控制与决策
Control and Decision

基于优先经验回放可迁移深度强化学习的高铁调度

代学武, 吴越, 石琦, 崔东亮, 俞胜平

引用本文:
代学武,吴越,石琦,崔东亮,俞胜平. 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.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2023.0479

基于优先经验回放可迁移深度强化学习的高铁调度

代学武†, 吴 越, 石 琦, 崔东亮, 俞胜平

(东北大学 流程工业综合自动化国家重点实验室，沈阳 110819)

摘 要: 高铁行车调度是一个复杂的多阶段序列决策问题,需要考虑列车、线路设备等条件,且决策空间随问题规模的增大呈指数增长. 而深度强化学习 (DQN) 兼备强大的搜索和学习能力,为高铁调度提供了新的解决方案,但存在经验利用效率低、迁移能力差等问题. 本文提出一种基于优先经验回放可迁移深度强化学习的高铁调度方法. 将包含股道运用计划等约束的高铁调度问题构建为多阶段序列决策过程,为提高算法的迁移能力,提出一种新的支持源域和目标域共享的状态向量和动作空间. 为提高经验的利用效率和算法的收敛速度,设计了一种融合优先经验回放的深度Q网络训练方法. 以徐兰线小规模案例为源域问题的经验学习实验表明,所提算法的经验利用效率和算法收敛速度优于传统DQN算法,并可适当增大优先级指数和调节权重参数以改善其收敛性能. 以京沪线繁忙路段的晚点案例为目标域问题,本文提出的在线决策算法相比于经典的混合整数规划算法,决策时间平均减少约75 %,且在近77 %的案例中,总晚点时间的性能损失在15 %以内.

关键词: 高速铁路；调度算法；深度强化学习；状态向量；动作空间；优先经验回放

中图分类号: U292.4；TP18 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2023.0479

引用格式: 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.

A transferable deep reinforcement learning high-speed railway rescheduling method based on prioritized experience replay

DAI Xue-wu†, WU Yue, SHI Qi, CUI Dong-liang, YU Sheng-ping

(State Key Laboratory of Synthetical Automation for Process Industries，Northeastern University，Shenyang 110819，China)

Abstract: High-speed railway train operation rescheduling is a complex multi-stage sequential decision problem which requires consideration of trains, line equipment and other conditions, and the decision space grows exponentially with the scale of the problem. Deep reinforcement learning (DQN), which combines powerful search and learning capabilities, provides a new solution for high-speed railway rescheduling, but has problems such as inefficient use of experience and poor transferability. This paper proposes a transferable deep reinforcement learning high-speed railway rescheduling method based on prioritized experience replay. The high-speed railway rescheduling problem which contains constraints such as the track utilization plan is constructed as a multi-stage sequential decision-making process, and in order to improve the transferability of the algorithm, a new state vector and action space that supports the sharing of source and target domains is proposed. A deep Q-network training method combining prioritized experience replay for high-speed railway rescheduling is designed to improve the efficiency of experience utilization and the convergence speed of the algorithm. The experience learning experiments using the small-scale cases of Xulan line as the source domain problems show that the experience utilization efficiency and algorithm convergence speed of the proposed algorithm are better than that of the traditional DQN algorithm, and the convergence performance can be improved by appropriately increasing the priority exponent and adjusting the weight parameters. Taking the delay cases in the busy section of Jinghu line as the target domain problems, the online decision making algorithm proposed in this paper reduces the decision time by about 75 % on average compared with the classical mixed integer programming algorithm, and the performance loss of the total delay time is within 15 % in nearly 77 % of the cases.

Keywords: high-speed railway；dispatching algorithm；reinforcement learning；state vector；action space；prioritized experience replay

收稿日期: 2023-04-16；**录用日期:** 2023-07-11.

基金项目: 国家自然科学基金项目 (61790574).

责任编委: 叶丹.

†通讯作者. E-mail: daixuewu@mail.neu.edu.cn.

0 引 言

随着我国高速铁路“十纵十横”规划的稳步推进,高铁在我国现代综合交通运输体系中的骨干作用日益凸显,而路网规模的不断扩大和列车开行数量的持续增长,使繁忙线路行车密度接近饱和,行车追踪间隔更小,时刻表冗余时间更少,在发生灾害天气、设备故障等突发事件导致列车运行延误、运行秩序紊乱时,列车晚点的横向和纵向传播速度更快、范围更大,对调度指挥系统的调度决策和应急响应能力提出了更高要求. 尽管高铁调度采取分散自律调度集中系统(CTC)等新技术,解决了调度指令下发等办公流程自动化的问题,但调度方案的制定还是以人工凭经验为主,劳动强度大,人工调度的精细化不足. 高铁调度的本质是通过调整到发时间、行车顺序等来疏解冲突,减少延误,决策过程中需要考虑列车属性、线路资源、列车之间的关联和冲突等诸多因素,是一个复杂的组合优化问题,其决策空间随着车站和列车数量的增加呈指数级增长,且对实时性要求较高.

高铁调度的研究方法可以分为4类:数学规划方法、仿真系统模拟方法、智能优化方法和机器学习方法. 基于数学规划的高铁调度方法,如文献[1]建立了混合整数规划模型,在此基础上,为了提高求解速度,相关学者提出了分支定界算法[2]、拉格朗日松弛算法[3]等,但在大规模问题下,会产生组合爆炸,使计算耗时难以满足高铁调度实时响应的要求. 而且数学规划方法依赖于精确的数学模型,高铁系统的复杂性使精确建模十分困难,通常需要对一些约束进行简化或者忽略,导致数学规划所得调度方案的可行性需要进一步验证. 而基于仿真系统模拟的调度方法,通过构建仿真系统模拟真实的高铁运行过程,获得的调度方案可行性更好,如周妍等[4]设计的高铁客运系统的半实物仿真平台和杨鹏鑫等[5]设计的列车运控调度协同仿真器等. 虽然所得调度方案的可行性得以提高,但仿真计算代价更高,难以满足快速决策的要求.

综合考虑调度方案的可行性和算法求解的实时性要求,启发式智能优化方法求解高铁调度引起了更多重视,如蚁群算法(ACO)[6-7]、粒子群算法(PSO)[8-9]和萤火虫算法(FA)[10]等. 但是这类基于群体智能和迭代优化的启发式算法受制于基于问题的启发式规则,普遍存在迁移性较差的问题,在求解不同运行场景时(如初始晚点不同、晚点列车不同、不同时刻表、不同调度线路等),均需要重新迭代求解,计算量仍然偏大.

高铁调度也是典型的多阶段序列决策问题,是一个逐次决策每个列车在每个车站到发时间的过程. 而强化学习兼备了强大搜索能力和自主学习能力[11],适合求解高铁调度的多阶段序列决策问题. 近年来,国内外开始提出用强化学习解决高铁调度. 如Šemrov等[12]首次提出将强化学习算法Q-learning应用于列车调度. 在此基础上,文献[13-15]从线路全局的角度出发定义状态和动作,虽然决策信息丰富,但是其动作空间随着问题规模的扩大呈指数增长,导致学习效率降低,收敛速度缓慢. 文献[16]提出将运行图作为状态,将列车的发车顺序作为动作,但是由于运行图上的列车运行线无法完整表征线路设备等信息,导致状态表征不全,且动作与问题规模(列车、车站数量)耦合紧密,可迁移性较差. 文献[17]利用局部的高铁运行信息作为状态,并将列车的前进和停止作为动作,提高了算法对于特定列车调度问题的适应能力,但是控制列车启停的动作很容易导致决策链过长,训练过程收敛缓慢. 上述研究在求解高铁调度问题时,主要通过交互学习收敛到重调度方案,对无学习的快速调度方法研究较少,这种迭代优化的方式一般很难满足实时性要求. 因此,本文进一步研究算法的迁移应用能力,以平衡大规模问题对于求解精度和实时性的要求. 此外,所提出算法在具体应用时,需要先基于大量案例进行源域经验学习,以获取高铁调度的可迁移深度Q网络. 然后,对于待求解的运行场景通过目标域在线决策进行无学习的快速调度,并利用冲突检测与冲突消解来保障所做决策的合理性,从而实现实时调度.

本文针对强化学习方法应用于高铁调度时存在的经验利用效率低、收敛速度慢和迁移能力差等问题,提出一种基于优先经验回放可迁移深度强化学习的高铁调度方法. 首先,针对高铁调度问题对于求解精度和实时性要求较高等挑战,提出源域经验学习和目标域在线决策的算法框架;然后,通过解耦线路属性、时刻表和问题规模等信息,设计可迁移的状态向量和动作空间;最后,为了提高学习过程的经验利用效率,实现在有限资源下的快速收敛,设计一种融合优先经验回放的深度Q网络训练方法.

1 问题描述

本文主要研究高铁的行车调度,即当列车发生运行延误或者线路能力受限(如大风限速)时,调度员需要根据调度区段的路网和列车运行情况,通过改变列车到发时间、改变列车运行顺序、压缩停站时间、调节区间运行时间、利用冗余时间等调度手段,尽量减

少延误并尽快恢复正常运行. 高铁调度本质是一个约束条件多且复杂的组合优化问题,其决策空间会随着问题规模的扩大呈指数增长. 同时,高铁调度也是一个典型的多阶段序列决策问题,即列车在每一个车站的到发作业可视作一个决策事件,调度的过程是一系列到发时间的决策过程,基于当前阶段的路网运行状态,决策列车的下一个到发时间,如此循环,直至完成所有列车在所有车站到发时间的决策.

1.1　参数定义

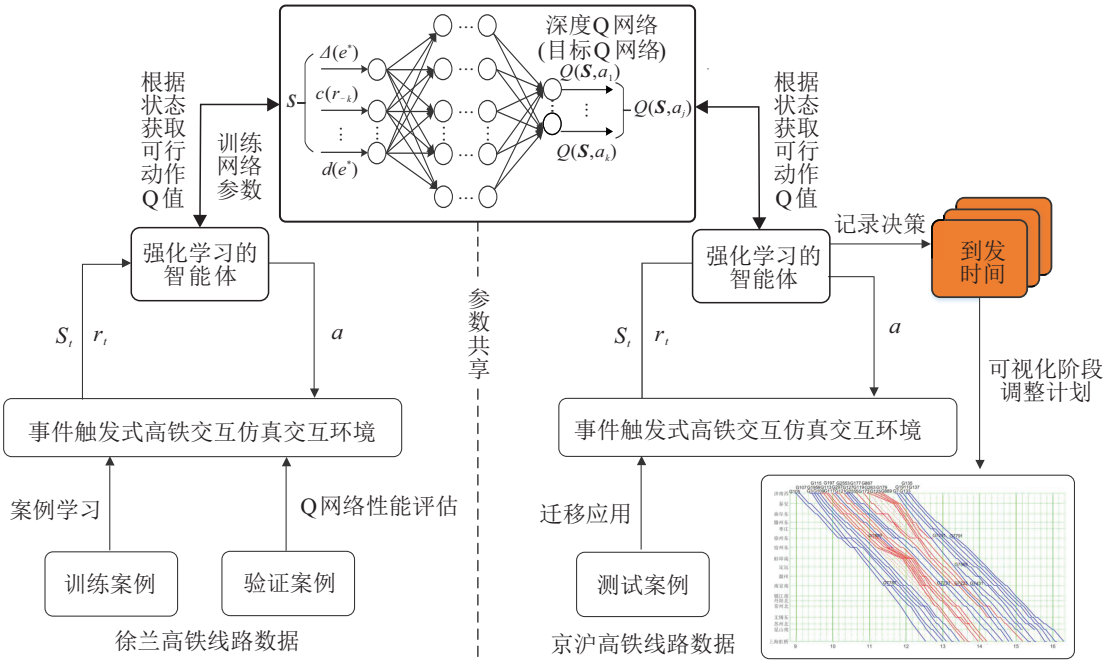
本文所涉及的重要符号和变量定义如表1所示.

表 1　符号和变量定义

符号和变量	含　义
$a^*_{i,j}$	列车 i 在车站 j 的计划到站时间
$d^*_{i,j}$	列车 i 在车站 j 的计划发车时间
$a_{i,j}$	列车 i 在车站 j 的实际到站时间
$d_{i,j}$	列车 i 在车站 j 的实际发车时间
$t_{i,k}$	列车 i 在区间 k 的最小区间运行时间
$t_{i,j}$	列车 i 在车站 j 的最小停留作业时间
C^*_j	车站 j 的到发线总数
C^t_j	时刻 t 车站 j 的到发线实际使用数量
$G^*_{i,j}$	列车 i 在车站 j 计划使用的股道
$G_{i,j}$	列车 i 在车站 j 实际使用的股道
e^*	到发事件列表中最小决策时间对应的到发事件
$\Delta(e^*)$	到发事件 e^* 相对于其计划到发时间的晚点时间
$r(e^*)$	事件 e^* 所属的资源类型
$c(r_i)$	事件 e^* 所属资源附近第 i 个资源的可用股道数量
$\boldsymbol{S}$	面向列车运行的状态向量
a	面向列车决策的动作
$Q(\boldsymbol{S},a)$	在状态向量 $\boldsymbol{S}$ 下执行动作 a 的未来奖励的期望
p_c	经验池中学习经验 c 对应的优先级
$P(c)$	经验池中学习经验 c 被随机重要性采样抽中的概率
ω_i	样本集中学习经验 i 对应的动态权重

1.2　基于可迁移深度强化学习的高铁调度框架

为解决强化学习用于高铁调度时存在的收敛速度慢、迁移能力差、快速求解难的问题,本文从兼顾调度求解精度和速度的角度出发,提出一种源域经验学习与目标域在线决策相结合的可迁移深度强化学习方法. 如图1所示,所提出基于可迁移深度强化学习的高铁调度框架包括源域经验学习和目标域在线决策两大模块.



(a)　源域经验学习　　(b)　目标域在线决策

图 1　基于可迁移深度强化学习的高铁智能调度框架

图1(a)中的经验学习模块是在源域问题上,如徐(州)-兰(州)高铁徐州东站到兰考南站区段,基于各种晚点场景下,通过与仿真环境的交互,获取大量交互数据(称作经验)保存于经验池中,通过对经验池中经验的抽样学习,将经验保存在深度Q网络中. 基于丰富的高铁运行场景,对所选源域问题对应的调度区段设置不同基本图、不同晚点延误(晚点列车数量、晚点多少、发生晚点的空间位置)和不同路网状态(限速、最小运行时间)等,构造出训练案例集和验证案例集. 在经验池抽样学习中,每个迭代学习回合都从训练案例中随机抽取一个作为本回合学习的初始场景,初始化高铁运行仿真环境,通过后继的动态交互,智能体对各种不同的决策进行探索,并在每个学习回合结束后利用验证案例中的全部晚点场景评估深度Q

网络的调度能力,经过若干回合的训练在大量交互学习的基础上获得一个稳定的深度Q网络.

图1(b)中的在线决策模块是针对目标域问题,如京沪高铁济南西站到上海虹桥站区段,基于源域经验学习所获得的深度Q网络,通过参数共享的方式构造具有相同Q网络的决策智能体,并利用该Q网络直接根据路网的当前状态进行动作选择,制定决策方案. 由于采用了多阶段决策方式,在制定目标域问题的调度方案时,仍然需要智能体与仿真环境进行不断交互,对各个列车在各个车站的到发时间进行决策,形成调度方案.在线决策时,首先根据目标域问题需要调度的晚点信息初始化高铁运行仿真环境,然后根据列车在交互环境中返回的状态信息,完全凭借经验学习模块共享的深度Q网络决策动作控制高铁交互环境的列车运行,记录多阶段决策过程的到发时间,直到所有列车完成行驶,从而得到阶段调整计划. 需要指出的是,在目标域执行在线决策求解时,只需要同环境进行一个回合的交互,与源域学习不同,在目标域决策过程中不再更新Q网络,不再对动作进行探索,而是直接利用Q网络进行决策.

1.3 高铁调度的目标和奖励函数

高铁调度的目标是将列车从运行延误和运行秩序紊乱状态中快速恢复为按原有行车计划行车,包括恢复列车的计划到发时间和股道运用计划等. 如前所述,本文考虑的行车调度主要通过调整列车到发时间和列车运行顺序来完成,常用的调度目标是最小化所有列车在全部车站的总晚点时间. 考虑某调度区段包含m个车站,需要对n辆列车的到发时间进行调度,则总晚点时间最小的目标函数定义为

$$\min \sum_{i=1}^{n}\sum_{j=1}^{m}[|a_{i,j}-a_{i,j}^{*}|+|d_{i,j}-d_{i,j}^{*}|]. \tag{1}$$

其中:$a_{i,j}^{*}$和$d_{i,j}^{*}$分别为列车i在车站j的计划到站时间和发车时间,决策变量$a_{i,j}$和$d_{i,j}$为调度调整后新的到发时间.

除了调整到发时间,随着高铁大密度、公交化、多路网的发展趋势[18],车站股道的利用率接近设计容量,列车发生延误后,晚点列车和正点列车在接发进路、股道使用等方面也更易发生资源占用冲突,需要调整股道运用计划以疏解冲突. 但调整股道运用计划涉及重新安排接发车进路,不仅增加调度员的工作量,还需要临时更换旅客乘车站台,影响乘客满意度. 因此,需要考虑股道运用的调整,进而将股道的使用情况纳入目标函数,以减少接发车股道的调整次数,在列车运行延误时尽可能保持列车按原计划使用股道. 最大化股道运用计划的目标函数定义为

$$\max \sum_{i=1}^{n}\sum_{j=1}^{m}\varphi(t_a,t_d,G_{i,j},G_{i,j}^{*}).$$

$$\varphi(t_a,t_d,G_{i,j},G_{i,j}^{*})=\begin{cases}1,\ t_a=t_d=0\ \text{ and }\ G_{i,j}=G_{i,j}^{*};\\ 0,\ t_a>0\ \text{ or }\ t_d>0\ \text{ or }\ G_{i,j}\neq G_{i,j}^{*}.\end{cases} \tag{2}$$

其中:$G_{i,j}^{*}$和$G_{i,j}$为整数变量,代表股道编号,分别为列车i在车站j接发车使用的计划股道和调度后的实际使用股道;$t_a=|a_{i,j}-a_{i,j}^{*}|$和$t_d=|d_{i,j}-d_{i,j}^{*}|$分别为到站和发车时间的晚点量. 最大化股道运用计划可以更好地反映在高密度行车时车站到发线的分配和使用情况,以实现在总晚点时间相差不大的情况下,优先选择到发线调整次数更少的调度方案.

综上所述,兼顾最小化总晚点时间和最大化股道运用计划的需求,本文提出的综合性目标函数为

$$\min \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}[|a_{i,j}-a_{i,j}^{*}|+|d_{i,j}-d_{i,j}^{*}|]}{\sum_{i=1}^{n}\sum_{j=1}^{m}\varphi(t_a,t_d,G_{i,j},G_{i,j}^{*})}. \tag{3}$$

高铁调度问题是典型的延迟奖励问题,一般需要所有列车在调度区段终到或交出后,才能全面准确地评估调度方案的好坏. 基于此,结合高铁调度的优化目标,设计面向高铁调度的奖励函数如下所示:

$$r=\begin{cases}0,\ t<t_{\text{end}};\\ \mu\dfrac{w_1\cdot\sum\limits_{i=1}^{n}\sum\limits_{j=1}^{m}\varphi(t_a,t_d,G_{i,j},G_{i,j}^{*})}{w_2\cdot\sum\limits_{i=1}^{n}\sum\limits_{j=1}^{m}[|a_{i,j}-a_{i,j}^{*}|+|d_{i,j}-d_{i,j}^{*}|]},\ t=t_{\text{end}}.\end{cases} \tag{4}$$

其中:μ为奖励增益因子,w_1为股道运用计划一致程度的权重参数,w_2为总晚点时间的权重参数. 调度过程中的奖励设置为零,以减弱不完整信息对于智能体的影响. 终止时刻的奖励设置为目标函数的倒数,并引入权重参数调节两项目标的比重,以提高奖励的目标导向性和可区分性.

1.4 高铁行车调度约束

高铁列车运行有严格的安全约束要求,列车的到达、出发、通过车站股道和站间区间行驶均需要满足一系列约束条件,强化学习所获得的解需要满足这些约束条件才是一个可行的有效调度方案. 高铁行车调度的发车时间约束、区间最小运行时间约束、通

过和停站时间约束、车站容量约束、越行约束如下所示:

$$d_{i,j} \geqslant d^*_{i,j}, \tag{5}$$

$$a_{i,j+1} - d_{i,j} \geqslant t_{i,k}, \tag{6}$$

$$d_{i,j} - a_{i,j} \geqslant t_{i,j}, \tag{7}$$

$$C^t_j \leqslant C^*_j, \tag{8}$$

$$(d_{i,j} - d_{i',j})(a_{i,j+1} - a_{i',j+1}) > 0. \tag{9}$$

上述约束的详细定义详见文献[14]. 此外, 本文将追踪间隔时间约束进一步细分为4种情况, 如图2所示.

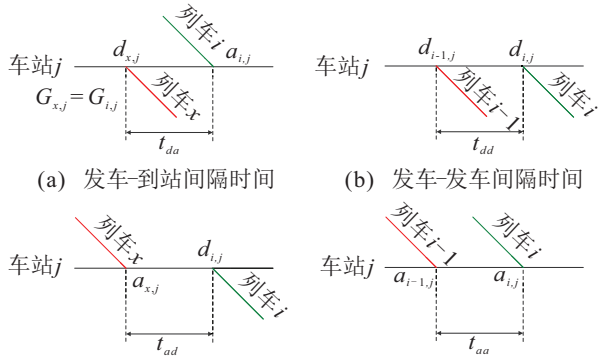


图 2　追踪间隔时间示意图

1) 发车-到站间隔时间约束.

自列车 x 由车站 j 出发的发车时间 $d_{x,j}$ 起, 至同方向另一列车 i 到达该站 j 同一股道时的到站时间 $a_{i,j}$ 之间需要满足最小间隔时间 t_{da}, 如下所示:

$$a_{i,j} - d_{x,j} \geqslant t_{da}[1 - (G_{i,j} \oplus G_{x,j})]. \tag{10}$$

其中: $G_{i,j}$ 和 $G_{x,j}$ 分别为列车 i 和 x 在车站 j 的实际使用股道; $\oplus$ 为异或运算. 如果 $G_{i,j}$ 与 $G_{x,j}$ 相同, 则 $G_{i,j} \oplus G_{x,j} = 0$, 即列车 i 和 x 需要考虑同一股道最小发车-到站间隔时间 t_{da}; 如果 $G_{i,j}$ 与 $G_{x,j}$ 不相同, 则 $G_{i,j} \oplus G_{x,j} = 1$, 即不需要考虑 t_{da}.

2) 发车-发车间隔时间约束.

列车 i 在车站 j 的发车时间 $d_{i,j}$ 与相邻的前一辆列车 $i-1$ 在车站 j 的发车时间 $d_{i-1,j}$ 之间需要满足最小时间间隔 t_{dd}, 如下所示:

$$d_{i,j} - d_{i-1,j} \geqslant t_{dd}. \tag{11}$$

3) 到站-发车间隔时间约束.

自列车 x 到达车站 j 时的到站时间 $a_{x,j}$ 起, 至该站发出同方向另一列车 i 时的发车时间 $d_{i,j}$ 之间需要满足最小间隔时间 t_{ad}, 如下所示:

$$d_{i,j} - a_{x,j} \geqslant t_{ad}. \tag{12}$$

4) 到站-到站间隔时间约束.

列车 i 在车站 j 的到站时间 $a_{i,j}$ 与相邻的前一辆列车 $i-1$ 在车站 j 的到站时间 $a_{i-1,j}$ 之间需要满足最小时间间隔 t_{aa}, 如下所示:

$$a_{i,j} - a_{i-1,j} \geqslant t_{aa}. \tag{13}$$

由于高铁调度求解的搜索空间巨大, 上述约束的存在导致大量决策动作是不佳的, 形成较差的调度方案, 导致稀疏奖励问题, 从而使学习收敛速度缓慢. 本文利用上述已知的高铁行车约束, 设计面向高铁调度强化学习的冲突检测和冲突消解辅助模块, 辅助智能体进行动作选择, 以减少无效动作, 避免无效探索, 提高学习效率. 所提出的冲突检测和消解主要通过对智能体拟探索的动作执行约束(5)～(13)的检查, 如果约束不满足, 则检测到冲突, 然后按照冲突消解规则, 修正该动作(即列车的到发时间), 并作为智能体执行的动作. 冲突检测和冲突消解能够有效避免无效的动作导致环境无法执行动作的问题, 减少无效探索, 提高学习效率.

2　高铁调度的可迁移状态向量和动作空间

2.1　面向列车运行的状态向量

由于高铁调度问题的复杂性, 运行状态难以准确表征, 状态向量的设计需要充分全面考虑, 对各种路网运行状态需要有较好的区分度. 另一方面, 不同调度任务由于问题规模不同、时刻表不同、线路布局不同等, 在设计状态向量时还需要充分考虑不同调度任务的共性, 最好能实现状态向量与线路属性、时刻表和问题规模等信息的解耦, 从而提高强化学习求解高铁调度问题的迁移能力.

综上考虑, 为实现更好的迁移性, 本文提出一种面向列车的强化学习调度, 以及相应的路网运行状态向量 $\boldsymbol{S}$, 将列车在车站的到达或者出发视作一个调度决策事件, 高铁调度过程是这样一系列事件组成的对所有列车在各个车站到发时间进行调整的多阶段决策过程, 直至完成所有列车在所有车站到发时间的决策. 设 e^* 表示多阶段决策中的当前决策事件, 设当前决策事件为列车 i 在车站 j 的到站(或者出发)事件, 所提出状态向量 $\boldsymbol{S}$ 包括待调度列车 i 的晚点量、列车 i 附近资源的拥挤程度以及列车 i 的位置速度信息3个部分, 如下所示:

$$\begin{aligned}\boldsymbol{S} = &\\ &\{\Delta(e^*), c(r_{-k}), \ldots, c(r_{-1}), c(r_0), c(r_1),\\ &\ldots, c(r_k), r(e^*), t(e^*), d(e^*)\}.\end{aligned} \tag{14}$$

第1部分 $\Delta(e^*)$ 是待调度列车的晚点量, 表示事件 e^* 对应的实际到发时间相对于计划到发时间的偏离时间. $\Delta(e^*)$ 可直观反映待调度列车 i 的延误情况,

相较于以列车的到发时间$a_{i,j}$和$d_{i,j}$作为状态元素的强化学习列车调度,提出以晚点量作为状态元素,可以避免状态向量与时刻表耦合,有助于经验的迁移,同时也克服了状态空间维数与问题规模(车站和列车数量)关联,减少了搜索空间.

第 2 部分$c(r_{-k}),\dots,c(r_{-1}),c(r_0),c(r_1),\dots,c(r_k)$是待调度列车附近资源的拥挤程度,$r_0$表示待调度列车$i$所处的当前资源(车站或者闭塞区间),$r_{-k}$和$r_k$分别表示列车$i$的前方和后方第$k$个资源,$c(r_k)$表示资源$r_k$的可用股道数量. 这部分状态相当于使用一个滑动窗口追踪待调度列车前后资源的使用情况,窗口中每个资源的可用股道数量用来描述该资源的拥挤程度. 采用滑动窗口定义的状态元素,仅包含与调度直接相关的资源信息,而非调度线路的全部资源,有利于状态向量在不同线路的迁移.

第3部分$r(e^*),t(e^*)$和$d(e^*)$是待调度列车的位置速度信息,分别表示当前资源的类型(站间区间资源或车站资源)、当前事件e^*所对应的列车作业的计划作业时间(区间计划运行时间或计划停站时间)和最小作业时间(最小区间运行时间或最小车站停站时间). 将与调度列车在当前位置密切相关的作业时间信息、作业类型作为状态的一部分,避免与线路和时刻表绑定,以具有更好的迁移性.

综上,本文提出的面向列车运行的状态向量,考虑到列车的运行状态在很大程度上仅取决于以目标列车为中心的环境信息,通过避免与线路属性、时刻表和问题规模等的强关联,可更好地实现迁移学习.

2.2 面向列车决策的动作空间

为了提高学习效率,动作空间的设计应尽量减少动作的数量. 在面向线路的强化学习调度中,通常将n辆列车的到发事件绑定到一起作为一个动作向量,这样的动作向量是n维的,其动作空间的大小与列车数量呈指数关系,存在维数灾难的问题. 本文设计的动作采用面向列车的决策方法,将动作定义为待调度列车i延误时间的变化量,用于刻画待调度列车i晚点的减少或增加程度,如下所示:

$$a=\Delta(e^*_{\text{next}})-\Delta(e^*). \tag{15}$$

其中:$\Delta(e^*)$为目标列车的当前到发晚点;e^*_{next}为e^*对应的下一个到发事件;$\Delta(e^*_{\text{next}})$为$e^*_{\text{next}}$对应的实际到发时间相对于计划到发时间的晚点,即下一到发晚点;a为动作,表示下一到发晚点相对于当前到发晚点的增量. 基于此,动作空间的定义如下所示:

$$\mathbf{A}=\{a\mid l\leqslant a\leqslant u,\ a\in\mathbf{Z}\}. \tag{16}$$

其中:l为动作的下界,u为动作的上界,$\mathbf{Z}$为整数集. 考虑到高铁在每个资源的冗余时间约为其使用时间的20%,一般不超过10 min,所以下一到发晚点相对于当前到发晚点可减少量的最大值为对应的冗余时间. 为了通用性,以10 min作为可能的最大减少量,即$l=-10$. 考虑到下一到发晚点可以由前一到发晚点与动作的加和来表示,所以动作不需要很大的值即可描述足够大的下一到发晚点. 为了兼顾动作空间的维数最小和动作的多样性,将最大晚点增量设置为120 min,即$u=120$.

综上所述,提出一种面向列车决策的动作空间,使动作空间的维数与列车数量解耦,有利于迁移到不同规模的问题. 此外,晚点增量的动作设计一方面增大了离散决策事件之间的间隔时间,减小决策链的长度,另一方面可以用较小的动作空间描述丰富的晚点信息.

基于所提出的状态向量和动作空间的定义,高铁调度的多阶段序列决策过程可以直观地描述为如图3所示的状态转移过程.

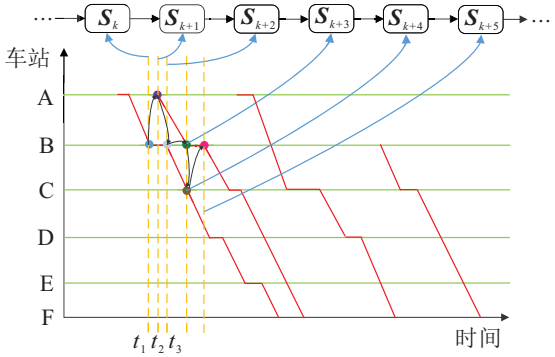


图 3 高铁调度的多阶段决策状态转移示意图

图3中的每个圆点表示一个列车到发决策事件,垂直虚线表示决策事件的时间,各个决策事件按照时间先后顺序排序形成决策序列. 在每个决策事件e_k处,设对应时间为t_1,对于该事件的待调度列车i,按照状态定义(14)构造状态向量$\boldsymbol{S}_k$. 智能体根据状态向量$\boldsymbol{S}_k$选择动作a_k,该动作决定了列车i下一个到发事件e_{k+2}的时间点t_3. 列车执行该动作a_k,直到下一个到发决策事件e_{k+1}在t_2处触发($t_2<t_3$). 对于事件e_{k+1}对应的待调度列车$i+1$,按照式(14)构造状态向量$\boldsymbol{S}_{k+1}$并选择动作a_{k+1},执行该动作直到下一个事件e_{k+2}. 以此类推循环执行,直到所有列车都到达终点或交出. 值得指出的是,本文所提出调度决策序列中,事件的执行顺序是按照事件发生的先后顺序执行的,相邻事件未必是同一个列车,事件e_k和事件e_{k+1}可能是两个不同的列车.

3　融合优先经验回放的学习和决策

3.1　融合优先经验回放的深度Q网络训练方法

为了实现高铁调度问题的源域经验学习与目标域在线决策,需要建立状态与动作的映射关系,并实现对于未知状态的决策. 基于此,设计的深度Q网络如图4所示.

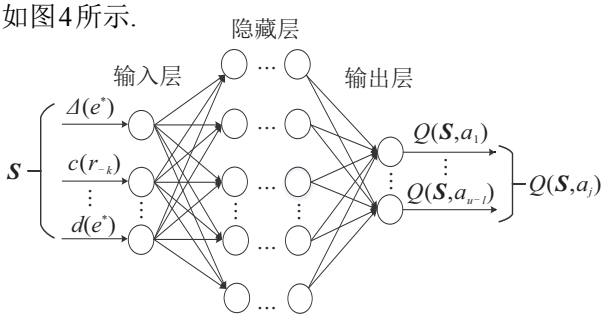


图 4　面向高铁调度的深度Q网络

图4中,以面向列车运行的状态向量$\boldsymbol{S}$作为深度Q网络的输入值,并以该状态下所有动作的Q值为输出值,其中输入层节点的个数与状态向量的维数一致,输出层节点的个数与面向列车决策动作空间的维数一致. 考虑到状态向量的维数为$n_{\boldsymbol{S}}=2k+5$,动作空间的规模大小为$n_{\mathbf{A}}=u-l$,在完成算法设计后二者都是固定的,因此能够实现不同线路、不同时刻表和不同晚点场景的深度Q网络具有相同的结构,实现不同的调度问题能够共享同一个深度Q网络,从而具备可迁移性.

深度强化学习算法在训练上述深度Q网络时,通过在经验池中随机抽样进行学习,但由于探索阶段存在大量尝试,容易抽取到不佳的经验,从而影响算法的学习效率和收敛速度. 针对此问题,将优先经验回放机制[19]融入深度Q网络的更新过程,其基本思想是根据学习经验的重要性有选择地进行抽样学习.

因为经典的深度强化学习(DQN)算法中的TD误差能够反映环境在决策时刻t从状态$\boldsymbol{S}_t$转移到其后继状态$\boldsymbol{S}_{t+1}$的学习量,所以用其来衡量每个学习经验的重要性. 设在决策时刻t的学习经验$(\boldsymbol{S}_t,a_t,r,\boldsymbol{S}_{t+1})$为经验池中第$c$个训练数据,其优先级定义如下:

$$p_c=|\delta_t|+\epsilon. \tag{17}$$

其中:ϵ为一个很小的正数,用于避免学习经验的TD误差为0,因此不被重视;δ_t为t时刻从状态$\boldsymbol{S}_t$转移到后继状态$\boldsymbol{S}_{t+1}$的TD误差,定义为

$$\delta_t=r+\gamma\max_a Q_t(\boldsymbol{S}_{t+1},a)-Q_e(\boldsymbol{S}_t,a_t). \tag{18}$$

这里:r为状态$\boldsymbol{S}_t$在动作a_t的作用下转变为后继状态$\boldsymbol{S}_{t+1}$所得的即时奖励,γ为奖励的折扣率.

由于优先经验回放会引入偏差,需要引入动态权重进行平衡,则样本集中学习经验i对应的动态权重ω_i定义如下:

$$\omega_i=\Big(\frac{P(i)}{\min\limits_c P(c)}\Big)^{-\beta}. \tag{19}$$

其中:β由β_0线性增长到1;$P(c)$为学习经验c被抽中的概率,定义为

$$P(c)=\frac{{p_c}^{\sigma}}{\sum\limits_{i=1}^{C}{p_i}^{\sigma}}. \tag{20}$$

这里:C为经验池的容量;σ为优先级指数,反映了学习经验的优先级p_c对于该经验被选中概率的影响程度,当$\sigma=0$时表示均匀随机采样. 基于此,将抽样过程中学习经验i的动态权重引入损失函数进行平衡,该损失函数定义如下:

$$E=\frac{\sum\limits_{i=1}^{B}\omega_i\cdot{\delta_i}^2}{B}, \tag{21}$$

其中B为优先经验回放中每次抽取的样本数.

综上所述,所设计的融合优先经验回放的深度Q网络训练方法如图5所示.

图5中,融合优先经验回放的经验池以满二叉树结构为基础,其叶子节点存储学习经验c的信息$(\boldsymbol{S}_t,a_t,r,\boldsymbol{S}_{t+1})$及其对应的重要程度${p_c}^{\sigma}$,非叶子节点只存储其左右孩子节点学习经验重要程度的和,依次递推至根节点. 图5所示的随机重要性采样过程如下,设根节点存储的优先级重视程度总和为τ,则采样B个学习经验时,先将区间$[0,\tau)$均匀分为B段,即$[0,\tau/B),[\tau/B,2\tau/B),\cdots,[(B-1)\tau/B,\tau)$,在每个区间内生成一个随机值$v_i$. 然后以根节点为初始父节点自上而下查找每个$v_i$对应的学习经验,规则如下:如果$v_i$小于等于父节点左孩子节点的值$v_l$,则将左孩子作为新父节点;否则$v_i=v_i-v_l$,并将右孩子作为新父节点. 重复上述操作直至叶子节点,获取其中的学习经验作为一个样本.

3.2　算法设计和实现

本文设计的基于优先经验回放可迁移深度强化学习的高铁调度算法由源域经验学习算法和目标域在线决策算法两部分组成. 源域经验学习算法流程如下.

step 1: 初始化基本参数,包括最大迭代次数N、当前回合数$n=0$、学习次数$l=0$、学习率α、奖励折扣因子γ、优先级指数σ、网络更新间隔K、经验池容量C、每次抽取的样本数量B、状态向量中考虑的前后固定资源数量k、计划时刻表和资源列表,并基于上述参数创建评估Q网络Q_e和目标Q网络Q_t.

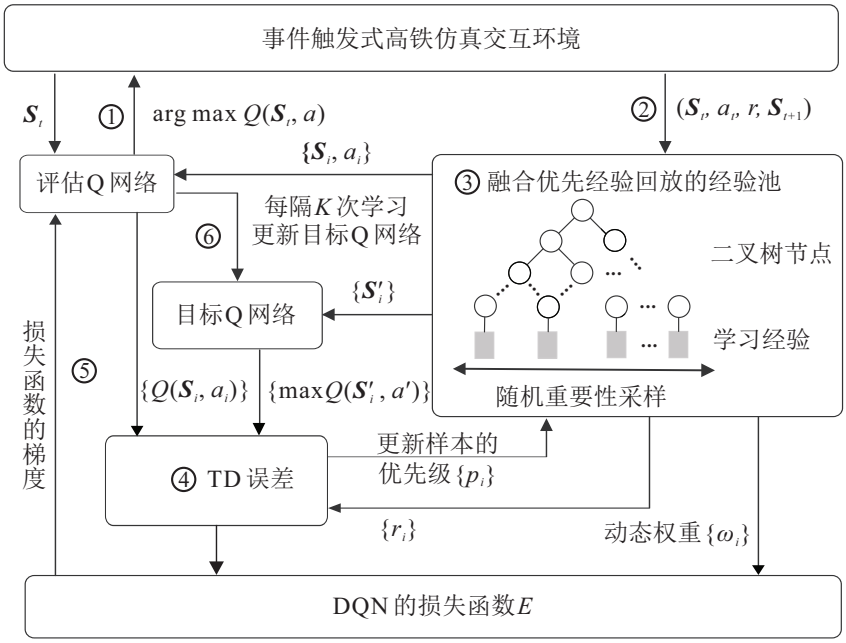


图5 融合优先经验回放的深度Q网络训练方法

step 2: 从源域问题中随机抽取一个初始晚点作为当前晚点场景,并根据回合数n计算当前回合的随机探索概率ε.

step 3: 开始一次决策,先获取交互环境决策时刻t的状态$\boldsymbol{S}_t$,再根据ε-greedy策略在动作空间$\mathbf{A}$中选择动作a_t控制目标列车运行,使高铁环境产生后继状态$\boldsymbol{S}_{t+1}$,然后计算奖励r,并将此转换经验$(\boldsymbol{S}_t, a_t, r, \boldsymbol{S}_{t+1})$存入经验池中,接着更新当前状态$\boldsymbol{S}_t = \boldsymbol{S}_{t+1}$.

step 4: 如果经验池中转换经验的数量小于其最大容量,则重复step 3,否则进入step 5,可以进行学习.

step 5: 如果学习次数满足网络更新间隔K,则使用评估Q网络Q_e的权重参数更新一次目标Q网络Q_t,否则直接进入step 6.

step 6: 开始一次学习,首先通过随机重要性采样在经验池获取B个经验作为样本集,并计算每个样本经验的动态权重ω_i,然后由式(21)计算的损失函数进行反向传播[20],并利用梯度下降算法更新评估网络Q_e的权重值,更新学习次数$l = l + 1$.

step 7: 判断当前状态是否为终止状态,即所有列车是否完成行驶,如果不是终止状态则返回step 3,否则进入step 8.

step 8: 更新当前回合数$n = n + 1$,并判断其是否大于最大迭代次数.如果不大于则返回step 2,否则完成学习过程,得到收敛的深度Q网络.

目标域在线决策算法是在源域经验学习算法的基础上采用参数共享的方式,利用深度Q网络中的调度经验,在目标域进行无学习的决策. 决策流程简述如下,首先根据目标域问题的晚点场景初始化高铁交互环境,并通过参数共享的方式加载源域经验学习得到的目标Q网络Q_t,设置随机探索概率$\varepsilon = 0$. 执行step 3,目标域在线决策过程中,根据交互环境提供的状态,仅利用深度Q网络来决策动作,然后判断当前状态是否为终止状态. 如果不是则返回step 3继续下一步决策,直到所有列车到达终点站或者交出,输出调整后的调度方案.

4 仿真验证和结果分析

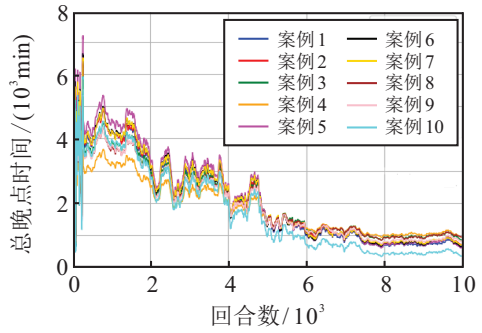
为验证所提出基于优先经验回放可迁移深度强化学习的高铁调度算法具有较好的经验利用效率和算法收敛速度,采用传统的深度强化学习算法DQN作为源域学习过程的对比算法. 同时,为了验证所提出算法具有可迁移的能力,利用源域经验学习得到的深度Q网络直接迁移到目标域大规模未学习的复杂场景进行晚点案例的调度,并与混合整数规划(MIP)算法的调度结果进行对比分析.

4.1 源域经验学习

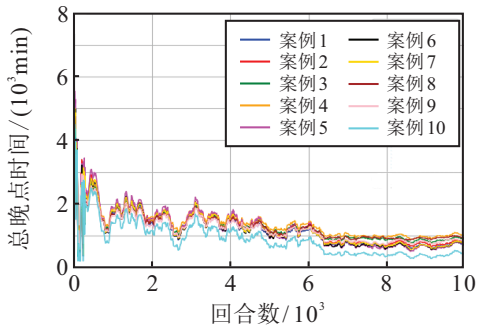
选取徐兰(徐州-兰州)高铁调度作为源域问题进行学习以得到高铁调度在线决策所需的深度Q网络. 源域问题涉及徐州东站到兰考南站之间7站8区间9：00～11：40之间下行5辆列车的晚点案例进行经验学习.

源域经验学习共采用30个具有不同初始晚点的调度案例,按照2：1的比例分为20个训练案例和10个验证案例. 训练案例包括10个发车晚点和10个到站晚点场景,验证案例包括5个发车晚点和5个到站

晚点场景. 所有案例中最小初始晚点为 10 min, 最大晚点为 45 min. 源域经验学习时, 每个训练回合随机从训练案例中选取一个晚点场景初始化交互环境, 通过实验确定深度 Q 网络的输入层节点 9 个, 隐藏层 1 为 32 个节点, 隐藏层 2 为 64 个节点, 输出层 130 个节点. DQN-PER 算法的学习率 $\alpha = 0.1$、奖励的折扣率 $\gamma = 0.86$、优先级指数 $\sigma = 0.96$, 传统 DQN 算法的学习率 $\alpha = 0.1$、奖励的折扣率 $\gamma = 0.86$, 两个算法的最大学习回合数 $N = 10\,000$、经验池的容量 $C = 2\,000$、目标网络的更新间隔 $K = 100$、每次学习抽取的样本数量 $B = 32$、随机探索概率 $\varepsilon = \dfrac{1}{1+\mathrm{e}^{\frac{10\times(n-0.5\times N)}{N}}}$、状态向量中向前向后各分别考虑的资源数量 $k = 2$、奖励的增益因子 $\mu = 500$、权重参数 $w_1/w_2 = 10$. 按照所提出源域经验学习算法对深度 Q 网络进行训练, 10 000 回合后完成学习. 学习过程的收敛曲线和性能对比如图 6 和图 7 所示.



(a) DQN 算法学习过程的收敛曲线



(b) DQN-PER 算法学习过程的收敛曲线

图 6　学习过程验证案例的调度结果曲线

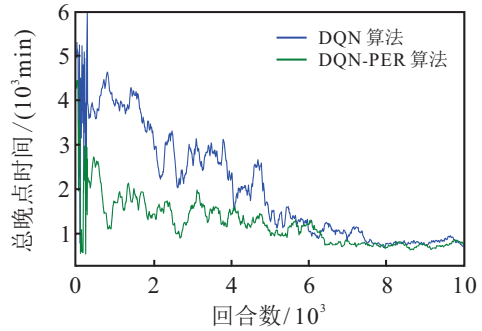


图 7　学习过程验证案例平均总晚点时间的收敛曲线

图 6(a) 和图 6(b) 分别为传统 DQN 算法和所提出 DQN-PER 算法的收敛曲线. 在训练的每一个回合结束后, 为更准确地衡量所获得的深度 Q 网络性能, 利用该深度 Q 网络对 10 个验证案例进行调度决策, 对所生成的阶段调整计划的总晚点时间进行评估, 获得的 10 条收敛曲线如图 6 所示. 值得指出的是, 该过程仅利用不探索. 对上述 10 个验证案例的收敛曲线取统计平均后如图 7 所示. 在图 7 中, DQN-PER 算法的收敛曲线始终位于 DQN 算法的下方, 表明经过相同的学习回合数后, DQN-PER 算法在验证案例的总晚点时间比传统 DQN 算法更小, 即具有更高的经验利用效率. 且 DQN-PER 算法在 6 411 回合处即接近于收敛值 700 min, 而 DQN 算法在 7 646 回合处才开始趋近于该收敛值, 前者比后者的收敛速度提高了约 16 %, 表明 DQN-PER 算法能够在有限资源的情况下实现快速收敛. 两者的最终收敛结果十分接近, 表明在足够的学习回合数下, 二者都可以收敛到具有相近调度能力的深度 Q 网络. 综上所述, 所提出算法比传统的 DQN 算法具有更高的经验利用效率和算法收敛速度, 能够更充分地利用之前的经验和学习结果, 从而避免重复努力和不必要的探索, 实现在资源有限情况下的快速收敛, 有利于高铁调度问题的解决.

考虑到所提出算法的优先级指数 σ 能够反映学习经验优先级 p_c 对于该经验被选中概率的影响程度, 是优先经验回放方法的主要参数之一, 进一步分析了该参数变化对收敛效果的影响, 设置 5 组不同的优先级指数, 分别为 0.5、0.6、0.7、0.8 和 0.96, 结果见图 8.

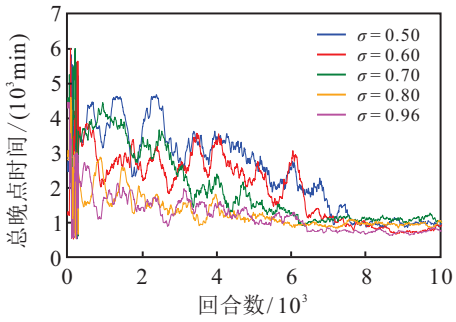


图 8　不同优先级指数对收敛效果的影响

图 8 中, 无论是收敛速度还是收敛结果, $\sigma = 0.96$ 均是 5 组参数中结果最好的, 并且从 $\sigma = 0.5$ 到 $\sigma = 0.96$ 收敛效果的整体表现是逐步提升的, 表明所提出算法能够通过调节优先级指数实现对重要程度较高学习经验的频繁学习, 提高经验的利用效率, 使算法能够更充分地利用之前的经验和学习结果, 从而避免重复努力和不必要的探索, 实现在有限资源情况下的快速收敛. 值得指出的是, 在 $\sigma = 0.8$ 和 $\sigma = 0.96$ 中,

效果没有提升很多,这表明优先级指数达到一定阈值之后,进一步增加优先级指数对算法的性能改善影响不大,需要通过实验确定最佳参数.

此外,本文还对奖励函数中权重参数 w_1 和 w_2 在不同比值下进行对比实验,包括基本权重比值 $w_1/w_2=1$ 以及放大和缩小10倍的3组实验,结果如图9所示. 通过放大权重比值,算法的收敛速度和结果得到了提升,表明奖励函数的放大能够更有效地引导智能体朝向目标靠拢. 然而,进一步增加权重比值对算法性能的改善影响不大,可能存在一定的饱和效应. 因此,需要注意权重比值的选择,以平衡算法性能的提升与计算成本之间的关系. 这些发现对于解决列车调度问题具有重要的指导意义.

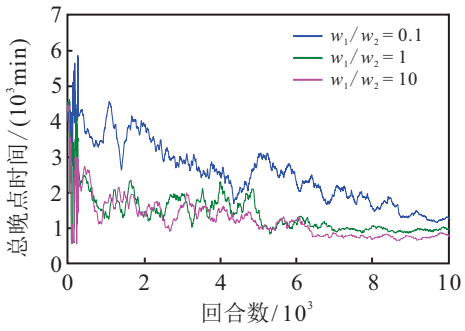


图9 不同权重参数对收敛效果的影响

4.2 目标域复杂场景在线决策

目标域问题为更加复杂的京沪(北京-上海)高铁济南西站到上海虹桥站之间18站17区间9 : 00 ~ 16 : 00之间下行35辆列车的调度. 目标域问题包含了中国高铁最繁忙的“徐蚌瓶颈”,行车密度高、追踪间隔短、冗余时间小,发生晚点后调度难度大. 所选目标域问题的计划运行图如图10所示.

目标域问题包含35个具有不同初始晚点的案例,其中包括16个单车和19个多车晚点案例. 单车初始晚点最小值为7 min,最大值为50 min,多车晚点包括2辆车、 3辆车和4辆车的晚点案例,平均初始晚点时间最小为12 min,最大为55 min. 在线决策时,对于每个晚点案例,通过参数共享利用经验学习所获得的深度Q网络作为在线决策的深度Q网络,完全根据其中存储的学习经验进行决策. 实验使用的计算机配置为: 主机型号DELL precision 7920,处理器Intel(R) Xeon(R) Silver 4210 CPU@2.20 GHz 2.19 GHz,内存16 GB,显卡NVIDIA Quadro RTX 4000. 程序设计为: DQN-PER算法基于python3.8设计,MIP算法采用Gurobi 9.1.1求解.

图11和图12为其中一个多车晚点案例下所提出DQN-PER算法和传统数学规划方法MIP算法所获得的调度方案. 所选多车晚点案例的初始晚点为11 : 30列车G173在曲阜东站晚发14 min、11 : 46列车G109在蚌埠南站晚到23 min和11 : 55列车G263在泰安站晚到42 min. MIP算法调度方案的总晚点时间为1 872 min,用时164.66 s; DQN-PER算法在线决策调度方案的总晚点时间为1 979 min,对应的实时决策用时32.05 s. MIP调度方案和DQN-PER调度方案的晚点恢复时间均为161 min. 图中虚线是计划运行图,实线是实绩运行图,完全重合表明经过调度后列车恢复计划运行图.

对比图11和图12可见,在DQN-PER调度方案和MIP调度方案中,对于G173,两个算法均在滕州东站通过合理安排越行,让列车G887适当延迟3 min发车,避让G173,并安排G887利用曲阜东站到徐州东站之间冗余时间赶点,从而恢复正点. 对于G263晚点引起的G889的晚到,图11中MIP算法为了追求总

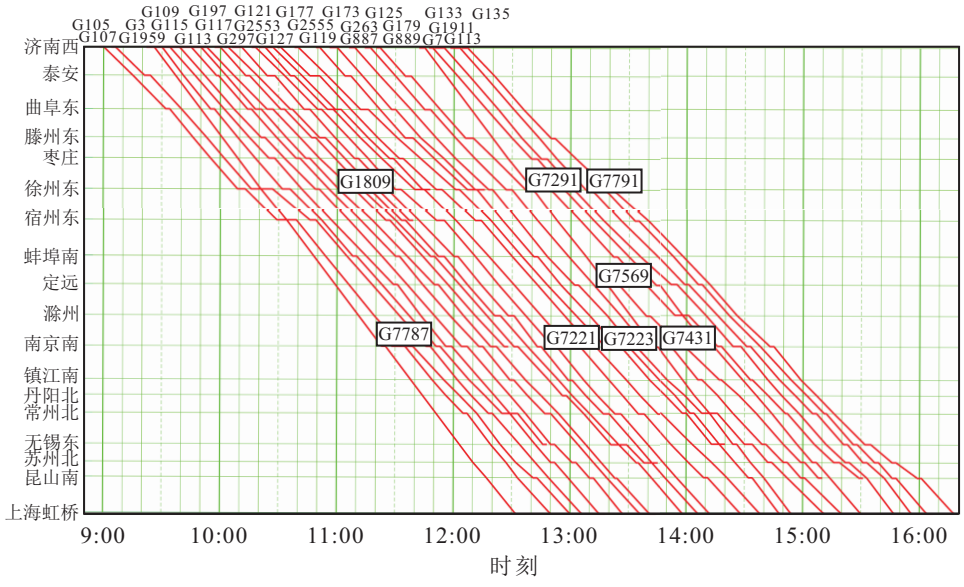


图10 济南西-上海虹桥的计划运行图

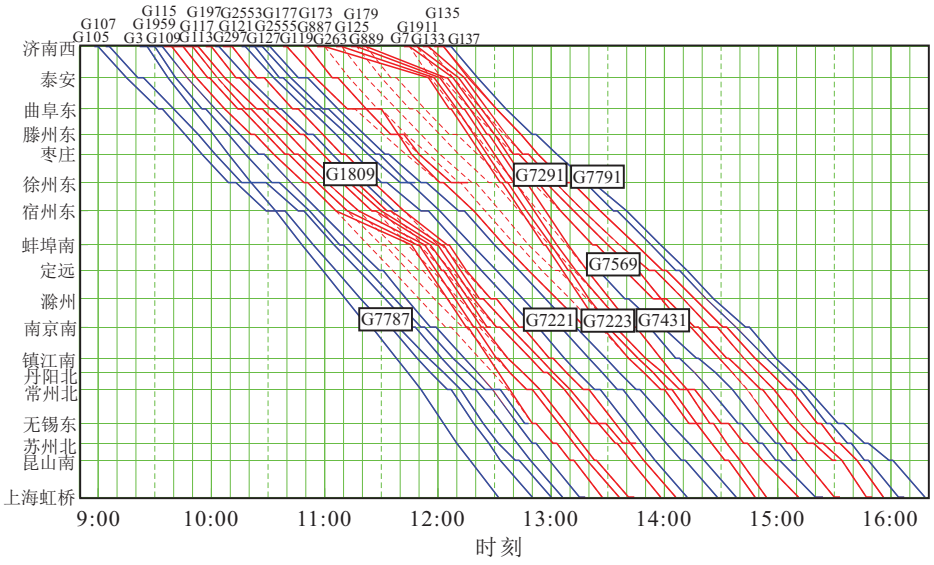


图 11　MIP 算法的调度方案

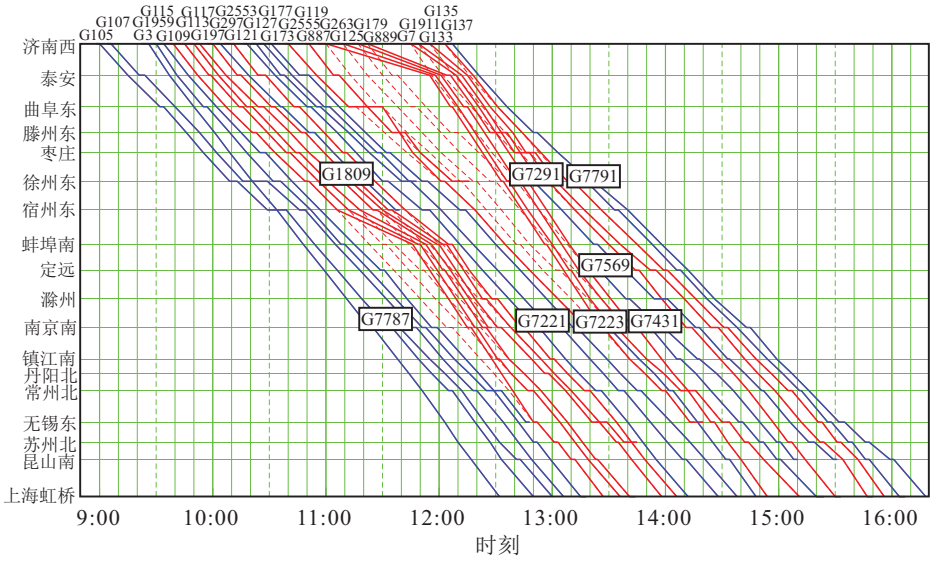


图 12　DQN-PER 算法的调度方案

晚点时间最小, 安排 G 889 尽量赶点运行, 造成了 G 7291 在徐州东站从徐兰线转入京沪线时晚点, 随后通过徐州东站到宿州东站赶点恢复其正点运行. 图 12 中, DQN-PER 算法安排 G 889 在滕州东站延长停车时间 5 min, 安排 G 7 和 G 1911 越行, 虽然 G889 在徐州东晚点时间比 MIP 调度方案增加了 5 min, 但避免了对正点列车 G 7291 的影响. 可见 MIP 算法以追求总晚点时间最小为目标, 采用了激进的“正点列车晚点＋利用冗余时间赶点”策略, 利用一切列车的冗余时间赶点, 达到总晚点时间最小, 但会造成更多正点列车产生延误, 晚点列车数量更多, 所生成调度方案中几乎未保留冗余时间.

而在利用冗余时间赶点和晚点列车数量上, DQN-PER 算法所学习到的经验是根据状态向量在车站停车和区间运行作业中适当保留部分冗余时间, 在尽量减少总晚点时间的同时提高了调度方案的鲁棒性, 即当新增突发事件导致晚点时, 仍有部分冗余时间可用于消除新增晚点, 避免重调度. 这可能是由于 DQN-PER 算法从大量的案例中学习到了如何避免重调度的隐式约束, 取多种案例下的折中调度方案, 虽然造成了总晚点时间的增加, 但是这种保留部分冗余时间的方案的鲁棒性更好, 发生新增晚点时调整次数更少, 调度员工作强度更低.

图 13 为图 11 和图 12 的重调度方案在每个车站所有列车晚点时间之和的变化图, 五角星所示为突发事件导致的 3 辆列车发生的初始晚点. 由图 13 可见, DQN-PER 调度方案和 MIP 调度方案的晚点分布十分接近. MIP 调度方案的晚点时间略小于 DQN-PER 调度方案, 二者都在丹阳北站的所有列车晚点时间之和小于 5 min. 综上, 所提出算法在复杂场景下的迁移应用具有较好的效果, 并且可以考虑到高铁客运系统中的隐式约束, 具有实际应用的价值.

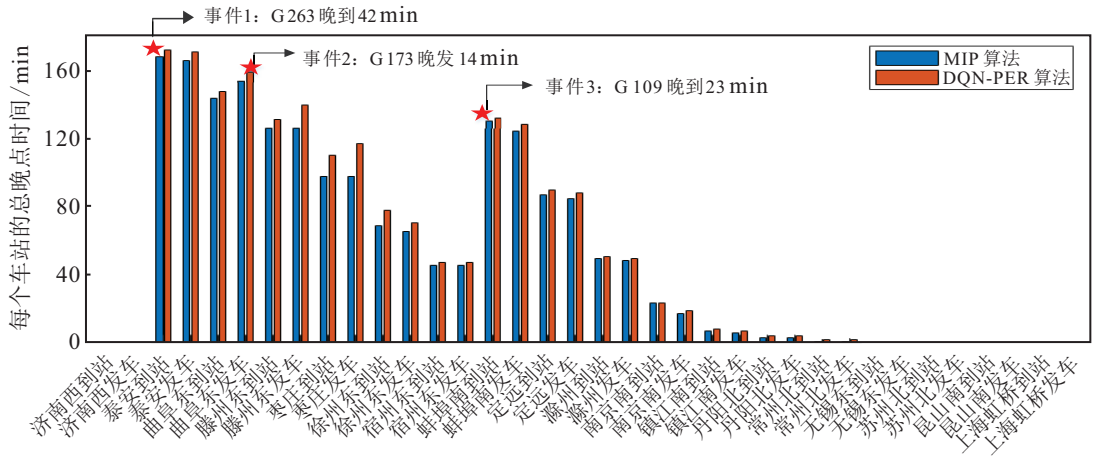
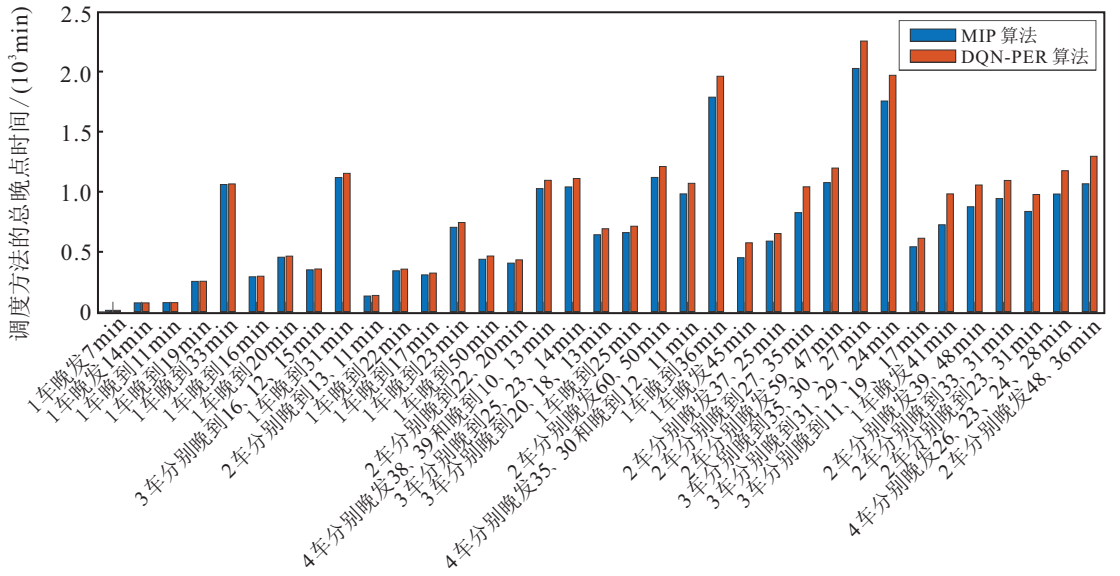
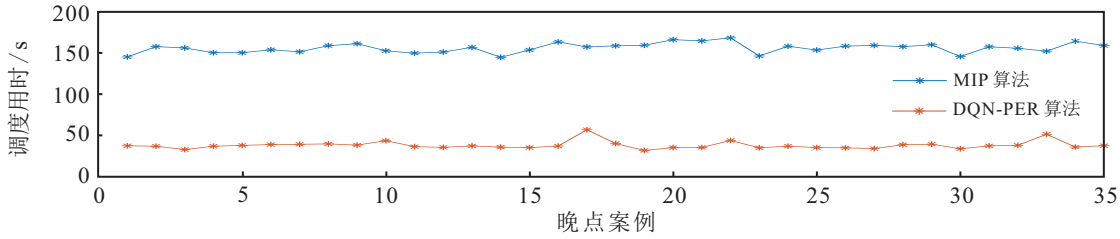


图 13　每个站所有列车晚点时间之和的变化图



(a)　晚点案例调度方案的总晚点时间对比图



(b)　晚点案例的调度用时对比曲线

图 14　测试案例的调度结果对比

将 DQN-PER 算法在线决策与 MIP 算法对测试集全部 35 个案例所得的调度方案总晚点时间和决策所需计算用时进行对比,结果如图 14 所示.

图 14(a) 为 35 个测试案例下所提出 DQN-PER 算法生成的调度方案总晚点时间(橙色)与 MIP 算法调度方案总晚点时间(蓝色)的对比图. 可见,DQN-PER 算法在具有更复杂场景的目标域问题上与 MIP 算法的目标函数值接近,但是存在个别案例具有较明显的差距. 总体来看,在约 77 % 的案例下 DQN-PER 算法与 MIP 算法调度方案总晚点时间的差异小于 15 %,表明所提出算法通过相对简单的源域问题的学习以较小的代价学习到了源域和目标域问题的共性,当通过参数共享方式迁移到目标域问题时,可以获得较好的调度效果. 但是在个别场景下所获得的调度方案比理论上 MIP 算法所获得的最优解还有较大差异,表明对于目标域上少数的特定场景,迁移求解能力还有不足,还需进一步研究,可以采取邻域搜索、参数微调等方式进一步提高迁移求解能力. 对于没有学习到的场景,会存在不佳的调度方案.

图 14(b) 比较了两个算法的调度决策计算耗时,

在 18 站 17 区间 35 车的目标域 35 个晚点运行场景下, MIP 算法的调度决策耗时 140～170 s, 始终远高于所提出 DQN-PER 算法的 30～60 s, 二者的调度用时均比较稳定. DQN-PER 算法的调度决策用时比 MIP 算法平均减少了约 75 %, 这表明对于复杂场景下的高铁调度问题, 采用所提出基于优先经验回放可迁移深度强化学习的高铁调度方法, 计算耗时较小, 能更好地满足高铁调度实时性的要求.

5　结　论

针对强化学习算法在求解高铁调度问题时存在的经验利用效率低、收敛速度较慢和迁移能力差等问题, 本文基于优先经验回放机制, 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. 所设计的面向列车运行的状态向量, 可以使得调度决策方法与高铁调度问题规模(车站和列车数量等)和问题参数(计划时刻表运行时间和路网布局)解耦, 提高了算法的可迁移能力. 所设计的面向列车决策的动作空间, 通过减小其空间规模的大小和决策链的长度来提高学习效率. 所设计的融合优先经验回放的深度 Q 网络训练方法, 能够根据经验的重要程度进行学习, 进一步提高了算法的学习效率和收敛速度. 源域经验学习的实验结果表明, 所提出算法比传统 DQN 算法具有更高的经验利用效率和算法收敛速度, 并可通过适当增大优先级指数和调节权重因子等方式改善算法收敛效果. 以京沪线晚点案例为目标域问题的在线决策实验表明, 所提出算法的调度结果十分接近 MIP 算法, 在约 77 % 的情况下总晚点时间的性能损失小 15 %, 并且调度耗时平均减少了约 75 %. 此外, 所提出算法的迁移能力还可以进一步提高, 下一步拟通过多线路集成学习、目标域邻域搜索和参数微调等进一步提高算法的迁移能力.

参考文献(References)

[1] Sato K, Tamura K, Tomii N. A MIP-based timetable rescheduling formulation and algorithm minimizing further inconvenience to passengers[J]. Journal of Rail Transport Planning & Management, 2013, 3(3): 38-53.

[2] Shafia M A, Aghaee M P, Sadjadi S J, et al. Robust train timetabling problem: Mathematical model and branch and bound algorithm[J]. IEEE Transactions on Intelligent Transportation Systems, 2011, 13(1): 307-317.

[3] 廖正文, 苗建瑞, 孟令云, 等. 基于拉格朗日松弛的双线铁路列车运行图优化算法[J]. 铁道学报, 2016, 38(9): 1-8.
(Liao Z W, Miao J R, Meng L Y, et al. An optimization algorithm for double-track railway train timetabling based on Lagrangian relaxation[J]. Journal of the China Railway Society, 2016, 38(9): 1-8.)

[4] 周妍, 周磊山. 高速铁路行车调度指挥一体化仿真实验平台设计与研究[J]. 铁道学报, 2012, 34(6): 1-7.
(Zhou Y, Zhou L S. Study on simulation and experiment platform of integrated high-speed railway traffic control[J]. Journal of the China Railway Society, 2012, 34(6): 1-7.)

[5] 杨鹏鑫, 崔东亮, 代学武, 等. 基于运控和调度协同仿真的列车阶段调整计划评估[J]. 计算机集成制造系统, 2022, 28(11): 3454-3463.
(Yang P X, Cui D L, Dai X W, et al. Joint scheduling and train operation simulator for rescheduled time table evaluation[J]. Computer Integrated Manufacturing Systems, 2022, 28(11): 3454-3463.)

[6] Samà M, Pellegrini P, D'Ariano A, et al. Ant colony optimization for the real-time train routing selection problem[J]. Transportation Research — Part B: Methodological, 2016, 85: 89-108.

[7] 刘辉, 代学武, 崔东亮, 等. 基于参数自适应蚁群算法的高速列车行车调度优化[J]. 控制与决策, 2021, 36(7): 1581-1591.
(Liu H, Dai X W, Cui D L, et al. Optimization of high-speed train operation scheduling based on parameter adaptive improved ant colony algorithm[J]. Control and Decision, 2021, 36(7): 1581-1591.)

[8] 贾传峻, 胡思继, 杨宇栋. 列车运行调整微粒群算法研究[J]. 铁道学报, 2006, 28(3): 6-11.
(Jia C J, Hu S J, Yang Y D. Study on the particle swarm optimization algorithm for train operation adjustment[J]. Journal of the China Railway Society, 2006, 28(3): 6-11.)

[9] 林博, 俞胜平, 刘子源, 等. 基于改进粒子群算法的高铁列车动态调度[J]. 控制工程, 2021, 28(7): 1334-1341.
(Lin B, Yu S P, Liu Z Y, et al. High-speed train dynamic scheduling method based on improved particle swarm optimization algorithm[J]. Control Engineering of China, 2021, 28(7): 1334-1341.)

[10] 段少楠, 戴胜华. 离散萤火虫算法在高速列车运行调整中的应用[J]. 计算机工程与应用, 2018, 54(15): 209-213.
(Duan S N, Dai S H. Application of discrete firefly algorithm in high-speed train operation adjustment[J]. Computer Engineering and Applications, 2018, 54(15): 209-213.)

[11] 李茹杨, 彭慧民, 李仁刚, 等. 强化学习算法与应用综述[J]. 计算机系统应用, 2020, 29(12): 13-25.
(Li R Y, Peng H M, Li R G, et al. Overview on algorithms and applications for reinforcement learning[J]. Computer Systems & Applications, 2020, 29(12): 13-25.)

[12] 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.

[13] 韩忻辰, 俞胜平, 袁志明, 等. 基于Q-learning的高速铁路列车动态调度方法[J]. 控制理论与应用, 2021, 38(10): 1511-1521.
(Han X C, Yu S P, Yuan Z M, et al. High-speed railway dynamic scheduling based on Q-learning method[J]. Control Theory & Applications, 2021, 38(10): 1511-1521.)

[14] 代学武, 程丽娟, 崔东亮, 等. 基于强化学习的高速列车群运行调整方法[J]. 中国科学: 信息科学, 2022, 52(5): 890-906.
(Dai X W, Cheng L J, Cui D L, et al. Rescheduling of high-speed trains: A reinforcement learning approach[J]. Scientia Sinica: Informationis, 2022, 52(5): 890-906.)

[15] 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.)

[16] Ning L B, Li Y D, Zhou M, et al. A deep reinforcement learning approach to high-speed train timetable rescheduling under disturbances[C]. IEEE Intelligent Transportation Systems Conference. Auckland, 2019: 3469-3474.

[17] Khadilkar H. A scalable reinforcement learning algorithm for scheduling railway lines[J]. IEEE Transactions on Intelligent Transportation Systems, 2019, 20(2): 727-736.

[18] 王云鹏, 郭戈. 城市交叉口车辆速度与交通信号协同优化控制[J]. 控制与决策, 2019, 34(11): 2397-2406.
(Wang Y P, Guo G. Joint optimization of vehicle speed and traffic signals at a signalized intersection[J]. Control and Decision, 2019, 34(11): 2397-2406.)

[19] Schaul T, Quan J, Antonoglou I, et al. Prioritized experience replay[J/OL]. 2015, arXiv: 1511.05952.

[20] 董豪, 杨静, 李少波, 等. 基于深度强化学习的机器人运动控制研究进展[J]. 控制与决策, 2022, 37(2): 278-292.
(Dong H, Yang J, Li S B, et al. Research progress of robot motion control based on deep reinforcement learning[J]. Control and Decision, 2022, 37(2): 278-292.)

作者简介

代学武(1976−), 男, 教授, 博士生导师, 从事动态系统鲁棒状态估计、无线传感测量与控制等研究, E-mail: daixuewu@mail.neu.edu.cn;

吴越(1996−), 男, 硕士生, 从事高铁调度优化的研究, E-mail: 2506491179@qq.com;

石琦(1998−), 男, 硕士生, 从事高铁调度优化的研究, E-mail: 1394844735@qq.com;

崔东亮(1976−), 男, 讲师，博士, 从事高铁智能调度的研究, E-mail: cuidongliang@mail.neu.edu.cn;

俞胜平(1976−), 男, 副教授，博士, 从事列车调度指挥及优化控制等研究, E-mail: spyu@mail.neu.edu.cn.

特邀专家 代学武, 教授, 博士生导师. 2016年获国家特聘专家(青年计划), 加入东北大学流程工业综合自动化国家重点实验室. 主要研究多智能体系统的网络化感知、调度控制一体化, 包括无线网络控制系统的鲁棒状态估计和状态监测、任务关键型无线网络的时间同步、工业物联网设备智能管控、网络系统的调度控制一体化、高速列车群动态调度的机器学习智能求解等. 任《控制工程》副主编、自动化学会大数据专委会委员、《IEEE自动化科学与工程汇刊(TASE)》客座编辑等, 获英国EPSRC Visiting Research Fellow、IEEE ICIRT 2018最佳论文、IEEE CACA 2020最佳理论论文等.

专家寄语 百年韶华多峥嵘, 万里前程更辉煌. 值此百年华诞之际, 回看个人科研道路, 离不开东北大学的培育和柴天佑院士等大师的引领; 展望学校未来发展, 愿东北大学桃李更芬芳, 薪火相传点亮壮丽新百年, 继往开来续写绚丽新华章.