中国科学: 信息科学 2023 年 第 53 卷 第 10 期: 1892–1923
SCIENTIA SINICA Informationis
《中国科学》杂志社
SCIENCE CHINA PRESS
评述

博弈智能的研究与应用

郝建业[1,2], 邵坤[2], 李凯[2], 李栋[2], 毛航宇[2], 胡舒悦[3], 王震[4,5*]

1. 天津大学智能与计算学部, 天津 300350
2. 华为诺亚方舟实验室, 北京 100085
3. 上海人工智能实验室, 上海 200232
4. 西北工业大学网络空间安全学院, 西安 710072
5. 西北工业大学光电与智能研究院, 西安 710072
* 通信作者. E-mail: w-zhen@nwpu.edu.cn

收稿日期: 2023–01–06; 修回日期: 2023–02–07; 接受日期: 2023–03–11; 网络出版日期: 2023–10–16

国家杰出青年科学基金 (批准号: 62025602)、国家自然科学基金 (批准号: U22B2036, 11931015, U1836214)、天津市新一代人工智能科技重大专项 (批准号: 19ZXZNGX00010) 和腾讯第三届 “科学探索奖” 资助项目

摘要 博弈智能是一个涵盖博弈论、人工智能等方向的交叉领域, 重点研究个体或组织间的交互作用, 以及如何通过对博弈关系的定量建模进而实现最优策略的精确求解, 最终形成智能化决策和决策知识库. 近年来, 随着行为数据的海量爆发和博弈形式的多样化, 博弈智能吸引了越来越多学者的研究兴趣, 并在现实生活中得到广泛应用. 本文围绕博弈智能这一研究领域, 分别从 3 个方面进行了系统的调研、分析和总结. 首先, 回顾了博弈智能的相关背景, 涵盖了单智能体马尔可夫 (Markov) 决策过程, 基于博弈论的多智能体建模技术, 以及强化学习、博弈学习等多智能体求解方案. 其次, 依照智能体之间的博弈关系不同, 将博弈分为合作博弈、对抗博弈以及混合博弈这三大类范式, 并分别介绍了每种博弈智能范式下的主要研究问题、主流研究方法以及当前典型应用. 最后, 总结了博弈智能的研究现状, 以及亟待解决的主要问题与研究挑战, 并展望了学术界和工业界的未来应用前景, 为相关研究人员提供参考, 进一步推动国家人工智能发展战略.

关键词 博弈智能, 博弈论, 人工智能, 多智能体系统, 强化学习, 均衡求解

1 引言

博弈论 (game theory) [1,2] 作为 20 世纪经济学伟大的成果之一, 主要研究个体或群体在特定约束下的策略优化问题, 在经济学、运筹学、政治学、信息技术、数理科学以及军事推演领域具有广泛应用. 人工智能 (artificial intelligence, AI) 自从 1956 年在达特茅斯 (Dartmouth) 会议上被提出, 经过六十多年的快速发展, 已在现代人类生活的方方面面取得了应用, 世界各主要国家也分别制定了国家人工智能战略. 博弈智能作为一门新兴交叉领域, 其融合了人工智能和博弈论各自方法的优势, 通过对博

引用格式: 郝建业, 邵坤, 李凯, 等. 博弈智能的研究与应用. 中国科学: 信息科学, 2023, 53: 1892–1923, doi: 10.1360/SSI-2023-0010
Hao J Y, Shao K, Li K, et al. Research and applications of game intelligence (in Chinese). Sci Sin Inform, 2023, 53: 1892–1923, doi: 10.1360/SSI-2023-0010

弈关系的定量建模进而实现最优策略的精确求解, 最终形成决策智能化和决策知识库. 近年来, 随着行为数据的海量爆发和博弈形式的多样化 (例如, 人机博弈、非对称博弈、非完全信息博弈等), 博弈智能受到了不同领域学者的广泛关注, 并在现实生活中得到广泛应用. 博弈智能主要研究多智能体系统中的博弈策略学习与求解问题, 一个典型的多智能体系统 (multi-agent system, MAS) [3] 是由多个智能体组成的博弈系统, 其中每个智能体均在决策上具有一定独立性和自主性. 博弈智能旨在对复杂动态多智能体系统内的各智能体之间的交互关系进行建模, 实现对不同博弈参与方最优目标或策略的有效求解.

近年来, 以深度神经网络为代表的机器学习研究提升了决策系统的感知和认知能力, 进而极大加速了博弈智能的发展. 在以围棋为代表的两人零和博弈中, DeepMind 团队 [4,5] 研发的 AlphaGo 综合深度神经网络、蒙特卡洛树搜索和自博弈等技术, 击败了人类围棋冠军并引起学术界的广泛关注. 在以星际争霸、Dota2 为代表的不完美信息博弈中, AlphaStar [6] 和 OpenAI Five [7] 也达到了人类顶级专家水平, 将博弈智能的研究推到了一个新的高度. 这也为后续博弈智能技术发展提供了新的思路, 例如, 在训练过程中引入对抗训练和基于种群的训练, 大幅提升了策略的鲁棒性. 此外, 在早期双重预言机 (double oracle) [8] 技术的基础上, 结合强化学习 (reinforcement learning, RL) [9] 衍生出了一系列对抗博弈的求解方法, 例如, 策略空间的应对预言机 (policy space response oracle, PSRO) [10] 等. 虽然两人博弈研究取得了快速进展, 但是在多人对抗博弈中如何提升策略的鲁棒性和多样性, 高效求解均衡, 目前仍是一个开放问题. 此外, 棋类、视频游戏和现实世界中的群体决策问题尚存在较大差异, 现有博弈智能方法的实际应用也面临诸多挑战. 例如, 复杂动态的自动驾驶路口交互场景下, 交通参与者均有其各自目标, 且相互间的博弈关系存在空间和时间上的动态性, 导致各交通参与者的最优博弈策略难以求解. 因此, 如何缩短虚拟环境下博弈智能技术到现实物理世界落地的差距, 实现大规模实际场景的博弈关系建模与策略求解, 是博弈智能发展的重要目标.

根据智能体间的博弈关系, 博弈智能可分为合作博弈智能、对抗博弈智能和混合博弈智能等形式. 在合作博弈智能中, 所有智能体共享一个全局奖励函数, 所有智能体需要相互合作来最大化整体的求解效率和性能. 从学习范式角度, 合作博弈可划分为独立学习 [11]、联合学习 [12]、协作学习 [13]、集中训练分布执行 [14] 等形式. 由于所有智能体共享同一个奖励函数, 如何划分每个智能体的贡献, 设计各智能体间的信誉分配和通信机制, 是学习最优合作策略的挑战问题. 此外, 由于实际问题中往往涉及大量智能体, 如何实现大规模多智能体高效合作是未来的一个重要研究方向. 在对抗博弈智能中, 博弈双方处于对抗竞争的关系, 每方优化自身的收益都会降低对方的收益. 根据博弈参与方数量的不同又可分为两人零和博弈、两队零和博弈和多人零和博弈. 不同于合作博弈以明确的收益作为优化目标, 对抗博弈以更为复杂的隐含均衡作为目标, 从而对策略求解的高效性、多样性和鲁棒性提出了挑战. 在混合博弈智能中, 所有智能体都有各自的优化目标, 参与方之间需要进行有机的协作, 既要保证优化个体目标, 也要实现系统整体收益的最大化 [15,16]. 混合博弈研究重点关注社会困境和自主协作两大难题, 前者研究如何鼓励智能体学习更加符合群体利益的行为, 或是如何惩罚逾越道德规则的行为. 后者研究在最优策略难以快速求解的情况下, 如何实现各独立智能体在最大化自身收益的同时进行自主协作.

本文围绕博弈智能, 首先总结相关技术背景, 涵盖强化学习、多智能体系统建模求解、博弈论等基础知识. 其次, 本文按照现有博弈智能范式 [17], 将博弈智能分为合作博弈、对抗博弈以及混合博弈这三大类, 分别介绍每一类下的主要研究问题、主流研究方法以及当前典型应用. 具体而言, 针对合作博弈的研究, 介绍主要学习范式、智能体间通信、信誉分配等基础研究问题和主流方法. 对于多智能体系统的可扩展性, 分析几种主要可扩展架构. 同时从游戏 AI、求解器参数优化、无线通信性能优

化、推荐系统这 4 个代表性场景介绍合作博弈的实际应用. 对于对抗博弈, 根据博弈参与方的智能体数量划分为两人零和博弈、两队零和博弈、多人零和博弈, 并分别介绍 3 种类型博弈的主流研究方法, 以及对抗博弈在游戏、网络攻防、军事推演等领域的应用. 针对混合博弈, 介绍以社会困境求解和群体协作博弈为代表的混合博弈主要研究问题和方法, 并分析能源分配和自动驾驶两种代表性的混合博弈应用. 最后, 本文总结当前博弈智能技术的研究现状, 以及复杂任务训练慢、行为策略迁移难、基线构建不充分、推理架构待优化等亟待解决的主要挑战, 并给出学术界和工业界的未来展望, 为博弈智能提供系统性的参考.

2 背景

2.1 多智能体系统建模

多智能体系统 [3] 是由在同一个环境中相互交互的多个智能体组成的系统, 主要依靠集中式、分布式等架构实现智能体的策略学习. 多智能体系统已经在各种领域得到广泛的应用, 包括游戏 AI、智能电网、智慧交通、自动驾驶等. 本小节主要从单智能体与马尔可夫 (Markov) 决策过程, 以及多智能体系统与博弈论两方面展开介绍.

单智能体与马尔可夫决策过程. 单智能体强化学习模型主要用来刻画单个智能体与环境的交互作用, 是对多智能体系统进行建模的基础. 考虑一个智能体通过不断根据环境状态选择相应动作以实现最大化累积收益的序列性决策问题. 该问题可以建模成一个马尔可夫决策过程 (Markov decision process, MDP), 它表示为一个五元组 $\mathcal{M}=(\mathcal{S},\mathcal{A},P,r,\gamma)$, 其中, $\mathcal{S}$ 是状态空间, $\mathcal{A}$ 是动作空间, $P:\mathcal{S}\times\mathcal{A}\to\Delta(\mathcal{S})$ 是满足马尔可夫性的状态转移函数, $r:\mathcal{S}\times\mathcal{A}\to\mathbb{R}$ 是奖励函数, $\gamma\in(0,1)$ 是折现因子. 定义智能体选择动作的策略为 $\pi:\mathcal{S}\to\Delta(\mathcal{A})$, 即选择动作的概率分布是仅与当前状态 s 有关的马尔可夫策略[1)]. 智能体与环境交互的过程如下: 在任意时刻 t, 智能体观察到所处的环境状态 s_t, 根据策略 π 采用动作 $a_t\sim\pi(\cdot|s_t)$, 完成该动作后会从环境中获得即时奖励 $r_t=r(s_t,a_t)$, 并转移到下一状态 $s_{t+1}\sim P(\cdot|s_t,a_t)$. 智能体的目标是求解得到最优策略 π^*, 实现期望累积奖励 $\mathbb{E}\left[\sum_{k=0}^{\infty}\gamma^k r(s_k,a_k)\right]$ 的最大化. 在任意状态 $s\in\mathcal{S}$ 下, 智能体使用策略 π 时的状态值函数表示为 $V^\pi(s)=\mathbb{E}\left[\sum_{k=t}^{\infty}\gamma^{k-t}r(s_k,a_k)\middle|s_t=s\right]$. 对任意状态 – 动作对 $(s,a)\in\mathcal{S}\times\mathcal{A}$, 智能体使用策略 π 时的动作值函数表示为 $Q^\pi(s,a)=\mathbb{E}\left[\sum_{k=t}^{\infty}\gamma^{k-t}r(s_k,a_k)\middle|s_t=s,a_t=a\right]$. 策略 π 的贝尔曼方程 (Bellman equation) 可以表示为 $Q^\pi(s,a)=r(s,a)+\gamma\cdot\mathbb{E}_{s'\sim P(\cdot|s,a)}\left[V^\pi(s')\right]$, 其中 $V^\pi(s)=\mathbb{E}_{a\sim\pi(\cdot|s)}\left[Q^\pi(s,a)\right]$. 该方程刻画了马尔可夫决策过程的动态规划求解过程, 是强化学习方法的主要建模形式和重要理论依据. 假定智能体所有的策略集合为 Π. 可以证明, 存在最优策略 π^* 使得 $V^{\pi^*}(s)=\sup_{\pi\in\Pi}V^\pi(s)$ 在任意状态 s 下成立. 特别地, 存在确定性的最优策略 $\pi^*:\mathcal{S}\to\mathcal{A}$.

多智能体系统与博弈论. 对多智能体交互的建模和研究涉及多个领域, 包括博弈论、多智能体学习、分布式优化、多智能体通信、逻辑和推理、机制设计等 [18]. 本文重点关注多智能体系统的博弈模型, 以及使用学习方法进行求解.

博弈论 [1, 2] 关注参与方之间的行为交互, 是研究具有竞争或合作关系下最优策略求解的数学理论和方法, 在计算机科学、经济学、社会学、运筹学中有广泛的研究. 在博弈中, 参与方通过动作的选择进行交互, 以最大化自身收益为目标. 具体地, 一个 n 人标准形式博弈 (normal-form game) 由 n 个参与方、每个参与方 $i\in\{1,2,\ldots,n\}$ 的动作空间 $\mathcal{A}_i$, 以及收益函数 $u_i:\prod_{i=1}^n\mathcal{A}_i\to\mathbb{R}$ 构成. 若对任意

1) 若无特别说明, 下文中所提及的策略均为马尔可夫策略.

动作组合 $\boldsymbol{a}$, 参与方的收益关系满足 $\sum_i u_i(\boldsymbol{a})=0$, 则将此博弈称为零和博弈 (zero-sum game). 而在一般和博弈 (general-sum game) 中, 参与方的收益不存在这种约束, 使得它可以刻画更广泛的交互关系. 当一个博弈包含多轮决策行为时, 博弈可以表示为扩展式 (extensive-form game), 参与方在每个决策的节点可以根据历史信息或当前状态选择自身的动作. 在扩展式博弈中, 若每个参与方在决策时都可以完美地获悉先前发生的所有事件, 称此博弈为完美信息博弈 (perfect-information game), 否则该博弈包含隐藏信息, 称其为不完美信息博弈 (imperfect-information game). 例如, 围棋是典型的完美信息博弈, 而德州扑克是不完美信息博弈. 定义参与方 i 的策略 $\pi_i\in\Delta(\mathcal{A}_i)$ 是其动作空间上的概率分布. 记所有参与方的策略组合 (strategy profile) 为 $\pi=(\pi_i)_{i=1}^n$, 同时记 $\pi_{-i}=(\pi_j)_{j=1,j\neq i}^n$ 为除参与方 i 外的策略组合. 用 $\boldsymbol{a}=(a_i)_{i=1}^n\in\prod_{i=1}^n\mathcal{A}_i$ 表示所有参与方的一个动作组合. 给定策略组合 $\pi=(\pi_i,\pi_{-i})$, 参与方 i 的期望收益表示为 $U_i^\pi=\mathbb{E}_{a_i\sim\pi_i,\boldsymbol{a}_{-i}\sim\pi_{-i}}[u_i(\boldsymbol{a})]$. 由于所有博弈参与方都假定是理性的, 当其他参与方的策略组合为 π_{-i} 时, 参与方 i 总希望最大化其期望收益, 由此引出参与方 i 对 π_{-i} 的最优回应 (best response, BR), 即 $\mathrm{BR}_i(\pi_{-i})=\{\pi_i\in\Delta(\mathcal{A}_i)|U_i^{(\pi_i,\pi_{-i})}=\max_{\mu\in\Delta(\mathcal{A}_i)}U_i^{(\mu,\pi_{-i})}\}$. 当每个参与方 i 使用的策略均是对其余参与方的最优回应时, 则所有参与方的策略组合构成一个纳什均衡 (Nash equilibrium, NE) [19], 即当且仅当对每个参与方 i 有 $\pi_i^*\in\mathrm{BR}_i(\pi_{-i}^*)$ 时, 策略组合 π^* 是纳什均衡.

由于多智能体的交互往往涉及到环境的动态变化, 因而多智能体强化学习 (multi-agent reinforcement learning, MARL) [20] 任务往往使用马尔可夫博弈 (Markov game) [21] 描述, 其在标准形式博弈的基础上引入了环境状态. 马尔可夫博弈表示为一个六元组 $\mathcal{MG}=(\mathcal{N},\mathcal{S},\{\mathcal{A}_i\}_{i=1}^n,P,\{r_i\}_{i=1}^n,\gamma)$, 其中 $\mathcal{N}=\{1,\dots,n\}$ 是参与方集合, $\mathcal{S}$ 是状态空间, $\mathcal{A}_i$ 是参与方 i 的动作空间, $P:\mathcal{S}\times\prod_{i=1}^n\mathcal{A}_i\to\Delta(\mathcal{S})$ 是状态转移函数, $r_i:\mathcal{S}\times\prod_{i=1}^n\mathcal{A}_i\to\mathbb{R}$ 是参与方 i 的奖励函数, $\gamma\in(0,1)$ 是折现因子. 记参与方 i 的策略为 π_i. 所有参与方与环境的交互如下: 在任意时刻 t, 所有参与方观察到当前状态 s_t, 根据策略选择动作组合 $\boldsymbol{a}_t=(a_{i,t}\sim\pi_i(\cdot|s_t))_{i=1}^n$, 每个参与方 i 从环境中可获得即时奖励 $r_{i,t}=r_i(s_t,\boldsymbol{a}_t)$, 随后所有参与方转移到下一状态 $s_{t+1}\sim P(\cdot|s_t,\boldsymbol{a}_t)$. 记所有参与方采用的策略组合为 $\pi=(\pi_i:\mathcal{S}\to\Delta(\mathcal{A}_i))_{i=1}^n\in\prod_{i=1}^n\pi_i$, 其中 π_i 是参与方 i 的策略集合. 定义 $\pi_{-i}=(\pi_j)_{j=1,j\neq i}^n$ 为除参与方 i 以外的策略组合. 可以看出, 当 π_{-i} 固定时, 马尔可夫博弈转变成马尔可夫决策过程. 对于任意状态 $s\in\mathcal{S}$, 参与方 i 的价值函数定义为 $V_i^\pi(s)=V_i^{\pi_i,\pi_{-i}}(s)=\mathbb{E}\left[\sum_{k=t}^\infty\gamma^{k-t}r_i(s_k,\boldsymbol{a}_k)\middle|s_t=s\right]$; 对任意状态 – 动作组合对 $(s,\boldsymbol{a})\in\mathcal{S}\times\prod_{i=1}^n\mathcal{A}_i$, 参与方 i 使用策略 π_i 时的 Q 函数定义为 $Q_i^\pi(s,\boldsymbol{a})=Q_i^{\pi_i,\pi_{-i}}(s,\boldsymbol{a})=\mathbb{E}\left[\sum_{k=t}^\infty\gamma^{k-t}r_i(s_k,\boldsymbol{a}_k)\middle|s_t=s,\boldsymbol{a}_t=\boldsymbol{a}\right]$. 类似地, 策略组合 π 的贝尔曼 (Bellman) 方程可以表示为 $Q_i^\pi(s,\boldsymbol{a})=r(s,\boldsymbol{a})+\gamma\cdot\mathbb{E}_{s'\sim P(\cdot|s,\boldsymbol{a})}\left[V_i^\pi(s')\right]$, 其中 $V_i^\pi(s)=\mathbb{E}_{\boldsymbol{a}\sim\pi(\cdot|s)}\left[Q^\pi(s,\boldsymbol{a})\right]$.

在多智能体系统中, 参与方 i 的收益不仅与它采用的策略 π_i 有关, 还会受到其他参与方的策略组合 π_{-i} 的影响. 因此, 一个重要的问题是确定多智能体系统的优化目标, 即求解概念 (solution concept) [20]. 这里根据参与方的收益关系分为 3 种情况展开讨论: (1) 在合作博弈中, 每个参与方的奖励函数相同, 即对任意状态 – 动作组合对 $(s,\boldsymbol{a})\in\mathcal{S}\times\prod_{i=1}^n\mathcal{A}_i$ 和任意 $i,j\in\mathcal{N}$, 有 $r_i(s,\boldsymbol{a})=r_j(s,\boldsymbol{a})$. 由于所有参与方的期望收益函数都相同, 因此优化目标是寻找到最优的策略组合 π^* 以最大化期望收益. (2) 在对抗博弈中, 参与方的收益关系往往是零和的, 即对任意状态 – 动作组合对 $(s,\boldsymbol{a})\in\mathcal{S}\times\prod_{i=1}^n\mathcal{A}_i$, $\sum_{i=1}^n r_i(s,\boldsymbol{a})=0$ 成立. 由于参与方 i 收益的增加必定导致其他参与方的收益降低, 因此参与方之间形成了对抗关系. 所有参与方的目标之一是学习到纳什均衡, 此时任何一方单独改变策略都无法增加自身的收益. (3) 在混合博弈中, 参与方之间既有合作也有对抗, 可以使用一般和博弈进行刻画. 在这类博弈中, 多智能体系统的求解目标往往比较复杂, 需要根据实际场景具体分析. 除了以纳什均衡为常用博弈求解目标外, 多智能体系统往往需要考虑参与方的协作, 以帕累托最优 (Pareto optimality) 为系统的求解目标, 即达到一种最优配置状态, 无法在不损失其中一方收益的情况下优化其他参与方的

收益.

2.2 多智能体系统的求解方案

强化学习方法. 强化学习 [9] 是机器学习的研究分支, 主要研究如何通过与环境交互而更新动作或策略, 以取得最大化的累积奖励. 强化学习的基本思想是选择那些可以为智能体带来丰厚回报的动作, 而避免采用会造成长期收益下降的行为. 其主要关注点在于寻找探索 (exploration) 和利用 (exploitation) 的平衡, 在游戏、机器人、自动驾驶等领域有着广泛的应用. Q- 学习 (Q-learning) [22] 是一种被广泛使用的强化学习算法, 它的学习框架如下: 在算法执行过程中, 智能体维护一个记录 Q 值的表格, 即 $Q:\mathcal{S}\times\mathcal{A}\to\mathbb{R}$. 在任意时刻 t, 智能体观察到当前状态 s_t, 根据表格中记录的 Q_t 值采用 ϵ-贪心算法选择动作 a_t, 从环境中获得即时奖励 $r_t:=r(s_t,a_t)$, 并转移到下一状态 $s_{t+1}\sim P(\cdot|s_t,a_t)$. 智能体对 Q 值更新方式如下: $Q(s_t,a_t)\leftarrow(1-\alpha_t)Q(s_t,a_t)+\alpha_t\left(r_t+\gamma\cdot\max_{a'\in\mathcal{A}}Q(s_{t+1},a')\right)$, 其中 α_t 是 t 时刻的学习步长. 可以证明, 在一定条件下, Q 值收敛到最优策略的值函数 Q^*.

博弈学习方法. 多智能体的交互往往使用博弈模型刻画, 因此可以使用博弈求解技术对多智能体系统进行分析和求解. 这里重点介绍针对标准形式博弈的学习方法, 即使用迭代的方式进行策略优化, 求解博弈的均衡. 博弈学习方法可以根据策略学习范式分为集中式学习和分布式学习. 集中式学习的典型代表是双重预言机 [8], 以及它的扩展式框架策略空间响应预言机 [10]. 它们通过不断增加策略种群的多样性, 求解由策略构成的元博弈, 最终实现原始博弈的均衡求解. 分布式学习则通过让每个智能体分别采用固定学习规则进行迭代, 最终使得它们的动力学收敛到某种均衡, 可以高效地解决集中式学习所导致的动作空间随着智能体数量增加而指数爆炸的问题. 分布式学习的典型代表是虚拟对局 (fictitious play, FP) [23], 博弈的各方在每一轮针对对手们的统计平均策略作出最优应对. 可以证明, 这种简明的学习规则可以保证在两人零和博弈等多种博弈中, 平均策略的组合收敛到纳什均衡 [24]. 此外, 得益于在线学习 [25] 的发展, 使用在线学习算法进行博弈求解的技术也受到了广泛关注. 在线学习算法使用遗憾值 (regret) 来衡量算法的性能, 它刻画的是随着轮数的增加算法取得的收益与最优固定动作收益的差值. 研究发现当所有参与方都使用无憾学习 (no-regret learning) 算法时, 即算法的平均遗憾值趋近于 0, 它们的平均策略的组合在两人零和博弈等多种博弈中都可以收敛到纳什均衡 [26]. 无憾学习作为一大类在线学习算法的统称, 包括各种不同更新规则的算法, 如跟随正则化的领导 (follow the regularized leader, FTRL) [27]、乘法权重更新 (multiplicative weight update, MWU) [28]、对冲算法 (hedge) [29]、遗憾匹配 (regret matching) [30] 等.

多智能体强化学习方法. 对于需要与环境交互的多智能体系统, 通常使用马尔可夫博弈进行建模, 利用多智能体强化学习方法进行求解. 与上文所述的单智能体强化学习不同, 多智能体强化学习需要根据参与方的收益关系确定对应的求解概念, 然后再使用系统级别的强化学习算法求解相应目标. 在合作式多智能体强化学习 (cooperative MARL) 任务中, 所有参与方能同时最大化其收益, 因此参与方 i 的目标是学习到最优策略组合 $\pi^*=(\pi_i^*)_{i=1}^n$ 使得对于任意状态 $s\in\mathcal{S}$, 其价值函数均达到最优, 即 $V_i^{\pi^*}(s)=\sup_{\pi_i\in\Pi_i}V_i^{\pi_i,\pi^*_{-i}}(s)$. 而在对抗式多智能体强化学习 (competitive MARL) 任务中, 所有参与方的收益是零和的, 因此系统的优化目标之一是学习到纳什均衡. 可以证明, 在状态和动作有限的马尔可夫博弈中纳什均衡存在, 记相应的策略组合为 π^*, 价值函数为 V^*, 以及 Q 函数为 Q^*. 根据参与方数量可以将对抗式多智能体强化学习分为两人 ($n=2$) 零和的马尔可夫博弈和多人 ($n\geqslant 3$) 零和的马尔可夫博弈. 一种求解两人零和马尔可夫博弈的算法是最小最大 Q 学习 (minimax Q-learning) [21, 31], 其学习框架如下: 与单智能体强化学习中的 Q 学习类似, 参与方在算法执行过程中维护一个记录 Q 值的表格, 即 $Q:\mathcal{S}\times\mathcal{A}_1\times\mathcal{A}_2\to\mathbb{R}$. 在任意时刻 t, 参与双方观察到当前状态 s_t,

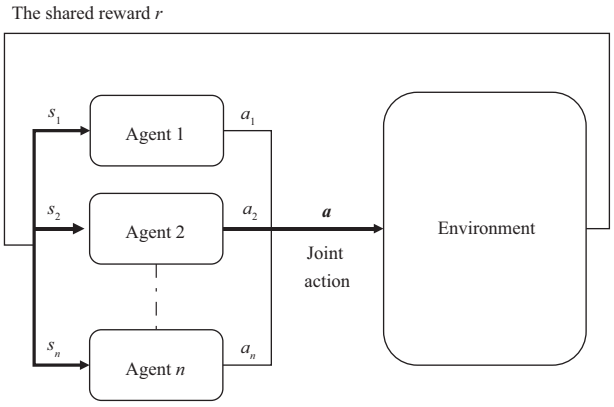

图 1 合作式马尔可夫博弈示意图, 所有智能体共享相同的奖励

Figure 1 Illustration of cooperative Markov game, where all agents share the same reward r

根据当前策略 $\pi=(\pi_1,\pi_2)$ 选择动作 $\boldsymbol{a}_t=(a_{1,t}\sim\pi_1(\cdot|s),a_{2,t}\sim\pi_2(\cdot|s))$, 分别从环境中获取即时奖励 $r_{i,t}=r_i(s_t,\boldsymbol{a}_t)$, 并转移到下一状态 $s_{t+1}\sim P(\cdot|s_t,\boldsymbol{a}_t)$, 参与方对 Q 值和状态 s 处的策略分别更新如下:

$$\begin{aligned}&Q(s_t,a_{1,t},a_{2,t})=(1-\alpha_t)Q(s_t,a_{1,t},a_{2,t})+\alpha_t(r_{1,t}+\gamma\cdot V(s_{t+1})),\\&(\pi_1(\cdot|s_t),\pi_2(\cdot|s_t))=\underset{\mu\in\Delta(\mathcal{A}_1)}{\arg\max}\ \underset{\nu\in\Delta(\mathcal{A}_2)}{\arg\min}\ \mathbb{E}_{a_1\sim\mu,a_2\sim\nu}\left[Q(s_t,a_1,a_2)\right],\\&V(s_t)=\mathbb{E}_{a_1\sim\pi_1(\cdot|s_t),a_2\sim\pi_2(\cdot|s_t)}\left[Q(s_t,\cdot,\cdot)\right].\end{aligned}\tag{1}$$

需要注意的是, 由于在零和博弈中参与方双方的 Q 函数也为零和, 即 $\sum_{i=1}^{2}Q_i^{\pi}(s,a_1,a_2)=0$, 因此可以仅维持一个记录 Q 值的表格. 可以证明, 最大最小 Q 学习使得 Q 收敛到 Q^*. 而对于一般的多人零和马尔可夫博弈, 目前不存在能够收敛到纳什均衡的高效算法.

3 合作博弈智能

任何博弈的研究范畴都主要包含博弈建模和博弈求解. 从博弈建模的角度看, 马尔可夫博弈 [21] 是用来建模多智能体博弈的一般性框架 (详细定义参考第 2 节), 图 1 则展示了典型的合作式马尔可夫博弈的流程. 与其他博弈相比, 合作式博弈最大的特点是, 所有智能体的奖励函数完全相同, 即 $r=r_1=r_2=\cdots=r_n$. 从博弈求解的角度看, 由于所有智能体共享奖励函数, 这就要求所有智能体相互配合, 学习一种最优的联合策略 $\pi(\boldsymbol{a}|s)=\pi(\langle a_1,a_2,\ldots,a_n\rangle|s)$ 来最大化共同的累积奖励, 这本质上是寻找一种社会最优解 (social optimal solution). 合作博弈在现实社会中有着广泛的应用, 例如在交通信号灯自主控制问题中, 所有信号灯组成合作多智能体, 需要观测不同的车流情况, 即 $\langle s_1,\ldots,s_n\rangle$, 一起决定每个信号灯的显示颜色, 即 $\langle a_1,\ldots,a_n\rangle$, 从而最大化整个城市的交通安全和通行效率, 即 r. 本节主要关注合作博弈的求解方案, 将首先详细调研合作博弈智能的主要研究热点, 在此基础上探索合作博弈技术在实际中的应用. 本节的架构和涉及到的代表算法如表 1 [6,12~14,32~63] 所示.

3.1 学习范式

3.1.1 独立学习方法

最简单的多智体合作博弈方法是直接将单智能体方法独立地应用到每一个智能体, 即构成独立学习算法. 图 2 是以智能体 i 的视角的独立学习方法示意图. 这类方法将环境中其他所有智能体也看作

表 1　合作博弈的内容架构和代表性算法

Table 1　Content structure and representative algorithms of cooperative game

Content structure	Category	Representative algorithms
Learning paradigm	Independent learning	Independent DQN [11], Independent PPO [32]
	Joint learning	MDP-Learner [12], MAT [33]
	Coordination graph	Max-Plus [34], DCG [13]
	Centralized training decentralized execution	VDN [14], QMIX [35], MADDPG [36], ATT-MADDPG [37]
Communication	Whom to communicate	DIAL [38], CommNet [39]
	When to communicate	IC3Net [40], Gated-ACML [41]
	What to communicate	Cooperative Q-learning [42], DIAL [38]
Credit assignment	Based on heuristic rule	CLEAN reward [43], Counterfactual baseline [44]
	Based on CTDE	QPD [45], Qatten [46], LICA [47]
	Based on Shapley-value	SQDDPG [48], Shapley counterfactual credits [49]
Scalable architecture	Based on graph structure	Rochico [50], BGC [51], SEIHAI [52]
	Based on mean field	MF-MARL [53], Multi-type MF-MARL [54]
	Based on special network	ASN [55], DyAN [56], G2ANet [57], HPN [58]
Applications	Game AI	AlphaStar [6]
	Solver	MA-DAC [59], BOFiP [60]
	Wireless network	Gated-ACML [41], NCC-MARL [61]
	Recommendation system	DeepChain [62], MAAB [63]

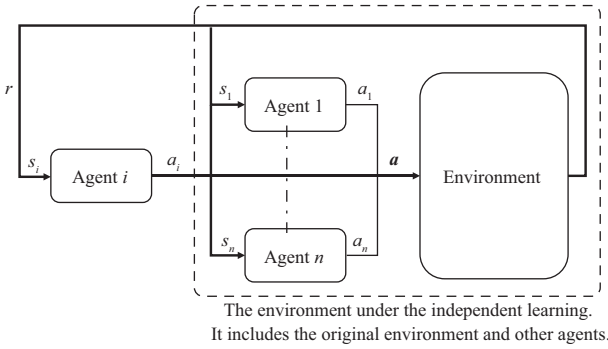

图 2　独立学习方法示意图. 原始环境和其他智能体都被抽象成智能体 i 视角下的环境

Figure 2　Illustration of the independent learning method. The original environment and other agents are abstracted as the environment under the viewpoint of the agent i

是环境的一部分. 以 Q 学习算法为例, 每个智能体 i 只需要学习自己的 Q_i 值函数:

$$Q_i(s,a_i)=Q(s,a_i)+\alpha\left(r+\gamma\max_{a_i'\in A_i}Q_i(s',a_i')-Q_i(s,a_i)\right),\tag{2}$$

并采用贪心策略 $\pi_i(a_i|s)=\arg\max_{a_i\in A_i}Q_i(s,a_i)$ 来选择博弈动作进行执行.

独立学习方法的本质是将联合策略进行独立假设, 即 $\pi(\boldsymbol{a}|s)=\pi(\langle a_1,a_2,\ldots,a_n\rangle|s)=\prod_i\pi_i(a_i|s)$, 而智能体间的合作仅依靠共享的奖励值 $r=r_1=r_2=\cdots=r_n$ 来实现. 这种合作模式虽然不够紧

密, 理论上会造成智能体学习过程的非稳定性 (non-stationary), 但是在一些比较简单的问题下反而很高效 [42]. 随着近几年深度强化学习的发展, 独立学习智能体通过深度神经网络处理环境信息, 已经能够解决一些特定的复杂问题 [11,32]. 例如, 在独立深度 Q 学习 (independent DQN, IDQN) 方法 [11] 中, 每个智能体由一个独立的 DQN 算法控制, 它们之间的合作只通过共享的奖励函数实现. 研究发现, 当共享的奖励足够大时, IDQN 控制的两个智能体能够合作地玩 Pong 游戏, 随着共享的奖励信号逐渐减小, 相应的合作性能也会逐渐降低; 然而, IDQN 很难在类似于星际争霸多智能体挑战赛 (StarCraft multi-agent challenge, SMAC) [64] 等复杂的场景中控制较多的智能体, 因此往往作为研究的基线算法进行对比. 独立近端策略优化 (independent PPO) [32] 是目前已知性能最好的独立学习方法, 它通过参数共享 (即所有智能体使用同一份策略网络)、改进的通用优势估计 (generalized advantage estimation)、值函数剪枝 (value clipping) 等技巧, 能够在 SMAC 等复杂场景中实现多智能体高效合作.

3.1.2 联合学习方法

为了增强合作, 一种直观的方法是将多个合作智能体看成一个联合的 "超级智能体", 这样单智能体方法也可以迁移到这个 "超级智能体". 以 Q 学习算法为例, 这个 "超级智能体" 需要学习一个联合 Q 值函数:

$$Q(s,\boldsymbol{a}) = Q(s,\boldsymbol{a}) + \alpha\left(r + \gamma \max_{\boldsymbol{a}'\in\boldsymbol{A}} Q(s',\boldsymbol{a}') - Q(s,\boldsymbol{a})\right), \tag{3}$$

并采用贪心策略 $\pi(\boldsymbol{a}|s) = \arg\max_{\boldsymbol{a}\in\boldsymbol{A}} Q(s,\boldsymbol{a})$ 来提取最优的联合动作 $\boldsymbol{a} = \langle a_1, a_2, \ldots, a_n\rangle$, 将每个动作分量 a_i 下发给对应的智能体 i 来执行, 这样也可以合作地优化共同目标.

可以看出, 联合学习方法直接在联合策略空间上进行学习, 因此学到的策略具有很强的合作特性, 当满足特定学习条件时 (例如足够多的探索次数以及大小合适的学习步长 α) 理论上可以逐渐逼近真正的最优策略 [12,42]. 然而, 式 (3) 中的算子 $\max_{\boldsymbol{a}'\in\boldsymbol{A}}$ 需要对整个联合策略空间进行遍历, 一方面遍历整个空间效率低, 另一方面没有足够多的样本来较好地评估对应的 Q 值函数, 因此联合学习方法的性能往往很难达到理论上线.

3.1.3 基于协作图的学习方法

为了结合独立学习和联合学习的优点, 研究人员又提出不同的方案将全局的联合策略空间转变成局部的联合策略空间, 同时又避免了完全独立的策略学习. 这类方法的研究 [12,65,66] 最近吸引了广泛关注[2)], 其中基于协作图 (coordination graph) 的研究 [34,67~69] 比较体系化.

基于协作图的方法根据智能体之间的交互依赖关系把所有智能体构建成一个图 $G = (V, E)$, 其中图的节点 $i \in V$ 表示智能体 i, 图的边 $(i,j) \in E$ 表示智能体 i 和 j 之间具有交互依赖关系. 每个智能体 i 在进行动作选择时只需要协调和它有依赖关系的智能体 $j \in \Gamma(i)$, 而不需要考虑不相关的智能体 $k \notin \Gamma(i)$, 从而将学习过程限定在局部的联合策略空间中. 例如, 对于图 3 所示的协作图, 联合 Q 值函数可以表示为如下形式:

$$Q(s,\boldsymbol{a}) = Q_1(s,a_1,a_2) + Q_2(s,a_2,a_4) + Q_3(s,a_1,a_3) + Q_4(s,a_3,a_4), \tag{4}$$

其中 Q_i 在基于协作图的一系列工作中被称为回报函数 (payoff function), 只涉及部分智能体的联合策略.

2) 例如, 文献 [65] 提出不同的分布式值函数 (distributed value functions) 方法, 通过其他智能体 j 的奖励值或者值函数来计算当前智能体 i 的值函数; 文献 [12] 提出基于稀疏合作关系的 Q 值学习方法, 在智能体之间存在紧密交互关系时使用联合学习方案, 在智能体之间存在稀疏交互关系时使用独立学习方案; 文献 [66] 提出基于网络智能体 (networked agent) 的方法, 通过在动态变化的智能体关系图中学习策略并保证收敛.

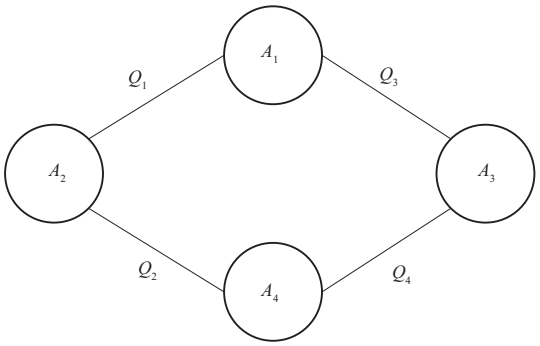

图 3　一种协作图示例, Q_1 可以捕获动作 A_1 和 A_2 的协作关系

Figure 3 An example of coordination graph, where Q_1 can capture the coordination relationship between actions A_1 and A_2

在一般的协作图中, 一个智能体可以和 2 个以上的智能体产生联系, 但是为了简化智能体之间的交互, 可以进一步假设每个回报函数最多涉及 2 个智能体, 此时联合 Q 值函数可以进一步简化为如下形式 [34]:

$$Q(s,\boldsymbol{a})=\sum_{i\in V}Q_i(s,a_i)+\sum_{(i,j)\in E}Q_{ij}(s,a_i,a_j). \tag{5}$$

这种比较简单固定的表示形式非常适合使用深度神经网络来建模. 例如, 深度协作图 (deep coordination graph) [13] 使用一个参数共享的深度神经网络建模 $Q_i(s,a_i)$, 同时使用另一个参数共享的深度神经网络建模 $Q_{ij}(s,a_i,a_j)$, 并且提出低秩估计来近似 $Q_{ij}(s,a_i,a_j)$, 从而避免大量联合动作 $\langle a_i,a_j\rangle$ 无法被有效评估的问题 (3.1.2 小节已经提到, 这个问题在联合学习方法中更加严重).

有了构建好的协作图, 还需要解决另一个核心问题 —— 最优联合动作的选择问题. 实际上, 在基于协作图的方法中, 选择最优联合动作 $\boldsymbol{a}$ 可以转化为依次求解每个智能体的最优动作 a_i, 代表性方案包括基于变量消除 (variable elimination) 的精确算法 [67] 以及基于 Max-Plus 的近似算法 [34]. Max-Plus 是一种特殊的信念传播 (belief propagation) 方法, 其中 "信念" 的数学公式为 $\mu^{ij}(a^j)\leftarrow \max_{a^i}\{\frac{1}{|V|}Q_i(s,a_i)+\frac{1}{|E|}Q_{ij}(s,a_i,a_j)+\sum_{(k,i)\in E}\mu^{ki}(a^i)-\mu^{ji}(a^i)\}$, 它表达了智能体 j 选择动作 a^j 的情况下, 智能体 i 对于可能的最优收益函数的估计. 有了这个估计值后, 智能体 i 需要将这个信息传播给邻居智能体 j, 并且迭代此公式直到信念值收敛. 收敛后任何一个智能体 i 做决策时的策略可以表示为 $\pi_i(a_i|s)=\arg\max_{a_i\in A_i}\{\frac{1}{|V|}Q_i(s,a_i)+\sum_{(k,i)\in E}\mu^{ki}(a^i)\}$.

基于协作图的方法集成了独立学习和联合学习的优势, 具备很大的潜力成为多智能体合作博弈最核心的方法之一, 未来的研究重点可关注于: (1) 基于动态协作图的方法, 这样更符合实际问题的动态性; (2) 基于协作超图 (hypergraph) 的方法, 这样能够对更复杂的智能体交互关系进行准确建模.

3.1.4　集中式训练分布式执行方法

基于协作图的学习方法通过引入协作图表征智能体之间的交互关系, 并基于协作图把联合 Q 值函数进行加性分解, 从而巧妙地结合了独立学习和联合学习的优点. 然而, 实际问题中往往很难获得一个准确的协作图, 而集中式训练分布式执行方法可以自动化地学习对联合 Q 值函数的分解. 这类方法假设联合 Q 值函数和个体 Q_i 值函数 (individual Q-function, 与前述回报函数含义相同, 只是在不

同研究子领域的名词术语有差异) 之间的关系为

$$\arg\max_{\boldsymbol{a}} Q(s,\boldsymbol{a}) = \left\langle \arg\max_{a_1} Q_1(s,a_1), \ldots, \arg\max_{a_n} Q_n(s,a_n) \right\rangle. \tag{6}$$

也就是说, 基于联合 Q 值函数的 “超级智能体” 和基于个体 Q_i 值函数的智能体 i 在选择最优动作时需要保持一致, 即个体全局最大化定理 (individual global max principle, IGM) [70].

求解式 (6) 的经典方法包括 VDN [14], QMIX [35] 和 QTRAN [70]. VDN 通过加和形式 $Q(s,\boldsymbol{a}) = \Sigma_{i\in\{1,\ldots,n\}} Q_i(s,a_i)$ 实现 IGM; QMIX 通过混合网络 (mixing network) 将个体 Q_i 值函数编码为联合 Q 值函数, 并通过单调性 $\frac{\partial Q(s,\boldsymbol{a})}{\partial Q_i(s,a_i)} \geqslant 0, \forall i \in \{1,\ldots,n\}$ 实现 IGM; 加和以及单调性都是 IGM 的充分条件, QTRAN 提出了实现 IGM 的充要条件, 但在具体实现时进行了近似.

由于上述方法需要学习联合 Q 值函数, 这不可避免地要使用全局状态 s 和所有智能体的联合动作 $\boldsymbol{a}$, 这些全局信息可以通过消息传递、通信的方式获取; 但在经典文献中, 研究人员用了更简单的设置 —— 集中式训练, 此时获取这些全局信息就很容易. 相应地, 在真正执行动作时, 每个智能体 i 只需要根据个体 Q_i 值函数来选择最优策略 $\pi_i(a_i|s) = \arg\max_{a_i} Q_i(s,a_i)$, 因此是可以根据个体信息进行分布式执行的. 这就是集中式训练分布式执行 (centralized training decentralized execution, CTDE). 从这个角度看, 前面提到的独立学习方法也可以称为分布式训练分布式执行 (decentralized training decentralized execution, DTDE), 联合学习方法也可以称为集中式训练集中式执行 (centralized training centralized execution, CTCE). 此外, 还有一些其他学习范式, 例如, 集中式教师分布式学生 (centralized teacher decentralized student, CTDS) [71] 采用蒸馏的思想将集中式训练的教师策略蒸馏为分布式执行的学生策略; 个性化训练蒸馏执行 (personalized training with distillated execution, PTDE) [72] 采用智能体个性化的全局信息进行训练, 并将训练后的策略进行蒸馏从而达到分布式执行的目的.

3.2 多智能体通信

智能体可以通过通信的方式获取其他智能体的信息, 进而缓解部分可观测问题, 增强与其他智能体的合作性能, 因此智能体通信是一个经久不衰的研究热点 [73,74]. 具体而言, 多智能体通信包括 3 个核心子问题, 即通信对象、通信时机、通信内容.

通信对象. 其可以大致分为两类, 即与所有智能体通信 [38,39,75,76] 以及与部分智能体通信 [61,77~79]. 前者适用于智能体较少的情况, 后者适用于智能体较多的情况. 这是因为在智能体较多的情况下, 和所有智能体通信会造成信息冗余的问题, 进而妨碍策略的学习; 而在智能体较少的情况下, 即便信息有冗余也能得到一定的缓解 [41].

通信时机. 大多数通信方法都是在每个时刻通信一次, 然而正如上面所说, 并不是所有消息都有利于智能体合作, 此外在特殊场景下 (例如, 有限带宽场景) 也不能时时刻刻发送消息, 因此考虑通信时机也很重要. 目前主流的方案是采用门控机制来判断当前时刻是不是一个合适的通信时机, 具体的门控值一般采用类似于监督学习的方案进行训练 [41].

通信内容. 通信内容相对比较直观, 早期的工作一般关注通信智能体的观测、动作甚至值函数 [42], 而基于深度学习的方案往往将这些信息通过神经网络编码后才通信给其他智能体 [39,41,75,76]. 实际应用时, 基于深度学习的方案通常将通信策略网络嵌入到行为策略网络中, 并且将两者进行联合训练, 这样能达到更好的性能.

3.3 多智能体信誉分配

在合作博弈中, 虽然所有智能体共享相同的奖励函数, 但是每个智能体对整个系统的贡献往往不

同. 信誉分配 (credit assignment, CA) 主要研究如何识别出每个智能体对于系统的真实贡献, 从而将共享的奖励信号合理地分配给不同的智能体, 进而加速合作. 常见的方法分为以下几种.

启发式信誉分配. 传统方法采用一些启发式规则进行信誉分配, 例如差分奖励 (difference reward) [80] 将考虑智能体 i 的全局奖励 $r(s, a_i, a_{-i})$ 与不考虑智能体 i 的反事实奖励 $r(s, \tilde{a}, a_{-i})$ 的差值作为智能体 i 对系统的实际贡献, 其中 $\tilde{a}$ 表示一个默认动作; 相似的方法还包括干净奖励 (clean reward) [43]、反事实基线 (counterfactual baseline) [44] 和奖励设计 (reward design) [81] 等.

基于 CTDE 范式的信誉分配. 基于 CTDE 范式的方法通过将全局 Q 值函数分解为个体 Q_i 值函数实现长期的信誉分配. 例如 QPD [45] 通过引入路径积分实现信誉分配; Qatten [46] 通过引入注意力机制实现信誉分配; LICA [47] 通过判别器超网络 (critic hypernetwork) 实现隐式信誉分配 (implicit CA); ECAQ [82] 通过对个体 Q_i 值函数进行线性加权实现直接信誉分配 (explicit CA).

基于 Shapley-value 的信誉分配. Shapley-value [83] 理论本身就是一种通过考虑参与者的贡献来合理公平地分配收益的方法, 通过将 Shapley-value 和基于 CTDE 范式的方法相结合, 可以从所有可能的智能体组合中去判断一个智能体的期望贡献, 实现有理论保证的信誉分配 [48, 49].

3.4 多智能体可扩展架构

当智能体数量较少时, 独立学习方法和联合学习方法都可以达到不错的效果. 但是在智能体数量变多时, 独立学习方法缺少合作性, 而联合学习方法需要遍历所有智能体的联合动作, 这些都影响学习效率. 因此, 考虑到多智能体合作的一个本质特点就是智能体数量较多, 大规模智能体合作问题, 即可扩展性问题, 是多智能体领域的一个核心问题. 当前能够实现大规模智能体合作的主流方法有以下几种.

基于图结构的方法. 一般而言, 基于图结构的分布式学习方法往往更容易解决可扩展性问题, 因为每个智能体只需要跟邻域内部的智能体进行显示的交互即可. 除了第 3.1.3 小节介绍的基于协作图的方法外, 还有一些方法 [50~52, 84] 是基于分层结构 (可以看作是一种特殊的图) 来实现可扩展性的.

基于平均场的方法. 平均场 (mean-field) 的思想是将智能体 i 与其他所有智能体 $-i$ 的交互通过智能体 i 和一个虚拟智能体 $\overline{-i}$ 来近似, 其中这个虚拟智能体 $\overline{-i}$ 是其他所有智能体 $-i$ 的均值表示, 这样就能够实现很好的可扩展性 [53, 85], 甚至实现多类型 (multi-type) 智能体之间的平均场近似 [54, 86].

基于特殊神经网络结构的方法. 当多智能体通过深度神经网络实现合作时, 也可以设计一些特殊的网络结构实现其可扩展性. 例如 ASN [55] 通过引入体现动作语义的网络结构对动作影响进行显式区分从而具有可扩展性; DyAN [56] 通过引入图神经网络对可变数目的其他实体信息进行状态聚合从而具有可扩展性; G2ANet [57] 通过引入双层注意力机制约简并赋权智能体间的交互关系从而实现可扩展性; HPN [58] 通过引入多智能体环境中针对多个实体之间的置换不变性和置换同变性, 极大地约简智能体的联合状态 – 动作空间, 从而提高了算法的可扩展性.

3.5 合作博弈的应用

对于合作博弈, 一方面既需要继续深挖多智能体合作博弈的基础问题, 从而拓宽其应用范围; 另一方面又需要不遗余力地将该技术推广到真实应用中, 使该技术能够真的为人类社会带来益处. 本节重点介绍一些潜在的高价值应用场景.

游戏 AI. 游戏与人工智能、合作博弈一直有着非常紧密的关系. 一方面, 游戏中往往会涉及到多人共同控制多个角色来联合完成某些任务, 很容易建模成合作型多智能体任务, 因此游戏是研究合作

博弈技术最理想的环境之一; 另一方面, 游戏往往规则明确, 场景可视化良好, 各种控制动作造成的后果也可控, 因此游戏也是合作博弈 (以及其他博弈) 技术最成熟的应用之一 [87]. 具体来说, 合作博弈在游戏场景中大概有以下几大类应用: (1) 构建策略多样的智能体, 从而实现人机协作或者人机对抗; (2) 构建探索能力很强的智能体, 引领人类玩家体验之前没有遇到过的游戏场景, 增加人类体验感; (3) 探索人类认知边界, 例如 AlphaStar 只需基于人类玩家的历史数据即可自主学会战胜人类顶级选手的新策略 [6].

求解器参数优化. 在数学规划求解器中, 例如 Gurobi[3], COPT[4] 和 SCIP[5] 求解器, 往往有成百上千的参数共同决定求解器的性能, 并且这些参数之间耦合性强, 业界常用的黑盒优化方法 (例如贝叶斯优化) 很难对大规模参数进行优化. 相反, 合作式博弈智能方法可以通过大规模参数分组优化、参数取值空间划分等方式, 有效提升数学规划求解器在混合整数规划问题上的求解效率. 具体地, 可以把每一个参数或一类看成是一个智能体, 则一个复杂的有众多约束的规划求解问题, 可以看作是一个多智能体协同优化的问题; 通过协调不同智能体的策略, 可以有效提升规划问题整体的求解效率. 此外, 如何将合作博弈智能同传统优化求解技术融合, 来提升工业场景下传统规划、控制、排产等一系列实际业务问题的求解效率, 是未来重要的研究方向.

无线场景性能优化. 无线网络由大量的接入点 (access point) 构成, 接入点的主要工作是广播无线信号, 以便计算机、手机等设备监测和接入. 接入点广播的无线信号具备多个可调节参数, 例如发射方位角、发射强度、发射频段等, 而不同的参数设置往往会引起效果不一样的相互干扰, 进而影响信号质量. 针对不同场景寻找不同参数设置进而优化信号质量是无线场景的关键问题之一, 但是由于参数较多、参数之间的相互干扰强、外部的环境非稳定 (例如很多用户都可以随意移动), 目前人类专家经验还很难处理比较复杂的场景. 针对这样的无线多频段参数优化问题, 合作式博弈智能方法可以将每个接入点建模成一个智能体, 并且基于智能体之间的数据共享、拓扑关系学习等方式, 更好地实现多接入点的合作式协同超参数优化 [41,61].

推荐系统优化. 推荐系统的目的是根据用户的行为 (例如浏览、点赞、评论、转发等) 来推测用户的偏好, 进而给用户推送合适的项目 (例如文章、音乐、视频等), 从而提升长期期望的点击率、购买率、参与程度等指标. 在实际场景中, 用户往往会顺序访问多个场景 (例如入口页面和详情页面), 并且每个场景都有不同的推荐策略, 此时可以将每个场景建模成一个智能体, 进而基于多智能体技术联合优化不同场景的推荐策略 [62].

合作式博弈智能在产业化中应用广泛, 除上述场景外还包括分布式资源调度 [88]、智能电网调度 [89]、机器人控制 [90]、交通信号灯控制等场景. 虽然现有合作式博弈方法已经在理论和应用方面取得了一定的成果, 如何在大规模产业化场景中实现通用落地, 还面临众多挑战, 需要进一步研究和探索.

4 对抗博弈智能

对抗博弈描述的是参与多方的利益相互冲突, 一方所得收益等于其他参与方所失的博弈场景. 各方在利益的驱动下会展开激烈的竞争. 对抗场景在现实中广泛存在, 从棋牌类对局到即时策略类游戏, 对抗博弈是检验智能程度的 “试金石”; 从网络攻防到军事推演, 对抗博弈是安全领域重要的演练

3) https://www.gurobi.com/.
4) https://www.shanshu.ai/copt.
5) https://www.scipopt.org/.

场 [91]. 对抗博弈智能不仅可以用于这些现实中的对抗场景, 以获得博弈参与方的最优策略; 也可以构造性地用于诸如图像生成、鲁棒优化等问题中, 将优化问题转化为均衡求解. 因此, 对抗博弈智能研究不仅可以在动态且复杂的对抗场景中得出最优决策, 还可以为传统优化问题提供新的思路, 对学术界和产业界的应用均有重要意义.

在博弈论中, 完全对抗的交互场景往往用零和博弈来刻画, 即各方的收益加和为零. 以两方博弈为例, 作为博弈中的一方, 在面对特定的对手时, 最优策略往往是直接探测对手的弱点并加以利用. 然而, 一方面, 对手的策略是动态变化的, 在利用对手的策略时常常会暴露自身的弱点反而被对手利用 [92]; 另一方面, 现实中的对抗博弈场景往往是非常复杂的, 很难实现对手建模并根据博弈收益作出最优应对. 这些挑战要求博弈参与方具有高级别的对抗博弈智能, 在面对多样化的对手时都能有优异的表现. 在博弈论中, 通用方案是求解零和博弈的均衡, 即博弈双方均不断优化各自收益后得到的一对稳态策略. 博弈双方的均衡策略互为彼此的最优应对, 并且均衡也是关于对手行为的合理预测 [93]. 考虑参与方 X 和 Y 的两人零和博弈, 根据均衡的定义可知, 当参与方 X 采用均衡策略, 面对参与方 Y 的任意策略时, X 至少可以获得均衡收益; 并且当 Y 偏离均衡时, X 可能取得优势从而获得超过均衡的收益; 而当参与方 Y 采用均衡策略时也有类似的效果. 这些性质确保博弈参与方可以使用均衡策略来面对多样化的对手, 为实现对抗博弈智能提供了理论保证. 因此本节会首先探讨两人零和博弈均衡求解相关技术的发展.

在对抗博弈中, 博弈的一方不仅可以是单个智能体, 还可以是团体, 例如团队作战的军事对抗. 团体对抗不仅涉及与对手的竞争, 还需考虑团体内部智能体的分布式执行、智能体间的合作等与完全合作博弈类似的问题. 然而, 相比于完全合作博弈以明确的团体收益作为优化目标, 团体对抗以更为复杂的两队博弈的均衡作为目标, 需要研究解决团体策略性能鲁棒提升、团队成员策略的多样性等特殊问题. 因此, 本节还会详细讨论两队零和博弈的具体技术及其发展.

多方对抗也是现实中常见的场景, 例如多人德州扑克、麻将、军事对抗等. 当对抗博弈中存在多个参与方时, 博弈的解概念相对于两人零和或两队零和博弈会有较大差异. 因此, 在本节最后, 会简要介绍多人零和博弈的求解思路和相关成果.

对抗博弈涉及的算法众多, 在经典算法的基础上不断创新的工作也层出不穷, 为了更系统地理解对抗博弈的求解思路, 选取一些代表性算法进行分类总结, 如表 2 [7, 8, 10, 17, 21, 23, 25, 94~101] 所示.

4.1 两人零和博弈

判断均衡是否存在是求解均衡的首要条件. 对于有限的两人零和博弈, von Neumann 在其著名的 Minimax 定理中证明均衡一定存在, 并且此博弈的所有均衡都是等价的, 即会获得相同的收益 [102]. 具体而言, 考虑参与方 X 和 Y 的两人零和博弈, 其策略分别为 $x \in \Delta_m$, $y \in \Delta_n$, X 的收益矩阵为 $A \in \mathbb{R}^{m\times n}$, 那么 Minimax 定理保证

$$\max_{x\in\Delta_m}\min_{y\in\Delta_n} x^{\mathrm{T}}Ay = \min_{y\in\Delta_n}\max_{x\in\Delta_m} x^{\mathrm{T}}Ay = v, \tag{7}$$

其中 v 被称为博弈值. Shapley 将这一结论扩展到马尔可夫博弈, 证明了在状态和动作有限的马尔可夫博弈中存在一对稳态策略也满足这种均衡性质 [103]. 然而, 对于一般的连续空间博弈, 并没有均衡的存在性保证.

接下来详细回顾一下均衡求解的方法. 因为博弈的形式繁多, 例如, 完美信息和不完美信息, 离散空间和连续空间, 标准形式博弈、扩展形式博弈和马尔可夫博弈等各种要素的组合, 这里没有按照常

表 2 对抗博弈的代表算法

Table 2 Representative algorithms of competitive game

Game structure	Category	Representative algorithms
Two-player	Mathematical programming	Linear programming [17], Sequence-form linear programming [94]
	Searching	Minimax search, Alpha-beta pruning [95], MCTS [96]
	Reinforcement learning	Value iteration [21], Minimax-Q learning [21]
	Distributed dynamics	Fictitious play [23], No-regret learning [25], CFR [97], Replicator dynamics [98]
	Oracle-based methods	Double oracle [8], PSRO [10]
Two-team	Reinforcement learning	OpenAI Five (self-play) [7], FTW (population-based methods) [99]
Multiplayer	Game-theoretic methods	Pluribus (multiplayer CFR) [100]
	Reinforcement learning	Suphx (self play) [101]

规的博弈类型进行方法的划分, 而是采用参与方是否可以提前获悉或利用博弈的规则和模型这一原则来对方法进行分类.

4.1.1 基于规则和模型的求解方案

首先讨论博弈双方在已知规则和博弈模型的情况下进行均衡求解的方案, 然后再进一步讨论当无法获取这些信息时对原始博弈进行建模的方法.

求解两人零和博弈最为经典的方法是线性规划和值迭代算法. 根据式 (7), 可以写出参与方 X 或 Y 视角下的线性规划形式. 事实上, 线性规划的强对偶性和 Minimax 定理是等价的 [17]. 对于包含多轮决策行为的扩展式博弈, 虽然可以将其转化为标准形式然后使用上述的线性规划, 但这种转化的复杂度较高, 效率更高的做法是采用序列形式的线性规划 [94]. 而对于包含环境状态的马尔可夫博弈, 虽然它的一些特定形式 (例如单个参与方控制环境转移的设定等) 可以使用线性规划求解, 但因为马尔可夫博弈的博弈值可能存在无理数, 这导致无法在一般情况下使用线性规划进行精确求解; 但可以使用动态规划中的值迭代算法基于贝尔曼方程进行近似求解 [21].

对于棋类对局这种具有完美信息的交替性扩展式博弈, 博弈树可以通过对局的规则生成, 因而博弈的均衡可以通过在树上进行 Minimax 搜索进行递归计算. 然而, 现实中游戏的博弈树往往很大, 无法使用完全的 Minimax 搜索. Alpha-beta 剪枝维护两个变量 α 和 β, 分别表示两个参与方在搜索节点上能够取得的最优值; 算法通过利用 α 和 β 的关系在深度优先的 Minimax 搜索中抛弃无用搜索分支, 提升了搜索效率 [95]. 尽管搜索效率有所提升, 但该算法依旧面临着由于序列决策导致的策略空间指数爆炸的问题, 这类问题可以通过值函数和高效的策略搜索算法来共同解决. 使用值函数可以对博弈树的节点进行值估计, 以此替代最终收益, 从而有效地将博弈在深度上截断. 使用值函数估计和 Alpha-beta 剪枝的技术已经在国际象棋、跳棋中取得了超越人类的表现. 对于围棋, 不仅需要在博弈树的搜索深度上进行截断, 还需在搜索宽度上进行约简. AlphaGo [4, 5] 在使用强化学习得到策略网络和值网络后, 通过高效的蒙特卡洛树搜索 (Monte Carlo tree search, MCTS) [96] 得出最终的策略, 实现了性能和搜索效率之间的平衡.

对于牌类对局这种具有不完美信息的扩展式博弈, 经典方法是序列性线性规划以及反事实遗憾最小化算法 (counterfactual regret minimization, CFR). 通过引入策略的序列性表示, 研究人员发现扩展式博弈可以直接使用序列性线性规划进行求解, 不再需要将博弈转化为标准形式再求解, 从而实现了计算时间随博弈表示呈多项式增长的求解算法 [104]. 随着扑克比赛的发展, 博弈树规模显著增加, 研究

人员一直在开发更为高效的求解算法, 其中最为成功的是反事实遗憾最小化算法 CFR [97]. CFR 通过让两个无憾学习算法进行自我对局来迭代逼近扩展式博弈的纳什均衡. 遗憾值是衡量算法没有选择某个最优确定策略所遭受的损失, 而无憾学习算法是指算法的遗憾值随时间次线性增长, 从而平均的遗憾值趋近于 0. 第 4.1.2 小节将详细讨论无憾学习算法. CFR 的核心是利用算法在全部信息集和相应动作上的瞬时反事实遗憾值 (immediate counterfactual regret) 之和作为整个策略的平均遗憾值的上界, 从而只需在博弈树的全部信息集上优化瞬时反事实遗憾值即可. 随着 CFR 技术的发展, 一系列更加高效的求解算法逐渐被实现, 包括蒙特卡洛 CFR [105], CFR+ [105] 等. 来自阿尔伯塔大学 (University of Alberta) 的研究团队使用 CFR+ 技术离线计算并存储了有限注德州扑克的近似纳什均衡策略, 从而解决了有限注德州扑克这个难题 [106]. 然而对于无限注的情况, 因为动作空间的显著增大, 博弈树的规模指数生长, 无法直接离线计算均衡策略. 为了解决这个问题, DeepStack [107] 通过动作抽象和使用神经网络进行值估计, 约束了搜索的宽度和深度, 缓解了策略空间指数爆炸的问题. 与此同时, 来自卡耐基梅隆大学 (Carnegie Mellon University) 的研究团队开发了 Libratus [108], 在无限注德州扑克比赛中取得了超越人类专家的表现. Libratus 首先通过动作抽象缩小博弈树的规模, 使用蒙特卡洛 CFR 离线计算并存储这种小规模博弈的均衡策略, 称之为 “蓝图策略” (blueprint strategy); 在正式比赛时, Libratus 在决策节点上使用嵌套子博弈求解技术 (nested subgame solving), 利用蓝图策略的博弈值估计, 求解出更为精细化的博弈策略; 并且 Libratus 会利用对局中人类对手的动作来优化动作抽象的方式, 不断增强蓝图策略的效果, 实现自我提升.

对于连续动作空间的博弈, 往往需要同时优化两个智能体的策略. 这需要算法本身不仅在连续空间中进行策略学习与优化, 还需要考虑两个智能体间联合优化的问题. 一个最典型的例子就是生成对抗网络 (generative adversarial networks, GAN) [109], 其通过对生成器和判别器分别设置不同的迭代次数, 引入更恰当的概率散度等, 缓解两个智能体共同优化导致的梯度不稳定问题. 此外, 设计更为合理的优化器、引入课程学习和自监督预训练等都是值得探索的方向.

对于无法事先获悉博弈规则的情况, 可以采用有模型强化学习. MuZero [110] 通过有模型强化学习, 为棋类博弈和 Atari 游戏建立隐式模型, 使得算法可以在对规则一无所知的情况下使用搜索算法. 这里的隐式模型并非对环境动力学的完整刻画, 而是通过隐含状态表征算法需要的状态信息. 通过在隐式模型上运行 MCTS 算法, MuZero 在多个棋类博弈中都取得了最优效果.

4.1.2 无模型求解方案

当智能体无法获取完整的博弈规则和模型时, 除了对此进行建模外, 还可以使用类似强化学习的方法, 在与环境的交互中不断更新策略, 最终收敛到均衡. 这类方法的优势是不需要明确了解领域知识与具体博弈的模型, 从而有更强的适用性.

与强化学习中的无模型 Q 学习类似, Littman 将有模型的值迭代算法进行改进, 提出了最小最大 Q 学习算法 [21, 31], 通过 Q 函数从单个智能体的视角对零和博弈的状态 – 动作信息进行评估. Q 函数的更新规则为

$$V(s) = \max_{\pi_1(\cdot|s)\in\Delta(\mathcal{A}_1)} \min_{a_2\in\mathcal{A}_2} \sum_{a_1} \pi_1(a_1|s)Q(s,a_1,a_2). \tag{8}$$

与 Q 学习不同的是, 这里值函数 V 并不是对 Q 函数直接取最大值得到的, 而是使用零和博弈中的 Minimax 思想逐渐逼近马尔可夫博弈真实的值函数.

除了最小最大 Q 学习这种中心化的更新迭代方式以外, 使用分布式动力学逐渐逼近均衡的方法也有很长的研究历史. 虚拟对局 [23] 是最早被证明可以收敛到均衡的算法之一. 虚拟对局要求, 在每一

轮中, 博弈双方都根据对手的历史行为分布作出最优应对, 那么双方在时间上平均的行为分布最终收敛到均衡. Heinrich 等 [111] 将虚拟对局算法推广到扩展式博弈, 提出了 XFP, 等效于标准形式的 FP 但计算效率更高; 他们也同时提出了 FSP, 一类基于采样和学习的算法用于高效近似 XFP. Heinrich 和 Silver [112] 将虚拟对局与深度强化学习算法结合, 提出了 NFSP 算法, 在有限注德州扑克上取得了超越人类的表现. 值得注意的是, 尽管虚拟对局可以收敛到均衡, 但是它的收敛速率并不高: 需要博弈规模的指数轮才能收敛. 近期的研究表明通过分布式动力学收敛到均衡最快的方法是无憾学习 [25]. 无憾学习通过在线迭代减小算法的 "遗憾", 使得算法的平均遗憾逐渐趋近于 0, 这样当博弈双方都采用无憾学习时, 时间上平均的策略可以快速收敛到均衡. CFR 利用无憾学习这种性质在不完美信息博弈中进行均衡计算 [97]. 复制动力学 (replicator dynamics) 是流行于演化博弈研究的一种连续时间迭代算法, 动力学方程会让策略强化具有更高适应度的动作 [98]. 复制动力学与无憾学习中的 FTRL 算法具有紧密联系, 只需将 FTRL 中的正则项选为策略熵即可得到它的更新方程. 因此, 复制动力学也具有无遗憾的特性, 在均衡求解中具有广泛应用. 文献 [113] 分析了离散时间版本的复制动力学和 Softmax 梯度下降间的联系, 并将复制动力学与函数近似方法相结合, 提出了神经复制动力学 (neural replicator dynamics). DeepMind 团队以神经复制动力学为核心, 结合奖励改造及深度强化学习技术, 提出 DeepNash 算法, 成功实现对军旗中的纳什均衡近似求解 [114].

现实中的博弈往往规模很大并且环境动力学复杂, 一种直观的想法是把原始博弈的规模缩小, 然后在这种小博弈上进行均衡求解. 双重预言机 [8] 算法通过求解子博弈的均衡策略、使用预言机计算原始博弈的最优应对策略、扩充子博弈的规模这种迭代方式求解大型博弈的均衡. 因为双重预言机在最坏情况下是直接求解原始博弈的均衡的, 所以此算法可以确保收敛到均衡. 经验博弈 (empirical game) [115], 或者称为元博弈, 扩展了子博弈的概念, 通过策略模拟的方式生成新的收益表, 然后利用收益表进行策略空间的元分析. 使用经验博弈分析的一种流行且有效的框架是策略空间的应对预言机 PSRO [10], 可以看作是双重预言机和虚拟对局的扩展. 该算法通过引入强化学习来训练预言机、引入元策略求解器来求解均衡, 使算法可以用于求解马尔可夫博弈. 受 PSRO 算法的启发, 后续一系列研究工作围绕经验博弈和策略空间的分析展开. 例如, Balduzzi 等 [116] 利用博弈分解对博弈图景进行刻画, 提出了刻画种群策略性能的指标, 在此基础上开发了 PSRO_{rN} 算法. McAleer 等 [117] 提出了 PSRO 的分布式版本 Pipeline PSRO, 缓解了原始 PSRO 每次迭代都需要使用强化学习训练预言机的效率问题. 面对即时战略游戏星际争霸, AlphaStar [6] 提出基于联盟的训练方式, 融合了优先级虚拟自我对局 (prioritized fictitious self-play, PFSP) 技术, 学习对抗历史上智能体某个分布的策略, 在零和博弈中该分布最后会趋向纳什均衡.

4.2 两队零和博弈

两队零和博弈描述的是两个团体之间的对抗交互场景, 如足球比赛、5v5 游戏和非对称游戏等. 若团队内部的智能体全部采用中心化控制, 如星际争霸游戏, 那么可以将其规约到两人零和博弈进行求解. 因此, 这里重点关注团队内部的智能体采用分布式控制的场景. 与两人零和博弈不同的是, 两队零和博弈随着每队智能体数量的增加, 决策过程不仅涉及团队之间的对抗, 也包含团队内部的协作与配合, 这使得策略优化的空间复杂度显著增加. 算法需要以均衡求解为目标, 解决团队内部的奖励设计、策略多样性和求解效率等一系列问题.

OpenAI Five [7] 是 OpenAI 团队基于 Dota2 游戏开发的 AI 系统, 在 2019 年成为首个在电子竞技中击败世界冠军的 AI 系统. Dota2 游戏是一个典型的 5v5 对抗博弈, 游戏的参与方面临着长时决策、不完美信息、复杂动作空间等一系列挑战. OpenAI 的研究团队使用深度强化学习技术, 在系统设计、

策略探索、模型迭代等方面展开了一系列的研究. 在系统设计方面, 研究团队通过分布式和共享的智能体网络结构降低整个决策空间的复杂度和提高模型的复用性; 通过将强化学习算法中的 γ 增大提高模型的长时决策能力; 通过为每个智能体设计精细化的奖励函数帮助个体策略的稳步提升, 通过个体奖励和团队平均奖励的折中帮助团队内部协作的达成. 在策略探索方面, OpenAI Five 从自我对局中学习, 自然地实现了对环境探索的过程; 为了提高模型在策略空间的探索能力, 研究人员将环境的相关配置 (如游戏单元的血量、速度等特性) 进行随机化, 从而让模型探索不同状态和设定下的鲁棒策略优化. 在模型迭代方面, 因为模型结构和训练策略、支持的游戏设定、游戏发布版本的不断变化, 对每个版本都从零开始训练是无法实现的, 更为合理的做法是对先前学得的策略进行迁移. 研究人员开发了一系列的工具, 称之为 "手术" (surgery), 将原有的策略平滑地迁移到新的设定上, 实现了模型的持续训练.

DeepMind 的研究团队在模型结构和训练方式上进行创新, 开发了 FTW (for the win) 智能体 [99], 在第一人称 3D 游戏 "夺旗" 中超越了人类玩家. 在模型结构上, FTW 智能体结合了快速和慢速两种循环神经网络, 以及共享存储单元, 提升了模型使用存储单元的能力以及生成更为一致的动作序列. 在训练方式上, FTW 智能体的训练采用种群训练方式, 让种群中的智能体进行合作和对抗, 从而提升了策略的多样性; 此外, 对于奖励函数的设计, 研究人员没有针对团队内部的智能体进行特定的奖励设计, 而是通过两层的优化过程 (外层优化团队收益, 内层生成智能体的内部奖励), 让智能体自适应地学会生成合理的奖励. 这些模型结构和训练过程的精巧设计, 使训练得到的 FTW 智能体不仅能够适应不同的地图、队友数量, 还可以适应不同的队友类型, 甚至和人类玩家组队.

对于求解两队零和博弈, 除了上述两个例子, 设计分层的决策模型 (例如对 "意识" 和 "操作" 分别建模 [118])、使用合理的课程学习 [119]、提高采样效率 [120]、引入高效的团队组织结构, 对于缓解策略空间爆炸等问题都有出色的效果.

4.3 多人零和博弈

现实中除了两方的对抗博弈以外, 还存在大量的多方对抗场景. 像多人德州扑克、麻将、绝地求生这类多人游戏, 一方的收益是其他方的损失, 因此这类游戏通常可以建模为多人零和博弈. 尽管多人零和博弈在收益形式上和两人零和博弈类似, 并且纳什均衡在两类有限博弈中都保证存在, 但这两类博弈在均衡性质以及求解方案上存在显著差异. 首先, 两人零和博弈中所有均衡的收益都相同, 而多人零和博弈可能存在收益不同的多个均衡, 并且不存在博弈值的概念. 均衡的多重性使得多人博弈的参与方面临均衡选择的挑战, 并且每个参与方各自选择的均衡策略的组合不一定是纳什均衡. 因此, 作为多人零和博弈中的一方, 采用均衡策略不一定是明智的. 其次, 求解三人及以上零和博弈的计算复杂性至少是求解一般性的两人博弈的难度, 目前还没有高效求解的算法, 并且理论上做近似求解也是非常困难的 [121, 122]. 因为纳什均衡在多人博弈中的这些固有缺陷, 并且很难找到其他的博弈解概念进行克服, 所以在多人零和博弈上的 AI 研究往往不以求解某个具体的博弈解概念为目标, 而是采用自我对局以及遗憾最小化或强化学习技术来获得一些强力策略. 尽管这些方法缺乏很强的理论保证, 但在现实中经常能取得超越人类的表现. 这方面的典型工作包括多人德州扑克和麻将的研究.

多人无限注德州扑克是一个典型的多人不完美信息零和博弈, 通常包含 6 个参与方. Brown 和 Sandholm 在 Libratus 的基础上开发了多人德州扑克 AI 系统 Pluribus, 通过和自身算法的 5 个 "分身" 对战来不断提升模型能力, 不依赖人类经验, 取得了超越人类的性能 [100]. 具体来说, Pluribus 的策略分为离线训练的蓝图策略和在线提升的搜索策略. 离线训练阶段, Pluribus 通过对相似的动作和信息集进行合并和抽象, 减少算法决策的复杂度; 使用改进版本的蒙特卡洛反事实遗憾最小化算法进

行自我对局, 求解得到蓝图策略并存储在本地. 在线提升阶段, Pluribus 在蓝图策略的基础上进行子博弈的重求解, 从而获得更为精细化的可用策略.

麻将也是历史悠久的多人不完美信息零和博弈. 与多人德州扑克相似的是, 麻将也包含巨大的状态空间以及存在大量的隐藏信息. 与之不同的是, 因为麻将存在一些特殊的动作 (如 “碰”、“杠” 等) 能够改变参与方的出牌顺序, 很难得到明确的博弈树; 并且麻将的计分方式比较复杂, 在多轮博弈中需要采用不同的策略, 这些特征为 AI 求解带来新的挑战. 微软亚洲研究院开发的麻将 AI 系统 Suphx 利用监督学习、强化学习自我对局等技术, 在日式麻将上超越了顶级人类选手的平均水平 [101]. 在强化学习的训练过程中, 为了应对不完美信息的挑战, Suphx 使用先知教练 (oracle guiding) 技术, 利用隐藏信息来引导模型的训练方向, 逐步实现知识蒸馏. 为了解决麻将的复杂计分机制, Suphx 使用全盘预测技术分析和理解每轮比赛对终盘的影响. 在实际比赛中, Suphx 在离线策略的基础上使用蒙特卡洛方法对缩小了的状态子空间进行针对性的探索, 从而更好地利用本局比赛信息做出自适应的决策.

4.4 对抗博弈的应用

对抗博弈在现实场景中有着广泛的应用, 不仅包括对抗场景中的策略优化, 还涵盖通过对抗学习实现系统的鲁棒性优化. 随着人工智能技术的发展, 未来对抗博弈智能也会在更多的学术和工业场景中发挥重要作用.

对抗场景的策略优化. 对抗博弈智能在游戏、网络攻防、军事推演等对抗场景中具有重要作用. 对抗博弈智能最直接的应用是在棋牌类对局和电子游戏中. 通过均衡求解或近似, 智能体可以帮助人类在专业对局中更好地训练, 例如使用棋类程序进行人机对抗训练. 另一方面, 使用训练得到的智能体和玩家协同作战, 也可以让玩家体会到更多的游戏乐趣 [99]. 对于电子游戏开发者而言, 对抗博弈智能可以帮助其设计更为强大而智能的游戏角色. 在网络攻防方面, 对抗博弈智能对于攻击方和防守方的对抗策略优化都有重要意义. 对于攻击方, 研究表明可以利用深度强化学习生成对抗样本, 用于攻击静态的可执行文件杀毒引擎 [123]. 对于防守方, 除了可以利用对抗智能寻找系统漏洞外, 还可以通过程序自动部署补丁、设计漏洞进行反击等方式实现智能防守. 在军事模拟和推演方面, 对抗博弈智能可以帮助模拟战局形势、提供策略支持, 甚至是直接参与对抗, 长期以来都受到国家层面的关注. 2016 年, 美国的空战智能系统 ALPHA 在空战模拟仿真器中完胜著名空军教官 [124]. 此外, 美国 DARPA 针对现代战争也提出了新的作战构想, 例如 “马赛克战” (Mosaic warfare), 使用 AI 技术将综合集成的复杂作战体系拆分, 通过寻找一种类似于 “马赛克” 的、灵活可组的标准化作战模块, 使用随机组合、快速拼接形成不断变化的新图像, 从而让己方拥有更多选择, 让对手陷于更复杂、更不确定的战场态势, 最终在体系对抗中赢得主动 [125].

场景生成与鲁棒优化. 学术界和工业界的很多场景生成和鲁棒优化问题都可以建模为对抗博弈, 采用对抗博弈智能进行求解. 例如, 在图像和文本生成设计领域, 生成对抗网络 [109] 将生成问题建模为生成器和判别器的对抗博弈, 通过迭代优化双方的策略, 求解此博弈的均衡, 即生成器产生的分布与输入样本的原始生成分布几乎一致. 在工业领域, 使用对抗学习进行场景生成也是至关重要的. 以自动驾驶为例, 通过对抗学习生成自动驾驶的长尾场景 [126, 127], 可以帮助自动驾驶车辆获得更强的泛化能力. 例如, 可以通过对抗学习来对场景中交通参与者的行为进行改变, 提升自动驾驶算法的碰撞概率, 以创造自动驾驶的难例场景 [127]. 然而, 仅通过对抗学习生成的长尾场景可能会不满足物理规律、交通规则、行为惯例等约束, 这方面的研究还有待进一步探索. 在鲁棒优化问题中, 常常需要考虑环境存在很大不确定性情况下的优化目标. 此时, 可以将问题建模为优化器与环境的对抗博弈, 分析最坏情况下的算法收益, 通过求解 Minimax 均衡获得鲁棒优化的解 [128]. 这种方法的优势

是不需要显示地刻画环境的不确定性, 只需不断优化算法性能的下界, 即可得到在不确定性非常大的环境中依然鲁棒的策略. 这种通过 Minimax 求解鲁棒策略的方法在机器学习的多个领域中均有重要应用 [129, 130].

5 混合博弈智能

第 3 和 4 节介绍的合作与对抗博弈中, 参与者通常为具有固定形式奖励 (payoff) 函数的纯动机智能体, 此时参与者们的收益关系具有固定形式, 即彼此目标一致或完全相反. 然而, 现实场景中也普遍存在着大量混合动机参与者, 即奖励函数形式不固定, 即组成奖励的关键权重可能随环境动态变化; 或者不同参与者间的目标关系各不相同, 既无法应用合作博弈智能中参与者目标与系统整体目标一致的假设, 也无法应用对抗博弈智能中参与者间目标彼此相反的假设. 因此, 本文将此类场景定义为 "混合博弈 (mixed-motive games)", 其中各个博弈参与者保持自身奖励最大化的目标, 但彼此间独立, 不共享自身状态与参数. 在这类博弈中, 多智能体系统的求解目标往往比较复杂, 需要根据实际场景具体分析. 除了以纳什均衡为博弈求解目标外, 多智能体系统往往需要考虑参与方的协作, 以帕累托最优 (Pareto optimality) 为系统的求解目标.

针对混合博弈的相关理论, 目前绝大多数研究工作基于博弈论中的均衡解概念 (如: 纳什均衡), 使用一般和博弈的形式来表示混合博弈问题, 即一种可能存在多个纳什均衡点的矩阵博弈模型. 以猎鹿博弈 (stag hunt game) 为例, 在参与者奖励函数 r 明确的情况下, 一个亲社会的 (prosocial) 参与者 i 的收益函数如下:

$$u_i = (1-\alpha)r_i + \alpha r_{-i}, \tag{9}$$

其中 α 代表参与者的亲社会程度, 即参与者对其他参与者目标的 "妥协" 程度. 对于不同的边界条件, $\alpha = 0$ 表示参与者是完全自私的, $\alpha = 1$ 表示参与者是完全无私的. 此时, 两参与者同时选择 "狩猎" 所对应的概率大小随着双方参与者亲社会级别的增加而增加. 从纳什均衡求解的角度, 存在 $\bar{\alpha} \in (0, 1]$, 使得当任意参与者的收益函数中亲社会程度满足 $\alpha \geqslant \bar{\alpha}$, 上述问题均能唯一收敛到由参与者共同 "狩猎" 所主导的纳什均衡 [131]. 由于上述亲社会等级等复杂概念的引入, 一般和博弈与零和博弈和合作博弈相比, 在求解理论与具体应用角度上均具有更大的挑战性. 从理论角度来说, 以较为简单的一般和正规形式博弈为例. 一方面, 虽然纳什定理确保该类博弈存在混合纳什均衡, 但在该类博弈中求解纳什均衡是一个 PPAD 完全问题 [121, 122], 即目前不存在一个多项式时间算法能高效求解该类博弈的纳什均衡. 另一方面, 该类博弈的纳什均衡往往也不具有价值唯一性, 即如果同时存在多个纳什均衡, 参与者在不同的纳什均衡中可能会得到不同奖励. 从应用角度来说, 早期一般和博弈的相关算法主要面向表格式的有限状态空间, 例如, 纳什 Q 学习 [132]、相关 Q 学习 [133]、朋友或敌人 Q 学习 [134]. 然而, 面向现实世界中普遍存在的高维或连续的状态空间, 以及交互参与者间复杂多变的行为空间, 上述算法难以直接应用.

近年来, 针对如何构建面向实际问题的混合博弈智能, 学术界主要存在以下两个研究热点.

• **社会困境.** 混合博弈可用于描述个体利益与集体利益相冲突的情况, 即社会困境问题. 此类场景中通常容易出现更接近人类的复杂行为, 如, 在一个大团队中, 某参与者采用欺骗 (deception)、搭便车 (free-riding) 等方式获得更高的个体奖励, 但损害了团队整体利益. 因此, 如何鼓励参与者学习更加符合群体利益的行为, 或如何惩罚具有上述逾越道德规则行为的参与者, 都是目前社会困境面临的技术难题.

表 3 混合博弈的内容架构和代表算法

Table 3 Content structure and representative algorithms of mix-motive game

Content structure	Category	Representative algorithms
Solution of social dilemma	Environmental analysis	Melting Pot [135] Inequity aversion [136],
	Auxiliary task	Imitation [15], Partner selection [137], Roth-Erev [138]
	Reward shaping	Social value orientation [16], RUSP [139]
	Social norm	Social evaluation [140], Penalty mechanism [141, 142]
Autonomous coordination	Agent behavior modeling	Motion prediction [143, 144],
	Level-K reasoning	Iterative best response [145, 146]
Applications	Resource allocation	Water resource allocation [147, 148]
	Autonomous driving	Recursive estimation of equilibrium [149~151]

表 4 社会困境: 囚徒困境

Table 4 An example of social dilemma: prisoner's dilemma

	Cooperation	Betrayal
Cooperation	3, 3	−1, 5
Betrayal	5, −1	1, 1

• **协作博弈**[6]**.** 在混合博弈中, 不同于一般的合作博弈, 某一参与者通常不仅需要对其他参与者的行为做出最佳反应, 同时也需要预测其他参与者对自身行为可能的最佳反应, 此时需引入递归推理的决策范式, 这极大增加了参与者间交互行为的灵活度与建模难度. 以自动驾驶场景为例, 车辆在多车交互场景中既需要动态推理其他车辆的意图, 还需要针对其他车辆的策略作出最优应对. 因此, 如何实现各个独立参与者在最大化自身奖励的同时自主地进行协作, 是另一个亟待解决的技术难题.

接下来, 本文将从混合博弈智能的上述两个研究热点出发, 介绍相关的最新学术进展、尚未解决的问题和未来研究方向. 本节的架构和涉及到的代表算法如表 3 [15, 16, 135~151] 所示.

5.1 社会困境求解

社会困境是指集体理性与参与者理性之间存在冲突关系的一类常见问题. 在这类问题中, 合作策略能使所有参与者都取得更好的长期收益, 但合作的各个参与者短期内都需要付出一定成本. 此时, 仅使用最大化自身奖励的目标难以引导参与者间达成一种长期、可持续的合作机制, 甚至还会出现搭便车、欺骗等极端利己策略. 如表 4 所示, 这是一个典型的社会困境, 表 4 中分别给出了两个参与者分别执行合作、背叛行为后的结果收益, 该博弈的唯一纳什均衡解为双方均选择背叛. 虽然双方选择合作能最大化两者的总体收益, 然而, 对于参与者而言, 在单轮中选择背叛能带来更好的收益, 从而难以出现双方均选择合作的情况. 社会困境已经得到广泛的研究, 它与多个不同学科息息相关. 在政治方面, 世界各国制定应对气候变化的生产政策时, 往往会面临社会困境问题. 一个国家可能会因为侧重于短期的收益而减少其可持续性发展, 忽视了如果每个国家都采取同样的行为, 长期发展的目标将受到负面影响. 因此, 各国需要学会与他国协作, 从而保证集体的长期共同利益. 在经济方面, 税收、福利分配也往往涉及社会困境. 在经济系统中, 每个社会参与者之间时刻会发生交互关系, 并可能产生贪婪行为以获得更多利益, 甚至不惜损害别人的利益. 长此以往会造成财富两极分化等一系列社会问

6) 与前文的合作博弈不同, 这里协作博弈的主要区别在于每个参与者有自身的回报函数, 而不是共享相同的回报函数. 参与者在最大化自身累积奖励的同时进行相互协作.

题. 因此, 需要制定强有力的宏观经济调控方式来转移财富、促进公平, 从而有助于最大限度地提高社会的整体回报 [152]. 接下来, 本文主要从计算科学的角度, 将上述社会困境建模为一种基于长时序列的多智能体交互决策问题, 即序列社会困境 (sequential social dilemmas, SSD), 总结现有的主要研究工作, 探究如何优化独立参与者的每一步行为, 实现最佳、可持续的集体协作策略.

为更好地理解序列社会困境的实际意义, 学术界近年来建立了两类经典的序列社会困境问题: 生产者困境, 即单个参与者为了提供公共资源必须付出成本; 消费者困境, 即单个参与者为了自身利益会自私地占有公共资源. 针对上述概念, DeepMind 团队在 Melting Pot 项目 [135] 中分析了在这样的利益关系下, 至少有一个自私的参与者会倾向于背叛或搭便车. 因为与长期收益的损失相比, 该参与者可以获得比其他紧密合作的参与者更高的收益, 且其他的协作者越多, "搭便车" 的收益越大, 也就是常说的 "浑水摸鱼" [136]. 文献进一步指出, 在这样的问题设定下, 现有的合作博弈智能无法引导自私参与者服从集体利益, 且合作难度随参与者数量增加而极大增加, 急需一套与合作博弈、对抗博弈不同的视角看待这类问题, 即混合博弈. 由此, 上述问题以及类似的衍生问题, 也被学术界确立为混合博弈的经典测试场景.

目前, 在序列社会困境的求解方面, 国内外团队展开了积极的研究. 在研究的初期阶段, DeepMind 团队首次从马尔可夫博弈的角度出发, 分析了序列社会困境的物理意义, 并且尝试在 Harvest 和 Cleanup 两个环境上研究序列社会困境问题, 将参与者分为一个独立学习的创新者和多个以指标模仿学习作为辅助奖励的模仿者, 在上述序列社会困境的场景中实现了参与者奖励的相对一致 [15]. Anastassacos 等 [137] 主要探究如何将多人协作博弈分解为多个双人协作博弈, 引入基于序列神经网络的参与者选择机制, 动态形成临时的双人协作博弈, 最终形成整体的协作, 实现参与者间从互相背叛到互相协作的学习模式. Merhej 等 [138] 从 Roth-Erev 理论角度分析如何提高序列社会困境问题上的训练效果, 将整个任务的成功率作为标签引入到纳什均衡的计算中, 并考虑参与者奖励的平等性和对群组的贡献. 进一步地, 受社会学相关研究的启发, McKee 等 [16] 首次提出基于 Harvest 和 Cleanup 两个序列社会困境场景研究参与者间协作, 引入了混合意图概念, 从而考虑社会价值取向 (social value orientation, SVO) 所带来的影响. 它们对环境中的 IPPO 参与者设置 SVO 值, 并与实际自身与其他参与者奖励的比例形成自回归偏差, 将其作为一个内在回报, 重塑当前参与者的奖励. 类似地, OpenAI 团队在自研的序列社会困境场景中加入一个具有社会意识的指引机制 RUSP, 从而让一般的多智能体强化学习算法可以学习亲社会行为 [139]. 具体地, 该团队设计了一个增强环境信息的随机不确定社会偏好, 建立参与者间交互关系、奖励单调关系权重的分块矩阵, 并附加在原有的参与者奖励上, 适当引入噪声保证策略的鲁棒性, 从而实现了对序列社会困境更加精准的建模效果. 综上所述, 目前主要的研究进展如下: 在问题建模方面, 使用谢林图等方式, 通过引入合作团体中背叛行为的增益, 验证该问题是否属于序列社会困境; 针对 DTDE 下参与者无法直接通信带来的挑战, 引入基于网络的多智能体学习框架和好奇心机制等, 探索最有利于参与者协作的状态特征, 从而实现参与者间的协作; 针对最大化自身奖励的自私行为, 引入不平等厌恶、SVO [153] 等社会学模型, 用其他参与者的奖励合理重塑自身奖励, 从而学习集体协作策略.

此外, 如何防止参与者滥用自己的激励功能仍然是一个开放的研究问题. 如果参与者采取欺骗策略掩盖他们的真实意图, 这种滥用可能会被其他参与者发现. 理解不诚实和不道德的行为是一个重要的研究途径, 因为有能力欺骗的参与者对参与者之间的关系构成了严重威胁. 在社会学习和序列社会困境研究领域, 欺骗被认为是参与者仅考虑最大化参与者利益、背离群体利益所采取的一种更加激进的 "交互" 策略. 这里主要考虑参与者通过隐藏其真实意图或故意说服其他参与者采取某种次优策略符合其最佳利益来欺骗其他参与者. 混合博弈使参与者能够重塑其他参与者的奖励函数, 在混合博弈

表 5 群体自主协作: 性别博弈

Table 5 An example of autonomous collaboration: game of battle of sex

	Play football	Watch opera
Play football	10, 7	0, 0
Watch opera	0, 0	7, 10

中促进协作以获得更高的集体奖励. 然而, 这种修改允许参与者访问彼此的学习过程, 当参与者没有意识到自己被欺骗而采取实际上不符合其自身最佳利益的策略时, 这可能会增加被操纵的风险. 关于社会学习中的准则, 主要包括引入社会准则规约参与者行为或引入一些机制促进社会准则的学习这两个方向. 针对社会准则引入, Anastassacos 团队 [140] 从建立社会评价的角度出发规范自私参与者的合作行为. 在 Q 学习基础上, 引入定义好的社会规则, 将参与者间是否遵循这一规则作为更新策略使用的内在回报, 促进合作种群的形成. 针对社会准则学习, Köster 团队 [141] 为更好地促进异构参与者的合作, 设计公众制裁的额外动作空间. 参与者可惩罚触犯规则的参与者从而建立社会准则. 目前这方面的研究主要是实验型研究, 社会准则解释为一种实验设置或者实验现象. 社会准则能否基于神经网络来预测是一个未来的研究方向 [154].

5.2 群体协作博弈

不同于具有系统整体收益定义、参与者可以参数共享的合作博弈智能, 当问题最优策略未知时, 混合博弈智能旨在如何让各个独立参与者在最大化自身收益的同时自主地进行协作, 即群体协作博弈. 如表 5 所示, 这是一个典型的群体协作博弈问题, 表 5 中分别给出了两个参与者分别执行踢足球、看歌剧行为后的结果收益, 该博弈同时存在多个纳什均衡解. 其中, 纯策略纳什均衡解为双方选择踢足球和双方选择看歌剧, 然而参与者对于这两个纳什均衡解存在不同偏好. 在混合博弈智能中, 参与者可以通过观察环境中其他参与者的行为来学习, 这使参与者能够发现仅通过参与者探索难以学到的有效策略, 并快速适应新的环境. 这也是群体中参与者认知和发展的核心特征, 通过利用集体智慧, 可以使参与者 “站在巨人的肩膀上”.

在混合博弈中实现协作的一种有效范式是对手行为建模. 在这种范式中, 参与者会考虑他们对其他参与者预期学习的影响. 由于混合博弈中参与者的高度动态性和多样性, 需要探索合理的行为建模方式, 以及高效的博弈求解方法, 来实现更精准的意图识别. 这类方法主要聚焦于通过对其他参与者的策略、目标、类别等建模来实现更好的协作. 例如, Raileanu 团队 [143] 在自研的异构多智能体协作博弈任务中尝试给每个参与者一个固定目标, 采用神经网络推断其他参与者的后续行为, 并基于真实行为进行交叉熵更新. 这个任务中的目标是确定且不随时间变化的, 多步推断时神经网络的参数不变, 并且要先观察其他参与者的行动再更新自身, 因此具有一定的局限性. 进一步地, 刘思齐团队 [144] 提出基于 PBT 的连续控制框架, 结合奖励塑型的自动优化, 在连续控制的环境中进行端到端的学习. 同时, 该团队引入了收益函数中差异化的折现因子促进稀疏回报长时任务下的协作博弈, 并采用反事实的策略更新方式来分析未知参与者的行为.

除此之外, 在合作博弈智能与对抗博弈智能的现有工作中, 也存在一些基于迭代最优反馈算法的相关工作对混合博弈的问题做了相应的讨论. 例如, OpenAI 与牛津大学 (University of Oxford) 团队在策略梯度计算时尝试引入额外的修正项, 来考量单一参与者策略对其余参与者策略学习过程的影响. 由于该修正项需要其他参与者的策略梯度, 因此提出迭代式建模其他参与者策略的方法, 用各个参与者在同一状态下的历史动作, 构建了一个极大似然估计问题 [145]. 北京大学卢宗青团队 [146] 则以多步

迭代最优反馈的视角来建立不同水平的决策模型, 考虑到较低水平的参与者策略模型可以由真实的参与者行为微调得到, 而较高水平的参与者策略模型有更强的合作/对抗模式, 因此对所有水平的模型使用贝叶斯融合的方式得到一个混合的策略 $\pi_{\text{mix}}(\cdot|s)$ 作为决策模型. 该方法可以实现其他参与者策略建模与自身策略优化的同步进行, 以及基于参与者策略模型的环境动态推理, 具有直接延伸到混合博弈问题的价值. 目前, 尽管这些工作在混合博弈的问题中做了一定程度的讨论, 但有待进一步的理论分析和实验验证.

5.3 混合博弈的应用

混合博弈这种广泛存在的博弈形式, 在社会学、经济学以及行为学等领域更为普遍. 如何将混合博弈的相关知识运用到人类发展、国际合作、制度制定等问题上, 不仅有着重要的学术价值, 而且更具深刻的社会价值. 例如, 在制度制定方面, 各个地方单位的利益时常会出现冲突, 此时应该根据各单位的具体能力与特长合理分配不同任务, 一方面促进公平, 另一方面保证政府的整体利益. 近年来, 在资源分配和自动驾驶场景上, 国内外研究团队通过引入混合博弈智能技术尝试解决资源的公共利益最优性和多车交互的决策问题.

资源分配. 在最新的一些研究中, 国内外研究团队开始将序列社会困境应用在资源分配的自研场景中. 例如, Hauser 等 [155] 研究在人才资源分配问题中个体间生产力、利益导向不同的情况, 并构建针对不平等个体互惠互利导向的通用资源分配算法, 促进长期稳定的异构群体协作行为. Pretorius 等 [147] 研究在自研的水资源分配任务中如何避免自私行为的产生. 其中, 每个参与者需要根据自己的紧急程度控制管道的流量, 基于网络化的分布式多智能体博弈框架, 参与者间通过高效的通信机制, 共享邻近参与者的观测信息实现协同决策. Hostallero 等 [148] 在自研的资源共享环境上, 研究参与者之间的评价学习, 从而最大化系统整体的外部奖励. 在独立 DQN 基础上, 参与者使用一个额外的 DQN 广播对自身转移的评价向量, 并接收来自其他参与者自身的转移评价向量, 将其平均后作为辅助奖励, 重塑该参与者的回报. Barfuss 等 [156] 提出一种评估长期不稳定因素下气候变化的序列社会困境建模方法, 确保在社会个体利益彼此耦合的情况下公共利益的最大化.

自动驾驶. 基于混合博弈智能的思想, 学术界近年来提出了一些更加侧重于多车交互、协作博弈的自动驾驶仿真环境, 如华为提出的 SMARTS [157]、Meta 提出的 Nocturne [158]. 在这样的仿真环境中, 新手驾驶员能够在部分观测信息中看到其他驾驶员, 同时利用这些信息学习驾驶策略. 在无保护路口、环岛和高速匝道等交互场景中, 自动驾驶车辆需要对交通参与者进行意图推理并与其频繁交互来决定驾驶策略, 确保安全舒适与交通流畅 [159]. 而基于有限状态机的传统方法通常只能处理特定场景、特定维度的决策问题, 泛化能力差, 交互规则繁杂耦合 [160]. 在路口多车交互场景下, 基于混合博弈的自动驾驶多车交互决策框架通过借鉴 Level-K 迭代推理的方式 [149], 估计各交通参与者最优动作序列, 得到自车的最优决策. 在这种多轮迭代推理框架下, 每一轮都结合上一轮各交通参与者的推理状态, 对各交通参与者进行多步推演, 在每一步选择合理的动作空间采样, 获取累计回报高的最优动作序列. 不同类型的博弈参与者会形成不同的博弈求解均衡. 自动驾驶中的交通参与者通常具有不同的驾驶风格和特征, 比如在换道汇入场景中车辆选择抢行还是让行. 由于不同参与者的意图和风格未知, 可能的博弈求解类型包括纳什均衡、相关均衡、Stackelberg 均衡以及合作均衡等. 一些早期的工作提出了基于交互信息预测对手的博弈求解类型, 结合 Level-K 递归推理各交通参与者最优动作序列, 可以得到自车的最优决策 [150]. 近年来, Schwarting 等 [151] 首次在自动驾驶问题中引入 SVO, 用来建模不同交通参与者的协作程度. 通过历史交互轨迹使用最大熵似然估计每辆车的 SVO 值, 进而基于迭代最优反馈计算最优决策动作. 通过引入这种协作倾向建模, 可有效提升自动驾驶车辆在路口等博弈场

景下的交互性能.

6 博弈智能的挑战

博弈智能在合作、对抗以及混合等多种形式下已经取得了令人瞩目的成果, 然而在研究以及应用方面, 仍然面临如下挑战.

复杂任务训练慢. 在博弈智能中, 合作、对抗、混合博弈场景的求解和优化目标均有所不同, 且由于参与者的联合观测和动作空间相对于参与者数量呈指数级增长, 最优解空间维度会非常庞大, 从而导致优化缓慢. 目前大多数已有方案 (例如独立学习方法 [32]、基于协作图的学习方法 [13]、集中式训练分布式执行方法 [35]) 都是假设高维寻优空间中存在某些低维结构, 进而通过降维等方式实现低维寻优. 然而, 已有方法通常假设状态特征维度低, 或者智能体关系结构固定 (例如独立学习方法在任何时刻参与者之间都独立 [32]; 基于协作图的学习方法在整个寻优过程中协作图保持不变 [13]), 这显然很难准确建模真实世界的复杂交互关系. 因此, 一种潜在的未来方向是, 研究自适应动态低维结构发现方法. 例如可以根据不同场景的时空信息, 将参与者之间的目标关系识别出来, 自适应地实现参与者间的高效交互, 从而达成更好的合作.

行为策略迁移难. 在实际的博弈场景中, 往往不能观测到全局信息, 需要有效整合各个博弈参与方的局部信息. 然而, 现有博弈智能方法专注于优化策略本身, 缺少合适的参与者行为模型, 从而导致策略的优化效率、泛化能力均有所限制. 在合作博弈方面, 已有参与者行为建模方法主要依靠在对共享奖励进行信誉分配的同时, 引入对其他参与者行为的辅助预测任务. 这种方式需要对共享的全局奖励有精准的定义 [80,81,161], 并且建立的参与者行为模型很难迁移到其他任务中. 在对抗博弈方面, 已有参与者行为建模方法主要依靠迭代最优反馈的方式, 通过树结构搜索、价值函数迭代等形式隐式或显式地构建对手策略模型 [111,112], 然而模型往往缺乏风格多样性, 依赖一个元策略库从多目标、多角度建模对手策略. 在混合博弈方面, 已有参与者行为建模方法主要依靠对其他参与者意图的识别 [143,144], 通过对已有行为序列的观察, 估计参与者之间的关系, 然而现实场景中的意图通常不是一成不变的, 需要从固定一段已有行为序列中精准判断其他参与者一段事件内的意图变化. 综上所述, 如何构建更加灵活通用、便于定义的参与者行为模型, 如何从多目标、多角度分析智能体周边参与者的可能策略, 如何在复杂多变的实际场景中精准预测其他参与者的潜在意图, 都是亟待解决的技术问题.

基线构建不充分. 构建合适的基线环境对于提升现有博弈智能的鲁棒性至关重要. 针对合作博弈, 已有 SMAC, MPE 等主流的基线环境. 尤其是基于 SMAC 的合作博弈算法研究, 极大促进了该领域的发展. 合作博弈领域所使用的主流环境在近几年来的广泛研究下, 产生了众多高质量解决方案, 可解决绝大多数 MPE 环境的问题, 或是在几乎所有的 SMAC 地图中获得满胜率. 然而, 上述环境的仿真速度依然较慢, 并且从最优解探索的角度看, 需要更具多样性、挑战性的基线环境来激发合作博弈的进一步研究. 针对对抗博弈, 目前主要研究以 OpenSpiel [162] 为代表的棋牌类环境, 这类环境包含德州扑克、围棋、麻将等任务. 针对混合博弈, 目前主要研究以 Melting Pot [135] 为代表的社会困境环境, 其中的 Harvest 和 Cleanup 两个经典场景在相关学术成果中较多提及. 相比合作博弈, 对抗博弈和混合博弈目前还没有广泛使用的基线环境, 这也阻碍了相关领域算法的公平比较. 因此, 立足于游戏 AI、黑盒优化等核心业务需求, 构造更广泛的基线环境、真实数据集以及更丰富的基线算法, 并制定合理的评判标准和对比过程, 是今后博弈智能的另一研究重点.

推理架构待优化. 目前主流的多智能体强化学习算法都是基于奖励函数进行合作策略学习的. 然

而, 奖励函数往往只能表达期望的最终结果, 但却很难直接指导多个智能体之间形成直接的合作或者竞争关系, 这也造成算法训练效率低下. 一种潜在的解决方案是, 在多智能体系统中引入认知学习 [163, 164], 让智能体不仅对环境有认知 (例如知道环境此时的难易程度), 还要对其他智能体有认知 (例如知道不同智能体的合作意愿、竞争强弱), 进而实现高层次语义级别的认知一致、价值对齐, 这样能够提高多智能体系统训练效率、多智能体合作性能.

7 总结与展望

本文围绕着博弈智能这一前沿交叉领域, 总结了博弈智能的相关技术背景, 涵盖马尔可夫决策过程、基于博弈论的交互建模等多智能体建模技术, 以及强化学习、博弈学习等多智能体求解方案. 本文按照博弈问题中智能体之间的不同关系, 依次介绍了合作博弈、对抗博弈、混合博弈这三大范式的博弈智能技术的发展现状与典型应用, 并总结了目前研究的成果与不足.

博弈智能是一个具有重大前景的研究方向, 在现实生活中有着广泛的应用. 该领域的研究发展有助于理解群体认知规律和运行规则, 也有助于实现更接近人类水平或超越人类水平的群体智能. 然而, 现有博弈智能取得的成果是初步的, 在解决真实世界的复杂问题时, 仍面临着复杂任务训练慢、行为策略迁移难、基线构建不充分、推理架构待优化等多方面的挑战. 未来的博弈智能研究将带来很多崭新的研究方向.

有模型博弈智能. 基于模型的方法通过学习博弈过程的动力学模型或多智能体交互模型, 可以提升训练样本的利用率, 实现博弈过程的高效推演.

层次化博弈智能. 通过对复杂的多智能体决策问题进行逐级解耦, 或引入专家先验进行层次划分, 有效降低多智能体决策的计算复杂度.

元学习博弈智能. 从模型参数初始化或上下文学习角度, 元学习技术有助于实现大规模场景下博弈智能的迁移泛化.

表征博弈智能. 通过结合基础模型的多模态和多任务学习技术, 实现博弈智能在高维复杂场景下的通用表征学习.

鲁棒博弈智能. 真实问题往往存在大量的鲁棒安全性要求, 通过约束带入或者严格惩罚等方式发展鲁棒安全的博弈智能技术对于真实应用至关重要.

离线博弈智能. 现有博弈智能技术需要在仿真环境中不断演练推理, 然而构建和真实应用匹配的仿真环境非常困难. 发展基于离线数据的博弈智能技术有助于降低对仿真环境的依赖.

因果博弈智能. 通过构建因果推理模型, 识别博弈智能中不同概念主体的高阶联系, 进一步提升博弈模型的鲁棒性.

认知博弈智能. 认知博弈智能的发展离不开研究人机博弈、意识理论等前沿智能技术, 这也是决策智能上升到认知智能的关键. 本文希望通过相关介绍, 对该领域研究提供一些帮助, 引导研究人员继续加大力度去解决博弈智能的基础问题, 实现博弈智能理论研究和应用实践的协同发展, 推动博弈智能技术的真实落地, 更好地让博弈智能技术赋能人类社会.

参考文献

1 Morgenstern O, von Neumann J. Theory of Games and Economic Behavior. Princeton: Princeton University Press, 1953
2 Osborne M J, Rubinstein A. A Course in Game Theory. Cambridge: The MIT Press, 1994
3 Weiss G. Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. Cambridge: The MIT Press, 1999
4 Silver D, Huang A, Maddison C J, et al. Mastering the game of Go with deep neural networks and tree search. Nature, 2016, 529: 484–489
5 Silver D, Schrittwieser J, Simonyan K, et al. Mastering the game of Go without human knowledge. Nature, 2017, 550: 354–359
6 Vinyals O, Babuschkin I, Czarnecki W M, et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. Nature, 2019, 575: 350–354
7 Berner C, Brockman G, Chan B, et al. Dota2 with large scale deep reinforcement learning. 2019. ArXiv:1912.06680
8 McMahan H B, Gordon G J, Blum A. Planning in the presence of cost functions controlled by an adversary. In: Proceedings of the 20th International Conference on Machine Learning, 2003. 536–543
9 Sutton R S, Barto A G. Reinforcement Learning: An Introduction. Cambridge: The MIT Press, 1998
10 Lanctot M, Zambaldi V, Gruslys A, et al. A unified game-theoretic approach to multiagent reinforcement learning. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017. 4193–4206
11 Tampuu A, Matiisen T, Kodelja D, et al. Multiagent cooperation and competition with deep reinforcement learning. Plos One, 2017, 12: e0172395
12 Kok J R, Vlassis N. Sparse cooperative Q-learning. In: Proceedings of the 21st International Conference on Machine Learning, 2004. 481–488
13 Böhmer W, Kurin V, Whiteson S. Deep coordination graphs. In: Proceedings of the 37th International Conference on Machine Learning, 2020. 980–991
14 Sunehag P, Lever G, Gruslys A, et al. Value-decomposition networks for cooperative multi-agent learning based on team reward. In: Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems, 2018. 2085–2087
15 Leibo J Z, Zambaldi V, Lanctot M, et al. Multi-agent reinforcement learning in sequential social dilemmas. In: Proceedings of the 16th Conference on Autonomous Agents and Multiagent Systems, 2017. 464–473
16 McKee K R, Gemp I, McWilliams B, et al. Social diversity and social preferences in mixed-motive reinforcement learning. In: Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, 2020. 869–877
17 Adler I. The equivalence of linear programs and zero-sum games. Int J Game Theor, 2013, 42: 165–177
18 Shoham Y, Leyton-Brown K. Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge: Cambridge University Press, 2008
19 Nash J J F. Equilibrium points in n-person games. Proc Natl Acad Sci USA, 1950, 36: 48–49
20 Busoniu L, Babuska R, de Schutter B. A comprehensive survey of multiagent reinforcement learning. IEEE Trans Syst Man Cybern C, 2008, 38: 156–172
21 Littman M L. Markov games as a framework for multi-agent reinforcement learning. In: Proceedings of the 11th International Conference on Machine Learning, 1994. 157–163
22 Watkins C J, Dayan P. Q-learning. Machine Learning, 1992, 8: 279–292
23 Brown G W. Iterative solution of games by fictitious play. Act Anal Prod Allocation, 1951, 13: 374
24 Robinson J. An iterative method of solving a game. Annals of Mathematics, 1951, 54: 296–301
25 Cesa-Bianchi N, Lugosi G. Prediction, Learning, and Games. Cambridge: Cambridge University Press, 2006
26 Freund Y, Schapire R E. Game theory, on-line prediction and boosting. In: Proceedings of the 9th Annual Conference on Computational Learning Theory, 1996. 325–332
27 Kalai A, Vempala S. Efficient algorithms for online decision problems. J Comput Syst Sci, 2005, 71: 291–307
28 Arora S, Hazan E, Kale S. The multiplicative weights update method: a meta-algorithm and applications Theor Comput, 2012, 8: 121–164
29 Littlestone N, Warmuth M K. The weighted majority algorithm. Inf Computation, 1994, 108: 212–261

30 Hart S, Mas-Colell A. A simple adaptive procedure leading to correlated equilibrium. Econometrica, 2000, 68: 1127–1150
31 Littman M L. Value-function reinforcement learning in Markov games. Cogn Syst Res, 2001, 2: 55–66
32 de Witt C S, Gupta T, Makoviichuk D, et al. Is independent learning all you need in the starcraft multi-agent challenge? 2020. ArXiv:2011.09533
33 Wen M, Kuba J G, Lin R, et al. Multi-agent reinforcement learning is a sequence modeling problem. In: Proceedings of the 35th International Conference on Neural Information Processing Systems, 2022
34 Kok J R, Vlassis N. Collaborative multiagent reinforcement learning by payoff propagation. J Machine Learning Res, 2006, 7: 1789–1828
35 Rashid T, Samvelyan M, Schroeder C, et al. QMIX: monotonic value function factorisation for deep multi-agent reinforcement learning. In: Proceedings of the 35th International Conference on Machine Learning, 2018. 4295–4304
36 Lowe R, Wu Y, Tamar A, et al. Multi-agent actor-critic for mixed cooperative-competitive environments. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017. 6382–6393
37 Mao H, Zhang Z, Xiao Z, et al. Modelling the dynamic joint policy of teammates with attention multi-agent DDPG. In: Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems, 2019. 1108–1116
38 Foerster J, Assael I A, de Freitas N, et al. Learning to communicate with deep multi-agent reinforcement learning. In: Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016. 2145–2153
39 Sukhbaatar S, Szlam A, Fergus R. Learning multiagent communication with backpropagation. In: Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016. 2145–2153
40 Singh A, Jain T, Sukhbaatar S. Learning when to communicate at scale in multiagent cooperative and competitive tasks. In: Proceedings of the 7th International Conference on Learning Representations, 2019
41 Mao H, Zhang Z, Xiao Z, et al. Learning agent communication under limited bandwidth by message pruning. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020. 5142–5149
42 Tan M. Multi-agent reinforcement learning: independent vs. cooperative agents. In: Proceedings of the 10th International Conference on Machine Learning, 1993. 330–337
43 Tumer K, Agogino A. Distributed agent-based air traffic flow management. In: Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems, 2007. 1–8
44 Foerster J, Farquhar G, Afouras T, et al. Counterfactual multi-agent policy gradients. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 2018. 2974–2982
45 Yang Y, Hao J, Chen G, et al. Q-value path decomposition for deep multiagent reinforcement learning. In: Proceedings of the 37th International Conference on Machine Learning, 2020. 10706–10715
46 Yang Y, Hao J, Liao B, et al. Qatten: a general framework for cooperative multiagent reinforcement learning. 2020. ArXiv:2002.03939
47 Zhou M, Liu Z, Sui P, et al. Learning implicit credit assignment for cooperative multi-agent reinforcement learning. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020. 11853–11864
48 Wang J, Zhang Y, Kim T K, et al. Shapley Q-value: a local reward approach to solve global reward games. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020. 7285–7292
49 Li J, Kuang K, Wang B, et al. Shapley counterfactual credits for multi-agent reinforcement learning. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021. 934–942
50 Li W, Wang X, Jin B, et al. Structured diversification emergence via reinforced organization control and hierachical consensus learning. In: Proceedings of the 20th International Conference on Autonomous Agents and Multiagent Systems, 2021. 773–781
51 Zhou T, Zhang F, Tang P, et al. BGC: multi-agent group belief with graph clustering. In: Proceedings of the 3rd International Conference on Distributed Artificial Intelligence, 2021. 52–63
52 Mao H, Wang C, Hao X, et al. SEIHAI: a sample-efficient hierarchical AI for the MineRL competition. In: Proceedings of the 3rd International Conference on Distributed Artificial Intelligence, 2021. 38–51
53 Yang Y, Luo R, Li M, et al. Mean field multi-agent reinforcement learning. In: Proceedings of the 35th International Conference on Machine Learning, 2018. 5571–5580
54 Subramanian S G, Poupart P, Taylor M E, et al. Multi type mean field reinforcement learning. In: Proceedings of

the 19th International Conference on Autonomous Agents and Multiagent Systems, 2020. 411–419
55 Wang W, Yang T, Liu Y, et al. Action semantics network: considering the effects of actions in multiagent systems. In: Proceedings of the 7th International Conference on Learning Representations, 2019
56 Wang W, Yang T, Liu Y, et al. From few to more: large-scale dynamic multiagent curriculum learning. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020. 7293–7300
57 Liu Y, Wang W, Hu Y, et al. Multi-agent game abstraction via graph attention neural network. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020. 7211–7218
58 Jianye H, Hao X, Mao H, et al. Boosting multiagent reinforcement learning via permutation invariant and permutation equivariant networks. In: Proceedings of the 11th International Conference on Learning Representations, 2023
59 Xue K, Xu J, Yuan L, et al. Multi-agent dynamic algorithm configuration. In: Proceedings of the 36th International Conference on Neural Information Processing Systems, 2022
60 Mathesen L, Pedrielli G, Smith R L. Scaling Bayesian optimization with game theory. 2021. ArXiv:2110.03790
61 Mao H, Liu W, Hao J, et al. Neighborhood cognition consistent multi-agent reinforcement learning. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020. 7219–7226
62 Zhao X, Xia L, Zou L, et al. Whole-chain recommendations. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, 2020. 1883–1891
63 Wen C, Xu M, Zhang Z, et al. A cooperative-competitive multi-agent framework for auto-bidding in online advertising. In: Proceedings of the 15th ACM International Conference on Web Search and Data Mining, 2022. 1129–1139
64 Samvelyan M, Rashid T, de Witt C S, et al. The starcraft multi-agent challenge. In: Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems, 2019. 2186–2188
65 Schneider J, Wong W K, Moore A, et al. Distributed value functions. In: Proceedings of the 16th International Conference on Machine Learning, 1999. 371–378
66 Zhang K, Yang Z, Liu H, et al. Fully decentralized multi-agent reinforcement learning with networked agents. In: Proceedings of the 35th International Conference on Machine Learning, 2018. 5872–5881
67 Guestrin C, Koller D, Parr R. Multiagent planning with factored MDPs. In: Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic, 2001. 1523–1530
68 Guestrin C, Lagoudakis M G, Parr R. Coordinated reinforcement learning. In: Proceedings of the 19th International Conference on Machine Learning, 2002. 227–234
69 Kok J R, Vlassis N. Using the max-plus algorithm for multiagent decision making in coordination graphs. In: Proceedings of Robot Soccer World Cup, 2005. 1–12
70 Son K, Kim D, Kang W J, et al. QTRAN: learning to factorize with transformation for cooperative multi-agent reinforcement learning. In: Proceedings of the 36th International Conference on Machine Learning, 2019. 5887–5896
71 Zhao J, Hu X, Yang M, et al. CTDS: centralized teacher with decentralized student for multi-agent reinforcement learning. IEEE Trans Games, 2022. doi: 10.1109/TG.2022.3232390
72 Chen Y, Mao H, Zhang T, et al. PTDE: personalized training with distillated execution for multi-agent reinforcement learning. 2022. ArXiv:2210.08872
73 Wang H, Yu Y, Jiang Y. Review of the progress of communication-based multi-agent reinforcement learning. Sci Sin Inform, 2022, 52: 742–764 [王涵, 俞扬, 姜远. 基于通信的多智能体强化学习进展综述. 中国科学: 信息科学, 2022, 52: 742–764]
74 Sun J C, Wang J L, Chen J, et al. Cooperative communication based on swarm intelligence: vision, model, and key technology. Sci Sin Inform, 2020, 50: 307–317 [孙佳琛, 王金龙, 陈瑾, 等. 群体智能协同通信: 愿景, 模型和关键技术. 中国科学: 信息科学, 2020, 50: 307–317]
75 Mao H, Gong Z, Ni Y, et al. ACCNET: actor-coordinator-critic net for "learning-to-communicate" with deep multi-agent reinforcement learning. 2017. ArXiv:1706.03235
76 Mao H, Zhang Z, Xiao Z, et al. Learning multi-agent communication with double attentional deep reinforcement learning. Auton Agent Multi-Agent Syst, 2020, 34: 32
77 Jiang J, Lu Z. Learning attentional communication for multi-agent cooperation. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems, 2018. 7265–7275
78 Niu Y, Paleja R, Gombolay M. Multi-agent graph-attention communication and teaming. In: Proceedings of the 20th International Conference on Autonomous Agents and Multiagent Systems, 2021. 964–973

79 Das A, Gervet T, Romoff J, et al. TarMAC: targeted multi-agent communication. In: Proceedings of the 36th International Conference on Machine Learning, 2019. 1538–1546

80 Agogino A, Turner K. Multi-agent reward analysis for learning in noisy domains. In: Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems, 2005. 81–88

81 Mao H, Gong Z, Xiao Z. Reward design in cooperative multi-agent reinforcement learning for packet routing. 2020. ArXiv:2003.03433

82 Mao H, Hao J Y, Li D, et al. Learning explicit credit assignment for multi-agent joint Q-learning. Openreview, 2021

83 Shapley L. A value for n-person games, contributions to the theory of games. In: Proceedings of Classics in Game Theory, 2020. 69–79

84 Zhang T, Liu Z, Pu Z, et al. Hierarchical cooperative swarm policy learning with role emergence. In: Proceedings of the IEEE Symposium Series on Computational Intelligence, 2021. 1–8

85 Guo X, Hu A, Xu R, et al. Learning mean-field games. In: Proceedings of the 33rd International Conference on Neural Information Processing Systems, 2019. 32

86 Yang F, Vereshchaka A, Chen C, et al. Bayesian multi-type mean field multi-agent imitation learning. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020. 33: 2469–2478

87 Shao K, Tang Z, Zhu Y, et al. A survey of deep reinforcement learning in video games. 2019. ArXiv:1912.10944

88 Xing M, Mao H, Xiao Z. Fast and fine-grained autoscaler for streaming jobs with reinforcement learning. In: Proceedings of the 31st International Joint Conference on Artificial Intelligence, 2022. 564–570

89 Xi L, Chen J F, Huang Y H, et al. Smart generation control based on deep reinforcement learning with the ability of action self-optimization. Sci Sin Inform, 2018, 48: 1430–1449 [席磊, 陈建峰, 黄悦华, 等. 基于具有动作自寻优能力的深度强化学习的智能发电控制. 中国科学: 信息科学, 2018, 48: 1430–1449]

90 Duan H B, Zhang D F, Fan Y M, et al. From wolf pack intelligence to UAV swarm cooperative decision-making. Sci Sin Inform, 2019, 49: 112–118 [段海滨, 张岱峰, 范彦铭, 等. 从狼群智能到无人机集群协同决策. 中国科学: 信息科学, 2019, 49: 112–118]

91 Huang K Q, Xing J L, Zhang J G, et al. Intelligent technologies of human-computer gaming. Sci Sin Inform, 2020, 50: 540–550 [黄凯奇, 兴军亮, 张俊格, 等. 人机对抗智能技术. 中国科学: 信息科学, 2020, 50: 540–550]

92 Ganzfried S, Sandholm T. Safe opponent exploitation. ACM Trans Econ Comput, 2015, 3: 1–28

93 Maskin E. Commentary: Nash equilibrium and mechanism design. Games Economic Behav, 2011, 71: 9–11

94 Koller D, Megiddo N. The complexity of two-person zero-sum games in extensive form. Games Economic Behav, 1992, 4: 528–552

95 Knuth D E, Moore R W. An analysis of alpha-beta pruning. Artif Intell, 1975, 6: 293–326

96 Coulom R. Efficient selectivity and backup operators in monte-carlo tree search. In: Proceedings of the 5th International Conference on Computers and Games, 2006. 72–83

97 Zinkevich M, Johanson M, Bowling M, et al. Regret minimization in games with incomplete information. In: Proceedings of the 20th International Conference on Neural Information Processing Systems, 2007. 1729–1736

98 Hofbauer J, Sigmund K. Evolutionary game dynamics. Bull Amer Math Soc, 2003, 40: 479–519

99 Jaderberg M, Czarnecki W M, Dunning I, et al. Human-level performance in 3D multiplayer games with population-based reinforcement learning. Science, 2019, 364: 859–865

100 Brown N, Sandholm T. Superhuman AI for multiplayer poker. Science, 2019, 365: 885–890

101 Li J, Koyamada S, Ye Q, et al. Suphx: mastering mahjong with deep reinforcement learning. 2020. ArXiv:2003.13590

102 von Neumann J. Zur theorie der gesellschaftsspiele. Math Ann, 1928, 100: 295–320

103 Shapley L S. Stochastic games. Proc Natl Acad Sci USA, 1953, 39: 1095–1100

104 Koller D, Pfeffer A. Representations and solutions for game-theoretic problems. Artif Intelligence, 1997, 94: 167–215

105 Lanctot M, Waugh K, Zinkevich M, et al. Monte Carlo sampling for regret minimization in extensive games. In: Proceedings of the 22nd International Conference on Neural Information Processing Systems, 2009. 1078–1086

106 Bowling M, Burch N, Johanson M, et al. Heads-up limit hold'em poker is solved. Science, 2015, 347: 145–149

107 Moravčík M, Schmid M, Burch N, et al. DeepStack: expert-level artificial intelligence in heads-up no-limit poker. Science, 2017, 356: 508–513

108 Brown N, Sandholm T. Superhuman AI for heads-up no-limit poker: libratus beats top professionals. Science, 2018, 359: 418–424

109 Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets. In: Proceedings of the 27th International Conference on Neural Information Processing Systems, 2014. 2672–2680

110 Schrittwieser J, Antonoglou I, Hubert T, et al. Mastering Atari, Go, chess and shogi by planning with a learned model. Nature, 2020, 588: 604–609

111 Heinrich J, Lanctot M, Silver D. Fictitious self-play in extensive-form games. In: Proceedings of the 32nd International Conference on Machine Learning, 2015. 805–813

112 Heinrich J, Silver D. Deep reinforcement learning from self-play in imperfect-information games. 2016. ArXiv:1603.01121

113 Hennes D, Morrill D, Omidshafiei S, et al. Neural replicator dynamics: multiagent learning via hedging policy gradients. In: Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, 2020. 492–501

114 Perolat J, de Vylder B, Hennes D, et al. Mastering the game of Stratego with model-free multiagent reinforcement learning. Science, 2022, 378: 990–996

115 Wellman M P. Methods for empirical game-theoretic analysis. In: Proceedings of the 21st National Conference on Artificial Intelligence, 2006. 1552–1555

116 Balduzzi D, Garnelo M, Bachrach Y, et al. Open-ended learning in symmetric zero-sum games. In: Proceedings of the 36th International Conference on Machine Learning, 2019. 434–443

117 McAleer S, Lanier J B, Fox R, et al. Pipeline PSRO: a scalable approach for finding approximate Nash equilibria in large games. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020. 33: 20238–20248

118 Wu B. Hierarchical macro strategy model for MOBA game AI. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence, 2019. 1206–1213

119 Ye D, Chen G, Zhang W, et al. Towards playing full MOBA games with deep reinforcement learning. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020. 33: 621–632

120 Zha D, Xie J, Ma W, et al. DouZero: mastering DouDizhu with self-play deep reinforcement learning. In: Proceedings of the 38th International Conference on Machine Learning, 2021. 12333–12344

121 Chen X, Deng X, Teng S H. Settling the complexity of computing two-player Nash equilibria. J ACM, 2009, 56: 1–57

122 Daskalakis C, Goldberg P W, Papadimitriou C H. The complexity of computing a Nash equilibrium. SIAM J Comput, 2009, 39: 195–259

123 Anderson H S, Kharkar A, Filar B, et al. Evading machine learning malware detection. Black Hat, 2017, 2017: 1–6

124 Ernest N, Carroll D, Schumacher C, et al. Genetic fuzzy based artificial intelligence for unmanned combat aerial vehicle control in simulated air combat missions. J Def Manag, 2016, 6: 144

125 Clark B, Patt D, Schramm H. Mosaic warfare: exploiting artificial intelligence and autonomous systems to implement decision-centric operations. Center for Strategic and Budgetary Assessments, 2020.

126 Ding W, Chen B, Xu M, et al. Learning to collide: an adaptive safety-critical scenarios generating method. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2020. 2243–2250

127 Wang J, Pun A, Tu J, et al. AdvSim: generating safety-critical scenarios for self-driving vehicles. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021. 9909–9918

128 Wald A. Contributions to the theory of statistical estimation and testing hypotheses. Ann Math Statist, 1939, 10: 299–326

129 Lanckriet G R, Ghaoui L E, Bhattacharyya C, et al. A robust minimax approach to classification. J Machine Learning Res, 2002, 3: 555–582

130 Moos J, Hansel K, Abdulsamad H, et al. Robust reinforcement learning: a review of foundations and recent advances. Machine Learning Knowledge Extraction, 2022, 4: 276–315

131 Kraines D, Kraines V. The threshold of cooperation among adaptive agents: Pavlov and the stag hunt. In: Intelligent Agents III Agent Theories, Architectures, and Languages. Berlin: Springer, 2001. 3: 219–232

132 Hu J, Wellman M P. Nash Q-learning for general-sum stochastic games. J Machine Learning Res, 2003, 4: 1039–1069

133 Greenwald A, Hall K, Serrano R, et al. Correlated Q-learning. In: Proceedings of the 20th International Conference on Machine Learning, 2003. 242–249

134 Littman M L. Friend-or-Foe Q-learning in general-sum games. In: Proceedings of the 18th International Conference

on Machine Learning, 2001. 322–328
135 Leibo J Z, Dueñez-Guzman E A, Vezhnevets A, et al. Scalable evaluation of multi-agent reinforcement learning with melting pot. In: Proceedings of the 38th International Conference on Machine Learning, 2021. 6187–6199
136 Hughes E, Leibo J, Phillips M, et al. Inequity aversion improves cooperation in intertemporal social dilemmas. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems, 2018. 3330–3340
137 Anastassacos N, Hailes S, Musolesi M. Partner selection for the emergence of cooperation in multi-agent systems using reinforcement learning. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020. 7047–7054
138 Merhej R, Santos F P, Melo F S, et al. Cooperation between independent reinforcement learners under wealth inequality and collective risks. In: Proceedings of the 20th International Conference on Autonomous Agents and Multiagent Systems, 2021. 898–906
139 Baker B. Emergent reciprocity and team formation from randomized uncertain social preferences. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020. 15786–15799
140 Anastassacos N, García J, Hailes S, et al. Cooperation and reputation dynamics with reinforcement learning. In: Proceedings of the 20th International Conference on Autonomous Agents and Multiagent Systems, 2021. 115–123
141 Köster R, Hadfield-Menell D, Everett R, et al. Spurious normativity enhances learning of compliance and enforcement behavior in artificial agents. Proc Natl Acad Sci USA, 2022, 119: e2106028118
142 Yang J, Li A, Farajtabar M, et al. Learning to incentivize other learning agents. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020. 15208–15219
143 Raileanu R, Denton E, Szlam A, et al. Modeling others using oneself in multi-agent reinforcement learning. In: Proceedings of the 35th International Conference on Machine Learning, 2018. 4257–4266
144 Liu S, Lever G, Merel J, et al. Emergent coordination through competition. In: Proceedings of the 7th International Conference on Learning Representations, 2019
145 Foerster J, Chen R Y, Al-Shedivat M, et al. Learning with opponent-learning awareness. In: Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems, 2018. 122–130
146 Yu X, Jiang J, Zhang W, et al. Model-based opponent modeling. In: Proceedings of the 36th International Conference on Neural Information Processing Systems, 2022
147 Pretorius A, Cameron S, van Biljon E, et al. A game-theoretic analysis of networked system control for common-pool resource management using multi-agent reinforcement learning. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020. 9983–9994
148 Hostallero D E, Kim D, Moon S, et al. Inducing cooperation through reward reshaping based on peer evaluations in deep multi-agent reinforcement learning. In: Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, 2020. 520–528
149 Cui B, Hu H, Pineda L, et al. K-level reasoning for zero-shot coordination in Hanabi. In: Proceedings of the 35th International Conference on Neural Information Processing Systems, 2021. 8215–8228
150 Ji K, Orsag M, Han K. Lane-merging strategy for a self-driving car in dense traffic using the Stackelberg game approach. Electronics, 2021, 10: 894
151 Schwarting W, Pierson A, Alonso-Mora J, et al. Social behavior for autonomous vehicles. Proc Natl Acad Sci USA, 2019, 116: 24972–24978
152 Hilbe C, Šimsa Š, Chatterjee K, et al. Evolution of cooperation in stochastic games. Nature, 2018, 559: 246–249
153 Dai Z, Zhou T, Shao K, et al. Socially-attentive policy optimization in multi-agent self-driving system. In: Proceedings of the 6th Annual Conference on Robot Learning, 2022
154 Koster R, Balaguer J, Tacchetti A, et al. Human-centred mechanism design with Democratic AI. Nat Hum Behav, 2022, 6: 1398–1407
155 Hauser O P, Hilbe C, Chatterjee K, et al. Social dilemmas among unequals. Nature, 2019, 572: 524–527
156 Barfuss W, Donges J F, Vasconcelos V V, et al. Caring for the future can turn tragedy into comedy for long-term collective action under risk of collapse. Proc Natl Acad Sci USA, 2020, 117: 12915–12922
157 Zhou M, Luo J, Villella J, et al. SMARTS: scalable multi-agent reinforcement learning training school for autonomous driving. 2020. ArXiv:2010.09776
158 Vinitsky E, Lichtlé N, Yang X, et al. Nocturne: a scalable driving benchmark for bringing multi-agent learning one step closer to the real world. In: Proceedings of the 36th Conference on Neural Information Processing Systems

Datasets and Benchmarks Track, 2022
159 Ma N, Li D Y, He W, et al. Future vehicles: interactive wheeled robots. Sci China Inf Sci, 2021, 64: 156101
160 Bae S H, Joo S H, Pyo J W, et al. Finite state machine based vehicle system for autonomous driving in urban environments. In: Proceedings of the 20th International Conference on Control, Automation and Systems (ICCAS), 2020. 1181–1186
161 Zhang X, Liu Y, Xu X, et al. Structural relational inference actor-critic for multi-agent reinforcement learning. Neurocomputing, 2021, 459: 383–394
162 Lanctot M, Lockhart E, Lespiau J B, et al. OpenSpiel: a framework for reinforcement learning in games. 2019. ArXiv:1908.09453
163 Juliani A, Arulkumaran K, Sasai S, et al. On the link between conscious function and general intelligence in humans and machines. 2022. arXiv:2204.05133
164 Yuan L, Gao X, Zheng Z, et al. In situ bidirectional human-robot value alignment. Sci Robot, 2022, 7: eabm4183

Research and applications of game intelligence

Jianye HAO[1,2], Kun SHAO[2], Kai LI[2], Dong LI[2], Hangyu MAO[2], Shuyue HU[3] & Zhen WANG[4,5*]

1. *College of Intelligence and Computing, Tianjin University, Tianjin* 300350*, China*;
2. *Huawei Noah's Ark Lab, Beijing* 100085*, China*;
3. *Shanghai Artificial Intelligence Laboratory, Shanghai* 200232*, China*;
4. *School of Cybersecurity, Northwestern Polytechnical University, Xi'an* 710072*, China*;
5. *School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an* 710072*, China*
* Corresponding author. E-mail: w-zhen@nwpu.edu.cn

Abstract Game intelligence is a cross-disciplinary field that encompasses game theory and artificial intelligence. It focuses on interactions between individuals or organizations and the accurate solution of optimal strategies through quantifying game relationships with modeling, finally forming intelligent decision-making and its knowledge base. In recent years, with the explosion of massive behavioral data and the diversification of game forms, game intelligence has increasingly attracted the interest of researchers and has been used widely in real life. This paper systematically surveys game intelligence in three aspects. First, it reviews the relevant background of game intelligence, including single-agent Markov decision processes, multiagent modeling techniques based on game theory, and multiagent solution methods such as reinforcement learning and game learning. Second, based on the different game relationships between intelligent agents, this paper categorizes games into three paradigms: cooperative games, adversarial games, and mixed games, and introduces the main research problems, the mainstream research methods, and typical applications in each game intelligence paradigm. Finally, this paper summarizes the current research status of game intelligence, the main problems and research challenges that need to be addressed, and the prospects for application in academia and industry, providing a reference for related research and further promoting the development of the national artificial intelligence strategy.

Keywords game intelligence, game theory, artificial intelligence, multiagent systems, reinforcement learning, equilibrium computing