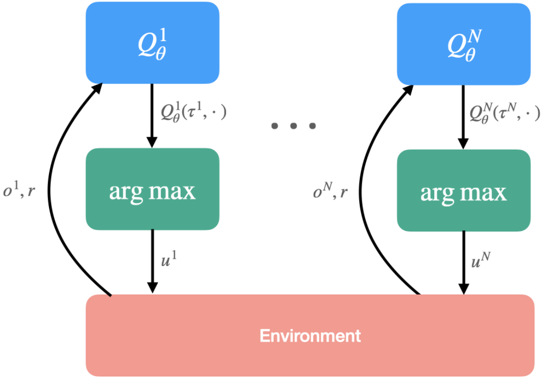
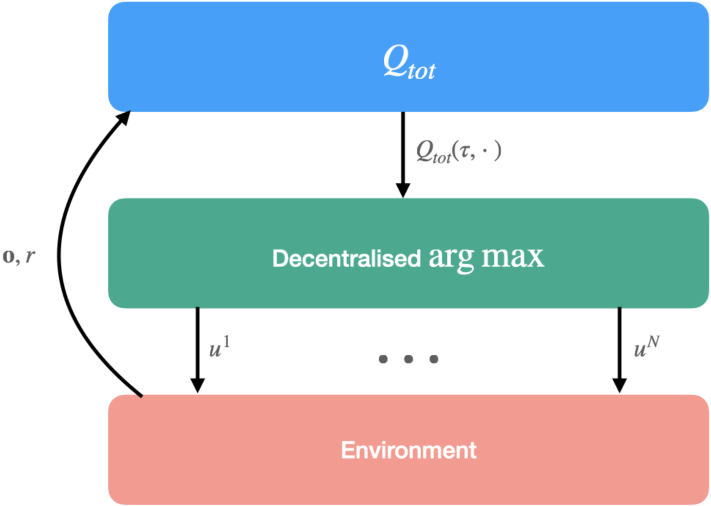
Independent Q-Learning [Tan 1993]

- Each agent learns its own Q-function
- Treats others as part of environment
- Speed learning with *parameter sharing*
- Different inputs, including a, induce different behaviours
- Still independent: value functions condition only on τ^a and u^a
- Nonstationary learning



Centralised Q-Functions

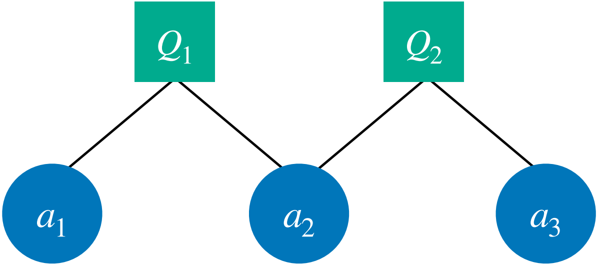


Factored Joint Value Functions

- *Factored value functions* [Guestrin et al. 2003] can improve scalability:

$$Q_{tot}(\boldsymbol{\tau}, \mathbf{u}; \boldsymbol{\theta}) = \sum_{e=1}^{E} Q_e(\tau^e, \mathbf{u}^e; \theta^e)$$

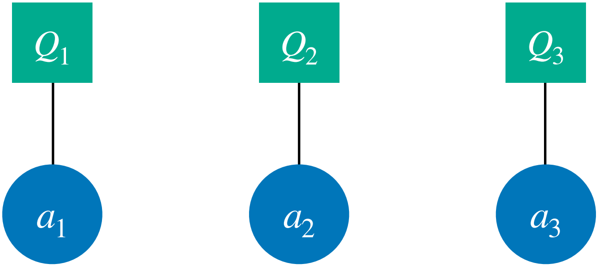
where each e indicates a subset of the agents



Value Decomposition Networks [Sunehag et al., 2017]

- Most extreme factorisation: one per agent:

$$Q_{tot}(\boldsymbol{\tau}, \mathbf{u}; \boldsymbol{\theta}) = \sum_{a=1}^{N} Q_a(\tau^a, u^a; \theta^a)$$



Decentralisability

- Added benefit of decentralising the max and arg max:

$$\max_{\mathbf{u}} Q_{tot}(\boldsymbol{\tau}, \mathbf{u}; \boldsymbol{\theta}) = \sum \max_{u^a} Q_a(\tau^a, u^a; \theta^a)$$

$$\arg\max_{\mathbf{u}} Q_{tot}(\boldsymbol{\tau}, \mathbf{u}; \boldsymbol{\theta}) = \begin{pmatrix} \arg\max_{u^1} Q_1(\tau^1, u^1; \theta^1) \\ \vdots \\ \arg\max_{u^n} Q_n(\tau^n, u^n; \theta^n) \end{pmatrix}$$

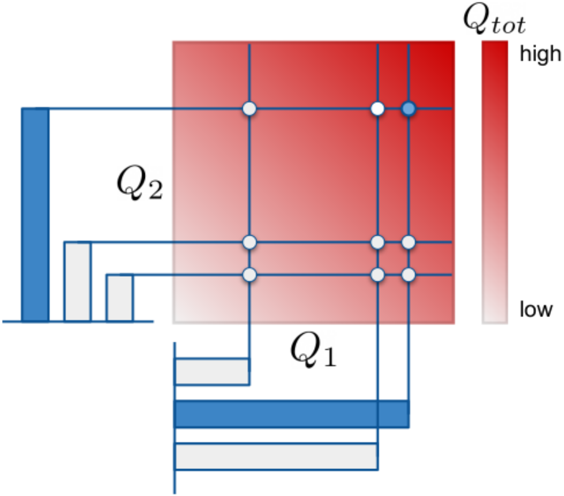
- DQN loss with centralised Q-function:

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^{b} \left[\left(y_i^{\text{tot}} - Q_{tot}(\boldsymbol{\tau}, \mathbf{u}; \boldsymbol{\theta}) \right)^2 \right],$$

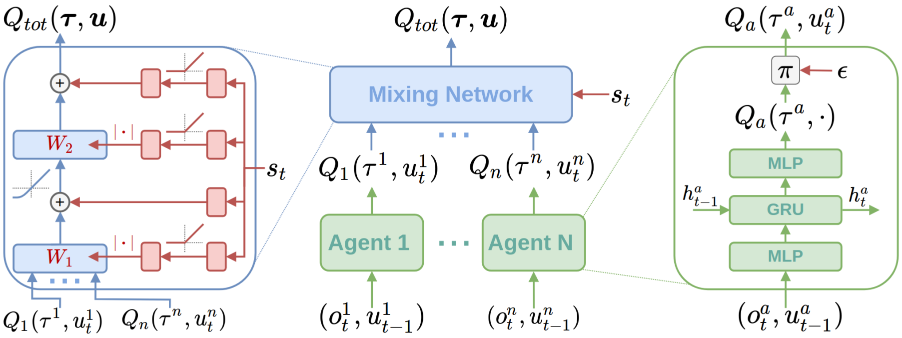
$$y_i^{\text{tot}} = r_i + \gamma \max_{\mathbf{u}'} Q_{tot}(\boldsymbol{\tau}'_i, \mathbf{u}'; \boldsymbol{\theta}^-)$$

QMIX's Monotonicity Constraint

To decentralise max / arg max, it suffices to enforce: $\frac{\partial Q_{tot}}{\partial Q_a} \geq 0, \ \forall a \in A$



QMIX [Rashid et al. 2018]



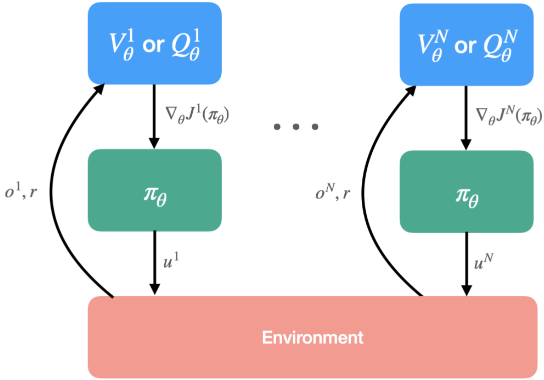
- Agent network: represents $Q_i(\tau^a, u^a; \theta^a)$
- Mixing network: represents $Q_{tot}(\boldsymbol{\tau})$ using nonnegative weights
- Hypernetwork: generates weights of hypernetwork based on global s

Beyond QMIX

- Better representations:
  - QPLEX [Wang et al. 2020]
  - MAVEN [Mahajan et al. 2019]
- Better exploration:
  - MAVEN [Mahajan et al. 2019]
  - UneVEn [Gupta et al. 2020]
- Leveraging unrestricted value functions:
  - QTRAN [Son et al. 2019]
  - QTRAN++ [Son et al. 2020]
  - WQMIX [Rashid et al. 2020]

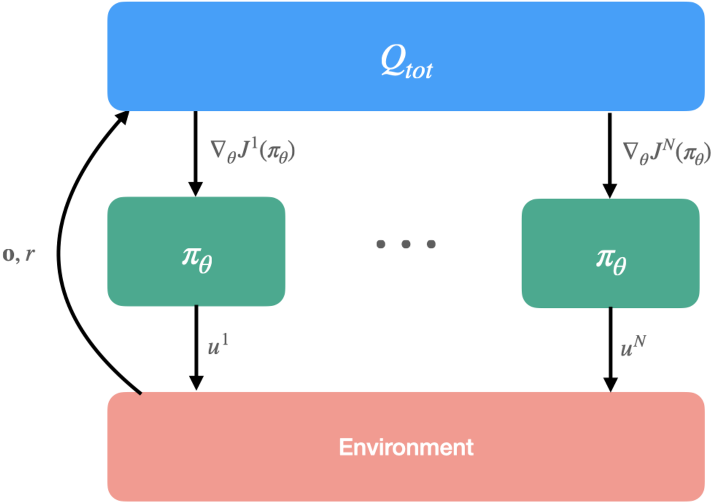
Independent Actor-Critic

- Analogous to independent Q-learning
- Actors execute decentralised policies: $\pi^a(u^a|\tau^a)$
- Each actor has its own critic $V^a(\tau^a)$ or $Q^a(u^a|\tau^a)$
- No attempt to model joint values
- Parameter sharing in both actor and critic



Centralised Critics

Centralised $V(s, \boldsymbol{\tau})$ or $Q(s, \boldsymbol{\tau}, \mathbf{u}) \rightarrow$ hard greedification $\rightarrow$ actor-critic



Counterfactual Multi-Agent Policy Gradients (COMA) [Foerster et al. 2018]

- Centralised critic with decentralised actors
- Counterfactual baseline addresses *multi-agent credit assignment*
- Estimated gradient for agent a:

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

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