11 多智能体强化学习

在强化学习中，复杂的应用需要多个智能体的介入来同时学习并处理不同的任务。然而，智能体数目的增加会对管理其之间的交互带来挑战。根据每个智能体的优化问题，均衡的概念被提出并用于规范多智能体的分布式动作。结合典型的多智能体强化学习算法，我们进一步分析了在多种场景下智能体之间合作与竞争的关系，以及一般性的博弈架构如何用于建模多智能体多种类型的交互场景。通过对博弈架构中每一部分优化和均衡的分析，每一个智能体最优的多智能体强化学习策略将得到指引和进一步探索。

11.1 简介

基于规则和环境反馈，一个智能体可以通过强化学习学到动作策略并且表现优异。然而，人工智能中有很多应用具有大规模的环境背景和复杂的学习任务，这不仅要求一个智能体做出明智的动作，而且我们希望有多个智能体可以通过有限的通信共同做出明智的决策。因此，我们需要在多个智能体的情况下为每一个智能体制定有效的强化学习策略。考虑到多个智能体之间的相互交流和影响，多智能体强化学习的概念被提出并受到广泛的关注和探索。

为了方便分析和理解，在多智能体强化学习中，我们列出三个基本元素，分别是智能体、策略和效用函数。

- **智能体**: 智能体是一群具有自主决策意识的个体，它们中每一个个体都可以独立地和环境进行交互。为了能使自己获得最大的收益和最小的损失，每一个智能体会基于对其他智能体动作的观察、学习并制定自己的动作策略。在本章我们要考虑的情况中，会有多个智能体同时存在。智能体的数目为 1 时即普通强化学习的场景。
- **策略**: 在多智能体强化学习中，每一个智能体会制定策略来最大化自身的收益并且最小化

损失。其制定的策略基于智能体对环境的感知，并且会被其他智能体的策略影响。

- **效用函数**: 考虑到每个智能体自身的需求和对环境及其他智能体的依赖关系，每一个智能体都会有独自的效用函数。一般来说，效用函数定义为智能体在实现各种目标时获得的总收益和总成本之差。在多智能体的场景下，在对周围环境和其他智能体的学习过程中，每一个智能体会以最大化自身的效用函数为最终目标。

在多智能体强化学习中，每一个智能体会有自身的效用函数，并以最大化其效用价值为目标，基于对环境的观察和交互自主地学习并制定策略。由于每一个智能体在自主学习时不会考虑到其策略对其他智能体效用函数的影响，因此，在多个智能体相互交互影响下会存在竞争或合作的情况。考虑到智能体之间相互交互的多种复杂情况，博弈论普遍被用来对智能体的决策进行具体分析 (Fudenberg et al., 1991)。针对不同的多智能体强化学习的场景，可以采用不同的博弈框架来模拟交互的场景，整体上可以分为如下三种类别。

- **静态博弈**: 静态博弈是模拟智能体间交互的最基本形式。在静态博弈中，所有智能体同时做出决策，并且每一个智能体只做出一个决策动作。由于每个智能体只行动一次，所以其可以做出一些出乎常规的欺骗和背叛策略来使自己在博弈中获益。因此，在静态博弈中，每一个智能体在制定策略时需要考虑并防范其他智能体的欺骗和背叛来降低自身的损失。
- **重复博弈**: 重复博弈是多个智能体在相同的状态下采取重复多次的决策动作。因此，每个智能体的总效益函数是其在每次决策动作所带来的效益价值的总和。由于所有智能体会做出多次动作，当某个智能体在某一次动作时采取了欺骗或背叛的决策时，在未来的动作中，该智能体可能会收到其他智能体的惩罚和报复。因此，相比于静态博弈，重复博弈大大地避免了多智能体之间恶意的动作决策，从而整体上提高了所有智能体总效益价值之和。
- **随机博弈**: 随机博弈（或马尔可夫博弈）可以看作是一个马尔可夫过程，其中存在多个智能体在多个状态下多次做出动作决策。随机博弈模拟出了多个智能体做多次决策的一般情况，每个智能体会根据自身所处的状态，通过对环境的观察和对其他智能体动作的预测，做出提升自身效用函数的最佳动作决策。

在本章中，在单智能体强化学习的基础上，我们更多地关注智能体之间的交互和关联，寻求在多智能体强化学习中所有智能体之间达到均衡状态，并且每个智能体都能获得相对较高和稳定的效用函数。

11.2　优化和均衡

由于每个智能体以提高自身的效用函数为目标，多智能体强化学习可以看成一个求解多个优化问题的数学问题，其中每个智能体对应一个优化问题。为了分析智能体之间的关系，设有 m 个智能体，用 $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times, \cdots, \times \mathcal{X}_m$ 表示所有智能体的决策空间，用 $\boldsymbol{u} = (u_1(\boldsymbol{x}), \cdots, u_m(\boldsymbol{x}))$ 表示所有智能体在采取决策 $\boldsymbol{x}, \boldsymbol{x} \in \mathcal{X}$ 时的效用空间。因此，每个智能体 i，$\forall i \in \{1, 2, \cdots, m\}$，需要在和其他智能体的交互情况下，最大化其自身的效用函数。在多智能体强化学习下，一般来

说，就是同时或者顺序求解多个优化问题，来保证每个智能体都能获得最优的效用函数。

因为每个智能体的收益函数和所有智能体的决策动作相关，在求解多智能体的优化问题中，我们希望所有智能体最终都能有稳定的决策策略，在其状态下，每一个智能体都不能通过只改变自身的决策策略而使自己获得更高的收益。因而，在多智能体强化学习中，我们提出了均衡的概念。为了更好地理解和分析，在不失一般性的前提下，我们通过胆小鬼博弈（Chicken Dare Game，或被称为斗鸡博弈）及其延伸来介绍多种均衡概念。经典的胆小鬼博弈是一种静态博弈模型，其中涉及两个智能体之间的交互关系。两个智能体可以相互独立地选择怯懦（简称为“C”）或者勇敢（简称为“D”）作为自身的动作决策。基于两个智能体所有可能的动作决策，两个智能体获得的效用价值由图 11.1 所示。当两个智能体选择“D”即勇敢时，两者各自都会获得最低的效用价值 0；当其中一个智能体选择“D”即勇敢，另一个智能体选择“C”即怯懦时，选择勇敢的智能体获得其最佳的效用价值 6，选择怯懦的智能体获得相对较低的效用价值 3。当两个智能体都选择“C”即怯懦时，两者都会获得相对较高的收益 5。

	C	D
C	5, 5	3, 6
D	6, 3	0, 0

图 11.1 胆小鬼博弈

11.2.1 纳什均衡

根据图 11.1 所示的胆小鬼博弈 (Rapoport et al., 1966) 的场景，我们设定规则要求两个智能体同时做出决策。因而，当两个智能体同时选择“C”时，假设对方在保持当前决策动作，每个智能体都想要选择“D”而使自己获得更高的效用价值。当两个智能体同时选择“D”时，两者都只能获得最低的效用价值 0，因而希望改变策略“D”而获得更高的收益。然而，当一个智能体选择“C”，而另一个智能体选择“D”时，在假设对方不会改变当前决策动作的前提下，两个智能体都不能只单独改变自己的决策而提高自己的效用价值。因此，我们称一个智能体选择“C”，而另一个智能体选择“D”这种情况在当前场景下达到了纳什均衡 (Nash et al., 1950)，其定义如下：

定义 11.1 令 $(\mathcal{X}, \boldsymbol{u})$ 表示 m 个智能体下的静态场景，其中 $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times, \cdots, \times \mathcal{X}_m$ 表示智能体的策略空间。当所有智能体采取策略 $\boldsymbol{x}$，其中 $\boldsymbol{x} \in \mathcal{X}$ 时，$\boldsymbol{u} = (u_1(\boldsymbol{x}), \cdots, u_m(\boldsymbol{x}))$ 表示智能体的效用空间。我们同时设 x_i 为智能体 i 的策略，设 $\boldsymbol{x}_{-i}$ 为除智能体 i 外其他所有智能体的策略集合。当 $\forall i, x_i \in \mathcal{X}_i$ 时，

$$u_i(x_i^*, \boldsymbol{x}_{-i}^*) \geqslant u_i(x_i, \boldsymbol{x}_{-i}^*). \tag{11.1}$$

策略 $x^* \in \mathcal{X}$ 使当前场景达到纳什均衡。

纯策略纳什均衡

根据定义所示，在多智能体强化学习的静态场景下，所有智能体同时采取一次决策动作。在其他智能体的决策动作不改变的前提下，每个智能体不能通过改变当前的决策动作而获得更高的收益，我们称所有的智能体达到纯策略纳什均衡。在胆小鬼博弈的例子中存在两个纯策略纳什均衡，其中一个智能体选择怯懦动作，另一个智能体选择勇敢动作。一般来说，纯策略纳什均衡不一定存在，因为智能体的纯策略动作不能保证其他智能体通过改变当前的动作来获得更高的效用价值。

混合策略纳什均衡

在纯决策动作之外，每个智能体还可以制定并采取决策的策略，并根据策略基于不同的概率随机选择不同的决策动作。因而，智能体制定策略可以在其相互交互的过程中带来随机性和不可确定性，并可以考虑其他智能体的策略调整改变自己的策略组合而达到混合策略纳什均衡。一般来说，混合策略纳什均衡总是存在。以胆小鬼博弈为例子，我们设智能体 1 采取怯懦的概率是 p，相对应地，其采取勇敢的概率是 $1-p$。为了保证智能体 1 策略的制定没有使其对手智能体 2 的动作有偏见，从而使智能体 2 产生最佳的纯策略动作，需要满足如下关系：

$$5p + 3(1-p) = 6p + 0(1-p). \tag{11.2}$$

我们得到 $p = 0.75$。从智能体 2 的角度来说，依此类推，即当两个智能体选择“C”的概率均为 0.75，并且选择"D" 的概率为 0.25 时，两个智能体达到了混合策略纳什均衡，其中每个智能体获得的期望效益价值为 4.5。

综上所述，我们将胆小鬼博弈的结果用图 11.2 表示，其中 X 轴表示智能体 1 的效用函数，Y 轴表示智能体 2 的效用函数。基于图 11.1 表示的智能体之间的关系，点 A 对应两个智能体同时选择“C”的情况，点 B 表示智能体 1 采用动作“C”智能体 2 采用动作“D”的结果，点 C 表示智能体 1 采用动作“D”智能体 2 采用动作“C”的结果，点 D 对应两个智能体同时选择“D”的情况。因此，两个智能体采取所有可能的决策策略结果落在在四边形 $ABDC$ 区域中，其中点 B 和点 C 为纯策略纳什均衡的结果，线段 BC 的中点 E 即为混合策略纳什均衡的结果。对于所有纳什均衡的结果，两个智能体效用函数之和相同，等于 9。

11.2.2　关联性均衡

在胆小鬼博弈的纳什均衡中，两个智能体总效益之和为 9，小于所有两个智能体总效益之和的最大可能值 10。然而，两个智能体需要都选择策略“C”使得总效益之和达到 10，在绝对分布式的方式下是不稳定的。因此，为了更好地提高所有智能体的总效益价值并同时保证每个智能体能够拥有稳定的收益，关联性均衡被进一步提出。

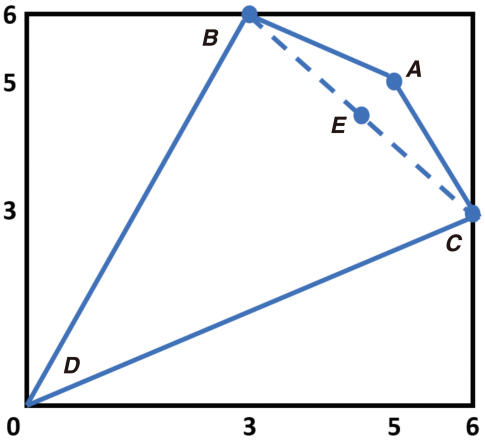

图 11.2 胆小鬼博弈中的纳什均衡

在胆小鬼博弈的例子中，我们设定两个智能体选择“CC”（第一个“C”对应智能体 1 的决策动作，第二个“C”对应智能体 2 的决策动作），“CD”、“DC”和“DD”的可能性为 $\boldsymbol{v}$。当两个智能体相关联并且设定每种情况的可能性为 $\boldsymbol{v} = [1/3, 1/3, 1/3, 0]$ 时，两个智能体的总效用价值为 9.3333，比纳什均衡的结果要高。不仅如此，假设当智能体 1 宣布将选择“C”时，为了满足每种情况的可能性保持为 $\boldsymbol{v}$，其对手智能体 2 需要采取混合策略，其选择“C”和“D”的可能性分别均为 0.5。那么当智能体 1 真实选择“C”的时候，能获得的效益价值为 $0.5 \times 5 + 0.5 \times 3 = 4$。但如果智能体 1 私自改变了决策动作“D”，在智能体 2 策略不发生改变的情况下，智能体 1 能够收到的效益价值为 $0.5 \times 6 + 0.5 \times 0 = 3$，低于选择“C”情况下的效益价值 4。相对应地，当智能体 1 宣布将选择“D”时，为了满足每种情况的可能性保持为 $\boldsymbol{v}$，其对手智能体 2 需要以 100% 的概率做出决策动作“C”，那么智能体 1 依然不能将宣布的动作私自改变到“C”而获得更高的效用价值。因此，其相关联的概率分布 $\boldsymbol{v}$ 让两个智能体达到了关联性均衡，具体定义如下：

定义 11.2 关联性均衡 (Aumann, 1987) 定义为智能体之间能够相关联实现概率分布 $\boldsymbol{v}$，并且满足如下关系

$$\sum_{\boldsymbol{x}_{-i} \in \mathcal{X}_{-i}} v(x_i^*, \boldsymbol{x}_{-i})[u_i(x_i^*, \boldsymbol{x}_{-i}) - u_i(x_i, \boldsymbol{x}_{-i})] \geqslant 0, \forall x_i \in \mathcal{X}_i, \tag{11.3}$$

其中 $\mathcal{X}_i$ 表示智能体 i 的策略空间，$\mathcal{X}_{-i}$ 表示除智能体 i 外所有其他智能体的策略空间。

因此，在假设两个智能体服从相关联分布的前提下，每个智能体不能改变当前相关联的策略而获得更高的效用价值。为了更直观地表现出关联性均衡的优势，我们在图 11.3 中用点 F 标注出本例中关联性均衡的结果。一般来说，在图中 ABC 区域中，只要满足公式 (11.3) 所示的关系，其结果均可达到关联性均衡。

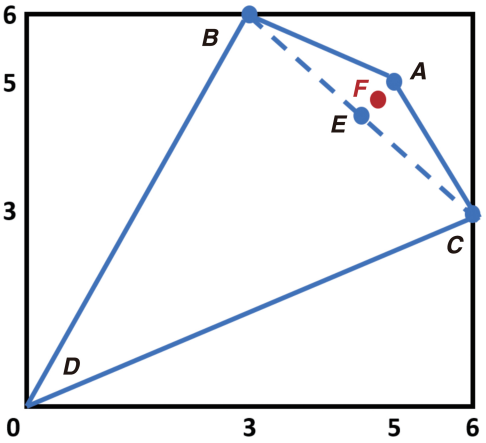

图 11.3　胆小鬼博弈中的关联性均衡

11.2.3　斯塔克尔伯格博弈

除了同时做出决策的情况，智能体之间还可能会顺序做出决策。在顺序做出决策的情况下，智能体会分别被定义为领导者和追随者，其中领导者会先做出决策，追随者随后做出决策 (Bjorn et al., 1985)。因而，领导者在决策时会有先发优势（First-Mover Advantage），可以通过预测追随者对其决策的反应来决定能够给自身带来最大收益的最佳决策。在胆小鬼博弈的例子中，如果我们扩展场景使两个智能体的决策是顺序决定的，并令智能体 1 为领导者，智能体 2 为追随者，那么智能体 1 会选择策略动作“D”，因为智能体 1 可以预测到，当其选择“D”时，为了获得更高的收益，智能体 2 一定会选择动作“C”，从而使自己的效用价值为所有可能结果中的最大值 6，并且在顺序执行的前提假设下，两个智能体能够达到斯塔克尔伯格均衡。斯塔克尔伯格均衡的定义如下：

定义 11.3 设 $\big((\mathcal{X},\boldsymbol{\Pi}),(g,f)\big)$ 为顺序执行的场景，其中有 m 个领导者同时先做出策略动作，n 个追随者同时后做出策略动作。$\mathcal{X}=\mathcal{X}_1\times\mathcal{X}_2\times,\cdots,\times\mathcal{X}_m$ 和 $\boldsymbol{\Pi}=\boldsymbol{\Pi}_1\times\boldsymbol{\Pi}_2\times,\cdots,\times\boldsymbol{\Pi}_n$ 分别表示领导者和追随者的策略空间，$g=(g_1(\boldsymbol{x}),\cdots,g_m(\boldsymbol{x}))$ 为领导者 $\boldsymbol{x}\in\mathcal{X}$ 的效用函数。$f=(f_1(\boldsymbol{\pi}),\cdots,f_n(\boldsymbol{\pi}))$ 为追随者 $\boldsymbol{\pi}\in\boldsymbol{\Pi}$ 的效用函数。设 x_i 为领导者 i 的决策策略，$\boldsymbol{x}_{-i}$ 为除领导者 i 外其他领导者的决策策略集合。同样地，设 π_j 为追随者 j 的决策策略，$\boldsymbol{\pi}_{-j}$ 为除追随者 j 外其他追随者的决策策略集合。那么对于 $\forall i,\forall j\ x_i\in\mathcal{X}_i,\pi_j\in\boldsymbol{\Pi}_j$，策略集合 $x^*\in\mathcal{X}$，$\boldsymbol{\pi}^*\in\boldsymbol{\Pi}$ 可以达到多领导者多追随者的斯塔克尔伯格均衡，并且满足如下关系：

$$g_i(x_i^*,\boldsymbol{x}_{-i}^*,\boldsymbol{\pi}^*)\geqslant g_i(x_i,\boldsymbol{x}_{-i}^*,\boldsymbol{\pi}^*)\geqslant g_i(x_i,\boldsymbol{x}_{-i},\boldsymbol{\pi}^*),\tag{11.4}$$

$$f_j(\boldsymbol{x},\boldsymbol{\pi}_j^*,\boldsymbol{\pi}_{-j}^*)\geqslant f_j(\boldsymbol{x},\boldsymbol{\pi}_j,\boldsymbol{\pi}_{-j}^*).\tag{11.5}$$

11.3 竞争与合作

在上一节中，我们以胆小鬼博弈为例子介绍了多智能体强化学习中优化和均衡的概念。除此之外，在不同的应用中，多智能体之间的关系会多种多样，在本节，我们会更多分析在分布式的场景下，多智能体之间竞争和合作的关系。在没有特殊说明的情况下，我们设所考虑的场景中存在 m 个智能体，$\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times, \cdots, \times \mathcal{X}_m$ 表示所有智能体的策略空间，$\boldsymbol{u} = (u_1(\boldsymbol{x}), \cdots, u_m(\boldsymbol{x}))$ 表示所有智能体在采用策略集合 $\boldsymbol{x}$ 的情况下的效用集合，其中 $\boldsymbol{x} \in \mathcal{X}$。

11.3.1 合作

当多个智能体相互合作的时候，一般来说，所有智能体的效用价值之和会期望高于不合作的情况下的效用价值之和，并且在分布式的场景下，每个智能体会更多地考虑自身的效用价值。因此，为了使智能体能够加入合作联盟，每个智能体自身需要在合作的情况下获得比不在合作的时候更高的效用价值。因而，其对智能体 $i, \forall i \in \{1, 2, \cdots, m\}$ 的优化问题可以归纳为

$$\begin{aligned} &\max_{x_i} \sum_{k=1}^{k=m} u_k(x_k | \boldsymbol{x}_{-k}), \\ &\text{s.t. } u_i(x_i^* | \boldsymbol{x}_{-i}^*) \geqslant u_i(x_i | \boldsymbol{x}_{-i}^*), \end{aligned} \tag{11.6}$$

11.3.2 零和博弈

零和博弈 (VINCENT, 1974) 在许多应用中被频繁使用。为了简化问题但不失一般性，我们设有两个智能体，每个智能体可以采取策略“A”或者“B”，因而，在博弈中不同情况下的效用函数如图 11.4 所示，其中，每种情况下智能体收益价值之和总是为零。在一般性的零和博弈中，每个智能体需要基于对其他智能体的动作预测最大化其自身的效用价值并且最小化其他智能体的效用价值之和。因而，其对智能体 $i, \forall i \in \{1, 2, \cdots, m\}$ 优化问题可以总结如下

$$\max_{x_i} \min_{\boldsymbol{x}_{-i}} u_i. \tag{11.7}$$

	A	B
A	1, -1	-1, 1
B	-1, 1	1, -1

图 11.4 零和博弈

基于此优化问题，在文献 (Littman, 1994) 对一个简化的踢足球问题进行分析并建模为零和博弈。在足球游戏中，存在两个智能体，每个智能体都努力地把球踢进来提高自身的效用价值并且防守对方智能体来最小化其对手的效用价值。因此，在该问题中，对于智能体 i，其优化问题具体表示为

$$\max_{\pi_i} \min_{\boldsymbol{a}_{-i}} \sum_{a_i} Q(s, a_i, \boldsymbol{a}_{-i}) \pi_i, \tag{11.8}$$

其中 π_i 表示智能体 i 的策略，a_i 代表智能体基于策略 π_i 实际的动作。在足球游戏中，智能体 i 努力提高自己的价值函数，然而其对手采取动作 $\boldsymbol{a}_{-i}$ 努力降低该价值函数。

11.3.3　同时决策下的竞争

除了零和博弈，一般来说，还有很多应用在多种智能体同时做出决策时存在竞争的关系。在同时决策下的竞争，所有智能体需要在相同的时间下同时做出决策动作，因而其优化问题可以总结如下：

$$\max_{x_i} u_i(\boldsymbol{x_i} | \boldsymbol{x_{-i}}). \tag{11.9}$$

在文献 (Hu et al., 1998) 中，Q 学习被提出来解决一般情况下多智能体之间的竞争问题。其具体算法如算法 11.35 所示，基于交互过程中经历的积累，每个智能体 i 都会维护一个 Q 列表，用于指导指定策略 π_i。随着更多经历的积累，Q 列表更新方程如下：

$$Q_i(s, a_i, \boldsymbol{a}_{-i}) = (1 - \alpha_i) Q_i(s, a_i, \boldsymbol{a}_{-i}) + \alpha_i [r_i + \gamma \pi_i(s') Q_i(s', a_i', \boldsymbol{a}'_{-i}) \boldsymbol{\pi}_{-i}(s')]. \tag{11.10}$$

算法 11.35 多智能体一般性 Q-learning

设定 Q 表格中初始值 $Q_i(s, a_i, \boldsymbol{a}_{-i}) = 1, \forall i \in \{1, 2, \cdots, m\}$。
for episode = 1 to M **do**
　设定初始状态 $s = S_0$
　for step = 1 to T **do**
　　每个智能体 i 基于 $\pi_i(s)$ 选择决策行为 a_i , 其行为是根据当前 Q 中所有智能体混合纳什均衡决策策略
　　观测经验 $(s, a_i, \boldsymbol{a}_{-i}, r_i, s')$ 并将其用于更新 Q_i
　　更新状态 $s = s'$
　end for
end for

在多智能体的场景下，由于 Q 列表的更新和其他智能体 $\boldsymbol{\pi}_{-i}$ 的策略相关，因而，智能体 i 需要同时建立并估计所有其他智能体的 Q 列表。根据由这些 Q 列表推导出对其他智能体策略 $\boldsymbol{\pi}_{-i}$

的预测，智能体 i 才可以更好制定策略 π_i，以使所有智能体的策略集合 $(\pi_i, \boldsymbol{\pi}_{-i})$ 最终达到混合策略纳什均衡的结果。

除了基本的 Q 学习，其他深度强化学习的方法也在尝试探索在多智能体强化学习中的应用。基于单智能体深度确定性策略梯度（Deep Deterministic Policy Gradient，DDPG）算法，多智能体深度确定性策略梯度（Multi-Agent Deep Deterministic Policy Gradient，MADDPG）(Lowe et al., 2017) 在所有智能体同时做出决策的场景下，为每一个智能体提供策略。MADDPG 算法如算法 11.36 所示，每个智能体对应一个分布式的行动者（Actor），为其决策提供建议。另一方面，批判者（Critic）是集中控制的，并整体维护一个和所有智能体动作集合相关的 Q 列表。

算法 11.36 多智能体深度确定性策略梯度

for episode = 1 to M **do**
  设定初始状态 $s = S_0$
  for step = 1 to T **do**
    每个智能体 i 基于当前决策策略 π_{θ_i} 选择决策行为 a_i
    同时执行所有智能体的决策行为 $\boldsymbol{a} = (a_1, a_2, \cdots, a_m)$
    将 $(s, \boldsymbol{a}, r, s')$ 存在重放缓冲区 $\mathcal{M}$
    更新状态 $s = s'$
    for 智能体 i = 1 to m **do**
      从回访缓冲区 $\mathcal{M}$ 中采样批量历史经验数据
      对于行动者和批判者网络，计算网络参数梯度并根据梯度更新参数
    end for
  end for
end for

特别来说，对于每个行动者 i，其期望回报的梯度表示为

$$\nabla_{\theta_i^\pi} J(\pi_i) = \mathbb{E}[\nabla_{\theta_i^\pi} \boldsymbol{\pi}_i(o_i|\theta_i^\pi) \nabla_{a_i} Q_i^{\boldsymbol{\pi}}(o_1, \cdots, o_m, a_1, \cdots, a_m|\theta_i^Q)], \tag{11.11}$$

其中，设 $o_1, \cdots, o_m$ 分别为 m 个智能体的观察样本。$\boldsymbol{\pi}_i$ 为智能体 i 的确定性策略，因而其决策动作满足 $a_i = \boldsymbol{\pi}_i(o_i)$。

相对应地，批判者对于智能体 i 的损失函数是 Q 值的 TD-error，表示为

$$\mathcal{L}_i = \mathbb{E}[(Q_i^{\boldsymbol{\pi}}(o_1, \cdots, o_m, a_1, \cdots, a_m|\theta_i^Q) - r_i - \gamma Q_i^{\boldsymbol{\pi}'}(o_1', \cdots, o_m', a_1', \cdots, a_m'|\theta_i^{Q'}))^2], \tag{11.12}$$

其中 $\theta_i^{Q'}$ 指 Q 预测的延迟参数，$\boldsymbol{\pi}'$ 表示在延迟参数 $\theta_i^{\pi'}$ 下的目标决策策略。

11.3.4 顺序决策下的竞争

在某些应用中，不同类型的智能体可能在做出决策时会有时间先后之分。因而，在竞争中，多个智能体之间可能会顺序做出决策动作，并且先做出决策的智能体会有先发优势。设

$((\mathcal{X}, \boldsymbol{\Pi}), (g, f))$ 为一般情况下 m 个领导者和 n 个追随者的顺序决策场景。其中 $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times, \cdots, \times \mathcal{X}_m$ 和 $\boldsymbol{\Pi} = \boldsymbol{\Pi}_1 \times \boldsymbol{\Pi}_2 \times, \cdots, \times \boldsymbol{\Pi}_n$ 分别表示领导者和追随者的决策策略空间。设 $g = (g_1(\boldsymbol{x}), \cdots, g_m(\boldsymbol{x}))$ 为领导者 $\boldsymbol{x} \in \mathcal{X}$ 的效用函数，$f = (f_1(\boldsymbol{\pi}), \cdots, f_n(\boldsymbol{\pi}))$ 为追随者 $\boldsymbol{\pi} \in \boldsymbol{\Pi}$ 的效用函数。那么追随者 $j, \forall j \in \{1, 2, \cdots, n\}$ 的优化问题可以表示为

$$\max f_j(\pi_j | \boldsymbol{\pi}_{-j}, \boldsymbol{x}). \tag{11.13}$$

领导者 $i, \forall i \in \{1, 2, \cdots, m\}$ 的优化问题为

$$\begin{aligned} &\max g_i(x_i | \boldsymbol{x}_{-i}, \boldsymbol{\pi}), \\ &\text{s.t. } \pi_j = \arg\max f_j(\pi_j | \boldsymbol{\pi}_{-j}, \boldsymbol{x}), \quad \forall j \in \{1, 2, \cdots, n\}. \end{aligned} \tag{11.14}$$

11.4　博弈分析架构

基于对多智能体之间关系的分析，我们总结出一个满足一般性多智能体博弈分析架构，如图 11.5 所示。在此架构中，我们设定一个循环迭代的场景，其中所有的智能体能够在不同时间段中多次做出决策。在同一个时间段中，我们将所有智能体进一步分为多个层级，在最高层级的智能体先做出动作，基于对高层级智能体动作的观察，低层级的智能体相对应地做出利于自身的决策，并且在每一个层级中，可以存在多个智能体同时做出决策。因而，在不同层级之间，所有智能体期望达到斯塔克尔伯格均衡，如果多个智能体存在相同层级中，根据这些智能体可否相关联，期望能够达到纳什均衡或者关联性均衡的结果而使所有智能体获得稳定效用价值。

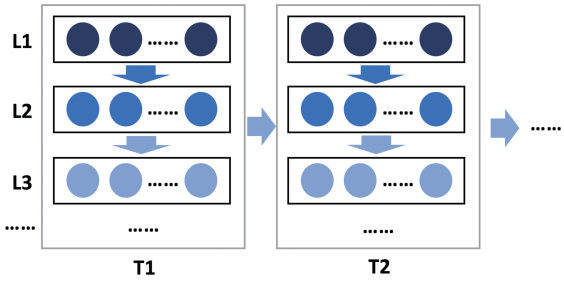

图 11.5　一般性多智能体博弈分析架构

博弈分析架构一般可以用来建模并处理所有多智能体强化学习的问题。为了更好地测试并且评估，各种多智能体强化学习平台目前已经建立并广受关注。比如 AlphaStar 可以很好模拟《星际争霸》游戏中多智能体之间的关系和动作。多智能体互联自动驾驶（MACAD）平台 (Palanisamy, 2019) 很好地学习并且模拟在公路上驾驶汽车的环境场景。谷歌研究足球 (Kurach et al., 2019) 则是一个模拟多个有自主意识的智能体一起踢足球的平台等等。基于适用于多种不同场景的多智能体

学习平台，我们期待在博弈分析架构下的多智能体强化学习策略可以得到更具体的分析和研究。

参考文献

AUMANN R J, 1987. Correlated equilibrium as an expression of bayesian rationality[J]. Econometrica: Journal of the Econometric Society: 1-18.

BJORN P A, VUONG Q H, 1985. Econometric modeling of a stackelberg game with an application to labor force participation[J].

FUDENBERG D, TIROLE J, 1991. Game theory, 1991[J]. Cambridge, Massachusetts, 393(12): 80.

HU J, WELLMAN M P, 1998. Multiagent reinforcement learning: Theoretical framework and an algorithm[C]//International Conference on Robotics and Automation (ICRA).

KURACH K, RAICHUK A, STACZYK P, et al., 2019. Google research football: A novel reinforcement learning environment[Z].

LITTMAN M L, 1994. Markov games as a framework for multi-agent reinforcement learning[C]// Proceedings of the International Conference on Machine Learning (ICML). 157-163.

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

NASH J F, et al., 1950. Equilibrium points in n-person games[J]. Proceedings of the national academy of sciences, 36(1): 48-49.

PALANISAMY P, 2019. Multi-agent connected autonomous driving using deep reinforcement learning[Z].

RAPOPORT A, CHAMMAH A M, 1966. The game of chicken[J]. American Behavioral Scientist, 10 (3): 10-28.

VINCENT P, 1974. Learning the optimal strategy in a zero-sum game[J]. Econometrica, 42(5): 885-891.