计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.200700217

无模型强化学习研究综述

秦智慧[1,2]　李　宁[1]　刘晓彤[1,3,4,5]　刘秀磊[1,2]　佟　强[1,2]　刘旭红[1,2]
1 北京材料基因工程高精尖创新中心(北京信息科技大学)　北京 100101
2 北京信息科技大学数据与科学情报分析实验室　北京 100101
3 中国科学院煤炭化学研究所煤转化国家重点实验室　太原 030001
4 中科合成油技术有限公司国家能源煤基液体燃料研发中心　北京 101400
5 中国科学院大学　北京 100049
(qzh@bistu.edu.cn)

摘　要　强化学习(Reinforcement Learning,RL)作为机器学习领域中与监督学习、无监督学习并列的第三种学习范式,通过与环境进行交互来学习,最终将累积收益最大化。常用的强化学习算法分为模型化强化学习(Model-based Reinforcement Learning)和无模型强化学习(Model-free Reinforcement Learning)。模型化强化学习需要根据真实环境的状态转移数据来预定义环境动态模型,随后在通过环境动态模型进行策略学习的过程中无须再与环境进行交互。在无模型强化学习中,智能体通过与环境进行实时交互来学习最优策略,该方法在实际任务中具有更好的通用性,因此应用范围更广。文中对无模型强化学习的最新研究进展与发展动态进行了综述。首先介绍了强化学习、模型化强化学习和无模型强化学习的基础理论;然后基于价值函数和策略函数归纳总结了无模型强化学习的经典算法及各自的优缺点;最后概述了无模型强化学习在游戏 AI、化学材料设计、自然语言处理和机器人控制领域的最新研究现状,并对无模型强化学习的未来发展趋势进行了展望。

关键词:人工智能;强化学习;深度强化学习;无模型强化学习;马尔可夫决策过程
中图法分类号　TP181

Overview of Research on Model-free Reinforcement Learning

QIN Zhi-hui[1,2],LI Ning[1],LIU Xiao-tong[1,3,4,5],LIU Xiu-lei[1,2],TONG Qiang[1,2] and LIU Xu-hong[1,2]
1 Beijing Advanced Innovation Center for Materials Genome Engineering (Beijing Information Science and Technology University), Beijing 100101,China
2 Laboratory of Data Science and Information Studies,Beijing Information Science and Technology University,Beijing 100101,China
3 State Key Laboratory of Coal Conversion,Institute of Coal Chemistry,Chinese Academy of Sciences,Taiyuan 030001,China
4 National Energy Center for Coal to Liquids,Synfuels China Co.,Ltd,Beijing 101400,China
5 University of Chinese Academy of Sciences,Beijing 100049,China

Abstract　Reinforcement Learning (RL) is a different learning paradigm from supervised learning and unsupervised learning. It focuses on the interacting process between agent and environment to maximize the accumulated reward. The commonly used RL algorithm is divided into Model-based Reinforcement Learning (MBRL) and Model-free Reinforcement Learning (MFRL). In MBRL,there is a well-designed model to fit the state transition of the environment. In most cases,it is difficult to build an accurate enough model under prior knowledge. In MFRL,parameters in the model are fine-tuned through continuous interactions with the environment. The whole process has good portability. Therefore,MFRL is widely used in various fields. This paper reviews the recent research progress of MFRL. Firstly,an overview of basic theory is given. Then,three types of classical algorithms of

到稿日期:2020-07-31　返修日期:2020-12-02　本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。
基金项目:国家重点研发计划(2018YFC0830202);北京信息科技大学"勤信人才"培育计划项目(2020);北京信息科技大学促进高校内涵发展——信息+项目一面向大数据的竞争情报分析关键技术研究;北京市教育委员会科技计划一般项目(KM202111232003);北京市自然基金(4204100)
This work was supported by the National Key R&D Program of China(2018YFC0830202),Qin Xin Talents Cultivation Program, Beijing Information Science & Technology University(2020),Beijing University of Information Science and Technology to Promote the Development of the Connotation of Colleges and Universities——Information+Project—Key Technology Research for Competitive Analysis of Big Data,Beijing Education Commission for General Project of Science and Technology Plan(KM202111232003) and Beijing Natural Science Foundation(4204100).
通信作者:刘秀磊(xiuleiliu@hotmail.com)

MFRL based on value function and strategy function are introduced. Finally, the related researches of MFRL are summarized and prospected.

Keywords Artificial intelligence, Reinforcement learning, Deep reinforcement learning, Model-free reinforcement learning, Markov decision process

1 引言

强化学习又称增强学习，在学术界对 RL 与统计学、优化理论和其他数学学科的互动研究有了突破后，RL 逐渐成为了各领域的研究热点[1]。随着深度学习（Deep Learning，DL）[2]的兴起，融合深度神经网络和 RL 的深度强化学习（Deep Reinforcement Learning，DRL）[3]技术的研究和应用日益增多。

RL 是一种不同于监督学习的学习方式。监督学习通过外部提供的标注数据集进行学习，每一个样本都是训练中的“监督者”[4]。而 RL 中并不存在这样的“监督者”，因此 RL 提出了奖励信号这个概念。它与监督学习中的监督信号不同，为了考虑智能体（agent）整体的累积收益，它是被延迟反馈的。同时，监督学习的训练数据之间一般是独立的，而 RL 处理的是序贯决策问题，每一步在顺序上都具有依赖关系。

RL 也是一种不同于无监督学习的学习方式。无监督学习的主要目标是寻找未标注数据集中隐含的结构关系，而 RL 的目标是最大化累积收益。同时，无监督学习没有 RL 的奖励信号，其数据之间一般也是独立的。

在实际应用中，根据 agent 是否通过与环境交互获得的数据来预定义环境动态模型，将 RL 分为模型化强化学习和无模型强化学习[5]，具体如图 1 所示。

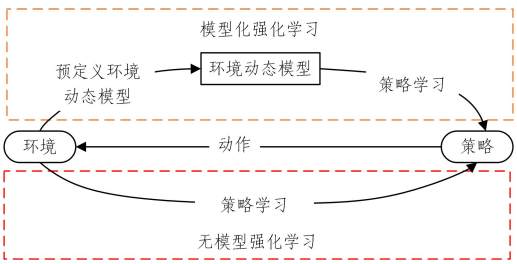

图 1 两类强化学习模式简图

Fig. 1 Two types of RL model diagram

模型化强化学习指先在与环境交互的数据中创建环境动态模型，然后基于该模型学习最优策略。它一般包含状态转移预测和奖励预测两个独立模型。如果两个模型可以准确描述真正的环境动态模型，那么当输入一个状态和动作时就不需要与环境进行实时交互，可以直接基于模型预测得到新的状态和动作奖励，从而极大地提高数据的利用率。但当面对的问题具有复杂的状态动作空间时，准确估计环境动态模型存在巨大挑战。尤其是在交互前期得到的数据较少时，环境动态模型极易存在模型误差，利用不准确的环境动态模型进行学习，极易导致双重近似误差[6]。针对模型的准确性，有很多改进算法被提出。例如，学习控制的概率推理方法（Probabilistic Inference for Learning Control，PILCO）[7]，其将环境动态模型建模为高斯过程（Gaussian Process，GP），但这种高斯假设以及需呈特定指数形式的奖励函数极大地限制了 PILCO 算法在复杂问题中的应用。之后，研究人员又提出了基于最小二乘条件密度估计的模型化策略搜索方法（Model-based Policy Gradients with Parameter-based Exploration by Least-squares Conditional Density Estimation，Mb-PGPE-LSCDE）[5]，但其仅在采样预算有限时具有良好效果，难以处理高维度问题。面对各领域复杂的应用场景，模型化强化学习若存在模型误差，其性能将远低于无模型强化学习[8]。

无模型强化学习指 agent 与环境进行实时交互和探索，并直接对得到的经验数据进行学习，最终实现累积收益最大化或达到特定目标[4]。无模型强化学习不需要拟合环境动态模型，经过与环境的实时交互可以保证 agent 渐近收敛得到最优解。然而，无模型强化学习通常需要大量的训练样本和训练时间，因此如何提高数据利用率和学习效率是无模型强化学习的研究重点。

本文将围绕无模型强化学习展开综述，首先介绍 RL 的基础知识，然后归纳总结无模型强化学习的经典算法及相关工作，最后概述无模型强化学习的研究进展，并对未来发展趋势进行展望。

2 马尔可夫决策过程

RL 指 agent 通过不断试错来积累经验，探索优化状态到动作之间的映射，最终得到最优策略，同时最大化累积收益的过程。马尔可夫决策过程（Markov Decision Process，MDP）[9]是序贯决策问题的经典表达形式，是通过交互学习实现最终目标的理论框架。

MDP 一般被描述为五元组 (S, A, P, R, γ)，其中 S 代表有限状态集，A 代表有限动作集，P 代表状态转移概率，R 代表奖励函数，γ 代表用于计算整个过程累积收益的折扣因子[10-11]。式(1)、式(2)定义了 MDP 的过程，其中函数 P 为状态转移概率函数，E 为期望奖励函数。假设状态转换具有马尔可夫性，即转换到当前状态的概率仅与上一状态和上一动作有关。环境从状态 s 通过动作 a 转移至 s' 的概率 $P_{ss'}^{a}$ 和期望奖励 R_s^a 可表示为：

$$P_{ss'}^{a}=P(S_{t+1}=s' \mid S_t=s, A_t=a) \tag{1}$$

$$R_s^a=E(R_{t+1} \mid S_t=s, A_t=a) \tag{2}$$

在 RL 中，agent 的目标为最大化累积收益的期望值[4]。价值函数利用收益期望值评估当前状态或给定状态与动作下的 agent 表现。式(3)中，$v_\pi(s)$ 为策略 π 下状态 s 的状态价值函数。对于 MDP，$v_\pi(s)$ 可表示为：

$$v_\pi(s)=E_\pi(G_t \mid S_t=s) \tag{3}$$

其中，G_t 表示从当前状态开始到终止状态结束这个过程中所有收益按一定比例衰减的总和。因此，$v_\pi(s)$ 又可表示为：

$$\begin{aligned} v_\pi(s)&=E_\pi(R_{t+1}+\gamma R_{t+2}+\gamma^2 R_{t+3}+\cdots \mid S_t=s) \\ &=E_\pi(R_{t+1}+\gamma G_{t+1} \mid S_t=s) \\ &=E_\pi(R_{t+1}+\gamma v_\pi(S_{t+1}) \mid S_t=s) \end{aligned} \tag{4}$$

式(4)被称作v_π的贝尔曼方程。同理,式(5)中$q_\pi(s,a)$代表策略π下,状态s采取动作a的动作价值函数。对于MDP,$q_\pi(s,a)$可表示为:

$$q_\pi(s,a)=E_\pi(G_t|S_t=s,A_t=a)$$
$$=E_\pi(R_{t+1}+\gamma q_\pi(S_{t+1},A_{t+1})|S_t=s,A_t=a) \quad (5)$$

式(5)被称作q_π的贝尔曼方程。由以上贝尔曼方程可以看出,价值函数由该状态的即时奖励期望和下一状态的价值期望与衰减系数的乘积两部分组成。

为解决RL问题,可通过寻找较优的价值函数来得到较优策略,通常最优价值函数定义为所有策略下对应价值函数中的最大者,对应的策略即为最优策略,即:

$$v_*(s)=\max_\pi v_\pi(s) \quad (6)$$

$$q_*(s,a)=\max_\pi q_\pi(s,a) \quad (7)$$

其中,v_*代表最优状态价值函数,q_*代表最优动作价值函数。为了得出最优策略π_*,RL引入了贝尔曼最优方程,如式(8)、式(9)[12-13]所示。其阐述了最优策略对应的最优价值函数下,各状态的价值一定等于此状态下最优动作的收益期望。

$$v_*(s)=\max_a q_{\pi_*}(s,a)$$
$$=\max_a E_{\pi_*}(R_{t+1}+\gamma G_{t+1}|S_t=s,A_t=a)$$
$$=\max_a E(R_{t+1}+\gamma v_*(S_{t+1})|S_t=s,A_t=a)$$
$$=\max_a R_s^a+\gamma\sum_{s'\in s}P_{ss'}^a v_*(s') \quad (8)$$

$$q_*(s,a)=E(R_{t+1}+\gamma\max_{a'}q_*(S_{t+1},a')|S_t=s,A_t=a)$$
$$=R_s^a+\gamma\sum_{s'\in s}P_{ss'}^a\max_{a'}q_*(s',a') \quad (9)$$

3 无模型强化学习方法

在现实问题中,环境往往是不可预知的,无模型强化学习端到端的特点使其在各领域被广泛应用。无模型强化学习算法分为基于价值函数和基于策略函数两类[4]。本节以无模型强化学习中的经典算法为核心,分别综述其主要思想、具体流程以及相应的优缺点。

3.1 基于价值函数的方法

3.1.1 时序差分法

时序差分法(Temporal-Difference,TD)结合了动态规划和蒙特卡洛算法的优点,其不需要环境动态模型,也不需要完整的状态序列,而是根据贝尔曼方程得到的收益期望值近似表示收益。因此,TD只需要两个连续状态和对应的即时收益即可求解RL问题。

TD分为在线策略(on-policy)和离线策略(off-policy)两类学习算法,最常见的在线策略是SARSA算法,最常见的离线策略是Q-Learning算法,两者的区别主要在于:在线策略一般只有一个策略进行价值迭代,即ε-婪法;而离线策略一般有两个策略,ε-贪婪法用于选择动作,贪婪法用于更新价值函数。

SARSA算法由Singh等[14]提出,其中S代表状态(State),A代表动作(Action),R代表奖励(Reward)[15]。在迭代过程中,agent首先基于ε-贪婪法在当前状态s选择一个动作a,执行动作后环境向前推进一个时间步长并转移到下一个状态s',同时agent获得一个即时奖励R。在新的状态s',agent基于ε-贪婪法选择一个动作a',但并不执行,只用于更新价值函数$q(s,a)$[16-17],其中α为迭代步长。

$$q(s,a)=q(s,a)+\alpha(R+\gamma q(s',a')-q(s,a)) \quad (10)$$

Q-Learning算法与SARSA的不同之处在于,agent基于状态s'使用贪婪法选择动作,即直接选择Q值最大的动作来更新价值函数$q(s,a)$[18]。

$$q(s,a)=q(s,a)+\alpha(R+\gamma\max_a(q(s',a)-q(s,a))) \quad (11)$$

SARSA与Q-Learning算法的拓扑图如图2所示。

(a)SARSA算法拓扑图 (b)Q-Learning算法拓扑图

图2 两种算法的拓扑图

Fig. 2 Topology diagram of two algorithms

比较TD的两类算法,SARSA算法在学习最优策略的过程中不断鼓励探索新策略,学习过程较平滑,不易遇到Q-Learning算法中可能出现的最优"陷阱"[19]。但是,SARSA为了保证收敛性,需要逐渐减小ε-贪婪法中的探索率。

3.1.2 深度Q-Learning算法

当RL问题中状态维度较高或动作空间规模较大时,可采用近似表示价值函数的方法进行建模。函数$\hat{v}$和函数$\hat{q}$分别表示包含参数w的状态价值函数和动作价值函数,其接受状态s和动作a作为输入,从而计算得到近似的价值函数。

$$\hat{v}(s,w)\approx v_\pi(s) \quad (12)$$

$$\hat{q}(s,a,w)\approx q_\pi(s,a) \quad (13)$$

最常见的价值函数近似表示方法为深度神经网络,因此研究人员基于Q-Learning算法,结合深度神经网络提出了深度Q-Learning算法(Deep Q-Learning Network,DQN)。

DQN包含两个结构相同的Q网络。一个用于与环境进行交互,选择动作,并优化模型参数,被称为主Q网络;另一个用于计算目标Q值,优化主Q网络,被称为目标Q网络。为了减小目标Q值和当前Q值的相关性,主Q网络每隔若干时间步将网络参数传递给目标Q网络一次,即延时更新。DQN的另一个主要特点是使用了经验回放(experience replay)[20]。其将与环境交互得到的状态转移情况和对应得到的奖励作为经验数据存入经验回放池,用于计算目标Q值。由于目标Q值与通过主Q网络计算的Q值之间存在误差,因此使用均方误差梯度下降,并通过反向传播来更新主Q网络参数,从而求解RL问题。

DQN为了加速Q值向优化目标靠拢,使用max函数选择最优动作,容易导致过度估计[21]。因此,研究人员基于DQN算法提出了深度双Q网络算法(Double Deep Q-learning Network,DDQN)[22]、动态跳帧的DQN算法[23]、竞争网络结构(Dueling Network)[24]及利用循环神经网络(Recurrent Neural Network,RNN)的DRQN(Deep Recurrent Q-Learning Network)模型[25]等。

3.2 基于策略函数的方法

基于价值函数的RL方法在连续动作空间方面处理问题的能力不足,且无法解决随机策略问题,因此研究人员相继提

出了诸多基于策略函数的 RL 方法，其采用与价值函数近似表示相同的思路对策略函数进行拟合，此时策略 π 被描述为包含参数 θ 的策略函数：

$$\pi(a|s,\theta)\approx\pi(a|s) \tag{14}$$

策略梯度[26]通过端到端的方式不断计算当前策略下的 agent 累积收益与策略参数的梯度，最终梯度收敛得到最优策略。基于策略梯度的 RL 算法首先需要一个可最大化的目标函数，该函数可定义为初始状态收获的期望。但当没有明确初始状态时，该函数可定义为平均价值，或每一时间步的平均奖励。

在策略梯度方法中，只要 $\pi(a|s,\theta)$ 对参数可导，策略就可通过任意方式参数化。一般常用的策略参数化方法是 SoftMax 和高斯策略。SoftMax[27]通过描述状态和动作的特征来估计一个参数化的数值偏好 $h(s,a,\theta)$，从而得出动作发生的概率，其主要应用于离散动作空间。SoftMax 的分布如下：

$$\pi(a|s,\theta)=\frac{e^{h(s,a,\theta)}}{\sum_b e^{h(s,b,\theta)}} \tag{15}$$

其中，e 是自然对数的底，分母的作用是使在每个状态下选择动作的概率总和为 1。

高斯策略[28]对应地从高斯分布 $N(\varphi(s)^{\mathrm{T}}\theta,\sigma^2)$ 中产生动作，主要应用于连续动作空间。

3.3 基于价值策略结合的方法

基于价值函数的 RL 中，agent 每次选择一个最大价值动作来执行，易出现过度估计问题。而基于策略函数的 RL 为了使学习过程更加平滑，在策略选择时仅在某一参数梯度上缓慢优化，效率不高，且这类方法使用收益作为状态价值估计，噪声较大，具有较高的变异性。因此，研究人员提出了诸多价值函数和策略函数相结合的 RL 方法。

3.3.1 行动者评论家算法

基于价值函数和策略函数相结合的方法称为行动者评论家算法（Actor-Critic）[29]。Actor-Critic 包括 Actor 和 Critic 两部分，其中 Actor 基于策略函数，负责与环境交互进而选择动作；Critic 基于价值函数，负责评估 Actor 并指导其下一状态动作。在 Actor-Critic 算法中需要分别对策略函数和价值函数做近似表示[30]。

总的来说，Critic 计算状态最优价值 v_t，Actor 利用 v_t 迭代更新策略函数的参数，进而选择动作，从而得到即时奖励和下一状态，Critic 利用奖励和新的状态更新价值函数的参数。

3.3.2 异步的优势行动者评论家算法

异步的优势行动者评论家算法（Asynchronous Advantage Actor-critic，A3C）[31]利用多线程异步执行与环境进行交互。相比 Actor-Critic，A3C 的优点主要有 3 个方面。

（1）异步训练框架，具体如图 3 所示。Global Network 是包括 Actor 和 Critic 的公共神经网络模型，其包含若干 worker 线程，每个线程独立与环境进行交互得到经验数据。当每个线程获得一定量的经验后，当前线程计算神经网络损失函数梯度，更新公共的神经网络。公共神经网络参数每隔若干时间步被传递给线程一次。

（2）网络结构的优化。Actor-Critic 中使用了两个不同的网络，即 Actor 和 Critic。而 A3C 集合了两个网络，输入状态 S，即可输出状态价值和其对应的策略[32]。

（3）Critic 评估点的优化。Actor-Critic 中 Critic 以动作价值函数和状态价值函数的差值为评估点，即优势函数 A。不考虑参数的情况下，函数 A 在 t 时刻可表示为：

$$A(s,a,t)=q(s,a)-v(s) \tag{16}$$

其中，$q(s,a)$ 通过单步采样得到。

$$q(s,a)=R+\gamma v(s') \tag{17}$$

因此，Actor-Critic 中的 Critic 评估点表示为：

$$A(s,t)=R+\gamma v(s')-v(s) \tag{18}$$

而 A3C 使用 N 步采样来加速收敛效果。A3C 中的 Critic 评估点表示为：

$$A(s,t)=R_t+\gamma R_{(t+1)}+\cdots+\gamma^{(n-1)}R_{(t+n-1)}+\gamma^n v(s')-v(s) \tag{19}$$

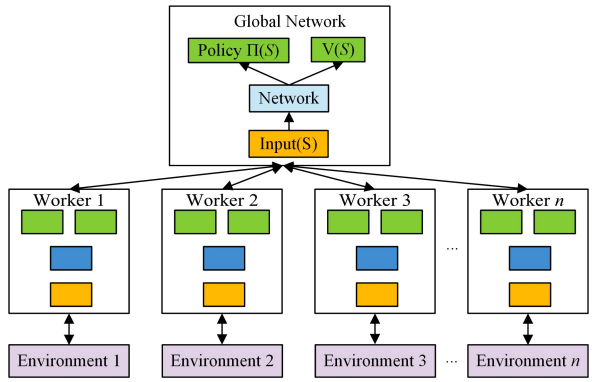

图 3 A3C 异步训练框架图

Fig. 3 A3C asynchronous training framework diagram

4 研究进展

RL/DRL 已在诸多领域取得了辉煌成就，DRL 从输入到输出的直接控制预测使无模型强化学习被应用到很多具有高维状态和动作空间的决策问题领域中[8]，例如游戏博弈[33-34]、机器人控制[35]、仿真模拟[36]等。无模型强化学习基于价值函数的各种方法，以 DQN 为基础，不断改进网络结构和训练方法。研究人员以策略梯度为基础，将基于策略函数的 DRL 方法与深度神经网络相结合，相继提出了 TD3[37]，TRPO[38]，NAF[39]，ACER[40]，PPO[41]等算法。同时，各种算法的并行化版本也陆续出现，更加适用于解决复杂的 RL 问题。本节将围绕游戏 AI、化学材料设计、自然语言处理和机器人控制领域对无模型强化学习近期的研究工作进行综述。

4.1 游戏 AI 领域的无模型强化学习

鉴于在高维状态和动作空间中学习过程的黑盒性质，无模型深度强化学习方法的知识表示方式一直是一项重大挑战。过去几年，DeepMind AlphaGo[42]的出现为解决上述问题提供了新思路。经过不断地深入研究，即时战略游戏（Real Time Strategy，RTS）成为了关注热点，星际争霸 AI AlphaStar[43]、OpenAI Five[44]等项目引起了全球人工智能社区的广泛关注。在国内，腾讯 AI Lab 也一直致力于应用无模型强化学习的 AI＋游戏研究。

4.1.1 新型分层宏观战略模型

与围棋相比，RTS 游戏的难度主要体现在计算复杂度

高、多智能体、信息不完整、奖励稀疏及延迟4个方面[45]。为了使agent掌握RTS游戏，RL模型需要在宏观策略操作和微观执行两个方面都具有较强的处理能力。然而，最近的研究大多集中于微观执行，在宏观策略操作方面一直缺乏完整的解决方案[46-48]。

2019年，腾讯AI Lab的Wu等[49]提出了一种基于学习的新型分层宏观战略(Hierarchical Macro Strategy，HMS)模型，用于掌握RTS游戏中的多人在线战术竞技(Multi-player Online Battle Arena，MOBA)游戏。通过HMS模型训练，agent可以明确做出宏观战略决策，并进一步指导其微观执行。宏观战略操作过程可以分为阶段识别、注意力预测、执行3个阶段。因此，Wu等提出双层宏观战略架构建模这一过程，其中包含阶段层和注意力层：阶段层用于识别当前agent的游戏阶段；注意力层用于预测地图上英雄的最佳位置。阶段层和注意力层为宏观执行提供高级指导。

此外，他们还提出了一种新型模仿跨agent通信机制，使agent在独立做出战略决策的同时能够相互合作[49]。经过训练的HMS模型可以用作无模型强化学习框架的初始策略。在MOBA游戏中，HMS模型与排行榜前1%的人类玩家团队竞技，5个AI agent的胜率可达48%，体现了AI的巨大潜力。同时，在大规模状态动作空间应用中，也体现了无模型强化学习的优势。

4.1.2 基于深度强化学习的游戏动作预测框架

在MOBA游戏中，多智能体1v1场景比传统1v1场景涉及的状态和动作空间的复杂度更高，导致探索到达人类水平的控制策略变得更加困难。

2020年，腾讯AI Lab的Ye等[50]提出利用无模型深度强化学习为agent预测游戏动作，设计了一种DRL框架，并对算法进行了优化。整个框架通过低耦合和高扩展性的模块设计，实现了大规模高效探索。DRL框架由RL学习系统、AI服务器、调度模块和内存池组成：1)RL学习系统是分布式训练环境，用于加速agent的训练更新；2)AI服务器用于与环境进行交互，收集经验数据；3)调度模块用于并行收集经验数据，由多个AI服务器构成；4)内存池用于存储收集的经验数据，并将其分发到RL学习系统。

针对RL算法设计，DRL框架采用Actor-Critic算法，其主要包含注意力机制、LSTM(Long Short-Term Memory)、改进的PPO算法和动作掩码(action mask)4个方面：1)注意力机制用于预测对战中的攻击目标选择；2)LSTM用于学习连续动作；3)改进的PPO算法指包含多标签PPO算法和大规模分布式训练的dual-clips PPO算法，用于解耦动作相关性，增加动作多样性，保证大批量处理过程的收敛性；4)动作掩码是在人类玩家的先验知识上提出的一种指导算法，能够降低训练难度，指导无模型强化学习的探索过程。

通过使用多种不同类型的英雄对比游戏AI和专业选手可知，AI在各项指标结果中都体现了强大优势，且在多场与顶级玩家的1v1竞技测试中胜率高达99.8%。

4.2 化学材料设计领域的无模型强化学习

目前，开发新材料的大多数方法都遵循缓慢的迭代过程，需要大量的时间与人力，成本昂贵且效率低下。因此，随着现代人工智能技术尤其是无模型强化学习的发展，这种固有的劳动密集型研究方法正在逐渐得到改变。

4.2.1 目的强化生成对抗网络

将无模型强化学习应用于新药物研发是当前DRL方法应用于化学领域的典型案例之一。2017年，Guimaraes等[51]提出了一种基于序列的目的强化生成对抗网络模型(Objective-Reinforced Generative Adversarial Networks，ORGAN)。ORGAN基于序列生成对抗网络模型(Sequence Generative Adversarial Networks，SeqGAN)[52]，将模型中的生成器作为无模型强化学习环境中的随机策略，并扩展了模型训练过程，其采用混合奖励机制，奖励值包含鉴别器奖励和特定目标值。同时，为了提高训练效果，该模型采用Wasserstein距离[53]作为鉴别器损失函数。

在基于分子的实验中，ORGAN模型采用SMILES(Simplified Molecular Input Line Entry Specification)分子表示方法。由于SMILES具有预先定义的语法规则，因此生成器生成的数据可能存在无效或重复分子，该模型将这类分子作为鉴别器的惩罚项之一。同时，为了引导生成器生成具有某些属性值的新分子，该模型选取了小分子药物研发的通用属性，如溶解度、药物相似性、合成性等作为鉴别器的另一惩罚项。最后，利用两种惩罚项的加权平均值优化生成器生成有效分子。在与最大似然估计(Maximum Likelihood Estimation，MLE)、SeqGAN和RL方法的比较实验中，该模型在保证样本多样性方面具有巨大优势。

4.2.2 深度强化学习拟合分子量分布

随着无模型强化学习在化学领域应用的增广，利用无模型强化学习方法控制化学反应的研究逐渐增多。2018年，Li等[54]提出了一种将无模型强化学习与原子转移自由基聚合(Atom Transfer Radical Polymerization，ATRP)数值模拟相结合的方法，获得聚合物分子量分布。该方法通过RL控制器控制化学反应进程中的投料时机，以获得指定产物分子量分布。RL控制器采用全连接和卷积神经网络结构，根据ATRP反应的当前状态进行决策。在使用Actor-Critic算法进行训练时，RL控制器能够发现并优化各种目标分子量分布(Molecular Weight Distributions，MWD)的控制策略。目标MWD包括不同宽度的高斯分布、双峰分布以及更多不同的形状。该方法学习到的控制策略具有较强的鲁棒性，即使在模拟噪声存在的情况下，也能转换到相似但不完全相同的ATRP反应设置，具有较强的泛化能力。

4.2.3 深度强化学习优化化学反应

除上述两个方面外，无模型强化学习在优化化学反应方面也有诸多应用。2017年，Zhou等[55]提出了一种利用无模型深度强化学习优化化学反应的深度反应优化器(Deep Reaction Optimizer，DRO)。该模型将化学反应结果作为经验数据，通过不断选择新的反应条件来优化反应结果。实验结果证明，该模型在模拟和实际反应中比最先进的黑盒优化算法(Covariance Matrix Adaption-Evolution Stratgey，CMA-ES[56])减少了71%的反应步骤。此外，该模型还提出了一种有效的探索策略，通过从特定的概率分布中提取反应条件，使模型的*regret*值从0.062下降到0.039。

DRO 模型通过尝试不同实验条件的组合,使反应以最佳的方式达到目标,并且期望步骤最少。该模型将所有反应条件组合的集合作为状态空间,将所有对反应条件更改的集合作为动作空间。实验采用一类非凸的"混合高斯密度函数"作为模拟反应环境,模型在特定的实验条件下反复进行实验,并记录最终产率。Zhou 等[55]在模型设计中还提出了一种改进的 LSTM 结构,结构中隐藏状态包含传递给下一时间步的历史经验数据,策略函数利用这些历史经验数据选择下一步实验条件。

4.3 自然语言处理领域的无模型强化学习

表征学习是自然语言处理(Natural Language Processing,NLP)中的一个基本问题。作为 NLP 最常见的任务之一,文本分类在很大程度上依赖于表征学习,并被广泛应用于知识问答、推荐系统和语言推理。

在现有的主流结构化表示模型中,结构主要作为模型的输入,或使用显式的结构标注进行监督预测,对于具有自动优化结构的学习表示的研究较少。2018 年,Zhang 等[57]提出了一种通过自动发现优化结构来学习句子表示的无模型强化学习方法。该方法在文本分类任务中应用了策略梯度算法来得到更好的句子结构化表征,从而得到更好的文本分类效果。

该方法由策略网络(Policy Network,PNet)、结构表示模型和分类网络(Classification Network,CNet)组成。PNet 针对每个单词的当前状态采用随机策略选择动作,然后产生整个句子的动作集合。结构表示模型执行动作集合,将句子结构化表示。其中,结构化表示模型分为两种:ID-LSTM(Information Distilled LSTM)和 HS-LSTM(Hierarchically Structured LSTM)。ID-LSTM 用于保留有用单词,删除无用单词,如"a""the"等。HS-LSTM 用于将整个序列划分为多个短语结构。CNet 使用交叉熵损失函数,基于结构化表示进行分类,并向 PNet 返回延迟奖励以指导对策略的学习。

实验结果表明该方法具有更先进的性能,并且能够在不需要显式结构注释的情况下发现与任务相关的结构。由于结构发现或重构输入的思想可以推广到其他任务和领域,因此可以将这种方法应用于其他类型的序列,继续进行研究。

4.4 机器人控制领域的无模型强化学习

为有腿机器人设计运动控制器是机器人控制领域面临的重大挑战之一。传统方法一般采用流水线方法,该方法通常需要一个十分精确的机器人动力学模型,然而设计动力学模型需要大量的专业知识,因此其一直是该领域发展的重大阻力之一。而无模型深度强化学习为解决上述问题提供了新的方向,依托于无模型强化学习不需要预先设定动力学模型的特点,研究人员不断将无模型强化学习应用于机器人控制领域,并且取得了巨大进展。

4.4.1 深度强化学习控制单类型机器人

AI 系统赋予机器人具备像人类一样掌握和操纵物体的能力已经逐渐成为可能。2019 年,Haarnoja 等[58]提出了一种基于最大熵强化学习的高效采样无模型深度强化学习算法和异步机器人 RL 系统。该工作不仅使机器人在改变任务时不需要做出过多调整,而且使神经网络的探索效率得到了很大提升。

RL 系统主要由 3 个部分组成,分别是收集机器人经验数据模块、根据机器人动作和测量得到的机器人位置计算奖励模块,以及更新神经网络模块。不同的模块之间异步运行,并使用时间戳同步未来数据流。在训练过程中,对于每一个控制步骤,系统通过收集观测状态,并将其输入神经网络来得到推理结果,然后执行动作以得到奖励。整个过程以元组形式记录到缓冲区,并通过不断迭代来学习最优动作。最终这项工作在真实的四足机器人上进行了测试,并取得了良好的性能。

4.4.2 深度强化学习控制多类型机器人

将无模型强化学习应用于机器人控制领域是当前解决复杂机器人控制的主要方法之一。2020 年,DeepMind 的 Hafner 等[59]提出了一个帮助有腿机器人学习多种复杂运动行为的无模型强化学习框架。整个框架依赖于一种处理数据高效、采用离线策略的多任务 RL 算法,并在整个实验过程中保持超参数设置和奖励设定不变。

该无模型强化学习框架依据机器人自身的陀螺仪和速度区分不同的运动行为,然后根据不同的运动行为设置不同的奖励。对于不同类型的机器人,该框架的动作空间以及观测空间也不完全相同。例如,针对三足机器人,该方法设置了 9 个动作空间和 127 个观测空间,而对于六足机器人,动作空间和观测空间分别上升到了 18 和 282。凭借多样化的动作空间以及观测空间,该框架在任务切换的过程中强迫 agent 访问不同的状态空间,以显著提高探索效率,并且不同任务之间的切换也显著提高了框架的健壮性。最终实验采用相同的 RL 算法对 9 种不同类型的机器人进行了测试,结果表明该框架使机器人对多种运动行为都拥有较强的学习能力,且不需要在任何特定平台或附加学习工具中进行调整。

5 未来展望

无模型强化学习是一个快速拓展的新兴热点,其中还有很多问题需要进一步研究,例如如何平衡学习过程中的探索与利用的问题、如何设计奖励机制以高效地达到收益最大化的问题等。

平衡探索与利用的问题是无模型强化学习中一个关键部分,在 DRL 中同样起到了很大作用。传统的探索与利用平衡问题一般采用多臂赌博机进行研究,如 ε-贪心方法、基于置信度上界的动作选择方法、梯度赌博机算法等[4]。然而,这些算法远远不能解决平衡探索与利用的问题。在 DRL 的背景下,Puigdomènech 等[60]于 2020 年基于 Atari 现有工作将 Q 函数分为用于近似环境的奖励和用于近似自发探索的奖励两个部分,并提出了一种自适应机制,再次刷新了 Atari 环境的 SOTA(state-of-the-art)结果,验证了平衡探索与利用的问题对算法性能起到了关键作用。此外,探索与利用平衡问题一直受限于环境中的内在随机性,且依赖于概率生成模型,随着近年来生成模型取得的巨大成果,其有很大的改良空间。

设计奖励机制的问题是无模型强化学习中的另一个关键部分,如果可以在训练过程中设计高效的奖励机制,则可以减少 agent 与环境的交互次数,提高样本利用率,从而降低实验成本。然而,奖励机制设计问题一直受限于不同的实验环境,基于直观解决思路,许多算法一般采用人为设计奖励的方式,

但其不具有通用性且易陷入局部最优[61]。针对人为设计奖励存在的问题，逆强化学习方法[62]于2000年被首次提出，研究人员转向从大量专家交互决策数据中逆向求解奖励函数，并基于DRL不断进行改进。但逆强化学习在奖励设计与学习过程中使用的专家轨迹不一定是最优的，因此奖励机制设计问题虽已取得一定进展，但在未来仍需进行进一步的研究。

同时，无模型强化学习在经验回放、多目标学习和辅助任务方面都具有巨大的研究潜力。将现有成果更好地应用到无模型强化学习领域，进行精准的价值函数预测和最优策略控制将是人工智能进步的强大推动力。

结束语 在很多现实问题中，无模型强化学习都取得了突破性进展。本文首先介绍了强化学习中的马尔可夫决策框架，然后基于价值函数和策略函数两方面对无模型强化学习经典算法进行了简要概述，最后介绍了无模型强化学习结合深度神经网络在游戏AI、化学材料设计、自然语言处理和机器人控制领域的创新应用，并对当前研究的不足和未来发展进行了展望。随着研究成果的不断涌现，可以预见无模型强化学习将为人工智能的发展提供有力的技术支撑。

参 考 文 献

[1] GAO Y, CHEN S F, LU X. Research on Reinforcement Learning Technology: A Review[J]. Acta Automatica Sinica, 2004, 30(1): 86-100.

[2] LECUN Y, BENGIO Y, HINTONG E, et al. Deep learning[J]. Nature, 2015, 521(7553): 436-444.

[3] HENDERSON P A, ISLAM R, BACHMANP, et al. Deep Reinforcement Learning that Matters[J]. arXiv: 1709.06560.

[4] SUTTON R S, BARTOA G. Reinforcement Learning: An Introduction[J]. IEEEE Transactions on Neural Network, 1998, 9(5): 1054.

[5] TANGKARATT V, MORI S, ZHAO T, et al. Model-based policy gradients with parameter-based exploration by least-squares conditional density estimation[J]. Neural Networks, 2014, 57: 128-140.

[6] WANG T, BAO X, CLAVERAI, et al. Benchmarking Model-Based Reinforcement Learning[J]. arXiv: 1907.02057.

[7] BARFOOT T D. State estimation for robotics[M]. Cambridge: Cambridge University Press, 2017.

[8] ZHAO T T, KONG L, HAN Y J, et al. A Review of Model-based Reinforcement Learning[J]. Journal of Frontiers of Computer Science & Technology, 2020, 14(6): 918-927.

[9] BELLMAN R. AMarkovian Decision Process [J]. Indiana University Mathematics Journal, 1957, 6(4): 679-684.

[10] SUTTON R S. Learning to predict by the method of temporal differences[J]. Machine Learning, 1988, 3(1): 9-44.

[11] ZHOU C H, XING Z H, LIU Z F, et al. Markov Decision Process Boundary Model Detection[J]. Chinese Journal of Computers, 2013, 36(12): 2587-2600.

[12] BELLMAHN R. Dynamic Programming[J]. Science, 1966, 153(3731): 34-37.

[13] RIDA M, MOUNCIF H, BOULMAKOUL A. Application of Markov Decision Processes for Modeling and Optimization of Decision-Making within a Container Port[J]. Soft Computing in Industrial Applications, 2011, 96: 349-358.

[14] SINGH S, JAAKKOLA T, LITTMAN M L. Convergence Results for Single-Step On-Policy Reinforcement-Learning Algorithms[J]. Machine Learning, 2000, 38(3): 287-308.

[15] WANG X, WANG F. A review of dynamic pricing strategy based on reinforcement learning[J]. Computer Applications and Software, 2019, 36(12): 1-6, 18.

[16] RUMMERY G A, NIRANJAN M. On-Line Q-Learning Using Connectionist Systems[R]. Department of Engineering, University of Cambridge, Cambridge, 1994.

[17] THAM C K. Modular on-line function approximation for scaling up reinforcement learning[D]. Cambridge: Cambridge University, 1994.

[18] WATKINS C, DAYAN P. Technical Note: Q-Learning[J]. Machine Learning, 1992, 8(3): 279-292.

[19] VAN SEIJEN H, VAN HASSELT H, WHITESON S, et al. A theoretical and empirical analysis of Expected Sarsa[C]// IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning. 2009: 177-184.

[20] MNIH V, KAVUKCUOGLU K, SILVER D, et al. Playing Atari with Deep Reinforcement Learning[J]. arXiv: 1312.5602.

[21] LAN Q, PAN Y, FYSHE A, et al. Maxmin Q-learning: Controlling the Estimation Bias of Q-learning[J]. arXiv: 2002.06487.

[22] VAN HASSELT H, GUEZ A, SILVER D, et al. Deep reinforcement learning with double Q-Learning[C]// National Conference on Artificial Intelligence. 2016: 2094-2100.

[23] LAKSHMINARAYANAN A S, SHARMA S, RAVINDRAN B, et al. Dynamic Frame skip Deep Q Network[J]. arXiv: 1605.05365.

[24] WANG Z, SCHAUL T, HESSEL M, et al. Dueling network architectures for deep reinforcement learning[C]// International Conference on Machine Learning. 2016: 1995-2003.

[25] HAUSKNECHT M, STONE P. Deep Recurrent Q-Learning for Partially Observable MDPs[J]. arXiv: 1507.06527v1.

[26] FORTUNATO M, AZAR M G, PIOT B, et al. Noisy Networks for Exploration[J]. arXiv: 1706.10295.

[27] ASADI K, LITTMAN M L. An alternative softmax operator for reinforcement learning[C]// International Conference on Machine Learning. 2017: 243-252.

[28] ENGEL Y, MANNOR S, MEIR R, et al. Reinforcement learning with Gaussian processes[C]// International Conference on Machine Learning. 2005: 201-208.

[29] SUTTON R S, MCALLESTER D, SINGH S, et al. Policy Gradient Methods for Reinforcement Learning with Function Approximation[C]// Neural Information Processing Systems. 1999: 1057-1063.

[30] KONDA V R, TSITSIKLIS J N. On Actor-Critic Algorithms [J]. Siam Journal on Control and Optimization, 2003, 42(4): 1143-1166.

[31] MNIH V, BADIA A P, MIRZA M, et al. Asynchronous methods for deep reinforcement learning[C]// International Conference on Machine Learning. 2016: 1928-1937.

[32] BABAEIZADEH M, FROSIO I, TYREE S, et al. Reinforcement

Learning through Asynchronous Advantage Actor-Critic on a GPU[J]. arXiv:1611.06256v3.

[33] TESAURO G. TD-Gammon, a self-teaching backgammon program, achieves master-level play[J]. Neural Computation, 1994, 6(2):215-219.

[34] WANG X, SANDHOLM T. Reinforcement Learning to Play an Optimal Nash Equilibrium in Team Markov Games[C]// Neural Information Processing Systems. 2002:1603-1610.

[35] KOBER J, BAGNELL J A, PETERS J, et al. Reinforcement learning in robotics: A survey[J]. The International Journal of Robotics Research, 2013, 32(11):1238-1274.

[36] FU Q M, LIU Q, WANG H, et al. A novel off policy Q(λ) algorithm based on linear function approximation[J]. Chinese Journal of Computers, 2014, 37:677-686.

[37] CHEN G. Merging Deterministic Policy Gradient Estimations with Varied Bias-Variance Tradeoff for Effective Deep Reinforcement Learning[J]. arXiv:1911.10527.

[38] SCHULMAN J, LEVINE S, ABBEEL P, et al. Trust Region Policy Optimization[C]// International Conference on Machine Learning. 2015:1889-1897.

[39] QI C, HUA Y, LI R, et al. Deep Reinforcement Learning With Discrete Normalized Advantage Functions for Resource Management in Network Slicing[J]. IEEE Communications Letters, 2019, 23(8):1337-1341.

[40] WANG Z, BAPST V, HEESS N, et al. Sample Efficient Actor-Critic with Experience Replay[J]. arXiv:1611.01224.

[41] SCHULMAN J, WOLSKI F, DHARIWALP, et al. Proximal Policy Optimization Algorithms[J]. arXiv:1707.06347v1.

[42] SILVER D, SCHRITTWIESER J, SIMONYAN K, et al. Mastering the game of Go without human knowledge[J]. Nature, 2017, 550(7676):354-359.

[43] ARULKUMARAN K, CULLY A, TOGELIUS J, et al. AlphaStar: an evolutionary computation perspective[C]// Genetic And Evolutionary Computation Conference. 2019:314-315.

[44] RAIMAN J, ZHANG S, WOLSKI F, et al. Long-Term Planning and Situational Awareness in OpenAI Five[J]. arXiv:1912.06721.

[45] VINYALS O, EWALDS T, BARTUNOV S, et al. StarCraft II: A New Challenge for Reinforcement Learning[J]. arXiv:1708.04782.

[46] TIAN Y, GONG Q, SHANG W, et al. ELF: An Extensive, Lightweight and Flexible Research Platform for Real-time Strategy Games[C]// Neural Information Processing Systems. 2017:2659-2669.

[47] SYNNAEVE G, BESSIERE P. A Bayesian model for RTS units control applied to StarCraft[C]// Computational Intelligence And Games. 2011:190-196.

[48] WENDER S, WATSON I. Applying reinforcement learning to small scale combat in the real-time strategy game StarCraft: Broodwar[C]// Computational Intelligence And Games. 2012:402-408.

[49] WU B, FU Q, LIANG J, et al. Hierarchical Macro Strategy Model for MOBA Game AI[J]. arXiv:1812.07887.

[50] YE D, LIU Z, SUN M, et al. Mastering Complex Control in MOBA Games with Deep Reinforcement Learning[C]// National Conference on Artificial Intelligence. 2020:6672-6679.

[51] GUIMARAES G L, SANCHEZLENGELING B, FARIAS P L, et al. Objective-Reinforced Generative Adversarial Networks (ORGAN) for Sequence Generation Models[J]. arXiv:1705.10843.

[52] YU L, ZHANG W, WANG J, et al. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient[C]// National Conference On Artificial Intelligence. 2016:2852-2858.

[53] ADLER J, LUNZ S. Banach Wasserstein GAN[C]// Neural Information Processing Systems. 2018:6755-6764.

[54] LI H, COLLINS C R, RIBELLI T G, et al. Tuning the molecular weight distribution from atom transfer radical polymerization using deep reinforcement learning[J]. Molecular Systems Design & Engineering, 2018, 3(3):496-508.

[55] ZHOU Z, LI X, ZARER N, et al. Optimizing Chemical Reactions with Deep Reinforcement Learning[J]. ACS central science, 2017, 3(12):1337-1344.

[56] GRAYVER A, KUVSHINOV A. Exploring equivalence domain in nonlinear inverse problems using Covariance Matrix Adaption Evolution Strategy (CMAES) and random sampling[J]. Geophysical Journal International, 2016, 205(2):971-987.

[57] ZHANG T, HUANG M, ZHAO L, et al. Learning Structured Representation for Text Classification via Reinforcement Learning[C]// National Conference on Artificial Intelligence. 2018:6053-6060.

[58] HAARNOJA T, HA S, ZHOU A, et al. Learning to Walk Via Deep Reinforcement Learning[J]. arXiv:1812.11103.

[59] HAFNER R, HERTWECK T, KLPPNERP, et al. Towards General and Autonomous Learning of Core Skills: A Case Study in Locomotion[J]. arXiv:2008.12228.

[60] PUIGDOMÈNECH B A, PIOT B, KAPTUROWSKI S, et al. Agent57: Outperforming the Atari Human Benchmark[J]. arXiv:2003.13350v1.

[61] OPENA I. Faulty Reward Functions in the Wild[EB/OL]. https://openai.com/blog/faulty-reward-functions. 2017.

[62] NG A Y, RUSSELLS J. Algorithms for inverse reinforcement learning[C]// International Conference on Machine Learning. 2000:663-670.

QIN Zhi-hui, born in 1996, postgraduate. Her main research interests include reinforcement learning and computational chemistry.

LIU Xiu-lei, born in 1981, Ph.D, associate professor, is a member of China Computer Federation. His main research interests include semantic sensor, semantic web, knowledge graph, semantic information retrieval, and so on.