Towards Understanding Asynchronous Advantage Actor-critic : Convergence and Linear Speedup

Hyelin Choi

Department of Mathematics
Sungkyunkwan University

May 9, 2024

Definition

- <u>Advantage function</u> $A_\pi(s, a) := Q_\pi(s, a) - V_\pi(s)$

- <u>Initial state distribution</u> η

Definition

- Discounted state visitation measure (induced by policy π)

$$d_\pi(s) := (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}(s_t = s \mid s_0 \sim \eta, \pi)$$

- State-action visitation distribution

$$d_\pi(s,a) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}(s_t = s \mid s_0 \sim \eta, \pi)\pi(a|s)$$

Definition

- <u>KL divergence</u>

$$
\begin{aligned}
D_{KL}(P \| Q) &= \sum_x P(x) \log \left(\frac{P(x)}{Q(x)} \right) \\
&= \sum_x P(x) \log P(x) - \sum_x P(x) \log Q(x) \\
&= \mathbb{E}_P \left[\log P(x) \right] - \mathbb{E}_P \left[\log Q(x) \right]
\end{aligned}
$$

A3C

Asynchronous Advantage Actor-critic

Actor-critic

- <u>Actor network</u>

Policy Gradient Method $\longrightarrow$ Policy optimization $\pi(a|s)$

- <u>Critic network</u>

Temporal Difference Learning Algorithm $\longrightarrow$ Policy evaluation $V(s)$

A3C

Asynchronous $\boxed{\text{Advantage}}$ Actor-critic

A3C

- <u>Actor network</u>

Advantage Function

Policy Gradient Method → Policy optimization

- <u>Critic network</u>

Temporal Difference Learning Algorithm → Policy evaluation

A3C

- <u>Policy Gradient Method</u>

$$\max_{\theta \in \mathbb{R}^d} J(\theta) \text{ with } J(\theta) := (1-\gamma)\mathbb{E}_{s\sim\eta}\left[V_{\pi_\theta}(s)\right]$$

$$\theta_{k+1} = \theta_k + \alpha \nabla J(\theta_k)$$

$$\text{where } \nabla J(\theta) = \mathbb{E}_{s,a\sim d_\theta}\left[\nabla \log \pi_\theta(s,a) Q_{\pi_\theta}(s,a)\right]$$

A3C

- <u>Policy Gradient Method</u>

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}\left[G_t \nabla_\theta \log \pi_\theta(a_t|s_t)\right]$$ REINFORCE

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}\left[Q_{\pi_\theta}(s,a) \nabla_\theta \log \pi_\theta(a_t|s_t)\right]$$ Q-value Actor-Critic

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}\left[A(s,a) \nabla_\theta \log \pi_\theta(a_t|s_t)\right]$$ Advantage Actor-Critic

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}\left[\left(r + \gamma V_{\pi_\theta}(s') - V_{\pi_\theta}(s)\right) \nabla_\theta \log \pi_\theta(a|s)\right]$$ TD Actor-Critic

A3C

$$\nabla J(\theta) = \mathbb{E}_{s,a\sim d_\theta}\left[\nabla \log \pi_\theta(s,a) Q_{\pi_\theta}(s,a)\right]$$

$$\Downarrow$$

$$\nabla J(\theta) = \mathbb{E}_{s,a\sim d_\theta}\left[\nabla \log \pi_\theta(s,a) \left(Q_{\pi_\theta}(s,a) - V_{\pi_\theta}(s)\right)\right]$$

- Need to show: $\mathbb{E}_{s,a\sim d_\theta}\left[\nabla \log \pi_\theta(s,a) B(s)\right] = 0$

A3C

$\boxed{\text{Asynchronous}}$ Advantage Actor-critic

A3C

Synchronous vs Asynchronous

A3C

- <u>Actor step</u>

$$\theta_{k+1} = \theta_k + \alpha \mathbb{E}_{s,a\sim d_\theta}\left[(Q_{\pi_\theta}(s,a) - V_{\pi_\theta}(s))\underbrace{\nabla \log \pi_\theta(s,a)}_{\psi_\theta(s_k,a_k)}\right]$$

$$\theta_{k+1} = \theta_k + \alpha\left[\left(r(s_k,a_k,s_{k+1}) + \gamma\phi(s_{k+1})^T w_k - \phi(s_k)^T w_k\right)\psi_\theta(s_k,a_k) + \lambda\psi_\theta(x^p)\right]$$

A3C

- <u>Regularization</u>

η_p: prior distribution of states

π_p: prior policy

Regularization term encourages π_θ to imitate π_p, incorporating prior knowledge into training process.

$$
\begin{aligned}
J_\lambda(\theta) &:= J(\theta) - \lambda \mathbb{E}_{s \sim \eta_p} \left[D_{KL} \left(\pi_p(\cdot|s) | \pi_\theta(\cdot|s) \right) \right] \\
&= J(\theta) + \lambda R(\theta)
\end{aligned}
$$

A3C

- Bellman equation

$$V_{\pi_\theta}(s) = \mathbb{E}_{a\sim\pi_\theta(\cdot|s),\, s'\sim\mathcal{P}(\cdot|s,a)}\left[r(s,a,s') + \gamma V_{\pi_\theta}(s')\right]$$

A3C

- Value Function Approximation

state feature mapping

$$V_{\pi_\theta}(s) \approx \hat{V}_w(s) = \phi(s)^T w$$

A3C

- <u>KL divergence</u>

$$
\begin{aligned}
D_{KL}(P\|Q) &= \sum_x P(x)\log\left(\frac{P(x)}{Q(x)}\right) \\
&= \sum_x P(x)\log P(x) - \sum_x P(x)\log Q(x) \\
&= \mathbb{E}_P\left[\log P(x)\right] - \mathbb{E}_P\left[\log Q(x)\right]
\end{aligned}
$$

A3C

$$\psi_\theta(s, a) := \nabla \log \pi_\theta(s, a)$$

$$x^p := (s^p \sim \eta_p, a^p \sim \pi_p(\cdot|s_p))$$

A3C

- <u>Unbiased Estimator</u>

$$\nabla J_\lambda(\theta) = \nabla J(\theta) + \lambda \nabla R(\theta)$$

$$\hat{\nabla} J_\lambda(\theta) = \hat{\nabla} J(\theta) + \lambda \hat{\nabla} R(\theta)$$

A3C

$$\theta_{k+1} = \theta_k + \alpha \left[\underbrace{\left(r(s_k, a_k, s_{k+1}) + \gamma \phi(s_{k+1})^T w_k - \phi(s_k)^T w_k \right) \psi_\theta(s_k, a_k)}_{v(x_k, \theta_k, w_k)\text{: unbiased estimator of } \nabla J(\theta)} + \underbrace{\lambda \psi_\theta(x^p)}_{\text{Unbiased estimator of } \nabla R(\theta)} \right]$$

$$\theta_{k+1} = \theta_k + \alpha \left[v(x_k, \theta_k, w_k) + \lambda \psi_\theta(x^p) \right]$$

A3C

- Critic step

TD(0) algorithm

$$w_{k+1} = w_k + \beta \hat{\delta}(x_k, w_k) \nabla \hat{V}_{w_k}(s_k)$$

A3C

- <u>TD(0) algorithm</u>

$$V(\boldsymbol{s}_t) \leftarrow V(\boldsymbol{s}_t) + \beta\left[r_{t+1} + \gamma V(\boldsymbol{s}_{t+1}) - V(\boldsymbol{s}_t)\right]$$

A3C

- TD(0) algorithm

$$V(\boldsymbol{s}_t) \leftarrow V(\boldsymbol{s}_t) + \beta \mathbb{E}_{s,a,s'} \left[\boldsymbol{r}(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{s}') + \gamma \boldsymbol{V}_{\pi_\theta}(\boldsymbol{s}') - \boldsymbol{V}_{\pi_\theta}(\boldsymbol{s}) \right]$$

A3C

- <u>Value Function Approximation</u>

$$V_{\pi_\theta}(s) \approx \hat{V}_w(s) = \phi(s)^T w$$

A3C

- TD error

$$\hat{\delta}(x_k, w_k) := r(s_k, a_k, s_{k+1}) + \left(\gamma\phi(s_{k+1}) - \phi(s_k)\right)^T w_k$$

A3C

Π_{R_w} is a projection operator that projects a vector to a l_2 norm ball with radius R_w.

It prevents the actor and critic updates from going too far in the wrong direction.

A3C

Critic step $$w_{k+1} = \Pi_{R_w}\left(w_k + \beta g(x_k, w_k)\right)$$

Actor step $$\theta_{k+1} = \theta_k + \alpha\left[v(x_k, \theta_k, w_k) + \lambda \psi_\theta(x^p)\right]$$

A3C

Critic step $$\omega_{k+1} = \Pi_{R_\omega}\left(\omega_k + \beta g(x_{(k)}, \omega_{k-\tau_k})\right)$$

Actor step $$\theta_{k+1} = \theta_k + \alpha\left(v(\hat{x}_{(k)}, \theta_{k-\tau_k}, \omega_{k-\tau_k}) + \lambda \psi_{\theta_{k-\tau_k}}(x^p_{(k)})\right)$$

- τ_k : the delay in the kth actor and critic updates

A3C

Algorithm 1 A3C: each worker's view.

Global initialize: Global counter $k=0$, initial θ_0, ω_0 in the shared memory.
Worker initialize: Counter $t=0$. Sample $s_0 \sim \eta$, $\hat{s}_0 \sim \eta$.
for $t = 0, 1, 2, \cdots$ **do**
    Read θ, ω in the shared memory.
    option 1 (i.i.d. sampling):
        $x_t = \big(s_t \sim \mu_{\pi_{\theta_t}}, a_t \sim \pi_{\theta_t}(\cdot|s_t), s'_t \sim \mathcal{P}(\cdot|s_t, a_t)\big)$.
        $\hat{x}_t = \big(\hat{s}_t \sim d_{\pi_{\theta_t}}, \hat{a}_t \sim \pi_{\theta_t}(\cdot|\hat{s}_t), \hat{s}'_t \sim \mathcal{P}(\cdot|\hat{s}_t, \hat{a}_t)\big)$.
    option 2 (Markovian sampling):
        $x_t = \big(s_t, a_t \sim \pi_\theta(\cdot|s_t), s_{t+1} \sim \mathcal{P}(\cdot|s_t, a_t)\big)$.
        $\hat{x}_t = \big(\hat{s}_t, \hat{a}_t \sim \pi_\theta(\cdot|\hat{s}_t), s'_{t+1} \sim \mathcal{P}(\cdot|\hat{s}_t, \hat{a}_t)\big)$.
        With probability γ: $\hat{s}_{t+1} = s'_{t+1}$; Otherwise: $\hat{s}_{t+1} \sim \eta$.
    Compute $g(x_t, \omega) = \hat{\delta}(x_t, \omega)\nabla_\omega \hat{V}_\omega(s_t)$.
    Compute $v(\hat{x}_t, \theta, \omega) = \hat{\delta}(\hat{x}_t, \omega)\psi_\theta(\hat{s}_t, \hat{a}_t)$.
    Compute $\psi_\theta(x_t^p)$ with $x_t^p = (s_t^p \sim \eta_p, a_t^p \sim \pi_p(\cdot|s_t^p))$.
    In the shared memory, perform update (13).
end for

$$\omega_{k+1} = \Pi_{R_\omega}\left(\omega_k + \beta g(x_{(k)}, \omega_{k-\tau_k})\right) \tag{13a}$$

$$\theta_{k+1} = \theta_k + \alpha\big(v(\hat{x}_{(k)}, \theta_{k-\tau_k}, \omega_{k-\tau_k}) + \lambda\psi_{\theta_{k-\tau_k}}(x^p_{(k)})\big) \tag{13b}$$

A3C

option 2 (Markovian sampling):

$$x_t = \big(s_t, a_t \sim \pi_\theta(\cdot|s_t), s_{t+1} \sim \mathcal{P}(\cdot|s_t, a_t)\big). \quad \text{Critic}$$

$$\hat{x}_t = \big(\hat{s}_t, \hat{a}_t \sim \pi_\theta(\cdot|\hat{s}_t), s'_{t+1} \sim \mathcal{P}(\cdot|\hat{s}_t, \hat{a}_t)\big). \quad \text{Actor}$$

With probability γ: $\hat{s}_{t+1} = s'_{t+1}$; Otherwise: $\hat{s}_{t+1} \sim \eta$

- In Markovian sampling case, we maintain separate Markov chains for actor and critic.
- For **Critic**, we generate samples following the original transition kernel $\boldsymbol{P}$.
- For **Actor**, we generate samples following a transition kernel $\widehat{\boldsymbol{P}} = \boldsymbol{\gamma P} + (\mathbf{1} - \boldsymbol{\gamma})\boldsymbol{\eta}$.

If the actor's chain evolves under P like critic, asymptotically the initial distribution η is forgotten, which will introduce asymptotic error.

A3C

option 1 (i.i.d. sampling):

$$x_t = \left(s_t \sim \mu_{\pi_{\theta_t}}, a_t \sim \pi_{\theta_t}(\cdot|s_t), s'_t \sim \mathcal{P}(\cdot|s_t, a_t)\right). \quad \text{Critic}$$

$$\hat{x}_t = \left(\hat{s}_t \sim d_{\pi_{\theta_t}}, \hat{a}_t \sim \pi_{\theta_t}(\cdot|\hat{s}_t), \hat{s}'_t \sim \mathcal{P}(\cdot|\hat{s}_t, \hat{a}_t)\right). \quad \text{Actor}$$

- μ_{π_θ} is the stationary distribution of the Markov chain with transition distribution P and π_θ.
- d_{π_θ} is the discounted state visitation measure induced by policy π_θ.

A3C

- Advantage

Training time roughly reduces linearly as the number of workers increases.

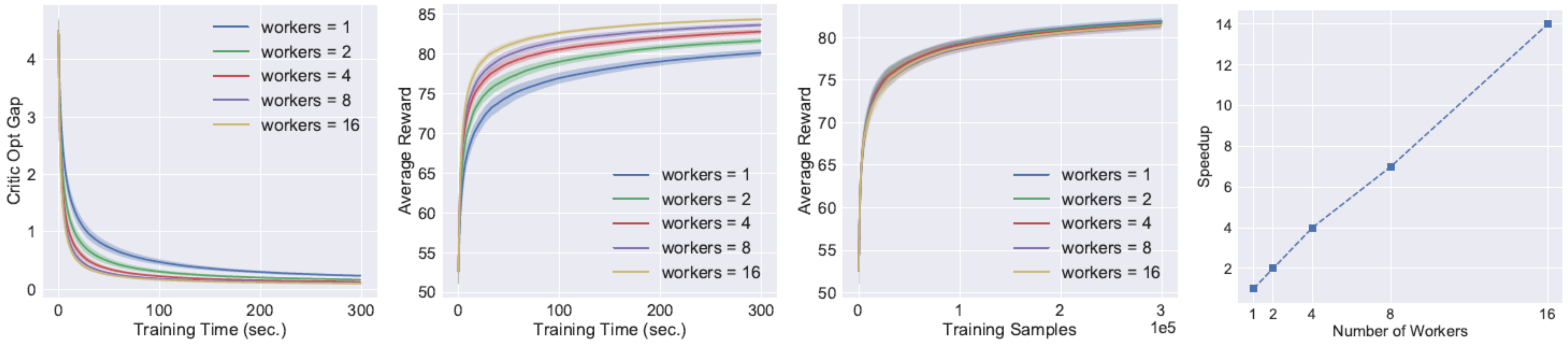

Figure 3: Convergence results of A3C with i.i.d. sampling in synthetic environment.

A3C

- Advantage

It operates in both discrete and continuous action spaces.

A3C

- Disadvantage

The delayed parameters introduce extra error.

A3C

<u>Critic step</u> $$w_{k+1} = \Pi_{R_w}\left(w_k + \beta g(x_k, w_k)\right)$$

<u>Actor step</u> $$\theta_{k+1} = \theta_k + \alpha\left[v(x_k, \theta_k, w_k) + \lambda\psi_\theta(x^p)\right]$$

- **<u>error</u>**

$$g(x, \omega_{k-\tau_k}) - g(x, \omega_k)$$
$$v(x, \theta_{k-\tau_k}, \omega_{k-\tau_k}) - v(x, \theta_k, \omega_k)$$

<u>Critic step</u> $$\omega_{k+1} = \Pi_{R_\omega}\left(\omega_k + \beta g(x_{(k)}, \omega_{k-\tau_k})\right)$$

<u>Actor step</u> $$\theta_{k+1} = \theta_k + \alpha\left(v(\hat{x}_{(k)}, \theta_{k-\tau_k}, \omega_{k-\tau_k}) + \lambda\psi_{\theta_{k-\tau_k}}(x^p_{(k)})\right)$$

A3C

<u>Critic step</u> $$w_{k+1} = \Pi_{R_w}\left(w_k + \beta g(x_k, w_k)\right)$$

<u>Actor step</u> $$\theta_{k+1} = \theta_k + \alpha\left[v(x_k, \theta_k, w_k) + \lambda\psi_\theta(x^p)\right]$$

- **<u>error</u>**

  $g(x, \omega_{k-\tau_k}) - g(x, \omega_k)$

  $v(x, \theta_{k-\tau_k}, \omega_{k-\tau_k}) - v(x, \theta_k, \omega_k)$

<u>Critic step</u> $$\omega_{k+1} = \Pi_{R_\omega}\left(\omega_k + \beta g(x_{(k)}, \omega_{k-\tau_k})\right)$$

<u>Actor step</u> $$\theta_{k+1} = \theta_k + \alpha\left(v(\hat{x}_{(k)}, \theta_{k-\tau_k}, \omega_{k-\tau_k}) + \lambda\psi_{\theta_{k-\tau_k}}(x^p_{(k)})\right)$$

Thank you for listening