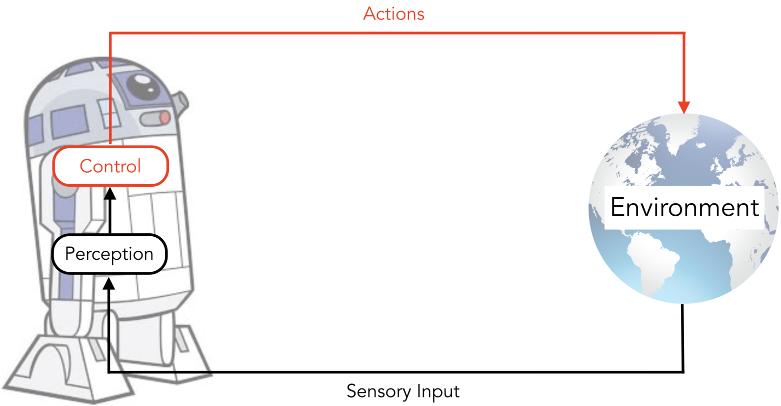
Counterfactual Multi-Agent Policy Gradients

Shimon Whiteson
Dept. of Computer Science
University of Oxford

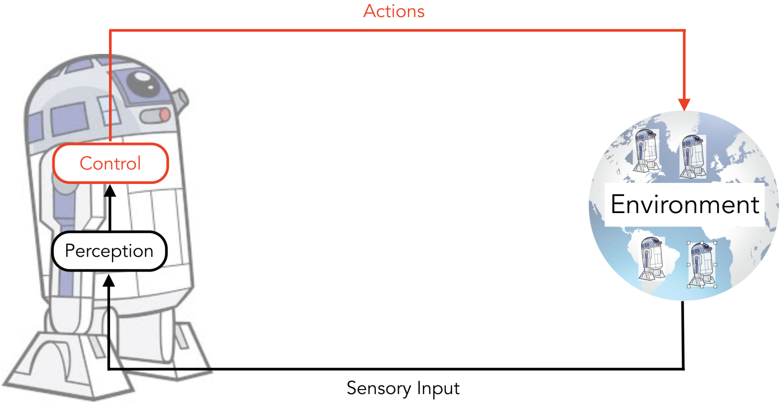
joint work with Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, and Nantas Nardelli

July 6, 2017

Single-Agent Paradigm



Multi-Agent Paradigm



Multi-Agent Systems are Everywhere

Types of Multi-Agent Systems

- *Cooperative:*
  - Shared team reward
  - Coordination problem
- *Competitive:*
  - Zero-sum games
  - Individual opposing rewards
  - Minimax equilibria
- *Mixed:*
  - General-sum games
  - Nash equilibria
  - What is the question?

Coordination Problems are Everywhere

Multi-Agent MDP

- All agents see the global state s
- Individual actions: $u^a \in U$
- State transitions: $P(s'|s, \mathbf{u}) : S \times \mathbf{U} \times S \to [0, 1]$
- Shared team reward: $r(s, \mathbf{u}) : S \times \mathbf{U} \to \mathbb{R}$
- Equivalent to an MDP with a factored action space

Dec-POMDP

- Observation function: $O(s, a) : S \times A \rightarrow Z$
- Action-observation history: $\tau^a \in T \equiv (Z \times U)^*$
- Decentralised policies: $\pi^a(u^a|\tau^a) : T \times U \rightarrow [0, 1]$
- Natural decentralisation: communication and sensory constraints
- Artificial decentralisation: coping with joint action space
- Centralised learning of decentralised policies

Key Challenges

- Curse of dimensionality in actions
- Multi-agent credit assignment
- Modelling other agents' information state

Single-Agent Policy Gradient Methods

- Optimise π_θ with gradient ascent on expected return:

$$J_\theta = \mathbb{E}_{s \sim \rho^\pi(s), u \sim \pi_\theta(s,\cdot)} \left[r(s,u) \right]$$

- Good when:
  - *Greedification* is hard, e.g., continuous actions
  - Policy is simpler than value function

- Policy gradient theorem [Sutton et al. 2000]:

$$\nabla_\theta J_\theta = \mathbb{E}_{s \sim \rho^\pi(s), u \sim \pi_\theta(s,\cdot)} \left[\nabla_\theta \log \pi_\theta(u|s) Q^\pi(s,u) \right]$$

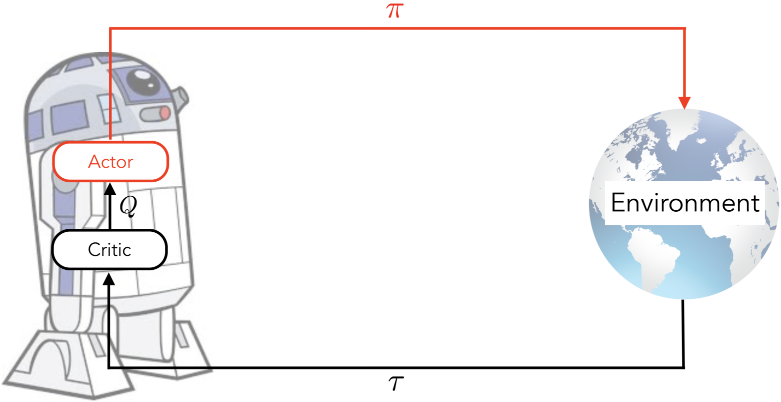
- REINFORCE [Williams 1992]:

$$\nabla_\theta J_\theta \approx g(\tau) = \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(u_t|s_t) R_t$$

Single-Agent Actor-Critic Methods [Sutton et al. 00]

- Reduce variance in $g(\tau)$ by learning a *critic* $Q(s, u)$:

$$g(\tau) = \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(u_t|s_t) Q(s_t, u_t)$$



Single-Agent Baselines

- Further reduce variance with a *baseline* $b(s)$:

$$g(\tau) = \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(u_t|s_t)(Q(s_t, u_t) - b(s_t))$$

- $b(s) = V(s) \implies Q(s, u) - b(s) = A(s, a)$, the *advantage function*:

$$g(\tau) = \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(u_t|s_t) A(s_t, u_t)$$

- *TD-error* $r_t + \gamma V(s_{t+1}) - V(s)$ is an unbiased estimate of $A(s_t, u_t)$:

$$g(\tau) = \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(u_t|s_t)(r_t + \gamma V(s_{t+1}) - V(s_t))$$

Single-Agent Deep Actor-Critic Methods

- Actor and critic are both deep neural networks
  - Convolutional and recurrent layers
  - Actor and critic share layers
- Both trained with stochastic gradient descent
  - Actor trained on policy gradient
  - Critic trained on TD(λ) or Sarsa(λ):

$$\mathcal{L}_t(\psi) = (y^{(\lambda)} - C(\cdot_t, \psi))^2$$

$$y^{(\lambda)} = (1-\lambda)\sum_{n=1}^{\infty} \lambda^{n-1} G_t^{(n)}$$

$$G_t^{(n)} = \sum_{k=1}^{n} \gamma^{k-1} r_{t+k} + \gamma^n C(\cdot_{t+n}, \psi)$$

Independent Actor-Critic

- Inspired by *independent Q-learning* [Tan 1993]
  - Each agent learns independently with its own actor and critic
  - Treats other agents as part of the environment
- Speed learning with *parameter sharing*
  - Different inputs, including a, induce different behaviour
  - Still independent: critics condition only on τ^a and u^a
- Variants:
  - IAC-V: TD-error gradient using $V(\tau^a)$
  - IAC-Q: Advantage-based gradient using $A(\tau^a, u^a) = Q(\tau^a, u^a) - V(\tau^a)$
- Limitations:
  - Nonstationary learning
  - Hard to learn to coordinate
  - Multi-agent credit assignment

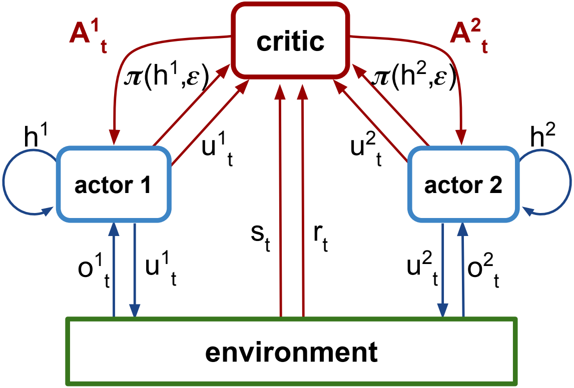
Counterfactual Multi-Agent Policy Gradients

- Centralised critic: stabilise learning to coordinate
- Counterfactual baseline: tackle multi-agent credit assignment
- Efficient critic representation: scale to large NNs

Centralised Critic

Centralisation → Hard greedification → actor-critic

$$g_a(\tau) = \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(u_t^a|\tau_t^a)(r_t + \gamma V(s_{t+1}) - V(s_t))$$



Wonderful Life Utility [Wolpert & Tumer 2000]

Difference Rewards [Tumer & Agogino 2007]

- Per-agent shaped reward:

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

where c^a is a *default action*

- Important property:

$$D^a(s, (\mathbf{u}^{-a}, \dot{u}^a)) > D^a(s, \mathbf{u}) \implies r(s, (\mathbf{u}^{-a}, \dot{u}^a)) > r(s, (\mathbf{u}^{-a}, a))$$

- Limitations:
  - Need (extra) simulation to estimate counterfactual $r(s, (\mathbf{u}^{-a}, c^a))$
  - Need expertise to choose c^a

Counterfactual Baseline

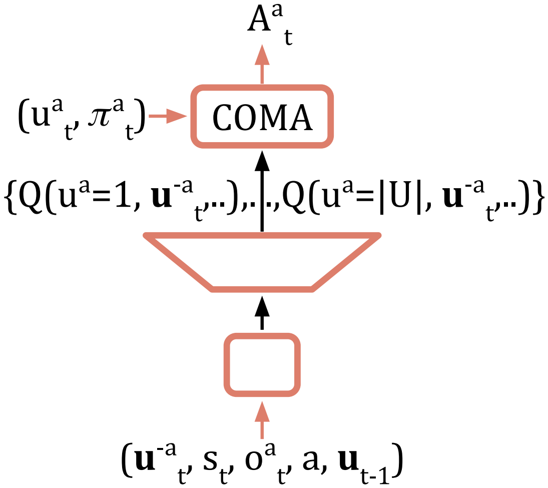
- Use $Q(s, \mathbf{u})$ to estimate difference rewards:

$$g_a(\tau) = \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(u_t^a | \tau_t^a) A^a(s_t, \mathbf{u}_t)$$

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

- Baseline marginalises out u^a
- Critic obviates need for extra simulations
- Marginalised action obviates need for default

Efficient Critic Representation



Starcraft

Starcraft Micromanagement [Synnaeve et al. 2016]

Centralised Performance

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

Decentralised Starcraft Micromanagement

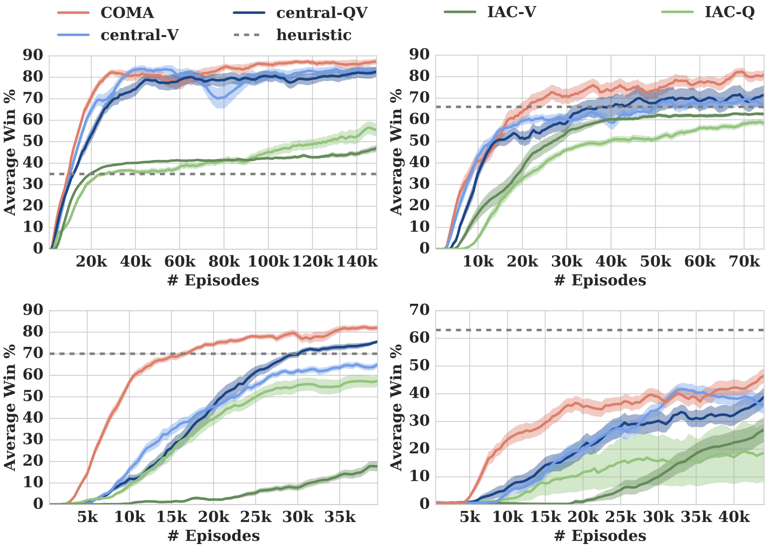
Heuristic Performance

	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	.56	.83	.83	**.87**	.98	.74	-	-
5m	.66	.63	.58	.67	.71	**.81**	.95	.98	.99	1.
5w	.70	.18	.57	.65	.76	**.82**	.98	.82	.70	.74
2d_3z	**.63**	.27	.19	.36	.39	.47	.65	.68	.61	.90

Baseline Algorithms

- *IAC-V:* independent actor-critic with $V(\tau^a)$
- *IAC-Q:* independent actor-critic with $A(\tau^a, u^a) = Q(\tau^a, u^a) - V(\tau^a)$
- *Central-V:* centralised critic $V(s)$ with TD-error-based gradient
- *Central-QV:*
  - Centralised critics $Q(s, \mathbf{u})$ and $V(s)$
  - Advantage gradient $A(s, \mathbf{u}) = Q(s, \mathbf{u}) - V(s)$
  - COMA but with $b(s) = V(s)$

Results (3m, 5m, 5w, 2d-3z)



Compared to Centralised Controllers

	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	.56	.83	.83	**.87**	.98	.74	-	-
5m	.66	.63	.58	.67	.71	**.81**	.95	.98	.99	1.
5w	.70	.18	.57	.65	.76	**.82**	.98	.82	.70	.74
2d_3z	**.63**	.27	.19	.36	.39	.47	.65	.68	.61	.90

Future Work

- Factored centralised critics for many agents
- Multi-agent exploration
- Starcraft macromanagement

Paper

Counterfactual Multi-Agent Policy Gradients

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson

`https://arxiv.org/abs/1705.08926`

Thank You Microsoft!

This work was made possible thanks to a generous donation of Azure cloud credits from Microsoft.