PPO for Bipedal Walker

Learning to Walk

Rushil Gupta

30th April 2025

Table of Contents

The Bipedal Walker Problem

- **Observation (state)** $s \in \mathbb{R}^{24}$: joint angles, velocities, LIDAR-like terrain scans, hull position/velocity.
- **Action** $a \in [-1, 1]^4$: continuous torques for two hips and two knees.
- **Reward**: distance progressed $+$ stability bonus $-$ joint power - penalties; episode terminates upon fall or timeout.

Two Difficulty Modes

Normal

Flat or mildly irregular terrain.

Hardcore

Random ladders, stumps, gaps and slippery surfaces — significantly sparser rewards and higher variance.

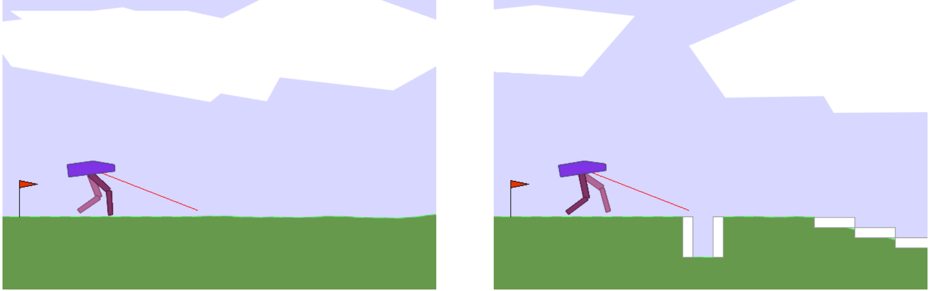

Figure: Environment visuals.

Table of Contents

Implementation Overview

- We will use PPO to learn the policy
- Our implementation consists of two key networks:
  - **Value Network**: Estimates state values $V(s)$ to compute advantages
    - Implemented as an MLP with ReLU activations that outputs a scalar
    - Used for critic updates and advantage estimation
  - **Gaussian Policy Network**: Produces continuous actions with exploration
    - Outputs mean $\mu(s)$ and state-dependent standard deviation $\sigma(s)$
    - Actions sampled from $\mathcal{N}(\mu(s), \sigma(s)^2)$ and squashed to $[-1, 1]$ with $\tanh$

PPO Clipped Objective

PPO maximises the surrogate objective function:

$$\mathcal{L}^{\text{clip}}(\theta) = \mathbb{E}_t\Big[\min\big(r_t(\theta)\,\hat{A}_t,\ \text{clip}\big(r_t(\theta), 1-\varepsilon, 1+\varepsilon\big)\,\hat{A}_t\big)\Big],$$

where:

- $r_t(\theta) = \dfrac{\pi_\theta(\boldsymbol{a}_t|\boldsymbol{s}_t)}{\pi_{\theta_{\text{old}}}(\boldsymbol{a}_t|\boldsymbol{s}_t)}$ is the probability ratio between new and old policies
- $\hat{A}_t$ is the advantage estimate, computed using GAE:

$$\delta_t = r_t + \gamma V_{\phi_{\text{old}}}(\boldsymbol{s}_{t+1}) - V_{\phi_{\text{old}}}(\boldsymbol{s}_t),$$

$$\hat{A}_t = \sum_{l=0}^{T-t-1} (\gamma\lambda)^l\,\delta_{t+l}.$$

PPO Complete Loss Function

The complete loss combines three components:

$$\mathcal{J}(\theta, \phi) = \mathcal{L}^{\text{clip}} - \beta\, \mathbb{E}[\mathsf{H}[\pi_\theta]] + \frac{c_v}{2}\, \mathbb{E}_t[(V_\phi - \hat{R}_t)^2].$$

- $\mathcal{L}^{\text{clip}}$: Policy loss with clipping to constrain policy updates
- $\mathsf{H}[\pi_\theta]$: Entropy term that encourages exploration
  - For Gaussian policy: $\mathsf{H}[\pi_\theta] = \frac{1}{2}\log(2\pi e \sigma^2)$
  - Higher entropy = more exploration (wider action distribution)
  - β coefficient is annealed over time to reduce exploration gradually
- Value loss: $\frac{c_v}{2}\, \mathbb{E}_t[(V_\phi - \hat{R}_t)^2]$ trains the critic to accurately predict returns

Training Loop (GAE + Normalisation)

- **Collect** N steps (or full episodes) with current policy.
- **Update running mean/std** μ, σ and normalise states: $\tilde{\boldsymbol{s}} = (\boldsymbol{s} - \mu)/\sigma$.
- **Compute** $\hat{A}_t$ and *standardise* them: $\hat{A} \leftarrow (\hat{A} - \bar{A})/\mathrm{Std}(\hat{A})$.
- **Optimise** K epochs over shuffled minibatches with the clipped loss.
- **Anneal** entropy coefficient β and cosine-decay learning rate.

Table of Contents

Demonstration - Normal Mode

Click here.

Table of Contents

Learning Challenge

- Sparse rewards and random obstacles $\Rightarrow$ frequent early terminations.
- The agent is not able to learn how to walk effectively, since it **falls**, or **stumbles** too often.

Demonstration - Hardcore (failed)

Click here.

Table of Contents

Warm-Start Strategy

1. **Pre-train** policy on normal terrain until convergence.
2. **Initialise** hardcore agent with $\theta_0 = \theta_{\text{normal}}$.
3. Continue PPO fine-tuning (smaller LR, larger ε).

Demonstration - Hardcore (after Transfer)

Click here.