Deep Reinforcement Learning

Actor-Critic

Actor-Critic

$$\theta^{new} \leftarrow \theta^{old} + \eta \nabla \bar{R}_{\theta^{old}}$$

$$\nabla \bar{R}_\theta \approx \frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T_n} R(\tau^n) \nabla log p(a_t^n | s_t^n, \theta)$$

Evaluated by critic

Advantage Function: $r_t^n - \left(V^{\pi_\theta}(s_t^n) - V^{\pi_\theta}(s_{t+1}^n)\right)$

Baseline is added

The reward r_t^n we truly obtain when taking action a_t^n

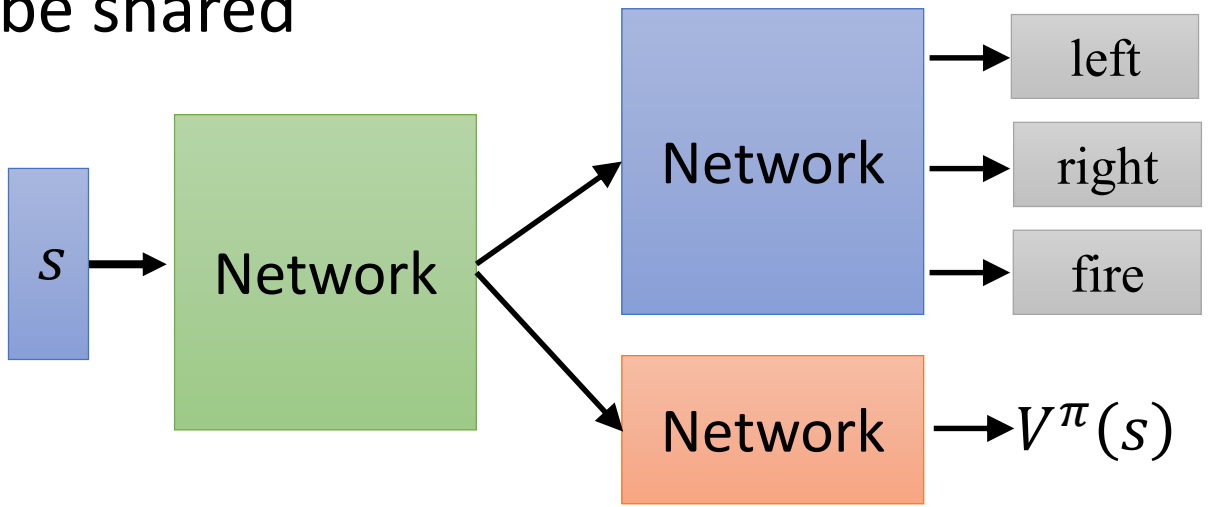
Expected reward r_t^n we obtain if we use actor π_θ

Positive advantage function ⟺ Increasing the prob. of action a_t^n

Negative advantage function ⟺ decreasing the prob. of action a_t^n

Actor-Critic

- Tips
  - The parameters of actor $\pi(s)$ and critic $V^{\pi}(s)$ can be shared

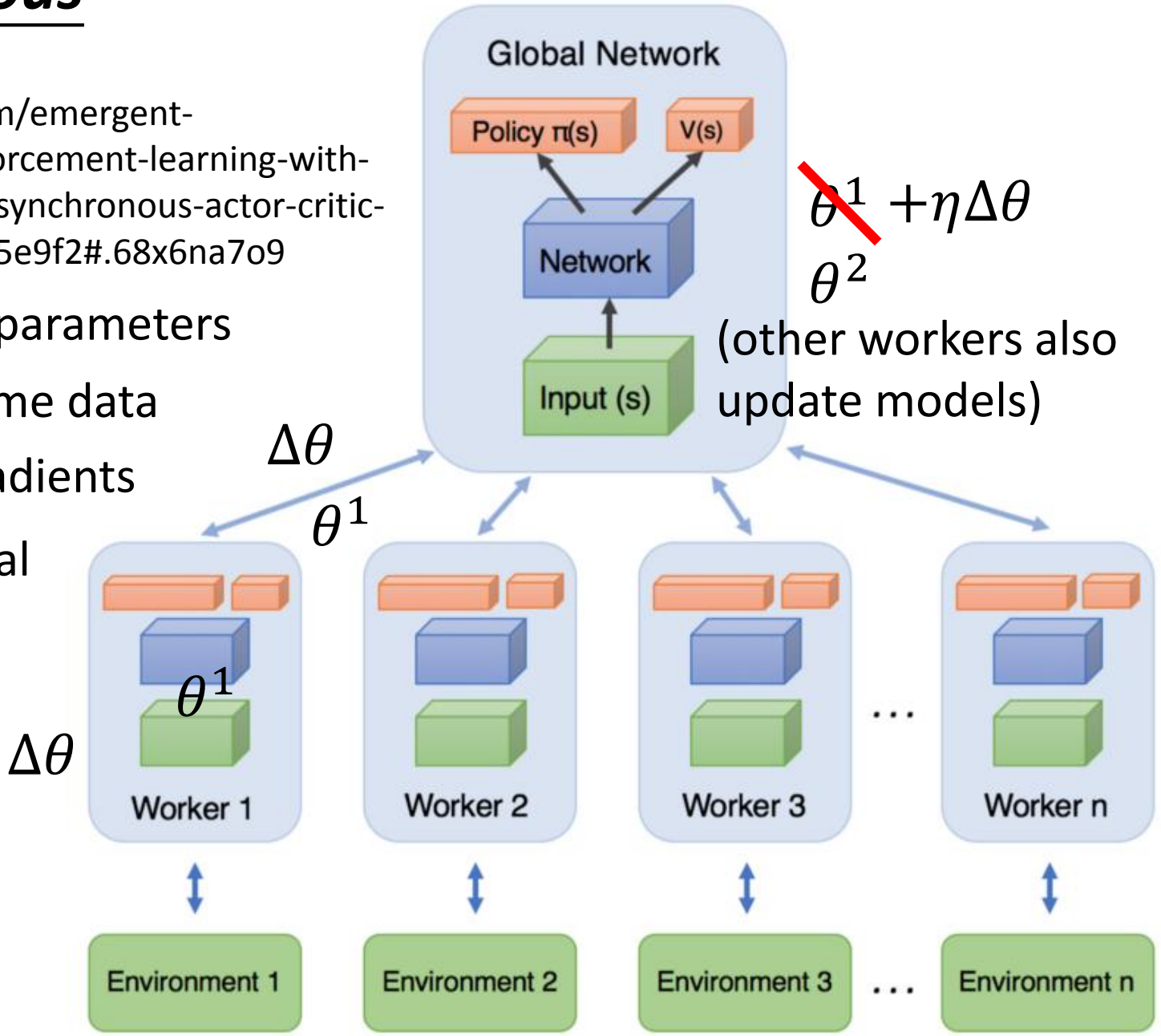


  - Use output entropy as regularization for $\pi(s)$
    - Larger entropy is preferred → exploration

Asynchronous

Source of image: https://medium.com/emergent-future/simple-reinforcement-learning-with-tensorflow-part-8-asynchronous-actor-critic-agents-a3c-c88f72a5e9f2#.68x6na7o9

1. Copy global parameters
2. Sampling some data
3. Compute gradients
4. Update global models



Demo of A3C

- DeepMind https://www.youtube.com/watch?v=nMR5mjCFZCw

Demo of A3C

- DeepMind https://www.youtube.com/watch?v=0xo1Ldx3L5Q