控制与决策
Control and Decision

# 基于过滤机制筛选信息的多智能体策略方法

陈亮, 郭婷, 刘韵婷, 杨佳明

引用本文:
陈亮,郭婷,刘韵婷,杨佳明. 基于过滤机制筛选信息的多智能体策略方法[J]. 控制与决策, 2022, 37(6): 1643−1648.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2020.1139

---

# 基于过滤机制筛选信息的多智能体策略方法

陈 亮, 郭 婷, 刘韵婷†, 杨佳明

(沈阳理工大学 自动化与电气工程学院，沈阳 110159)

**摘 要:** 多智能体系统在进行协作或竞争时,会面临联合信息空间扩大、智能体间信息提取效率降低的问题. 对此,采用增加过滤机制来筛选信息的多智能体强化学习策略方法(FMAC),以增强智能体间信息交流能力. 该方法通过找到彼此相关联的智能体,根据相关性计算智能体的信息贡献,过滤掉无关智能体信息,从而实现在合作、竞争或者混合环境下智能体间有效的沟通. 与此同时,采用集中训练分散执行的方式解决环境的非平稳性问题. 通过对比算法进行实验,结果表明改进算法提高了策略迭代效率以及泛化能力,并且智能体数量增多时仍可保持稳定的效果,有助于将多智能体强化学习应用到更广泛的领域.

**关键词:** 强化学习；多智能体决策；信息过滤；集中训练分散执行

**中图分类号:** TP273 **文献标志码:** A

**DOI:** 10.13195/j.kzyjc.2020.1139 **开放科学(资源服务)标识码(OSID):**

**引用格式:** 陈亮,郭婷,刘韵婷,等. 基于过滤机制筛选信息的多智能体策略方法[J]. 控制与决策, 2022, 37(6): 1643-1648.

# Research on multi-agent strategy based on filtering mechanism to filter information

*CHEN Liang, GUO Ting, LIU Yun-ting†, YANG Jia-ming*

(School of Automation and Electrical Engineering，Shenyang Ligong University，Shenyang 110159，China)

**Abstract:** When multi-agent systems cooperate or compete, the joint information space will be enlarged and the efficiency of information extraction between agents will be reduced. In this paper, a multi-agent reinforcement learning strategy (FMAC) with filtering mechanism to filter information is adopted to enhance the ability of information communication between agents. By finding the related agents and calculating their information contribution according to the correlation, the method filters out the irrelevant agent information so as to realize the effective communication between agents in cooperative competition or mixed environment. At the same time, the centralized training decentralized execution method is adopted to solve the non-stationarity of environment. In this paper, experiments are carried out by comparing algorithms to verify that the improved algorithm improves the strategy iteration efficiency and generalization ability, and can maintain stable effects when the number of agents increases, which is conducive to the application of multi-agent reinforcement learning to a wider range of fields.

**Keywords:** reinforcement learning; multi-agent system; filtering mechanism; centralized training decentralized execution

## 0 引 言

强化学习(reinforcement learning, RL)是机器学习的一个分支,主要是通过智能体与环境的交互,根据环境的反馈改变自己的行为策略,目的是在互动过程中使智能体获得最大奖励[1]. 如今,强化学习在很多领域有很好的应用,如Atari games、机器人、工业等领域[2],但是RL的大部分成功都是单智能体情况,单智能体在很大程度上不需要建模或预测环境中其他参与者的行为. 而实际生活中,智能体常常需要在动态的复杂环境中与其他智能体进行合作或竞争,即多智能体系统(multi-agent system, MAS)[3-4].

在传统的多智能体强化学习方法中,一类是将其他智能体视为环境的一部分,但在训练过程中,智能体的策略会随着时间、环境等信息的改变而改变,这就造成了环境是非平稳的[5-6],即不满足马尔科夫决策[7]. 对于解决此类问题,可采取集中训练分散执行方法. 此类方法是在Actor-critic算法基础上[8-10]加以改进,在训练时,每个智能体的评价网络都共享全部环境信息,测试时由各自的策略网络与局部环境进行交互[11-12]. 但是在实际应用中,如果让所有智能体都

**收稿日期:** 2020-08-17；**录用日期:** 2021-02-10.

†通讯作者. E-mail: liuyunting0224@163.com.

彼此交流,无差别地接收大量信息,则需要很高的带宽和长时间的延迟以及高计算复杂性[13-15],因此智能体之间有效的沟通成为又一重要研究方向[16].

近几年,集中训练分散执行的多智能体强化学习算法有很多,MADDPG(multi-agent deep deterministic policy gradient)[17]是此类算法开山之作. 但是,此算法实际测试时策略网络常常会遇到从未训练过的情况,MADDPG只能依靠评价网络曾经训练的方法去面对新的问题,根据得到的动作状态值函数更新策略网络,并且MADDPG是根据记忆库(replay buffer)中的数据训练各个网络的参数,这样也会导致模型泛化能力较弱.

为提高智能体间沟通效果,提高信息利用效率,MAAC(actor-attention-critic for multi-agent)[18]算法在MADDPG基础上进行了改进. 该算法也是采用集中训练分散执行的方式,并且引入注意力机制[19-20]帮助智能体有侧重点地接收其他智能体的信息,从而在复杂的交互过程中提高性能. MAAC也结合了COMA(counterfactual multi-agent policy gradients)的优势函数的思想解决信用分配问题. 虽然MAAC可以对信息进行侧重点的收录,但是对于不利于合作效果的智能体情况,采取的对策是少采纳而不是全部屏蔽,这样会导致智能体数目较多且环境更为复杂时,合作效果不理想.

本文提出一种增加过滤机制筛选信息的多智能体强化学习策略方法(filtering mechanism actor-critic,FMAC). 面对多智能体间信息交流的重要性,通过提高智能体获取的信息质量,实现增强多智能体决策能力[21]. 具体方法是从智能体信息关联性入手,通过结合注意力机制思想,对智能体信息进行相关性处理,找到有“关系”的智能体之后,筛选出对每个智能体更为有用的信息,实现信息的动态选择. 其后根据相关程度计算“有关”智能体的信息贡献,从而实现有针对性的更新策略,让智能体在复杂的交互过程中提高决策能力,有助于实现多智能体协作等目标.

## 1 基于过滤机制的多智能体强化学习策略

本文是在先前算法基础上进行改进,核心思路是基于信息过滤机制的集中训练分散执行的actor-critic. 在多智能体环境中,通过计算智能体间信息相关性判断智能体关联程度. 根据智能体的信息贡献计算优势函数,使智能体可以有针对性地更新策略. 此方法可以很好地解决智能体间信用分配问题,并且可以提高算法的泛化能力.

### 1.1 集中训练分散执行的actor-critic

整体采用的训练框架是actor-critic框架.

1) 集中训练.

① 训练时,评价网络根据$Q$估计与$Q$现实的平方差求梯度来进行训练,即最小化loss. 策略网络根据评价网络的反馈更新策略,当智能体数量较多时,为了得到更准确的$Q$值,策略网络共享所有智能体的动作、状态数据,通过其他智能体的信息更准确地评估当前动作.

② 测试时,只需策略网络与环境交互,根据当前状态选择一个动作.

2) 分散执行.

网络训练完毕之后,每个智能体的评价网络根据当前的状态、观测的局部环境,可采取合适的动作,不需要了解其他智能体的动作状态值.

### 1.2 基于过滤机制的强化学习算法结构

算法整体采用基于值的集中训练分散执行方法,即多个智能体共用1个评价网络进行学习,执行时各自用策略网络与局部环境进行交互,整体结构见图1.

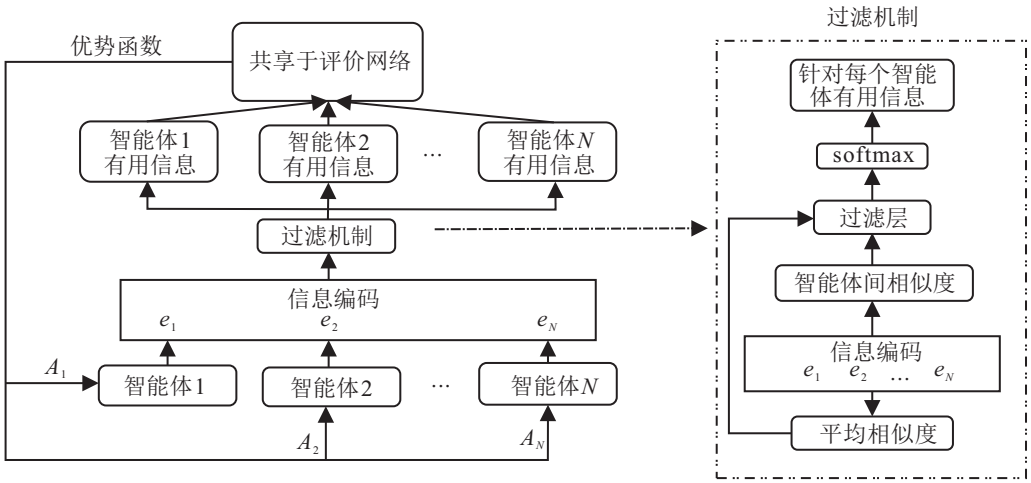



**图1** FMAC结构

智能体数量为$N$,第$i$个智能体观察到局部环境信息为$s_i$,策略网络根据此时的策略$\mu_i$与环境进行交互做出动作$a_i$,将这些信息进行编码$e_i = g_i(o_i, a_i)$以方便后期计算智能体相关性,其中$g_i$是一层感知机的嵌入式函数. 编码后的信息经过过滤机制,找到彼此有“关系”的智能体,筛选出对每个智能体有用的信息,信息整合之后通过评价网络计算各自的优势函数,以更新策略网络.

下面具体介绍过滤机制的工作原理以及如何通过中心化方法进行策略迭代更新参数.

### 1.3 过滤机制计算智能体信息贡献

过滤机制的作用是为每个智能体筛选出较为有用的信息,提高智能体间信息的利用效率,从而为智能体间合作提供有效帮助.

首先,将每个智能体的状态和动作信息进行编码,通过注意力机制[22-24]计算出其余智能体对智能体$i$的相似度,即智能体间的注意力权重$c_{i,j} = e_j^{\mathrm{T}} w_k^{\mathrm{T}} w_q e_i$. 其中: $w_q$是将$e_i$转化为查询键,$w_k$将$e_j$转化为特征键[25]. 将其余智能体对智能体$i$的平均相似度作为关系基线$B_i = \dfrac{1}{N-1}\sum_{j\neq 1} c_{i,j}$. 通过计算关系函数$R_{i,j} = c_{i,j} - B_i$作为评价智能体间是否存在“关系”的重要指标. 若$R_{i,j} > 0$,则表明智能体$j$对智能体$i$的注意力权重高于均值,视为存在一定“关系”; 若$R_{i,j} < 0$,则视为二者无关. 其后,在过滤层处,“有关”智能体注意力权重$c_{i,j}$不变,与智能体$i$无关的智能体在此处会将注意力权重设置为负无穷,这样经过后面softmax层后值接近于0,使无关智能体信息数据被过滤掉. 通过这种方法可以提高智能体间信息利用效率,有助于智能体间有效沟通.

经过过滤层后,智能体为了更好地利用各自的有用信息,对$Q$函数进行改进. 智能体$i$的$Q$函数$Q_i^{\mu}(0, a)$包含与$i$有关智能体的信息贡献$x_i$以及智能体$i$的观察和动作信息$e_i$,即

$$Q_i^{\mu}(0, a) = f_i(e_i, x_i). \tag{1}$$

其中: $f_i$是一个两层的多层感知器,与智能体$i$有关智能体的信息贡献$x_i$计算如下:

$$x_i = \sum_{j\neq 1} c_{i,j} h(V e_j). \tag{2}$$

其中: $V$是共享矩阵,$h$是ReLU激活函数.

通过以上操作可发现,注意力权重越大表示智能体间“关系”越紧密,获取的信息越多,反之获得信息越小,通过此方式即可为每个智能体筛选出对其有用的信息,之后输入到评价网络,计算出各智能体的优势函数,让智能体有针对性地进行更新.

### 1.4 计算优势函数解决信用分配

在DQN(deep Q-learning)中,用动作值函数与该状态下的状态值函数作比较表示优势函数. 优势函数可以评价当前行为的好坏. 若当前行为比此状态的平均值好,则朝梯度方向更新; 否则说明此行为比该状态的平均值更差,梯度将朝相反的方向更进一步.

在多智能体系统中,优势函数应计算的是智能体遵循当前策略进行决策得到的全局回报与一个反事实基准[26]之间的差值,简言之,比较当前智能体采取策略$\mu$与采取默认动作的好坏. 然而,这个默认动作(反事实基准)并不易表示,所以反事实基准通过计算中心化值函数关于该智能体局部观测以及动作的边缘分布的期望值来表示. 多智能体系统中优势函数计算如下:

$$\begin{aligned} &A_i(o, a) = Q_i^{\mu}(o, a) - b(o, a_{\setminus i}),\\ &b(o, a_{\setminus i}) = E_{a_i\sim\pi(o_i)}[Q_i^{\mu}(o, a_i, a_{\setminus i})] = \\ &\sum_{a_i'\in A_i} \pi(o_i, a_i) Q_i^{\mu}(o, a_i, a_{\setminus i}). \end{aligned} \tag{3}$$

其中: $\sum_{a_i'\in A_i} \pi(o_i, a_i) Q_i^{\mu}(o, a_i, a_{\setminus i})$是反事实基线公式,表示当前策略下所有可能的动作. 通过优势函数可计算出当前智能体对其他智能体而言是否有贡献,后期策略迭代时调用优势函数可以让智能体有侧重点地进行更新,从而解决置信分配问题,并且可以提高系统的泛化能力.

### 1.5 有侧重点地策略迭代

为解决非平稳性问题,算法采用集中训练分散执行方法. 策略网络的目标是最大化$Q$函数,利用梯度上升法更新参数,即

$$\begin{aligned} &\nabla_{\mu_4} J(\mu_i) = \\ &E_{o\sim D, a\sim\pi}[\nabla_{\mu_i} \log(\pi(o_i, a_i)) A_i(o, a)] = \\ &E_{o\sim D, a\sim\pi}[\nabla_{\mu_i} \log(\pi_{\mu_i}(o_i, a_i))(Q_i^{\mu}(o, a) - \\ &b(o, a_j) - \alpha \log(\pi_{\mu_i}(o_i, a_i)))]. \end{aligned} \tag{4}$$

其中: $\mu$是策略估计网络参数,$\alpha$是温度系数,此处引入了最大熵[27]来提高算法的泛化能力. 因为最大熵会使策略在输出动作时分布更加均匀,探索更多可能情况进行学习,所以学习的策略可以应对更多的复杂情况,增强模型探索能力,提高抗干扰能力. 同时也因可以学习到更多接近最优的行为,并且选择这些近似最优动作的概率相同,从而提高学习速度. 更新公式

中引入了优势函数,因此不同智能体参数更新效率不同,从而实现了智能体有侧重点地更新. 同时,策略网络是对当前智能体的所有动作进行采样,不局限于从记忆库中进行采样,因此可以对当前策略进行泛化.

评价网络目标为最小化损失,利用梯度下降法更新参数

$$L_Q(\theta)=\sum_{i=1}^{N}E_{(o,ay,o')\sim D}[(y_i-Q_i^{\theta}(o,a))^2],$$

$$y_i=$$

$$r_i+\nu E_{a'\sim\pi_{\bar{\mu}(o)}}[Q_i^{\bar{\theta}}(o',a')-\alpha\log(\pi\overline{\mu_i}(o_i',d_i'))]. \quad (5)$$

其中:$\theta$、$\bar{\theta}$、$\bar{\mu}$分别是评价的估计网络、目标网络和策略的目标网络参数,在评价网络目标函数中同样引入最大熵,用来平衡最大熵和回报.

## 2 实验及结果分析

### 2.1 实验环境及设置

为了测试和对比算法性能,本文采用OpenAI的multiagent particle envs作为测试平台. 该平台是具有连续空间和离散时间的二维世界,智能体可以在环境中采取物理动作,是用来测试多智能体强化学习算法的理想推演平台. 在simple spread和simple tag两个环境中进行对比.

针对simple tag环境,本文也做了一些变化,simple tag实验目标是红色智能体不碰撞黑色障碍物的同时围捕绿色智能体. 因为原环境代码版本中没有对智能体位置进行限制,这样会导致训练过程中出现智能体逐渐飞出画面的现象,所以在实验中对智能体位置进行设置,当超过一定边界就终止本轮进入下一轮.

在实验中,本文算法学习率为0.001,折损因子为0.96,每轮设置最大回合数300. 选取学习率时,大学习率收敛快、稳定性差,小学习率收敛慢、浪费时间,本文采取与DDPG相同的学习率0.001. 对于折损因子也遵循同样原则,过高会导致智能体过分小心,过低会让智能体肆无忌惮地探索,本文选取折中大小. 回合数不宜设置过大,若回合数较多,则环境一直不能得到“结束”的状态,可能会使智能体陷在某个状态出不来. 本文算法在上述两个环境下与MADDPG、MAAC进行对比,首先对比3种算法的基本信息,如表1所示.

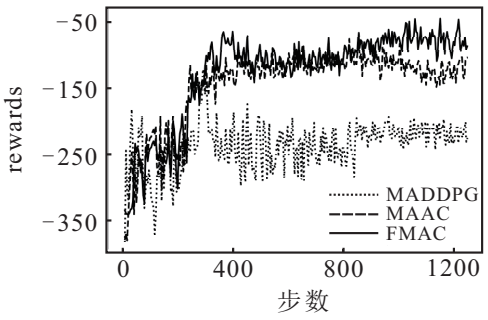
从表1中可见,MADDPG算法评价网络和策略网络数量与智能体数量相同,即有多少智能体便需配备相同数量的策略网络和评价网络,而FMAC与MAAC都是共享一个评价网络,因此FMAC和MAAC比MADDPG更能适应智能体较多的情况.

**表1 3种算法基本信息**

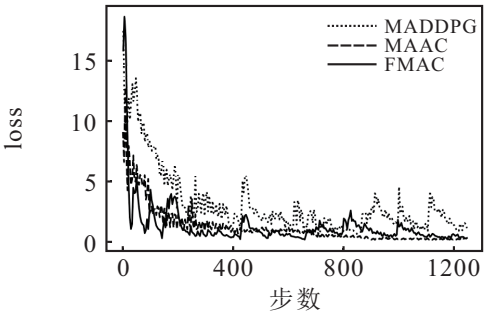
| 测试算法 | 基础算法 | 实现MARL原理 | 评价网络数量/策略网络数量($N$为智能体数量) |
|---|---|---|---|
| FMAC(本文) | MAAC | 过滤机制筛选信息 | $1/N$ |
| MADDPG | DDPG | 多个DDPG连接 | $N/N$ |
| MAAC | MADDPG | 注意力机制 | $1/N$ |

### 2.2 实验结果与分析

在simple spread环境中, MADDPG、MAAC、FMAC模型进行对比实验的回报和损失曲线如图2所示.




(a) simple spread奖励曲线



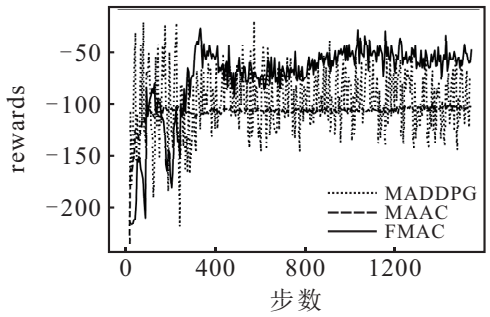

(b) simple spread损失曲线

**图2 simple spread奖励对比**

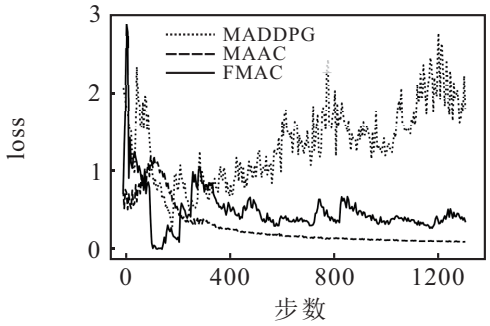
从图2(a)可见:MADDPG获得奖励能力低于MAAC和FMAC. 在交互趋于稳定时,每轮中FMAC获得奖励在6～7分之间,MAAC得分5.5分左右,MADDPG得分4分左右. 图2(b)在迭代效率方面:MAAC收敛最快,在50步左右收敛;FMAC损失收敛速度与之相近,70步左右收敛;MADDPG较慢,并且一直出现小范围的震荡.

图3是在围捕环境simple tag中的对比实验结果. simple tag较simple spread提升了环境复杂度,simple spread的目标是各自捕捉一个黑色目标,而simple tag需要在躲避障碍物的同时通过合作进行围捕获得奖励,增加了躲避和围捕环节. 随着环境复杂程度的提高,3个模型稳定性都有所变化. 图3(a)在获得奖励方面:MADDPG震荡明显,MAA和FMAC总

体获得奖励效果明显比MADDPG好,并且训练稳定后FMAC奖励比MAAC高出2分左右. 从损失曲线可见:MADDPG收敛速度较慢,甚至出现发散现象,MAAC收敛比在simple spread环境中慢,在100步左右保持稳定;FMAC在180步和220步出现小范围震荡.




(a) simple_tag奖励曲线




(b) simple_tag损失曲线

**图 3**　simple tag损失对比

MADDPG模型是通过经验回放进行采样,这样会导致智能体无法对当前策略进行调整,适应性较差,因此会出现获得奖励震荡的现象. 而本文算法FMAC是从当前智能体所有动作进行采样,从而提高了泛化能力. 当实验环境变为复杂时,也能较好地适应.

从奖励角度分析,MAAC只是通过注意力机制进行信息选择,相关性较小的智能体信息虽然贡献小,但是也被收录,这样会导致无助于合作的信息也被少量获取,而FMAC将无用信息直接过滤掉,如游离于逃跑者较远的围捕者信息即为无用信息,信息利用更为有效,提高了沟通能力,从而有助于协作,因此会出现FMAC获得奖励常常高于MAAC的情况,并且随着环境复杂程度的提高,两种算法得分差距随之拉大,也是FMAC过滤机制起效的体现.

为了进一步评估性能,在围捕游戏simple tag中,增加逃跑者数量实验,每个游戏进行300次测试,用0-1标准化的平均围捕者得分来显示. 如表2所示:MADDPG随着逃跑者数量增多,得分下降严重,3vs3时得分最低,为0.27;MAAC算法3vs1时得分最高,其值为0.69;当逃跑者数量最多,即为3个时,FMAC得分最高,其值为0.52. FMAC算法随着逃跑者数量增多,得分虽有变化,但在3个算法中表现最稳定,也验证了算法过滤机制可提高信息利用的有效性以及泛化能力.

**表 2**　simple tag不同逃跑者数量3种算法得分对比

| 算法 | 围捕者 vs 逃跑者 | | |
|---|---|---|---|
| | 3vs1 | 3vs2 | 3vs3 |
| MADDPG | 0.45 | 0.38 | 0.27 |
| MAAC | 0.69 | 0.5 | 0.45 |
| FMAC | 0.64 | 0.55 | 0.52 |

## 3　结　论

本文提出了一种基于过滤机制筛选信息的多智能体策略方法FMAC. 其过程是首先将所有智能体信息进行编码,通过过滤机制筛掉无关智能体的注意力权重. 其后,根据保留的注意力权重计算有关智能体的信息贡献,并输入到评价网络获得优势函数,从而实现有针对性的策略迭代. 通过本文方法可以有效利用各个智能体间的信息,提高沟通能力,实现合作决策. 实验结果表明,算法有较高的迭代效率和泛化能力,可以明显提高奖励得分,并且在复杂环境和智能体数量增多的情况下,优势更为突出. 针对FMAC损失偶尔出现震荡的问题,可以从过滤机制的评判原则入手,找到一个更为适合的基线,因此也是本算法以后的研究方向.

## 参考文献(References)

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

[2] Arulkumaran K, Deisenroth M P, Brundage M, et al. Deep reinforcement learning: A brief survey[J]. IEEE Signal Processing Magazine, 2017, 34(6): 26-38.

[3] 赵志宏, 高阳, 骆斌, 等. 多Agent系统中强化学习的研究现状和发展趋势[J]. 计算机科学, 2004, 31(3): 23-27.
(Zhao Z H, Gao Y, Luo B, et al. Reinforcement learning technology in multi-agent system[J]. Computer Science, 2004, 31(3): 23-27.)

[4] Anderson B D O, Yu C B, Fidan B, et al. Rigid graph control architectures for autonomous formations[J]. IEEE Control Systems Magazine, 2008, 28(6): 48-63.

[5] Hernandez-Leal P, Kaisers M, Baarslag T, et al. A survey of learning in multiagent environments: Dealing with

non-stationarity[J/OL]. 2017, arXiv:1707.09183.

[6] Matignon L, Laurent G J, Le Fort-Piat N. Independent reinforcement learners in cooperative Markov games: A survey regarding coordination problems[J]. The Knowledge Engineering Review, 2012, 27(1): 1-31.

[7] 张健, 潘耀宗, 杨海涛, 等. 基于蒙特卡洛Q值函数的多智能体决策方法[J]. 控制与决策, 2020, 35(3): 637-644.
(Zhang J, Pan Y Z, Yang H T, et al. Multi-agent decision making using Monte Carlo Q-value function[J]. Control and Decision, 2020, 35(3): 637-644.)

[8] Littman M L. Markov games as a framework for multi-agent reinforcement learning[M]. Machine Learning Proceedings. Amsterdam: Elsevier, 1994: 157-163.

[9] Konda V, Tsitsiklis J. Actor-critic algorithms[J]. SIAM Journal on Control and Optimization, 2003, 42(4): 1143-1166.

[10] Kraemer L, Banerjee B. Multi-agent reinforcement learning as a rehearsal for decentralized planning[J]. Neurocomputing, 2016, 190: 82-94.

[11] Peters J, Schaal S. Natural actor-critic[J]. Neurocomputing, 2008, 71(7/8/9): 1180-1190.

[12] Bhatnagar S, Sutton R S, Ghavamzadeh M, et al. Natural actor-critic algorithms[J]. Automatica, 2009, 45(11): 2471-2482.

[13] Ghavamzadeh M, Engel Y. Bayesian actor-critic algorithms[C]. Proceedings of the 24th International Conference on Machine Learning. New York: ACM Press, 2007: 20-24.

[14] Lowe R, Wu Y, Tamar A, et al. Multi-agent actor-critic for mixed cooperative-competitive environments[J]. Conference and Workshop on Neural Information Processing System(NeurIPS), 2017, 1: 6379-6390.

[15] Liu S Q, Geng M Y, Xu K L. Learning to communicate efficiently with group division in decentralized multi-agent cooperation[C]. 2019 IEEE International Conference on Service-Oriented System Engineering (SOSE). San Francisco, 2019: 331-337.

[16] Cheney D L, Seyfarth R M. Constraints and preadaptations in the earliest stages of language evolution[J]. The Linguistic Review, 2005, 22(2/3/4): 135-159.

[17] Sainbayar Sukhbaatar, Authur Szlam, Rob Ferus. Learning multiagent communication with backpropagation[J/OL]. 2016, arXiv: 1605.07736.

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

[19] Jeon W, Barde P, Nowrouzezahrai D, et al. Scalable multi-agent inverse reinforcement learning via actor-attention-critic[J/OL]. 2020, arXiv: 2002.10525.

[20] Laëtitia Matignon, Laurent Jeanpierre, Abdel-Illah Mouaddib, et al. Coordinated multi-robot exploration under communication constraints using decentralized markov decision processes[C]. Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence. Toronto, 2012: 2017-2023.

[21] Vinyals O, Kaiser, Koo T, et al. Grammar as a foreign language[J]. Advances in Neural Information Processing Systems, 2015, 28: 2773-2781.

[22] Luong M T, Pham H, Manning C D. Effective approaches to attention-based neural machine translation[J/OL]. 2015, arXiv: 1508.04025.

[23] Li D, Huang Q, He X, et al. Generating diverse and accurate visual captions by comparative adversarial learning[J/OL]. 2018, arXiv: 1804.00861.

[24] Hermann K M, Kocisky T, Grefenstette E, et al. Teaching machines to read and comprehend[J]. Advances in Neural Information Processing Systems, 2015, 28: 1693-1701.

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

[26] Le H M, Yue Y, Carr P, et al. Coordinated multi-agent imitation learning[J/OL]. 2017, arXiv:1703.03121.

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

## 作者简介

陈亮(1979−), 男, 副教授, 博士, 从事嵌入式仪器仪表、人工智能等研究, E-mail: kongkuchen@126.com;

郭婷(1991−), 女, 硕士生, 从事人工智能的研究, E-mail: 392547526@qq.com;

刘韵婷(1983−), 女, 副教授, 博士, 从事深度学习、数据分析及无线传感器网络等研究, E-mail: liuyunting0224@163.com;

杨佳明(1996−), 女, 硕士生, 从事人工智能、强化学习的研究, E-mail: 2439025742@qq.com.

(责任编辑：闫 妍)