



DOI:10.12404/j.issn.1671-1815.2405560

引用格式: 向征, 吴秋玥, 储同, 等. 基于深度强化学习的停机位分配[J]. 科学技术与工程, 2025, 25(16): 6977-6984.

Xiang Zheng, Wu Qiu-yue, Chu Tong, et al. Gate assignment based on deep reinforcement learning[J]. Science Technology and Engineering, 2025, 25(16): 6977-6984.

基于深度强化学习的停机位分配

向征, 吴秋玥, 储同, 岳伊杨

(中国民用航空飞行学院空中交通管理学院, 德阳 618000)

摘要 针对停机位分配问题展开系统研究, 目标是 minimized 远机位分配数量以及近机位空闲时间, 针对其多目标多约束特性, 提出以最小远机位分配数量和最小近机位空闲时间为目标的多目标数学模型, 该模型考虑了航班进出港实际时间、机型类别及停机位间相互关系等参数。结合深度强化学习方法, 特别是深度确定性策略梯度算法(deep deterministic policy gradient, DDPG), 对停机位分配过程进行优化。为提升算法的寻优能力与性能, 设计了改进后的 DDPG 算法, 融入优先级经验回放和多策略融合探索机制。通过对比实验, 表明改进后的算法更优, 显著减少了最小远机位分配数量并优化了近机位空闲时间, 且收敛更快、全局寻优能力更强, 充分证实了其有效性。

关键词 停机位分配; 深度学习; 强化学习; 深度确定性策略梯度算法 (DDPG)

中图分类号 V355; **文献标志码** A

Gate Assignment Based on Deep Reinforcement Learning

XIANG Zheng, WU Qiu-yue, CHU Tong, YUE Yi-yang

(College of Air Traffic Management, Civil Aviation Flight University of China, Deyang 618000, China)

[Abstract] A systematic study was conducted on the issue of gate assignment, with the goal of minimizing the number of remote gate assignments and the idle time of near gates. A multi-objective mathematical model was proposed to address the multi-objective and multi-constraint characteristics of the problem. The model was designed to minimize the number of remote gate assignments and the idle time of near gates while taking into account parameters such as actual flight arrival and departure times, aircraft types, and the interrelationships among gates. The gate assignment process was optimized using the deep reinforcement learning method, specifically the deep deterministic policy gradient (DDPG) algorithm. To enhance the optimization ability and performance of the algorithm, an improved DDPG algorithm was developed by incorporating prioritized experience replay and multi-strategy exploration mechanisms. Comparative experiments were conducted, and the results show that the improved algorithm significantly reduces the number of remote gate assignments and optimized time utilization. The algorithm also achieves faster convergence and stronger global optimization capabilities, confirming its effectiveness.

[Keywords] gate assignment; deep-learning; reinforcement-learning; deep deterministic policy gradient (DDPG) algorithm

随着全球经济的快速发展, 航空运输业在综合交通体系中的地位日益凸显, 机场停机位分配作为航空运营的重要环节, 直接影响着机场的运行效率和服务质量。近年来, 伴随着航班量的持续增长, 机场停机位资源愈加紧张, 停机位分配问题也变得越来越复杂。因此, 如何优化停机位分配以提高机场资源的利用率, 成为学术界和工业界的研究热点。

国外学者在停机位分配领域的研究主要聚焦于优化策略的探索, 提出多种创新性方法和模型。Deng 等^[1] 聚焦于最小化旅客行走距离、优化机位空闲时长、减少远机位使用频率与提升机位空间使用

效率, 采用改进型粒子群优化算法 (particle swarm optimization, PSO) 求解问题。Kim 等^[2] 提出考虑航班离港时机位冲突的模型, 旨在增强模型的鲁棒性。Schaijk 等^[3] 引入一种基于随机机位限制的方法, 构建线性规划模型。Deng 等^[4] 提出一个新的多目标优化模型, 以实现闲置时间合理分配、缩短旅客步行距离与减少远机位分配数量为目标, 并选用蚁群算法寻求最优解。Margarit 等^[5] 考虑机场随机性质, 结合贝叶斯建模与元启发式优点的方法求解停机位分配问题。Zhang 等^[6] 以优化飞行冲突、登机桥率及航空器滑行油耗为目标提出多目标优化

收稿日期: 2024-07-24; 修订日期: 2025-03-10

基金项目: 中央高校基本科研业务费专项 (24CAFUC03050)

第一作者: 向征 (1983—), 男, 汉族, 四川成都人, 博士, 副教授。研究方向: 空中交通管理自动化技术。E-mail: xz3_83@163.com。

投稿网址: www.stae.com.cn

模型,并用基于余弦变异和自适应分组的改进束群算法求解问题。Zhu 等^[7]通过构建预分配和动态重分配模型,以最大化登机口匹配度和靠近登机口的乘客分配率为优化目标,显著提高了乘客满意度和停机位资源利用率。Muhafiz 等^[8]运用强化学习算法,提出基于 Q 学习的序列决策方法,以最小化无停机位航班数量为目标,通过步进奖励函数兼顾鲁棒性和停机位利用率,为动态和随机性调度问题提供了新的解决方法。

国内研究则更多关注停机位分配的鲁棒性和乘客体验优化。朱春阳等^[9]以 NSGA-II 算法展开研究,以最小化旅客步行距离和总的鲁棒性成本为目标,用加权和方法将其合并为单一目标,建立停机位分配模型,采用邻域禁忌搜索算法获近似解,并用贪心随机策略防局部最优解。程博^[10]从评价函数和深度强化学习算法在机位分配的应用两方面进行研究。提出了多目标优化评价函数和一种基于深度强化学习的停机位分配算法。顾存昕^[11]从深度强化学习算法改进及停机位分配问题分析,利用 KNN(K-nearest neighbor)算法思想设计其与深度确定性策略梯度(deep deterministic policy gradient, DDPG)算法结合的机位分配算法,提升求解能力。刘继琳^[12]从机坪管制程序与航空器地面运行特性分析,提出两种适用于不同阶段的停机位分配优化模型,并设计差分进化算法求解。朱承元等^[13]提出基于多商品网络流的停机位分配模型,结合场面分区策略和离散粒子群算法,提高了停机位分配效率和机场运行效能,优化了资源利用。孙若飞等^[14]通过优化滑行路径和速度,提出一种绿色滑行优化模型,结合遗传算法和 Yen 算法,以降低滑行时间、尾气排放和滑行成本为目标,同时确保滑行与停机位分配的无冲突,为停机位衔接和场面运行优化提供了重要参考。刘海滨等^[15]基于鲁棒性模拟,构建停机位分配的整数线性规划模型,并引入爬山算法和大邻域搜索算法进行比较,提升分配方案的鲁棒性和求解效率,针对机场动态需求提供了高效优化方法。

尽管这些研究成果显著,但现有研究多存在模型的约束条件上处理较为简单,未能充分考虑实际运营中的复杂情况,缺乏对航班实际时间的动态调整,难以适应实际航班运行中的不确定性;对人工智能潜力挖掘不足等问题。

针对上述不足,综合考虑航班进出港实际时间、机型类别及停机位间相互关系等多方因素,构建以最小远机位分配数量和最小近机位空闲时间为目标的多目标数学模型。在算法方面,提出改进

型深度确定性策略梯度算法(deep deterministic policy gradient, DDPG),引入优先级经验回放与多策略融合探索机制,以提升优化性能与全局寻优能力。

1 多目标停机位分配数学模型

1.1 问题描述

停机位分配是指根据航空器和停机位的大小、航班地面服务时间以及航班进离场时刻等因素,将特定的停机位分配给航班^[16]。所构建的停机位分配优化模型的目标包括最小化远机位使用数量,提高近机位利用率,以及最小化近机位的空闲时间和,平衡航班进离场和近机位利用情况,从而提高起降效率,减少等待时间和浪费。

1.2 假设条件

- (1)先到达的航班优先分配,不使用组合机位。
- (2)停机位分配过程是连续的,没有明确的初始状态和终止状态,设置初始状态为所有机位都可用。
- (3)飞机一旦分配登机口,则一直占用登机口,不能移至其他登机口或临时停机坪,直到该机起飞^[17]。
- (4)为了提高近机位的使用率,假设航班停留时间超过 4 h 的航班,必须被分配到远机位。
- (5)假设航班的进场时间和离场时间是固定的,不考虑航班的进离场时间的不确定性。

1.3 停机位分配模型构建

1.3.1 模型参数定义

模型参数定义如表 1 所示。

决策变量为

$$X_{ik} = \begin{cases} 1, & \text{航班 } i \text{ 被分配到机位 } k \\ 0, & \text{其他} \end{cases} \quad (1)$$

式(1)中:决策变量 X_{ik} 为 0-1 变量,判断航班 i 的进港机位是否分配近机位 k 。

表 1 模型参数定义

Table 1 Model parameter definition

参数	定义
F	航班集合
i, j	航班编号
G	机位集合
k	机位编号
S_i	航班 i 进场时间
C_i	航班 i 离场时间
T_i	航班 i 的机型类别
T_k	机位 k 的大小类别
α	最小保障时间 45 min
β	相邻机位航班推出或驶入时间间隔 5 min
γ	同一机位最小安全间隔 15 min

$$B_{ijk} = \begin{cases} 1, & i \text{ 和 } j \text{ 航班连续分配到机位 } k \text{ 且 } i \text{ 在 } j \text{ 之前} \\ 0, & \text{其他} \end{cases} \quad (2)$$

式(2)中:决策变量 B_{ijk} 为 0-1 变量,判断航班 i 和航班 j 是否连续分配到机位 k ,且航班 i 在航班 j 之前。

$$Z_{kl} = \begin{cases} 1, & \text{机位 } k \text{ 与机位 } l \text{ 相邻} \\ 0, & \text{其他} \end{cases} \quad (3)$$

式(3)中:决策变量 Z_{kl} 为 0-1 变量,判断机位 k 与机位 l 是否是相邻机位。

$$Y_k = \begin{cases} 1, & \text{机位 } k \text{ 是远机位} \\ 0, & \text{其他} \end{cases} \quad (4)$$

式(4)中:决策变量 Y_k 为 0-1 变量,判断 k 机位是否是远机位。

1.3.2 目标函数

(1)最小远机位数量。

$$\min M_1 = \left(\sum_{i=1}^F \sum_{k=1}^G Y_k X_{ik} \right) \quad (5)$$

式(5)中: M_1 为远机位分配数量。

(2)近机位空闲时间和。

$$\min M_2 = \sum_{i=1}^f \sum_{k=1}^g (Y_k - 1) B_{ijk} X_{ik} X_{jk} (S_{jk} - C_{ik}) \quad (6)$$

式(6)中: M_2 为近机位空闲时间之和; X_{jk} 为 0-1 变量,判断航班 j 的进港机位是否分配近机位 k ,如果航班 j 分配到机位 k 则为 1,反之为 0; S_{jk} 为航班 j 的到达 k 机位时间; C_{ik} 为航班 i 的离开 k 机位时间。

式(5)表示所有航班都被分配到机位,以确保最少数量的航班被分配到远机位,从而提高近机位资源的利用率。式(6)表示为了减少近机位资源的空闲时间,使得每个近机位空闲时间和最小化。

1.3.3 约束条件

(1)航班唯一。一架飞机只能停靠一个停机位上。

$$\sum_{i=1}^f X_{ik} = 1, \quad \forall i \in F, \forall k \in G \quad (7)$$

(2)飞机大小与机位大小相匹配。

$$X_{ik}(T_k - T_i) \geq 0, \quad \forall i \in F, \forall k \in G \quad (8)$$

(3)航班最小保障时间。

$$X_{ik}(C_{ik} - S_{ik}) \geq \alpha, \quad \forall i \in F, \forall k \in G \quad (9)$$

(4)相邻机位航班推出或驶入时间最小间隔。

$$|Z_{kl} X_{ik} X_{jl} (S_{ik} - C_{jl})| \geq \beta \quad (10)$$

$$|Z_{kl} X_{ik} X_{jl} (S_{ik} - S_{jl})| \geq \beta \quad (11)$$

$$|Z_{kl} X_{ik} X_{jl} (C_{ik} - C_{jl})| \geq \beta \quad (12)$$

$$|Z_{kl} X_{ik} X_{jl} (C_{ik} - S_{jl})| \geq \beta \quad (13)$$

(5)同一机位最小安全间隔。

$$B_{ijk} |S_j - C_i| \geq \gamma, \quad \forall i, j \in F; \forall k \in G \quad (14)$$

(6)同一机位航班最多只有一个相邻后续航班。

$$\sum_{i=1}^F B_{ijk} \leq X_{jk}, \quad \forall i, j \in F; \forall k \in G \quad (15)$$

(7)同一机位航班最多只有一个相邻前续航班。

$$\sum_{j=1}^F B_{ijk} \leq X_{ik}, \quad \forall i, j \in F; \forall k \in G \quad (16)$$

(8)相邻航班 i 只能在 j 前或后。

$$B_{ijk} + B_{jik} \leq 1, \quad \forall i, j \in F; \forall k \in G \quad (17)$$

(9)如果航班停留时间超过 4 h,则分配远机位。

$$(C_{ik} - S_{ik}) \geq 4 \text{ h}, \quad \text{则 } Y_k X_{ik} = 1 \quad (18)$$

2 改进 DDPG 算法设计

深度确定性策略梯度(DDPG)算法自提出以来,已经被广泛应用于解决连续动作空间的强化学习问题。DDPG 结合了策略梯度方法的优点和深度学习的强大功能,显示出在多种任务中的优异性能。然而,在实际应用,如停机位分配问题中,由于环境的复杂性和不确定性,单纯的 DDPG 算法可能面临探索效率不足的挑战。有效的探索策略对于强化学习算法发现优质策略至关重要,特别是在环境反馈稀疏或者动作空间非常大时。因此,本章节旨在探讨如何通过融合奥恩斯坦-乌伦贝格噪声和退火策略来改进 DDPG 算法,以增强其在停机位分配问题中的表现。

2.1 马尔可夫决策过程模型设计

2.1.1 约束条件

(1)停机位信息。停机位信息分为停机位静态属性信息和机位占用情况信息两大部分,其中停机位静态属性信息包含停机位大小和停机位编号。

停机位大小 g_{si} 表示停机位 i 可以接受的最大停放的航空器类型。

停机位的编号(g_{ni})表示每一个停机位 i 的编号,便于判断一些相邻机位之间的特殊规则。

因此设定停机位静态属性矩阵 G : 设停机位数量为 M , 大小为 $M \times 2$ 表示每个停机位的静态属性。每一行包含停机位的大小和编号,即 $A = \{a_i\}_{i=1}^M$, 其中 $a_i = (g_{si}, n_{gi})$ 。

$$A = \begin{pmatrix} g_{s1} & g_{n1} \\ g_{s2} & g_{n2} \\ \vdots & \vdots \\ g_{sM} & g_{nM} \end{pmatrix} \quad (19)$$

设定停机位占用信息矩阵 O : 设停机位数量为 M , 时间步数为 $T+1$, 大小为 $(T+1) \times M$ 。这个矩阵

表示当前时刻以及后面时间步内每个停机位的占用状况。在这个矩阵中,每一行代表了一个特定的时间步,每一列对应一个停机位,元素值为0或1,表示停机位在相应时间步是否被占用和占用的时间。

$$O = \begin{pmatrix} o_{0,1} & o_{0,2} & \cdots & o_{0,M} \\ o_{1,1} & o_{1,2} & \cdots & o_{1,M} \\ \vdots & \vdots & & \vdots \\ o_{T,1} & o_{T,2} & \cdots & o_{T,M} \end{pmatrix} \quad (20)$$

(2) 航班信息。航班信息分为航班静态属性信息和航班资源信息。航班静态属性信息包含航班编号和航班机型大小信息。其中,航班编号(f_m)表示每个航班的唯一标识符。航空器类型(f_{ai})表示第*i*个航班所使用的航空器的型号,决定了航班需要的停机位大小。

定义航班属性矩阵 F : 设待分配航班数量为 N , 矩阵大小为 $N \times 2$ 。

$$F = \begin{pmatrix} f_{n1} & f_{a1} \\ f_{n2} & f_{a2} \\ \vdots & \vdots \\ f_{nN} & f_{aN} \end{pmatrix} \quad (21)$$

航班资源信息矩阵 C : 设航班进港时刻为 t_1 , 航班出港时刻为 t_2 , 则航班 f 预计停留在停机位的时间是 $t_1 \sim t_2$ 。其中, $T+1$ 为观察的时间步数, N 为待分配航班的数量, 则矩阵大小为 $(T+1)N$ 。每一列表示一个航班, 每一行表示一个时间步。每个元素 $c_{i,j}$ 可以明确地表示在时间步 i 时刻, 航班 j 的状态, 用二进制值表示, 1 表示停留, 0 表示未停留, 停留时间跨越多个时间步, 则相应的行向量中多个元素为 1。对于每个待分配的航班 i , 确定其停留时间范围, 并将对应的列向量 f_i 插入矩阵 C 的相应列中。图 1 为状态空间集合示意图。

$$C = \begin{pmatrix} c_{0,1} & c_{0,2} & \cdots & c_{0,N} \\ c_{1,1} & c_{1,2} & \cdots & c_{1,N} \\ \vdots & \vdots & & \vdots \\ c_{T+1,1} & c_{T+1,2} & \cdots & c_{T+1,N} \end{pmatrix} \quad (22)$$

图 1 中, 左侧矩阵代表停机位资源信息, 竖轴代表时间, 其中每一个单位代表一个时间步, 横轴代表停机位。其中空白格子表示当前机位空闲, 反之则表示机位正在被使用。右边矩阵代表待分配的航班资源信息, 其中竖轴代表时间每一个单位与停机位资源信息相同, 横轴代表待分配航班。因此停机位资源信息矩阵与航班资源信息矩阵共同组成了状态空间集合。

2.1.2 动作空间

在停机位分配问题中, 动作空间的设计至关重要

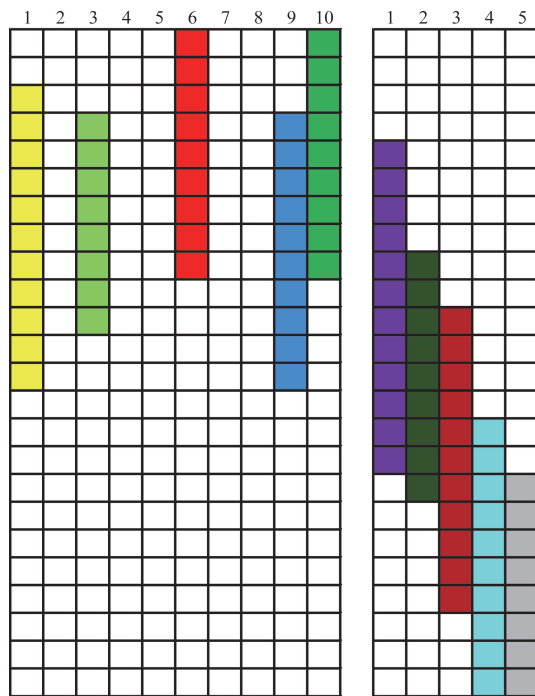


图 1 状态空间集合示意图
Fig. 1 Statespace set schematic diagram

要, 它直接影响了智能体在每个时间步中如何选择适当的停机位分配策略。通常情况下, 当有 M 个停机位和 N 个待分配的航班时, 不考虑任何约束情况, 可以定义一个动作空间 $A_{i,i}$, 表示航班 i 可能分配到的停机位的集合。这个集合可以用 $A_{i,i}(j_1, j_2, \dots, j_m)$ 表示, 其中 j_1, j_2, \dots, j_m 为停机位的编号, 因此空间集合为 M 。

2.1.3 奖励函数

在奖励函数中, 选择当航班被分配至近机位时, 反馈正奖励; 而当航班被分配至远机位时, 反馈负奖励。

$$R_1 = R_{near} \left(500 - \sum_{i=1}^f \sum_{k=1}^g Y_k X_{ik} \right) + R_{remote} \left(\sum_{i=1}^f \sum_{k=1}^g Y_k X_{ik} \right) \quad (23)$$

式(23)中: R_1 为近机位分配的奖励项; R_{near} 为近机位奖励; R_{remote} 为远机位奖励。

为了实现近机位的空闲时间最小化, 引入一个时间相关的奖励项。根据计算近机位的空闲时间的倒数, 当近机位的空闲时间越短, 即航班使用的效率越高时, 给予的奖励应该越大反之越小, 这一奖励机制可表示为

$$R_2 = \left[\sum_{i=1}^F \sum_{k=1}^G (Y_k - 1) B_{ijk} X_{ik} X_{jk} (S_{jk} - C_{ik}) \right]^{-1} \quad (24)$$

式(24)中: R_2 为时间最小化的奖励项。

为了确保最终的停机位分配方案符合实际约束条件,引入一个惩罚项。最终的奖励函数可以表示为

$$R_{\text{total}} = R_1 + R_2 + R_3 \quad (25)$$

式(25)中: R_3 为违反约束条件的惩罚项。

2.2 优先回放机制

优先经验重放 (prioritized experience replay, PER) 通过为样本中的每种经验分配优先级来控制抽样过程。这些优先级反映了每次试验对学习过程的重要性,根据时差 (time difference, TD) 误差来计算。TD 误差的绝对值越高,说明当前策略对某次试验的预测偏差越大,该次试验的学习潜力也越大,因此应赋予其更高的优先级。具体而言,每个经验样本的优先级可表示为

$$P(i) = |\delta_i| + \varepsilon \quad (26)$$

式(26)中: δ_i 为第*i*个经验样本的 TD 误差; ε 为一个很小的正数,用于确保每个样本都有非零的概率被选中。

在进行经验抽样时,每个样本被抽中的概率与其优先级成正比,可表示为

$$P_i = \frac{P(i)^\alpha}{\sum_k P(k)^\alpha} \quad (27)$$

式(27)中: α 为一个控制优先级影响程度的参数。

通过这种方式,具有更高优先级的经验更有可能被选中用于训练,从而提高学习的效率。

为了补偿这种非均匀抽样带来的偏差,PER 还引入重要性采样权重 (importance sampling weights, ISWs),用于调整学习更新的大小。第*i*个样本的 ISW 定义为

$$W_i = \left[\frac{1}{n} \frac{1}{P(i)} \right]^\beta \quad (28)$$

式(28)中: n 为经验缓冲池中的样本总数; β 为衡量 ISWs 影响程度的参数。

在训练过程中,这些权重会随着时间逐渐增加至 1,以减少偏差的影响。

通过引入 PER 机制,改进的 DDPG 算法能够更加高效地从经验中学习,加速收敛过程,并提高最终策略的性能。在停机位分配问题的背景下,这意味着算法能够更快地学习到高效的停机位分配策略,从而提高机场运营的效率 and 旅客的满意度。

2.3 多策略融合探索机制

2.3.1 OU 噪声

OU (Ornstein-Uhlenbeck) 噪声是一种用来模拟具有一定惯性的物理过程中的噪声,它的特点是时间相关的,可以产生更平滑的噪声序列。OU 噪声的一个关键特性是它会趋向于一个长期平均值,具

有一定的“回归”性质,这使得产生的噪声不会随时间无限制增长,而是围绕某个值波动。通常被添加到策略网络输出的动作上,以便在训练期间促进对环境的探索。计算公式为

$$dN_{\text{ouset}} = \theta_{\text{ou}} (\mu - N_{\text{ouset}}) dt + \sigma_{\text{ou}} dw_t \quad (29)$$

式(29)中: N_{ouset} 为当前噪声值; μ 为长期平均水平; θ_{ou} 为速率参数,表示噪声回归到平均值的速度; σ_{ou} 为噪声强度; dw_t 为维纳过程的增量; t 为当前时间步。

在训练阶段,每当 Actor 网络产生一个动作时,将 OU 噪声添加到该动作上,可表示为

$$a_t = \pi(s_t | \theta) + N_{\text{ouset}} \quad (30)$$

式(30)中: a_t 为最终执行的动作; θ 为 Actor 网络的参数; $\pi(s_t | \theta)$ 为 Actor 网络对于状态 s_t 的策略输出; N_{ouset} 为从 OU 过程生成的噪声。

在采取动作并观察到新的状态和奖励后,使用 Critic 网络来估计这个动作的价值,并通过梯度上升法更新 Actor 网络,使其产生更高价值的动作。

2.3.2 探索退火策略

探索退火 (exploration annealing) 是一种在强化学习中用于平衡探索 (exploration) 和利用 (exploitation) 的技术。其核心思想是随着学习的进行,逐渐减少对探索的依赖,从而让智能体更多地利用已学习到的知识。这种方法借鉴了模拟退火算法中的退火过程,通过逐步降低“温度”参数来减少随机性和增加确定性的决策。在学习的早期阶段,智能体对环境了解不足,需要进行大量的探索来发现有效的策略;随着学习的进行,智能体逐渐积累了更多关于环境的知识,此时应更多地利用这些知识来做出最优决策,减少探索带来的随机性。

在探索退火策略中设定探索率的初始值 ε_0 , 这个值决定了智能体在初始阶段探索动作空间的概率或动作噪声的初始强度。在每个时间步或训练回合后,根据选择的退火策略更新探索率 ε 。退火策略为线性退火,可表示为

$$\varepsilon = \max(\varepsilon_{\text{min}}, \varepsilon_0 - \delta t) \quad (31)$$

式(31)中: ε_{min} 为探索率的最小值; δt 为每个时间步减少的量。

在训练过程中根据退火策略更新探索率 ε , 同时根据探索率 ε 调整 OU 噪声的强度 σ_{ou} , 在环境中执行加噪声后的动作,观察新状态和奖励,再根据观察结果更新 Actor 和 Critic 网络,同时根据退火计划继续调整探索率和噪声强度。

2.4 改进 DDPG 算法流程图

在改进 DDPG 算法设计中,以改进经验回放机制和探索策略以增强 DDPG 算法的性能,因此在改

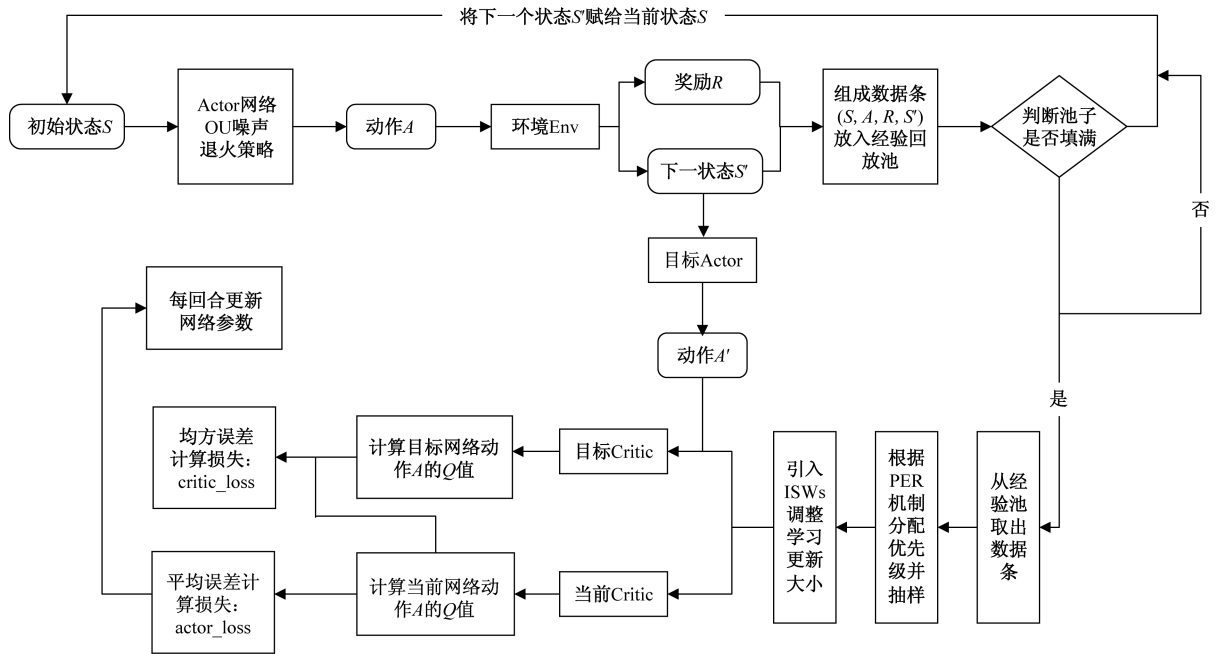


图2 改进 DDPG 算法流程图
Fig. 2 Improved DDPG algorithm flowchart

进 DDPG 算法的设计中当前 Actor-Critic 网络、目标 Actor-Critic 网络的设计、马尔可夫模型与 DDPG 算法相同。算法流程如图 2 所示。

3 仿真实验与结果对比

选用 DDPG 算法、改进 DDPG 算法对同一个问题求解所产生的结果进行分析,为了可以更直观地对其进行评价,直接选用从近机位的分配数量和近机位空闲时间进行结果上的对比。

3.1 最小远机位分配数量

在选用最小远机位分配数量为目标函数是为了使得让航班可以更多地分配到近机位,因此针对最终的近机位分配方案进行分析,DDPG 分配到近机位的航班架次为 222 架次,其中各机位分配架次如图 3 所示。

改进 DDPG 分配到近机位的航班架次为 248 架次,其中各机位分配架次如图 4 所示。

改进后的 DDPG 算法在经验探索策略和噪声策略上进行改进,因此最终的分配方案更优于原始方案。对比改进 DDPG 算法与 DDPG 在远机位分配数量的求解性能,对比曲线如图 5 所示。

图 5 中,红色曲线代表 DDPG 算法,蓝色曲线代表改进 DDPG 算法。初始阶段,两个算法的远机位分配数量相近。随着迭代次数的增加,改进 DDPG 算法表现出了更好的性能,分配的远机位数量较 DDPG 算法减少。改进 DDPG 算法在大约 280 次开始趋于稳定,在一个较低的分配数量附近波动,其

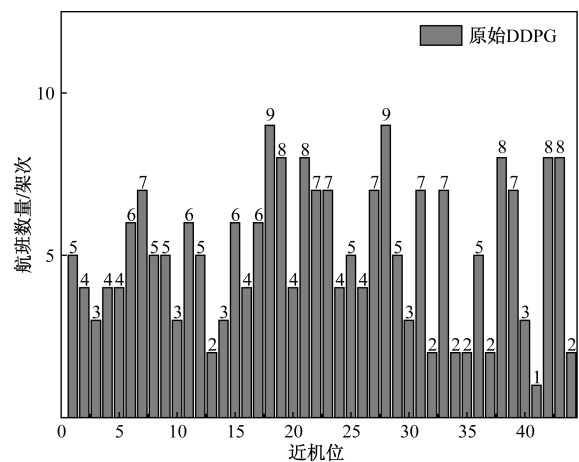


图3 近机位分配的航班数量(原始 DDPG)
Fig. 3 The number of flights for proximal gate assignment (original DDPG)

最后一次远机位分配结果为 262, DDPG 算法在大约 350 次开始趋于稳定,在一个较低的分配数量附近波动,其最后一次远机位分配结果为 278。因此从收敛速度和最终结果进行对比,表明改进 DDPG 算法比 DDPG 算法可以更快地找到了更有效的策略来减少远机位的使用。

3.2 近机位空闲时间

根据最小近机位空闲时间为优化目标,对比原始 DDPG 算法与改进 DDPG 算法在最终分配结果中近机位空闲时间的表现,如图 6 所示。

比较图 6 可知,比较 DDPG 算法与改进后的 DDPG 算法,其中红色曲线代表 DDPG 算法,蓝色曲

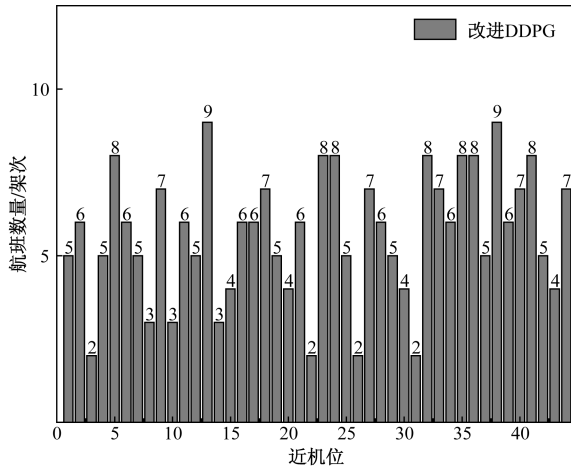


图4 近机位分配的航班数量(改进 DDPG)

Fig. 4 The number of flights for proximal gate assignment (improved DDPG)

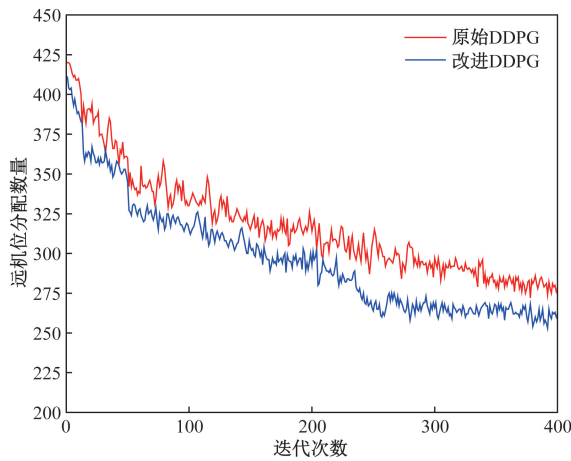


图5 DDPG 与改进 DDPG 远机位分配数量对比图

Fig. 5 Comparison chart of the number of remote gate assignments between DDPG and improved DDPG

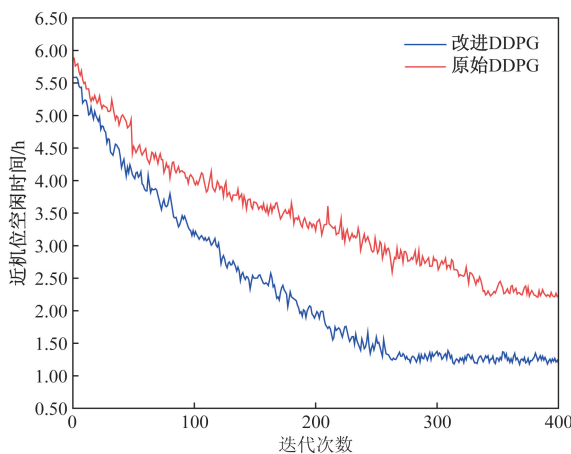


图6 近机位空闲时间算法迭代对比

Fig. 6 Iterative comparison of proximal gate idle time algorithms

线代表改进后的 DDPG 算法。通过观察两种算法随着迭代次数增加而分配时间的变化趋势,可见学习

过程持续改善。改进后的 DDPG 算法展现出更为显著的下陷趋势,暗示其可能具有更快的学习速率。从收敛速度的角度来看,改进后的 DDPG 算法在大约 280 次迭代后开始收敛,而传统的 DDPG 算法则需要大约 350 次迭代才开始收敛。在最终值方面,改进后的 DDPG 算法将近机位空闲时间降低至 1.4,而原始 DDPG 算法的最终值为 2.2。因此改进 DDPG 算法不仅加快了算法的收敛速度,而且还显著提高了优化结果的质量。

3.3 累计奖励

DDPG 算法与改进 DDPG 算法累计奖励迭代对比如图 7 所示。

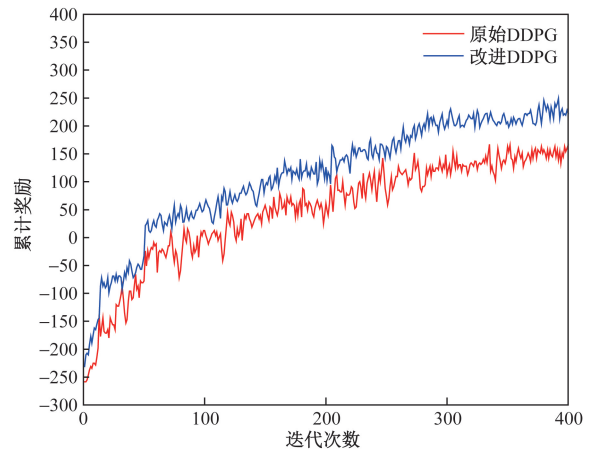


图7 累计奖励迭代对比

Fig. 7 Cumulative reward iteration comparison

如图 7 所示,红色曲线代表 DDPG 算法,蓝色曲线代表改进 DDPG 算法。DDPG 算法在迭代 400 次时累计奖励约为 165,改进 DDPG 在迭代 400 次时累计奖励约为 233,这表明改进 DDPG 算法在停机位分配任务中表现更优,能够更有效地将航班分配至近机位,减少远机位的使用,从而获得更高的正奖励。同时,它还能更好地优化近机位的空闲时间,使得近机位的使用效率更高,进一步增加累计奖励。该结果进一步证实了改进 DDPG 算法在性能上的优越性。

4 结论

综合运用多目标优化和深度强化学习,针对机场停机位分配问题提出一套新的解决方案。首先介绍改进 DDPG 算法的设计思路,包括算法的基本框架和关键改进措施。优先级经验回放机制通过对经验样本的重要性排序,加快了算法对高价值经验的学习速度;多策略融合探索机制则通过结合多种探索策略,增强了算法的探索能力,避免了过早收敛到局部最优解。通过对比实验,DDPG 算法和

改进 DDPG 算法,从最小化远机位分配数量和优化机位空闲时间两个目标出发,展示了改进 DDPG 算法在停机位分配问题上的优越性能。实验结果表明,改进的 DDPG 算法不仅在求解质量上有明显的提升,同时也在算法收敛速度和稳定性上表现出色。

参 考 文 献

- [1] Deng W, Zhao H, Yang X, et al. Study on an improved adaptive PSO algorithm for solving multi-objective gate assignment [J]. *Applied Soft Computing*, 2017, 59: 288-302.
- [2] Kim S, Feron E. Robust gate assignment against gate conflicts[J]. *Journal of Air Transportation*, 2017, 25: 1-8.
- [3] Schaijk O, Visser H. Robust flight-to-gate assignment using flight presence probabilities [J]. *Transportation Planning and Technology*, 2017, 40: 1-18.
- [4] Deng W, Sun M, Zhao H, et al. Study on an airport gate assignment method based on improved ACO algorithm[J]. *Kybernetes*, 2018, 47(1): 20-43.
- [5] Margarita B, Miguel M M. A multi-objective optimization with a delay-aware component for airport stand assignment[J]. *Journal of Air Transport Management*, 2020, 83: DOI: 10.1016/j.trd.2010.07.007.
- [6] Zhang Y, He Q, Yang L, et al. An improved tunicate swarm algorithm for solving the multi-objective optimisation problem of airport gate assignments [J]. *Applied Sciences*, 2022, 12(16): DOI: 10.1007/s00500-022-07794-2.
- [7] Zhu C, Wei Z, Lü Z, et al. Gate assignment algorithm for airport peak time based on reinforcement learning[J]. *Transportation Research Record*, 2024, 7: DOI: 10.1109/TSST.2015.2467190.
- [8] Muhafiz Y M, Avcı U, Örnek M A, et al. Flight gate assignment problem with reinforcement learning[C]//*International Conference on Intelligent and Fuzzy Systems*. Cham: Springer, 2023: 189-196.
- [9] 朱春阳, 郭晓彤, 孙浩然, 等. 基于极值点搜索和非支配排序的高维多目标优化算法[J]. *小型微型计算机系统*, 2018, 39(1): 32-36.
Zhu Chunyang, Guo Xiaotong, Sun Haoran, et al. Many-objective optimization algorithm based on nadir point search and nondominated sorting [J]. *Journal of Chinese Computer Systems*, 2018, 39(1): 32-36.
- [10] 程博. 基于深度强化学习的停机位分配算法研究[D]. 深圳: 中国科学院大学(中国科学院深圳先进技术研究院), 2020.
Cheng Bo. Gate assignment problem research via deep reinforcement learning [D]. Shenzhen: Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, 2020.
- [11] 顾存昕. 基于马尔科夫的机场机位分配建模与强化学习算法[D]. 武汉: 华中科技大学, 2021.
Gu Cunxin. Research on modeling of airport gate assignment based on Markov process and deep reinforcement learning [D]. Wuhan: Huazhong University of Science and Technology, 2021.
- [12] 刘继琳. 基于差分进化算法的机场停机位分配研究[D]. 广州: 中国民用航空飞行学院, 2022.
Liu Jilin. Research on the airport gate assignment based on difference evolution algorithm [D]. Guanghan: Civil Aviation Flight University of China, 2022.
- [13] 朱承元, 于海波. 停机位分配的多商品网络流模型及离散粒子群算法[J]. *科学技术与工程*, 2023, 23(10): 4417-4425.
Zhu Chengyuan, Yu Haibo. Multi-commodity network flow model and discrete particle swarm [J]. *Science Technology and Engineering*, 2023, 23(10): 4417-4425.
- [14] 孙若飞, 李杰, 万莉莉, 等. 基于速度分配的航空器绿色滑行优化[J]. *科学技术与工程*, 2022, 22(16): 6752-6758.
Sun Ruofei, Li Jie, Wan Lili, et al. Optimization of aircraft green taxi based on speed allocation [J]. *Science Technology and Engineering*, 2022, 22(16): 6752-6758.
- [15] 刘海滨, 王炬博, 巴博圣, 等. 基于鲁棒性模拟的停机位分配问题的数值方法比较[J]. *山东科学*, 2024, 37(2): 104-116.
Liu Haibin, Wang Jubo, Ba Bosheng, et al. A numerical comparison of methods for solving the gate allocation problem based on robustness simulation [J]. *Shandong Science*, 2024, 37(2): 104-116.
- [16] 冯程, 胡明华, 赵征. 一种新的停机位分配优化模型[J]. *交通运输系统工程与信息*, 2012, 12(1): 131-138.
Feng Cheng, Hu Minghua, Zhao Zheng. A new optimization model of airport gate assignment [J]. *Journal of Transportation Systems Engineering and Information Technology*, 2012, 12(1): 131-138.
- [17] 董兵, 吴郑源, 赖桂瑾, 等. 具有卫星厅的机场航站楼登机口分配研究[J]. *科学技术与工程*, 2020, 20(18): 7545-7551.
Dong Bing, Wu Zhengyuan, Lai Guijin, et al. Research on gate allocation of airport terminal with satellite hall [J]. *Science Technology and Engineering*, 2020, 20(18): 7545-7551.