第 十五 章 非合作关系设定下的多智能体强化学习

上一章研究了多智能体强化学习 (MARL) 中最简单的设定——完全合作关系，在这种设定下，所有的智能体有相同的奖励、回报、价值、目标函数。本章研究非合作关系，那么不同智能体各自有不同的奖励、回报、价值、目标函数。本章中采用的符号如图 15.1 所示。

第 15.1 节定义非合作关系设定下的策略学习、策略梯度方法、以及收敛判别。第 15.2 节推导非合作关系下的 A2C 方法，本书称之为 Multi-Agent Noncooperative A2C，缩写 MAN-A2C，可以用于离散控制问题。第 15.3 节用三种架构实现 MAN-A2C：完全去中心化、完全中心化、中心化训练 + 去中心化决策。第 15.4 介绍多智能体确定策略梯度方法，缩写 MADDPG，可以用于连续控制问题。

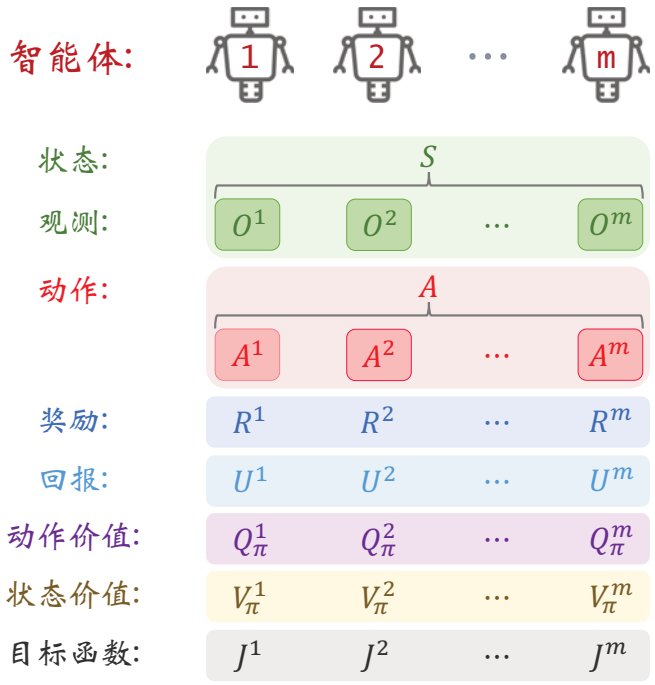

图 15.1: 多智能体强化学习 (MARL) 的符号。

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

15.1 非合作关系设定下的策略学习

上一章研究合作关系的 MARL，即所有智能体的奖励都相等：$R^1 = \cdots = R^m$。在这种设定下，所有智能体有相同的状态价值函数 $V_\pi(s)$ 和目标函数

$$J(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m) = \mathbb{E}_S\Big[V_\pi(s)\Big].$$

目标函数可以衡量策略网络参数 $\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m$ 的好坏。策略学习的目的是改进 $\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m$ 使得 J 变大。合作关系的设定下，策略学习的收敛标准很明确：如果找不到更好的 $\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m$ 使得 J 变大，那么当前的 $\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m$ 就是最优解。

非合作关系设定下的目标函数： 如果是非合作关系，那么不存在这样的关系：$R^1 = \cdots = R^m$。两个智能体的奖励不相等 ($R^i \neq R^j$)，那么它们的回报也不相等 ($U^i \neq U^j$)，回报的期望（价值函数）也不相等。把状态价值记作：

$$V^1(s), \quad V^2(s), \quad \cdots, \quad V^m(s).$$

第 i 个智能体的目标函数是状态价值的期望：

$$J^i(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m) = \mathbb{E}_S\Big[V_\pi^i(s)\Big].$$

J^i 的意义是回报 U^i 的期望，所以能反映出第 i 个智能体的表现好坏。

注 目标函数 $J^1, J^2, \cdots, J^m$ 是各不相同的，也就是说智能体没有共同的目标（除非是完全合作关系）。举个例子，在 Predator-Prey（捕食者—猎物）的游戏中，捕食者的目标函数 J^1 与猎物的目标函数 J^2 负相关：$J^1 = -J^2$。

注 第 i 个智能体的目标函数 J^i 依赖于所有智能体的策略网络参数 $\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m$。为什么一个智能体的目标函数依赖于其他智能体的策略呢？举个例子，捕食者改进自己的策略 $\boldsymbol{\theta}^1$，而猎物没有改变策略 $\boldsymbol{\theta}^2$。虽然猎物的策略 $\boldsymbol{\theta}^2$ 没有变化，但是它的目标函数 J^2 会减小。

非合作关系设定下的策略学习： 在多智能体的策略学习中，第 i 个智能体的目标是改进自己的策略参数 $\boldsymbol{\theta}^i$，使得 J^i 尽量大。多智能体的策略学习可以描述为这样的问题：

$$\begin{aligned}
&\text{第 1 个智能体求解：} && \max_{\boldsymbol{\theta}^1} J^1(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m),\\
&\text{第 2 个智能体求解：} && \max_{\boldsymbol{\theta}^2} J^2(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m),\\
&\vdots && \vdots\\
&\text{第 } m \text{ 个智能体求解：} && \max_{\boldsymbol{\theta}^m} J^m(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m).
\end{aligned}$$

注意，目标函数 $J^1, J^2, \cdots, J^m$ 是各不相同的，也就是说智能体没有共同的目标（除非是完全合作关系）。策略学习的基本思想是让每个智能体各自做策略梯度上升：

$$\begin{aligned}
&\text{第 1 号智能体执行：} && \boldsymbol{\theta}^1 \leftarrow \boldsymbol{\theta}^1 + \alpha^1 \cdot \nabla_{\boldsymbol{\theta}^1} J^1(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m),\\
&\text{第 2 号智能体执行：} && \boldsymbol{\theta}^2 \leftarrow \boldsymbol{\theta}^2 + \alpha^2 \cdot \nabla_{\boldsymbol{\theta}^2} J^1(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m),\\
&\vdots && \vdots\\
&\text{第 } m \text{ 号智能体执行：} && \boldsymbol{\theta}^m \leftarrow \boldsymbol{\theta}^m + \alpha^m \cdot \nabla_{\boldsymbol{\theta}^m} J^m(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m).
\end{aligned}$$

公式中的 $\alpha^1, \alpha^2, \cdots, \alpha^m$ 是学习率。由于无法直接计算策略梯度 $\nabla_{\boldsymbol{\theta}^i} J^i$，我们需要对其做

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

近似。各种策略学习方法的区别就在于如何对策略梯度做近似。

收敛的判别： 在合作关系设定下，所有智能体有相同的目标函数 ($J^1 = \cdots = J^m$)，那么判断收敛的标准就是目标函数值不再增长。也就是说改变任何智能体的策略都无法让团队的回报增长。

在非合作关系设定下，智能体的利益是不一致的、甚至是冲突的，智能体各有各的目标函数。该如何判断策略学习的收敛呢？不能用 $J^1 + J^2 + \cdots + J^m$ 作为判断收敛的标准。比如在 Predator-Prey（捕食者—猎物）的游戏中，双方的目标函数是冲突的：$J^1 = -J^2$。如果捕食者改进策略，那么 J^1 会增长，而 J^2 会下降。自始至终，$J^1 + J^2$ 一直等于零，不论策略学习有没有收敛。

在非合作关系设定下，收敛标准是纳什均衡。一个智能体在制定策略的时候，要考虑到其他各方的策略。在纳什均衡的情况下，每一个智能体都在以最优的方式来应对其他各方的策略。在纳什均衡的情况下，谁也没有动机去单独改变自己的策略，因为改变策略不会增加自己的收益。这样就达到了一种平衡状态，所有智能体都找不到更好的策略。这种平衡状态就被认为是收敛。在实验中，如果所有智能体的平均回报都不再变化，就可以认为达到了纳什均衡。

定义 15.1. 纳什均衡

在多智能体系统中，当其余所有智能体都不改变策略的情况下，一个智能体 i 单独改变策略 $\boldsymbol{\theta}^i$，无法让其期望回报 $J^i(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m)$ 变大。 ♣

评价策略的优劣： 有两种策略学习的方法 $\mathcal{M}_+$ 和 $\mathcal{M}_-$，把它们训练出的策略网络参数分别记作 $\boldsymbol{\theta}_+^1, \cdots, \boldsymbol{\theta}_+^m$ 和 $\boldsymbol{\theta}_-^1, \cdots, \boldsymbol{\theta}_-^m$。该如何评价 $\mathcal{M}_+$ 和 $\mathcal{M}_-$ 的优劣呢？在合作关系设定下，很容易评价两种方法的好坏。在收敛之后，把两种策略的平均回报记作 J_+ 和 J_-。如果 $J_+ > J_-$，就说明 $\mathcal{M}_+$ 比 $\mathcal{M}_-$ 好；反之亦然。

在非合作关系的设定下，不能直接用平均回报评价策略的优劣。以捕食者—猎物的游戏为例，我们用两种方法 $\mathcal{M}_+$ 和 $\mathcal{M}_-$ 训练策略网络，把它们训练出的策略网络记作：

$$\pi\left(a \,\middle|\, s, \boldsymbol{\theta}_+^{\text{predator}}\right), \qquad \pi\left(a \,\middle|\, s, \boldsymbol{\theta}_+^{\text{prey}}\right),$$
$$\pi\left(a \,\middle|\, s, \boldsymbol{\theta}_-^{\text{predator}}\right), \qquad \pi\left(a \,\middle|\, s, \boldsymbol{\theta}_-^{\text{prey}}\right).$$

设收敛时的平均回报为：

$$J_+^{\text{predator}} = 0.8, \qquad J_+^{\text{prey}} = -0.8,$$
$$J_-^{\text{predator}} = 0.1, \qquad J_-^{\text{prey}} = -0.1.$$

请问 $\mathcal{M}_+$ 和 $\mathcal{M}_-$ 孰优孰劣呢？假如我们的目标是学习捕食者 (Predator)，能否说明 $\mathcal{M}_+$ 比 $\mathcal{M}_-$ 好呢？答案是否定的。$J_+^{\text{predator}} > J_-^{\text{predator}}$ 可能是由于方法 $\mathcal{M}_+$ 没有训练好猎物 (Pray) 的策略 $\boldsymbol{\theta}_+^{\text{prey}}$，导致捕食者 (Predator) 相对有优势。$J_+^{\text{predator}} > J_-^{\text{predator}}$ 不能说明策略 $\boldsymbol{\theta}_+^{\text{predator}}$ 优于 $\boldsymbol{\theta}_-^{\text{predator}}$。

在非合作关系的设定下，该如何评价两种方法 $\mathcal{M}_+$ 和 $\mathcal{M}_-$ 的优劣呢？以捕食者—

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

猎物的游戏为例，我们让一种方法训练出的捕食者与另一种方法训练出的猎物对决：

$$\pi\left(a \mid s, \boldsymbol{\theta}_{+}^{\text{predator}}\right) \quad \text{对决} \quad \pi\left(a \mid s, \boldsymbol{\theta}_{-}^{\text{prey}}\right),$$
$$\pi\left(a \mid s, \boldsymbol{\theta}_{-}^{\text{predator}}\right) \quad \text{对决} \quad \pi\left(a \mid s, \boldsymbol{\theta}_{+}^{\text{prey}}\right).$$

记录下两方捕食者的平均回报，记作 J_{+}^{predator}、J_{-}^{predator}。两者的大小可以反映出 $\mathcal{M}_{+}$ 和 $\mathcal{M}_{-}$ 的优劣。

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

15.2 非合作设定下的多智能体 A2C

本节研究“非合作关系”设定下的多智能体 A2C 方法 (Multi-Agent Non-cooperative A2C)，缩写 MAN-A2C。

15.2.1 策略网络和价值网络

MAN-A2C 中，每个智能体有自己的策略网络和价值网络，记作：

$$\pi\big(a^i \,\big|\, s; \boldsymbol{\theta}^i\big) \qquad \text{和} \qquad v\big(s; \boldsymbol{w}^i\big).$$

第 i 个策略网络需要把所有智能体的观测 $s = [o^1, \cdots, o^m]$ 作为输入，并输出一个概率分布；第 i 个智能体依据该概率分布抽样得到动作 A^i。两类神经网络的结构与上一章的 MAC-A2C 完全相同。请注意上一章 MAC-A2C 与本章 MAN-A2C 的区别：

- 上一章的 MAC-A2C 用于完全合作关系，所有智能体有相同的状态价值函数 $V_\pi(s)$，所以只用一个神经网络近似 $V_\pi(s)$，记作 $v(s; \boldsymbol{w})$。
- 本章的 MAN-A2C 用于非合作关系，每个智能体各有一个状态价值函数 $V_\pi^i(s)$，所以每个智能体各自对应一个价值网络 $v_\pi(s; \boldsymbol{w}^i)$。

MAN-A2C 属于 Actor-Critic 方法：策略网络 $\pi(a^i|s; \boldsymbol{\theta}^i)$ 相当于第 i 个运动员，负责做决策；每个运动员都一个专属的评委 $v(s; \boldsymbol{w}^i)$，对运动员 i 的表现予以评价。请注意，虽然评委 $v(s; \boldsymbol{w}^i)$ 是对运动员 i 个人做出评价，但是评委会考虑到全局的状态 $s = [o^1, \cdots, o^m]$。举个例子，在足球比赛中，评委 i 只对运动员 i 做评价，目的在于改进该运动员的技术。在比赛中，想要评价运动员 i 的跑位、传球的好坏，当然要考虑到队友、对手的位置，所以评委 i 会考虑到场上所有球员的表现 $s = [o^1, \cdots, o^m]$。注意与上一章中 MAC-A2C 的区别：MAC-A2C 中只有一位评委，他会点评整个团队的表现，而不会给每位运动员单独一个评分。

15.2.2 算法推导

在非合作关系设定下，第 i 号智能体的动作价值函数记作 $Q_\pi^i(s, a)$，策略网络记作 $\pi\big(A^i \,\big|\, S; \boldsymbol{\theta}^i\big)$。不难证明下面的策略梯度定理：

定理 15.1. 非合作关系 MARL 的策略梯度定理

设基线 b 为不依赖于 $A = [A^1, \cdots, A^m]$ 的函数。那么有

$$\nabla_{\boldsymbol{\theta}^i} J^i\big(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m\big) \;=\; \mathbb{E}_{S,A}\Big[\Big(Q_\pi^i(S, A) - b\Big) \cdot \nabla_{\boldsymbol{\theta}^i} \ln \pi\big(A^i \,\big|\, S; \boldsymbol{\theta}^i\big)\Big].$$

期望中的动作 A 的概率质量函数为

$$\pi\big(A \,\big|\, S; \boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m\big) \;\triangleq\; \pi\big(A^1 \,\big|\, S; \boldsymbol{\theta}^1\big) \;\times\; \cdots \;\times\; \pi\big(A^m \,\big|\, S; \boldsymbol{\theta}^m\big).$$

♡

我们用 $b = V_\pi^i(s)$ 作为定理中的基线，并且用价值网络 $v(s; \boldsymbol{w}^i)$ 近似 $V_\pi^i(s)$。按照上

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

一章的算法推导，我们可以把策略梯度 $\nabla_{\boldsymbol{\theta}^i} J^i(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m)$ 近似成：

$$\tilde{\boldsymbol{g}}^i(s_t, a_t^i; \boldsymbol{\theta}^i) \triangleq \Big(r_t^i + \gamma \cdot v(s_{t+1}; \boldsymbol{w}^i) - v(s_t; \boldsymbol{w}^i)\Big) \cdot \nabla_{\boldsymbol{\theta}^i} \pi(a_t^i \,|\, s_t; \boldsymbol{\theta}^i).$$

观测到状态 s_t、s_{t+1}、动作 a_t^i、奖励 r_t^i，这样更新策略网络参数：

$$\boldsymbol{\theta}^i \leftarrow \boldsymbol{\theta}^i + \beta \cdot \tilde{\boldsymbol{g}}^i(s_t, a_t^i; \boldsymbol{\theta}^i).$$

更新价值网络 $v(s; \boldsymbol{w}^i)$ 的方法与 A2C 基本一样。在观测到状态 s_t、s_{t+1}、奖励 r_t^i 之后，计算 TD 目标：

$$\widehat{y}_t^i = r_t^i + \gamma \cdot v(s_{t+1}; \boldsymbol{w}^i).$$

更新参数 $\boldsymbol{w}^i$，使得 $v(s_t; \boldsymbol{w}^i)$ 更接近 $\widehat{y}_t^i$。

15.2.3 训练和决策

训练： 实现 MAN-A2C 的时候，应当使用目标网络缓解自举造成的偏差。第 i 号智能体的目标网络记作 $v(s; \boldsymbol{w}^{i-})$，它的结构与 $v(s; \boldsymbol{w}^i)$ 相同，但是参数不同。设第 i 号智能体策略网络、价值网络、目标网络当前的参数是 $\boldsymbol{\theta}^i_{\text{now}}$、$\boldsymbol{w}^i_{\text{now}}$、$\boldsymbol{w}^{i-}_{\text{now}}$。MAN-A2C 重复下面的步骤更新参数：

1. 观测到当前状态 $s_t = [o_t^1, \cdots, o_t^m]$，让每一个智能体独立做随机抽样：

   $$a_t^i \sim \pi(\cdot \,|\, s_t; \boldsymbol{\theta}^i_{\text{now}}), \qquad \forall i = 1, \cdots, m,$$

   并执行选中的动作。
2. 从环境中观测到奖励 $r_t^1, \cdots, r_t^m$ 与下一时刻状态 $s_{t+1} = [o_{t+1}^1, \cdots, o_{t+1}^m]$。
3. 让价值网络做预测：

   $$\widehat{v}_t^i = v(s_t; \boldsymbol{w}^i_{\text{now}}), \qquad \forall\, i = 1, \cdots, m.$$

4. 让目标网络做预测：

   $$\widehat{v}_{t+1}^{i-} = v(s_{t+1}; \boldsymbol{w}^{i-}_{\text{now}}), \qquad \forall\, i = 1, \cdots, m.$$

5. 计算 TD 目标与 TD 误差：

   $$\widehat{y}_t^i = r_t^i + \gamma \cdot \widehat{v}_{t+1}^{i-}, \qquad \delta_t^i = \widehat{v}_t^i - \widehat{y}_t^i, \qquad \forall\, i = 1, \cdots, m.$$

6. 更新价值网络参数：

   $$\boldsymbol{w}^i_{\text{new}} \leftarrow \boldsymbol{w}^i_{\text{now}} - \alpha \cdot \delta_t^i \cdot \nabla_{\boldsymbol{w}^i} v(s_t; \boldsymbol{w}^i_{\text{now}}), \qquad \forall\, i = 1, \cdots, m.$$

7. 更新目标网络参数：

   $$\boldsymbol{w}^{i-}_{\text{new}} \leftarrow \tau \cdot \boldsymbol{w}^i_{\text{new}} + (1-\tau) \cdot \boldsymbol{w}^{i-}_{\text{now}}, \qquad \forall\, i = 1, \cdots, m.$$

8. 更新策略网络参数：

   $$\boldsymbol{\theta}^i_{\text{new}} \leftarrow \boldsymbol{\theta}^i_{\text{now}} - \beta \cdot \delta_t \cdot \nabla_{\boldsymbol{\theta}^i} \ln \pi(a_t^i \,|\, s_t; \boldsymbol{\theta}^i_{\text{now}}), \qquad \forall\, i = 1, \cdots, m.$$

MAN-A2C 属于同策略 (On-policy)，不能使用经验回放。

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

决策： 在完成训练之后，不再需要价值网络 $v(s;\boldsymbol{w}^1),\cdots,v(s;\boldsymbol{w}^m)$。每个智能体可以用它自己的策略网络做决策。在时刻 t 观测到全局状态 $s_t=\left[o_t^1,\cdots,o_t^m\right]$，然后做随机抽样得到动作：

$$a_t^i \sim \pi\big(\cdot\,\big|\,s_t;\boldsymbol{\theta}^i\big),$$

并执行动作 a_t^i。智能体并不能独立做决策，因为策略网络需要知道所有的观测 $s_t=\left[o_t^1,\cdots,o_t^m\right]$。

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

15.3 三种架构

本节介绍 MAN-A2C 的三种实现方法：“中心化训练 + 中心化执行”、“去中心化训练 + 去中心化执行”、“中心化训练 + 去中心化决策”。

15.3.1 中心化训练 + 中心化决策

可以用完全中心化 (Fully Centralized) 的方式实现 MAN-A2C 的训练和决策。图 15.2 描述了系统的架构。最上面是中央控制器 (Central Controller)，里面部署了与所有 m 个价值网络和策略网络：

$$v(s\,|\,\boldsymbol{w}^1),\quad v(s\,|\,\boldsymbol{w}^2),\quad \cdots,\quad v(s\,|\,\boldsymbol{w}^m),$$
$$\pi(a^1\,|\,\boldsymbol{\theta}^1),\quad \pi(a^2\,|\,\boldsymbol{\theta}^2),\quad \cdots,\quad \pi(a^m\,|\,\boldsymbol{\theta}^m).$$

训练和决策全部由中央控制器完成。智能体负责与环境交互，执行中央控制器的决策 a^i，并把观测到的 o^i 和 r^i 汇报给中央控制器。这种中心化的方式严格实现了上一节的算法。

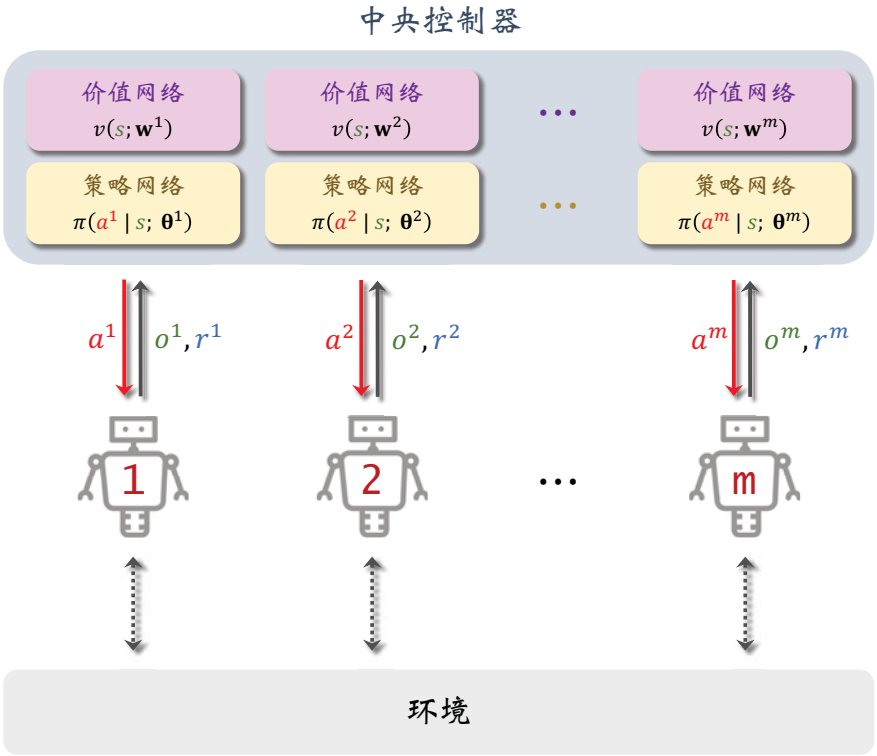

图 15.2: 中心化训练 + 中心化决策的系统架构。

在上一章中，我们用完全中心化的方式实现了 MAC-A2C（见图 14.5）。请注意 MAC-A2C 与此处的 MAN-A2C 的区别。第一，MAC-A2C 的中央控制器上只有一个价值网络，而此处 MAN-A2C 则有 m 个价值网络。第二，MAC-A2C 的每一轮只有一个全局的奖励 r，而 MAN-A2C 的每个智能体都有自己的奖励 r^i。

《深度强化学习》 2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

15.3.2 去中心化训练 + 去中心化决策

为了避免“完全中心化”中的通信，可以对策略网络和价值网络做近似，做到“完全去中心化”。把 MAN-A2C 中的策略网络和价值网络做近似：

$$\pi\big(a^i \,|\, s;\, \boldsymbol{\theta}^i\big) \quad \Longrightarrow \quad \pi\big(a^i \,|\, o^i;\, \boldsymbol{\theta}^i\big),$$
$$v\big(s;\, \boldsymbol{w}^i\big) \quad \Longrightarrow \quad v\big(o^i;\, \boldsymbol{w}^i\big).$$

图 15.3 描述了“完全去中心化”的系统架构。每个智能体上部署一个策略网络和一个价值网络，它们的参数记作 $\boldsymbol{\theta}^i$ 和 $\boldsymbol{w}^i$；智能体之间不共享参数。这样一来，训练就可以在智能体本地完成，无需中央控制器的参与，也无需通信。这种实现的本质是单智能体强化学习，而非多智能体强化学习。

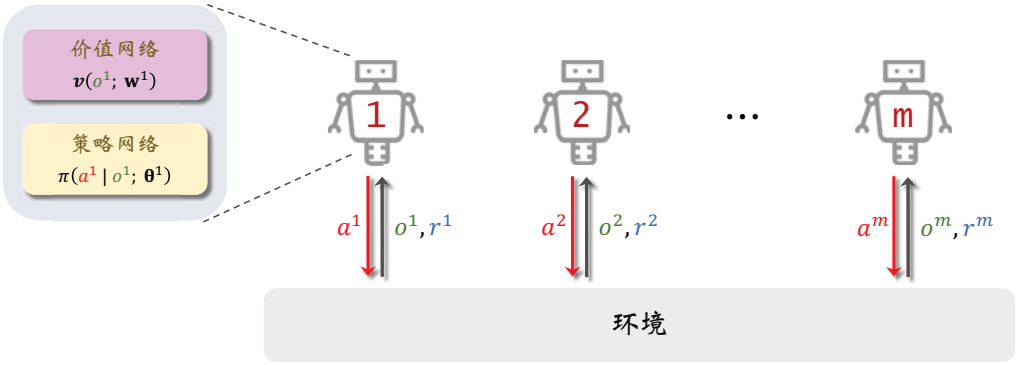

图 15.3: 去中心化训练 + 去中心化决策的系统架构。这种方法也叫做 Independent Actor-Critic。

此处的的实现与上一章中“完全合作关系”设定下的“完全去中心化”几乎完全相同（见图 14.6）。唯一的区别在于此处每个智能体获得的奖励 r^i 是不同的，而上一章中“完全合作关系”设定下的奖励是相同的 $r^1 = \cdots = r^m = r$.

15.3.3 中心化训练 + 去中心化决策

第三种实现方式是“中心化训练 + 去中心化决策”。与“完全中心化”的 MAN-A2C 相比，唯一的区别在于对策略网络做近似：

$$\pi\big(a^i \,|\, s;\, \boldsymbol{\theta}^i\big) \quad \Longrightarrow \quad \pi\big(a^i \,|\, o^i;\, \boldsymbol{\theta}^i\big), \qquad \forall\, i = 1, \cdots, m.$$

由于用智能体局部观测 o^i 替换了全局状态 $s = [o^1, \cdots, o^m]$，策略网络可以部署到每个智能体上。而价值网络仍然是 $v(s; \boldsymbol{w}^i)$，没有做近似。

图 15.4 描述了“中心化训练 + 去中心化决策”的系统架构。中央控制器上有所有的价值网络及其目标网络（图中没有画出目标网络）：

$$v\big(s;\, \boldsymbol{w}^1\big), \quad v\big(s;\, \boldsymbol{w}^2\big), \quad \cdots, \quad v\big(s;\, \boldsymbol{w}^m\big),$$
$$v\big(s;\, \boldsymbol{w}^{1-}\big), \quad v\big(s;\, \boldsymbol{w}^{2-}\big), \quad \cdots, \quad v\big(s;\, \boldsymbol{w}^{m-}\big).$$

中央控制器用智能体发来的观测 $[o^1, \cdots, o^m]$ 和奖励 $[r^1, \cdots, r^m]$ 训练这些价值网络。中央控制器把 TD 误差 $\delta^1, \cdots, \delta^m$ 反馈给智能体；第 i 号智能体用 δ^i 以及本地的 o^i、a^i 来

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

训练自己的策略网络。

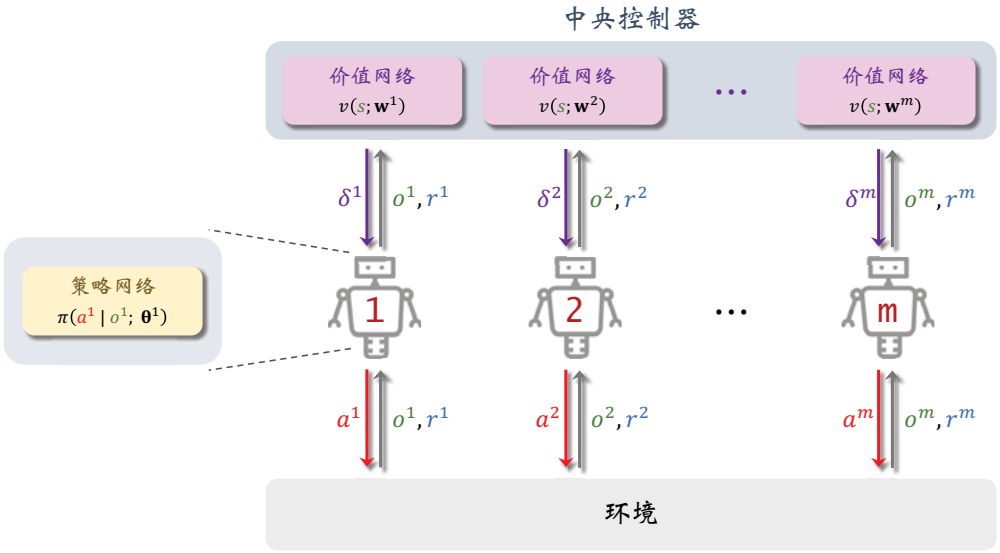

图 15.4: 中心化训练的系统架构。所有 m 个价值网络部署到中央控制器上，策略网络部署到每个智能体上。

上一章“完全合作关系”设定下的“中心化训练”的中央控制器上只有价值网络 $v(s;\boldsymbol{w})$。而此处中央控制器上有 m 个价值网络，每个价值网络对应一个智能体。这是因为此处是“非合作关系”，每个智能体各自对应一个状态价值函数 $V_\pi^i(s)$，而非有共用的 V_π。

中心化训练： 训练的过程需要所有 m 个智能体共同参与，共同改进策略网络参数 $\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m$ 与价值网络参数 $\boldsymbol{w}^1, \cdots, \boldsymbol{w}^m$。设第 i 号智能体的策略网络、价值网络、目标网络当前的参数分别是 $\boldsymbol{\theta}^i_{\text{now}}$、$\boldsymbol{w}^i_{\text{now}}$ 和 $\boldsymbol{w}^{i-}_{\text{now}}$。训练的流程如下：

1. 每个智能体 i 与环境交互，获取当前观测 o_t^i，独立做随机抽样：

$$a_t^i \sim \pi\big(\cdot \,\big|\, o_t^i; \boldsymbol{\theta}^i_{\text{now}}\big), \qquad \forall\, i = 1, \cdots, m, \tag{15.1}$$

   并执行选中的动作。

2. 下一时刻，每个智能体 i 都观测到 o_{t+1}^i 和收到奖励 r_t^i。

3. 每个智能体 i，向中央控制器传输观测 o_t^i、o_{t+1}^i、r_t^i；中央控制器得到状态

$$s_t = \big[o_t^1, \cdots, o_t^m\big] \qquad \text{和} \qquad s_{t+1} = \big[o_{t+1}^1, \cdots, o_{t+1}^m\big].$$

4. 让价值网络做预测：

$$\widehat{v}_t^i = v\big(s_t; \boldsymbol{w}^i_{\text{now}}\big), \qquad \forall\, i = 1, \cdots, m.$$

5. 让目标网络做预测：

$$\widehat{v}_{t+1}^{i-} = v\big(s_{t+1}; \boldsymbol{w}^{i-}_{\text{now}}\big), \qquad \forall\, i = 1, \cdots, m.$$

6. 计算 TD 目标与 TD 误差：

$$\widehat{y}_t^i = r_t^i + \gamma \cdot \widehat{v}_{t+1}^{i-}, \qquad \delta_t^i = \widehat{v}_t^i - \widehat{y}_t^i, \qquad \forall\, i = 1, \cdots, m.$$

《深度强化学习》 2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

7. 更新价值网络参数：

$$\boldsymbol{w}^i_{\text{new}} \leftarrow \boldsymbol{w}^i_{\text{now}} - \alpha \cdot \delta^i_t \cdot \nabla_{\boldsymbol{w}^i} v\big(s_t; \boldsymbol{w}^i_{\text{now}}\big), \qquad \forall\, i = 1, \cdots, m.$$

8. 更新目标网络参数：

$$\boldsymbol{w}^{i-}_{\text{new}} \leftarrow \tau \cdot \boldsymbol{w}^i_{\text{new}} + (1-\tau) \cdot \boldsymbol{w}^{i-}_{\text{now}}, \qquad \forall\, i = 1, \cdots, m.$$

9. 更新策略网络参数：

$$\boldsymbol{\theta}^i_{\text{new}} \leftarrow \boldsymbol{\theta}^i_{\text{now}} - \beta \cdot \delta_t \cdot \nabla_{\boldsymbol{\theta}^i} \ln \pi\big(a^i_t \,\big|\, o^i_t; \boldsymbol{\theta}^i_{\text{now}}\big), \qquad \forall\, i = 1, \cdots, m.$$

去中心化决策： 在完成训练之后，不再需要价值网络 $v(s; \boldsymbol{w}^1), \cdots, v(s; \boldsymbol{w}^m)$。智能体只需要用其本地部署的策略网络 $\pi(a^i | o^i; \boldsymbol{\theta}^i)$ 做决策，决策过程无需通信，因此决策速度很快。

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

15.4 连续控制与 MADDPG

前两节的 MAN-A2C 仅限于离散控制。本节研究连续控制问题，即动作空间 $\mathcal{A}^1, \cdots, \mathcal{A}^m$ 都是连续集合，动作 $\boldsymbol{a}^i \in \mathcal{A}^i$ 是向量。本节介绍一种适用于连续控制的多智能体强化学习 (MARL) 方法。多智能体深度确定策略梯度 (Multi-Agent Deep Deterministic Policy Gradient，缩写 MADDPG) 是一种很有名的 MARL 方法，它的架构是“中心化训练 + 去中心化决策”。

15.4.1 策略网络和价值网络

设系统里有 m 个智能体。每个智能体对应一个策略网络和一个价值网络：

$$\boldsymbol{\mu}(o^i; \boldsymbol{\theta}^i) \quad \text{和} \quad q(s, \boldsymbol{a}; \boldsymbol{w}^i).$$

策略网络是确定性的：对于确定的输入 o^i，输出的动作 $\boldsymbol{a}^i = \boldsymbol{\mu}(o^i; \boldsymbol{\theta}^i)$ 是确定的。价值网络的输入是全局状态 $s = [o^1, \cdots, o^m]$ 与所有智能体的动作 $\boldsymbol{a} = [\boldsymbol{a}^1, \cdots, \boldsymbol{a}^m]$，输出是一个实数，表示“基于状态 s 执行动作 $\boldsymbol{a}$”的好坏程度。第 i 号策略网络 $\boldsymbol{\mu}(o^i; \boldsymbol{\theta}^i)$ 用于控制第 i 号智能体，而价值网络 $q(s, \boldsymbol{a}; \boldsymbol{w}^i)$ 则用于评价所有动作 $\boldsymbol{a}$，给出的分数可以指导第 i 号策略网络做出改进；见图 15.5。因此 MADDPG 是一种 Actor-Critic 方法。

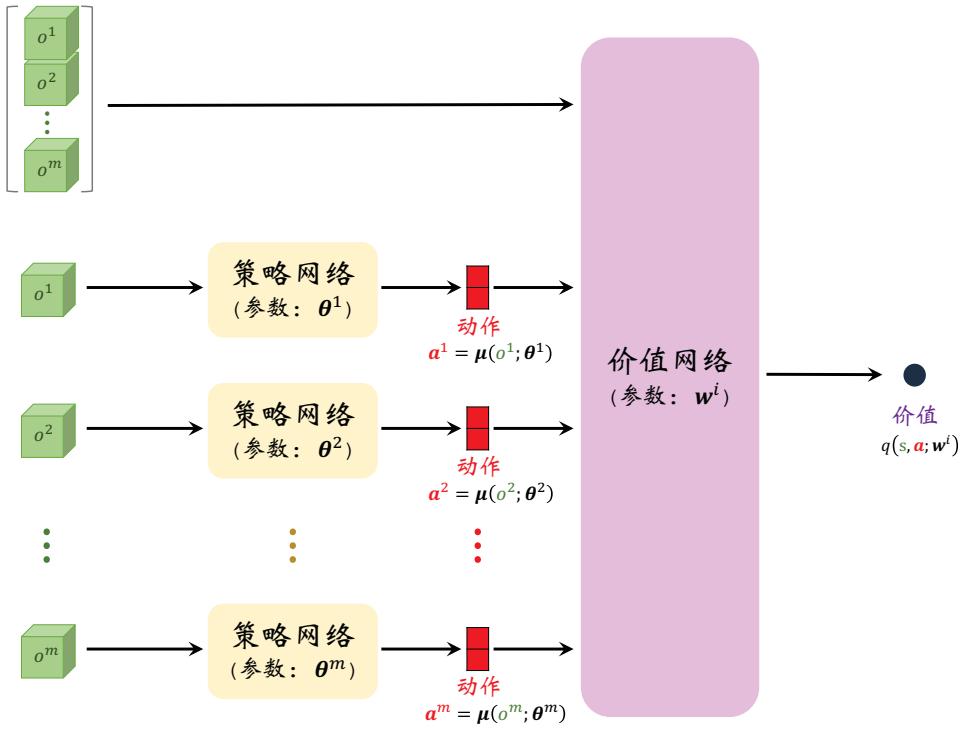

图 15.5: 所有智能体的策略网络与第 i 号智能体的价值网络。

《深度强化学习》 2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

15.4.2 算法推导

训练策略网络和价值网络的算法与第 10.2 节的单智能体 DPG 非常类似：用确定策略梯度更新策略网络，用 TD 算法更新价值网络。由于 DPG 方法是异策略 (Off-policy)，我们可以使用经验回放，重复利用过去的经验。我们用一个经验回放数组存储收集到的经验，每一条经验都是 $(s_t, \boldsymbol{a}_t, r_t, s_{t+1})$ 这样一个四元组，其中

$$\begin{aligned} s_t &= [o_t^1, \cdots, o_t^m], \\ \boldsymbol{a}_t &= [\boldsymbol{a}_t^1, \cdots, \boldsymbol{a}_t^m], \\ s_{t+1} &= [o_{t+1}^1, \cdots, o_{t+1}^m], \\ r_t &= [r_t^1, \cdots, r_t^m]. \end{aligned}$$

训练策略网络： 训练第 i 号策略网络 $\boldsymbol{\mu}(o^i; \boldsymbol{\theta}^i)$ 的目标是改进 $\boldsymbol{\theta}^i$，增大第 i 号价值网络的平均打分。所以目标函数是：

$$\widehat{J}^i(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m) = \mathbb{E}_S\Big[q\Big(S, [\boldsymbol{\mu}(O^1; \boldsymbol{\theta}^1), \cdots, \boldsymbol{\mu}(O^i; \boldsymbol{\theta}^i), \cdots, \boldsymbol{\mu}(O^m; \boldsymbol{\theta}^m)]; \boldsymbol{w}^i\Big)\Big].$$

公式中的期望是关于状态 $S = [O^1, \cdots, O^m]$ 求的。目标函数的梯度等于：

$$\nabla_{\boldsymbol{\theta}^i} \widehat{J}^i(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m) = \mathbb{E}_S\Big[\nabla_{\boldsymbol{\theta}^i} q\Big(S, [\boldsymbol{\mu}(O^1; \boldsymbol{\theta}^1), \cdots, \boldsymbol{\mu}(O^i; \boldsymbol{\theta}^i), \cdots, \boldsymbol{\mu}(O^m; \boldsymbol{\theta}^m)]; \boldsymbol{w}^i\Big)\Big].$$

接下来用蒙特卡洛近似公式中的期望。从经验回放数组中随机抽取一个状态：[1]

$$s_t = [o_t^1, \cdots, o_t^m],$$

它可以看做是随机变量 S 的一个观测值。用所有 m 个策略网络计算动作

$$\widehat{\boldsymbol{a}}_t^1 = \boldsymbol{\mu}(o_t^1; \boldsymbol{\theta}^1), \quad \cdots, \quad \widehat{\boldsymbol{a}}_t^m = \boldsymbol{\mu}(o_t^m; \boldsymbol{\theta}^m).$$

那么目标函数的梯度 $\nabla_{\boldsymbol{\theta}^i} \widehat{J}^i(\boldsymbol{\theta}^1, \cdots, \boldsymbol{\theta}^m)$ 可以近似成为：

$$\begin{aligned} \boldsymbol{g}_\theta^i &= \nabla_{\boldsymbol{\theta}^i} q\Big(s_t, \Big[\boldsymbol{\mu}(o_t^1; \boldsymbol{\theta}^1), \cdots, \boldsymbol{\mu}(o_t^i; \boldsymbol{\theta}^i), \cdots, \boldsymbol{\mu}(o_t^m; \boldsymbol{\theta}^m)\Big]; \boldsymbol{w}^i\Big) \\ &= \nabla_{\boldsymbol{\theta}^i} q\Big(s_t, [\widehat{\boldsymbol{a}}_t^1, \cdots, \widehat{\boldsymbol{a}}_t^m]; \boldsymbol{w}^i\Big). \end{aligned}$$

由于 $\widehat{\boldsymbol{a}}_t^i = \boldsymbol{\mu}(o_t^i; \boldsymbol{\theta}^i)$，用链式法则可得：

$$\boldsymbol{g}_\theta^i = \nabla_{\boldsymbol{\theta}^i} \boldsymbol{\mu}(o_t^i; \boldsymbol{\theta}^i) \cdot \nabla_{\widehat{\boldsymbol{a}}^i} q\Big(s_t, [\widehat{\boldsymbol{a}}_t^1, \cdots, \widehat{\boldsymbol{a}}_t^m]; \boldsymbol{w}^i\Big).$$

做梯度上升更新参数 $\boldsymbol{\theta}^i$：

$$\boldsymbol{\theta}^i \leftarrow \boldsymbol{\theta}^i + \beta \cdot \boldsymbol{g}_\theta^i.$$

注意，在更新第 i 号策略网络的时候，除了用到全局状态 s_t，还需要用到所有智能体的策略网络，以及第 i 号价值网络 $q\big(s, [\boldsymbol{a}^1, \cdots, \boldsymbol{a}^m]; \boldsymbol{w}^i\big)$。

训练价值网络： 训练第 i 号价值网络 $q(s, \boldsymbol{a}; \boldsymbol{w}^i)$ 的时候，可以使用 TD 算法，让价值网络的输出更好拟合价值函数 $Q_\pi^i(s, \boldsymbol{a})$。给定四元组 $(s_t, \boldsymbol{a}_t, r_t, s_{t+1})$，用所有 m 个策

[1] 更新策略网络只需要四元组 $(s_t, \boldsymbol{a}_t, r_t, s_{t+1})$ 中的 s_t，没有用其余三个。

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

略网络计算动作

$$\widehat{\boldsymbol{a}}_{t+1}^1 = \boldsymbol{\mu}(o_{t+1}^1; \boldsymbol{\theta}^1), \quad \cdots, \quad \widehat{\boldsymbol{a}}_{t+1}^m = \boldsymbol{\mu}(o_{t+1}^m; \boldsymbol{\theta}^m).$$

设 $\widehat{\boldsymbol{a}}_{t+1} = [\widehat{\boldsymbol{a}}_{t+1}^1, \cdots, \widehat{\boldsymbol{a}}_{t+1}^m]$。然后计算 TD 目标：

$$\widehat{y}_t^i \ = \ r_t^i + \gamma \cdot q\Big(s_{t+1}, \widehat{\boldsymbol{a}}_{t+1}; \ \boldsymbol{w}^i\Big).$$

再计算 TD 误差：

$$\delta_t^i \ = \ q\big(s_t, \boldsymbol{a}_t; \boldsymbol{w}^i\big) - \widehat{y}_t^i.$$

最后做梯度下降更新参数 $\boldsymbol{w}^i$：

$$\boldsymbol{w}^i \ \leftarrow \ \boldsymbol{w}^i - \alpha \cdot \delta_t^i \cdot \nabla_{\boldsymbol{w}^i} q\big(s_t, \boldsymbol{a}_t; \boldsymbol{w}^i\big).$$

这样可以让价值网络的预测 $q(s_t, \boldsymbol{a}_t; \boldsymbol{w}^i)$ 更接近 TD 目标 $\widehat{y}_t^i$。

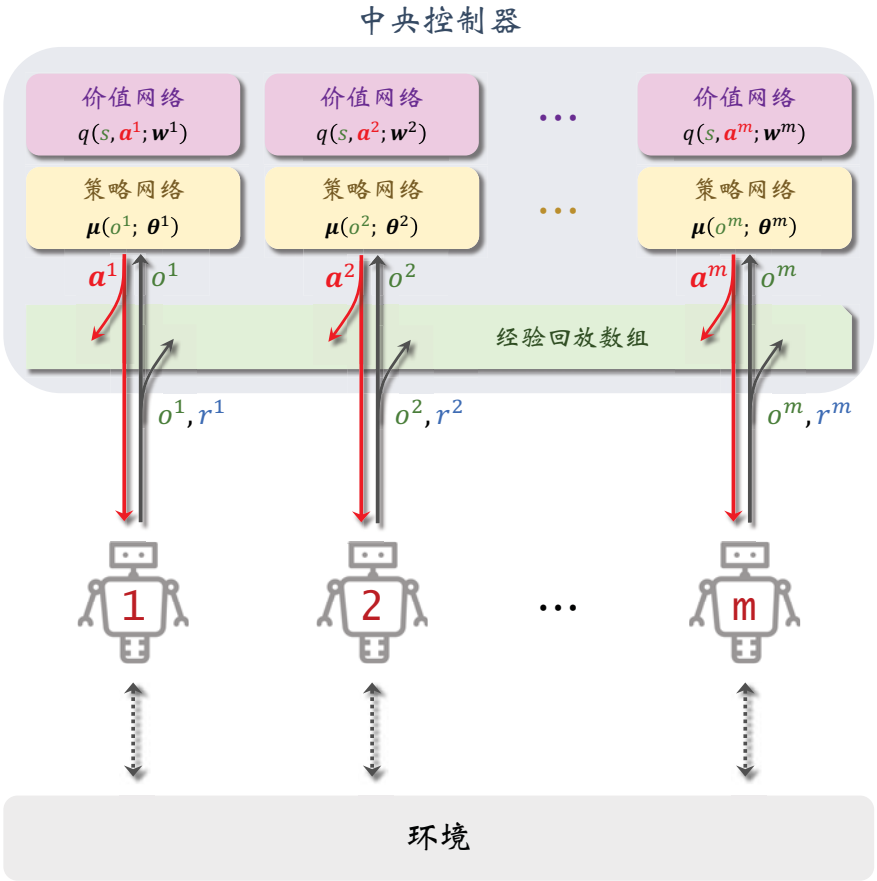

图 15.6: MADDPG 的中心化训练。

15.4.3 中心化训练

为了训练第 i 号策略网络和第 i 号价值网络，我们需要用到如下信息：从经验回放数组中取出的 s_t、$\boldsymbol{a}_t$、s_{t+1}、r_t^i、所有 m 个策略网络、以及第 i 号价值网络。很显然，一

《深度强化学习》 2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

个智能体不可能有所有这些信息，因此 MADDPG 需要“中心化训练”。

中心化训练的系统架构如图 15.6 所示。有一个中央控制器，上面有所有的策略网络和价值网络。训练过程中，策略网络部署到中央控制器上，所以智能体不能自主做决策，智能体只是执行中央控制器发来的指令。由于训练使用异策略，可以把收集经验和更新神经网络参数分开做。

用行为策略收集经验。行为策略 (Behavior Policy) 可以不同于目标策略 (Target Policy，即 $\boldsymbol{\mu}$)。行为策略是什么都无所谓，比如第 i 个智能体的行为策略可以是

$$\boldsymbol{a}^i \;=\; \boldsymbol{\mu}(o^i;\boldsymbol{\theta}^i_{\text{old}}) + \boldsymbol{\epsilon},$$

其中 $\boldsymbol{\epsilon}$ 是与 $\boldsymbol{a}^i$ 维度相同的向量，每个元素都是从正态分布中独立抽取的，相当于随机噪声。具体实现的时候，智能体把其观测 o^i 发送给中央控制器。控制器往第 i 号策略网络输出的动作向量中加入随机噪声，发送给给第 i 号智能体，智能体执行 $\boldsymbol{a}^i$。随后智能体观测到奖励 r^i，发送给控制器。控制器把每一轮的 $o^i, \boldsymbol{a}^i, r^i$ 都存入经验回放数组。

中央控制器更新策略网络和价值网络： 实际实现的时候，中央控制器上还需要有如下目标网络（图 15.6 中没有画出）：

$$\pi\big(\boldsymbol{a}^1 \,\big|\, o^1;\, \boldsymbol{\theta}^{1-}\big), \qquad \pi\big(\boldsymbol{a}^2 \,\big|\, o^2;\, \boldsymbol{\theta}^{2-}\big), \qquad \cdots, \qquad \pi\big(\boldsymbol{a}^m \,\big|\, o^m;\, \boldsymbol{\theta}^{m-}\big);$$

$$q\big(s, \boldsymbol{a};\, \boldsymbol{w}^{1-}\big), \qquad q\big(s, \boldsymbol{a};\, \boldsymbol{w}^{2-}\big), \qquad \cdots, \qquad q\big(s, \boldsymbol{a};\, \boldsymbol{w}^{m-}\big).$$

设第 i 号智能体当前的参数为：

$$\boldsymbol{\theta}^i_{\text{now}}, \qquad \boldsymbol{\theta}^{i-}_{\text{now}}, \qquad \boldsymbol{w}^i_{\text{now}}, \qquad \boldsymbol{w}^{i-}_{\text{now}}.$$

中央控制器每次从经验回放数组中随机抽取一个四元组 $(s_t, \boldsymbol{a}_t, r_t, s_{t+1})$，然后按照下面的步骤更新所有策略网络和所有价值网络：

1. 让所有 m 个目标策略网络做预测：

$$\widehat{\boldsymbol{a}}^{i-}_{t+1} \;=\; \boldsymbol{\mu}\big(o^i_{t+1};\, \boldsymbol{\theta}^{i-}_{\text{now}}\big), \qquad \forall\, i = 1, \cdots, m.$$

   把预测汇总成 $\widehat{\boldsymbol{a}}^{-}_{t+1} = [\widehat{\boldsymbol{a}}^{1-}_{t+1}, \cdots, \widehat{\boldsymbol{a}}^{m-}_{t+1}]$。

2. 让所有 m 个目标价值网络做出预测：

$$\widehat{q}^{i-}_{t+1} \;=\; q\big(s_{t+1}, \widehat{\boldsymbol{a}}^{-}_{t+1};\, \boldsymbol{w}^{i-}_{\text{now}}\big), \qquad \forall\, i = 1, \cdots, m.$$

3. 计算 TD 目标：

$$\widehat{y}^i_t \;=\; r^i_t + \gamma \cdot \widehat{q}^{i-}_{t+1}, \qquad \forall\, i = 1, \cdots, m.$$

4. 让所有 m 个价值网络做预测：

$$\widehat{q}^i_t \;=\; q\big(s_t, \boldsymbol{a}_t;\, \boldsymbol{w}^i_{\text{now}}\big), \qquad \forall\, i = 1, \cdots, m.$$

5. 计算 TD 误差：

$$\delta^i_t \;=\; \widehat{q}^i_t \;-\; \widehat{y}^i_t, \qquad \forall\, i = 1, \cdots, m.$$

6. 更新所有 m 个价值网络：

$$\boldsymbol{w}^i_{\text{new}} \;\leftarrow\; \boldsymbol{w}^i_{\text{now}} \;-\; \alpha \cdot \delta^i_t \cdot \nabla_{\boldsymbol{w}^i} q\big(s_t, \boldsymbol{a}_t;\, \boldsymbol{w}^i_{\text{now}}\big), \qquad \forall\, i = 1, \cdots, m.$$

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

7. 让所有 m 个策略网络做预测：

$$\widehat{\boldsymbol{a}}_t^i \ = \ \boldsymbol{\mu}\big(o_t^i; \boldsymbol{\theta}_{\text{now}}^i\big), \qquad \forall\, i = 1, \cdots, m.$$

把预测汇总成 $\widehat{\boldsymbol{a}}_t = [\widehat{\boldsymbol{a}}_t^1, \cdots, \widehat{\boldsymbol{a}}_t^m]$。请区别 $\widehat{\boldsymbol{a}}_t$ 与经验回放数组中抽出的 $\boldsymbol{a}_t$ 的。

8. 更新所有 m 个策略网络：$\forall\, i = 1, \cdots, m$，

$$\boldsymbol{\theta}_{\text{new}}^i \ \leftarrow \ \boldsymbol{\theta}_{\text{now}}^i \ - \ \beta \cdot \nabla_{\boldsymbol{\theta}^i} \boldsymbol{\mu}\big(s_t; \boldsymbol{\theta}_{\text{now}}^i\big) \cdot \nabla_{\boldsymbol{a}_t^i} q\big(s_t, \widehat{\boldsymbol{a}}_t; \boldsymbol{w}_{\text{now}}^i\big).$$

9. 更新所有 m 个目标策略网络：$\forall\, i = 1, \cdots, m$，

$$\begin{aligned} \boldsymbol{\theta}_{\text{new}}^{i-} \ &\leftarrow \ \tau \cdot \boldsymbol{\theta}_{\text{new}}^i \ + \ (1-\tau) \cdot \boldsymbol{\theta}_{\text{now}}^{i-}, \\ \boldsymbol{w}_{\text{new}}^{i-} \ &\leftarrow \ \tau \cdot \boldsymbol{w}_{\text{new}}^i \ + \ (1-\tau) \cdot \boldsymbol{w}_{\text{now}}^{i-}. \end{aligned}$$

改进方法： 可以用三种方法改进 MADDPG。第一，用第 10.3 节中 TD3 的三种技巧改进训练的算法：

- 用截断双 Q 学习 (Clipped Double Q-Learning) 训练价值网络 $q(s, \boldsymbol{a}; \boldsymbol{w}^i)$，$\forall\, i = 1, \cdots m$。
- 往训练算法第一步中的 $\widehat{\boldsymbol{a}}_{t+1}^{i-}$ 加入噪声。
- 减小更新策略网络和目标网络的频率，每更新 $k\ (> 1)$ 次价值网络，更新一次策略网络和目标网络。

第二，按照第 11 章中的方法，在策略网络和价值网络中使用 RNN，记忆历史观测。第三，在价值网络的结构中使用注意力机制，见下一章。

15.4.4 去中心化决策

在完成训练之后，不再需要价值网络，只需要策略网络做决策。如图 15.7 所示，把策略网络部署到对应的智能体上。第 i 号智能体可以基于本地观测的 o^i，在本地独立做决策：$\boldsymbol{a}^i = \boldsymbol{\mu}(o^i; \boldsymbol{\theta}^i)$。

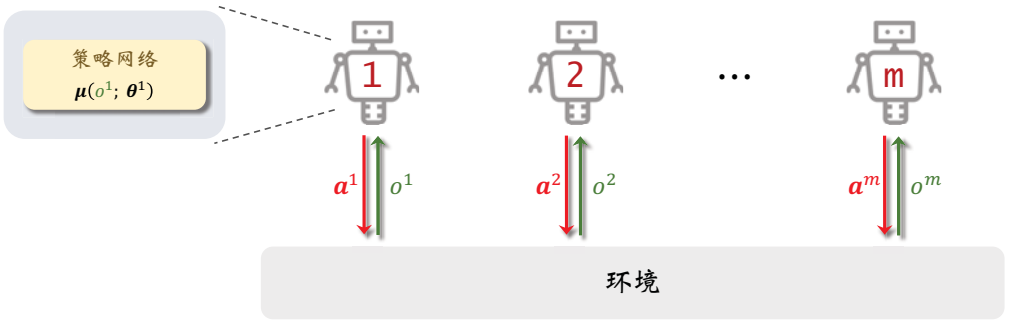

图 15.7: MADDPG 的去中心化决策。

《深度强化学习》 2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

第 十五 章 相关文献

MAN-A2C 是本书设计出来的简单方法，用于讲解非合作设定下的 MARL；MAN-A2C 这个名字并没有出现在任何文献中。本章介绍的 MADDPG 由 Lowe 等人 2017 年的论文提出 [66]。

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu

《深度强化学习》2021-02-09 尚未校对，仅供预览。
如发现错误，请告知作者 shusen.wang@stevens.edu