Counter-factual Multi-Agent Policy Gradients

反事实 MAPG

- 摘要

世界对于协作系统能使用 RL 方法学出分散执行的智能体很关注。因此作者提出一种叫反事实 MARL 的 AC 方法，即 COMA 算法。COMA 算法使用中心化 critic 估计 Q，使用分散式 actor 执行策略。另外该算法为了处理智能体信用分配问题，算法使用反事实 counter - factual baseline，就是在其他智能体动作固定时，边缘化该智能体的单一动作（边缘化指求得智能体的优势对于反事实基线）。COMA 还使用了一种 critic 表达，使得在一次单一向前传播，可以有效计算反事实基线。COMA 在 AC 方法中有很大改善。

- 介绍

分散执行策略有必要，但同时分散学习也需要，但可以通过只用一个学习器中心学习，通过额外的状态信息或者自由通信来个性化学习。

COMA 有三个创新点。一是使用中心训练的 critic。二是使用反事实基线，该技巧是由不同的 reward 思想得到的灵感。其中，每个智能体从一个形状化的奖励中学习，该奖励是将当前全局 reward 与该智能体的默认动作时收到的 reward 进行比较。同时不同的 reward 思想能很好解决信用分配问题，这种思想能使得智能体需要从学习器或者值估计函数中获得授权，同时一般来说选择默认动作是不清晰的。COMA 方法处理该问题的方法是使用中心化的 critic 去计算特定智能体形式的优势函数 Advantage Function，该优势函数将当前联合动作产生的估计累计回报值和当其他智能体动作固定时，能边缘化该智能体的单一动作的反事实基线进行比较。该种方法和 aristocrat utility 很相似，但是 COMA 的这种方法避免了策略和效用函数相互依赖的问题，因为反事实基线对于 PG 的贡献为 0。

因此 COMA 方法可以不使用额外激励，估计或者假设条件，就可以独立为每一个智能体从中心化 critic 中计算出来，而反事实只有在动作改变时会被推理出来。

三是使用了 critic 表达，该表达可以允许有效计算反事实基线。在一个单一前向传播，给定一个智能体，它会计算每一个不同的动作下的 Q 值，因为只有一个 critic，所以所有的 Q 都会在一次传播中计算。

- 背景
- 相关工作
- 方法
  - Independent AC

作者使用共享参数的方法，即只训练一个 actor 和一个 critic，但是策略都是单独一个智能体执行的，每个智能体都有自己的 ID，和自己的隐藏状态。

作者考虑两种 IAC 的变体，一种是每个智能体的 critic 估计 V，另一种是 critic 估计 Q，其中基于优势函数 A 计算梯度。优势函数是该动作的 Q - 该状态的 V。IAC 是前向的，但是缺乏共享信息，很难在协作策略上有突破。

  - 反事实 MAPG

该算法有三个特点。第一就不多说了。第二反事实基线：该想法由不同 reward 中得到启发，对于每一个智能体都学习一个形状化 reward：

$$D^a = r(s, \mathbf{u}) - r(s, (\mathbf{u}^{-a}, c^a))$$

就是全局 reward 和默认动作 Ca 下的 reward 进行比较，任何改善 D 的动作都会改善策略的 action，即全局 reward，因为默认动作 reward 和其他动作无关。反事实基线有效果的条件是默认动作下的 reward 可以计算出来。而学习器必须对每个智能体单独学习，因为智能体的反事实激励都是不同的。

在 AC 框架中，反事实方法会引入一个额外的估计误差。COMA 其实就是中心 critic 加上不同 reward。对 critic，每一个智能体选择动作来计算优势函数，而优势函数就是当前联合动作的 Q 减去反事实基线，公式如下：

$$A^a(s,\mathbf{u}) = Q(s,\mathbf{u}) - \sum_{u'^a} \pi^a(u'^a|\tau^a)Q(s,(\mathbf{u}^{-a},u'^a)). \tag{4}$$

对于反事实的理解：就是希望有替代方法更优一些，所以用优势函数。由上因为每个智能体的反事实基线不同，因此每个智能体都有自己的优势函数，critic 直接从智能体的经验中学习，而不是额外激励，reward 模型或者手工设计的默认动作。该优势函数和 aristocrat utility 形式一致，aristocrat utility 使用的是值函数作为基线会产生自我矛盾的问题，因为策略和效用函数在递归相互依赖，即效用函数的所有计算都和策略有关。

而 COMA 方法不一样，因为与其他策略梯度基线一样，反事实基线对梯度的期望贡献为 0，因此尽管策略和反事实基线有关，但是反事实基线的期望和策略没有关系。因此使用反事实基线不会产生自相矛盾的问题。（其他策略梯度也是和策略有关，期望和策略无关）。

COMA 结构如下：

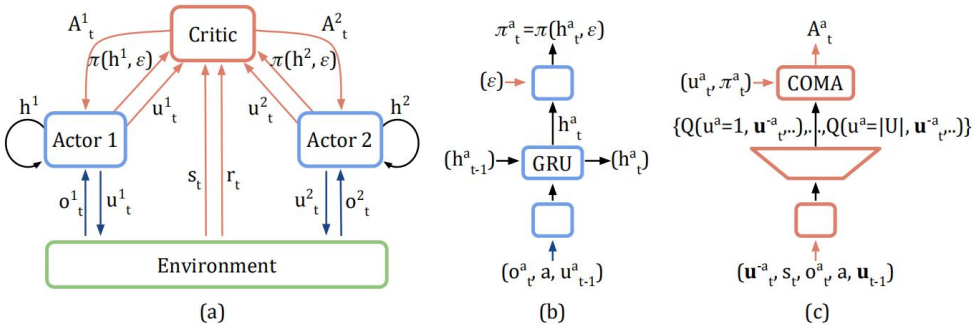


Figure 1: In (a), information flow between the decentralised actors, the environment and the centralised critic in COMA; red arrows and components are only required during centralised learning. In (b) and (c), architectures of the actor and critic.

COMA 的优势函数代替了带有 critic 评估的额外激励，带有评估的额外激励可能需要通过网络学习，复杂度高，因此 primitive COMA 也具有这种复杂度。此外，典型表达下的输出节点个数为|U|^n，复杂度也很高。为解决以上两个问题，COMA 采用一种不同的 critic 表达方法能使得 baseline 的评估高效。同时，其他智能体的动作也是网络输入，输出每个动作的 Q。因此，在单一前向传递中，critic 可以很高效地计算反事实基线，此外输出的动作节点个数为|U|，而不是|U|^n。作者说明 COMA 也可以很容易扩展到连续动作空间，通过高斯策略或者蒙特卡洛方法。

COMA 也需要收敛，以下定理说明 COMA 可以收敛到局部最优。证明是在单一智能体相同条件下的证明扩展。

Lemma 1. *For an actor-critic algorithm with a compatible TD(1) critic following a COMA policy gradient*

$$g_k = \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_a \nabla_{\theta_k} \log \pi^a(u^a|\tau^a) A^a(s,\mathbf{u})\right] \tag{5}$$

at each iteration k,

$$\liminf_k ||\nabla J|| = 0 \quad w.p.\ 1. \tag{6}$$

Proof. The COMA gradient is given by

$$g = \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_a \nabla_{\theta} \log \pi^a(u^a|\tau^a) A^a(s,\mathbf{u})\right], \tag{7}$$

$$A^a(s,\mathbf{u}) = Q(s,\mathbf{u}) - b(s,\mathbf{u}^{-a}), \tag{8}$$

where θ are the parameters of all actor policies, e.g. $\theta = \{\theta^1,\dots,\theta^{|A|}\}$, and $b(s,\mathbf{u}^{-a})$ is the counterfactual baseline defined in equation 4.

第一个考虑的策略梯度是基线的期望策略梯度。

$$g_b = -\mathbb{E}_{\boldsymbol{\pi}}\left[\sum_a \nabla_{\theta} \log \pi^a(u^a|\tau^a) b(s,\mathbf{u}^{-a})\right], \tag{9}$$

以下为推导，其中 d 为折现遍历状态分布，因为 log 的求导可以等于 f'/f，所以 10 可以推导到 11。

$$\begin{aligned} g_b = &-\sum_s d^{\boldsymbol{\pi}}(s) \sum_a \sum_{\mathbf{u}^{-a}} \boldsymbol{\pi}(\mathbf{u}^{-a}|\boldsymbol{\tau}{-}a) \cdot \\ &\sum_{u^a} \pi^a(u^a|\tau^a) \nabla_{\theta} \log \pi^a(u^a|\tau^a) b(s,\mathbf{u}^{-a}) \qquad (10) \\ = &-\sum_s d^{\boldsymbol{\pi}}(s) \sum_a \sum_{\mathbf{u}^{-a}} \boldsymbol{\pi}(\mathbf{u}^{-a}|\boldsymbol{\tau}{-}a) \cdot \\ &\sum_{u^a} \nabla_{\theta} \pi^a(u^a|\tau^a) b(s,\mathbf{u}^{-a}) \qquad (11) \\ = &-\sum_s d^{\boldsymbol{\pi}}(s) \sum_a \sum_{\mathbf{u}^{-a}} \boldsymbol{\pi}(\mathbf{u}^{-a}|\boldsymbol{\tau}{-}a) b(s,\mathbf{u}^{-a}) \nabla_{\theta} 1 \\ = &\ 0. \qquad (12) \end{aligned}$$

因此，剩下项如下：

$$g = \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_a \nabla_{\theta} \log \pi^a(u^a|\tau^a) Q(s,\mathbf{u})\right] \tag{13}$$

$$= \mathbb{E}_{\boldsymbol{\pi}}\left[\nabla_{\theta} \log \prod_a \pi^a(u^a|\tau^a) Q(s,\mathbf{u})\right]. \tag{14}$$

Writing the joint policy as a product of the independent actors:

$$\boldsymbol{\pi}(\mathbf{u}|s) = \prod_a \pi^a(u^a|\tau^a), \tag{15}$$

最后推出 COMA 的最终公式：

$$g = \mathbb{E}_{\boldsymbol{\pi}} \left[\nabla_\theta \log \boldsymbol{\pi}(\mathbf{u}|s) Q(s, \mathbf{u}) \right]. \qquad (16)$$

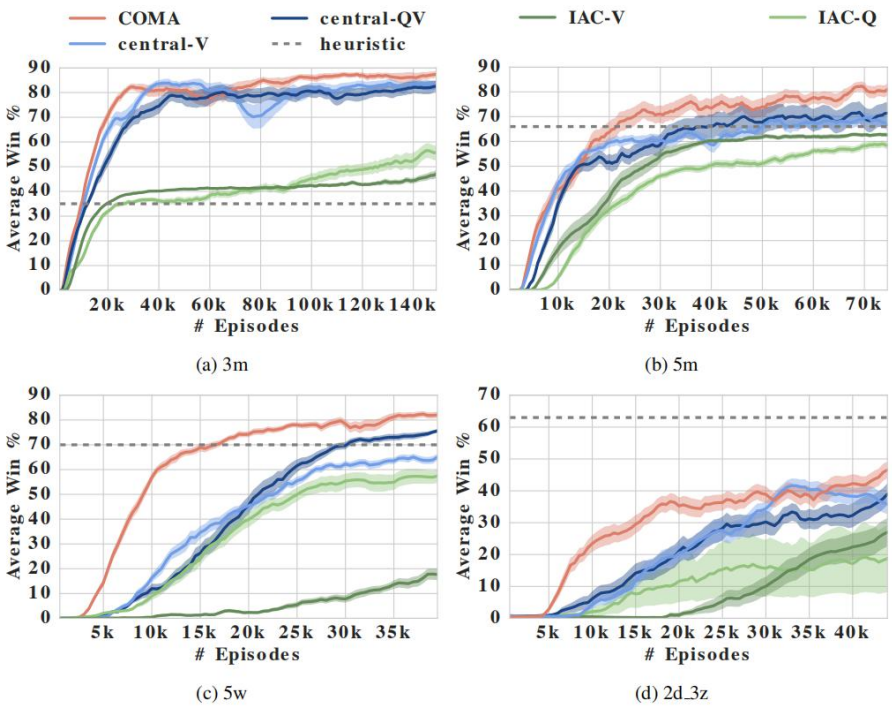
同时可以得到 COMA 可以达到局部最优，只要Π策略可微，Q 和Π更新尺度比较慢，Π要足够慢于 Q 的更新。Q 使用与Π兼容的表示。以上收敛证明的关键是 COMA 是中心化 critic，可以当作单一 critic 来证明收敛性。

算法具体过程如下所示：

Algorithm 1 Counterfactual Multi-Agent (COMA) Policy Gradients

Initialise θ_1^c, $\hat{\theta}_1^c$, θ^π
for each training episode e **do**
  Empty buffer
  for $e_c = 1$ **to** $\frac{\text{BatchSize}}{n}$ **do**
    s_1 = initial state, $t = 0$, $h_0^a = \mathbf{0}$ for each agent a
    while $s_t \neq$ terminal **and** $t < T$ **do**
      $t = t + 1$
      for each agent a **do**
        $h_t^a = \text{Actor}\left(o_t^a, h_{t-1}^a, u_{t-1}^a, a, u; \theta_i\right)$
        Sample u_t^a from $\pi(h_t^a, \epsilon(e))$
      Get reward r_t and next state s_{t+1}
    Add episode to buffer
  Collate episodes in buffer into single batch
  for $t = 1$ **to** T **do** // from now processing all agents in parallel via single batch
    Batch unroll RNN using states, actions and rewards
    Calculate TD(λ) targets y_t^a using $\hat{\theta}_i^c$
  for $t = T$ **down to** 1 **do**
    $\Delta Q_t^a = y_t^a - Q\left(s_j^a, \mathbf{u}\right)$
    $\Delta\theta^c = \nabla_{\theta^c} (\Delta Q_t^a)^2$ // calculate critic gradient
    $\theta_{i+1}^c = \theta_i^c - \alpha \Delta\theta^c$ // update critic weights
    Every C steps reset $\hat{\theta}_i^c = \theta_i^c$
  for $t = T$ **down to** 1 **do**
    $A^a(s_t^a, \mathbf{u}) = Q(s_t^a, \mathbf{u}) - \sum_u Q(s_t^a, u, \mathbf{u}^{-a}) \pi(u|h_t^a)$ // calculate COMA
    $\Delta\theta^\pi = \Delta\theta^\pi + \nabla_{\theta^\pi} \log \pi(u|h_t^a) A^a(s_t^a, \mathbf{u})$ // accumulate actor gradients
  $\theta_{i+1}^\pi = \theta_i^\pi + \alpha \Delta\theta^\pi$ // update actor weights

- 实验



(a) 3m (b) 5m

(c) 5w (d) 2d_3z

	Local Field of View (FoV)							Full FoV, Central Control		
map	heur.	IAC-V	IAC-Q	cnt-V	cnt-QV	COMA mean	COMA best	heur.	DQN	GMEZO
3m	35	47 (3)	56 (6)	83 (3)	83 (5)	**87** (3)	98	74	-	-
5m	66	63 (2)	58 (3)	67 (5)	71 (9)	**81** (5)	95	98	99	100
5w	70	18 (5)	57 (5)	65 (3)	76 (1)	**82** (3)	98	82	70	74[3]
2d_3z	**63**	27 (9)	19 (21)	36 (6)	39 (5)	47 (5)	65	68	61	90

- 总结

作者方法为 COMA，使用中心化 critic 为分散的智能体估计反事实优势。COMA 解决信用分配问题，使用反事实基线，在其他智能体动作固定时，边缘化单一智能体动作。

未来工作仍然是扩展智能体数量，希望有样本效率更高的变种。

《Social Influence as Intrinsic Motivation for Multi-Agent Deep Reinforcement Learning》

COMA notes:

https://zhuanlan.zhihu.com/p/72909208

https://blog.csdn.net/qq_38638132/article/details/103590015?utm_source=distribute.pc_relevant.none-task