基于多目标深度强化学习的需求响应式列车时刻表优化

高如虎†，刘 伟，焦治铎

(兰州交通大学 交通运输学院，兰州 730030)

摘 要: 客流在时空维度上呈现的时变特性对城市轨道交通运营管理提出了严峻挑战, 同时, 需求响应式列车时刻表优化问题的复杂性对于算法设计提出了更高的要求. 为此, 将人工智能领域的深度强化学习方法应用于地铁列车时刻表的优化问题, 以提升地铁运营管理的智能化水平. 将时变客流需求与列车时刻表决策互动关系构建为马尔可夫决策过程, 为智能体提供训练和学习环境. 其中, 以乘客服务、列车运量以及列车运行作为状态空间, 以列车发车间隔作为动作空间, 并设计“人-车-站”一体化的多维复合奖励函数. 开发一种基于自适应发车间隔和列车数量的多目标软演员-评论家算法提升求解效率. 以小规模算例进行超参数优化, 并验证需求响应式列车时刻表相对于均衡列车时刻表的优势. 以广州市地铁 8 号线进行仿真实验, 结果表明, 所提出方法相对于其他人工智能方法及启发式算法具有较快的收敛速度和求解效率. 此外, 针对不同客流扰动场景, 所提出方法能够在短时间内生成满意的运营方案, 表明所提出方法具有较好的泛化能力. 研究结果可为进一步提升地铁运营调度智能化水平提供理论和方法支撑.

关键词: 时变需求；列车时刻表；深度强化学习；软演员-评论家算法；多维复合奖励

中图分类号: U292.4 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2025.1334

引用格式: 高如虎, 刘伟, 焦治铎. 基于多目标深度强化学习的需求响应式列车时刻表优化 [J]. 控制与决策.

Optimization of demand-responsive train timetabling based on multi-objective deep reinforcement learning

GAO Ru-hu†，*LIU Wei*，*JIAO Zhi-duo*

(School of Traffic and Transportation，Lanzhou Jiaotong University，Lanzhou 730070，China)

Abstract: The time-varying characteristics of passenger flow in spatial and temporal dimensions impose significant challenges on urban rail transit operation management, while the complexity of demand-responsive train timetabling calls for more advanced algorithmic solutions. To address this, this paper introduces a deep reinforcement learning (DRL) approach to optimize subway train schedules, thereby improving the intelligence of operational management. The dynamic interaction between time-varying passenger demand and train timetable is formulated as a Markov decision process, which serves as a training environment for the learning agent. The state space includes passengers service, train capacity, and train operation, the action space is defined by dispatch intervals. A multi-dimensional reward function jointly considers passengers, trains, and stations. A multi-objective soft actor-critic algorithm with adaptive dispatch intervals and train numbers is developed for efficient optimization. Hyper-parameter tuning via a small-scale case and confirms that demand-responsive timetable outperforms fixed-interval timetable. Simulations on Guangzhou Metro Line 8 show faster convergence and higher solution efficiency compared to other AI and heuristic methods. Moreover, under various passenger flow disturbance scenarios, the method can generate satisfactory operation plans in a short period of time, demonstrating its strong generalization capability. These results provide theoretical and methodological support for intelligent subway scheduling.

Keywords: time-dependent demand； train timetabling； deep reinforcement learning； soft actor-critic algorithm； multi-dimensional reward

收稿日期: 2025-12-25；**录用日期:** 2026-04-05.
基金项目: 国家自然科学基金项目 (72361020).
责任编委: 龙建成.
†通信作者. E-mail: grh@mail.lzjtu.cn.

0 引 言

城市轨道交通作为城市公共交通的骨干, 承载着每日数以百万计的客流量. 其客流在时空维度上呈现出典型的时变与不均衡特征. 这对地铁运营管理提出了严峻挑战, 高峰时段运力不足导致拥挤, 而平峰时段则运力过剩造成资源闲置与运营成本攀升. 列车运行时刻表作为协调客流需求与运力供给的核心, 其科学性与适应性直接决定了地铁运输效率与服务品质. 因此, 面向时变客流需求特性, 开展精细化、智能化的列车时刻表优化研究, 对于实现城市轨道交通系统按需供给、提升运营效益与乘客服务质量具有至关重要的意义[1].

针对时变客流下的列车时刻表优化问题, 学术界已开展了广泛研究, 其核心目标通常是在满足运输能力与安全约束的前提下, 最小化乘客总等待时间[2]、列车总能耗[3] 等. NIU 等[4][5] 面向时变客流需求, 分别构建基于分钟和小时需求的列车时刻表优化模型. 李佳杰等[6] 构建了站外限流与列车时刻表协同优化模型, 实现站外站内乘客加权人均等待时间最小. TIAN 等[7] 提出了一种面向需求与考虑越行的高速铁路列车运行图优化方法, 将基于小时的客流需求精准分配至适配的列车上. BUCAK 等[8] 关注于客流超拥挤地铁线路上的列车时刻表优化问题以适应动态的客流需求. 陈治亚等[9] 以多时段发车频率为决策变量, 研究了客流不确定性下乘客的出行时段选择行为. 为了解决高峰期地铁线路上乘客拥挤的不均衡问题,SHI 等[10] 提出了一种安全导向的列车时刻表和停站方案综合优化方法. 针对旅客需求不确定性, 张春田等[11] 基于分布鲁棒优化方法对列车停站方案与时刻表进行协同优化. 钟林环等[12] 采用灵活编组模式综合优化列车开行频率和时刻表, 以达到运输能力供给与客流需求的精准匹配.

传统的列车时刻表优化方法主要通过构建确定的数学模型, 并设计精确算法或启发式算法进行求解. 一方面, 将列车到发时刻作为决策变量, 构建列车运行图优化问题的混合整数规划模型并利用求解器 (Cplex、Gurobi 等) 精确求解[13][14]. 另一方面, 利用分解算法 (拉格朗日松弛、列生成、交替方向乘子分解算法等) 将大规模的列车运行图优化问题分解为一系列容易求解的子问题也是一种常用的求解方法[15][16]. 虽然所得方案的可行性得以保证, 但计算代价更高, 难以满足快速决策的要求. 此外, 根据问题特征设计一些启发式规则和策略, 并利用经典的启发式算法 (遗传算法、大规模邻域搜索算法等) 可对列车运行图问题求解[17-19]. 此类方法能够为特定场景提供高质量的解, 但其建模过程往往需要对复杂的现实环境进行大量简化, 且求解效率随着问题规模的扩大而急剧下降. 更重要的是, 它们通常只针对具体的客流优化, 缺乏泛化能力, 限制了其在现实环境中的应用潜力.

随着人工智能技术的发展, 深度强化学习方法为解决复杂的优化决策问题提供了新的范式[20] SEMROV 等[21] 的工作开创性地将强化学习引入列车调度领域, 提出了一种基于 Q-learning 的强化学习方法优化列车的实时调度问题. LI 等[22] 将列车运行图优化问题刻画为马尔科夫决策过程, 并设计了基于多智能体演员-评论家算法的深度强化学习框架. 俞胜平等[23] 针对突发事件造成高铁列车延误晚点的动态调度问题, 提出一种基于策略梯度强化学习的高铁列车动态调度方法. 代学武等[24] 将高铁调度问题构建为多阶段序列决策过程, 并提出了一种采用源域经验学习和目标域在线决策的可迁移高铁调度算法. YANG 等[25] 提出了一种高速铁路列车运行图及实时调度问题的统一优化模型, 并将深度强化学习的决策策略与局部搜索策略相结合提升求解质量. 随着对深度强化学习在轨道交通运输组织领域的研究不断深入, 诸多成果突破了纯粹的列车冲突消解, 并延伸至更复杂的面向需求的列车时刻表优化问题. Ying 等人[26] 提出了一种基于 Actor-Critic 架构的深度强化学习方法, 在随机客流需求下实现了地铁列车时刻表与车辆交路的联合优化. Wang 等[27] 将列车调度问题表述为一个部分可观察的马尔可夫决策过程, 旨在最大限度地减少需求驱动的乘客等待时间和运营成本. 现有研究大部分固定了运力资源 (如列车数量) 的供给, 导致了时刻表优化的动态性不足; 另外, 部分研究仅关注一个运营时段 (如高峰期) 时刻表的优化, 缺乏全天方案的生成; 同时, 在设计奖励函数时往往仅考虑乘客或运营方的目标, 忽略了大客流场景下车站安全管理的要求.

本文旨在构建一个能够将时变客流需求与列车时刻表决策深度融合的优化框架, 并开发一种多目标深度强化学习算法高效求解. 与现有文献相比, 本文的创新点主要体现在以下三个方面. 首先, 提出了一种面向时变需求的列车时刻表优化深度强化学习新范式, 将列车时刻表优化模型转化为马尔可夫决策过程, 为列车智能体提供训练和经验学习的环境. 其次, 设计了“人-车-站”一体化的多维复合奖励函数, 整合了乘客体验、列车运营效率以及车站服务水平三个维度的核心指标, 并产生帕累托解集来满足

决策的不同偏好. 此外, 开发了一种基于自适应发车间隔和列车数量的软演员评论家算法, 此方法具有较好地泛化能力.

1　问题描述及符号定义

1.1　问题描述

本文聚焦于需求响应式列车时刻表的优化, “需求响应”指时刻表优化时能够灵活响应客流的时变特征, 打破传统固定间隔发车的模式, 实现“按需开车”. 在一条城市轨道交通线路上, 站点依次编号为1,2, …, $|N|$, 其中第 1 站和第$|N|$站分别表示起始站和终点站. 研究时段为 [0, T], 不失一般性, 将研究时段按照一定的时间粒度离散化处理. 研究时段内的客流需求可以通过历史客流数据按照时段的离散精度预测而来, 将时变客流需求定义为$\lambda_{u,v}(t)$, 表示任意时间t从u站去往v站的乘客人数.

为便于问题建模, 作以下假设:

假设 1: 假设所有列车区间运行时间固定, 并且列车停站方案及停站时间由开行方案预先确定.

假设 2: 为方便刻画客流需求在列车上的加载过程, 假设车站等待乘客乘车都遵循“先到先上”原则, 这是研究此问题的一个普遍假设[4].

假设 3: 本文研究对象为一条地铁线路走廊, 我们仅考虑直达客流需求, 对于换乘客流在本研究中暂不考虑, 在后续网络运营模式下的地铁列车时刻表优化研究中将进一步考虑.

在时刻表优化中将列车开行数量作为预先确定的参数是常见的做法, 但这种处理方法限制了列车时刻表根据客流需求动态匹配的灵活性, 可能会造成运力资源的浪费. 为此, 本文根据客流需求动态地确定最优的列车开行数量.

1.2　符号定义

模型中所涉及的集合与索引定义为: T, t为时间集合和索引; K, k为可用列车集合与索引; U, u, v为车站集合与索引.

所需参数定义为: $\lambda_{u,v}(t)$表示t时刻到达车站u并前往车站v的乘客人数; $h_{\max}$和$h_{\min}$分别表示列车的最大间隔和最小间隔; $ts_{k,u}$为列车k在u站的停站时间; $tr_{k,u}$为列车k在区间 $[u, u+1]$ 的运行时间; C_k为列车k的载客能力; θ_u为车站u的站台安全阈值; M表示一个极大的常数.

变量定义为:

x_k表示列车k是否在研究时段内提供服务, 若是为 1, 否则为 0;

h_k为相邻列车$k-1$与k的发车间隔;

$q_{k,u,v}(t)$表示在时刻t到达车站u站去往v站的乘客能否被列车k服务, 若是为 1, 否则为 0;

m表示最终确定发出的列车数量;

$td_{k,u}$和$ta_{k,u}$分别为列车k在车站u的出发与到达时刻;

$b_{k,u}$, $d_{k,u}$分别表示列车k在u站的上车人数和下车人数;

$\theta_{k,u}$, $\psi_{k,u}$分别表示列车k到达u站时的等待人数及从车站出发后的滞留人数;

$c_{k,u}$为列车k从u站出发时的载客人数.

2　马尔可夫决策过程

马尔可夫决策过程 (Markov Decision Process, MDP) 是深度强化学习的核心理论框架, 智能体通过状态-动作-奖励机制与环境交互, 实现策略的迭代优化. 因此, 智能体要素和环境的合理设计是强化学习建模的关键.

本文关注的列车时刻表优化问题, 其本质是基于客流需求决策列车在时空维度上的有序分布, 可抽象为阶段性序贯决策过程. 本文将时刻表优化建模为以运营列车为智能体的事件驱动型 MDP, 其中, 决策阶段k对应线路上开行的第k次列车, 列车智能体在当前状态下作出决策并影响后续运行.

2.1　状态空间设计

状态空间是智能体在环境中可观测到的所有可能状态的集合. 状态是智能体决策的依据, 需包含足够信息以推断最优动作. 定义阶段k状态变量为s_k, 可用以下元组来表示:

$$s_k = (\boldsymbol{\theta}_k, \mathbf{C}_k, \mathbf{P}_k). \tag{1}$$

其中, $\boldsymbol{\theta}_k$为客流服务相关的状态向量, 记录了在车站u前往v站的等候人数, 具体包含时变客流$\lambda_{u,v}(t)$的累积及滞留人数$\psi_{k-1,u}$; $\mathbf{C}_k$为列车运量$c_{k,u}$相关的状态向量; $\mathbf{P}_k$为与列车运行相关的状态向量, 包括列车k在各站的到发时间$ta_{k,u}$、$td_{k,u}$以及开行列车数m. 值得说明的是, 由于本研究考虑了灵活的列车数量, 在状态中引入列车开行数量的指示变量m. 当规划达到运营时段结束时间时, 终止决策流程, 此时规划的列车数量即为最终服务的列车数量, 这种处理避免了由于列车数量不确定而产生的建模与求解难题.

2.2　动作空间设计

动作空间是智能体在环境中可执行的所有可能动作的集合. 不同于直接决策每一列车具体的发车时刻所带来的庞大的解空间和安全间隔问题, 本文通过决策一系列相邻列车发车间隔的方法, 将搜索空间有效缩小, 不仅简化了智能体的学习任务, 而且

保证了动作的输出自然地遵守了列车运行间隔可行性约束.

具体地, 将第 k 阶段的动作定义为 a_k, $a_k \in [-1, 1]$. 由于智能体输出的动作是一个归一化到[−1,1] 区间的向量, 所以由解码函数 (2) 将动作 a_k 线性映射为实际列车发车间隔 h_k. 这种动作映射方法梯度平滑, 保留了连续空间中的梯度信息, 避免了离散动作空间带来的维度灾难和稀疏奖励问题.

$$h_k = round(h_{\min} + \frac{a_k + 1}{2} \times (h_{\max} - h_{\min})). \quad (2)$$

2.3 奖励函数设计

奖励函数是智能体在当前状态下执行动作后获得的反馈. 在与环境的交互过程中, 智能体需要明确在此种状态下执行此种动作的后果, 并对该动作产生的价值进行量化评估并反馈.

城市轨道交通运营本质上是一个多方利益博弈的复杂系统. 现有研究往往侧重于单一指标, 容易导致运营方案难以满足多方利益. 基于此, 本文以乘客等待时间最少、列车运营成本最小、车站拥挤度最低为优化目标, 提出了一种基于“人-车-站”三位一体的复合奖励函数, 并通过求解帕累托解集来使决策者根据不同偏好选择方案.

(1) 乘客等待时间

对于乘客而言, 在超拥挤条件下, 部分乘客因为列车能力的限制可能不能登乘临近到达的第一列列车, 从而导致较长的等待时间. 若等待时间过长, 乘客可能会放弃城市轨道交通而选择其他方式出行. 采用乘客总等待时间来衡量服务的质量, 这也是需求响应式列车时刻表问题中的一个常用优化目标.

$$\min r_p = \sum_{k\in K}\sum_{u\in U\setminus N}\sum_{v>u}\sum_{t\in T} q_{k,u,v}(t)\times \lambda_{u,v}(t) \times (td_{k,u} - t). \quad (3)$$

(2) 列车运行成本

由于考虑了灵活的列车数量, 对于铁路运营方而言, 列车数量与运营成本密切相关, 希望以最少的列车提供为乘客提供最优质的服务, 将列车开行的固定成本作为优化的目标.

$$min\ r_o = w \times m. \quad (4)$$

其中, w为每开行一列列车对应的固定成本.

(3) 车站拥挤惩罚

高峰期大量的乘客进入车站, 如果不能够及时服务, 导致客流聚集在站台上, 为站台的安全管理造成了很大的挑战. 在列车k从车站u出发之前, 当站台上的候车人数超过一个安全阈值θ_u时产生惩罚, 且超出的人数越多, 惩罚越大. 关于车站拥挤度优化目标可表示为,

$$minr_s = \sum_{k\in K}\sum_{u\in U} max\{0, \theta_{k,u} - \theta_u\}. \quad (5)$$

基于以上 3 个方面优化目标的建立, 最终总奖励函数被定义为:

$$r = -[\sigma \times r_p + (1-\sigma) \times r_o] - r_s. \quad (6)$$

其中, 将乘客等待时间和列车运营成本作为驱动智能体的主要奖励, 通过调整权重系数σ, 可以灵活地在“服务导向”或“成本导向”的运营策略之间进行决策. 而站台拥挤度作为惩罚性奖励, 在强化学习框架下, 任何诱发站台严重拥挤的调度策略均会因惩罚骤增而导致总奖励大幅下降, 进而在强化学习算法的策略评估与更新过程中被自动淘汰.

2.4 环境状态约束

本文建立的地铁列车运行仿真环境由乘客时变到达与列车离散运行事件共同构成. 综合考虑列车运行与时变客流特点, 环境状态约束条件可分为列车状态约束与客流状态约束.

(1) 列车状态约束

首先, 由于考虑灵活的列车数量为旅客提供服务, 在给定可用列车集合后, 在后续算法中依次规划列车是否上线提供服务, 因此, 存在以下关系,

$$x_{k-1} \geqslant x_k,\ \forall k \in K, \quad (7)$$

规划完成后, 所需的实际列车数量m可表示为:

$$m = \sum_{k\in K} x_k. \quad (8)$$

另外, 列车是否提供服务与列车的到发时刻存在如下关系:

$$\begin{cases} td_{k,1} \leqslant T + M \times (1 - x_k),\ \forall k \in K \\ td_{k,1} > T - M \times x_k,\ \forall k \in K \end{cases} \quad (9)$$

当规划的列车在始发站的出发时刻超出运营时段时, 表明决策终止, 此车次及后续车次终止开行.

列车在始发站的出发时刻与前一列车出发时刻及相邻列车的发车间隔的关系描述为:

$$td_{k,1} = td_{k-1,1} + h_k,\ \forall k \in K. \quad (10)$$

为保证列车运营安全和乘客的服务质量, 相邻列车之间必须满足最小安全间隔与最大服务间隔,

$$h_{min} \leqslant h_k \leqslant h_{max},\ \forall k \in K. \quad (11)$$

在确定了列车 k 在始发站的出发时间后, 即可根据公式 (12) 确定性的递推得到在后续各站的到达及出发时刻.

$$\begin{cases} ta_{k,u} = td_{k,u-1} + tr_{u-1},\ \forall k \in K, u \in U\setminus\{1\}, \\ td_{k,u} = ta_{k,u} + ts_{k,u},\ \forall k \in K, u \in U\setminus\{|N|\} \end{cases} \quad (12)$$

(2) 客流状态约束

根据前文假设, 客流加载须满足以下约束.

$$q_{k,u,v}(t) \leqslant x_k,\ \forall k \in K, u, v \in U, t \in T, \tag{13}$$

$$\sum_{k \in K} q_{k,u,v}(t) = 1,\ \forall u, v \in U, t \in T, \tag{14}$$

$$q_{k,u,v}(t) \leqslant q_{k,u,v}(t-1),\ \forall k \in K, u, v \in U, t \in T, \tag{15}$$

$$t - td_{k,u} \leqslant (1 - q_{k,u,v}(t)) \times M,\quad \forall k \in K, u, v \in U, t \in T. \tag{16}$$

约束 (13) 保证了乘客乘坐的列车必须为开行的列车; 约束 (14) 保证所有乘客需求都将被列车服务; 约束 (15) 保证将乘客依次加载在提供服务的列车车次上; 约束 (16) 保证 t 时刻到达的乘客只能乘坐在其之后出发的列车上.

当列车k到达车站u时, 列车内目的地为u站的乘客下车, 同时车站内候车乘客登上列车. 此状态更新过程如公式 (17) 和 (18) 所示, 分别表示列车k在u站的上车人数以及下车人数. 且列车从车站出发时的乘客人数不超过列车总能力, 如约束 (19)-(20) 所示.

$$b_{k,u} = \sum_{t \in [0, td_{k,u})} \sum_{u < v \leqslant |N|} q_{k,u,v}(t) \times \lambda_{u,v}(t),\quad \forall k \in K,\ \forall u, v \in U, \tag{17}$$

$$d_{k,u} = \sum_{t \in [0, T)} \sum_{1 \leqslant v < u} q_{k,v,u}(t) \times \lambda_{v,u}(t),\quad \forall k \in K, u, v \in U, \tag{18}$$

$$c_{k,u} = c_{k,u-1} + b_{k,u} - d_{k,u},\ \forall k \in K, u \in U \backslash \{|N|\}, \tag{19}$$

$$c_{k,u} \leqslant C_k,\ \forall k \in K, u \in U \backslash \{|N|\}. \tag{20}$$

另外, 在高峰时段, 客流密集到达车站, 当列车k从车站u出发时, 可能由于列车能力限制而导致部分乘客滞留, 将滞留乘客人数记为$\psi_{k,u}$.

$$\psi_{k,u} = \theta_{k,u} - b_{k,u},\ \forall k \in K, u \in U \backslash \{|N|\}. \tag{21}$$

在车站u等待列车k的乘客人数$\theta_{k,u}$包括两部分. 一部分是在列车$k-1$与k出发时间窗内时变客流$\lambda_{u,v}(t)$的累积; 另一部分为前序列车出发后滞留乘客$\psi_{k-1,u}$.

$$\theta_{k,u} = \psi_{k-1,u} + \sum_{t \in [td_{k-1,u}, td_{k,u})} \sum_{v > u} \lambda_{u,v}(t),\quad \forall k \in K, u \in U \backslash \{|N|\}. \tag{22}$$

基于以上状态, 动作, 奖励以及环境约束的定义, 关于时刻表优化问题的多阶段序列决策过程可以直观地描述为如图 1 所示的状态转移过程.

在该决策框架下, 时变客流的累积与列车运行在环境中进行交互. 对于任意决策阶段k(即开行第k列车), 在其初始时刻, 智能体观测前序列车$k-1$服务结束后的环境状态, 准确捕捉因前序列车容量受限而导致的各站滞留客流$\psi_{k-1,u}$以及$k-1 \rightarrow k$间隔内时变客流$\lambda_{u,v}(t)$的累积, 以此作为当前决策的输入. 智能体的策略网络基于观测状态s_k, 输出动作a_k, 并经解码函数映射为当前列车k相对于前一列车的发车间隔h_k. 环境仿真器接收发车间隔h_k后, 在底层驱动微观时间t上的物理演化.

首先, 基于列车运行约束严格推演出列车k在全线各站的到达时刻$ta_{k,u}$与出发时刻$td_{k,u}$; 随后, 在此间隔时间窗内持续到达客流, 并与前序列车滞留乘

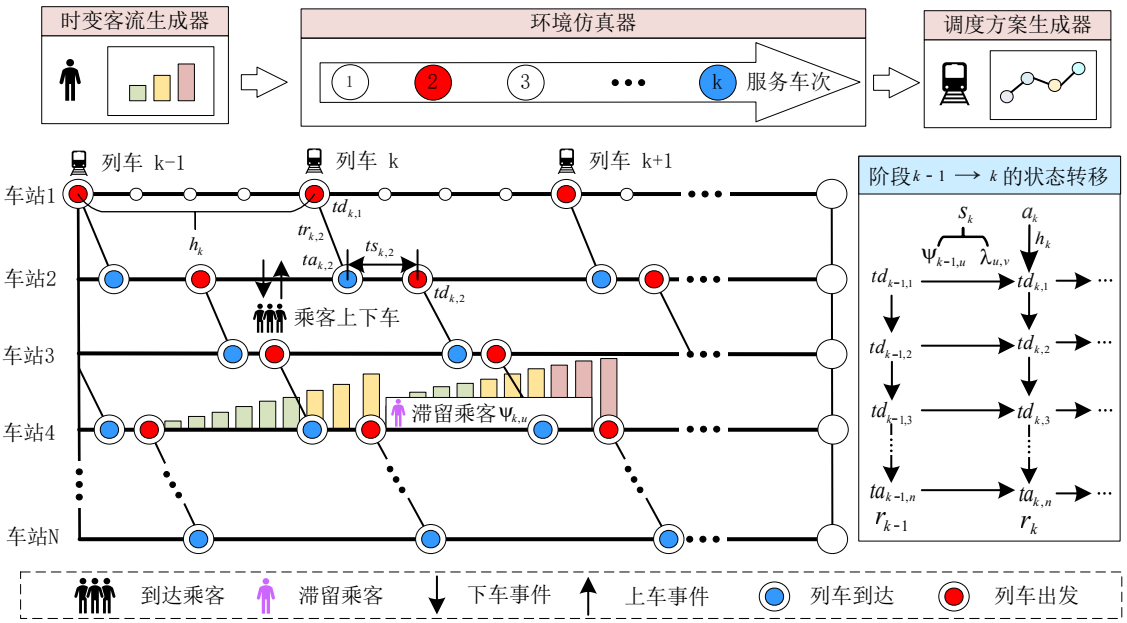


图1　环境状态转移过程

客$\psi_{k-1,u}$汇合成总候车乘客队列; 当列车k停靠站台时, 触发乘客乘降事件, 并严格遵循列车能力约束. 当列车完成乘降作业后, 环境状态发生确定性转移. 未能上车的乘客被截断并转化为当前阶段的新滞留状态$\psi_{k,u}$, 同时更新列车的载客状态$c_{k,u}$, 两者共同构成下一阶段的新状态s_{k+1}. 仿真器同步评估该阶段内累积的乘客等待时间、产生的列车开行成本以及站台拥挤超限水平, 计算并输出阶段奖励r_k.

上述过程 (即$s_k \xrightarrow{a_k} r_k, s_{k+1}$) 随着服务车次$k$的递增而依次推进, 直至规划运营时段结束, 最终计算回合总奖励$r = \sum r_k$. 这一基于车次步进的状态转移机制, 彻底解耦了高层策略探索与底层复杂动态的计算冲突, 为 SAC 算法规避维度灾难、实现高效的连续空间寻优提供了严密的仿真环境支撑.

3 软演员-评论家算法

大规模城市轨道交通时刻表优化问题由于其 NP-hard 的性质而难以解决, 使得现有的启发式算法无法在短时间框架内提供解决方案. 软演员-评论家算法 (Soft Actor-Critic, SAC) 为解决此类问题提供了一种全新的框架, 其具有以下优势:

(1) 相较于传统方法,SAC 算法天然支持对连续动作空间进行建模与求解, 从根本上避免了动作离散化所引发的“维度灾难”与训练不稳定性;

(2) 采用离策略 (Off-policy) 学习框架高效复用历史经验以提升样本利用率, 并通过将熵正则化项纳入优化目标, 实现了探索与利用的动态平衡;

(3) 其内置的温度参数自适应调节机制有效降低了模型对超参数调优的依赖, 进一步提升了算法的泛化能力与部署效率.

因此, 本文采用 SAC 算法作为列车时刻表优化决策框架. SAC 算法已较为成熟, 限于篇幅, 在此不过多赘述其相关理论, 仅对网络架构及训练流程进行简要介绍, 有关算法的详细内容可参考文献 [28].

3.1 网络架构

SAC 算法基于 Actor-Critic 架构, 其核心网络模块包含策略网络 (Actor)、双评价网络 (Critic1、Critic2) 及双目标评价网络 (Target Critic1、Target Critic2), 它们分别用于输出动作、评估状态-动作对的价值, 并通过目标网络进行稳定的软更新. 其中, 策略网络 (Actor) 网络以当前的环境状态作为输入, 在连续的高维动作空间中输出动作的概率分布特征, 进而重参数化采样生成发车间隔决策指令. 双 Critic 网络 (Q1,Q2) 是在对策略价值的综合评估时取两网络输出的较小者$\min(Q_{\theta_1}, Q_{\theta_2})$, 以解决奖励的“过估计”问题. 需要特别指出的是, 本架构中的两个 Critic 网络并非用于独立评估不同的子目标, 而是基于环境反馈的同一个奖励, 计算当前“状态-动作”对的期望回报. 双 Target Critic 网络 (T-Q1,T-Q2) 的结构与 Critic 网络相同, 但其参数不直接参与梯度的反向传播, 而是采用软更新机制平滑迭代来保持训练的稳定性. SAC 算法网络架构如图 2 所示.

3.2 训练流程

SAC 算法步骤可概括如下:

step1: 初始化. 采用随机网络参数θ_1, θ_2和ϕ, 并分别初始化双 Critic 网络$Q_{\theta_1}, Q_{\theta_2}$, 以及 Actor 网络

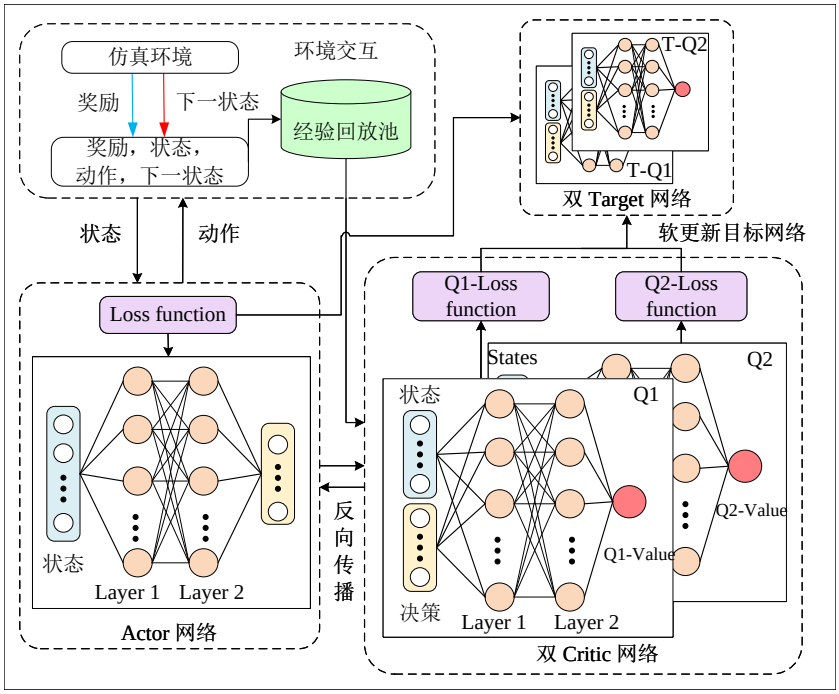


图2 Soft Actor-Critic 深度强化学习网络架构

$\pi_\phi(a|s)$及参数同步的双目标 Q 网络, 同时初始化经验回放池 B 并设置超参数.

step2: 环境交互. 获取动作序列, 并逐车次 k 应用到环境中, 进行列车运行与乘客加载过程, 获得奖励r_k, 并转移至下一状态s_k. 将状态、动作、奖励、下一状态(s_k, a_k, r_k, s_{k+1})等经验存入回放池.

step3: 经验回放. 当回放池数据量|B|≥N 时, 随机采样批量经验.

step4: 网络更新. 先依据式 (23) 计算目标 Q 值,

$$y = r(s_k, a_k) + \gamma \cdot (\min_{j=1,2} Q_{\theta_j^-}(s_k, a_k) - \alpha \log \pi_\phi) \tag{23}$$

并采用公式 (24) 更新当前双 Critic 网络.

$$L_Q(\theta) = \mathbb{E}[\frac{1}{2}(Q_\theta(s_k, a_k) - (r_i + \gamma V_\theta(s')))^2]. \tag{24}$$

step5: 策略更新. 重参数化采样动作, 依据式 (25) 进行 Actor 网络的更新.

$$L_\pi(\phi) = \mathbb{E}[\alpha \log \pi_\phi(a_k|s_k) - \min_{j=1,2} Q_\theta(s_k, a_k)]. \tag{25}$$

step6: 目标评价网络依据公式 (26) 进行软更新.

$$\theta_j^- = \tau\theta_j + (1-\tau)\theta_j^-, (j = 1, 2). \tag{26}$$

step7: 循环迭代. 重置环境状态, 回到 step2, 进行新回合的交互, 收集经验并更新模型参数, 直至达到最大训练轮数.

Step8: 输出. 最优策略ϕ.

重复以上训练过程, 直到达到预定义的训练轮数为止, 即可获得对应的最优策略. 当智能体的训练完成后, 无论问题的规模如何, 经过训练的控制器都可以生成具有令人满意性能的可行决策方案.

在线决策是在训练经验学习的基础上采用参数共享的方式, 利用深度 Critic 网络中的调度经验, 在目标场景下进行无学习的决策. 在线决策流程简述如下, 首先根据不同客流场景初始化地铁交互环境, 并通过参数共享的方式加载训练中学习得到的目标 Q 网络, 设置随机探索率为 0, 依次执行 step2-8, 并输出调度方案.

4　算例分析

为全面考察方法的有效性和适用性, 先通过小规模算例验证方法的合理性; 再依托广州地铁 8 号线开展大规模的实例分析, 评估所提方法在真实环境中的表现. 实验采用 Python 编程语言, 在 Pycharm 平台上进行编程实现, 运行环境为配备 AMD Ryzen 7 4800H 处理器和 16GB RAM 的本地计算机.

4.1　小规模算例

通过小规模算例验证算法可行性及寻找最优超参数配置, 在此算例中, 线路由 5 个车站组成, 研究时段为 1 小时, 高峰时段为 [20,40]. 设置列车最小发车间隔为 2 分钟, 最大发车间隔为 15 分钟, 列车能力为 150 人, 最大可用列车数 10.

(1) 超参数设置

在深度强化学习方法中, 优化结果不仅取决于算法和数据, 更在一定程度上取决于超参数的设置. 采用网格搜索的方法来寻找较优超参数配置, 为便于操作, 仅对关键敏感参数进行探索, 并遵循控制变量的原则来设计这些实验组. 具体地, 探究了经验回放批量大小 (Batch Size, BS)、学习率 (Learning Rate, LR) 和目标网络更新率 (τ) 等关键超参数对智能体最终性能、学习效率和稳定性的影响. 经过多种组合的尝试获得较优超参数配置, 限于篇幅, 在图 3 中展示仅部分代表性调参试验组合. 图中蓝色曲线是经过平滑后的奖励曲线, 主要体现训练的稳定性, 红色虚线及数值表示获得的最优奖励.

从图 3 可以看出, 过大或过小的回放批量大小都会导致训练的不稳定性, 而过低的学习率可能限制模型的探索和学习能力. 在 Group A 中, 当 LR=3×10^{-3}, τ=0.01 下表现最佳, 表明在标准学习率下, 一个更快的 τ(更频繁的目标网络更新) 可能是有益的. 综合考虑训练曲线的收敛性能以及获得的最佳奖励, 最终选取的主要超参数设置如表 1 所示.

(2) 可行性分析

为验证本文所提出方法的有效性, 将列车等间隔发车作为基线方案, 与本文所提出方法求解的结果对比. 为清晰起见, 分别绘制两种方案下的列车运行图, 如图 4 所示, 不同区间列车运行线颜色的深浅表示列车的满载率, 车站柱状图表示累积客流的多少, 也表示站台的拥挤度. 从对比图中不难发现, 由于基线运行图中列车均衡发车, 导致在客流高峰期大量客流聚集在站台上. 此外, 列车在不同区间的满载率也不均衡. 而对于 SAC 算法优化的时变客流驱动下的非均衡列车运行图, 在客流高峰期, 组织列车密集发车, 降低了车站的拥挤度. 具体的, 对比等间隔基线方案, 利用 SAC 算法优于化后时变客流下的列车时刻表对应的乘客总等待时间大幅降低, 从 6,431 min 减少至 3,264 min, 降低了 49.25%. 同时本文所优化的方法通过非均衡的发车间隔, 将峰值车站等待人数从 170 降低至 82, 减少了 51.76%.

其次, 我们通过文献 [17] 提出的动态规划方法得到的最优解对比 SAC 求解质量. 动态规划算法在列车数量为 10 列的情况下, 乘客总等待时间为 3050 min, 与 SAC 算法解的 Gap 为 7.01%. 值得说明

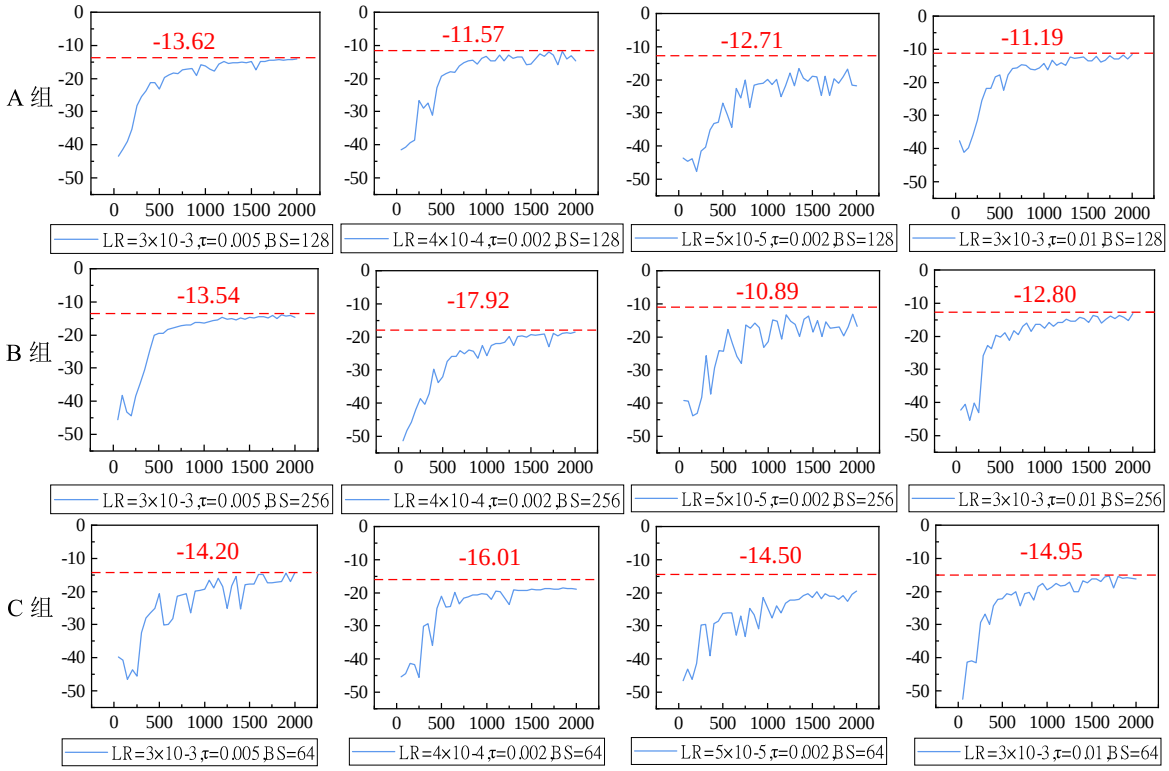
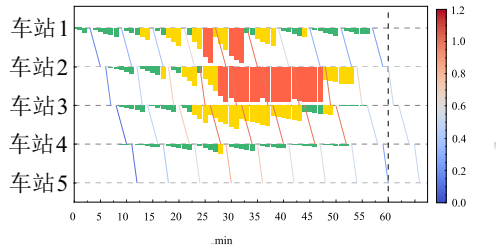


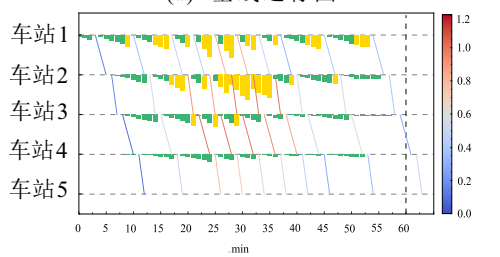
图3 超参数搜索实验

表1 超参数配置及说明

参数	取值	描述
α	0. 6	熵温度系数. 平衡奖励最大化和熵最大化(平衡探索与利用).
γ	0. 99	折扣系数. 控制未来奖励的重要性.
LR	3×10^{-3}	学习率. 梯度下降期间更新网络权重的步长, 决定了对网络参数的调整的幅度.
τ	0. 01	目标网络的软更新系数.
Batch Size	128	每次训练更新从经验回放池采样的样本数.
HiddenDim	128	神经网络每一层的神经元数量. 决定了神经网络的特征表达能力与模型复杂度.



(a) 基线运行图



(b) SAC优化运行图

图4 列车运行图对比

的是, 尽管针对小规模的数值实验, 可以利用动态规划算法等精确方法获得最优解, 但在实际应用中存在两大挑战. 一方面, 实际运行中所需列车数量呈现动态性, 它取决于需求驱动的状态决策演化. 另一方面, 精确求解方法泛化能力有限, 难以适应不同客流扰动场景. 而我们提出的深度强化学习方法通过环境交互来自行探寻最优调度策略, 在大规模时刻表决策问题中具有显著优势.

4.2 真实案例分析

以广州市地铁 8 号线为例, 进一步验证提出方法在大规模问题中的有效性. 该线路由 13 个站点组成, 研究时段为 [6:00-22:00], 客流需求如图 5 所示, 此时段内共到达乘客 164,710 人次, 最大可用列车数 150 列, 列车最大容量设为 1000. 依然以等间隔发车的均衡时刻表作为基线方案, 此方案下, 共运行列车 150 列, 可统计出乘客的总等待时间为 480, 953.00min, 人均等待时间 2.92 min. 在此方案下, 尽管开行大量列车, 但在早高峰存在明显的站台拥挤. 而且列车满载率时空分布严重不均衡, 平均列车满载率为 44. 84%, 列车早晚高峰期满载率极高, 平峰时段列车满载率较低, 造成运力严重浪费.

4.2.1 算法性能

实验首先将 SAC 算法与同属于离线策略 (Off-Policy) 的深度确定性策略梯度 (DDPG) 算法以及孪生延迟深度确定性策略梯度 (TD3) 算法进行比较. 训练收敛曲线如图 6(a) 所示. 训练曲线显示, 所有算

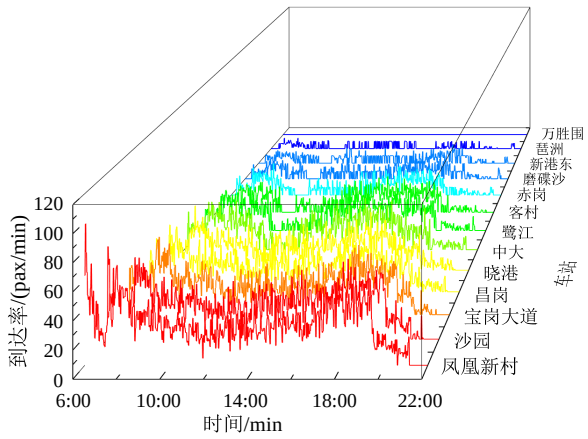
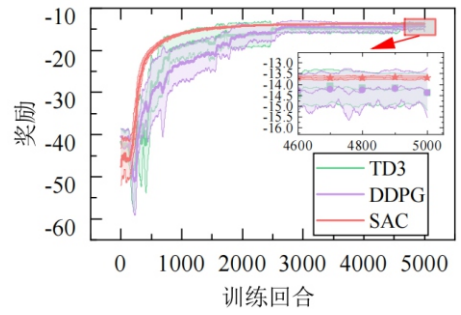
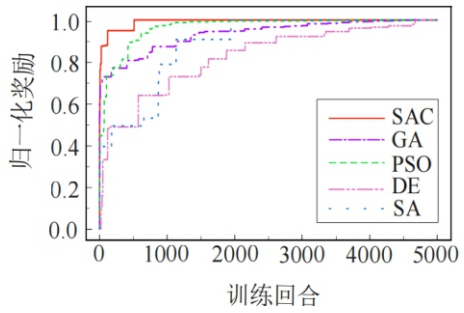


图5　广州地铁 8 号线的乘客需求



(a)　深度强化学习算法训练曲线对比



(b)　启发式方法收敛性能对比

图6　不同算法性能对比

法的奖励均随训练回合数的增加整体呈上升趋势, 表明随着训练进行, 不同算法均向最优解逼近, 这也证明本文设计的调度模型、交互环境及奖励函数是合理的. 其中, SAC 算法无论是在收敛速度还是最终解的质量上, 均展现出显著的优越性, 这得益于其网络架构与最大熵机制, 使得 SAC 学习更为平滑, 能更快聚焦高价值的策略, 有助于加速收敛. 而 DDPG 和 TD3 算法过度依赖外部噪声, 在不同随机种子下方差很大, 且训练也不够稳定.

此外, 基于智能算法的启发式方法是求解大规模列车时刻表问题的常用方法, 如模拟退火算法 (Simulated Annealing, SA), 遗传算法 (Genetic Algorithm, GA), 粒子群算法 (Particle Swarm Optimization, PSO), 以及差分进化算法 (Differential Evolution, DE), 因此将 SAC 算法与传统启发式方法进行对比, 如图 6(b) 所示. 由于深度强化学习方法与启发式方法训练过程存在显著差异, 因此在与启发式算法的对比中, 我们聚焦于各算法从劣质解到优质解的相对改进过程, 并以奖励提升性能作为纵轴, 进而直观、公平地对比不同算法的收敛动态. 从图中可以清晰看到, 与相比于其他启发式算法,SAC 算法性能提升速度更快, 能够更快的找到最优解.

不同算法对应的求解结果统计见表 2. 从最优奖励来看, SAC 获得了最高的累积奖励 (−13.74), 证明了此算法在寻找高质量优化方案上的优势. 另外, 在优化列车时刻表时, 考虑了灵活的列车数量, 所有算法优化的列车开行数量维持在 120-137 列, 其中 SAC 算法优化的列车数量为 132 列. 相较于基线方案, 列车数量明显减少, 但是人均等待时间增长并不明显 (从 2.92 min 增加到 3.13 min), 而对比同等车数方案 (Equal quantity, EQ) 则体现出显著的优化提升. 与传统启发式算法相比, 尽管 GA、PSO 等启发式算法优化后的列车数量略少, 但是却导致了更大的乘客等待时间, 并明显增加了站台拥挤度. 在算法求解时间方面, 整体而言, 尽管启发式方法相较于深度强化学习算法求解速度略快, 但是在后文算法泛化能力实验中表明, 深度强化学习算法具备较好的泛化能力, 经过预训练后, 针对不同的场景能在极短时间内求解出满意解, 而启发式算法面对不同场景则需要重

表2　不同方法调度方案性能比较

方案	奖励	开行列车数/列	总等待时间/min	人均等待时间/(min/pax)	列车平均满载率	平均站台拥挤人数(pax/min)	计算时间/s
基线	−104.38	150	480,953.20	2.92	44.84%	38.58	–
GA	−13.88	120	548,720.50	3.55	55.71%	46.88	772
PSO	−15.18	130	523,777.80	3.18	51.34%	41.95	828
DE	−14.46	122	589,661.80	3.58	54.87%	47.21	832
SA	−14.71	122	637.427.70	3.87	54.95%	51.14	778
TD3	−15.35	137	634,133.50	3.85	48.88%	50.82	1070
DDPG	−14.75	137	635,780.60	3.86	49.00%	50.96	1111
EQ	−109.82	132	619,309.60	3.76	50.33%	49.69	–
SAC	−13.74	132	515,542.30	3.13	50.75%	41.90	1096

新求解, 耗时显著增加.

利用 SAC 方法优化的列车运行图如图 7 所示, 以热力线条展示了不同方案下的列车满载率变化, 为我们评估运力与客流的匹配度提供了直观的表示. 相比于基线运行图列车满载率的极度不均衡, 优化后的列车运行图大部分区间的列车满载率都处于一个合理的范围, 没有出现过度拥挤与空载的情况. 相较于基线, 优化后峰值站台累积人数减少 33.40%, 进一步保证了站台的安全性, 由此可见优化后的需求响应式列车时刻表具有较强的现实意义.

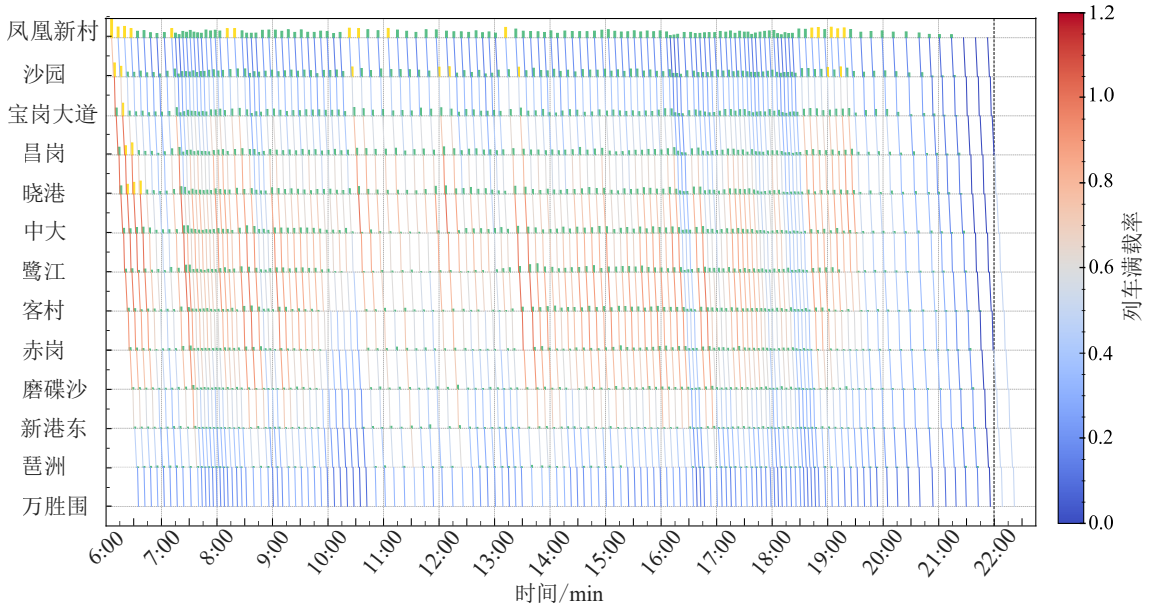


图7 SAC 优化后的列车运行图

进一步地, 对需求与发车间隔的匹配度进行分析, 如图 8 所示. 显然, 客流总量与列车发车间隔总体呈现负相关性, 即在客流平峰时段, 列车发车间隔较大, 而在客流高峰期, 列车密集发车. 这种现象也是我们在地铁日常运营中所希望的, 可以有效的降低乘客的等待时间和站台上客流的累积人数. 优化方法针对不同时段客流特征, 通过动态调整发车间隔, 精准匹配波动的客流出行需求, 这是典型的以服务为导向的需求响应式调度策略.

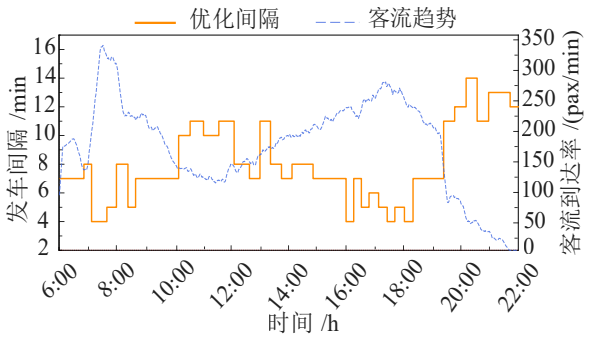


图8 发车间隔与需求匹配度

我们从微观层面考察该优化方案的质量, 重点剖析其在车站服务水平方面的表现, 图 9 通过小提琴图直观呈现了不同方案各站点乘客候车时间的分布特征. 图中显示, 优化后乘客的候车时间分布主体基本位于 [0,8] 分钟, 这也印证了优化后列车运行图在服务水平上的优越性. 且相比于等间隔方案, 如晓港站出现的极端情况, 乘客最大等待时间接近 40 min, 现实中这可能会导致乘客的大量流失, 而优化后可以将其控制在 20 min 以内.

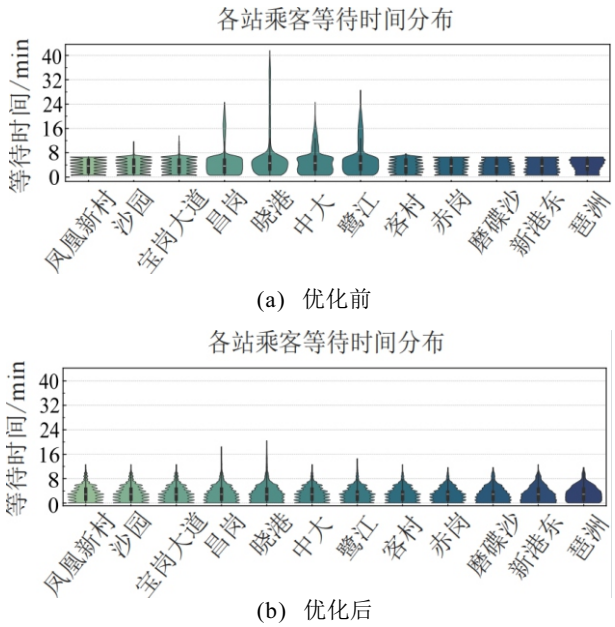
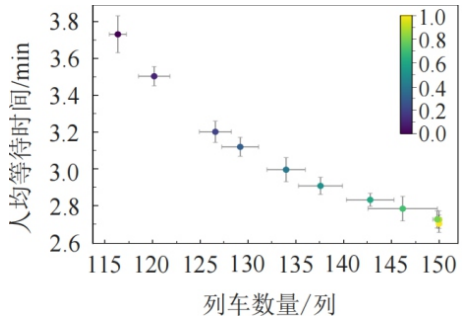


图9 优化前后各站等待时间分布图

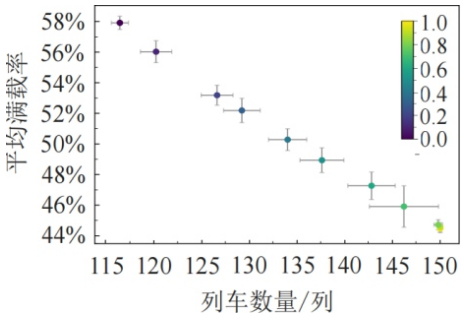
4.2.2 帕累托解集

通过引入帕累托系数σ作为决策偏好参数, 我们构建了一系列最优解集, 揭示了关键性能指标之间

的非线性关系, 并识别出特定场景下的最优运营策略. 由于强化学习的随机性 (初始权重、探索过程等), 单次运行的结果可能存在偶然性. 取不同随机种子进行多次实验并对结果进行统计分析, 得到帕累托解集如图 10 所示.



(a)　列车数量与等待时间权衡曲线



(b)　列车数量与列车满载率边际效益

图10　不同系数 σ 下的帕累托解集

图 10(a) 中的分析显示, 在σ从 0.0 增至 0.3 的阶段, 等待时间降幅显著, 投资回报率最高; 随后在σ处于 0.4 至 0.7 的阶段, 服务水平稳步提升, 代表了从"平衡最优"向"乘客体验优先"的过渡地带; 而当σ大于 0.8 后, 由于列车数量限制, 即使继续增加权重, 等待时间的改善也不明显. 由图 10(b) 可见, 列车数量与列车满载率之间存在明显的边际效益关系, 且完全符合边际效益递减规律. 整个过程中, 平均发车数从 116 趟增至系统上限 150 趟, 平均等待时间从 3.73 分钟降至 2.70 分钟, 但代价是列车平均满载率由 57.5% 降至 43.9%. 综合分析表明, σ在 0.5 至 0.7 的区间内实现了服务质量、运营效率与系统稳定性的最佳平衡.

基于此, 本文提出一个分级运营策略建议: 1) 标准运营情境, 推荐采纳$\sigma\approx0.6$ 的优化方案, 作为兼顾运营成本与乘客等待时间的日常时刻表; 2) 资源受限情境, 可采用$\sigma\approx0.1$-0.2 的策略, 以牺牲部分乘客等待时间换取较少的运营成本; 3) 高品质服务情境, 采用$\sigma\geqslant0.8$ 的策略可为乘客提供更优质的服务, 极大缩短等待时间, 但同时也需要通过技术手段或设施升级来缩短发车间隔, 以提升线路能力.

4.2.3　泛化能力测试

本小节旨在评估预训练的深度强化学习模型在不同客流扰动场景下进行列车时刻表优化任务的泛化能力. 模型被部署于四个独立的、具有挑战性的典型扰动场景中进行零样本推断. 具体地, 场景设计如下: A. 整体缩放场景: 将基线客流的所有 OD 值统一乘以 1.2, 模拟需求的增长; B. 高峰期时移场景: 将客流高峰时间段移动指定偏移量, 测试模型对客流时变性的识别能力; C. 客流波动场景: 增加客流的随机波动性 10%, 模拟需求的不确定性; D. 叠加场景: 综合以上三种变换, 创建一个总量、高峰时段和波动性均发生改变的极端复杂环境.

为了体现 SAC 算法的泛化优势, 与性能最优的启发式算法 GA 进行了对比实验, 结果见表 3 所示. 实验表明, 面对单一扰动场景 A、B、C, 尽管启发式算法能够在 1 000 秒内获得更优的结果, 但 SAC 经过预训练后, 能够在极短时间内生成满意的运营方案, 并且优化结果与启发式算法相差不大. 结果进一步表明, SAC 算法在应对单一维度的客流扰动时表现出较好的鲁棒性和适应性, 可以满足实时性要求.

表3　不同场景下的泛化结果对比

场景	A		B		C		D	
客流量/人	190, 524.00		164 710.00		170, 244.00		207, 948.00	
算法	SAC	GA	SAC	GA	SAC	GA	SAC	GA
列车数/列	130	133	129	122	131	121	132	133
平均等待时间/(min/人)	3.92	3.90	3.41	3.57	3.35	3.74	6.78	4.50
平均满载率	59.76%	58.29%	51.94%	54.83%	53.12%	56.95%	64.45%	63.84%
平均站台聚集人数/人	59.90	59.47	45.00	47.16	45.70	50.95	113.00	75.01
计算时间/s	0.05	954.20	0.03	905.90	0.03	912.70	0.04	1 012.20

然而, 在面对多重扰动叠加的复杂场景 D 时, SAC 性能出现了一定程度的下降, 这也体现了模型的泛化边界. GA 算法得到较优结果耗时超 1 000 秒, 而 SAC 仅耗时 0.04 秒便生成方案. 尽管方案的平均等待时间增至 6.78 min/人, 但可以看到列车满载率维持在 64.45%, 这客观反映了在运力逼近极限时, 模型对响应时效、资源投入与服务质量所做出的综合权衡. 针对 SAC 算法在多重扰动叠加的复杂场景下计算性能下降的问题, 需要进一步进行客流控制或灵活编组列车来保证服务质量和安全性, 这也将是一个未来具有挑战性的工作.

5　结　论

1) 设计了基于深度强化学习的优化方法, 求解

需求响应式地铁列车时刻表优化问题. 将问题刻画为马尔可夫决策过程, 为智能体提供训练和学习环境, 综合考虑了“人-车-站”一体化的多维复合奖励函数, 开发了一种基于自适应发车间隔和列车数量的多目标 SAC 优化算法提升求解效率.

2) 基于小规模算例, 验证了 SAC 求解的需求响应式列车时刻表相对于均衡时刻表的优势; 并通过广州市地铁 8 号线进行仿真实验, 结果表明, 所提出的方法相对于其他人工智能方法及启发式算法具有较快的收敛速度和求解效率. 另外, 此方法能够均衡列车开行数量、乘客等待时间、站台拥挤度等运营指标.

3) 针对不同客流扰动场景, 方法能够在极短时间内生成满意的运营方案, 证明方法具有良好的扩展性和泛化能力.

4) 未来进一步考虑融合列车停站方案、灵活编组等多种运营要素的列车时刻表优化研究, 同时进一步提升算法在跨线路等场景的泛化部署能力, 提升城市轨道交通整体调度的智能化水平. 另外, 网络运营环境下的换乘衔接优化问题是一个值得关注的研究方向.

参考文献 (References)

[1] 牛惠民. 轨道列车时刻表问题研究综述[J]. 交通运输系统工程与信息, 2021, 21(5): 114-124.
(Niu H M. Literature review on rail train timetabling problems[J]. Journal of Transportation Systems Engineering and Information Technology, 2021, 21(5): 114-124.)

[2] Barrena E, Canca D, Coelho L C, et al. Exact formulations and algorithm for the train timetabling problem with dynamic demand[J]. Computers & Operations Research, 2014, 44: 66-74.

[3] 周文梁, 黄裕, 邓连波. 考虑运行节能和车底运用的城轨时刻表优化[J]. 铁道科学与工程学报, 2023, 20(2): 473-482.
(Zhou W L, Huang Y, Deng L B. Optimization of train schedule for urban rail considering operation energy-saving and train circulation planning[J]. Journal of Railway Science and Engineering, 2023, 20(2): 473-482.)

[4] Niu H M, Zhou X S. Optimizing urban rail timetable under time-dependent demand and oversaturated conditions[J]. Transportation Research — Part C: Emerging Technologies, 2013, 36: 212-230.

[5] Niu H M, Zhou X S, Gao R H. Train scheduling for minimizing passenger waiting time with time-dependent demand and skip-stop patterns: Nonlinear integer programming models with linear constraints[J]. Transportation Research Part B: Methodological, 2015, 76: 117-135.

[6] 李佳杰, 柏赟, 周雨鹤, 等. 基于站外限流与时刻表调整的地铁换乘站大客流协同控制[J]. 铁道学报, 2020, 42(5): 9-18.
(Li J J, Bai Y , Zhou Y H, et al. Integrated model on inbound passenger flow control and timetable regulation at transfer station[J]. Journal of the China Railway Society, 2020, 42(5): 9-18.)

[7] Tian X P, Niu H M. Optimization of demand-oriented train timetables under overtaking operations: A surrogate-dual-variable column generation for eliminating indivisibility[J]. Transportation Research Part B: Methodological, 2020, 142: 143-173.

[8] Bucak S, Demirel T. Train timetabling for a double-track urban rail transit line under dynamic passenger demand[J]. Computers & Industrial Engineering, 2022, 163: 107858.

[9] 陈治亚, 欧阳灏, 徐光明, 等. 基于出行可靠性的城轨线路多时段发车频率优化[J]. 铁道科学与工程学报, 2022, 19(12): 3526-3535.
(Chen Z Y , Ouyang H, Xu G M, et al. Multi-period frequency optimization of urban rail lines based on travel reliability[J]. Journal of Railway Science and Engineering, 2022, 19(12): 3526-3535.)

[10] Shi J G, Yang J, Yang L X, et al. Safety-oriented train timetabling and stop planning with time-varying and elastic demand on overcrowded commuter metro lines[J]. Transportation Research — Part E: Logistics and Transportation Review, 2023, 175: 103136.

[11] 张春田, 戚建国, 杨凯, 等. 基于两阶段分布鲁棒优化的列车停站方案与时刻表协同研究[J]. 控制与决策, 2023, 38(4): 1065-1073.
(Zhang C T, Qi J G, Yang K, et al. Two-stage distributionally robust optimization for integrated train stop planning and timetabling[J]. Control and Decision, 2023, 38(4): 1065-1073.)

[12] 钟林环, 徐光明, 邓连波, 等. 面向灵活编组的城轨列车开行频率与时刻表综合优化[J]. 交通运输工程与信息学报, 2024, 22(2): 104-115.
(Zhong L H, Xu G M, Deng L B, et al. Integrated optimization of train frequency and timetable for urban railway trains for flexible train composition[J]. Journal of Transportation Engineering and Information, 2024, 22(2): 104-115.)

[13] Yang L X, Qi J G, Li S K, et al. Collaborative optimization for train scheduling and train stop planning on high-speed railways[J]. Omega, 2016, 64: 57-76.

[14] Cacchiani V, Qi J G, Yang L X. Robust optimization models for integrated train stop planning and timetabling with passenger demand uncertainty[J]. Transportation Research — Part B: Methodological, 2020, 136: 1-29.

[15] Gao R H, Niu H M. A priority-based ADMM approach for flexible train scheduling problems[J]. Transportation Research — Part C: Emerging Technologies, 2021, 123: 102960.

[16] Li S Q, Zhu X N, Shang P, et al. Optimizing a shared freight and passenger high-speed railway system: A

multi-commodity flow formulation with Benders decomposition solution approach[J]. Transportation Research — Part B: Methodological, 2023, 172: 1-31.

[17] Niu H M, Tian X P, Zhou X S. Demand-driven train schedule synchronization for high-speed rail lines[J]. IEEE Transactions on Intelligent Transportation Systems, 2015, 16(5): 2642-2652.

[18] 张京辉, 陈曦, 李博睿. 考虑车数限制的城市轨道交通非对称时刻表优化[J]. 控制与决策, 2023, 38(9): 2632-2640.
(Zhang J H, Chen X, Li B R. Optimization of asymmetric timetable in urban rail transit considering train quantity restriction[J]. Control and Decision, 2023, 38(9): 2632-2640.)

[19] Wang F S, Wang P L, Zhu Z X, et al. Robust optimization of train timetables with short-length and full-length services considering uncertain passenger volume and service choice behavior[J]. Transportation Research — Part C: Emerging Technologies, 2024, 169: 104855.

[20] 张浩然, 李君, 邢立宁, 等. 大模型与智能优化算法集成研究综述[J]. 控制与决策, 2026, 41(4): 871-891.
(Zhang H R, Li J, Xing L N, et al. A research review on integration of large models and intelligent optimization algorithms[J]. Control and Decision, 2026, 41(4): 871-891.)

[21] Šemrov D, Marsetič R, Žura M, et al. Reinforcement learning approach for train rescheduling on a single-track railway[J]. Transportation Research Part B: Methodological, 2016, 86: 250-267.

[22] Li W Q, Ni S Q. Train timetabling with the general learning environment and multi-agent deep reinforcement learning[J]. Transportation Research Part B: Methodological, 2022, 157: 230-251.

[23] 俞胜平, 韩忻辰, 袁志明, 等. 基于策略梯度强化学习的高铁列车动态调度方法[J]. 控制与决策, 2022, 37(9): 2407-2417.
(Yu S P, Han X C, Yuan Z M, et al. A policy gradient reinforcement learning algorithm for high-speed railway dynamic scheduling[J]. Control and Decision, 2022, 37(9): 2407-2417.)

[24] 代学武, 吴越, 石琦, 等. 基于优先经验回放可迁移深度强化学习的高铁调度[J]. 控制与决策, 2023, 38(8): 2375-2388.
(Dai X W, Wu Y, Shi Q, et al. A transferable deep reinforcement learning high-speed railway rescheduling method based on prioritized experience replay[J]. Control and Decision, 2023, 38(8): 2375-2388.)

[25] Yang W L, Liu L Y, Yuan H F, et al. Unified scheduling model for high-speed train timetable optimization and rescheduling based on deep reinforcement learning[J]. IEEE Transactions on Intelligent Transportation Systems, 2025, 26(6): 8178-8193.

[26] Ying C S, Chow A H F, Chin K S. An actor-critic deep reinforcement learning approach for metro train scheduling with rolling stock circulation under stochastic demand[J]. Transportation Research — Part B: Methodological, 2020, 140: 210-235.

[27] Wang S Y, Chow A H F, Ying C S. Adaptive and flexible rail transit network service dispatching as a partially observable Markov decision process[J]. Transportation Research — Part C: Emerging Technologies, 2025, 179: 105286.

[28] Wen L H, Hu L Y, Zhou W, et al. Soft actor-critic deep reinforcement learning for train timetable collaborative optimization of large-scale urban rail transit network under dynamic demand[J]. IEEE Transactions on Intelligent Transportation Systems, 2025, 26(5): 7021-7035.

作者简介

高如虎 (1989−), 男, 副教授, 博士, 博士生导师, 主要研究方向为交通运输组织优化, E-mail: grh@mail.lzjtu.cn;

刘伟 (2001−), 男, 硕士生, 主要研究方向为交通运输规划与管理、智慧交通, E-mail: 12241008@stu.lzjtu.edu.cn;

焦治铎 (2006−), 男, 本科生, 主要研究方向为智慧物流, Email: 32291786@qq.com.