计算机科学
COMPUTER SCIENCE

基于多智能体强化学习的端到端合作的自适应奖励方法

史殿习，赵琛然，张耀文，杨绍武，张拥军

引用本文

史殿习，赵琛然，张耀文，杨绍武，张拥军. 基于多智能体强化学习的端到端合作的自适应奖励方法[J]. 计算机科学，2022, 49(8): 247-256.

SHI Dian-xi, ZHAO Chen-ran, ZHANG Yao-wen, YANG Shao-wu, ZHANG Yong-jun. Adaptive Reward Method for End-to-End Cooperation Based on Multi-agent Reinforcement Learning[J]. Computer Science, 2022, 49(8): 247-256.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210700100

基于多智能体强化学习的端到端合作的自适应奖励方法

史殿习[1,2,4]　赵琛然[1]　张耀文[3]　杨绍武[1]　张拥军[2]
1 国防科技大学计算机学院　长沙 410073
2 军事科学院国防科技创新研究院　北京 100166
3 中国人民解放军 32282 部队　济南 250000
4 天津(滨海)人工智能创新中心　天津 300457
(dxshi@nudt.edu.cn)

摘　要　目前,多智能体强化学习算法大多采用集中训练分布执行的方法,且在同构多智能体系统中取得了良好的效果。但是,由不同角色构成的异构多智能体系统往往存在信用分配问题,导致智能体很难学习到有效的合作策略。针对上述问题,提出了一种基于多智能体强化学习的端到端合作的自适应奖励方法,该方法能够促进智能体之间合作策略的生成。首先,提出了一种批正则化网络,该网络采用图神经网络对异构多智能体合作关系进行建模,利用注意力机制对关键信息进行权重计算,使用批正则化方法对生成的特征向量进行有效融合,使算法向正确的学习方向进行优化和反向传播,进而有效提升异构多智能体合作策略生成的性能;其次,基于演员-评论家方法,提出了一种双层优化的自适应奖励网络,将稀疏奖励转化为连续奖励,引导智能体根据场上形势生成合作策略。通过实验对比了当前主流的多智能体强化学习算法,结果表明,所提算法在"合作-博弈"场景中取得了显著效果,通过对策略-奖励-行为相关性的可视化分析,进一步验证了所提算法的有效性。

关键词:多智能体强化学习;图注意力网络;自适应内在奖励
中图法分类号　TP391

Adaptive Reward Method for End-to-End Cooperation Based on Multi-agent Reinforcement Learning

SHI Dian-xi[1,2,4],ZHAO Chen-ran[1],ZHANG Yao-wen[3],YANG Shao-wu[1] and ZHANG Yong-jun[2]
1 School of Computer Science,National University of Defense Technology,Changsha 410073,China
2 National Innovation Institute of Defense Technology,Academy of Military Sciences,Beijing 100166,China
3 Unit 32282 of People's Liberation Army of China,Jinan 250000,China
4 Tianjin Artificial Intelligence Innovation Center,Tianjin 300457,China

Abstract　At present,most multi-agent reinforcement learning(MARL) algorithms using the architecture of centralized training and decentralized execution(CTDE) have good results in homogeneous multi-agent systems. However,for heterogeneous multi-agent systems composed of different roles,there is always the problem of credit assignment,which makes it difficult for agents to learn effective cooperation strategies. To tackle the above problems,an adaptive reward method with end-to-end cooperation based on multi-agent reinforcement learning is proposed. It can promote the cooperation between agents. First,a batch regularization network is proposed. It uses a graph neural network to model the cooperative relationship of heterogeneous multi-agents. And it uses the attention mechanism to calculate the weight of key information. Also,it uses the batch regularization method to generate feature vectors. Besides,it guides the algorithm to learn in the right direction,thereby effectively improving the performance of heterogeneous multi-agent cooperative strategy generation. Second,an adaptive intrinsic reward network based on the actor-critic method is proposed. It can convert sparse rewards into dense rewards,which can guide agents to generate cooperative strategies according to the situation on the field. Through experiments,compared with the current mainstream multi-agent reinforcement learning algorithms,the proposed method has achieved significantly good results in the "cooperative-game" scenario. In addition,the visual analysis of the strategy-reward-behavior correlation further verifies the effectiveness of the proposed method.

Keywords　Multi-agent reinforcement learning,Graph attention network,Adaptive intrinsic reward

到稿日期:2021-07-09　返修日期:2022-01-05
基金项目:国家自然科学基金(91948303)
This work was supported by the National Natural Science Foundation of China(91948303).
通信作者:张拥军(yjzhang@nudt.edu.cn)

1 引言

随着人工智能和自动化技术的飞速发展，多智能体系统(Multi-Agent System，MAS)开始被广泛应用于工业、安全、军事、科研等各个领域[1-4]。相比单智能体系统，多智能体系统通过智能体之间的协同，能够有效提升任务的执行效率，显著增强系统的生存能力和对复杂环境的适应能力[5]。随着单智能体强化学习研究的不断深入，研究人员开始将目光转向多智能体强化学习(Multi-Agent Reinforcement Learning，MARL)。集中训练分布执行(Centralized Training and Decentralized Execution，CTDE)方法是应对多智能体强化学习中的非平稳性问题和可扩展性问题的一种有效方法，并且是目前研究的热点[6]。

集中式训练分布式执行方法的优点在于，每个智能体都有其独立的策略，可以通过与环境交互进行更新。然而，这种方法的局限性在于，随着智能体种类、数量的增加，网络很难从庞大的动作状态联合空间中提取出精确的特征，导致智能体通过学习获得的策略效果较差、耗时较长[7]。同时，在多智能体系统中还存在着信用分配问题[8-9]，即无法清晰地区分每个角色的智能体在任务中的贡献。目前，大多数研究都集中在耦合智能体之间的关系上，包括平均场理论[10]和因果关系方法[11]。它们通过对智能体之间的关系进行建模，从而促进多智能体强化学习算法训练出合理的合作策略模型。然而，平均场理论用平均动作来表示智能体的行为，忽略了智能体之间的差异性，也忽略了智能体之间的属性关系。因果关系方法考虑了每种智能体的属性特征以及智能体之间的相互影响，但却忽略了群体行为的效应，因而难以学习到有效的合作策略。在群体合作方面，智能体之间的通信机制是促进合作的有效方法[12]，然而，随着多智能体系统的规模、类型的增加和变化，系统需要处理的信息空间越来越庞大，造成维度爆炸的问题。

为此，本文提出了一种基于多智能体强化学习的端到端合作的自适应奖励方法，该方法通过批正则化图网络从智能体本地观察向量出发，对智能体之间的关系进行建模，通过自适应奖励函数网络动态生成奖励，从而有效地促进合作策略的生成。本文的主要贡献如下：

(1)提出了一种批正则化图网络(Batch Normalized Graph Network，BNGNet)，该网络将图神经网络(Graph Neural Network，GNN)[13-14]、注意力机制[15-16]以及批正则化方法(Batch Normalization，BN)[17]有机结合，生成具有关系属性和合作偏向性的观察向量，输出给 Actor-Critic 网络；

(2)基于演员-评论家方法(Actor-Critic，AC)提出了一种双层优化的自适应奖励网络(Bi-level Optimization of Adaptive Reward Network，BOARNet)，该网络根据环境状态动态生成连续的自适应奖励值，引导智能体向着有益于合作的方向进行学习，从而有效地生成合作策略；

(3)基于星际争霸 II 及多智能体粒子测试平台，设计实现了一个原型系统，并通过一系列实验证明了本文算法优于当前主流的集中式训练分布式执行算法，同时，对生成的行为策略及内在连续奖励进行可视化分析，结果表明，本文提出的算法可以有效地生成群体协同策略。

2 相关工作

2.1 基于图神经网络的强化学习

图神经网络是神经网络的一种扩展，通过学习节点之间的关系或特征向量之间的关系，可以将节点进行分类或者排列，是当前一种主流的学习方法。

目前，研究人员对图神经网络与 MARL 相结合的方法进行了大量研究。G2ANet[13]和 DyMA[18]等方法的研究侧重于注意力机制的构造以及迁移效果的体现等方面，图神经网络在其中的作用是进行向量拼接；关系强化学习算法[19]在网络中加入了多头注意力机制，通过学习成对交互的智能体的状态，来提升智能体完成任务的能力；关系正向模型[20]使用监督学习方法，基于全局状态来预测其他所有智能体的行为，但是在部分可观测的环境条件下，该方法单靠局部观测很难作出准确的预测；MAGnet 算法[21]以关联图的形式学习相关信息，其中关系权重通过基于启发式规则的预定义损失函数来学习，需要以监督学习的方式来训练，因而损失了一定的泛化性。

相比上述方法，我们希望可以通过端到端的模式将智能体关系建模简化，利用图注意力机制来增强智能体之间的合作效果，从而提高学习效率，加快收敛速度。

2.2 注意力机制

近年来，注意力机制[15]被广泛应用于多智能体强化学习领域。其本质是在庞大的数据信息中选择出对当前任务来说最关键的相关特征区域。当前最为主流的注意力机制是软注意力机制，它是完全可微的，因此可以非常容易地通过端到端反向传播进行训练。

将注意力机制描述为两种变量的相关性计算，即存在查询变量$\boldsymbol{V}_Q$和键值变量$\boldsymbol{V}_K^i$的向量对，通过映射函数 $f(\cdot)$，输出指数的加权和的形式，其中，权重$\boldsymbol{w}_j$被分配给每个值，用以计算查询变量$\boldsymbol{V}_Q$与键值变量$\boldsymbol{V}_K^i$之间的关系程度，如式(1)所示：

$$\boldsymbol{w}_j=\frac{\exp(f(\boldsymbol{V}_Q,\boldsymbol{V}_K^i))}{\sum_k \exp(f(\boldsymbol{V}_Q,\boldsymbol{V}_K^k))} \tag{1}$$

2.3 批正则化

在深度学习中，一个重要的概念是数据要满足独立同分布(Independent and Identically Distributed，I. I. D)，其核心是数据在经过网络计算前后，输入与输出应满足相应的统计学约束。若不满足，则会导致算法的收敛速度非常慢，甚至无法收敛。在深度神经网络中，网络是分层的，如果把每一层视为一个单独的分类器，那么整个网络就是分类器的串联。在训练过程中，随着某一层分类器参数的改变，其输出的分布也会改变，导致下一层的分类器需要不断适应新的分布，造成整体的不稳定，从而导致算法难以收敛。

为此，文献[17]提出了一种批正则化方法，其核心思想是在数据通过网络前对数据进行预处理，将输入数据做偏移，强制其均值与方差为 0 和 1。这样做的好处有两点：1)通过对隐藏层各神经元的输入做类似的标准化处理，可以提高神经网络的训练速度；2)可以降低前面层的权重变化对后面层

造成的影响，从而使整体网络更具鲁棒性。

假定 $\boldsymbol{Z}$ 为一批输入数据，满足 $\boldsymbol{Z}\in\mathbb{R}^{d\times m}$，其中，$m$ 为样本个数，d 为神经元个数。假定关注当前第 j 个维度，即第 j 个神经元，则有 $\boldsymbol{Z}_j\in\mathbb{R}^{1\times m}$。批正则化的过程如式(2)所示：

$$\begin{aligned}\mu_j&=\frac{1}{m}\sum_{i=1}^{m}\boldsymbol{Z}_j^{(i)}\\ \sigma_j^2&=\frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{Z}_j^{(i)}-\mu_j)\\ \overset{\wedge}{\boldsymbol{Z}}_j&=\frac{\boldsymbol{Z}_j-\mu_j}{\sqrt{\sigma_j^2+\varepsilon}}\end{aligned}\tag{2}$$

3 端到端合作的自适应奖励方法

针对异构多智能体系统中的信用分配问题，本文提出了一种基于多智能体强化学习的端到端合作的自适应奖励方法。该方法的核心思想是：根据智能体之间的合作关系，对观察向量进行有侧重的学习，即智能体着重注意具有合作关系的相关智能体，并增加其观察向量的权重；同时，根据从环境中获取的状态，动态地生成奖励值，通过该奖励值引导智能体向着有益于集体获胜的方向进行学习。

该方法的框架结构如图 1 所示，由批正则化图网络 BNGNet 以及基于 Actor-Critic 网络的双层优化的自适应奖励网络 BOARNet(Bi-level Optimization of Adaptive Reward Network)构成。

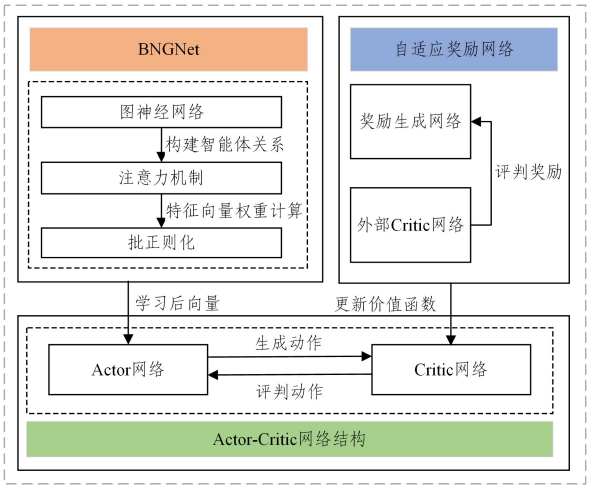


图 1 端到端合作策略学习框架

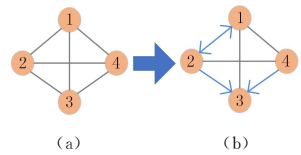
Fig. 1 End-to-end cooperative strategy learning framework

BNGNet 由图神经网络、注意力机制以及批正则化网络构成，其作用是利用图神经网络构建智能体之间的关系，通过注意力机制将获取的环境观察值进行特征计算，通过批正则化网络将智能体之间的关系与特征向量进行融合，生成具有关系属性和合作偏向性的观察向量，输出给 Actor-Critic 网络。双层优化的自适应奖励网络 BOARNet 由自适应奖励网络和 Actor-Critic 网络构成，其中，自适应奖励网络由外部集中式 Critic 网络和奖励生成网络构成，其作用是动态、实时地生成奖励函数值，外部集中式 Critic 网络用于评判生成的奖励；Actor-Critic 网络根据输入向量(即 BNGNet 产生的向量)，利用 Actor 网络生成动作，由 Critic 网络生成对应的价值函数来评价该动作，在 Critic 网络的更新过程中，通过引入自适应奖励网络生成动态奖励，更新价值函数以适应环境的动态变化，从而生成群体的协同策略。

3.1 批正则化图网络 BNGNet

3.1.1 基于图神经网络构建智能体之间的关系

利用图神经网络构建智能体之间的关系，将所有智能体视为节点，将其观察向量作为其属性特征。在不考虑具有偏向合作的情况下，每个智能体的地位相同，因此可以构建一个全连接图来表示，如图 2(a)所示；针对合作-博弈场景，当存在偏向性合作关系时，可以构建一个偏向合作图，如图 2(b)所示。



(a) (b)

图 2 合作全连接无偏图与偏向合作图

Fig. 2 Cooperative fully connected unbiased graph and biased cooperative graph

如图 2 所示，假设智能体 1 和 2 需要相互合作，智能体 2 和 4 需要与智能体 3 进行合作，那么，可以将该偏向性合作的相互关系用图邻接矩阵 $\boldsymbol{A}$ 来表示，如式(3)所示：

$$\boldsymbol{A}=\begin{bmatrix}1&1&0&0\\1&1&1&0\\0&0&1&0\\0&0&1&1\end{bmatrix}\tag{3}$$

矩阵 $\boldsymbol{A}$ 中的行和列均对应着智能体的编号。首先，考虑智能体自己与自己进行合作，则将对角线上各个元素置为 1；其次，如果两个智能体之间存在合作关系，那么就将其对应元素置为 1。

由于矩阵中元素 0 的出现不仅会使网络计算稀疏化，还会在后续运算过程中导致输入向量的特征丢失，因此，我们利用 softmax 函数对矩阵进行正则化，如式(4)所示：

$$\boldsymbol{A}\leftarrow\mathrm{softmax}(\boldsymbol{A})\tag{4}$$

根据式(4)的运算，矩阵 $\boldsymbol{A}$ 经过正则化后的形式如式(5)所示：

$$\boldsymbol{A}=\begin{bmatrix}0.37&0.37&0.13&0.13\\0.30&0.30&0.30&0.1\\0.17&0.17&0.48&0.17\\0.13&0.13&0.37&0.37\end{bmatrix}\tag{5}$$

此方法不仅消除了 0 元素的不良影响，还将合作与非合作的关系通过权重分开；不仅考虑了具有偏向性合作的关系，同时也考虑了非合作智能体的相关特征，促进了对策略的探索。

3.1.2 基于注意力机制的智能体之间相关性权重和特征向量计算

在利用图神经网络构建其智能体之间的关系的基础上，进一步利用注意力机制进行智能体之间相关性权重和特征向量的计算，其计算过程如图 3 所示，计算过程的具体描述如下。

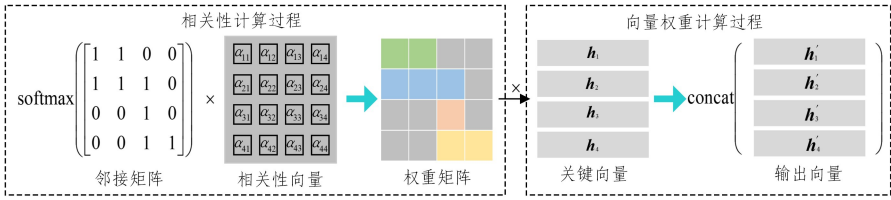


图 3 权重和特征向量计算过程

Fig. 3 Weight and eigenvector calculation process

智能体 i 的观察向量通常可以表示为一个四元组：$\boldsymbol{o}_t^i=(\boldsymbol{o}_t^{i,\text{env}},\boldsymbol{m}_t^i,\boldsymbol{o}_t^{i,\text{ally}},\boldsymbol{o}_t^{i,\text{enemy}})$，其中，$\boldsymbol{o}_t^{i,\text{env}}$ 为环境相关信息，$\boldsymbol{o}_t^{i,\text{ally}}$ 为队友相关信息，$\boldsymbol{o}_t^{i,\text{enemy}}$ 为敌方相关信息，m_t^i 为其他信息。针对智能体之间的合作关系，需要对 $\boldsymbol{o}_t^{i,\text{env}}$ 和 $\boldsymbol{o}_t^{i,\text{ally}}$ 进行提取，并进行关系图计算。定义关键向量为 $\boldsymbol{h}_i=concat(\boldsymbol{o}_t^{i,\text{env}},\boldsymbol{o}_t^{i,\text{ally}})$，其含义是将环境相关信息与队友相关信息进行向量拼接，目的是将每个智能体观察的环境与队友信息在图中突显出来，使得偏重于合作的相关队友的信息与环境的权重增大。

具体步骤如下：首先，通过权重矩阵 $\boldsymbol{W}_q$ 和 $\boldsymbol{W}_k$ 缩减智能体 i,j 的关键向量 $\boldsymbol{h}_i,\boldsymbol{h}_j$ 的维度，然后通过自注意力操作 $att(\cdot)$ 进行注意力系数 e_{ij} 的计算，e_{ij} 表示智能体 i 对智能体 j 的注意程度，可表示为：

$$e_{ij}=att(\boldsymbol{W}_q\boldsymbol{h}_i,\boldsymbol{W}_k\boldsymbol{h}_j) \tag{6}$$

为了使注意力系数 e_{ij} 更容易计算和比较，本文引入了 softmax 函数对所有智能体 i 的相邻节点 j 进行正则化，如式(7)所示：

$$\alpha_{ij}=\text{softmax}_j(e_{ij})=\frac{\exp(e_{ij})}{\sum_{k\in N_i}\exp(e_{ik})} \tag{7}$$

在此基础上，还引入了 LeakyRELU 非线性激励函数以及单层前馈网络 $\vec{a}$ 来计算智能体 i 与智能体 j 之间的相关性权重 α_{ij}，如式(8)所示：

$$\alpha_{ij}=\frac{\exp(\text{LeakyRELU}(\vec{a}\cdot e_{ij}))}{\sum_{k\in N_i}\exp(\text{LeakyRELU}(\vec{a}\cdot e_{ik}))} \tag{8}$$

其中，$\vec{a}$ 为单层前馈网络，用于进一步缩减维度、提取特征。将 α_{ij} 定义为相关性权重，其含义为智能体 i 与智能体 j 之间的相关性。

当我方具有 N 个智能体时，对于智能体 i 来说，则有 N 个相关性权重 α_{ij}，因此，可以定义相关性向量 $\widetilde{\boldsymbol{h}}_i$ 为：

$$\widetilde{\boldsymbol{h}}_i=(\alpha_{i1},\alpha_{i2},\cdots,\alpha_{iN}) \tag{9}$$

进一步地，我们利用图邻接矩阵 $\boldsymbol{A}$，通过点乘的方式，为相关性向量 $\widetilde{\boldsymbol{h}}_i$ 中的相关性权重 α_{ij} 赋予相对权重值，从而将偏向合作的智能体的相关性权重 α_{ij} 进一步放大，将非偏向合作的相关性权重 α_{ij} 缩小。乘以原关键向量 $\boldsymbol{h}_i$ 后，得到具有偏向注意效果的输出向量 $\boldsymbol{h}_i'$，用 $concat$ 操作进行向量拼接，获得最终结果 $\boldsymbol{h}'$(即对 $\boldsymbol{h}_i'(i\in\{1,2,\cdots\})$ 等多个向量进行拼接形成 $\boldsymbol{h}'$)，如式(10)所示。

$$\boldsymbol{h}'=(\boldsymbol{A}\otimes\widetilde{\boldsymbol{h}})\times\boldsymbol{h} \tag{10}$$

3.1.3 基于批正则化的特征融合

上一节中，输出的 $\boldsymbol{h}_i'$ 需要与敌方相关信息 $\boldsymbol{o}_t^{i,\text{enemy}}$ 和其他相关信息 $\boldsymbol{m}_t^i$ 进行特征融合，然而，由于数据类型不一致，直接进行特征融合会导致计算误差扩大。出现这个问题的根本原因是数据的独立同分布条件(I. I. D.)未能得到满足。为了解决这一问题，本文在特征融合前对数据进行批正则化处理，使之满足 I. I. D. 条件。

BNGNet 特征融合网络的结构如图 4 所示。特征融合过程分成 3 个阶段：1)编码与权重计算阶段；2)批正则化阶段；3)最终输出阶段。在编码和权重计算阶段，首先通过图注意力网络对 $\boldsymbol{o}_t^{i,\text{env}}$ 和 $\boldsymbol{o}_t^{i,\text{ally}}$ 进行侧重学习，输出结果为 $\boldsymbol{h}_i'$；同时，通过 MLP 网络层对 $\boldsymbol{o}_t^{i,\text{enemy}}$ 和 $\boldsymbol{m}_t^i$ 进行编码；在批正则化阶段，对图注意力网络学习后的向量、两个编码后的向量分别经过 BN 层进行正则化，形成 $\boldsymbol{\tau},\boldsymbol{\sigma}^{\text{enemy}}$ 和 $\boldsymbol{\sigma}^{\text{env}}$ 3 个向量；在最终输出阶段，经过批正则化的 $\boldsymbol{\tau},\boldsymbol{\sigma}^{\text{enemy}}$ 和 $\boldsymbol{\sigma}^{\text{env}}$ 具有相同的数据分布，因此可以进行向量拼接，得到输出向量 $\boldsymbol{\eta}_t^i$。

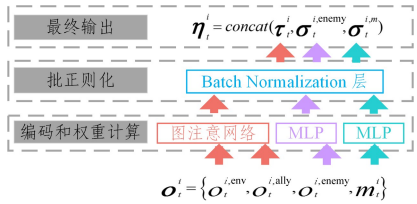


图 4 BNGNet 特征融合网络结构示意图

Fig. 4 Schematic diagram of BNGNet feature fusion network structure

在 BNGNet 网络中，智能体 i 的原始观察向量经过关系学习(通过图注意力网络实现)、编码(通过 MLP 实现)和特征融合(通过 BN 网络实现)后，生成了具有关系属性的新的特征向量，其对应的算法流程如算法 1 所示。

算法 1 BNGNet 算法

输入：原始观察向量 $\mathbf{o}_t^i$，图邻接矩阵 $\mathbf{A}$

输出：$\boldsymbol{\eta}_t^i$

1. 提取关键向量 $\mathbf{h}_i=\text{concat}(\mathbf{o}_t^{i,\text{env}},\mathbf{o}_t^{i,\text{ally}})$
2. 计算相关性权重 $\boldsymbol{\alpha}_i$ 与相关性向量 $\widetilde{\mathbf{h}}_i$
3. 利用邻接矩阵 $\mathbf{A}$ 通过式(10)计算出输出向量 $\mathbf{h}_i'$
4. 将 $\mathbf{o}_t^{i,\text{env}}$ 与 $\mathbf{o}_t^{i,\text{enemy}}$ 通过 MLP 进行编码
5. 将 $\mathbf{h}_i'$ 与 $\mathbf{o}_t^{i,\text{env}},\mathbf{o}_t^{i,\text{enemy}}$ 编码后的向量分别通过 BN 层进行计算，得到 $\tau_t^i,\boldsymbol{\sigma}_t^{i,\text{env}},\boldsymbol{\sigma}_t^{i,\text{enemy}}$
6. 令 $\boldsymbol{\eta}_t^i=\text{concat}(\boldsymbol{\tau}_t^i,\boldsymbol{\sigma}_t^{i,\text{env}},\boldsymbol{\sigma}_t^{i,\text{enemy}})$

3.2 双层优化的自适应奖励生成网络 BOARNet

在奖励生成方面，本文在 Actor-Critic 方法的基础上进行改进，设计了一种自适应奖励生成网络。从 CTDE 架构的角度来说，应该存在集体的价值评判网络(对应集体价值函数)；同时，对于个体来说，每个智能体也存在独立的价值评判网络(对应个体价值)。BOARNet 的具体结构如图 5 所示。该网络结构设计由 4 个部分组成：外部集中价值评判网络(Cen-

tralized Critic Network)、个体 Critic 网络、个体 Actor 网络和奖励函数网络。

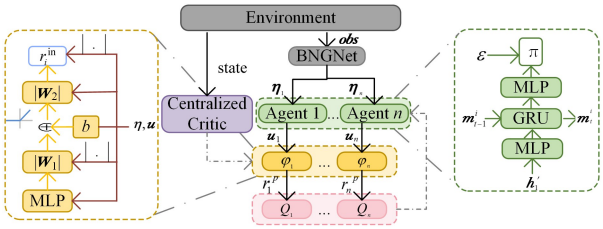


图 5 双层优化的自适应奖励生成网络结构示意图

Fig. 5 Schematic diagram of BOARNet

BOARNet 网络能够根据环境状态 s 与每个智能体的动作 u 自适应地生成奖励 $r_i^{\text{in}}(s_i,u_i)$，该奖励作为一种连续的奖励，可以根据环境与动作动态地生成，是对环境中原始外部奖励 $r_{i,t}^{\text{ex}}$ 的补充，新的奖励定义如式(11)所示：

$$r_{i,t}^{p}=r_{i,t}^{\text{ex}}+\lambda\, r_{\varphi_{i,t}}^{\text{in}}(s_i,u_i) \tag{11}$$

其中，$r_{i,t}^{\text{ex}}$ 为智能体 i 在 t 时刻的原始奖励，$r_{\varphi_{i,t}}^{\text{in}}(s_i,u_i)$ 为通过个体奖励网络 $\varphi_{i,t}$ 生成的奖励，其系数 λ 的定义如式(12)所示：

$$\lambda=\frac{\partial Q_t^{\text{tot}}(\boldsymbol{s},\boldsymbol{u})}{\partial Q_{i,t}(s_i,u_i)} \tag{12}$$

其中，集体状态为 $\boldsymbol{s}=(s_1,\cdots,s_N)$，集体动作为 $\boldsymbol{u}=(u_1,\cdots,u_N)$。设置该系数的目的是期望通过集体价值 $Q_t^{\text{tot}}(\boldsymbol{s},\boldsymbol{u})$ 与个体价值 $Q_{i,t}(s_i,u_i)$ 之间的一致性关系，对连续奖励 $r_i^{\text{in}}(s_i,u_i)$ 进行调节。当集体价值 $Q_t^{\text{tot}}(\boldsymbol{s},\boldsymbol{u})$ 与个体 i 的价值 $Q_{i,t}(s_i,u_i)$ 具有一致性增长或者减小时，说明该个体所评估的价值有益于集体(即使一致性减小，式(12)中的偏导项也为正值)，驱使智能体向着有益于集体利益的方向去学习，其网络结构如图 5 左侧部分所示，该网络构造了一个单调一致性的网络结构。首先，具有关系属性的输出向量 $\boldsymbol{\eta}$ 与动作 $\boldsymbol{u}$ 通过 MLP 网络层进行编码；然后，引入带有绝对值的网络层 $|\boldsymbol{W}_1|$，该网络层保证了输入的正值性，通过与常数项 b 的加和，可以形成非线性的网络拟合，而后引入 ReLU 激励函数与带有绝对值的网络层 $|\boldsymbol{W}_2|$，进一步保证了非线性拟合后的单调性，从而生成奖励 $r_{\varphi_{i,t}}^{\text{in}}(s_i,u_i)$。其单调性证明如下。

集体动作值函数 Q^{tot} 可表示为依赖于 $\boldsymbol{s}$ 和 $\boldsymbol{u}$ 的集体价值函数，因此，Q^{tot} 的表达式如式(13)所示：

$$Q^{\text{ex}}=Q^{\text{tot}}(\boldsymbol{s},\boldsymbol{u}) \tag{13}$$

同样地，个体价值函数 Q_i 的表达式如式(14)所示：

$$Q_i=Q_i(\boldsymbol{s}_i,\boldsymbol{u}_i) \tag{14}$$

利用隐函数定理，Q^{tot} 可以被表示为关于 Q_i 的函数，如式(15)所示：

$$Q^{\text{tot}}=Q^{\text{tot}}(\boldsymbol{s},\boldsymbol{u},Q_1,Q_2,\cdots,Q_n) \tag{15}$$

我们将 Q^{ex} 进行泰勒展开得到关于 Q_i 的计算式，如式(16)所示：

$$Q^{\text{tot}}=c(\boldsymbol{s})+\sum_i\mu_i Q_i(\boldsymbol{s}_i,\boldsymbol{u}_i)+^{\circ}(|Q_i-Q_i(\boldsymbol{s},\boldsymbol{u}_i^{o})|) \tag{16}$$

其中，$\boldsymbol{u}^{o}=\{\boldsymbol{u}_i^{o}\}_{i=1}^{n}$ 表示在当前环境状态 $\boldsymbol{s}$ 下的智能体联合最优行为。在式(16)中，高阶项 $^{\circ}(|Q_i-Q_i(\boldsymbol{s},\boldsymbol{u}_i^{o})|)$ 可以忽略不计，这是因为在 MAS 的环境状态 $\boldsymbol{s}$ 下，基于当前的最好行为 $\boldsymbol{u}^{o}$，系统已经在博弈环境中达到纳什均衡。忽略 $Q_i(\boldsymbol{s},\boldsymbol{u}_i)$ 和 $Q_i(\boldsymbol{s},\boldsymbol{u}_i^{o})$ 之间的高阶无穷项差异，可以得出泰勒展开式子中的参数 μ_i 的表示，如式(17)所示：

$$\mu_i=\frac{\partial Q^{\text{tot}}(\boldsymbol{s},\boldsymbol{u})}{\partial Q_i(\boldsymbol{s}_i,\boldsymbol{u}_i)} \tag{17}$$

其证明如下：首先，利用链式法则，可得到式(18)：

$$\frac{\partial Q^{\text{tot}}}{\partial \boldsymbol{u}_i}=\frac{\partial Q^{\text{tot}}}{\partial Q_i}\frac{\partial Q_i}{\partial \boldsymbol{u}_i} \tag{18}$$

对于优化目标 J，可以得到式(19)：

$$\nabla J(\theta)=\nabla_\theta\log\pi_\theta(\boldsymbol{u}|\boldsymbol{s})Q(\boldsymbol{s},\boldsymbol{u}) \tag{19}$$

如果该优化目标 J 有解，那么 ∇J 存在极值，则结果如式(20)所示：

$$\frac{\partial Q^{\text{tot}}}{\partial \boldsymbol{u}_i}=0 \tag{20}$$

因此得出式(21)：

$$\frac{\partial Q_i}{\partial \boldsymbol{u}_i}(\boldsymbol{u}_i^{o})=0,\frac{\partial Q^{\text{tot}}}{\partial Q_i}\neq 0 \tag{21}$$

若用反证法的思想，则式(22)成立：

$$\frac{\partial Q^{\text{tot}}}{\partial Q_i}=0 \tag{22}$$

则动作 $\boldsymbol{u}$ 可以取任意值，满足式(21)。但是，其与目标问题(19)相悖。因此，可以利用奖励值将价值 Q 函数表示为式(23)：

$$Q^{\pi}(\boldsymbol{s},\boldsymbol{u})=\mathbb{E}_{\boldsymbol{s},\boldsymbol{u}}^{\pi}\Big[\sum_{t=0}^{+\infty}\gamma^t r_{t+1}\,|\,\boldsymbol{s}_0=\boldsymbol{s},\boldsymbol{u}_0=\boldsymbol{u}\Big] \tag{23}$$

基于上述公式，Q^{ex} 的表示如式(24)所示：

$$\begin{aligned}Q^{\text{tot}}&=\mathbb{E}_{\boldsymbol{s},\boldsymbol{u}}\Big[\sum_{t=0}^{+\infty}\gamma^t r_{t+1}^{p}\Big]\\&=\mathbb{E}_{\boldsymbol{s},\boldsymbol{u}}\Big[\sum_{t=0}^{+\infty}\gamma^t(r_{t+1}^{\text{ex}}+\lambda\, r_{t+1}^{\text{in}})\Big]\\&=\mathbb{E}_{\boldsymbol{s},\boldsymbol{u}}\Big[\sum_{t=0}^{+\infty}\gamma^t r_{t+1}^{\text{ex}}\Big]+\lambda\,\mathbb{E}_{\boldsymbol{s},\boldsymbol{u}}\Big[\sum_{t=0}^{+\infty}\gamma^t r_{t+1}^{\text{in}}\Big]\end{aligned} \tag{24}$$

结合式(16)，可以得出式(25)：

$$\begin{aligned}c(\boldsymbol{s})&=\mathbb{E}_{\boldsymbol{s},\boldsymbol{u}}\Big[\sum_{t=0}^{+\infty}\gamma^t r_{t+1}^{\text{ex}}\Big]\\Q_i&=\mathbb{E}_{\boldsymbol{s},\boldsymbol{u}}\Big[\sum_{t=0}^{+\infty}\gamma^t r_{i,t+1}^{\text{in}}\Big]\end{aligned} \tag{25}$$

从而可以得出结论，如式(26)所示：

$$\lambda_i=\mu_i=\frac{\partial Q^{\text{tot}}}{\partial Q_i} \tag{26}$$

将 BOARNet 网络与 BNGNet 网络相结合，构建了一个端到端合作的双层优化的自适应奖励算法，简称 E2E-IRL 算法，算法描述如算法 2 所示。该算法的训练整体过程如图 5 所示。在正向传播过程中，首先，智能体从环境中获取到观察向量 $\boldsymbol{obs}$，通过 BNGNet 生成具有偏向合作关系的输出向量 $\boldsymbol{\eta}=(\boldsymbol{\eta}_1,\cdots,\boldsymbol{\eta}_N)$；其次，将 $\boldsymbol{\eta}$ 作为输入并通过 Actor 网络生成行为策略 π_i，根据策略与环境状态，我们将环境状态 $\boldsymbol{s}_i$ 与动作 $\boldsymbol{u}_i=\pi_i(\boldsymbol{s}_i)$ 作为自变量输入到奖励网络 φ 中，生成连续性奖励 $r_{\varphi_{i,t}}^{\text{in}}(\boldsymbol{s}_i,\boldsymbol{u}_i)$；然后，利用 $r_{\varphi_{i,t}}^{\text{in}}(\boldsymbol{s}_i,\boldsymbol{u}_i)$ 通过个体 Critic 网络生成个体价值 $Q_{i,t}(\boldsymbol{s}_i,\boldsymbol{u}_i)$；最后，对于集中式评判网络(Centralized Critic Network)，根据环境状态 $\boldsymbol{s}=(\boldsymbol{s}_1,\cdots,\boldsymbol{s}_N)$ 与集体动作 $\boldsymbol{u}=(\boldsymbol{u}_1,\cdots,\boldsymbol{u}_N)$ 生成集体价值函数 $Q_t^{\text{tot}}(\boldsymbol{s},\boldsymbol{u})$。在反向传播过程中，由个体的 $Q_{i,t}(\boldsymbol{s}_i,\boldsymbol{u}_i)$ 对 Actor 网络进行反向更新；由集体的 $Q_t^{\text{tot}}(\boldsymbol{s},\boldsymbol{u})$ 对奖励网络 φ 进行更新。

图5中右侧部分为动作策略生成网络Actor，该网络首先通过MLP层对输出向量 $\boldsymbol{\eta}$ 进行编码；然后通过循环网络模块GRU[22-24]，引入上一时间步的隐藏向量（该模块考虑了时间相关性，有利于网络继承上一时间步的网络结构，从而提高学习效率）；最后通过MLP层生成策略 π_i（生成的过程引入噪声 ε，目的是加大对生成的结构的干扰，避免陷入局部最优，以探索更多的行为策略）。

除了每个智能体的个体评判网络，BOARNet网络添加了一个外部的集中式评判网络，整体是一个双层优化的网络设计，其网络更新过程如下。

首先，双层优化的目标优化函数定义如式(27)所示：

$$\begin{aligned}&\max_{\varphi,\theta} J^{\mathrm{ex}}(\varphi)\\&\text{s.t. } \theta_i=\arg\max_{\theta} J_i(\theta,\varphi),\forall i\in A\end{aligned}\tag{27}$$

其中，θ_i 为Actor网络参数，用以生成独立策略 π_i。θ_i 可以通过最大化个体目标函数 $J_i:=\mathbb{E}_{\boldsymbol{\eta}_i,\boldsymbol{u}_i}[R_i^{\rho}]$，根据梯度下降法公式，可以得到个体目标的优化函数，如式(28)所示：

$$\begin{aligned}&\nabla_{\theta_i}J_i=\nabla_{\theta_i}\log\pi_{\theta_i}(\boldsymbol{\eta}_i\mid\boldsymbol{s}_i)Q_i(\boldsymbol{s}_i,\boldsymbol{\eta}_i)\\&R_{i,t}^{\rho}=\sum_{l=0}^{\infty}\gamma^l(r_{t+l}^{\mathrm{ex}}+\lambda\, r_{i,t+l}^{\mathrm{in}})\end{aligned}\tag{28}$$

其次，对于集中的目标函数 J^{ex} 来说，由于 $J^{\mathrm{ex}}(\varphi)$ 是关于奖励函数参数 φ 的网络，因此，采用链式法则进行展开，如式(29)所示：

$$\nabla_{\varphi_i}J^{\mathrm{ex}}=\nabla_{\theta_i'}J^{\mathrm{ex}}\cdot\nabla_{\varphi_i}\theta_i'\tag{29}$$

其中，θ_i' 为Actor网络的更新参数。

最后，由于该网络为双层网络结构，因此需要先更新策略参数 θ'，再将其带入外部更新 $\nabla_{\theta_i'}J^{\mathrm{ex}}$，以求得集体目标优化函数，如式(30)所示：

$$\begin{aligned}&\theta_i'=\theta_i+\beta\nabla_{\theta_i}\log\pi_i(\boldsymbol{u}_i\mid\boldsymbol{\eta}_i)Q_i(\boldsymbol{u}_i,\boldsymbol{\eta}_i)\\&\nabla_{\theta_i'}J^{\mathrm{ex}}=\nabla_{\theta_i'}\log\pi_{\theta_i'}(\boldsymbol{u}_i\mid\boldsymbol{\eta}_i)Q^{\mathrm{tot}}(\boldsymbol{s},\boldsymbol{u})\end{aligned}\tag{30}$$

根据上述训练过程与网络更新步骤，算法对应的伪代码流程如算法2所示。

算法2 E2E-IRL算法

输入：学习率超参数 α，策略更新超参数 β，图邻接矩阵 $\mathbf{A}$，迭代数T

输出：策略网络参数 θ，奖励网络参数 φ

1. 初始化网络参数 θ 和 φ
2. While t<T do
3. 采样 $D=(s_0,u_0,\cdots,s_N,u_N)$
4. 利用邻接矩阵 $\mathbf{A}$，通过BNGNet网络（算法1）得出输出向量 $\boldsymbol{\eta}$
5. 通过Actor网络得到策略 π_θ
6. 利用式(28)更新网络参数 $\theta\leftarrow\theta'$
7. 利用集中式评判网络生成 Q^{tot}
8. 通过式(29)、式(30)更新奖励网络参数 φ
9. End While

4 实验结果及可视化分析

4.1 原型系统设计与实现

本小节描述了原型系统设计与实现，主要包括：双层优化的端到端的合作策略学习机制（即E2E-IRL算法的具体实现）、模型训练机制和数据分析机制3个部分，原型系统架构如图6所示。

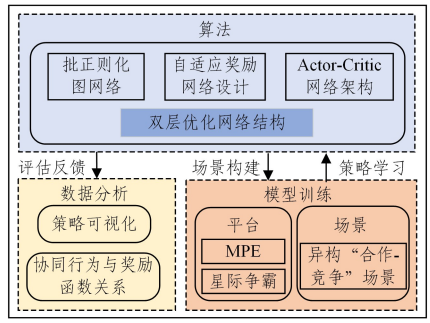


图6 端到端自适应合作模型训练原型系统设计

Fig. 6 Prototype system design of end-to-end adaptive cooperative model training

双层优化的端到端的合作策略学习机制是原型系统的核心部分，包括批正则化图网络（BNGNet）、自适应奖励生成网络以及Actor-Critic网络，该机制将环境中的观察向量通过学习融入关系属性，从而加快算法的收敛速度，并提升训练效果。基于构建的场景的学习训练，双层优化的端到端的合作策略学习机制学习、训练出对应的协同策略模型；同时，通过数据分析对双层优化的端到端的合作策略学习机制进行评估反馈，方便进行调参测试，以期训练出合理的优化协同策略。在训练方面，本文基于多智能体粒子环境MPE以及星际争霸II游戏平台进行测试，在多个异构“合作-竞争”场景下测试算法，通过胜率、奖励等指标来分析验证算法的有效性。在数据分析方面，通过策略可视化与协同行为与奖励函数的关系展示，对训练出的模型进行分析，以证明生成的模型能够促进智能体进行有效协作。

在具体实现中，该原型系统使用多智能体粒子环境（Multi-agent Particle Environment，MPE）以及开源的星际争霸II（Starcraft Ⅱ，SC Ⅱ）场景作为实验环境，使用Python语言来实现设计功能。该实验环境允许开发者自定义实验场景、创建实体（如智能体、兵种配置、环境地形设置等）、为实体分配功能、定义智能体的观察空间和动作空间以及奖励反馈等。

4.2 实验设置

我们将原型系统部署到多智能体粒子环境[25]以及星际争霸II[26-27]的游戏测试平台当中。

4.2.1 多智能体粒子环境

多智能体粒子环境是一个具有连续观察值和离散动作空间的简单的多智能体粒子世界，允许用户自定义多智能体环境和任务，便于进行MARL算法的实现和测试。

本文在MPE环境中设定了一个全新的多智能体协作对抗模式。该模式下有两支队伍：进攻队和防守队。进攻队包含突防智能体和干扰智能体，其中突防智能体（图7中的绿色圆形）的任务是在被巡逻智能体摧毁之前到达目标区域，而干扰智能体（图7中的蓝色圆形）的任务是干扰防守队智能体之间的通信。防守队包含巡逻智能体（图7中的粉色圆形），其任务是通过追捕、巡逻、中继和编队等手段阻止突防智能体到达目标区域。此外，场上还有随机出现的障碍物（图7中的黑色圆形）阻碍双方的行动。基于此模式，我们搭建了3个由

不同数量、不同种类的智能体组成的测试场景，分别是（巡逻智能体-突防智能体-干扰智能体）：3-1-1，3-2-1，4-2-1。图 7 给出了 3-1-1 场景的示意图，其中粉色线条右侧为突防智能体出生点，灰色线条左侧是其目标区域。

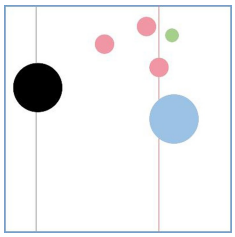


图 7 MPE 下突防-防御场景示意图（电子版为彩图）

Fig. 7 Schematic diagram of penetration-defense scenarios in MPE environment

实验选择的对比算法为 MADDPG，该算法中每个智能体的 Critic 会收集其他智能体的感知信息，用于个体价值的计算。

4.2.2 星际争霸 II 的实验环境

SCII 是一个实时战略游戏，可以通过手动操作或部署算法的方式与内置算法玩家（敌方）进行对抗，通过对兵力的部署调配击败所有敌人，以获得胜利。针对异构多智能体协同策略生成，选择其中的 3 个场景进行测试实验。这 3 个场景均为异构多智能体场景，需要不同兵种之间的相互配合，其中，Stalkers（S）为远程攻击兵种，Zealots（Z）为一个近战攻击单位，Colossi（C）为群体攻击单位，对于成群的敌人有很好的攻击效果。其地图详情如表 1 所列。

表 1 地图详情介绍

Table 1 Map details

地图	我方兵种	对方兵种	类型
2S3Z	2 Stalkers 3 Zealots	2 Stalkers 3 Zealots	对称异构
3S5Z	3 Stalkers 5 Zealots	3 Stalkers 5 Zealots	对称异构
1C3S5Z	1 Colossi 2 Stalkers 3 Zealots	1 Colossi 2 Stalkers 3 Zealots	对称异构

选择的对比算法 Independent Actor-Critic（IAC）[28]，Central-V[8]，COMA[8] 和 LIIR[29] 均为 CTDE 架构下的算法。

4.3 多智能体粒子环境的实验结果

场上存在着移动的干扰源与障碍物，会对我方巡逻智能体的观测空间和行为空间造成影响；突防智能体可被视为移动的目标点，并且其具有全局感知与最快的移动速度，给我方的围捕策略训练带来了挑战。

基于上述情况并结合图 8 可看出智能体组成对协同模型训练的影响：4-2-1 作为相对简单的场景，虽然智能体规模较大，但是对于协同策略的训练来说，4 个巡逻智能体可以有更多的合作组合策略可供学习探索，E2E-IRL 和 MADDPG 都可以达到 85％以上的胜率；对于 3-1-1 场景，E2E-IRL 算法同样能够训练出很好的围捕策略；对于 3-2-1 场景，无论哪种算法都不能训练出协同策略。这不仅是由约束过多造成的，同时也涉及多目标分配的问题（即我方智能体需要对两个突防智能体进行任务分配），因此，也能从侧面得出一个结论，即在存在移动通信干扰和障碍物的复杂场景中，巡逻方与进攻方的比例应至少保持 3∶1 的数量比例。

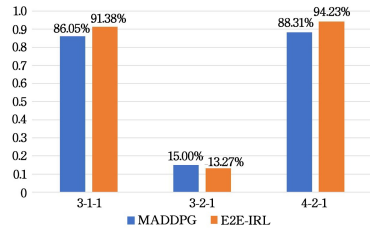
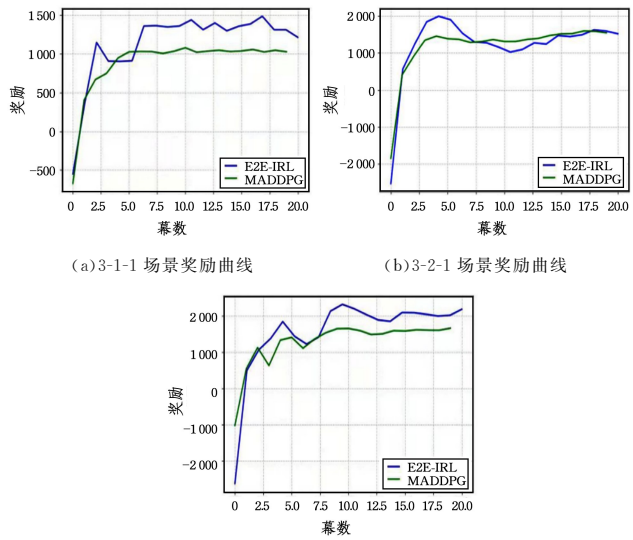


图 8 2 种训练模型在胜率上的比较

Fig. 8 Win rate comparison of two training models

进一步地，根据奖励曲线分析 2 种算法对策略生成的影响。首先，对于场景 3-1-1 与 4-2-1 来说，从图 9 可以看出，E2E-IRL 算法能够获得较高的环境奖励，但曲线波动较大。这种波动性是由内部奖励网络引起的，如第 3 节所述，E2E-IRL 根据环境自适应地生成相应的奖励，能够让智能体更好地理解环境，从而学习出更好的策略，但同时也会造成一定的非稳定性。对于场景 3-2-1 而言，由于该场景比较复杂，两种算法均未能训练出合理有效的模型，因此在奖励上也难以区分其优劣。



（a）3-1-1 场景奖励曲线 （b）3-2-1 场景奖励曲线

（c）4-2-1 场景奖励曲线

图 9 2 种训练模型在奖励曲线上的比较

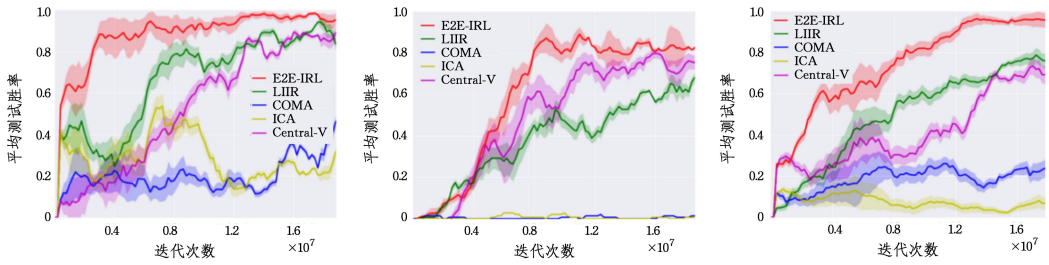
Fig. 9 Reward curve comparison of two training models

4.4 星际争霸 II 实验结果

4.4.1 胜率比较

基于上述设置，本文根据智能体的数量和组合类型递增的方式进行对比实验，并就胜率对算法进行分析比较。

首先，在 2S3Z 地图上进行了对比实验，该地图是相对简单的场景，由 2 个 S 与 3 个 Z 进行对称式的协同对抗，实验结果如图 10（a）所示。从 IAC 和 COMA 算法的波动性来看，这两种算法的胜率都较低，且具有非平稳性，表明未能在较短的时间内学习出合理、稳定的协同对抗策略；LIIR 和 Central-V 算法的收敛性和平均胜率均优于 IAC 算法，并且在训练后期可以训练出较好的合作模型，获得较高的胜率；本文提出的 E2E-IRL 方法可以更快地收敛，且胜率接近 100％。



(a)2S3Z (b)3S5Z (c)1C3S5Z

图 10 E2E-IRL 与其他算法在胜率上的比较

Fig. 10 Win rate comparison of E2E-IRL and other algorithms

其次，针对相同的兵种组合，3S5Z 在 2S3Z 的基础上进一步增加了智能体的数量，意味着进一步增加了学习协同策略的难度，实验结果如图 10(b)所示。可以看出，LIIR 和 Central-V 优于 COMA 和 IAC，且相比 2S3Z 场景，智能体规模的增长直接导致联合状态空间维度也有了增长，使得在相同的训练周期内，网络更难提取出准确的特征，收敛速度也比 2S3Z 慢。

最后，在 1C3S5Z 地图中增加了新角色 Colossi(C)，这意味着不仅增加了智能体的数量，同时增加了智能体种类，进而增加了训练难度，实验结果如图 10(c)所示。可以看出，相比上述方法，E2E-IRL 表现出了良好的训练效果，其胜率达到 90%左右。在战斗回放中，通过指定 C 与 Z 两种远程攻击单位的配合，在战斗初期对对方的团簇兵力进行集火攻击，同时，近战单位 Z 通过相互配合，消灭了弱势敌方。下面将具体讨论合作模式与可视化策略执行。

4.4.2 合作模式分析

本节在 3S5Z 和 1C3S5Z 场景中利用邻接矩阵来定义不同的协作模式，比较了不同邻接矩阵下的胜率，探索合作模式与策略生成之间的内在关系。

如图 11 所示，在 3S5Z 场景中定义了 4 种类型的协作模式。在邻接矩阵中，行和列前 3 个代表兵种 S，后 5 个代表兵种 Z，有颜色的部分表示对应位置的元素被置为 1。如 $M1$-1 矩阵意味着我方只需要相同兵种进行合作，即组内合作；而矩阵 $M1$-2 不仅要求组内合作，同时，每个 S 还需要关注其他的 Z(同理 Z 也需要关注 S)，即兵种之间也要求相互合作，称作组间合作。

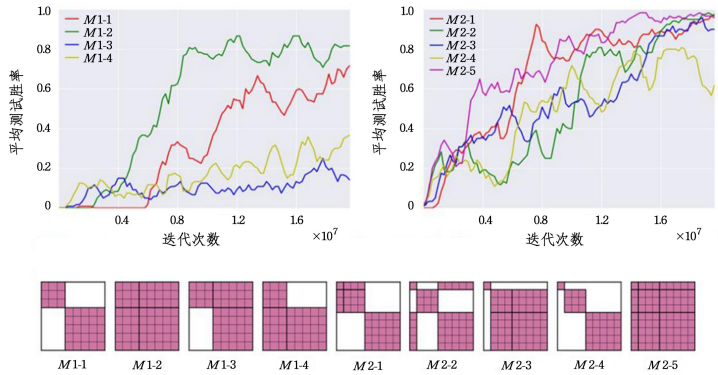


图 11 合作模式与胜率分析(电子版为彩图)

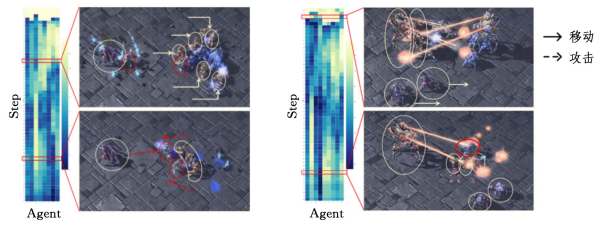
Fig. 11 Analysis of cooperation mode and win rate

从图 11 左图可以看出，$M1$-1 和 $M1$-2 的效果相对较好。其中，包含了组内合作与组间合作的 $M1$-2 模式具有更好的效果，而 $M1$-1 在前期进行了较长的策略探索，未能找到正确的学习方向，导致效果较差。结果表明兵种 S 与 Z 不存在有效的单向性协同策略。

基于 3S5Z 的实验分析，我们将重点分析 1C3S5Z 中的组间合作与策略生成。如图 11 右图所示，$M2$-4 的效果最差，而其他模式的胜率均能达到 90%以上；从协作模式来看，$M2$-4 只有组内合作，不存在组间合作，而其他模式均存在着不同程度的组间合作。这表明在该场景中，两种远程攻击兵种 C 与 S 存在单向性合作，能够训练出较好的协同策略。

4.4.3 可视化策略分析

本节对策略进行了可视化，通过热力图对智能体之间的行为进行定量分析。具体方法如下：从经验缓冲池中采样一批数据，取最后一次战斗回合中的数据，将这批数据的注意力权重的平均值制成热力图，如图 12 所示，热力图以智能体的 ID 编号为横坐标，以整个战斗回合时间步为纵坐标，图中较亮部分的像素代表了智能体在该时刻执行了攻击动作；为了更直观地了解该时刻智能体的具体行为，本文使用战斗回放中的截图作为辅助说明。



(a) (b)

图 12 可视化战斗策略

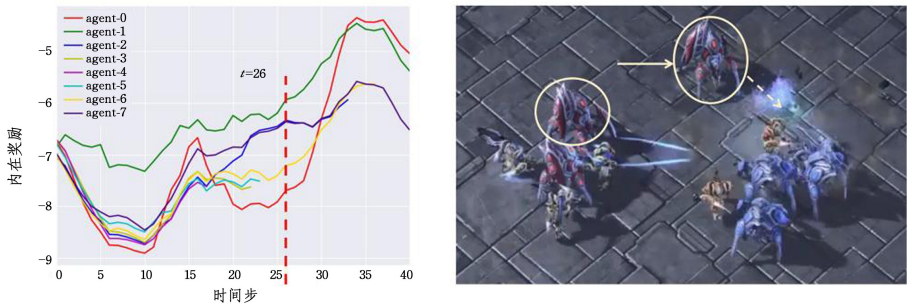
Fig. 12 Visual battle strategy

在图 12(a)的热力图中可以看到两个明亮的区域。右上方的亮区表明，后序编号的智能体，即近战攻击单位 Z，在执行攻击动作。在 $t=17$ 时刻，通过截图可知，我方 5 个 Z 单位对剩余的敌方施行包围并集中火力进行攻击，以局部“多打少”的方式优先消灭了对方的远程攻击单位。图 12(a)左下方的亮区表明，前序编号的智能体具有协同的攻击行为。由于已经处于战斗后期，在该批数据中可能存在死亡的智能体，因此亮度略暗于右上方区域。在 $t=48$ 时刻，回放截图显示，我方两个 Z 单位与一个远程 Z 单位协同攻击敌方剩余目标。

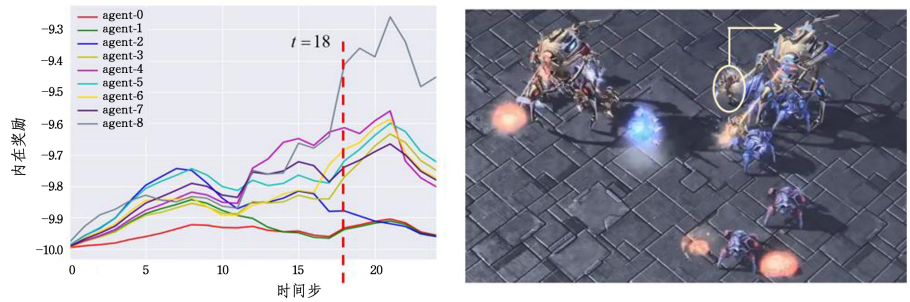
在图 12(b)中，相比 3S5Z，该场景引入了 C 单位(该单位可以对聚簇的单位造成巨大杀伤)。从 $t=3$ 时刻截图中的站位可知，我方 Z 首先移动至前方以吸引攻击，保护 C 单位；同时远程攻击单位 S 移动至敌侧方，通过分散站位，避免敌方 C 单位对我方进行大面积攻击。图 12(b)右上方亮区表明，在我方 C 对敌方造成重创后，Z 通过组内合作的方式进行协同进攻，整个战斗过程中，敌方 C 单位是需要优先消灭的目标。在上一节中，我们证明了 E2E-IRL 可以训练出接近 100%胜率的策略，因此在时刻 $t=49$ 截图中显示，敌方只剩下一个 Z 单位，而我方剩余较多的单位。所有热力图下方连续的亮区表示我方剩余单位集中火力对敌进行攻击；同时，从图 12(b)中可以看到，编号为 3 的智能体从战斗中期就一直处于暗色(死亡)。因此，我们需要进一步探究每个智能体在训练过程中实施的具体行为。

4.4.4 协同行为与奖励函数关联

本节主要讨论奖励值与具体行为之间的关系，并通过战斗回放截图来辅助说明。取 3S5Z 中的训练模型和 1C3S5Z 中的 M2-5 训练模型进行分析，两个模型经过 2.0×10^7 时间步的学习后，最终测试胜率分别为 85%和 98%，如图 13 所示，曲线图的横坐标为战斗回放的时间步，纵坐标为奖励值。



(a)3S5Z



(b)1C3S5Z

图 13 内部奖励与具体行为(电子版为彩图)

Fig. 13 Internal rewards and specific behaviors

在 3S5Z 场景中，通过回放及图 13(a)可知，智能体{3,5}(Z 单位)在中期死亡，智能体 1(S 单位)可以通过走位避免受到攻击。对比 $t=26$ 时的截图与曲线图可以看出，死去的智能体{3,5}得到了不完全的奖励，智能体 1 可以获得比其他智能体更高的奖励反馈。同时，在图 13(a)曲线图中，智能体 0 在初期处于低奖励值状态，在 $t=26$ 时开始上升。由图 13(a)截图可知，智能体 0(左黄色圆圈内)周围没有敌人，因此对于攻击的奖励开始上升，引导其攻击行为；在战斗后期，对抗趋于结束，因此所有智能体都有较低的奖励趋势。同时，由整个图 13(a)曲线的趋势来看，所有智能体的曲线都有相似的走向，说明该模型具有组内和组间合作的特征。

在图 13(b)中，智能体 8 在 $t=18$ 时刻所获得的内在奖励具有很大的提升。通过战斗回放可知，该智能体受到敌方 C 单位攻击后，沿着黄色箭头方向移动，以躲避攻击。随后，于敌后方继续攻击，该策略具有鲜明的优势，因此我方的智能体没有伤亡。在回放中还可观察到，智能体 0 一直躲在 C 单位下方，处于 C 庇护下，并持续开火。从图 13(b)曲线图中也可知，该智能体奖励值一直处于低水平状态且无明显变化。

结束语 本文提出了一种基于多智能体强化学习的端到端合作策略自适应奖励方法，通过设计批正则化图网络将智能体的合作关系与观察向量进行融合，加速算法的收敛；基于 Actor-Critic 方法，提出了一种自适应奖励函数网络，动态生成奖励以增加智能体行为的多样性，解决信用分配问题。在多智能体粒子环境和星际争霸 II 环境中对多种异构“合作-博弈”场景进行了测试，通过胜率、可视化策略与协同行为-奖励分析，证明了生成的策略模型可有效提高算法的收敛速度，提升异构场景中的效果，并生成了合理的协同策略模型。在未来的工作中，我们计划继续研究自适应动态分组在“合作-博弈”场景协同策略模型中的生成问题，并在更多的测试环境中进行实验。

参 考 文 献

[1] WIERING M A. Multi-agent reinforcement learning for traffic light control[C]// Machine Learning: Proceedings of the Seventeenth International Conference (ICML'2000). 2000: 1151-1158.

[2] SALLAB A E L, ABDOU M, PEROT E, et al. Deep reinforcement learning framework for autonomous driving[J]. Electronic Imaging, 2017, 2017(19): 70-76.

[3] ZHAI Y Y. Multi-agent reinforcement learning-driven dynamic channel allocation for unmanned aerial vehicles [J/OL]. Telecommunications Technology. http://kns.cnki.net/kcms/detail/51.1267.TN.20220304.1008.002.html.

[4] DENG Q T, HU DAN E, CAI T T, et al. Reactive Power Optimization Strategy of Distribution Network Based on Multi-Agent Deep Reinforcement Learning [J]. New Technology of Electrical Engineering, 2022, 41(2): 10-20.

[5] WU Y, ZHANG B, YANG S, et al. Energy-efficient joint communication-motion planning for relay-assisted wireless robot surveillance[C]// IEEE INFOCOM 2017-IEEE Conference on Computer Communications. IEEE, 2017: 1-9.

[6] WANG T, WANG J, ZHENG C, et al. Learning nearly decomposable value functions via communication minimization[J]. arXiv: 1910.05366, 2019.

[7] LOWE R, WU Y I, TAMAR A, et al. Multi-agent actor-critic for mixed cooperative-competitive environments[C]// Advances in Neural Information Processing Systems. 2017: 6379-6390.

[8] FOERSTER J, FARQUHAR G, AFOURAS T, et al. Counterfactual multi-agent policy gradients[J]. arXiv: 1705.08926, 2017.

[9] RASHID T, SAMVELYAN M, DE WITT C S, et al. QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning[J]. arXiv: 1803.11485, 2018.

[10] YANG Y, LUO R, LI M, et al. Mean field multi-agent reinforcement learning[J]. arXiv: 1802.05438, 2018.

[11] JAQUES N, LAZARIDOU A, HUGHES E, et al. Social influence as intrinsic motivation for multi-agent deep reinforcement learning[C]// International Conference on Machine Learning. PMLR, 2019: 3040-3049.

[12] SUKHBAATAR S, FERGUS R. Learning multiagent communication with backpropagation[C]// Proceedings of the 30th International Conference on Neural Information Processing Systems. 2016: 2252-2260.

[13] LIU Y, WANG W, HU Y, et al. Multi-Agent Game Abstraction via Graph Attention Neural Network[C]// AAAI. 2020: 7211-7218.

[14] YOU J, LIU B, YING Z, et al. Graph convolutional policy network for goal-directed molecular graph generation[C]// Advances in Neural Information Processing Systems. 2018: 6410-6421.

[15] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]// Advances in Neural Information Processing Systems. 2017: 5998-6008.

[16] KAPETANAKIS S, KUDENKO D. Reinforcement learning of coordination in heterogeneous cooperative multi-agent systems [M]// Adaptive Agents and Multi-Agent Systems II. Berlin: Springer, 2004: 119-131.

[17] IOFFE S, SZEGEDY C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]// International Conference on Machine Learning. PMLR, 2015: 448-456.

[18] WANG W, YANG T, LIU Y, et al. From Few to More: Large-Scale Dynamic Multiagent Curriculum Learning[C]// AAAI. 2020: 7293-7300.

[19] ZAMBALDI V, RAPOSO D, SANTORO A, et al. Relational deep reinforcement learning[J]. arXiv: 1806.01830, 2018.

[20] TACCHETTI A, SONG H F, MEDIANO P A M, et al. Relational forward models for multi-agent learning[J]. arXiv: 1809.11044, 2018.

[21] MALYSHEVA A, SUNG T T, SOHN C B, et al. Deep multi-agent reinforcement learning with relevance graphs[J]. arXiv: 1811.12557, 2018.

[22] ZHANG T, XU H, WANG X, et al. Multi-Agent Collaboration via Reward Attribution Decomposition[J]. arXiv: 2010.08531, 2020.

[23] SCHULMAN J, LEVINE S, ABBEEL P, et al. Trust region policy optimization[C]// International Conference on Machine Learning. PMLR, 2015: 1889-1897.

[24] WANG Q, XIONG J, HAN L, et al. Exponentially Weighted Imitation Learning for Batched Historical Data[C]// NeurIPS. 2018: 6291-6300.

[25] MORDATCH I, ABBEEL P. Emergence of Grounded Compositional Language in Multi-Agent Populations[J]. arXiv: 1703.04908, 2017.

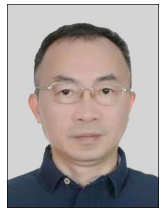
[26] VINYALS O, EWALDS T, BARTUNOV S, et al. Starcraft ii: A new challenge for reinforcement learning[J]. arXiv: 1708.04782, 2017.

[27] SAMVELYAN M, RASHID T, DE WITT C S, et al. The starcraft multi-agent challenge[J]. arXiv: 1902.04043, 2019.

[28] TAN M. Multi-agent reinforcement learning: Independent vs. cooperative agents[C]// Proceedings of the Tenth International Conference on Machine Learning. 1993: 330-337.

[29] DU Y, HAN L, FANG M, et al. Liir: Learning individual intrinsic reward in multi-agent reinforcement learning[C]// 33rd Conference on Neural Information Processing Systems (NeurIPS 2019). Vancouver, Cannada, 2019.

SHI Dian-xi, born in 1966, Ph.D, professor, Ph.D supervisor. His main research interests include distributed object middleware technology, adaptive software technology, artificial intelligence and robot operating systems.



ZHANG Yong-Jun, born in 1966, Ph.D, professor. His main research interests include artificial intelligence, multi-agent cooperation, machine learning and feature recognition.

(责任编辑:何杨)