控制与决策
Control and Decision

多智能体深度强化学习及其可扩展性与可迁移性研究综述

闫超, 相晓嘉, 徐昕, 王菖, 周晗, 沈林成

引用本文:
闫超,相晓嘉,徐昕,王菖,周晗,沈林成. 多智能体深度强化学习及其可扩展性与可迁移性研究综述[J]. 控制与决策, 2022, 37(12): 3083−3102.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2022.0044

多智能体深度强化学习及其可扩展性与可迁移性研究综述

闫 超, 相晓嘉†, 徐 昕, 王 菖, 周 晗, 沈林成

(国防科技大学 智能科学学院，长沙 410073)

摘 要: 得益于深度学习强大的特征表达能力和强化学习有效的策略学习能力,深度强化学习在一系列复杂序贯决策问题中取得了令人瞩目的成就. 伴随着深度强化学习在诸多单智能体任务中的成功应用,其在多智能体系统中的研究方兴未艾. 近年来,多智能体深度强化学习在人工智能领域备受关注,可扩展与可迁移性已成为其中的核心研究点之一. 鉴于此,首先阐释深度强化学习的发展脉络和典型算法,介绍多智能体深度强化学习的3种学习范式,分析两类多智能体强化学习的典型算法,即分解值函数方法和中心化值函数方法;然后归纳注意力机制、图神经网络等6类具有可扩展性的多智能体深度强化学习模型,梳理迁移学习和课程学习在多智能体深度强化学习可迁移性方向的研究进展;最后讨论多智能体深度强化学习的应用前景与研究方向,为未来多智能体深度强化学习的进一步发展提供可借鉴的参考.

关键词: 深度强化学习；多智能体系统；迁移学习；课程学习；可扩展性；可迁移性

中图分类号: TP183 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2022.0044

引用格式: 闫超, 相晓嘉, 徐昕, 等. 多智能体深度强化学习及其可扩展性与可迁移性研究综述 [J]. 控制与决策, 2022, 37(12): 3083-3102.

A survey on scalability and transferability of multi-agent deep reinforcement learning

YAN Chao, XIANG Xiao-jia†, XU Xin, WANG Chang, ZHOU Han, SHEN Lin-cheng

(College of Intelligence Science and Technology，National University of Defense Technology，Changsha 410073，China)

Abstract: Due to the powerful feature representation capability of deep learning and the effective policy learning capability of reinforcement learning (RL), deep reinforcement learning (DRL) has made remarkable achievements in a series of complex sequential decision-making problems. With the popularity of DRL in many single-agent tasks, its application in multi-agent systems is flourishing. Recently, multi-agent deep reinforcement learning (MADRL) has attracted increasing attention in the field of artificial intelligence, and the scalability and transferability have become one of the important issues. This paper first describes the development process and typical algorithms of DRL. Then, three types of learning paradigms of MADRL are introduced, and two typical classes of cooperative MADRL algorithms are analyzed, i.e., the value function decomposition approach and the centralized value function approach. In addition, we summarize six types of scalable MADRL models such as attention mechanisms and graph neural networks, and investigate the research progress of transfer learning and curriculum learning in the transferability of MADRL. Finally, we discuss the application prospects and research directions of MADRL, providing some insights for the further development of MADRL in the future.

Keywords: deep reinforcement learning；multi-agent system；transfer learning；curriculum learning；scalability；transferability

0 引 言

作为与监督学习、无监督学习并列的三大机器学习范式之一,强化学习(reinforcement learning, RL)主要用于解决序贯决策问题[1-2],其核心思路是智能体(agent)在与环境的持续交互中进行试错学习,即根据环境反馈(回报)学习最佳策略,使得智能体从环境中获取的累积回报达到最大[1]. 与传统机器学习算法一样,如何有效提取高质量的特征表示是经典强化学

收稿日期: 2022-01-06；**录用日期:** 2022-05-17.

基金项目: 科技创新2030-“新一代人工智能”重大项目(2020AAA0108200)；国家自然科学基金项目(61825305, 61906203, 61803377)；湖南省研究生科研创新项目(CX20210001).

责任编委: 贾英民.

†通讯作者. E-mail: xiangxiaojia@nudt.edu.cn.

习算法所面临的主要难题,“维度灾难”问题始终笼罩在强化学习的天空之上.

近年来,得益于数据规模的迅猛增长、计算能力的大幅提升和算法模型的持续演进, 深度学习(deep learning, DL)技术在图像检测、信号分析、自然语言处理、 序列预测等诸多领域取得了令人瞩目的成就[3]. 随着深度学习技术如火如荼的发展,其在表征学习(representation learning)方向上的进展为突破传统强化学习的局限带来了希望的曙光.

将深度学习的表征优势与强化学习的决策能力相结合[4-5],深度强化学习(deep reinforcement learning, DRL)技术应运而生,一扫传统强化学习领域之阴霾. 深度强化学习能够有效解决高维或连续状态空间和动作空间中的复杂实际问题,在棋类博弈、电子竞技、路径规划、跟随控制、自动驾驶等领域取得了巨大的成功[6]. 目前,该技术已成为机器学习乃至人工智能领域(artificial intelligence, AI)新兴的研究热点.

深度强化学习在单智能体任务中取得的巨大突破点燃了研究人员将其应用于多智能体系统(multi-agent system, MAS)中的热情. 将深度强化学习的思想和算法与多智能体系统相结合,催生了空前火热的多智能体深度强化学习(multi-agent deep reinforcement learning, MADRL). 经过数年的发展,作为解决复杂环境下决策控制问题的重要技术途径之一,多智能体深度强化学习技术已在博弈对抗、机器人避障、无人机编队、交通信号管理等诸多领域得到了成功的应用[7],正逐渐成为研究多智能体系统中群体智能涌现的关键方法.

目前,已有国内学者发布了有关多智能体深度强化学习的综述性研究报告,但其综述角度及内容安排均与本文有较大不同. 梁星星等[8]从设计与实践的角度出发,总结了多智能体深度强化学习的研究进展;孙彧等[9]分类阐述了多智能体深度强化学习经典算法的优缺点;孙长银等[7]主要着眼于多智能体深度强化学习的关键科学问题,分析了该领域的研究现状和发展趋势;殷昌盛等[10]基于分层的角度阐述了多智能体分层强化学习算法的研究现状和典型应用;王军等[11]从博弈论的角度梳理了近年来出现的多智能体强化学习算法,总结了博弈强化学习算法的重、难点与发展方向. 然而,随着深度强化学习在多智能体系统中研究范畴和应用领域的进一步深化和拓宽,多智能体深度强化学习近期在可扩展性与可迁移性方面取得了新的进展. 因此,本文在对(多智能体)深度强化学习的基本原理和研究现状进行分析与梳理后,围绕多智能体深度强化学习的可扩展性与可迁移性对该领域进行归纳总结[12].

论文的整体架构与主要内容如图1所示. 首先阐述两类深度强化学习的发展脉络和典型算法;然后介绍多智能体深度强化学习的3种学习范式,分析两类协作多智能体强化学习算法的主要原理和研究进展,并简述面向多智能体强化学习的典型训练平台;接着聚焦多智能体深度强化学习的可扩展性与可迁移性,归纳6类具有可扩展性的多智能体深度强化学习模型,梳理多智能体深度强化学习可迁移性方向的研究进展;最后展望多智能体深度强化学习的应用前景,讨论多智能体深度强化学习的未来发展方向.

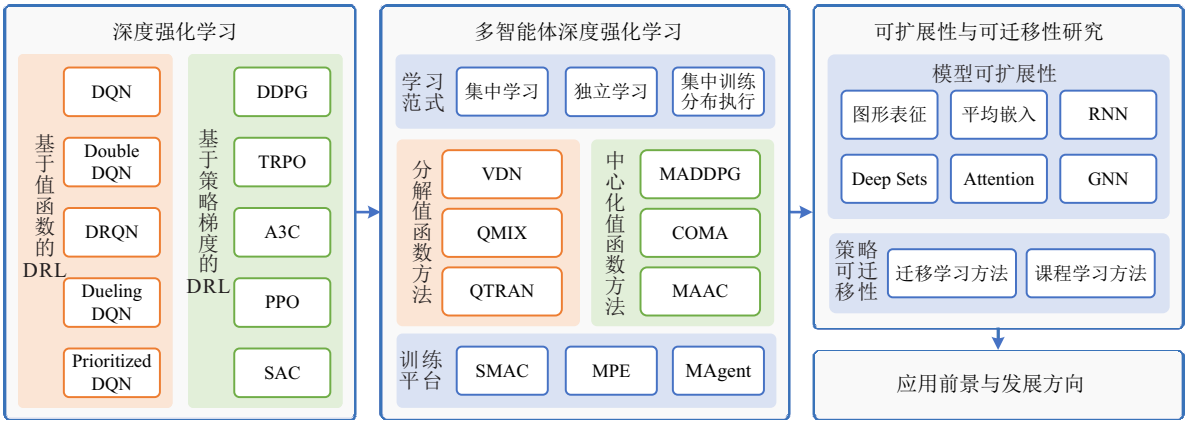

图 1 论文整体架构

1 深度强化学习

深度强化学习结合了强化学习的决策能力和深度学习的表征优势,在诸多领域取得了巨大的成功. 本节简要梳理基于值函数的深度强化学习和基于策略梯度的深度强化学习的典型算法.

1.1 基于值函数的DRL算法

2013年,Mnih等[13]开深度强化学习之先河,将卷积神经网络(convolutional neural network, CNN)和

经验回放技术引入Q学习算法中,首次提出了深度Q网络(deep Q-network, DQN)算法. 该算法直接从原始图像学习控制策略,在3款雅达利(Atari)游戏中的表现超过了人类玩家水平. 2015年,Mnih等[14]采用目标Q网络技术进一步完善DQN模型,相关研究成果发布在Nature杂志上. 改进后的DQN模型模拟人类玩家进行游戏的过程,直接将游戏界面原始像素图像作为输入,实现了端到端的控制,在49款Atari游戏中达到并超越了人类玩家的水平. 相较于Q学习算法,DQN算法主要有如下几点改进: DQN使用CNN作为函数逼近器,同时使用经验回放技术提高训练效率. 此外,DQN还设置单独的目标网络产生目标Q值,以提高算法的稳定性[14].

DQN算法的出世炸响了深度强化学习领域的第一声春雷,自此之后,双重DQN[15]、DRQN[16]、竞争架构[17]、优先经验回放[18]等新算法或技术相继出现以优化DQN算法的性能. 为解决DQN算法中存在的Q值过估计问题,Van Hasselt等[15]在双重Q学习[19]基础上提出双重DQN算法(double DQN, DDQN). 该算法使用两个不同的值函数(两套参数)解耦动作选择与策略评估,更加精确地估计出Q值,进而获得更稳定有效的策略. 为解决部分可观性问题,Hausknecht等[16]利用长短时记忆网络(long short-term memory, LSTM)记忆时间轴上连续的历史状态信息,提出了DRQN算法,减少了网络的计算和存储负担. 从网络模型结构设计的角度,Wang等[17]提出使用竞争架构(dueling architecture)以进一步提升DQN的性能,该架构在卷积网络后构建了两个支路的全连接网络,分别用于逼近状态值函数和动作优势函数(advantage function),最后通过“聚合”操作将二者组合起来以得到每个有效动作的Q值,提高了预测动作价值函数的准确性. 为进一步提高DQN算法的训练效率,Schaul等[18]提出了优先经验回放机制(prioritized experience replay, PER),不同于经验回放的均匀采样,PER根据经验数据的优先级从经验池进行重要性采样,Atari游戏中的实验结果表明,基于PER的DQN算法不仅收敛速度更快,而且表现出更好的性能.

1.2　基于策略梯度的DRL算法

基于值函数的深度强化学习算法在离散动作空间的控制任务中得到了广泛应用,并表现出优越的性能. 但由于价值函数离散型输出的限制,在面对连续动作空间中的强化学习任务时,上述算法往往无能为力. 而连续型控制任务正是基于策略梯度的深度强化学习算法的用武之地.

2015年,Lillicrap等[20]借鉴DQN成功的经验,结合执行器-评价器(actor-critic, AC)框架对确定性梯度策略(deterministic policy gradient, DPG)算法[21]进行拓展,提出了深度确定性策略梯度(deep deterministic policy gradient, DDPG)算法. 该算法能够有效解决高维状态空间和连续动作空间中的强化学习问题,在一系列连续动作空间的任务中表现稳定,且求解效率明显提高.

继DDPG后,TRPO[22]、A3C[23]、PPO[24]、SAC[25]等新型基于策略梯度的深度强化学习算法如雨后春笋般层出不穷,迸发出旺盛的生命力. Schulman等[22]提出了信赖域策略优化算法(trust region policy optimization, TRPO)以解决评价器训练时经常出现的振荡问题,该算法保证了策略优化过程的稳定提升. Mnih等[23]提出了异步优势执行器-评价器算法(asynchronous advantage actor-critic, A3C),该算法基于AC框架,采用多线程操作,基于异步采样的方式

表1　深度强化学习典型算法汇总

分类	典型算法	主要机制	成效及评价
基于值函数的DRL算法	DQN[14]	Q-learning、卷积神经网络、经验回放机制	深度强化学习的开创性工作,实现了端到端控制,在Atari游戏中取得超越人类的成绩
	双重DQN[15]	DQN、目标网络	有效减少了DQN算法对Q值的过高估计,提高了算法的鲁棒性
	DRQN[16]	DQN、循环神经网络	减少了网络的计算和存储负担,有效缓解了实际应用中状态信息部分可观的问题
	竞争架构DQN[17]	DQN、竞争架构	提高了预测动作价值函数的准确性,具有通用性,易于与其他算法或网络结构相结合
	优先经验回放DQN[18]	DQN、优先经验回放机制	大幅提高了学习效率,同时也表现出更好的性能
基于策略梯度的DRL算法	DDPG[20]	DQN、AC框架、DPG	能够有效解决连续动作空间中的强化学习问题,求解效率较高
	TRPO[22]	优势函数、函数近似、步长选择	减少了训练时的波动,保证了策略优化过程的稳定提升
	A3C[23]	优势函数、AC框架、异步算法	提升了算法的训练速度和性能表现,降低了算法对硬件的要求
	PPO[24]	TRPO、剪裁替代目标	稳定可靠,简化了算法的实现过程,提升了求解效率
	SAC[25]	最大熵、AC框架	提升了算法的鲁棒性,避免智能体收敛到次优策略

进行训练,大幅度提升了算法的训练速度和性能表现,同时显著降低了算法对硬件的要求. Schulman等[22]在TRPO算法的基础上提出了近端策略优化(proximal policy optimization, PPO)算法优化策略梯度算法的调参过程,该算法只使用了一阶优化算法,并对策略采用多步更新的算法,在保证算法稳定性和可靠性的同时简化了算法的实现过程并提升了算法性能. Haarnoja等[25]提出了基于最大熵的执行器-评价器(soft actor-critic, SAC)算法,该算法将熵的思想引入目标函数中,通过最大化熵鼓励智能体进行探索,在提升算法的鲁棒性的同时避免了智能体收敛到次优策略.

深度强化学习典型算法汇总如表1所示.

2 多智能体深度强化学习

结合深度学习方法、强化学习方法和多智能体系统而形成的多智能体深度强化学习目前正逐渐发展成为强化学习乃至人工智能领域最为火热的研究方向. 本节首先阐述深度强化学习扩展到多智能体深度强化学习的3种范式,进而将协作多智能体深度强化学习分为分解值函数方法和中心化值函数方法两类,并分析两类算法的主要原理和研究进展,最后简要介绍典型的多智能体训练平台.

2.1 多智能体学习范式

将深度强化学习扩展到多智能体系统中面临诸多挑战. 目前主要的扩展方式可分为集中学习(centralized learning)范式、独立学习(independent learning)范式和集中训练-分布执行(centralized-training decentralized-execution, CTDE)范式3类.

2.1.1 集中学习范式

集中学习范式简单直接地将单智能体强化学习方法推广到多智能体系统中,其核心思想是将所有智能体看作一个整体进行学习. 具体而言,该范式集中所有智能体的状态和动作构成联合状态和联合动作,并直接使用单智能体强化学习算法学习集中式控制策略. 该范式汇合了所有智能体的信息,暗含了智能体之间的沟通协同机制,能够较为容易地获取全局最优解.

与此同时,随着智能体数量的增长,联合状态空间和联合动作空间将呈指数级增长,以至于无法进行探索和训练,故该范式扩展性较差,仅适用于智能体数量较少的场景. 此外,受限于通信条件,在现实场景中智能体也很难获取全局状态. 因此,在多智能体深度强化学习的研究中,一般不使用集中学习范式,而是寻求分布式的方法以避免“维度灾难”问题.

2.1.2 独立学习范式

独立学习范式是单智能体强化学习方法直接推广到多智能体系统中的另一简单思路,其核心思想是假设智能体处于平稳环境中,不考虑智能体之间相互作用,将其他智能体看作环境的一部分,每个智能体直接使用单智能体强化学习算法学习各自的控制策略. 具体而言,在学习过程中,每个智能体独立地更新其各自的策略网络,即每个智能体根据其各自的观测,朝着最大化全局回报的方向优化其各自的策略. 该范式不考虑其他智能体的策略,不需要智能体之间进行协同,适用于离散状态和动作空间中的小规模多智能体问题,具有较强的可扩展性. 例如, Tampuu等[26]遵循独立Q学习(independent Q-learning, IQL)[27]算法的思路,将DQN算法直接应用到雅达利Pong游戏中,取得较为不错的效果. Gupta等[28]将TRPO算法扩展到多智能体场景中,同时引入参数共享机制提出PS-TRPO算法,在多个任务场景中取得了不错的效果.

在智能体训练过程中,其他智能体的策略也在同时变化,这打破了环境平稳性的假设,使得训练的稳定性和收敛性难以保证. 为了解决非稳定环境带来的问题, Castaneda等[29]将DQN算法与重复更新Q学习(repeated update Q-learning, RUQL)算法相结合,提出了深度重复更新Q网络(deep repeated update Q-network, DRUQN)算法,该算法以重复更新动作值的方式避免策略的偏差,重复更新的次数与选择动作的概率成反比. Foerster等[30]通过设计重要性采样方法剔除过时数据和给经验加入额外“指纹”(Fingerprints)信息来“消除抽样数据年龄歧义”两种手段并举,成功实现了经验回放机制在多智能场景中的应用,并在星际争霸任务中取得了较好的效果.

2.1.3 CTDE范式

CTDE范式融合了集中学习范式和独立学习范式的优点,是目前最为常见且典型的多智能体强化学习范式. 顾名思义,CTDE范式在训练阶段允许智能体利用全局信息进行集中学习,在训练结束之后则执行分布式策略. 具体而言,在训练阶段,假设智能体之间的信道不受物理限制,所有智能体能够获取全局信息,故可采用集中学习的方式进行训练;在执行阶段,每个智能体仅通过自身观测和局部信息交互选择动作. CTDE范式结合了集中学习和独立学习各自的优势,近年来基于该范式的多智能体强化学习算法不断涌现,后文第2.2节所介绍算法均属此列[31-45].

表2汇总了多智能体学习的3种范式.

表 2　多智能体学习范式汇总

学习范式	核心思想	存在问题	典型算法
集中学习	将所有智能体看作一个整体进行学习	维度灾难问题	—
独立学习	不考虑智能体之间的相互作用,智能体独自学习各自的策略	环境非平稳性问题	IQL[26]、PS-TPRO[28]、DRUQN[29]、Fingerprints[30] 等
集中训练分布执行	训练阶段利用全局信息集中学习,执行阶段根据局部观测独立选择动作	信用分配问题	VDN[32]、QMIX[33]、QTRAN[34]、MADDPG[41]、COMA[44]、MAAC[45] 等

2.2　协作学习算法研究

基于协作学习的多智能体强化学习算法[31]专注于解决多智能体系统的合作问题. 该类方法并不显式地学习智能体之间的通信协议或通讯消息,而是通过隐式通信的方式实现全局协作,即每个智能体在决策时考虑其他智能体的局部观测和行为策略. 协作多智能体强化学习算法可依据值函数的评估方式分为“分解值函数”方法和“中心化值函数”方法两类.

2.2.1　分解值函数方法

分解值函数方法的核心思想是将智能体从环境中获取的全局奖励按照各自的贡献进行分解,使智能体的环境信息和其他智能体的观测进行分离,进而对策略进行优化. 该方法的主要假设可称为“值函数可分解”假设:每个智能体都拥有自己的“动作值函数”(个体Q函数),而中心化的联合动作值函数(联合Q函数)可以分解为每个智能体个体Q函数的某种组合. 经过分解后的个体Q函数仅与智能体自身的历史状态和动作有关. 该方法不仅能够在一定程度上解决环境非平稳性问题,还能够独立地学习智能体的局部模型以解耦智能体之间的复杂关系.

值分解网络算法(value decomposition network, VDN)[32]是该类方法的开山之作. 该算法在“值函数可分解”假设基础上又做了一个更强的线性假设,即联合Q函数可分解为每个智能体个体Q函数的线性和. 在具体实现时,VDN使用LSTM作为Q网络来近似估计每个智能体的个体Q函数,所有个体Q函数累加得到联合Q函数,进而利用该值函数对智能体策略进行优化. 训练完毕后,每个智能体基于独立的局部环境观测,利用各自的Q网络选择动作,从而实现分布式执行. VDN算法较好地解决了多智能体信度分配问题,不过由于线性假设的存在,VDN动作值函数的表示能力有限,故其在复杂任务中往往表现较差.

QMIX算法[33]是VDN算法的进一步扩展. QMIX放宽了VDN所作的“值函数可线性分解”这一强假设,使用非线性函数将个体Q函数组合为全局Q函数. 但为了保证独立全局最大化操作,需满足联合Q函数与个体Q函数是单调关系这一“单调性”假设约束. 为实现上述目的,QMIX设计了特殊的非线性混合网络(mixing network),并在其中引入超网络(hypernetworks)架构,每个超网络均由线性层与绝对值激活函数构成,以确保混合网络中每层网络的权重(不含偏置)非负. 在具体实现时,QMIX使用门循环单元网络(gate recurrent unit, GRU)作为Q网络近似估计每个智能体的个体Q函数,集中训练时利用全局状态信息和超网络得到值分解所需的非负系数信息,将所有个体Q函数与所得系数加权求和得到联合Q函数,从而对智能体策略进行优化;在分布执行时,智能体只需局部环境观测即可根据各自的个体Q函数独立地选择并执行动作. QMIX放宽了VDN的假设约束,解决了VDN无法捕捉智能体间复杂相互关系的问题,但其“单调性”假设依然严格限制了值函数分解的结构形式,只能解决部分多智能体强化学习问题.

QTRAN算法[34]是在VDN算法和QMIX算法的基础上发展而来的一种更为通用的值分解方法. 受制于VDN和QMIX的强假设约束,二者只能解决一部分多智能体强化学习问题. 对于VDN方法和QMIX无法分解的任务,QTRAN另辟蹊径,将复杂的联合Q函数逼近分解为两步:先将个体Q函数之和作为联合Q函数的直接近似,然后使用状态值网络修正个体Q函数之和与联合Q函数之间的差异. QTRAN具有更加复杂的Q函数网络结构,理论分析表明该方法能够解决更为丰富的多智能体强化学习任务. QTRAN的分解思想进一步扩大了值分解方法的适用范围,有效地提升了值分解方法在多智能体环境中的学习效果,尤其是当环境呈现出明显的非单调性特征时. 尽管QTRAN算法具有很强的理论保证,但其复杂而松散的不等约束使得优化问题难以求解,在复杂任务上的实际表现不甚理想.

以上述算法为基础,分解值函数方法近期又有了新的发展. WQMIX算法[35]将QMIX转化为最优化

问题,通过引入权重的方式解决了QMIX最优动作值函数“欠估计”的问题,弥补了QMIX在解决复杂协作任务时性能不佳的缺陷. Qatten算法[36]从理论角度充分挖掘协作的特点,研究了极值点附近联合Q函数的分解性质,同时引入注意力机制(attention mechanism)[37]逼近联合Q函数的近似分解,为联合Q函数的分解提供了理论支撑. QTRAN++算法[38]进一步增强了QTRAN中的不等约束,降低了QMIX中全局信息的使用,使得其更贴近于现实环境,更好地解决了QTRAN在大规模环境下性能不佳的问题. QPLEX算法[39]针对CTDE训练模式的约束条件进行探索,将CTDE训练模式下的约束条件融入到双重竞争网络(duplex dueling network)中,从而降低学习联合Q函数线性分解结构的难度. QPD算法[40]将累积梯度归因技术运用到MADRL中,沿着轨迹路径直接分解全局Q函数,更好地解决了多智能体信用分配问题.

2.2.2 中心化值函数方法

区别于分解值函数方法,基于中心化值函数的协作多智能体强化学习算法通常以执行器-评价器(actor-critic, AC)框架为基础. 得益于该框架的特殊结构,中心化值函数方法利用全局信息学习共享的全局(中心化)评价器,然后在评价器的指导下,每个智能体独立学习各自的执行器. 该类方法拥有一定抵抗非平稳环境的能力.

MADDPG算法[41]是经典单智能体深度强化学习算法DDPG[20]在多智能体领域的扩展. MADDPG采用CTDE范式,每个智能体都拥有各自的评价器和执行器. 在训练过程中,评价器利用全局环境状态对执行器的更新进行指导;在执行过程中,执行器仅通过各自的局部环境观测进行决策. 此外,MADDPG还引入额外的网络,观测其他智能体的历史动作对其建模,用于推断其他智能体的策略;同时引入集成学习(ensemble learning)的思想,训练多个智能体策略以提高算法的鲁棒性. MADDPG算法有效地解决了环境非平稳问题,在合作型、竞争型、混合型等多种环境中取得了较好的效果. 然而,该算法为每个智能体配置了独立的全局评价器,使得智能体学习周期较长,训练成本高,可扩展性差,仅适用于智能体数量较少的任务场景. 为弥补以上缺陷,MADDPG诞生后陆续出现了一些补充性工作. PS-MADDPG算法[42]引入参数共享(parameter sharing, PS)机制,提高了MADDPG算法的可扩展性,并在学习速度和存储效率方面具有一定的优势. ATT-MADDPG算法[43]利用注意力机制对队友策略进行建模,提高了MADDPG算法的可扩展性和鲁棒性.

COMA算法[44]旨在解决多智能体强化学习任务中的信用分配问题. 有别于MADDPG,该算法中所有智能体共享同一个评价器网络,以全局状态、联合动作向量和联合奖励为输入,对联合动作进行评估;每个智能体都拥有独立的执行器网络,在全局评价器的指导下,智能体基于各自的观测历史更新执行器的网络参数. 为解决多智能体信用分配问题,COMA引入反事实基线(counterfactual baseline)的概念,利用反事实的思想推断每个智能体对整体任务的贡献度. 通过固定其他智能体的行动,使用边际法确定反事实基线,确定每个智能体的信用分配,然后计算评价器网络近似的Q函数与反事实基线之差,得到优势函数,进而对执行器网络进行更新. COMA算法较好地解决了多智能体信度分配问题和环境非平稳性问题,提高了多智能体之间的信息共享率与智能体之间的协作能力,在收敛速度和最终性能上均具有一定优势. 然而,该算法采取完全集中的训练方式,可扩展性较差,集中式训练器容易维度爆炸,仅适用于各智能体奖励相同的合作型任务.

MAAC算法[45]是MADDPG算法和COMA算法的继承和发展. 该算法在MADDPG的基础上,将DDPG算法[20]替换为更为先进的SAC算法[25],同时引入COMA算法中的反事实基准以解决多智能体信用分配问题. MAAC的核心思想是在评价器网络中引入注意力机制来评估全局Q函数. 与MADDPG同等对待其他智能体不同,MAAC利用多头注意力机制(multiple attention heads)[37]给不同智能体以不同的关注度,充分利用智能体的局部观测以及动作,提取最有效的信息进行学习. MAAC算法较好地解决了环境非平稳性问题,基于注意力机制的评价器有效提高了智能体的学习效率和策略的稳定性,一定程度上解决了MADDPG在智能体数量增加时扩展性不佳的问题,但其在异构多智能体系统上的性能表现有待进一步提升.

表3给出了分解值函数方法与中心化值函数方法的对比分析.

2.3 典型多智能体训练平台

构建训练环境平台是开发测试强化学习算法的基础. 多智能体强化学习尤其需要合适的训练平台,以便研究人员对多智能体问题进行探索. 研究人员针对多智能体强化学习问题开发了多种训练平台,SMAC[46]、MPE[41]和MAgent[47]是其中的典型代表.

表 3　分解值函数方法与中心化值函数方法对比分析

分类	典型算法	核心思想	优点	缺点
分解值函数方法	VDN[32]	按照智能体对联合回报的贡献将联合 Q 函数分解为个体 Q 函数的线性和	较好地解决了多智能体信度分配问题	分解方法单一, 需要满足“值函数可线性分解”强假设约束
	QMIX[33]	使用非线性混合网络将个体 Q 函数拟合为联合 Q 函数	较好地解决了多智能体信度分配问题, 放宽 VDN 的假设约束	分解方法单一, 需要满足“单调性”假设约束
	QTRAN[34]	将个体 Q 函数之和看作联合 Q 函数近似值, 使用状态值网络拟合联合 Q 函数真实值与近似值之差	较好地解决了多智能体信度分配问题, 有理论保证, 进一步扩大了值分解方法的适用范围	约束复杂, 优化困难, 在复杂任务上的实际表现较差
中心化值函数方法	MADDPG[41]	采用 AC 框架和 CTDE 范式, 每个智能体均拥有集中式评价器接受全局信息	较好地解决了环境非平稳性问题, 适用于合作型、竞争型、混合型等多种任务场景	可扩展性差, 训练周期较长, 仅适用于智能体数量较少的场景
	COMA[44]	采取完全集中的训练方式, 引入反事实基线的概念进行信用分配	较好地解决了多智能体信度分配问题和环境非平稳性问题	可扩展性较差, 集中式训练器容易维度爆炸, 仅适用于合作型任务
	MAAC[45]	在集中式评价器中引入多头注意力机制来评估全局 Q 函数	较好地解决了环境非平稳性问题, 提高了学习效率和稳定性, 具有较好的可扩展性	在异构多智能体系统上表现欠佳

《星际争霸》(StarCraft) 是即时战略游戏的代表作, 该游戏玩法丰富多样, 是训练多智能体强化学习的理想环境. DeepMind 在 StarCraft II 机器学习 API 的基础上开发了针对强化学习训练的环境工具包 PySC2. 游戏环境中主要涉及两类操作, 即宏观操作和微观操作. 其中, 微观操作需要对多个实体进行控制决策, 是典型的多智能体问题, 因此可以基于此环境设计训练多智能体强化学习算法. SMAC 平台在 PySC2 的基础上进行了改进, 游戏中每个单元由独立的智能体进行控制, 每个智能体必须根据局部环境观测进行决策. 该平台涉及到多个智能体的动作和观测空间, 为多智能体强化学习算法的训练和测试提供了标准的环境场景.

MPE 环境是一个时间离散、空间连续的二维环境, 该环境涵盖合作型、竞争型、混合型等多种任务场景, 用户可根据需要自定义智能体数量和任务类型, 如围捕、避碰等. MADDPG 算法[41] 首次使用该环境验证了其有效性, 由此该环境声名大噪, 逐渐成为多智能体强化学习算法测试和比较的基准环境.

不同于小规模的 SMAC 和 MPE 平台, MAgent 平台聚焦于大规模智能体的任务和应用. 该平台不仅能够支撑多智能体强化学习问题的研究, 还能够观察多智能体群体中的个体行为, 探究群体行为的涌现机制. MAgent 训练平台资源占用量小、部署灵活, 在单机条件下即可支撑成百上千的智能体, 且具有良好的可拓展性和方便的自定义能力.

3　可扩展性与可迁移性研究

为增强多智能体深度强化学习解决群体智能问题的能力, 可扩展性和可迁移性已成为该领域的核心研究点. 本节首先归纳6种具有可扩展性的多智能体深度强化学习模型, 然后梳理多智能体深度强化学习可迁移性方向的研究进展.

3.1　模型可扩展性研究

在多智能体系统中, 智能体规模的动态变化给深度强化学习算法的应用带来了巨大的挑战. 深度强化学习使用深度神经网络近似最佳策略, 而神经网络通常需要固定维度的状态表示作为输入. 由于群体规模的不确定性, 系统状态表示的维度是动态变化的. 这一矛盾限制了深度强化学习算法的可扩展性, 学习到的最佳策略往往难以适应群体规模的动态变化. 为解决这一矛盾, 一些学者设计了特殊的表征方法用以处理变长输入的问题.

3.1.1　图形表征

在电子游戏领域, 智能体通常以游戏界面原始图像 (全局或局部) 为输入学习控制策略. 无论游戏中元素或实体数量如何变化, 游戏界面截图的维度始终是固定不变的. 因此, 将环境状态表征为图片或类似图片形式的数据成为解决变长输入问题的一种可行手段.

在路径规划任务中, 环境通常用栅格地图进行表示, 这为该类方法在此类任务中的成功应用创造了良

好的条件. Sartoretti等[48-49]将障碍位置、目标位置、邻居位置、智能体位置等环境信息分别表征为局部地图,并以此为输入学习控制策略,用以解决多智能体路径搜索问题. 类似地,Theile等[50]充分提取并利用局部地图和全局地图所蕴含的信息以解决无人机路径规划问题,随后Bayerlein等[51]成功地将其应用于多无人机数据搜集任务中, 展现出良好的扩展能力. 此外,Hüttenrauch等[52]使用联合直方图描述邻近智能体的空间分布特征,并以此为输入学习控制策略使得智能体涌现出复杂的群体行为. 然而,上述方法仅描述了智能体/障碍物的空间分布特征. 为表征更为复杂的智能体特征,Han等[53]基于编码器-解码器模型(encoder-decoder)构建了GridNet,将所有智能体的全部属性信息以栅格地图的方式综合表征,并以全局地图为输入学习集中式协作策略,实现了电子游戏中任意数量智能体的网格粒度控制(grid-wise control). Zhou等[54]提出了一种基于统计图的特征表示方法描述邻近无人机的数量、位置、速度等信息,并以此为输入学习分布式协作策略,有效解决了大规模无人机集群多目标搜索与跟踪任务中的可扩展性问题. Yan等[55]建立威胁评估模型,将邻近无人机的位置、航向、速度信息表征为局部态势图,实现了动态无人机编队的群集控制与避碰.

图形表征法简单直接,能够将任意数量的智能体和障碍物表征为维度固定的特征向量,且具有置换不变性. 然而,该方法需要对状态空间进行离散化,地图精度与计算开销之间存在天然的矛盾,构建精细化地图必然会引起计算时长的增加,故该类方法在智能体/障碍物数量较多、决策实时性要求较高的场景中往往表现较差.

3.1.2 平均嵌入

在处理变长输入问题时,平均嵌入(mean embedding)是一种可行的方法. 无论智能体的数量如何变化,智能体状态表示平均值的维度总是固定不变的. 因此,计算不同智能体状态的平均表示,即可将变长维度的状态编码为固定长度的特征向量.

Yang等[56]借用平均场理论(mean-field theory)的思想给出多智能体系统的一个近似假设:对于某个智能体而言,所有邻近智能体对其产生的相互作用可以用一个均值替代. 换言之,智能体与其邻近智能体之间的交互作用简化为两个智能体之间的交互作用,即该智能体与其所有邻居智能体的均值,这极大地简化了智能体数量增长带来的“维度灾难”问题. Wang等[57]基于平均嵌入提出了求解平均场多智能体强化学习问题的MF-FQI算法,并分析了MF-FQI算法的非渐近收敛特性. 类似地,Sunehag等[58]通过计算捕食者的平均值组合其状态表示. Hüttenrauch等[59]也使用平均嵌入方法解决集群系统中的表示学习问题. 这一状态表示方式具有置换不变性和维度不变性,在解决分布式集群控制问题时具有明显优势. 在此基础之上,Gebhardt等[60]提出了深度M嵌入(deep M-embeddings)方法,进一步丰富了平均嵌入的实现方式与应用案例. Zhang等[61]针对分布式多无人车协同车围捕问题,使用平均嵌入法表征邻居状态,提高了算法的泛化能力.

平均嵌入法以计算均值的方式表征任意数量智能体的状态信息,易于实现,且具有维度和置换不变性. 但是,该类方法忽略了智能体的数量信息和重要程度,只适用于智能体数量较少的场景.

3.1.3 循环神经网络

在自然语言处理(natural language processing, NLP)领域,循环神经网络(recurrent neural network, RNN)能够将任意长度的输入编码为固定长度的特征向量. 借助这一思想,LSTM在多智能体系统领域中得到了广泛的应用. 在实现时,按照距离远近逆序排列,依次将邻近智能体与中心智能体的相对状态输入LSTM,LSTM的隐含层变量即为编码后的固定长度特征向量.

Everett等[62]将LSTM引入策略网络,使得智能体(地面移动机器人)能够适应其他智能体数量的动态变化,从而完成避障任务. 同理,Sui等[63-64]使用LSTM处理任意数量的障碍(地面移动机器人)信息,仿真实验和实物实验验证了算法在编队与避障控制场景中的有效性和实用性. Bai等[65]使用LSTM处理不同数量机器人的编队控制与避障问题. 类似地,Zhang等[66]将这一网络结构扩展到多机器人系统目标围捕中,仿真结果验证了算法的有效性和通用性. Schlichting等[67]基于LSTM设计了空间编码器,能够适应智能体数量的不确定性,实现时变多智能体系统的路径规划,仿真结果及四架旋翼无人机飞行实验验证了算法的可行性.

使用循环神经网络对智能体状态序列进行编码,序列中位置越靠后的智能体对循环神经网络的影响越大. 因此,在构建输入序列时可利用先验信息(如距离越近的邻近智能体越重要)确定智能体的重要性. 然而,该类方法需要使用循环单元进行多次迭代计算,灵活性不高,且策略输出受智能体状态序列的排列影响较大,不适合大规模多智能体系统.

3.1.4 深度集合网络

面向定义在集合上的机器学习任务, Zaheer 等[68] 设计了深度集合 (deep sets) 神经网络架构处理以集合为输入的问题. 该架构具有置换不变性 (permutation invariance), 可以处理不同大小的集合输入, 在点云分类与异常检测[68]、机器人排序[69] 等任务中有着成功的应用.

Huegle 等[70] 将这一架构引入到深度强化学习中, 提出了 DeepSet-Q 和 Set2Set-Q 算法. 仿真结果表明, 所提出算法在主动车道变换场景中表现良好, 能够很好地处理周围车辆数量的动态变化对换道决策过程的影响, 具有较强的泛化能力. 随后, Huegle 等[71] 在 deep sets 的基础上进一步提出了 deep scenes 架构. 该架构能够处理车辆、车道、交通标志等多个对象类型的变长序列, 融合不同特征表示的可变尺寸的输入集合, 提高了深度强化学习在自动驾驶领域的适用性和可扩展性. 与此同时, Shi 等[72] 使用 deep sets 以无索引或置换不变的方式对多无人机交互进行编码, 使得所提出 Neural-Swarm 集群算法能够适应集群数量和编队构型的变化. 在此基础上, Shi 等[73] 针对异构无人机集群规划与控制问题, 进一步提出了 Neural-Swarm2 算法. 该算法设计了异构 deep sets 用于学习任意规模的异构无人机交互, 提高了算法的扩展能力和泛化能力. 此外, Li 等[74] 基于 deep sets 架构实现了其提出的平均场 PPO 算法, 实验结果和理论分析表明, 该算法具有置换不变性, 在大规模智能体场景中具有良好的扩展性.

将多智能体协作学习定义成在集合上的机器学习任务, 使用深度集合网络能够将不同大小的集合输入编码为维度固定的特征向量. 该类方法具有置换不变性, 能够支持异构多智能体系统. 然而, 其将所有智能体同等对待, 忽略了不同智能体的重要程度, 在复杂交互场景中的性能表现有待进一步提升.

3.1.5 注意力机制

近年来, 注意力机制在机器翻译、文本分类等 NLP 任务中大放异彩[37], 受其启发, 已有学者将注意力机制应用到多智能体系统中[75-76]. 不同于平均嵌入方法, 注意力机制能够赋予不同智能体以不同的权重, 计算各智能体状态特征的加权和即可得到维度不变的系统状态表征.

Chen 等[77] 针对密集人群场景下的机器人导航任务设计了基于自注意力 (self-attention) 机制[37] 的社会关注度池化 (social attentive pooling) 模块. 该模块以数据驱动的方式学习每个邻居的相对重要性, 能够将任意维度的输入处理为固定长度的输出. Oroojlooy 等[78] 将注意力机制引入交通信号控制问题, 提出了 AttendLight 模型. 该模型包含两个注意力模块, 能够处理任意数量的道路、车道、车辆和动作集合, 大大提高了算法的通用性能和扩展能力. Liu 等[79] 利用注意力机制对邻近智能体的关系进行建模, 设计了注意力关系型编码器 (attentive relational encoder, ARE) 来聚合任意数量邻近智能体的特征表示, 使得算法能够扩展到大规模智能体上. Iqbal 等[80] 将注意力机制引入 QMIX 算法, 提出了基于随机实体分解 (randomized entity-wise factorization) 的 REFIL 方法处理不同类型、不定数量的实体和智能体. Hsu 等[81] 开发了一种基于注意力机制的随机多智能体强化学习方法, 能够处理任意数量的外部智能体. Batra 等[82] 基于 deep sets 架构和注意力机制分别实现了具有扩展能力的分布式无人机集群控制器, 注意力机制在实验中展现出更强的性能.

作为 Self-Attention 的一种典型结构, Transformer[81] 在多智能体强化学习中也得到了初步的应用. Hu 等[83] 首次提出使用 Transformer 处理动态特征, 设计了通用策略分解 Transformer (universal policy decoupling transformer, UPDeT). UPDeT 将变长观测特征拆分为几组基于实体的特征, 并使用 Transformer 模块生成不同的动作. 该模型打破了传统模型结构输入和输出维度固定的有关的限制, 提高了模型的可扩展性和通用性. Zhang 等[84] 针对临时团队游戏 (ad hoc team play) 问题提出了协作 Q 学习 (collaborative Q-learning, CollaQ) 算法. 该算法堆叠多层注意力模块形成 Transformer 架构处理维度时变的观测信息, 提高了算法的可扩展性和泛化性. Zhou 等[85] 针对多智能体协调知识 (coordination knowledge) 迁移问题, 基于 Transformer 设计了 PIT (population invariant agent with transformer) 网络结构, 提高了算法在不同规模场景下的泛化能力.

基于注意力机制的特征聚合方法能够适应智能体数量的动态变化, 具有置换不变性, 可以赋予不同智能体以不同的权重, 从而区分出智能体的重要程度. 目前, 该类方法在多智能体学习领域得到了广泛应用, 并取得了较好的效果. 然而, 在实际应用中, 智能体之间复杂的交互关系使得各智能体的贡献度难以有效区分, 相关研究仍需进一步深化.

3.1.6 图神经网络

近年来, 随着图神经网络 (graph neural network, GNN)[86-88] 在人工智能领域的异军突起, 其在多智能

体领域的应用也愈发多样. 若将智能体定义为图上的节点,智能体之间的交互关系定义为图上的连接关系,则通过GNN即可针对图上节点间的连接关系进行特征表示的聚合与学习[89].

基于这一思想,Khan等[90]将图卷积网络(graph convolutional network, GCN)[86]应用于大规模无人机编队控制中,显著提高了分布式控制器的扩展能力. Liu等[91]基于GCN设计实现了具有置换不变性的集中式评价器,样本效率和可扩展性较传统评价器有了显著的提升. Wang等[92]使用GNN的“聚合”操作(AGGREGATE operator)[88]处理变长维度的输入,从而使得DRL学习到的知识能够从智能体数量少的简单场景迁移到智能体数量多的复杂场景.

结合注意力机制,GNN能够获得更强的特征表示能力. Naderializadeh等[93]在QMIX算法的基础上引入GNN和注意力机制解决值函数分解和多智能体信用分配问题,改进后的GraphMIX算法提高了QMIX算法的性能和可扩展性. Ryu等[94]在图注意力网络(graph attention network, GAT)[87]的基础上提出了分层GAT模型用来学习多智能体场景中的状态表示. 该模型包含两个注意力网络,即群组间(inter-group)注意力网络和个体间(inter-agent)注意力网络,分别用于学习智能体组间的状态表示和组内智能体之间的状态表示. 仿真结果表明,分层GAT模型提高了策略的可扩展性. 在此基础上,Ye等[95]进一步结合GRU和分层GAT模型,提出了深度循环图网络(deep recurrent graph network)算法,提高了策略的可扩展性和可解释性,实现了部分可观条件下的多无人机导航任务.

将多智能体系统描述为图结构,使用图神经网络进行信息传递与表征学习,能够扩大智能体的通信范围,有助于学习更为有效的协同策略. 同时,图表征具有置换不变性,可应用于大规模的异构多智能体系统. 然而,多智能体系统的拓扑结构动态时变,加大了图表征学习的难度. 此外,该类方法需要智能体之间频繁交换信息,在实际应用时对通信条件的要求较为严苛.

可扩展的多智能体深度强化学习模型汇总分析如表4所示.

表 4 可扩展的多智能体深度强化学习模型汇总分析

模型分类	核心思想	优点	缺点	主要文献
图形表征	将环境状态表征为图片或类似图片形式的数据	简单直接、具有置换不变性	需要对状态空间进行离散化, 地图精度与计算开销之间存在天然矛盾	[48-55]
平均嵌入	将各智能体状态特征的平均值作为系统状态表征	易于实现、具有置换不变性	忽略了智能体的数量信息和重要程度, 只适用于智能体数量较少的场景	[56-61]
循环神经网络	将各智能体的状态特征拼接为序列数据, 使用循环神经网络对输入序列进行编码	能够利用先验信息确定智能体的重要性	需要使用循环单元进行多次迭代计算, 灵活性不高, 不适合大规模多智能体系统	[62-67]
深度集合网络	将多智能体协作定义成在集合上的机器学习任务, 使用深度集合网络处理不同大小的集合输入	具有置换不变性, 支持异构多智能体系统	将所有智能体同等对待, 忽略了不同智能体的重要程度	[70-74,82,96]
注意力机制	基于注意力机制赋予不同智能体以不同的权重, 而后计算各智能体状态特征的加权和	具有置换不变性, 能够自动区分出智能体的重要程度	智能体之间复杂的交互关系使得各智能体的贡献度难以有效区分	[77-85,93-96]
图神经网络	将多智能体系统描述为图结构, 使用图神经网络进行信息传递与表征学习	具有置换不变性, 可应用于大规模/异构多智能体系统	动态图表征学习难度大、通信开销大	[89-96]

3.2 策略可迁移性研究

随着群体智能技术的发展,多智能体系统的规模日益增大,其所面临的任务场景也愈发多样. 庞大的智能体数量和复杂的任务环境使得系统的状态-动作空间维度呈现指数级增长趋势. 因此,直接在复杂场景中的大规模群体上训练控制策略往往难以收

敛. 为解决这一问题,一些学者提出了具有可迁移性的多智能体深度强化学习算法,使得在小规模简单场景中学习的控制策略可以迁移并应用到大规模复杂场景中去,从而加快学习进度并提高模型性能,最终实现多智能体深度强化学习算法在大规模复杂场景中的实际应用.

3.2.1 迁移学习方法

迁移学习(transfer learning, TL)[97]能够将源任务中学习到的知识和经验应用到目标任务中,使得目标任务的训练更为高效. 在强化学习中,迁移学习可以利用从过去相关任务中学到的知识加速DRL在新任务中的学习,从而降低解决新任务的难度. 近几年,迁移学习已被广泛应用于多种深度强化学习场景中. 多智能体深度强化学习中的迁移学习[98-99]主要有任务间迁移和智能体间迁移两个研究方向.

1) 任务间迁移.

任务间迁移是指将在源任务中学到的知识或策略向相似但不相同的目标任务迁移. 比较源任务与目标任务的相似度是一类常见的迁移强化学习方法. Song等[100]设计了度量两个MDP相似度的方法,并依据相似度迁移价值函数. 另一类常见方法是建立源任务与目标任务的映射关系. Gamrian等[101]使用生成对抗网络(generative adversarial network, GAN)创建映射,将目标任务中的图像转换为源任务中的相应图像,解决了传统强化学习方法无法适应游戏界面背景变化的问题.

此外,评估多个源策略在目标任务上的性能,选择合适的源策略进行迁移也是一类行之有效的方法. Li等[102]将策略迁移建模为多臂赌博机模型,在目标任务上比较源策略的性能,选择最高者进行迁移. 随后,Li等[103]将策略迁移建模为选项(option)学习问题,提出了利用选项进行策略迁移的CAPS方法,该方法估计选项的价值,选择价值最高选项所对应的源策略进行迁移. 在此基础上,Yang等[104]开发了利用选项进行策略迁移的PTF框架,该框架为每个源策略更新价值和终止概率,选择价值最高的源策略进行迁移. 为了更为准确地评估源策略在目标任务上的性能,常田等[105]设计了随机集成策略迁移SEPT算法,利用集成学习的思想从源策略库中集成出教师策略进行迁移,而不是简单地选择某一个策略.

尽管任务间迁移学习方法在单智能体场景中得到了广泛的应用,但在多智能体领域相关研究尚不丰富. 早期研究一般通过显示计算多智能体学习任务之间的相似度以实现知识迁移[106-107]. 近年来,Liu等[108]对MDP的相似度进行扩展,提出了根据N步返回值计算相似度的方法,同时设计了一种可扩展的迁移强化学习方法,显著加速了多智能体的学习过程. Qin等[109]通过建模不同任务之间的状态转移及回报函数的相似性捕捉不同任务的共同结构,提出了基于任务关系(task relationship)建模的多智能体迁移强化学习框架MATTAR,有效提升了多智能体的策略迁移能力与泛化能力.

在多智能体迁移学习中,可迁移知识的表征形式可以是经验、策略和特征. 针对两个多智能体环境之间的经验迁移问题,Niu等[110]提出了通过测量环境差异进行经验知识共享的新方法,该方法将相似性定义为奖励预测与真实奖励之差,进而以相似度作为采样权重从源环境经验池中抽取样本进行训练,显著提高了学习效率. 针对多智能体任务间的策略迁移问题,Gao等[111]基于知识蒸馏(knowledge distillation)提出了KnowRU知识复用方法,通过模仿源策略可有效利用智能体的历史经验,不仅缩短了智能体在新任务上的训练时间,而且提高了渐近性能. 针对多智能体学习中的跨任务特征迁移问题, Shi等[112]提出了一种通用的多智能体横向迁移(multi-agent lateral transfer, MALT)算法,该算法引入注意力机制衡量不同策略所传递的特征,并使用横向连接将特征从源任务迁移到目标任务中,大大减轻了多智能体强化学习的训练负担,可实现异构智能体之间的知识迁移.

2) 智能体间迁移.

在多智能体场景中,在智能体之间进行知识共享或迁移可显著提高多智能体学习效率. 目前,已有学者利用教师-学生(teacher-student)框架成功实现了智能体之间的知识迁移. Omidshafiei等[113]针对协作多智能体强化学习中的点对点教学(peer-to-peer teaching)问题,提出了学习协调与教导强化框架(learning to coordinate and teach reinforcement, LeCTR),使得每个智能体学会在合适的时间向其他智能体提出合适的建议,同时利用其他智能体的建议以提升自身性能. 在此基础上,Kim等[114]引入分层强化学习的思想提出分层多智能体教学(hierarchical multi-agent teaching, HMAT)框架,提高了教师信用分配的准确性,解决了LeCTR扩展到复杂环境中表现不佳的问题. 然而,考虑到信息交换时的通信开销,以上两种算法只适用于两个智能体的场景. 为此,Liang等[115]将上述想法扩展到多智能体场景,提出了平行注意迁移(parallel attentional transfer, PAT)知识迁移框架以解决智能体如何有选择地从其他智能体学习

知识这一问题. 该框架设计两种行为模式,即学生模式和自主学习模式,并使用共享注意力机制从其他智能体中选择行为知识以加速学生智能体的学习,提高了团队学习速度和整体性能,具有良好的灵活性和可移植性. 此外,为降低传递知识时的通信开销,Ye等[116]引入“自我指导”(self-advising)的概念,提出了一种基于模型的自我指导多智能体学习方法. 通过self-advising,处于陌生状态中的智能体能够充分利用其在相似状态下获得的经验来生成建议,在提高学习效率的同时显著降低了通信开销.

另有一些研究使用策略蒸馏(policy distillation)方法进行知识迁移. Wadhwania等[117]针对高效协同探索问题,提出了价值匹配蒸馏(distillation with value matching, DVM)算法,在同构智能体之间进行知识传递,显著减少了智能体的探索空间,从而加快了收敛速度. Gao等[118]针对多智能体强化学习在新任务中训练时间长、资源消耗大的问题,提出了KnowSR知识共享方法. 该方法采用知识蒸馏范式在智能体之间共享知识,使得智能体能够有效学习同构队友的策略,不仅显著提升了智能体的学习效率,同时易于与现有多智能体强化学习算法相整合. Xue等[119]针对分布式环境中点对点知识迁移问题,设计了学习与教导分类强化框架(learning and teaching categorical reinforcement, LTCR),通过模型蒸馏在多个“学生”之间重复使用经验并传递价值函数. 该框架使用Categorical DQN算法解决Q函数不稳定且无界的问题,同时设计了一种高效的通信协议来利用多个分布式智能体之间的异构知识,稳定并加速了学习过程,同时提高了团队的整体性能.

此外,选项学习框架近期也被用于在智能体之间传递知识. Yang等[120]针对智能体如何从其他智能体学习知识这一开放性问题,将多智能体策略迁移建模为选项学习问题,设计了基于选项的多智能体策略迁移框架(multi-agent policy transfer framework, MAPTF). MAPTF易于与现有深度强化方法相结合,可显著提高现有方法在离散及连续状态空间中的性能表现.

3.2.2 课程学习方法

2009年,Bengio首次提出了课程学习(curriculum learning, CL)[121-122]的概念. 人类的学习过程一般遵循着先易后难、由易到难的顺序,借鉴这一学习思想,课程学习主张让模型先从简单的样本/任务开始学习,然后逐步过渡到复杂的样本/任务,从而减少训练时间并提高最终的渐近性能. 2016年,Narvekar等[123]首次将课程学习思想应用于强化学习领域,自此开启了课程强化学习(curriculum reinforcement learning, CRL)的大门.

强化学习中的课程学习方法,实际上是一种特殊的迁移学习方法[121]. 其核心是创建一系列与最终目标任务相似但难度不同的任务序列,进而通过迁移学习方法在任务序列之间进行策略迁移,从而在最终任务上达到加快学习速率、提高渐近性能的目的[124]. 如何科学合理地设置“课程”是课程强化学习所面临的关键问题,现有研究可简单归结为任务定制课程和种群定制课程两类.

1) 任务难度定制课程.

基于任务难度的课程定制方法根据任务的复杂程度,按照先易后难的原则生成任务序列. 根据先验知识设置环境属性变量以生成一系列难度不同的任务是一种典型的课程生成方法. Liu等[125]针对多智能体路径规划问题,提出了分阶段的训练策略,即先在少障碍物和近目标的场景中学习,后增加智能体和动态障碍的数量,同时将目标区域扩展至整张地图,获得了良好稳定的导航策略. De Souza等[126]针对分布式多智能体协同围捕任务,基于课程学习的思想设计了多智能体训练算法,即从最终任务的简单版本开始学习,而后逐渐增大逃逸者速度或减小捕获半径以提高围捕任务难度,直到达到实际难度,显著提升了围捕策略的性能. Grupen等[127]将环境塑造(environment shaping)的概念引入围捕任务中,通过逐渐降低逃逸者与追捕者速度之比创建一个难度逐渐增大的任务序列,有效解决了奖励稀疏的多智能体协同围捕问题.

另一种有效的课程生成方法是将最终任务分解成一系列易于完成的子任务或易于实现的子目标. Jia等[128-129]在其开发的《潮人篮球》游戏AI中,将篮球运动分解为进攻、防守、助攻、控球等5个子任务,通过加权级联课程训练完成任务间的切换,同时设计课程切换器强化整个团队的协同配合,最终形成贯穿整场篮球比赛的综合策略. Du等[130]为解决复杂城市环境下多无人机协同围捕任务中的稀疏回报问题,利用分治求解的思想,将围捕任务分解为追逐、包围、收缩、抓捕4个简单的子任务,通过顺序完成4个子任务引导智能体学习复杂的围捕策略,显著提升了复杂环境中非完备信息约束下的学习速度和围捕效率.

2) 种群规模定制课程.

基于种群规模的课程定制方法根据智能体种群规模的大小,按照自小而大的原则生成任务序

列. 在多智能体尤其是大规模多智能体场景中,智能体的数量越多意味着任务的复杂程度越高,因此根据种群规模的大小设置课程是自然而合理的选择. Shao等[131]将强化学习、迁移学习与课程学习相结合控制StarCraft中的多个单元(智能体). 具体而言,提出基于参数共享的多智能体梯度下降Sarsa(λ)算法(parameter sharing multi-agent gradient descent Sarsa(λ), PS-MAGDS)训练单元,使用迁移学习方法将模型扩展到更困难的场景以加快训练过程并提高学习性能,使用课程迁移学习方法逐步训练在大规模场景中一组单元. 实验结果表明,所提出方法能够学到适当的策略,并能在各种情况下击败内置AI. Agarwal等[132]首次在分布式框架下研究合作行为的多智能体迁移和课程学习问题,其将多智能体交互建模为共享智能体-实体图,基于图神经网络构建多智能体课程强化学习模型. 该模型独立于环境中智能体或实体的数量,且与实体的顺序或排列无关. 实验结果表明,学习到的策略能够快速迁移到不同智能体规模的场景中,课程学习方式能够较好地解决复杂任务难以训练的问题. Yang等[133]为实现高效探索及多智能体不同动作与目标之间的信用分配,基于课程学习设计了一种用于完全协作的多目标多智能体强化学习(cooperative multi-goal multi-stage multi-agent reinforcement learning, CM3)模块化方法. CM3考虑多目标的设置,采用课程学习的方式将问题拆分为两个阶段:在实现多智能体合作之前,先学习实现单智能体目标,同时使用函数增强(function augmentation)方法衔接课程之间的价值和策略函数. 3种典型任务的仿真结果表明,CM3在学习效率、鲁棒性和性能表现等方面均具有明显优势. Wang等[92]针对大规模多智能体环境中的协作学习问题设计了动态多智能体课程学习(dynamic multiagent curriculum learning, DyMA-CL)框架,其核心思想是从小规模的多智能体场景开始学习,并逐步增加智能体数量,从而解决大规模问题. DyMA-CL框架提出了3种跨课程的知识迁移机制以加快学习过程,同时设计了动态智能体数量网络(dynamic agent-number network, DyAN)处理不同课程输入的动态变化. 实验结果表明,使用DyAN的DyMA-CL方法提高了大规模多智能体学习的性能.

然而,在早期阶段的小规模智能体场景中成功训练的策略不一定是适应后期阶段大规模智能体场景的最佳选择. 针对这一问题,Long等[134]在课程学习的基础上进一步引入进化学习的思想,提出了进化种群课程(evolutionary population curriculum, EPC)范式. EPC在每个学习阶段都会维护多个智能体集合,通过跨集合混合匹配和微调在大规模场景中进化,并将适应性最好的智能体集合迁移到下一阶段. 同时,基于自注意力机制在不同“课程”中处理不同数量的智能体,使得EPC能够在参数量固定的情况下泛化到任意数量的智能体场景中. 实验结果表明,随着智能体数量的增长,EPC在稳定性和性能上具有明显优势.

表5简要总结了多智能体深度强化学习策略可迁移性方向的主要研究进展.

表5　多智能体深度强化学习策略可迁移性研究汇总

方法分类	研究方向	主要含义	主要文献
迁移学习方法	任务间迁移	将在源任务中学到的知识向相似但不相同的目标任务迁移	[106-111]
	智能体间迁移	在同一任务中的不同智能体之间进行知识共享或策略迁移	[113-120]
课程学习方法	任务难度定制课程	根据任务复杂度的难易生成任务序列,先易后难	[125-130]
	种群规模定制课程	根据智能体规模的大小生成任务序列,自小而大	[92,131-134]

4　应用前景与发展方向

多智能体深度强化学习已在诸多领域得到广泛的应用,但依然存在一些有待突破的技术瓶颈. 本节展望多智能体深度强化学习的应用前景,并讨论其未来的研究方向.

4.1　应用前景展望

多智能体深度强化学习目前已广泛用于解决复杂环境下的决策控制问题. 在可以预见的未来,随着理论成熟度的进一步提高、相关配套技术的进一步发展,多智能体深度强化学习在集群控制、智慧城市、智能制造等领域的应用前景将会更为广阔.

4.1.1　集群控制

集群机器人系统是群体智能的一个重要应用领域,也是机器人系统未来发展的重要方向之一. 以无人机、无人车、无人艇为典型代表的集群机器人近年来发展迅速,应用广泛. 得益于深度强化学习等人工智能技术的日益成熟,集群机器人协同学习逐渐成为一种发展趋势,在编队控制、路径规划、协同围捕、目标搜索与跟踪等典型任务中扮演着愈发重要的角

色. 随着计算能力的不断提升和训练数据量的持续增加,多智能体深度强化学习有望解决复杂动态环境中的大规模异构集群机器人协同控制问题. 融合了多智能体深度强化学习方法的智能机器人集群,必然会在生产生活中发挥更为重要的作用.

4.1.2 智慧城市

智慧城市作为未来城市的发展方向,已经成为世界各国提升城市治理水平、破解大城市病、提高公共服务质量、发展数字经济的战略选择. 随着以人工智能、云计算、大数据、物联网等为代表的新一代信息技术不断成熟,智慧城市快速发展,成效显著. 得益于多智能体深度强化学习的独特优势,其在新型智慧城市建设过程中发挥着不可替代的重要作用. 可以预见的是,随着多智能体深度强化学习算法的进一步成熟完善,其在智能交通、自动驾驶、智能物流等各个领域将会得到更好的落地应用,从而推动新型智慧城市加速发展.

4.1.3 智能制造

制造业是一个国家的支柱,智能制造是我国制造业发展的必然趋势. 依托由智能机器和人类专家组成的人机集成智能系统,智能制造系统可以在制造过程中进行分析、推理、判断、概念和决策等智能活动. 在智能制造中,多智能体深度强化学习在大数据的基础上有机融合了感知和决策,可用于建立自学习、自适应、高效的智能机器,势必能在智能装配、智能调度、智能运输、智能过程控制等任务中发挥重要作用. 未来越来越多的生产过程可通过智能机器实现, 真正实现无人化和规模化生产.

4.2 未来发展方向

尽管多智能体深度强化学习拥有广阔的应用前景,并已在诸多领域取得了成功的应用,但仍然面临着一系列亟待解决的难题,如部分可观问题、环境非平稳性问题、过拟合问题、信用分配问题等[7,9]. 尤其在可扩展性与可迁移性方面,相关研究尚处于起步阶段,需要在未来的研究工作与实际应用中进一步地探索.

4.2.1 置换同变网络

在未来涉及大规模多智能体系统的应用场景中,模型可扩展性依然是多智能体深度强化学习的核心问题. 如表4所示,现有此方向的研究成果可归结为6类具有可扩展性的神经网络模型. 然而,图形表征法与平均嵌入法想法简单,局限性大;循环神经网络对输入序列排序敏感,不具备置换不变性;虽然深度集合网络、注意力机制、图神经网络3类模型通过共享的输入嵌入层和池化层实现了置换不变性,但共享嵌入层表征能力有限,可能导致模型性能较差. 最近,自加权混合网络(self-weighting mixing network)[135]、深度置换网络(deep permutation network, DPN)[96]、智能体置换不变超网络(agent-permutation-invariant Hypernetwork, API-HPN)[96]等新型网络结构的出现在一定程度上解决了上述问题,但依然有很大的改进空间. 因此,开发具有可扩展性、置换不变性与同变性(permutation invariant and equivariant),具备高效表征能力的神经网络架构,并将其融入多智能体深度强化学习是未来极具价值的研究方向之一.

4.2.2 元强化学习

多智能体深度强化学习算法在实际应用中普遍面临样本效率低、学习速度慢、泛化能力差(每个新任务需重新训练)的性能瓶颈. 当前强化学习的目标是面向特定的环境和任务通过训练得到一个特定的策略,未来的强化学习势必要在这一基础上进行扩展. 在新的方法中,智能体不但能够处理特定的任务,还能根据环境和任务的变化不断调整策略. 对于未接触过的问题,还要掌握一套关于如何学习并解决新问题的方法,这也是未来需要实现的最终目标之一[136]. 无论是现在还是未来,提高多智能体强化学习算法的样本效率和泛化能力始终是该领域的一项重要研究课题.

元强化学习(meta reinforcement learning)[137]的出现为上述目标的实现开辟了一条有希望的技术途径. 该类方法将元学习(meta learning)方法[138]运用到强化学习领域,其目的是自动从一组相关任务中学习有益的归纳偏置,从而为后续强化学习服务. 在多智能体系统中,这方面的工作尚处于起步阶段,且面临着巨大的挑战,如复杂度高、鲁棒性差等. 因此,研究面向海量智能体、复杂应用环境的元强化学习方法是未来多智能体深度强化学习研究的重要方向.

4.2.3 自动课程学习

课程强化学习的主要挑战在于如何科学地为多智能体设置一系列从易到难的课程,从而加快学习速度. 现有的课程生成方法,无论是面向任务的还是基于种群规模的,一般是根据先验知识或专家经验,将复杂任务拆分成一系列难度不同任务的序列. 然而,由于专家经验很难获取、先验知识并不总是可靠,这种手工设计课程的方式亟需改进. 近年来,部分学者提出了自动课程学习(automatic curriculum learning, ACL)[139]的概念与方法,以数据驱动的方式实现了课程设计的自动化. 自动课程学习意味着课

程顺序完全由智能体自主决定,不需要任何先验知识或人为帮助. 可以肯定的是,自动课程学习将是未来多智能体学习领域的一个重要研究方向.

自我对弈(self-play)是一种天然的自动课程生成方法[140],已在高维连续环境下的技能学习任务中展现出巨大潜力[141-142]. 通过Self-play进行自动课程学习,DeepMind公司开发的StarCraft II游戏AI AlphaStar[143]、OpenAI公司开发的Dota 2游戏AI OpenAI Five[144]、腾讯公司开发的《王者荣耀》游戏AI "绝悟"[145]均达到甚至超越了人类玩家的水平. 最近,Yang等[146]探讨了维持多样性意识的自动课程(diversity-aware auto-curriculum)在多智能体强化学习实际应用中所发挥的至关重要的作用,Chen等[147]从变分推理(variational inference)的视角提出了旨在解决协作多智能体强化学习中稀疏奖励问题的变分自动课程学习(variational automatic curriculum learning, VACL)方法,很好地启发了未来自动课程强化学习的研究思路.

5 结 论

多智能体深度强化学习已成为人工智能领域最受关注的研究方向之一. 本文聚焦多智能体深度强化学习的可扩展性与可迁移性,按照从"单"到"多"(智能体数量)、自"小"而"大"(智能体规模)的内在逻辑,对多智能体深度强化学习的背景基础、发展脉络、经典算法和研究进展进行了梳理和归纳. 具体而言,首先阐释了典型的基于值函数的DRL算法和基于策略梯度的DRL算法;然后简述了将深度强化学习扩展到多智能体场景中的3种学习范式,进而从值函数的评估方式入手,将协作多智能体深度强化学习算法分为分解值函数方法和中心化值函数方法两类,并简要介绍了面向多智能体深度强化学习的典型训练平台;接着归纳了6类具有可扩展性的多智能体深度强化学习模型,梳理了迁移学习和课程学习在多智能体深度强化学习可迁移性方向的研究进展;最后讨论了多智能体深度强化学习的应用前景与发展方向.

作为人工智能领域新兴的研究话题,多智能体深度强化学习依然面临采样效率低、回报函数设计难、稳定性差、泛化能力弱等一系列技术瓶颈,其未来发展之路可谓任重而道远. 需要注意的是,从本文调查的研究进展看,DeepMind、OpenAI等国外研究机构在该领域占据绝对领先的优势地位,国内学者的原创性优秀成果凤毛麟角. 因此,亟需更加深入地开展有关多智能体深度强化学习基础理论和前沿算法的研究. 这对于巩固提升我国AI的理论基础和应用水平,缩小与国外先进水平的差距具有重要意义.

参考文献(References)

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

[2] 高阳, 陈世福, 陆鑫. 强化学习研究综述[J]. 自动化学报, 2004, 30(1): 86-100.
(Gao Y, Chen S F, Lu X. Research on reinforcement learning technology: A review[J]. Acta Automatica Sinica, 2004, 30(1): 86-100.)

[3] 付文博, 孙涛, 梁藉, 等. 深度学习原理及应用综述[J]. 计算机科学, 2018, 45(S1): 11-15.
(Fu W B, Sun T, Liang J, et al. Review of principle and application of deep learning[J]. Computer Science, 2018, 45(S1): 11-15.)

[4] 刘全, 翟建伟, 章宗长, 等. 深度强化学习综述[J]. 计算机学报, 2018, 41(1): 1-27.
(Liu Q, Zhai J W, Zhang Z Z, et al. A survey on deep reinforcement learning[J]. Chinese Journal of Computers, 2018, 41(1): 1-27.)

[5] 孙辉辉, 胡春鹤, 张军国. 移动机器人运动规划中的深度强化学习方法[J]. 控制与决策, 2021, 36(6): 1281-1292.
(Sun H H, Hu C H, Zhang J G. Deep reinforcement learning for motion planning of mobile robots[J]. Control and Decision, 2021, 36(6): 1281-1292.)

[6] 赵冬斌, 邵坤, 朱圆恒, 等. 深度强化学习综述: 兼论计算机围棋的发展[J]. 控制理论与应用, 2016, 33(6): 701-717.
(Zhao D B, Shao K, Zhu Y H, et al. Review of deep reinforcement learning and discussions on the development of computer Go[J]. Control Theory & Applications, 2016, 33(6): 701-717.)

[7] 孙长银, 穆朝絮. 多智能体深度强化学习的若干关键科学问题[J]. 自动化学报, 2020, 46(7): 1301-1312.
(Sun C Y, Mu C X. Important scientific problems of multi-agent deep reinforcement learning[J]. Acta Automatica Sinica, 2020, 46(7): 1301-1312.)

[8] 梁星星, 冯旸赫, 马扬, 等. 多Agent深度强化学习综述[J]. 自动化学报, 2020, 46(12): 2537-2557.
(Liang X X, Feng Y H, Ma Y, et al. Deep multi-agent reinforcement learning: A survey[J]. Acta Automatica Sinica, 2020, 46(12): 2537-2557.)

[9] 孙彧, 曹雷, 陈希亮, 等. 多智能体深度强化学习研究综述[J]. 计算机工程与应用, 2020, 56(5): 13-24.
(Sun Y, Cao L, Chen X L, et al. Overview of multi-agent deep reinforcement learning[J]. Computer Engineering and Applications, 2020, 56(5): 13-24.)

[10] 殷昌盛, 杨若鹏, 朱巍, 等. 多智能体分层强化学习综述[J]. 智能系统学报, 2020, 15(4): 646-655.
(Yin C S, Yang R P, Zhu W, et al. A survey

on multi-agent hierarchical reinforcement learning[J]. CAAI Transactions on Intelligent Systems, 2020, 15(4): 646-655.)

[11] 王军, 曹雷, 陈希亮, 等. 多智能体博弈强化学习研究综述[J]. 计算机工程与应用, 2021, 57(21): 1-13.
(Wang J, Cao L, Chen X L, et al. Overview on reinforcement learning of multi-agent game[J]. Computer Engineering and Applications, 2021, 57(21): 1-13.)

[12] Sutton R, Barto S. Introduction to reinforcement learning[J]. Machine Learning, 2005, 16(1): 285-286.

[13] Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning[J/OL]. 2013, arXiv: 1312.5602.

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

[15] Van Hasselt H, Guez A, Silver D. Deep reinforcement learning with double Q-learning[C] . AAAI Conference on Articial Intelligence. Piscataway: IEEE, 2016: 2094-2100.

[16] Hausknecht M, Stone P. Deep recurrent Q-learning for partially observable MDPs[C]. Proceedings of the AAAI Fall Symposium on Sequential Decision Making for Intelligent Agents. Arlington: AAAI, 2015: 1-8.

[17] Wang Z, Schaul T, Hessel M, et al. Dueling network architectures for deep reinforcement learning[C]. International Conference on Machine Learning. Piscataway: IEEE, 2016: 1995-2003.

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

[19] Hasselt H V . Double Q-learning[C]. Advances in Neural Information Processing Systems. Piscataway: IEEE, 2010: 2613-2621.

[20] Lillicrap T P, Hunt J J, Pritzel A, et al. Continuous control with deep reinforcement learning[J/OL]. 2015, arXiv: 1509.02971.

[21] Silver D, Lever G, Heess N, et al. Deterministic policy gradient algorithms[C]. International Conference on Machine Learning. Piscataway: IEEE, 2014: 387-395.

[22] Schulman J, Levine S, Abbeel P, et al. Trust region policy optimization[C]. International Conference on Machine Learning. Piecataway: IEEE, 2015: 1889-1897.

[23] Mnih V, Badia A P, Mirza M, et al. Asynchronous methods for deep reinforcement learning[C]. International Conference on Machine Learning. Piscataway: IEEE, 2016: 1928-1937.

[24] Schulman J, Wolski F, Dhariwal P, et al. Proximal policy optimization algorithms[J/OL]. 2017, arXiv: 1707.06347.

[25] Haarnoja T, Zhou A, Abbeel P, et al. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor[J/OL]. 2018, arXiv: 1801.01290.

[26] Tampuu A, Matiisen T, Kodelja D, et al. Multiagent cooperation and competition with deep reinforcement learning[J]. PLoS One, 2017, 12(4): e0172395.

[27] Tan M. Multi-agent reinforcement learning: Independent vs. cooperative agents[M]. Amsterdam: Elsevier, 1993: 330-337.

[28] Gupta J K, Egorov M, Kochenderfer M. Cooperative multi-agent control using deep reinforcement learning[C]. Autonomous Agents and Multiagent Systems. Springer, 2017: 66-83.

[29] Castaneda A O. Deep reinforcement learning variants of multi-agent learning algorithms[D]. Edinburgh: University of Edinburgh, 2016.

[30] Foerster J, Nardelli N, Farquhar G, et al. Stabilising experience replay for deep multi-agent reinforcement learning[C]. Proceedings of the 34th International Conference on Machine Learning. Piscataway: IEEE, 2017: 1146-1155.

[31] OroojlooyJadid A, Hajinezhad D. A review of cooperative multi-agent deep reinforcement learning[J/OL]. 2019, arXiv: 1908.03963.

[32] Sunehag P, Lever G, Gruslys A, et al. Value-decomposition networks for cooperative multi-agent learning based on team reward[C]. Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems. Piscataway: IEEE, 2018: 2085-2087.

[33] Rashid T, Samvelyan M, Schroeder C, et al. QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning[C]. International Conference on Machine Learning. Piscataway: IEEE, 2018: 4295-4304.

[34] Son K, Kim D, Kang W J, et al. QTRAN: Learning to factorize with transformation for cooperative multi-agent reinforcement learning[C] . International Conference on Machine Learning. Piscataway: IEEE, 2019: 5887-5896.

[35] Rashid T, Farquhar G, Peng B, et al. Weighted QMIX: Expanding monotonic value function factorisation for deep multi-agent reinforcement learning[C]. Advances in Neural Information Processing Systems. Piscataway: IEEE, 2020, 33: 10199-10210.

[36] Yang Y D, Hao J Y, Liao B, et al. Qatten: A general framework for cooperative multiagent reinforcement learning[J/OL]. 2020, arXiv: 2002.03939.

[37] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]. Advances in Neural Information Processing Systems. Piscataway: IEEE, 2017: 5998-6008.

[38] Son K, Kim D, Kang W J, et al. QTRAN: Learning to factorize with transformation for cooperative multi-agent reinforcement learning[J/OL]. 2019, arXiv: 1905.05408.

[39] Wang J H, Ren Z Z, Liu T, et al. QPLEX: Duplex

dueling multi-agent Q-learning[EB/OL]. 2020, arXiv: 2008.01062.

[40] Yang Y D, Hao J Y, Chen G Y, et al. Q-value path decomposition for deep multiagent reinforcement learning[J/OL]. 2020, arXiv: 2002.03950.

[41] Lowe R, Wu Y, Tamar A, et al. Multi-agent actor-critic for mixed cooperative-competitive environments[J/OL]. 2017, arXiv: 1706.02275.

[42] Chu X X, Ye H J. Parameter sharing deep deterministic policy gradient for cooperative multi-agent reinforcement learning[J/OL]. 2017, arXiv: 1710.00336.

[43] Mao H, Zhang Z, Xiao Z, et al. Modelling the dynamic joint policy of teammates with attention multi-agent DDPG[C]. International Conference on Autonomous Agents and Multiagent Systems. Piscataway: IEEE, 2019: 1108-1116.

[44] Foerster J, Farquhar G, Afouras T, et al. Counterfactual multi-agent policy gradients[J/OL]. 2017, arXiv: 1705.08926.

[45] Iqbal S, Sha F. Actor-attention-critic for multi-agent reinforcement learning[J/OL]. 2018, arXiv: 1810.02912.

[46] Samvelyan M, Rashid T, de Witt C S, et al. The StarCraft multi-agent challenge[J/OL]. 2019, arXiv: 1902.04043.

[47] Zheng L, Yang J, Cai H, et al. Magent: A many-agent reinforcement learning platform for articial collective intelligence[J/OL]. 2017, arXiv: 1712.00600.

[48] Sartoretti G, Kerr J, Shi Y F, et al. PRIMAL: Pathfinding via reinforcement and imitation multi-agent learning[J]. IEEE Robotics and Automation Letters, 2019, 4(3): 2378-2385.

[49] Damani M, Luo Z Y, Wenzel E, et al. PRIMAL_2: Pathfinding via reinforcement and imitation multi-agent learning-lifelong[J]. IEEE Robotics and Automation Letters, 2021, 6(2): 2666-2673.

[50] Theile M, Bayerlein H, Nai R, et al. UAV path planning using global and local map information with deep reinforcement learning[C]. The 20th International Conference on Advanced Robotics. Ljubljana, 2021: 539-546.

[51] Bayerlein H, Theile M, Caccamo M, et al. Multi-UAV path planning for wireless data harvesting with deep reinforcement learning[J]. IEEE Open Journal of the Communications Society, 2021, 2: 1171-1187.

[52] Hüttenrauch M, Šošić A, Neumann G. Local communication protocols for learning complex swarm behaviors with deep reinforcement learning[C]. Swarm Intelligence. Springer, 2018: 71-83.

[53] Han L, Sun P, Du Y, et al. Grid-wise control for multi-agent reinforcement learning in video game ai[C]. International Conference on Machine Learning. Piscataway: IEEE, 2019: 2576-2585.

[54] Zhou W H, Liu Z H, Li J, et al. Multi-target tracking for unmanned aerial vehicle swarms using deep reinforcement learning[J]. Neurocomputing, 2021, 466: 285-297.

[55] Yan C, Wang C, Xiang X J, et al. Deep reinforcement learning of collision-free flocking policies for multiple fixed-wing UAVs using local situation maps[J]. IEEE Transactions on Industrial Informatics, 2022, 18(2): 1260-1270.

[56] Yang Y D, Luo R, Li M, et al. Mean field multi-agent reinforcement learning[J/OL]. 2018, arXiv: 1802.05438.

[57] Wang L X, Yang Z R, Wang Z R. Breaking the curse of many agents: Provable mean embedding Q-iteration for mean-field reinforcement learning[J/OL]. 2020, arXiv: 2006.11917.

[58] Sunehag P, Lever G, Liu S, et al. Reinforcement learning agents acquire ocking and symbiotic behaviour in simulated ecosystems[C]. Articial Life Conference Proceedings. New York: MIT Press, 2019: 103-110.

[59] Hüttenrauch M, Adrian S, Neumann G. Deep reinforcement learning for swarm systems[J]. Journal of Machine Learning Research, 2019, 20(54): 1-31.

[60] Gebhardt G H W, Hüttenrauch M, Neumann G. Using M-embeddings to learn control strategies for robot swarms[Z]. Swarm Intelligence, 2019.

[61] Zhang Z, Wang X H, Zhang Q R, et al. Multi-robot cooperative pursuit via potential field-enhanced reinforcement learning[J]. 2022, arXiv: 2203. 04700.

[62] Everett M, Chen Y F, How J P. Motion planning among dynamic, decision-making agents with deep reinforcement learning[C]. IEEE/RSJ International Conference on Intelligent Robots and Systems. Madrid, 2018: 3052-3059.

[63] Sui Z, Pu Z, Yi J, et al. Formation control with collision avoidance through deep reinforcement learning[C]. International Joint Conference on Neural Networks. Piscataway: IEEE, 2019: 1-8.

[64] Sui Z Z, Pu Z Q, Yi J Q, et al. Formation control with collision avoidance through deep reinforcement learning using model-guided demonstration[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 32(6): 2358-2372.

[65] Bai C C, Yan P, Pan W, et al. Learning-based multi-robot formation control with obstacle avoidance[J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(8): 11811-11822.

[66] Zhang T L, Liu Z, Wu S G, et al. Multi-robot cooperative target encirclement through learning distributed transferable policy[C]. International Joint Conference on Neural Networks. Glasgow, 2020: 1-8.

[67] Schlichting M R, Notter S, Fichter W. LSTM-based spatial encoding: Explainable path planning for time-variant multi-agent systems[C] AIAA Scitech 2021 Forum.

Reston, 2021: 1860.

[68] Zaheer M, Kottur S, Ravanbhakhsh S, et al. Deep sets[C]. Advances in Neural Information Processing Systems. Piscataway: IEEE, 2017: 3394-3404.

[69] Lee R, Mou S, Dasagi V , et al. Zero-shot sim-to-real transfer with modular priors[J]. 2018, arXiv: 1809. 07480.

[70] Huegle M, Kalweit G, Mirchevska B, et al. Dynamic input for deep reinforcement learning in autonomous driving[C]. IEEE/RSJ International Conference on Intelligent Robots and Systems. Macao, 2019: 7566-7573.

[71] Huegle M, Kalweit G, Werling M, et al. Dynamic interaction-aware scene understanding for reinforcement learning in autonomous driving[C]. IEEE International Conference on Robotics and Automation. Paris, 2020: 4329-4335.

[72] Shi G Y, Hönig W, Yue Y S, et al. Neural-swarm: Decentralized close-proximity multirotor control using learned interactions[C]. IEEE International Conference on Robotics and Automation. Paris, 2020: 3241-3247.

[73] Shi G Y, Hönig W, Shi X C, et al. Neural-Swarm2: Planning and control of heterogeneous multirotor swarms using learned interactions[J]. IEEE Transactions on Robotics, 2022, 38(2): 1063-1079.

[74] Li Y, Wang L X, Yang J C, et al. Permutation invariant policy optimization for mean-field multi-agent reinforcement learning: A principled approach[J/OL]. 2021, arXiv: 2105.08268.

[75] Hoshen Y. Multi-agent predictive modeling with attentional commnets[C]. Advances in Neural Information Processing Systems. Piscataway: IEEE, 2017: 2698-2708.

[76] Jiang J C, Lu Z Q. Learning attentional communication for multi-agent cooperation[J/OL]. 2018, arXiv: 1805.07733.

[77] Chen C G, Liu Y J, Kreiss S, et al. Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning[C]. International Conference on Robotics and Automation. Montreal, 2019: 6015-6022.

[78] Oroojlooy A, Nazari M, Hajinezhad D, et al. AttendLight: Universal attention-based reinforcement learning model for trafc signal control[J/OL]. 2020 , arXiv: 2010.05772.

[79] Liu X Y, Tan Y. Attentive relational state representation in decentralized multiagent reinforcement learning[J]. IEEE Transactions on Cybernetics, 2022, 52(1): 252-264.

[80] Iqbal S, Witt C D, Peng B, et al. Randomized entity-wise factorization for multi-agent reinforcement learning[C]. International Conference on Machine Learning. Piscataway: IEEE, 2021: 4596-4606.

[81] Hsu C D, Jeong H, Pappas G J, et al. Scalable reinforcement learning policies for multi-agent control[C]. IEEE/RSJ International Conference on Intelligent Robots and Systems. Prague, 2021: 4785-4791.

[82] Batra S, Huang Z H, Petrenko A, et al. Decentralized control of quadrotor swarms with end-to-end deep reinforcement learning[J/OL]. 2021, arXiv: 2109.07735.

[83] Hu S Y, Zhu F D, Chang X J, et al. UPDeT: Universal multi-agent reinforcement learning via policy decoupling with transformers[J/OL]. 2021, arXiv: 2101.08001.

[84] Zhang T J, Xu H Z, Wang X L, et al. Multi-agent collaboration via reward attribution decomposition[J/OL]. 2020, arXiv: 2010.08531.

[85] Zhou T Z, Zhang F B, Shao K, et al. Cooperative multi-agent transfer learning with level-adaptive credit assignment[J/OL]. 2021, arXiv: 2106.00517.

[86] Kipf T N, Welling M. Semi-supervised classication with graph convolutional networks[J/OL]. 2017, arXiv: 1609.02907.

[87] Velikovi P, Cucurull G, Casanova A, et al. Graph attention networks[J/OL]. 2017, arXiv: 1710.10903.

[88] Xu K, Hu W H, Leskovec J, et al. How powerful are graph neural networks? [J/OL]. 2018, arXiv: 1810.00826.

[89] Jiang J C, Dun C, Huang T J, et al. Graph convolutional reinforcement learning[J/OL]. 2018, arXiv: 1810.09202.

[90] Khan A, Tolstaya E, Ribeiro A, et al. Graph policy gradients for large scale robot control[J/OL]. 2019, arXiv: 1907.03822.

[91] Liu I J, Yeh R A, Schwing A G. PIC: Permutation invariant critic for multi-agent deep reinforcement learning[J/OL]. 2019, arXiv: 1911.00025.

[92] Wang W, Yang T, Liu Y , et al. From few to more: Large-scale dynamic multiagent curriculum learning[J]. AAAI Conference on Articial Intelligence, 2020, 34(5): 7293-7300.

[93] Naderializadeh N, Hung F, Soleyman S, et al. Graph convolutional value decomposition in multi-agent reinforcement learning[J/OL]. 2020, arXiv: 2010. 04740.

[94] Ryu H, Shin H, Park J. Multi-agent actor-critic with hierarchical graph attention network[J/OL]. AAAI Conference on Articial Intelligence, 2020, 34(5): 7236-7243.

[95] Ye Z H, Wang K, Chen Y N, et al. Multi-UAV navigation for partially observable communication coverage by graph reinforcement learning[J]. IEEE Transactions on Mobile Computing, DOI: 10. 1109/TMC.2022.3146881.

[96] Hao X T, Wang W X, Mao H Y, et al. API: Boosting multi-agent reinforcement learning via agent-permutation-invariant networks[J/OL]. 2022, arXiv: 2203.05285.

[97] Pan S J, Yang Q. A survey on transfer learning[J]. IEEE Transactions on Knowledge and Data Engineering, 2010,

22(10): 1345-1359.

[98] Zhu Z, Lin K, Zhou J. Transfer learning in deep reinforcement learning: A survey[J/OL]. 2020, arXiv: 2009.07888.

[99] Da Silva F L, Costa A H R. A survey on transfer learning for multiagent reinforcement learning systems[J]. Journal of Articial Intelligence Research, 2019, 64: 645-703.

[100] Song J H, Gao Y, Wang H, et al. Measuring the distance between finite Markov decision processes[C]. Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems. Piscataway: IEEE, 2016: 468-476.

[101] Gamrian S, Goldberg Y. Transfer learning for related reinforcement learning tasks via image-to-image translation[J/OL]. 2018, arXiv: 1806.07377.

[102] Li S, Zhang C. An optimal online method of selecting source policies for reinforcement learning[J]. AAAI Conference on Articial Intelligence, 2018, 32(1): 3562-3570.

[103] Li S, Gu F, Zhu G, et al. Context-Aware policy reuse[C]. International Conference on Autonomous Agents and Multiagent Systems. Piscataway: IEEE, 2019: 989-997.

[104] Yang T P, Hao J Y, Meng Z P, et al. Efficient deep reinforcement learning via adaptive policy transfer[J/OL]. 2020, arXiv: 2002.08037.

[105] 常田, 章宗长, 俞扬. 随机集成策略迁移 [J]. 计算机科学与探索, DOI: 10.3778/j.issn.1673-9418.2105043.
(Chang T, Zhang Z Z, Yu Y . Stochastic ensemble policy transfer[J]. Journal of Frontiers of Computer Science and Technology, DOI: 10.3778/j.issn.1673-9418.2105043.)

[106] Boutsioukis G, Partalas I, Vlahavas I. Transfer learning in multi-agent reinforcement learning domains[C]. Recent Advances in Reinforcement Learning. Berlin, Heidelberg: Springer, 2011: 249-260.

[107] Didi S, Nitschke G. Multi-agent behavior-based policy transfer[C]. Applications of Evolutionary Computation. Springer, 2016: 181-197.

[108] Liu Y, Hu Y, Gao Y, et al. Value function transfer for deep multi-agent reinforcement learning based on n-step returns[C]. International Joint Conference on Articial Intelligence. Piscataway: IEEE, 2019: 457-463.

[109] Qin R J, Chen F, Wang T H, et al. Multi-agent policy transfer via task relationship modeling[J/OL]. 2022. arXiv: 2203.04482.

[110] Niu L, Liang W, Tao J, et al. Multi-agent reinforcement learning policy transfer by buffer[C]. The 7th International Conference on Big Data and Information Analytics. Piscataway: IEEE, 2021: 491-495.

[111] Gao Z, Xu K, Ding B, et al. KnowRU: Knowledge reuse via knowledge distillation in multi-agent reinforcement learning[J]. Entropy, 2021, 23(8): 1043.

[112] Shi H, Li J, Mao J, et al. Lateral transfer learning for multiagent reinforcement learning[J]. IEEE Transactions on Cybernetics, DOI: 10.1109/TCYB.2021.3108237.

[113] Omidshafiei S, Kim D K, Liu M, et al. Learning to teach in cooperative multiagent reinforcement learning[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33: 6128-6136.

[114] Kim D K, Liu M, Omidshaei S, et al. Learning hierarchical teaching policies for cooperative agents[C]. Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems. Piscataway: IEEE, 2020: 620-628.

[115] Liang Y, Li B. Parallel knowledge transfer in multi-agent reinforcement learning[J]. 2020, arXiv: 2003.13085.

[116] Ye D, Zhu T, Zhu C, et al. Model-based self-advising for multi-agent learning[J]. IEEE Transactions on Neural Networks and Learning Systems, DOI: 10.1109/TNNLS.2022.3147221.

[117] Wadhwania S, Kim D K, Omidshafiei S, et al. Policy distillation and value matching in multiagent reinforcement learning[C]. IEEE/RSJ International Conference on Intelligent Robots and Systems. Macao, 2019: 8193-8200.

[118] Gao Z, Xu K, Ding B, et al. KnowSR: Knowledge sharing among homogeneous agents in multi-agent reinforcement learning[J/OL]. 2021, arXiv: 2105.11611.

[119] Xue Z, Luo S, Wu C, et al. Transfer heterogeneous knowledge among peer-to-peer teammates: A model distillation approach[J/OL]. 2020, arXiv: 2002.02202.

[120] Yang T P, Wang W X, Tang H Y, et al. An efficient transfer learning framework for multiagent reinforcement learning[J/OL]. 2020, arXiv: 2002.08030.

[121] Bengio Y, Louradour J, Collobert R, et al. Curriculum learning[C]. International Conference on Machine Learning. Piscataway: IEEE, 2009: 41-48.

[122] Wang X, Chen Y D, Zhu W W. A survey on curriculum learning[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(9): 4555-4576.

[123] Narvekar S, Sinapov J, Leonetti M, et al. Source task creation for curriculum learning[C]. Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems. Piscataway: IEEE, 2016: 566-574.

[124] Narvekar S, Peng B, Leonetti M, et al. Curriculum learning for reinforcement learning domains: A framework and survey[J/OL]. 2020, arXiv: 2003.04960.

[125] Liu Z X, Chen B M, Zhou H Y, et al. MAPPER: multi-agent path planning with evolutionary reinforcement learning in mixed dynamic environments[C]. IEEE/RSJ International Conference on Intelligent Robots and Systems. Las Vegas, 2020: 11748-11754.

[126] De Souza C, Newbury R, Cosgun A, et al. Decentralized multi-agent pursuit using deep reinforcement learning[J].

IEEE Robotics and Automation Letters, 2021, 6(3): 4552-4559.

[127] Grupen N A, Lee D D, Selman B. Multi-agent curricula and emergent implicit signaling[J/OL]. 2021, arXiv: 2106.11156.

[128] Jia H, Ren C, Hu Y, et al. Mastering basketball with deep reinforcement learning: An integrated curriculum training approach[C]. International Conference on Autonomous Agents and MultiAgent Systems. Piscataway: IEEE, 2020: 1872-1874.

[129] Jia H T, Hu Y J, Chen Y F, et al. Fever basketball: A complex, flexible, and asynchronized sports game environment for multi-agent reinforcement learning[J/OL]. 2020, arXiv: 2012.03204.

[130] Du W B, Guo T, Chen J, et al. Cooperative pursuit of unauthorized UAVs in urban airspace via multi-agent reinforcement learning[J]. Transportation Research — Part C: Emerging Technologies, 2021, 128: 103122.

[131] Shao K, Zhu Y H, Zhao D B. StarCraft micromanagement with reinforcement learning and curriculum transfer learning[J]. IEEE Transactions on Emerging Topics in Computational Intelligence, 2019, 3(1): 73-84.

[132] Agarwal A, Kumar S, Sycara K, et al. Learning transferable cooperative behavior in multi-agent team[C]. International Conference on Autonomous Agents and Multiagent Systems. Piscataway: IEEE, 2020: 1741-1743.

[133] Yang J C, Nakhaei A, Isele D, et al. CM3: Cooperative multi-goal multi-stage multi-agent reinforcement learning[J/OL]. 2018, arXiv: 1809.05188.

[134] Long Q, Zhou Z H, Gupta A, et al. Evolutionary population curriculum for scaling multi-agent reinforcement learning[J/OL]. 2020, arXiv: 2003.10423.

[135] Chai J, Li W, Zhu Y, et al. UNMAS: Multiagent reinforcement learning for unshaped cooperative scenarios[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, DOI: 10.1109/TNNLS.2021.3105869.

[136] 万里鹏, 兰旭光, 张翰博, 等. 深度强化学习理论及其应用综述[J]. 模式识别与人工智能, 2019, 32(1): 67-81.
(Wan L P, Lan X G, Zhang H B, et al. A review of deep reinforcement learning theory and application[J]. Pattern Recognition and Artificial Intelligence, 2019, 32(1): 67-81.)

[137] 谭晓阳, 张哲. 元强化学习综述[J]. 南京航空航天大学学报, 2021, 53(5): 653-663.
(Tan X Y, Zhang Z. Review on meta reinforcement learning[J]. Journal of Nanjing University of Aeronautics & Astronautics, 2021, 53(5): 653-663.)

[138] 李凡长, 刘洋, 吴鹏翔, 等. 元学习研究综述[J]. 计算机学报, 2021, 44(2): 422-446.
(Li F C, Liu Y, Wu P X, et al. A survey on recent advances in meta-learning[J]. Chinese Journal of Computers, 2021, 44(2): 422-446.)

[139] Portelas R, Colas C, Weng L L, et al. Automatic curriculum learning for deep RL: A short survey[J/OL]. 2020, arXiv: 2003.04664.

[140] Sukhbaatar S, Lin Z, Kostrikov I, et al. Intrinsic motivation and automatic curricula via asymmetric self-play[J/OL]. 2017, arXiv: 1703.05407.

[141] Baker B, Kanitscheider I, Markov T, et al. Emergent tool use from multi-agent autocurricula[J/OL]. 2019, arXiv: 1909.07528.

[142] Du Y Q, Abbeel P, Grover A. It takes four to tango: Multiagent selfplay for automatic curriculum generation[J/OL]. 2022, arXiv: 2202.10608.

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

[144] Berner C, Brockman G, Chan B, et al. Dota 2 with large scale deep reinforcement learning[J/OL]. 2019, arXiv: 1912.06680.

[145] Ye D H, Chen G B, Zhang W, et al. Towards playing full MOBA games with deep reinforcement learning[J/OL]. 2020, arXiv: 2011.12692.

[146] Yang Y D, Luo J, Wen Y, et al. Diverse auto-curriculum is critical for successful real-world multiagent learning systems[J/OL]. 2021, arXiv: 2102.07659.

[147] Chen J Y, Zhang Y X, Xu Y F, et al. Variational automatic curriculum learning for sparse-reward cooperative multi-agent problems[J/OL]. 2021, arXiv: 2111.04613.

作者简介

闫超 (1995−), 男, 博士生, 从事无人机集群控制、深度强化学习的研究, E-mail: yanchao17@nudt.edu.cn;

相晓嘉 (1980−), 男, 研究员, 博士, 从事无人机任务规划等研究, E-mail: xiangxiaojia@nudt.edu.cn;

徐昕 (1974−), 男, 教授, 博士, 从事机器人优化决策与控制、自动驾驶、强化学习等研究, E-mail: xinxu@nudt.edu.cn;

王菖 (1985−), 男, 讲师, 博士, 从事人机协作、强化学习等研究, E-mail: wangchang07@nudt.edu.cn;

周晗 (1986−), 女, 副教授, 博士, 从事智能仿生控制等研究, E-mail: zhouhan@nudt.edu.cn;

沈林成 (1965−), 男, 教授, 博士, 从事智能无人系统等研究, E-mail: lcshen@nudt.edu.cn.

(责任编辑：郑晓蕾)