目录

1 分布式人工智能简介 1

1.1 研究历史 1

1.1.1 前深度学习时代 1

1.1.2 深度学习时代 3

1.2 主要研究领域 5

1.2.1 算法博弈论 5

1.2.2 分布式问题求解 6

1.2.3 多智能体规划 7

1.2.4 多智能体学习 8

1.2.5 分布式机器学习 9

1.3 相关研究应用 11

1.3.1 足球 11

1.3.2 安全博弈 12

1.3.3 扑克和麻将 12

1.3.4 视频游戏 14

1.4 未来研究挑战 14

1.4.1 超大规模分布式人工智能系统 14

1.4.2 分布式人工智能系统的鲁棒性和安全性 15

1.4.3 分布式人工智能决策的可解释性 16

1.4.4 传统和深度学习的方法结合 16

参考文献 17

1

分布式人工智能简介

1.1 研究历史

我们首先将对分布式人工智能的历史进行扼要的回顾。分布式人工智能初创于上世纪七十年代，在过去发展的五十多年的过程中，可大致分为前深度学习时代和深度学习时代。前深度学习时代的分布式人工智能研究主要采用传统的优化算法，而在深度学习时代人们利用深度神经网络的强大表示能力来解决更大规模和更复杂的问题。

1.1.1 前深度学习时代

继 1956 年约翰 • 麦卡锡（John McCarthy）在著名的达特茅斯研讨会上提出“人工智能”（Artificial Intelligence）这一概念后，智能体（Intelligent agent）领域便开始兴起。“智能体”这一概念在早期的人工智能文献中就已经出现，例如阿兰 · 图灵（Alan Turing）提出用来判断一台机器是否具备人类的智能的图灵测试。在这个测试中，测试者通过一些监视设备向被测试的实体（也就是我们现在所说的智能体）提问。如果测试者不能区分被测试对象是计算机程序还是人，那么根据图灵的观点，被测试的对象就被认为是智能的。尽管智能体的概念很早就出现，将多个智能体作为一个功能上的整体（即能够独立行动的自主集成系统）加以研究的思路在上世纪 70 年代之前都发展甚微。这期间，一些研究着眼于构建一个完整的智能体或多智能体系统，其中比较有影响的有 Hearsay-II 语音理解系统 [22]、STRIPS 规划系统 [23]、Actor 模型 [33] 等。1978 年 12 月，DARPA 分布式传感器网络研讨会作为最早致力于多智能体系统问题的研讨会在卡耐基梅隆大学（CMU）举办。1980 年 6 月 9-11 日，分布式人工智能这

个新领域的首次研讨会在麻省理工学院（MIT）举办。该研讨会持续了大约 15 年，在此期间，欧洲和远东的分布式人工智能研讨会也随着研究队伍的壮大而逐步开展。在 1980 年这次会议上，研究人员开始谈论分布式问题求解、多智能体规划、组织控制、合同网、协商、分布式传感器网络、功能精确的协作分布式系统、大规模行为者模型，以及智能体规范逻辑框架等重要的多智能体系统研究问题。在此之后，集成智能体构建和多智能体系统研究的各个分支领域都有较大的发展。分布式人工智能领域自诞生之际起就海纳百川，包罗各个学科，直到今天依然是跨学科交叉的沃土。然而，恰如其名，分布式人工智能和人工智能领域之间始终维系着最强的纽带。

随着 80 年代后期《Distributed Artificial Intelligence》和《Readings in Distributed Artificial Intelligence》的出版 [9, 34]，重要的文献有了集中的去处，分布式人工智能领域开始联合并显著地扩张。在这段成长的时期，建立在博弈论和经济学概念上的自私智能体交互的研究逐步兴盛起来 [71]。随着协作型和自私型智能体研究的交融，分布式人工智能逐渐演变并最终采用了一个更广阔、更包罗的新名字—“多智能体系统”。

上世纪 90 年代初期，得益于同期互联网的迅猛扩张，多智能体系统研究快速发展。此时互联网已从一个学术圈外鲜为人知的事物演变成全球商业和休闲不可或缺的日常工具。如果说计算的未来在于分布式的、联网的系统，人们亟需新的计算模型去挖掘这些分布式系统的潜能，那么互联网的爆炸式增长或许就是最好的说明。毫无疑问，互联网将成为多智能体技术的主要试验场之一。基于和用于互联网的智能体的出现催生了新的相关软件技术。人们提出了面向智能体的编程范式（Agent-oriented programming，AOP），将软件的构建集中在软件智能体的概念上。相比面向对象的编程以对象（通过变量参数提供方法）为核心的理念，AOP 以外部指定的智能体（带有接口和消息传递功能）为核心。很快，一批移动智能体框架以 Java 包的形式被开发和发布。

90 年代中期，工业界对智能体系统的兴趣集中在标准化这一焦点上。一些研究人员开始认为智能体技术可能会因为公认的国际标准的缺失而难以得到广泛的认可。90 年代初开发的两种颇具影响力的智能体通信语言——KQML 和 KIF——就一直没有被正式地标准化。因此，FIPA 运动在 1995 年开始了标准化多智能体的工作，它的核心是一套用于智能体协作的语言。90 年代后期，随着互联网的继续发力，电子商务应运而生，“.com”公司席卷全球。很快，人们意识到电子商务代表着一个自然且利益丰厚的多智能体系统应用领域，这其中的基本想法是智能体驱动着电子商务中从产品搜寻到实际的商谈的各个环节。到世纪之交，以智能体为媒介的电子商务或许已经成为智能体技术的最大单一应用场合，为智能体系统在谈判和拍卖领域的发展提供了商业和科学上的巨大推动力。2000 年以后，以促进和鼓励高质量的交易智能体研究为目的的交易智能体竞赛（Trading Agent Competition）得以推出并吸引了很多研究人员的参

与。到 90 年代后期，研究人员开始在不断拓宽的现实领域寻求多智能体系统的新发展。机器人世界杯（RoboCup）应运而生。RoboCup 的目的很清晰，即证明在五十年之内，一支机器人足球队能够战胜世界杯水准的人类球队，其基本理论在于一支出色的球队需要一系列的技能，比如利用有限的带宽进行实时动态的协作。RoboCup 的热度在世纪之交暴涨，定期举办的 RoboCup 锦标赛吸引了来自世界各地的数百支参赛队伍。2000 年，RoboCup 推出了一项新的活动——RoboCup 营救。这项活动以发生在 90 年代中期的日本神户大地震为背景，以建立一支能够通过相互协作完成搜救任务的机器人队伍为目标。

高水平的学术会议对于一个领域的发展至关重要。1995 年，第一届 ICMAS 大会（International Conference on Multi-Agent Systems）在美国旧金山举办。紧接着，ATAL 研讨会（International Workshop on Agent Theories, Architectures, and Languages）和 AA 会议（International Conference on Autonomous Agents）相继举办。国际智能体及多智能体系统协会（International Foundation for Autonomous Agents and Multiagent Systems, IFAAMAS）以促进人工智能、智能体与多智能体系统领域的科技发展为宗旨成立。为了提供一个统一、高质量并广受国际关注的智能体和多智能体系统理论和实践研究论坛，IFAAMAS 在 2002 年将上述三个会议合并为智能体及多智能体系统国际会议（International Joint Conference on Autonomous Agents and Multi-Agent Systems, AAMAS），一般在每年的 5 月举行，在美洲、欧洲、亚洲和大洋洲轮流召开。自那以后，AAMAS 会议成为了多智能体系统和分布式人工智能领域的顶级会议，发表了许多具有开创性意义的文章，极大地促进了这一领域的发展。

1.1.2　深度学习时代

图 1.1　AlphaGo 对战李世石

深度学习自 2006 年提出之后，为学术界和工业界带来了巨大的变革，从计算机视觉领域出发，进而影响到了强化学习和自然语言处理，并在近几年越来越多地用于分布式人工智能领域。深度学习用于分布式人工智能引起广泛关注的是 2016 年来自 DeepMind 的科学家开发的用于求解围棋策略的 AlphaGo [77]，它是第一个击败人类职业围棋选手、第一个击败围棋世界冠军的人工智能算法。基于 AlphaGo，DeepMind 后续开发了 AlphaGo Zero [79], AlphaZero [78] 和 Muzero [74]，极大地推进了该方面的研究。AlphaGo 中采用的算法可以用来很好地求解完全信息博弈下的策略，但不适用于非完全信息博弈。2017 年阿尔伯塔大学开发了 DeepStack 系统，它被用来解决在 2 人无限下注的德州扑克问题，并最终史无前例地击败了职业的扑克玩家 [58]。在此之后，人们的目光转移到了斗地主和麻将上面，借助于深度学习和强化学习的强大能力，依然取得了击败人类职业玩家的良好效果。

深度学习在分布式约束问题 (DCOP) 和组合优化问题的求解问题上也得到了应用。深度学习的优势在于其借助深度神经网络强大的表示能力可以对大规模的复杂问题进行表示，这也是分布式约束问题和组合优化常常需要的。因而在 2014 年，研究者将深度学习用到了分支定界的算法当中，取得了不错的效果 [30]。2015 年来自 Google 的研究者提出了 Pointer Network (Ptr-Net) [89]，这是一类专为解组合优化问题提出的神经网络，作者成功地将其应用在了旅行商问题（TSP）上面。基于 Pointer Netwok 的相关的后续工作包括将强化学习和 Pointer Network 结合解决对于标签的依赖 [6]，利用注意力模型和强化学习解决车辆路径规划问题 [61]，以及使用自然语言处理中的 Transformer 模型提升算法效果 [40]。另一方面，由于图神经网络 (Graph Neural Network (GNN)) 研究的兴起，研究者也开始将其用于解决组合优化问题 [14]。早期的尝试 [18] 包括利用 Structure2Vec GNN 结构结合强化学习算法解决旅行商问题，以及最大割和最小顶点覆盖问题。后续工作包括利用监督学习和 GNN 解决决策版本的旅行商问题 [64]，含有更多约束的问题如带有时间窗口的旅行商问题 [15]。这类方法也被应用于解决布尔可满足性 (SAT) 问题，依然取得了不错的效果 [13, 75]。

不仅在博弈和分布式优化问题上，研究者也把深度学习的技术应用到了分布式人工智能的其他领域。近几年发展尤为迅速的是多智能体强化学习（MARL）领域。2017 年提出的 MADDPG 算法 [50] 提出了 “集中式训练，分布式执行” 的训练模式，很好地在简单问题上解决了不同智能体之间在没有通信的情况下的合作问题。2018 年牛津大学提出了 QMIX 算法 [67]，并且开源了 StarCraft II Multi-Agent Challenge (SMAC) 环境 [72] 作为测试 MARL 算法效果的通用环境。SMAC 是一组基于暴雪公司开发的星际争霸 (StarCraft) 游戏精心设计的智能体之间合作的任务，它可用来测试算法在不同合作模式下的训练效果。自那之后基于 SMAC 的工作出现了很多，受到研究者关注的有 QTRAN [80], QPLEX [90] 和 Weighted QMIX [66]。虽然 SMAC 是一组理想的

测试环境，但是任务本身难度较小，具有一定的局限性，因而 DeepMind 的研究者基于完全版本的星际争霸来训练智能体，最终他们训练的智能体达到了 Grandmaster 水准 [88]。几乎与此同时，OpenAI 的研究者利用深度学习训练了打 Dota 2 游戏的多智能体系统也击败了人类职业玩家 [7]。这些研究被认为是多智能体强化学习在复杂决策问题上可以超过人类的典型例子。

中国学者为分布式人工智能的发展也贡献了自己的力量。2019 年，第一届分布式人工智能国际会议 (International Conference on Distributed Artificial Intelligence (DAI)) 在北京召开。DAI 的主要目标是将相关领域（例如通用人工智能、多智能体系统、分布式学习、计算博弈论）的研究人员和从业者聚集在一起，为促进分布式人工智能的理论和实践发展提供一个高水平的的国际研究论坛。自创办的两三年里，该会议得到了中外许多研究者的关注和参与，其发表的文章也获得了研究者的引用，激发了许多后续的研究，形成了一定的影响力。

1.2 主要研究领域

在这一部分，我们将介绍分布式人工智能的四个代表性的研究领域。

1.2.1 算法博弈论

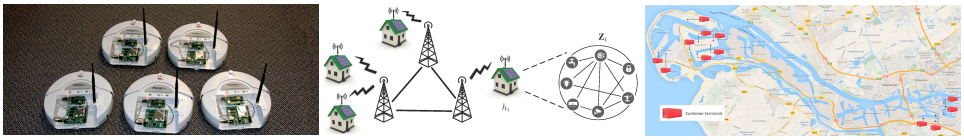
算法博弈论是博弈论和算法设计的交叉领域。算法博弈论问题中算法的输入通常分布在许多参与者上，这些参与者对输出有不同的个人偏好。在这种情况下，这些智能体参与者可能因为个人利益而隐瞒真实的信息。除了经典算法设计理论要求的多项式运算时间及较高近似度外，算法设计者还需要考虑智能体动机的约束。

当前算法博弈论的研究者主要关注在完全信息博弈和非完全信息博弈中的 Nash 均衡的求解。Nash 均衡是指当所有人执行其 Nash 均衡策略是，没有人能够通过单方面改变自己的策略增加自己的收益。在每个参与者的动作都是有限的情况下，混合策略的 Nash 均衡存在于任意博弈中。虽然 Nash 均衡在算法博弈论中应用广泛，但是其求解是非常困难的，对于二人及以上的非零和博弈，其计算难度是 PPAD-complete 的 [19]。幸运的是，对于二人零和博弈，其 Nash 均衡可以在多项式时间内求解。目前研究者关注的 2 人有限下注和无限下注的德州扑克都是属于二人零和博弈。小型的博弈可以通过求解线性规划来得到，但对于大规模博弈其线性规划表示很难求解。因而对于求解大规模二人零和博弈最广泛应用的算法是反事实遗憾最小化算法 (CFR) [100] 和 Fictitious (Self)-Play [31, 44]。其中，CFR 被成功地用于求解二人德州扑克，并决定性地打败了人类职业选手 [11]。在多人零和或者非零和博弈中，研究者利用类似的方

法取得了一些实验结果，但依然缺乏相应的理论支撑。多人博弈目前还是一个 Open 的问题。

进入到深度学习时代以来，上述提到的反事实遗憾最小化算法 (CFR) 和 Fictitious (Self)-Play 也提出了其相应的深度学习版本，也即 Deep CFR [10] 和 Neural Fictitious Self-Play (NFSP) [32]，这类算法成功地应用于大规模的博弈问题的求解，取得了很好的结果 [47, 93, 98]。另一方面，基于经典的 Double Oracle 算法 [35, 54]，DeepMind 的研究者提出了 Policy Space Response Oracle (PSRO) 算法以及其拓展变体，并将一些现有算法统一起来 [42, 59]。基于深度学习的算法具有强大的表示能力，但是其训练相对更为耗时，所以更适用于博弈的表示学习方法是一个值得探索的方向。

1.2.2 分布式问题求解



(a) 信道分配 [97]　(b) 智能家居设备调度 [24]　(c) 港口船舶调度 [46]

图 1.2　真实世界中的分布式问题求解

作为分布式人工智能的一个子领域，分布式问题求解着眼于让多个智能体共同解决一个问题。这些智能体通常都是合作性的。在众多分布式问题求解模型中，分布式约束推理（Distributed Constraint-Reasoning，DCR）模型，如分布式约束满足问题（Distributed Constraint-Satisfaction Problem，DCSP）和分布式约束优化问题（Distributed Constraint-Optimization Problem，DCOP）的使用和研究较为广泛。DCR 模型历史较长，在各种分布式问题上都有应用，包括无线电信道分配和分布式传感器网络。自 90 年代中期以来，DCSP 和 DCOP 算法设计（包括完全的和非完全的）获得了学术界的广泛关注。根据搜索策略（最佳优先搜索或深度优先分支定界搜索）、智能体间同步类型（同步或异步）、智能体间通讯方式（约束图邻居间点对点传播方式或广播方式）以及主要通讯拓扑结构（链式或树形）的不同，可以对学术界提出的众多 DCR 算法进行归类。例如，ADOPT [57] 采用的是最佳优先搜索、异步同步、点对点的通讯和树形通讯拓扑结构。近来，研究人员在多个方面延伸了 DCR 模型和算法，从而更精确地建模和求解现实问题 [97]。隐私保护是使用 DCR 建模诸如分布式会议安排这类问题的一个动机。例如当两名用户没有安排同一个会议的需求时，他们应当无权知悉对方对于会议时间的偏好。不幸的是，隐私泄露常常难以避免。为此，研究人员引入一些指标去衡量 DCR 的隐私泄露程度，并设计新的 DCR 算法来更好地保护隐私。DCR

模型也被扩展为动态 DCR 模型，如动态 DCSP 和动态 DCOP。一个典型的动态 DCR 问题由一连串（静态的）DCR 问题组成，问题与问题之间有一定的变化。其他相关的扩展还包括：智能体以截止时间决定价值的连续时间模型、智能体对所处环境认知不完全的模型。研究人员还将 DCOP 扩展为多目标 DCOP 及资源受限的 DCOP。

由于其出色的表征能力，近年来深度学习被广泛应用于求解大规模约束规划、推理问题，涌现出一大批优秀算法。2018 年来自南加州大学的研究者成功地将卷积神经网络（Convolutional Neural Network (CNN)）应用于表征约束满足问题，并据此提出用于预测可满足性的 CSP-CNN 模型 [92]。在布尔满足问题上，加州大学伯克利分校结合图神经网络通过强化学习得到分支启发信息（Branching Heuristics），极大地提升了现有算法的求解效率 [43]。深度桶消除（Deep Bucket Elimination (DBE)）算法 [68] 则是利用深度神经网络来表示高维效用表，解决了约束推理中指数级的内存消耗问题。类似的想法也应用于求解其他 DCOP 问题 [20]。然而，大多数基于深度学习的算法很难提供理论上的质量保障，使得其在大量关键场景下（例如船舶导航、灾难救援等）无法得到广泛应用。因此，如何更好地将深度学习和符号推理相结合，以开发具有理论保障的约束推理算法是未来亟待解决的问题。

1.2.3 多智能体规划

不确定性是多智能体系统研究面临的最大挑战之一。即便在单智能体系统中，行动的输出也会存在不确定性（如行动失败的几率）。此外，在许多问题中，环境的状态也会因噪声或传感器能力所限而带有不确定性。在多智能体系统中，这些问题更加突出。一个智能体只能通过自己的传感器来观测环境状态，因此智能体预知其他智能体动向的能力十分有限，合作也会因此变得复杂。如果这些不确定性不能得到很好的处理，各种糟糕的情况都有可能出现。理论上，智能体通过相互交流和同步它们的信念协调行动。然而，由于带宽的限制，智能体往往不可能将必要的信息广播至其他所有智能体。此外，在许多实际场景中，通讯是不可靠的，这就不可避免地给对其他智能体行动的感知带来了不确定性。近年来，大量的研究集中于寻求以规范的方式处理多智能体系统不确定性的方法，建立了各种模型和求解方法。分散型马尔科夫决策过程（Decentralized Markov decision process，Dec-MDP）和分散型部分可观测马尔科夫决策过程（Decentralized partially observable Markov decision process，Dec-POMDP）是不确定性情形下多智能体规划建模最为常用的两种模型。不幸的是，求解 Dec-POMDP（即计算最佳规划）通常是很难的 (NEXP-complete) [8]，即便是计算绝对误差界内的解也是 NEXP-complete 的问题 [65]。特别地，联合策略的数量随智能体数量和观察次数呈指数增长，随问题规模呈双重指数增长。尽管这些复杂性预示对所有问题都行之有效的方法很难找到，开发更好的 Dec-POMDP 优化求解方法仍然是一项至关重要且近来

广受关注的问题。

一些相关研究也试图将深度学习尤其是深度强化学习应用到多智能体规划问题中。在多智能体路径寻找（Multi-Agent Path-Finding, MAPF）[81] 中，研究者将传统启发式算法和深度强化学习结合起来获得了比传统方法更好的结果 [69]，以及提出了基于深度网络的对于多种算法的选择算法来综合不同算法的优势以达到最好的求解效果 [38]。在更贴近真实场景的多智能体运动规划（Multi-Agent Motion Planning）中，研究者利用深度强化学习和基于力的（Force-based）规划算法来提高到达目标的准确率和减少到达目标的时间 [76]。相对来讲，基于深度学习的多智能体规划算法的研究还相对较少，仍未能完全发觉深度学习的潜力。

1.2.4 多智能体学习

多智能体学习（Multi-agent Learning，MAL）将机器学习技术引入多智能体系统领域，研究如何设计算法去创建动态环境下的自适应智能体。MAL 领域被广泛研究的技术是强化学习（Reinforcement Learning）。单智能体的强化学习通常在马尔科夫决策过程（Markov Decision Processes，MDP）的框架内就能被较好地描述。一些独立的强化学习算法（如 Q-learning）在智能体所处环境满足马氏性且智能体能够尝试足够多行动的前提下会收敛至最优的策略。尽管 MDP 为单智能体学习提供了可靠的数学框架，对多智能体学习却并非如此。在多个自适应智能体相互作用的情况下，一个智能体的收益通常依赖于其他个体的行动，学习环境已经不再是静态的了。此时每个智能体面临一个目标不断变化的问题——单个智能体需要学习的内容依赖于其他智能体学到的内容，并随之改变。因此，有必要对原有的 MDP 框架作相应的扩展，这其中有马尔科夫博弈和联合行动学习机等 [16, 48]。在这些扩展中，学习发生在不同智能体的状态集和行动集的积空间上。因而当智能体、状态或行动的数量太大时，这些扩展面临积空间过大的问题。此外，共享的联合行动空间也未必可用。比如在信息不完全的情况下，智能体未必能观察到其他智能体的行动。如何处理复杂的现实问题，如何高效地处理大量的状态、大量的智能体以及连续的策略空间已经成为目前 MAL 研究的首要问题。

随着深度强化学习的兴起，上述问题的解决迎来了新的转机，多智能体强化学习（Multi-Agent Reinforcement Learning, MARL）乘势兴起。多智能体强化学习中，每个智能体都采用强化学习对自己的策略进行训练，其中智能体的策略利用深度网络来表示。为解决在同时训练的过程中每个智能体的外部环境都不是静态的问题，以及多智能体之间的收益分配问题，“集中训练，分布式执行”的训练范式 [25, 50] 被提出，并成为后续工作中的一个基本训练范式。该训练范式的核心思路是在执行过程中每个智

能体只能根据自己的观察做出决策，但在训练过程中对于每个决策的评价可以通过一个使用全局信息的模块来进行，这种全局模块可以是一个 Critic 网络 [50]，一个反事实遗憾值 [25]，或者一个专门用来做收益分配的 Mix 网络 [67, 80, 90]。这样的设计在基于粒子的环境 [50] 和基于 StarCraft II 的一组合作任务 [72] 上都取得了很好的效果。在当下的研究趋势中，研究者正在把 MARL 应用到更为复杂和更大规模的多智能体学习任务上，诸如地面和空中交通管控、分布式监测、电子市场、机器人营救和机器人足球赛、智能电网等一系列实际应用场合。

1.2.5 分布式机器学习

随着“大数据”时代的来临，各大平台每天产生的数据高达 PB（1PB=1024TB）的量级。这样量级的数据在单机上存储和应用都是极度困难的，因而将数据部署到多台、多类型的机器上进行并行计算是非常必要的解决方式，这带来了分布式机器学习的兴起。分布式机器学习的目标是将具有庞大数据和计算量的任务分布式地部署到多台机器上，以提高数据计算的速度和可扩展性，减少任务的耗时。随着数据量和计算量的不断攀升，以及对于数据处理的速度和计算准确性的严格要求，分布式机器学习需要从软件和硬件的两条道路分别进行提升。

分布式机器学习平台能够从软件层面极大地为研究者带来便利。Apache Spark 是一个开源集群运算框架，最初是由 UCBerkeley AMPLab 所开发。Spark 框架是基于数据流的模型，其在处理大规模、高维度的大数据时, 巨大的参数规模使得模型训练异常耗时。为解决该困难，研究人员提出了参数服务器模型，如 Petuum 框架。在后续的长期开发和应用中，研究和工程人员深感两类模型各有其利弊，应该扬长弃短，搭配使用。Google Brain 团队于 2015 年推出了 TensorFlow [1]，其综合使用了两类模型，将计算任务抽象成有向图，可以更方便地进行分布式训练，一跃成为当时机器学习研究者使用的首选框架。晚些时候的 2017 年，Facebook 的科学家推出了 PyTorch [63]，是 Python 开源科学计算库 NumPy 的替代者，支持 GPU 且具有更高级的性能，为深度学习研究平台提供了最高的灵活性和最快的速度。PyTorch 和 TensorFlow 类似，都是将网络模型的符号表达式抽象成计算图，而不同的是 PyTorch 的计算图在运行时动态构建，而 TensorFlow 是事先经过编译的，也即静态图。PyTorch 因其简洁易用深得研究者的喜欢，而 TensorFlow 在工业界则更受欢迎。一些小众的框架如 MXNet 在特殊领域获得应用。

软件和硬件相互搭配，才能够发挥分布式机器学习的最大潜力。其中大部分研究者使用的是英伟达（Nvidia）公司的图形处理器（Graphics Processing Unit，GPU），利用其在进行图像计算中的优越性来加速机器学习算法的训练速率。除了通用的处理器，

其可以 Google 在 2016 年 5 月公布了其张量处理单元（Tensor Processing Unit，TPU）[37]，其是 Google 专门为 TensorFlow 开发的专用集成电路芯片（Application-Specific Integrated Circuit，ASIC）。随着自然语言处理和计算机视觉的研究对于计算量的要求越来越高，TPU 已经成为了处理该类任务的首选项。Google 在其云平台上开放了 TPU 租用服务，让更多的研究者可以利用 TPU 训练模型。

图 1.3　不同范式的联邦学习。图源 [94]

随着分布式机器学习的发展，数据孤岛和数据隐私问题获得了更多的关注。数据孤岛是指由于商业公司的数据大多具有极大的潜在价值，并且涉及到用户的隐私问题，数据很难在商业公司之间，甚至商业公司的部门之间进行共享，因而数据大多以孤岛的形式出现。在这种情况下，联邦学习应运而生 [39, 94]。联邦学习作为分布式的机器学习范式，可以有效解决数据孤岛问题，让参与方在不共享数据的基础上联合建模，能从技术上打破数据孤岛，实现 AI 协作。联邦学习最早于 2016 年提出，原本用于解决安卓手机终端用户在本地更新模型的问题 [53]。在这篇文章中，Google 的研究者提出了 Federated Averaging（FedAvg）算法，其基本想法是对所有本地模型的梯度进行平均来对模型进行更新，这样既避免了本地用户之间的数据共享，还可以利用其梯度信息进行联合建模。在此基础上，联邦学习拓展到了更多的场景，如适用于数据集共享相同特征空间但样本不同的情况下的横向联邦学习，适用于两个数据集共享相同的样本 ID 空间但特征空间不同的情况下纵向联邦学习，和适用于两个数据集不仅在样本上而且在特征空间上都不同的情况下的联邦迁移学习 [94]。2018 年 12 月，IEEE 标准协会批准了关于联邦学习架构和应用规范的标准立项。

1.3 相关研究应用

在这一部分，我们将回顾分布式人工智能的研究应用。

1.3.1 足球

图 1.4 Water 团队在 RoboCup 2019 的一次射门场景。图源自 [5]。

RoboCup 设立于 1997 年，经过二十余年的发展，现在已经成长为包括足球联赛 (Soccer)、救援联赛 (Rescue)、居家联赛 (@home)、青少年联赛 (Junior) 和工业联赛 (Industrial) 的大型国际性竞赛，参赛队伍也达到了 350-450 支。其中足球联赛历史最为悠久，可细分为仿真组 (包括 2D 和 3D)，小型组，中型组，标准平台组和类人组 [5]。其中参与团队采用的技术也由开始的基于规则 [4]，状态机 [70] 和规划算法 [2]，变得以深度学习 [26, 52, 83] 和深度强化学习 [55, 62, 91] 为主，这一趋势体现了近二十年科研趋势的发展，也产出了许多高水平的研究成果。在近十年的 RoboCup 中，来自中国高校的代表队表现非常亮眼，来自中国科学技术大学的陈小平教授的团队获得多项比赛的冠军。

无独有偶，2020 年，Google 公司与英超曼城俱乐部在数据科学社区和数据科学竞赛平台 Kaggle 上举办了首届 Google 足球竞赛，这次竞赛基于 Google Research Football 强化学习环境，采取 11vs11 的赛制，参赛团队需要控制其中 1 个智能体与 10 个内置智能体组成球队。经过多轮角逐，最终腾讯 AI Lab 研发的绝悟 WeKick 版本成为冠军球队。另一方面，来自 DeepMind 的科学家与利物浦足球俱乐部合作，对人工智能帮助球员和教练分析对战数据进行了探索 [86]。举例来讲，他们分析了过去几年的点球数据，发现球员在踢向自己最强的一侧更容易得分。他们还在一个可以仿真球员关节动作且决策间隔精确到毫秒的足球环境中训练了 AI 模型，发现 AI 可以自发地形成一个队伍进行合作来取得胜利 [49]。

1.3.2 安全博弈

保护关键公共基础设施和目标是各国安全机构面对的一项极具挑战性的任务。有限的安全资源使得安全机构不可能在任何时候都提供全面的安全保护。此外，安全部门的对手可以通过观察来发现安全机构的保护策略的固定模式和弱点，并据此来选择最优的攻击策略。一种降低对手观察侦查能力的方式是随机调度安全部门的保护行为，如警察巡逻、行李检测、车辆检查以及其他安全程序。然而，安全部门在进行有效的随机安全策略调度时面临许多困难 [84]，特别是有限的安全资源不能无处不在或者每时每刻提供安全保护。安全领域资源分配的关键问题是如何找出有限的安全资源最优配置方案，以获取最佳的安全保护方案。

博弈论提供了一个恰当的数学模型来研究有限的安全资源的部署，以最大限度地提高资源分配的有效性。尽管安全博弈模型是基于 20 世纪 30 年代的 Stackelberg 博弈模型，但它是一个相对年轻的领域，是在 Vincent Conitzer 和 Tuomas Sandholm 2006 年的经典论文 [17] 发表后迅速发展起来的。安全博弈论初期研究的主要参与者包括南加利福尼亚大学 Milind Tambe 教授领导的 TEAMCORE 研究小组，现在越来越多的学者参与到这项研究中，近 100 篇论文发表于人工智能领域的顶级会议 AAMAS、AAAI 和 IJCAI。过去十年，博弈论在安全领域的资源分配及调度方面的理论——安全博弈论逐渐建立并且在若干领域（如机场检查站的设置及巡逻调度、空中警察调度、海岸警卫队的巡逻、机场安保、市运输系统安全）得到成功应用。最新兴起并得到关注的是绿色安全博弈 (Green Security Game)，其包括对野生动物，鱼类以及森林的保护 [21]。虽然安全博弈论已经成功应用在一些领域，但依然面临很多挑战，包括提高现有的安全博弈算法的可扩展性，提高安全资源分配策略的鲁棒性，对恐怖分子的行为建模，协同优化，多目标优化等 [3, 85]。

1.3.3 扑克和麻将

扑克和麻将是世界流行的娱乐活动，不同的扑克和麻将游戏具有不同的牌数和不同的胜负判定规则。扑克和麻将游戏可以被非完全信息序贯博弈来建模，而研究者关心的是如何计算这类游戏的均衡策略，特别是大规模的游戏。举例来讲，对于 2 人有限下注的德州扑克，可能出现的不同牌面状态是 10^{14}。因而扑克和麻将的求解对于应用和研究领域都是一个非常具有挑战性的问题。

相关的研究和应用最早在德州扑克上面出现。德州扑克可以写成一个非常宽但是相对较浅的博弈树，来自卡耐基梅隆大学的 Tuomas Sandholm 教授团队经过十余年的

[1] https://projects.iq.harvard.edu/files/teamcore/files/lecture_04_teamcore_ijcai_2018_ppt.pdf

图 1.5　绿色安全博弈[1]

研究最终在 2017 年开发出了第一个在二人无限下注的德州扑克上打败人类职业玩家但没有使用深度学习的算法 Libratus [11]，其中最主要应用的技术是先根据博弈的特殊性质进行抽象然后应用 CFR 算法进行训练。在此之后，他们将相同的算法应用在多人无限下注的德州扑克上面，也即 Pluribus [12]，依然取得了很好的结果。在此之后，研究者将目光转移到了更复杂的扑克游戏上，如斗地主这类决策轮数更多，牌数更多的游戏上面。由于博弈树变得更宽而且更深，而 Libratus 和 Pluribus 的基础算法 CFR 假设每一次的搜索都能够到结束节点，这在斗地主上面是不容易满足的。最新的研究成果 DeltaDou [36] 和 DouZero [98] 都是基于强化学习算法，它只需要向下搜索一步或几步就可以进行更新，具有更强的适用性。其中 DouZero 打败了之前的所有 344 个 AI 智能体，成为了当前最强的算法。

麻将相比于德州扑克和斗地主是更为复杂和困难的游戏，其难点主要来自三个方面: 1. 麻将具有更多的玩家和更多的牌，因而麻将的牌面状态的数量更多，可以达到惊人的 10^{121} 数量级；2. 麻将的计分规则更复杂，因而决策需要考虑的因素更多也更复杂；以及 3. 麻将中每个玩家可以选择的动作更多，包括碰、杠和吃。研究者解决麻将的努力也没有停止过，与研究领域整体发展相呼应，研究者提出了基于抽象算法、蒙特卡洛搜索、对手建模和深度学习尤其是强化学习的 AI 算法 [27, 41, 56]。其中取得了最好效果的是微软亚洲研究院的 Suphx 系统 [45]，其分别训练了丢牌模型、立直模型、吃牌模型、碰牌模型以及杠牌模型来处理麻将中的复杂决策情况。另外，Suphx 还有一个基于规则的赢牌模型决定在可以赢牌的时候要不要赢牌。Suphx 已在天凤平台房和其他玩家对战了 5000 多场，达到了目前的最高段位 10 段，其稳定段位达到了 8.7 段，超过了平台上另外两个知名 AI 算法 Bakuuch [56] 和 NAGA [87] 的水平以及顶

级人类选手的平均水平。

1.3.4 视频游戏

视频游戏是现代社会人们的广泛的娱乐方式。一直广受欢迎的游戏有即时战略 (Real-Time Strategy (RTS)) 游戏星际争霸和魔兽世界，近几年最流行的是多人在线战术竞技 (Multiplayer online battle arena (MOBA)) 游戏，如 Dota, 王者荣耀和英雄联盟。视频游戏提供了理想的研究环境，它们具有和现实问题相似的复杂性，可以很好地测试算法在复杂环境中寻找到好的决策策略的能力。

星际争霸 II 是暴雪公司在 2010 年推出的即时战略游戏 (Real-Time Strategy Game)，游戏中玩家透过俯瞰的视角模式观阅整个战场并对军队下达指令，最终目标就是击败战场上的对手们，主要的游戏技巧着重在资源上，玩家用采集的资源建造不同的建筑、军队并进行升级。2019 年 1 月份，DeepMind 的研究者推出的 AlphaStar，在 1v1 的对战中打败了人类职业选手，并最终在天梯榜上达到了 Grandmaster 层级，超过了 99.8% 的人类选手，相关研究发表在《自然》杂志 [88]。

同样受到关注的还有 MOBA 游戏，相比于 RTS 游戏，MOBA 在单个玩家的决策上要简单，但更关注不同玩家之间的合作，另外由于 MOBA 游戏一般拥有多个玩家，如王者荣耀中，17 个英雄可以组成的组合有 4,900,896 中，40 个英雄可以组成的组合有 213,610,453,056 种，由于不同的英雄具有不同的技巧，因此对战状况是非常不同的。如何训练 AI 算法支撑不同的战队组合并打赢对手，是一个非常具有挑战性的问题。2019 年 OpenAI 的研究者提出 OpenAI Five，可以在 5v5 的 Dota 2 对战中打败人类玩家 [7]。但是其一大缺点是只支持 17 个英雄，而 Dota 2 中共有 117 个英雄，这极大地降低了问题的难度以及可玩性。2020 年腾讯的研究人员腾讯公司的研究者开发的 “绝悟” 在王者荣耀游戏中达到职业电竞选手水平 [96]。这些研究不仅在学术界产生影响，还可以直接应用在游戏中的人机对打模式，为游戏公司带来经济效益。

1.4 未来研究挑战

在这一部分我们将扼要论述未来研究挑战。

1.4.1 超大规模分布式人工智能系统

我们的世界包含着许多超大规模的分布式系统，比如一个城市的所有红绿灯系统，协调一个城市的所有车辆来避免交通堵塞。这些超大规模的分布式系统普遍且重要，而如今人们对此的解决方案大多还是经验性的。因为未来的研究人们会更多地关

注超大规模的分布式系统。大规模分布式系统为分布式人工智能带来的一大挑战就是其可能出现的状态和可能采取的决策都随着系统规模呈指数性增长，如何找到最优或者有效的策略是极度困难的。为了解决这个挑战，研究者提出的方法大致可以分为如下三个方向：1. 对大规模系统进行抽象和近似，如将平均场理论应用于分布式系统 [95]，但平均场理论所处理的场景是非原子性的，也即其忽略了每个个体的差异性而只关注大量个体构成的分布，而多智能体系统的智能体是原子性的，需要指定不同个体的行为，这两个领域的方法在理论和应用上都有不同，需要研究者更多的研究。2. 利用离线训练来降低训练所需的时间，离线数据能够在训练中使得模型具有相应的领域知识，而避免从随机初始化的状态开始训练所带来的大量训练数据和实践的需要，但是离线数据的质量是很重要的，如果质量不够好，反而会发生负向迁移，对于离线训练模型的在线泛化和安全性也是研究的一大热点 [29]。3. 将大规模系统分为小规模系统并利用通信进行分布式训练，这类训练范式随着分布式训练和联邦学习的发展获得了长足的进步，但是联邦学习依然不是绝对安全的，因而隐私和安全问题和在超大规模情况下如何进行有效的协调依然是当下和未来的研究热点 [99]。

1.4.2 分布式人工智能系统的鲁棒性和安全性

利用算法来指导现实世界中的决策能够带来效率和表现的提升，但同时也带来了一些担忧，尤其是以深度学习和深度神经网络为基础的算法，其对于现实世界中可能存在的扰动、不确定性甚至蓄意的攻击都可能带来可怕的后果。许多研究表明，深度学习模型对扰动是不够鲁棒的，所以当下研究的一大热点是提升算法的鲁棒性和安全性，让它们能够在不确定性甚至对抗性的决策环境中都能够做出相对较好的决策。因此研究者主要采用的方法有：1. 对抗训练，针对深度学习模型的脆弱性，研究者提出了对抗训练来提升模型的鲁棒性，其假定有对抗者来采取对于决策者最差的行为对环境和问题进行扰动，然后决策者需要求解最大化最小收益的策略，近来相关进展包括如何将深度学习模型拓展到训练时没有见过的攻击手段，以及如何应对多重攻击手段的综合攻击 [82]。2. 安全决策 [28]，不同于模型的鲁棒性，安全性是深度学习模型应用于现实世界另外需要考虑的问题，其要求不仅需要计算决策策略，还需要根据专业知识对策略的风险性进行评估，并且在应用过程中对其所带来的风险性进行感知和控制，因为基于数据训练的深度学习模型对于一些不常出现的场景很难提供合理的处理决策，如百年一遇的暴雨或者洪水，这都需要人工介入来提升模型的安全性。

1.4.3 分布式人工智能决策的可解释性

基于深度学习的分布式人工智能方法虽然在许多问题上带来了令人兴奋的进展，但是对于深度学习的“黑箱”性质对于研究者提出了增加模型可解释性的要求。可解释性是指算法在得到决策的同时，也应当解释为什么做这种决策，这在算法应用在现实世界中，尤其是有人合作参与或者监督的场景下至关重要。为解决算法决策的可解释性问题，研究者在模型中引入了因果推断模型和表征学习模型来对决策所依据的表征进行表示，让人们能够理解算法做出的决策。相关的解决思路包括：1. 因果推断，这是图灵奖得主 Yoshua Bengio 在最近的研究中推动的对于可解释性的技术思路，其主要想法是将传统的因果关系表述引入深度学习模型，并将其表述成为人类可以理解的因果关系模型 [73]。2. 分层目标，对于序贯的多步决策，如何能够理解在每一步智能体做出决策的目标是很重要的，分层目标能够给出智能体在达成一个大的目标的过程中每个阶段执行的小目标，这可以极大地增加人类对于智能体决策的理解 [51]。

1.4.4 传统和深度学习的方法结合

将深度学习方法应用于分布式人工智能是当前研究的一大趋势，并且取得了一定的成果。但需要注意到的是，深度学习方法有其弊端，主要体现在：1. 需要大量的数据，而在机器人领域中，数据量一般是比较小的；2. 大量的算力，这在一些边缘设备 (edge devices)（如手机）是无法满足的；3. 缺乏理论保证如解的最优性，这使得一些对精度要求较高的领域是无法使用的。与此相反，传统方法可以在一定程度上缓解这些弊端。因而未来的一个重要的研究方向是将传统和深度学习的方法相结合，取二者的优势而避免二者的弊端，这将会更大地提升分布式人工智能的可应用性。近来获得最多关注的是来自 Google 的科学家利用深度学习模型学习求解混合整数规划，其利用离线数据训练的深度模型来选择分支定界方法中分支的变量，这在经典方法中是通过复杂的计算来决定的，这样可以降低算法的运行时间 [60]。虽然其分支方法是深度模型决定的，但在每一次的求解中，该算法依然利用的是传统方法，这可以保证解的最优性。这样的传统和深度学习结合的方法取得了两方面的平衡，相信该领域在未来的时间内能够取得更快速的进展。

参考文献

[1] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al.: TensorFlow: A system for large-scale machine learning. In: 12th USENIX symposium on operating systems design and implementation (OSDI 16), pp. 265–283 (2016)

[2] Allgeuer, P., Behnke, S.: Hierarchical and state-based architectures for robot behavior planning and control. arXiv preprint arXiv:1809.11067 (2018)

[3] An, B.: Game theoretic analysis of security and sustainability. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI), pp. 5111–5115 (2017)

[4] Asada, M., Stone, P., Kitano, H., Drogoul, A., Duhaut, D., Veloso, M., Asamas, H., Suzuki, S.: The RoboCup physical agent challenge: Goals and protocols for phase I. In: Robot Soccer World Cup, pp. 42–61. Springer (1997)

[5] Asada, M., Stone, P., Veloso, M., Lee, D., Nardi, D.: RoboCup: A treasure trove of rich diversity for research issues and interdisciplinary connections. IEEE Robotics & Automation Magazine **26**(3), 99–102 (2019)

[6] Bello, I., Pham, H., Le, Q.V., Norouzi, M., Bengio, S.: Neural combinatorial optimization with reinforcement learning. In: ICLR (2017)

[7] Berner, C., Brockman, G., Chan, B., Cheung, V., Dębiak, P., Dennison, C., Farhi, D., Fischer, Q., Hashme, S., Hesse, C., et al.: Dota 2 with large scale deep reinforcement learning. arXiv preprint arXiv:1912.06680 (2019)

[8] Bernstein, D.S., Givan, R., Immerman, N., Zilberstein, S.: The complexity of decentralized control of Markov decision processes. Mathematics of Operations Research **27**(4), 819–840 (2002)

[9] Bond, A.H., Gasser, L.: Readings in Distributed Artificial Intelligence. Morgan Kauf-

mann (1988)

[10] Brown, N., Lerer, A., Gross, S., Sandholm, T.: Deep counterfactual regret minimization. In: ICML, pp. 793–802 (2019)

[11] Brown, N., Sandholm, T.: Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. Science **359**(6374), 418–424 (2018)

[12] Brown, N., Sandholm, T.: Superhuman AI for multiplayer poker. Science **365**(6456), 885–890 (2019)

[13] Cameron, C., Chen, R., Hartford, J., Leyton-Brown, K.: Predicting propositional satisfiability via end-to-end learning. In: AAAI, pp. 3324–3331 (2020)

[14] Cappart, Q., Chételat, D., Khalil, E., Lodi, A., Morris, C., Veličković, P.: Combinatorial optimization and reasoning with graph neural networks. arXiv preprint arXiv:2102.09544 (2021)

[15] Cappart, Q., Goutierre, E., Bergman, D., Rousseau, L.M.: Improving optimization bounds using machine learning: Decision diagrams meet deep reinforcement learning. In: AAAI, pp. 1443–1451 (2019)

[16] Claus, C., Boutilier, C.: The dynamics of reinforcement learning in cooperative multiagent systems. In: AAAI, pp. 746–752 (1998)

[17] Conitzer, V., Sandholm, T.: Computing the optimal strategy to commit to. In: EC, pp. 82–90 (2006)

[18] Dai, H., Khalil, E.B., Zhang, Y., Dilkina, B., Song, L.: Learning combinatorial optimization algorithms over graphs. In: NeurIPS, pp. 6351–6361 (2017)

[19] Daskalakis, C., Goldberg, P.W., Papadimitriou, C.H.: The complexity of computing a Nash equilibrium. SIAM Journal on Computing **39**(1), 195–259 (2009)

[20] Deng, Y., Yu, R., Wang, X., An, B.: Neural regret-matching for distributed constraint optimization problems. In: IJCAI, pp. 146–153 (2021)

[21] Fang, F., Nguyen, T.H.: Green security games: Apply game theory to addressing green security challenges. ACM SIGecom Exchanges **15**(1), 78–83 (2016)

[22] Fennell, R., Lesser, V.: Parallelism in ai problem solving: A case study of hearsay 2. Tech. rep., CARNEGIE-MELLON UNIV PITTSBURGH PA DEPT OF COMPUTER SCIENCE (1975)

[23] Fikes, R.E., Nilsson, N.J.: STRIPS: A new approach to the application of theorem proving to problem solving. Artificial intelligence **2**(3-4), 189–208 (1971)

[24] Fioretto, F., Yeoh, W., Pontelli, E.: A multiagent system approach to scheduling devices in smart homes. In: AAMAS, pp. 981–989 (2017)

[25] Foerster, J., Farquhar, G., Afouras, T., Nardelli, N., Whiteson, S.: Counterfactual

multi-agent policy gradients. In: AAAI, pp. 2974–2982 (2018)

[26] Fukushima, T., Nakashima, T., Akiyama, H.: Mimicking an expert team through the learning of evaluation functions from action sequences. In: Robot World Cup, pp. 170–180. Springer (2018)

[27] Gao, S., Okuya, F., Kawahara, Y., Tsuruoka, Y.: Building a computer mahjong player via deep convolutional neural networks. arXiv preprint arXiv:1906.02146 (2019)

[28] Garcıa, J., Fernández, F.: A comprehensive survey on safe reinforcement learning. Journal of Machine Learning Research **16**(1), 1437–1480 (2015)

[29] Gulcehre, C., Wang, Z., Novikov, A., Le Paine, T., Gomez Colmenarejo, S., Zolna, K., Agarwal, R., Merel, J., Mankowitz, D., Paduraru, C., et al.: RL Unplugged: Benchmarks for offline reinforcement learning. arXiv e-prints pp. arXiv–2006 (2020)

[30] He, H., Daumé III, H., Eisner, J.: Learning to search in branch-and-bound algorithms. In: NeurIPS, pp. 3293–3301 (2014)

[31] Heinrich, J., Lanctot, M., Silver, D.: Fictitious self-play in extensive-form games. In: ICML, pp. 805–813 (2015)

[32] Heinrich, J., Silver, D.: Deep reinforcement learning from self-play in imperfect-information games. arXiv preprint arXiv:1603.01121 (2016)

[33] Hewitt, C.: Viewing control structures as patterns of passing messages. Artificial intelligence **8**(3), 323–364 (1977)

[34] Huhns, M.N.: Distributed Artificial Intelligence. Pitman Publishing Ltd. London, England (1987)

[35] Jain, M., Korzhyk, D., Vaněk, O., Conitzer, V., Pěchouček, M., Tambe, M.: A double oracle algorithm for zero-sum security games on graphs. In: AAMAS, pp. 327–334 (2011)

[36] Jiang, Q., Li, K., Du, B., Chen, H., Fang, H.: DeltaDou: Expert-level doudizhu AI through self-play. In: IJCAI, pp. 1265–1271 (2019)

[37] Jouppi, N.P., Young, C., Patil, N., Patterson, D., Agrawal, G., Bajwa, R., Bates, S., Bhatia, S., Boden, N., Borchers, A., et al.: In-datacenter performance analysis of a tensor processing unit. In: Proceedings of the 44th annual international symposium on computer architecture, pp. 1–12 (2017)

[38] Kaduri, O., Boyarski, E., Stern, R.: Algorithm selection for optimal multi-agent pathfinding. In: ICAPS, pp. 161–165 (2020)

[39] Kairouz, P., McMahan, H.B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A.N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al.: Advances and open problems in federated learning. arXiv preprint arXiv:1912.04977 (2019)

[40] Kool, W., van Hoof, H., Welling, M.: Attention, learn to solve routing problems! In: ICLR (2018)

[41] Kurita, M., Hoki, K.: Method for constructing artificial intelligence player with abstractions to markov decision processes in multiplayer game of mahjong. IEEE Transactions on Games **13**(1), 99–110 (2020)

[42] Lanctot, M., Zambaldi, V., Gruslys, A., Lazaridou, A., Tuyls, K., Pérolat, J., Silver, D., Graepel, T.: A unified game-theoretic approach to multiagent reinforcement learning. In: NeurIPS, pp. 4193–4206 (2017)

[43] Lederman, G., Rabe, M., Seshia, S., Lee, E.A.: Learning heuristics for quantified boolean formulas through reinforcement learning. In: ICLR (2020)

[44] Leslie, D.S., Collins, E.J.: Generalised weakened fictitious play. Games and Economic Behavior **56**(2), 285–298 (2006)

[45] Li, J., Koyamada, S., Ye, Q., Liu, G., Wang, C., Yang, R., Zhao, L., Qin, T., Liu, T.Y., Hon, H.W.: Suphx: Mastering mahjong with deep reinforcement learning. arXiv preprint arXiv:2003.13590 (2020)

[46] Li, S., Negenborn, R.R., Lodewijks, G.: Distributed constraint optimization for addressing vessel rotation planning problems. Engineering Applications of Artificial Intelligence **48**, 159–172 (2016)

[47] Li, S., Zhang, Y., Wang, X., Xue, W., An, B.: CFR-MIX: Solving imperfect information extensive-form games with combinatorial action space. In: IJCAI, pp. 3663–3669 (2021)

[48] Littman, M.L.: Markov games as a framework for multi-agent reinforcement learning. In: ICML, pp. 157–163 (1994)

[49] Liu, S., Lever, G., Wang, Z., Merel, J., Eslami, S., Hennes, D., Czarnecki, W.M., Tassa, Y., Omidshafiei, S., Abdolmaleki, A., et al.: From motor control to team play in simulated humanoid football. arXiv preprint arXiv:2105.12196 (2021)

[50] Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., Mordatch, I.: Multi-agent actor-critic for mixed cooperative-competitive environments. In: NeurIPS, pp. 6382–6393 (2017)

[51] Lyu, D., Yang, F., Liu, B., Gustafson, S.: SDRL: Interpretable and data-efficient deep reinforcement learning leveraging symbolic planning. In: AAAI, pp. 2970–2977 (2019)

[52] MacAlpine, P., Torabi, F., Pavse, B., Sigmon, J., Stone, P.: UT Austin Villa: RoboCup 2018 3D simulation league champions. In: Robot World Cup, pp. 462–475 (2018)

[53] McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-

efficient learning of deep networks from decentralized data. In: Artificial intelligence and statistics, pp. 1273–1282 (2017)

[54] McMahan, H.B., Gordon, G.J., Blum, A.: Planning in the presence of cost functions controlled by an adversary. In: ICML, pp. 536–543 (2003)

[55] Mendoza, J.P., Simmons, R., Veloso, M.: Online learning of robot soccer free kick plans using a bandit approach. In: ICAPS, pp. 504–508 (2016)

[56] Mizukami, N., Tsuruoka, Y.: Building a computer mahjong player based on monte carlo simulation and opponent models. In: 2015 IEEE Conference on Computational Intelligence and Games (CIG), pp. 275–283. IEEE (2015)

[57] Modi, P.J., Shen, W.M., Tambe, M., Yokoo, M.: An asynchronous complete method for distributed constraint optimization. In: AAMAS, pp. 161–168 (2003)

[58] Moravčík, M., Schmid, M., Burch, N., Lisỳ, V., Morrill, D., Bard, N., Davis, T., Waugh, K., Johanson, M., Bowling, M.: Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. Science **356**(6337), 508–513 (2017)

[59] Muller, P., Omidshafiei, S., Rowland, M., Tuyls, K., Perolat, J., Liu, S., Hennes, D., Marris, L., Lanctot, M., Hughes, E., et al.: A generalized training approach for multi-agent learning. In: ICLR (2019)

[60] Nair, V., Bartunov, S., Gimeno, F., von Glehn, I., Lichocki, P., Lobov, I., O'Donoghue, B., Sonnerat, N., Tjandraatmadja, C., Wang, P., et al.: Solving mixed integer programs using neural networks. arXiv preprint arXiv:2012.13349 (2020)

[61] Nazari, M., Oroojlooy, A., Takáč, M., Snyder, L.V.: Reinforcement learning for solving the vehicle routing problem. In: NeurIPS, pp. 9861–9871 (2018)

[62] Ocana, J.M.C., Riccio, F., Capobianco, R., Nardi, D.: Cooperative multi-agent deep reinforcement learning in a 2 versus 2 free-kick task. In: Robot World Cup, pp. 44–57. Springer (2019)

[63] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: PyTorch: An imperative style, high-performance deep learning library. NeurIPS **32**, 8026–8037 (2019)

[64] Prates, M., Avelar, P.H., Lemos, H., Lamb, L.C., Vardi, M.Y.: Learning to solve NP-complete problems: A graph neural network for decision TSP. In: AAAI, pp. 4731–4738 (2019)

[65] Rabinovich, Z., Goldman, C.V., Rosenschein, J.S.: The complexity of multiagent systems: The price of silence. In: AAMAS, pp. 1102–1103 (2003)

[66] Rashid, T., Farquhar, G., Peng, B., Whiteson, S.: Weighted QMIX: Expanding monotonic value function factorisation. arXiv e-prints pp. arXiv–2006 (2020)

[67] Rashid, T., Samvelyan, M., Schroeder, C., Farquhar, G., Foerster, J., Whiteson, S.: Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In: ICML, pp. 4295–4304 (2018)

[68] Razeghi, Y., Kask, K., Lu, Y., Baldi, P., Agarwal, S., , Dechter, R.: Deep bucket elimination. In: IJCAI (2021)

[69] Reijnen, R., Zhang, Y., Nuijten, W., Senaras, C., Goldak-Altgassen, M.: Combining deep reinforcement learning with search heuristics for solving multi-agent path finding in segment-based layouts. In: 2020 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 2647–2654 (2020)

[70] Risler, M., von Stryk, O.: Formal behavior specification of multi-robot systems using hierarchical state machines in XABSL. In: AAMAS08-Workshop on Formal Models and Methods for Multi-Robot Systems, p. 7 (2008)

[71] Rosenschein, J.S., Genesereth, M.R.: Deals among rational agents. In: IJCAI, pp. 91–99 (1985)

[72] Samvelyan, M., Rashid, T., Schroeder de Witt, C., Farquhar, G., Nardelli, N., Rudner, T.G., Hung, C.M., Torr, P.H., Foerster, J., Whiteson, S.: The StarCraft multi-agent challenge. In: AAMAS, pp. 2186–2188 (2019)

[73] Schölkopf, B., Locatello, F., Bauer, S., Ke, N.R., Kalchbrenner, N., Goyal, A., Bengio, Y.: Toward causal representation learning. Proceedings of the IEEE **109**(5), 612–634 (2021)

[74] Schrittwieser, J., Antonoglou, I., Hubert, T., Simonyan, K., Sifre, L., Schmitt, S., Guez, A., Lockhart, E., Hassabis, D., Graepel, T., et al.: Mastering Atari, Go, chess and shogi by planning with a learned model. Nature **588**(7839), 604–609 (2020)

[75] Selsam, D., Lamm, M., Benedikt, B., Liang, P., de Moura, L., Dill, D.L., et al.: Learning a SAT solver from single-bit supervision. In: ICLR (2018)

[76] Semnani, S.H., Liu, H., Everett, M., de Ruiter, A., How, J.P.: Multi-agent motion planning for dense and dynamic environments via deep reinforcement learning. IEEE Robotics and Automation Letters **5**(2), 3221–3226 (2020)

[77] Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al.: Mastering the game of Go with deep neural networks and tree search. Nature **529**(7587), 484–489 (2016)

[78] Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., et al.: A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. Science **362**(6419), 1140–1144 (2018)

[79] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al.: Mastering the game of go without human knowledge. Nature **550**(7676), 354–359 (2017)

[80] Son, K., Kim, D., Kang, W.J., Hostallero, D.E., Yi, Y.: QTRAN: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In: ICML, pp. 5887–5896 (2019)

[81] Stern, R., Sturtevant, N.R., Felner, A., Koenig, S., Ma, H., Walker, T.T., Li, J., Atzmon, D., Cohen, L., Kumar, T.S., et al.: Multi-agent pathfinding: Definitions, variants, and benchmarks. In: Twelfth Annual Symposium on Combinatorial Search (2019)

[82] Stutz, D., Hein, M., Schiele, B.: Confidence-calibrated adversarial training: Generalizing to unseen attacks. In: ICML, pp. 9155–9166. PMLR (2020)

[83] Suzuki, Y., Nakashima, T.: On the use of simulated future information for evaluating game situations. In: Robot World Cup, pp. 294–308 (2019)

[84] Tambe, M.: Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned. Cambridge University Press (2011)

[85] Tambe, M., Jiang, A.X., An, B., Jain, M., et al.: Computational game theory for security: Progress and challenges. In: AAAI spring symposium on applied computational game theory (2014)

[86] Tuyls, K., Omidshafiei, S., Muller, P., Wang, Z., Connor, J., Hennes, D., Graham, I., Spearman, W., Waskett, T., Steel, D., et al.: Game plan: What AI can do for football, and what football can do for AI. Journal of Artificial Intelligence Research **71**, 41–88 (2021)

[87] Village, D.M.: NAGA: Deep learning mahjong AI. `https://dmv.nico/ja/articles/mahjong_ai_naga/`. Accessed on 2021-07-16.

[88] Vinyals, O., Babuschkin, I., Czarnecki, W.M., Mathieu, M., Dudzik, A., Chung, J., Choi, D.H., Powell, R., Ewalds, T., Georgiev, P., et al.: Grandmaster level in StarCraft II using multi-agent reinforcement learning. Nature **575**(7782), 350–354 (2019)

[89] Vinyals, O., Fortunato, M., Jaitly, N.: Pointer networks. In: NeurIPS, pp. 2692–2700 (2015)

[90] Wang, J., Ren, Z., Liu, T., Yu, Y., Zhang, C.: QPLEX: Duplex dueling multi-agent q-learning. In: ICLR (2020)

[91] Watkinson, W.B., Camp, T.: Training a robocup striker agent via transferred reinforcement learning. In: Robot World Cup, pp. 109–121. Springer (2018)

[92] Xu, H., Koenig, S., Kumar, T.S.: Towards effective deep learning for constraint satisfaction problems. In: CP, pp. 588–597 (2018)

[93] Xue, W., Zhang, Y., Li, S., Wang, X., An, B., Yeo, C.K.: Solving large-scale extensive-form network security games via neural fictitious self-play. In: IJCAI, pp. 3713–3720 (2021)

[94] Yang, Q., Liu, Y., Chen, T., Tong, Y.: Federated machine learning: Concept and applications. ACM Transactions on Intelligent Systems and Technology (TIST) **10**(2), 1–19 (2019)

[95] Yang, Y., Luo, R., Li, M., Zhou, M., Zhang, W., Wang, J.: Mean field multi-agent reinforcement learning. In: ICML, pp. 5571–5580 (2018)

[96] Ye, D., Chen, G., Zhang, W., Chen, S., Yuan, B., Liu, B., Chen, J., Liu, Z., Qiu, F., Yu, H., et al.: Towards playing full moba games with deep reinforcement learning. arXiv preprint arXiv:2011.12692 (2020)

[97] Yeoh, W., Yokoo, M.: Distributed problem solving. AI Magazine **33**(3), 53–53 (2012)

[98] Zha, D., Xie, J., Ma, W., Zhang, S., Lian, X., Hu, X., Liu, J.: DouZero: Mastering doudizhu with self-play deep reinforcement learning. arXiv preprint arXiv:2106.06135 (2021)

[99] Zhu, L., Han, S.: Deep leakage from gradients. In: Federated learning, pp. 17–31. Springer (2020)

[100] Zinkevich, M., Johanson, M., Bowling, M., Piccione, C.: Regret minimization in games with incomplete information. In: NeurIPS, pp. 1729–1736 (2007)