times W \times A_{\text{num}} \times \text{Num})$.

由以上的对比分析可知,HARL求解方法利用任务分解策略减小了启发式算法每次分配火力资源时需要遍历的火力单元总数,启发式算法只需要对确定的火力平台进行资源的组合分配,要规划的火力单元数量从$M \times W$变为$m \times W$,从而有效地减小了算法的计算复杂度.

4 实验仿真

4.1 实验设置

本文在配置为Core i7-9800x 3.8 GHz CPU,16 GB内存,单GPU 2080 Ti,Windows 10操作系统的计算机上,使用Python 3进行实验仿真. 以复杂约束环境下的多平台协同火力打击问题为仿真实验背景,在海空联合作战典型场景下对所提出算法的性能进行了评估验证. 仿真实验的目标是击毁目标区域(北纬[25°, 30°],东经[90°, 110°])的多个战略驻地,以实现对该区域的控制. 在仿真实验中,我方作战平台主要是多艘战舰和多架战机,其中战舰分别部署于A地外海(27°N, 95°E)和B地以北(27°N, 105°E),战机均是从C地(29°N, 105°E)起飞,各武器平台上分别携带数枚不同的火力单元. 战舰上主要配置弹头1-1、弹头1-2和弹头1-3,战机携带数枚弹头2-1、弹头2-2、弹头2-3以及弹头2-4,各火力单元的射程范围和歼击机的航程约束如表1所示.

表1 火力单元参数表

武器名称	武器代号	射程约束/km	搭载平台
弹头1-1	1	[300, 2 000]	a类舰船, b类舰船
弹头1-2	2	[400, 1 500]	a类舰船
弹头1-3	3	[500, 2 000]	a类舰船
弹头2-1	4	[50, 200]	b类战机
弹头2-2	5	[100, 300]	b类战机, a类战机
弹头2-3	6	[100, 200]	b类战机
弹头2-4	7	[50, 300]	b类战机

本文基于上述复杂环境下多火力平台协同目标打击的作战想定,通过使用改变目标点数量,作战平台数量及其携带的火力单元数量,共生成9种测试案例,各测试案例的具体设置如表2所示.

表2 测试案例武器平台和弹药量明细

	a类舰船				b类舰船		b类战机					a类战机		目标点数量
	数量	火力单元			数量	火力单元	数量	火力单元				数量	火力单元	
		1	2	3		1		4	5	6	7		5	
exp.1	2	24	24	0	0	0	2	4	4	0	0	0	0	8
exp.2	2	24	24	0	0	0	2	4	4	0	0	0	0	9
exp.3	2	24	24	0	0	0	2	4	4	0	0	0	0	12
exp.4	2	24	24	0	0	0	2	4	4	0	0	0	0	16
exp.5	4	30	30	0	0	0	2	4	4	0	0	0	0	16
exp.6	4	24	22	20	0	0	4	4	0	0	0	0	0	16
exp.7	4	24	22	20	0	0	2	4	4	4	4	0	0	16
exp.8	2	24	24	0	2	24	2	4	4	0	0	0	0	8
exp.9	2	24	24	0	0	0	2	4	4	0	0	3	4	8

为了全面比较所提出HARL算法的性能,本文从时效性和优化性两方面进行测试,并以经典强化学习模型DQN以及蚁群优化算法(ant colony optimizaion algorithm, ACO)和模拟退火方法(simulate anneal algorithm, SA)两种有效的传统优化算法作为对比算法,并通过多次实验,确定各对比算法的最优参数设置. 实验设计可分为以下两方面.

1) 为了验证所提出算法模型的学习能力,对比所提出3种HARL算法(HDQNG、HDQNGA、HDQNLNS)与DQN的学习曲线.

2) 为了验证所提出算法模型的时效性和最优化,将3种HARL算法与ACO算法、SA算法、DQN算法

在 9 种测试案例上进行实验对比.

4.2　实验结果对比和分析

收敛曲线是分析强化学习模型学习能力的一种重要手段, 曲线收敛得越快, 模型的学习能力越强; 曲线的损失值越小, 模型在训练样本上的优化性越高. 本文对比了所提出 3 种 HARL 算法与单一的 DQN 算法在经过 100 回合训练的收敛曲线, 训练结果如图 2 所示. 由图 2 可见, 所提出 3 种算法在相同训练环境下迭代相同回合数后, 均能够很快收敛. 另外值得注意的是, HDQNG 虽然收敛后的损失值略高于 DQN, 但是, 单一的 DQN 模型在训练过程中不稳定, 训练曲线震荡较严重, HARL 算法明显要比单一的 DQN 模型更加稳定. 因此可得到如下结论: 面对大规模、多约束的复杂决策问题, 单一的强化学习模型易受到动作空间过大的影响, 收敛速度慢甚至不稳定, 学习效率低下. 通过结合启发式算法, 能够极大地提高强化学习模型的学习能力和收敛速度. 此外, 对比 3 类 HARL 算法可以发现, HDQNG 相比于 HDQNGA 与 HDQNLNS 的学习曲线更为平缓, 但是, 收敛后的损失值也更高. 这一现象是由于贪心算法相比于遗传算法与大规模邻域算法搜索规则更简单, 使得它对解空间的搜索能力也相对更弱, 能够较快地收敛于局部最优解. 遗传算法和大规模邻域算法能够探索到更广阔的解空间, 有助于寻找更有潜力的解, 从而在经过相同训练回合后模型性能更好.

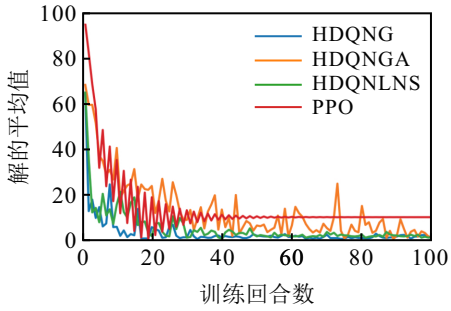

图 2　强化学习算法训练曲线

除了对比 HARL 算法的学习能力, 本文还对比了所提出 HARL 算法与 3 种对比算法在测试案例上的求解时间以及所消耗武器单元的总数. 图 3 为 6 种算法在 exp.1 ~ exp.3 上的求解时间对比. 由图 3 可见, 随着目标点数量的增多, 6 种算法的求解时间均有所增加. 相比于两类启发式算法, 强化学习模型的求解时间要远小于它们. 如在 exp.1 上: HDQNG 的求解耗时为 16.2 s, HDQNGA 为 20.4 s, HDQNLNS 为 31.3 s, DQN 为 9.6 s, SA 却需要 92 s, ACO 需要 113.4 s. 通过简单计算可知, 所提出 HDQNG 算法在 exp.1 上的计算耗时约为 ACO 耗时的 10.4 % 和 SA 耗时的 8.4 %; 3 类 HARL 算法中耗时最长的 HDQNLNS 算法也仅为 ACO 耗时的 27.6 % 和 SA 耗时的 34.1 %. 此外, 在 3 个仿真实例上, HARL 方法的求解时间均小于等于 1 min, 由此验证了强化学习模型具有良好的时效性. 综上所述, HARL 算法在计算耗时上均远低于传统的启发式算法, 计算时效性的提升超过 20 %.

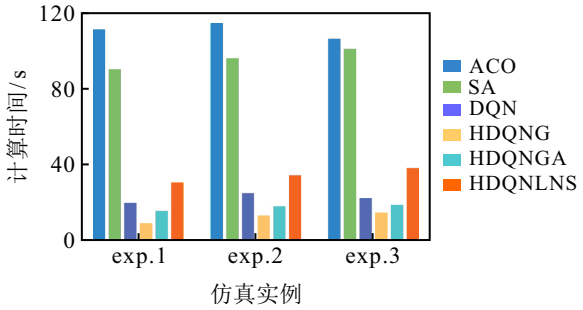

图 3　各实验算法在测试案例上计算时间对比

为了更加有力地验证所提出算法的优异性, 表 3 中展示了所有测试算法在测试案例上火力规划所消耗的武器数量. 由表 3 可见, 3 种 HARL 算法在 9 个测试案例上的结果均为最优, 在摧毁敌方所有目标时能够消耗更少的武器数量. 以 exp.6 为例, HDQNG 的武器消耗量比 DQN 降低了 16.7 %, 比 ACO 降低了 35.2 %, 比 SA 降低了 7.9 %; HDQNLNS 的武器消耗量比 DQN 减少了 21.4 %, 比 ACO 减少了 38.9 %, 比 SA 减少了 13.2 %; HDQNGA 的武器消耗量比 DQN 下降了 23.8 %, 比 ACO 下降了 40.7 %, 比 SA 下降了 15.8 %.

表 3　实验算法在不同测试案例上决策所消耗的武器数量

实验名	ACO	DQN	SA	HDQNG	HDQNGA	HDQNLNS
exp.1	24	30	26	16	17	16
exp.2	28	40	31	21	18	17
exp.3	38	49	38	27	24	25
exp.4	40	38	39	32	32	32
exp.5	37	55	42	33	32	32
exp.6	54	42	38	35	33	32
exp.7	39	47	37	32	32	32
exp.8	28	32	31	25	19	18
exp.9	27	38	33	20	21	20

图 4 为不同算法在仿真实验上的武器消耗量对

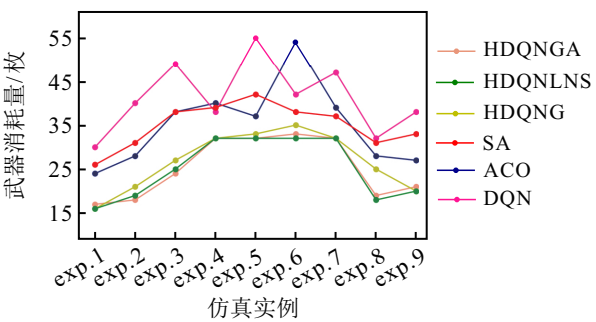

图 4　不同算法在仿真实验上的武器消耗量对比

比. 由图4可见,3种HARL算法对应的武器消耗量曲线始终处于图表的最下方,与其他算法具有很大的差距. HARL方法通过结合启发式算法,使得火力分配方案的攻击成本相比于单一的DQN模型下降了15%左右.

此外,为了进一步比较算法的整体性能,本文进行了Freidmen非参数检验. 图5为不同算法在仿真实验结果的Freidmen检验. 图5中:各算法对应的小横条越靠左,表明算法性能越好;两种算法的横条的重叠区域越多,表明这两种算法的性能越接近. 由图5可见,3种HARL算法的性能更优. 其中:HDQNLNS性能最优,而HDQNGA性能稍逊,HDQNG算法性能是3种HARL算法中最差的.

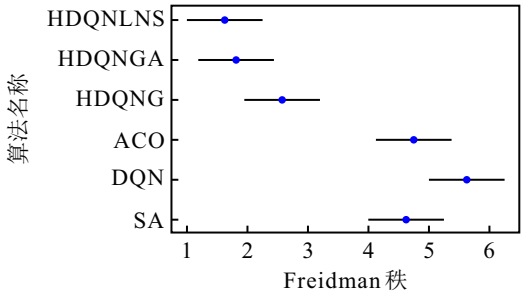

图5 不同算法在仿真实验结果的Freidmen检验

5 结 论

复杂作战环境下的多平台火力协同分配是作战决策中一个亟待解决的问题,直接采用单一的强化学习模型存在奖励稀疏、训练难收敛等技术瓶颈,而传统启发式方法的求解时效性难以保证. 因此,本文基于任务分解策略,提出了一种结合启发式算法的强化学习求解方法,本文的研究工作具有以下几点重要意义.

1) 基于任务分解策略结合启发式算法和强化学习的求解方法,能够有效地加快强化学习模型的收敛速度,同时减小计算复杂度,相比于经典的启发式算法求解时间缩短了20%以上. 一系列仿真实验表明,基于任务分解策略的HARL算法能够更加有效地应用于实际作战环境.

2) 本文首次探索了结合启发式算法和强化学习求解复杂决策问题的可能性,并通过实验验证了融合不同的启发式算法会对所提出HARL方法的求解性能造成一定的影响. 本文的研究工作可令读者对算法融合技术有所启发,提出更有效的融合算法解决不同领域的决策问题.

参考文献(References)

[1] Rai R N, Bolia N. Optimal decision support for air power potential[J]. IEEE Transactions on Engineering Management, 2014, 61(2): 310-322.

[2] Murphey R A. Target-based weapon target assignment problems[M]. Boston: Combinatorial Optimization, 2000: 39-53.

[3] Ahuja R K, Kumar A, Jha K C, et al. Exact and heuristic algorithms for the weapon-target assignment problem[J]. Operations Research, 2007, 55(6): 1136-1146.

[4] Lloyd S P, Witsenhausen H S. Weapons allocation is NP-complete[C]. Proceedings of the IEEE Summer Simulation Conference. Reno, 1986: 1054-1058.

[5] Olwell D, Washburn A. Internetting of fires[R]. Monterey: Naval Postgraduate School, 2002: 1-35.

[6] Eckle A R, Burr S A. Mathematical models of target coverage and missile allocation[M]. Alexandria: Military Operations Research Society, 1972: 1-282.

[7] Cai H P, Liu J X, Chen Y W, et al. Survey of the research on dynamic weapon-target assignment problem[J]. Journal of Systems Engineering and Electronics, 2006, 17(3): 559-565.

[8] Denbroeder G G J, Ellison R E, Emerling L. On optimum target assignments[J]. Operations Research, 1959, 7(3): 322-326.

[9] Chang S C, James R M, Shaw J J. Assignment algorithm for kinetic energy weapons in boost phase defence[C]. The 26th IEEE Conference on Decision and Control. Los Angeles, 2007: 1678-1683.

[10] Khosla D. Hybrid genetic approach for the dynamic weapon-target allocation problem[C]. Aerospace/Defense Sensing, Simulation, and Controls. Orlando, 2001: 244-259.

[11] 于博文, 吕明. 基于D-NSGA-GKM算法的多阶段武器协同火力分配方法[J]. 控制与决策, 2022, 37(3): 605-615.
(Yu B W, Lv M. Optimization method for multi-stage collaborative weapon firepower distribution based on D-NSGA-GKM algorithm[J]. Control and Decision, 2022, 37(3): 605-615.)

[12] 孙海文, 谢晓方, 庞威, 等. 基于改进火力分配模型的综合防空火力智能优化分配[J]. 控制与决策, 2020, 35(5): 1102-1112.
(Sun H W, Xie X F, Pang W, et al. Integrated air defense firepower intelligence optimal assignment based on improved firepower assignment model[J]. Control and Decision, 2020, 35(5): 1102-1112.)

[13] 易凯, 张修社, 韩春雷, 等. 分布式智能作战决策应用发展与关键技术[J]. 现代导航, 2021, 12(1): 46-51.
(Yi K, Zhang X S, Han C L, et al. Application development and key technologies of distributed intelligent operation decision-making[J]. Modern Navigation, 2021, 12(1): 46-51.)

[14] Li B, Wu G, He Y, et al. An overview and experimental study of learning-based optimization algorithms for vehicle routing problem[J/OL]. 2021, arXiv: 2107.07076.

[15] Bellemare M G, Naddaf Y, Veness J, et al. The

arcade learning environment: An evaluation platform for general agents[J/OL]. 2012, arXiv: 1207.4708.

[16] Silver D, Huang A, Maddison C J, et al. Mastering the game of Go with deep neural networks and tree search[J]. Nature, 2016, 529(7587): 484-489.

[17] Levine S, Finn C, Darrell T, et al. End-to-end training of deep visuomotor policies[J]. The Journal of Machine Learning Research, 2016, 17(1): 1334-1373.

[18] Xu K, Ba J L, Kiros R, et al. Show, attend and tell: Neural image caption generation with visual attention[C]. Proceedings of the 32nd International Conference on Machine Learning. Lille, 2015: 2048-2057.

[19] Ayodele T O. Types of machine learning algorithms[J]. New Advances in Machine Learning, 2010, 3: 19-48.

[20] Bishop C M, Nasrabadi N M. Pattern recognition and machine learning[M]. New York: Springer, 2006: 1-32.

[21] Kaelbling L P, Littman M L, Moore A W. Reinforcement learning: A survey[J]. Journal of Artificial Intelligence Research, 1996, 4: 237-285.

[22] Sutton R S, Barto A G. Reinforcement learning: An introduction[J]. IEEE Transactions on Neural Networks, 1998, 9(5): 1054.

[23] 黄亭飞, 程光权, 黄魁华, 等. 基于DQN的多类型拦截装备复合式反无人机任务分配方法[J]. 控制与决策, 2022, 37(1): 142-150.
(Huang T F, Cheng G Q, Huang K H, et al. Task assignment method of compound anti-drone based on DQN for multi type interception equipment[J]. Control and Decision, 2022, 37(1): 142-150.)

[24] Luo W L, Lv J H, Liu K X, et al. Learning-based policy optimization for adversarial missile-target assignment[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2022, 52(7): 4426-4437.

[25] 阎栋, 苏航, 朱军. 基于DQN的反舰导弹火力分配方法研究[J]. 导航定位与授时, 2019, 6(5): 18-24.
(Yan D, Su H, Zhu J. Research on fire distribution method of anti-ship missile based on DQN[J]. Navigation Positioning and Timing, 2019, 6(5): 18-24.)

[26] 于博文, 吕明, 张捷. 基于分层强化学习的联合作战仿真作战决策算法[J]. 火力与指挥控制, 2021, 46(10): 140-146.
(Yu B W, Lv M, Zhang J. Joint operation simulation decision-making algorithm based on hierarchical reinforcement learning[J]. Fire Control & Command Control, 2021, 46(10): 140-146.)

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

[28] Cordeau J F, Laporte G. Tabu search heuristics for the vehicle routing problem[M]. Boston: Kluwer Academic Publishers, 2005: 145-163.

[29] Kang Q M, He H, Song H M. Task assignment in heterogeneous computing systems using an effective iterated greedy algorithm[J]. Journal of Systems and Software, 2011, 84(6): 985-992.

[30] Quimper C G, Rousseau L M. A large neighbourhood search approach to the multi-activity shift scheduling problem[J]. Journal of Heuristics, 2010, 16(3): 373-392.

[31] Eun Y, Bang H. Cooperative task assignment/path planning of multiple unmanned aerial vehicles using genetic algorithm[J]. Journal of Aircraft, 2009, 46(1): 338-343.

[32] Bai X S, Yan W S, Ge S S, et al. An integrated multi-population genetic algorithm for multi-vehicle task assignment in a drift field[J]. Information Sciences, 2018, 453: 227-238.

[33] Mitchell M. An introduction to genetic algorithms[M]. Cambridge: MIT Press, 1998: 1-221.

[34] Shaw P. A new local search algorithm providing high quality solutions to vehicle routing problems[R]. Glasgow: University of Strathclyde, 1997.

[35] Schrimpf G, Schneider J, Stamm-Wilbrandt H, et al. Record breaking optimization results using the ruin and recreate principle[J]. Journal of Computational Physics, 2000, 159(2): 139-171.

作者简介

伍国华(1986−), 男, 教授, 博士生导师, 从事智能优化与决策等研究, E-mail: guohuawu@csu.edu.cn;

李冰洁(1998−), 女, 硕士生, 从事强化学习、决策优化等研究, E-mail: csulbj@163.com;

袁于斐(1998−), 男, 硕士生, 从事运筹优化、物流调度等研究, E-mail: yyf1403@163.com;

陆志沣(1980−), 男, 研究员, 从事系统仿真与智能决策等研究, E-mail: luzhifeng424@163.com.