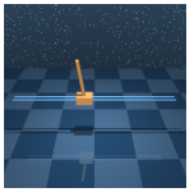
Individual Assignment – learn to balance the cart-pole using an actor-critic continuous control method of your choice (such as DDPG, TD3, SAC, PPO).

Due date: 11/30/2025

Report:

- Approximately 2 pages in IEEE publication format.
- Include an accessible link (GitHub/Colab link) to your implementation code (failed to do so will result in major deduction).
- Discuss your method and its implementation.
- Provide your hyperparameter choices.
- Train your control policy using 3 seeds (0, 1, 2).
- Evaluate your control policy using 1 seed (10).
- Plot your learning curve (reward vs learning updates/epochs) using (mean ± std) to show both training and evaluation results.



Cart-pole (4, 1, 5): Swing up and balance an unactuated pole by applying forces to a cart at its base. The physical model conforms to (Barto et al., 1983). Four benchmarking tasks: in `swingup` and `swingup_sparse` the pole starts pointing down while in `balance` and `balance_sparse` the pole starts near the upright.

Remarks

- You can discuss with your team. But you need to create your own product.
- This is a small problem that can be done using a typical CPU.
- You may take reference of RL libraries such as Stable-Baselines3, CleanRL.
- Grades will emphasize correctness, reproducibility, and clarity of presentation more than raw performance of high reward value.
- Details about the task and the Deepmind Control Suite (DMC) can be found below.

DeepMind Control Suite

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez,
Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel,
Andrew Lefrancq, Timothy Lillicrap, Martin Riedmiller

January 3, 2018

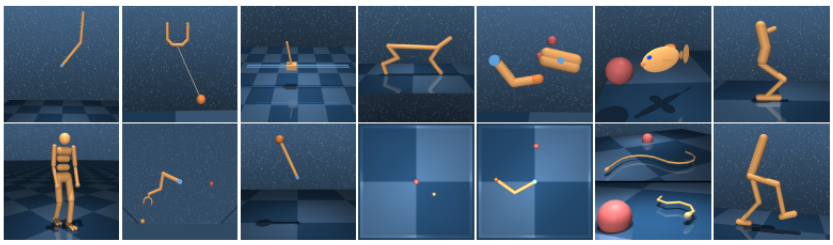


Figure 1: Benchmarking domains. *Top*: Acrobot, Ball-in-cup, Cart-pole, Cheetah, Finger, Fish, Hopper. *Bottom*: Humanoid, Manipulator, Pendulum, Point-mass, Reacher, Swimmer (6 and 15 links), Walker.