Selected Topics in Reinforcement Learning Final Project Report

314553017 蕭希敏

1. Methodology Introduction

Circle Map (PPO)

使用 `final_project_env/scripts/train_circle_c2_env.py` 與 `final_project_env/racecar_gym/env_circle_gray_resize.py`：

- 採用 C2-style 設定，目標是快速穩定收斂。
- 使用較小的 frame stack (4) 以提高訓練效率。
- Reward shaping 主要抑制撞牆與抖動，並加入速度獎勵。

Austria Map (PPO)

使用 `final_project_env/scripts/train_austria_env.py` 與 `final_project_env/racecar_gym/env_austria_gray_resize.py`：

- 地圖彎道多、難度高，採用較長的 frame stack (8) 來保留動態資訊。
- Reward shaping 著重 progress 放大與 checkpoint，並加入動作平滑與撞牆懲罰。

TD3

TD3 作為連續控制 baseline，選用的原因是本作業屬於連續動作控制（motor/steering），TD3 是代表的 off-policy 演算法，可檢驗在相同觀測與 reward 設計下，off-policy 是否能帶來更佳樣本效率或更穩定的控制表現。

---

2. Experiment Design and Implementation

2.1 Observation Preprocessing

- Gray scale
- Resize to 84 x 84
- Frame stack (circle 使用 4; austria 使用 8)

2.2 Circle 訓練流程 (PPO)

使用 `final_project_env/scripts/train_circle_c2_env.py`：

- Scenario: `circle_cw_competition_collisionStop`
- Init mode: `random` (env reset 時固定為 random)
- Observation: gray + resize 84, frame stack 4
- `SubprocVecEnv` with 8 envs
- PPO hyperparameters:
  - `learning_rate=3e-4`

    - `n_steps=1024`, `batch_size=256`, `n_epochs=4`
    - `gamma=0.99`, `gae_lambda=0.95`, `clip_range=0.2`, `vf_coef=0.5`
    - `total_timesteps=3e5`
- Evaluation:
    - `eval_freq=20_000`, `n_eval_episodes=5`
- Logs and checkpoints:
    - `runs/circle_C2_env`
    - `checkpoints/circle_C2_env.zip`
    - `checkpoints/best_circle_C2_env`

2.3 Austria 訓練流程 (PPO)

使用 `final_project_env/scripts/train_austria_env.py`：

- Scenario: `austria_competition_collisionStop`
- Reset when collision: `False`
- Observation: gray + resize 84, frame stack 8
- `SubprocVecEnv` with 40 envs
- PPO hyperparameters:
    - `learning_rate=3e-4`, `use_sde=True`
    - `n_steps=1024`, `batch_size=64`, `n_epochs=10`
    - `clip_range=0.2`
    - `total_timesteps=5e6`
- Evaluation:
    - `eval_freq=20_000`, `n_eval_episodes=5`
- Logs and checkpoints:
    - `runs/austria_env`
    - `checkpoints/austria_env/final_model.zip`
    - `checkpoints/best_austria_env`

2.4 Reward Shaping (Circle & Austria)

Circle (`env_circle_gray_resize.py`)：

- 基礎 reward 以 progress 為主，但容易抖動與撞牆。
- 加入 collision penalty，降低撞牆頻率。
- 加入 action smoothness 與 steering penalty，改善左右擺動。
- 加入速度獎勵提升平均速度，但需用 reward clip 限制過激行為。

Pseudo code:

```
reward = base_progress_reward
if wall_collision: reward -= collision_penalty
reward += speed_reward_scale * ||velocity_xy||
reward -= steering_penalty_scale * abs(steering)
if last_action exists:
    reward -= action_smoothness_penalty * ||action - last_action||
reward = clip(reward, -reward_clip, reward_clip)
```

Austria (`env_austria_gray_resize.py`) ：

- 初版 progress 訊號太弱，車子常停滯或失控。
- 增加 checkpoint reward，提供短期目標引導。
- 加入 motor reward，鼓勵持續前進。
- 加入 action smoothness 與 steering penalty，讓轉向更穩定。
- 撞牆給大額負獎勵並終止，避免反覆碰撞。

Pseudo code:

```
reward = 0
if checkpoint changed: reward += checkpoint_reward
reward += motor_reward * motor_action
reward -= smoothness_penalty * (|motor - prev_motor| + |steer -
prev_steer|)
reward -= steering_penalty_scale * abs(steer)
if progress increased:
    reward += progress_scale * delta_progress
else if progress stalled:
    reward -= stall_penalty
if wall_collision:
    reward = collision_penalty
    terminate episode
```

2.5 Neural Network Architecture

採用 Stable-Baselines3 的 `CnnPolicy` (NatureCNN) 作為影像特徵提取器：

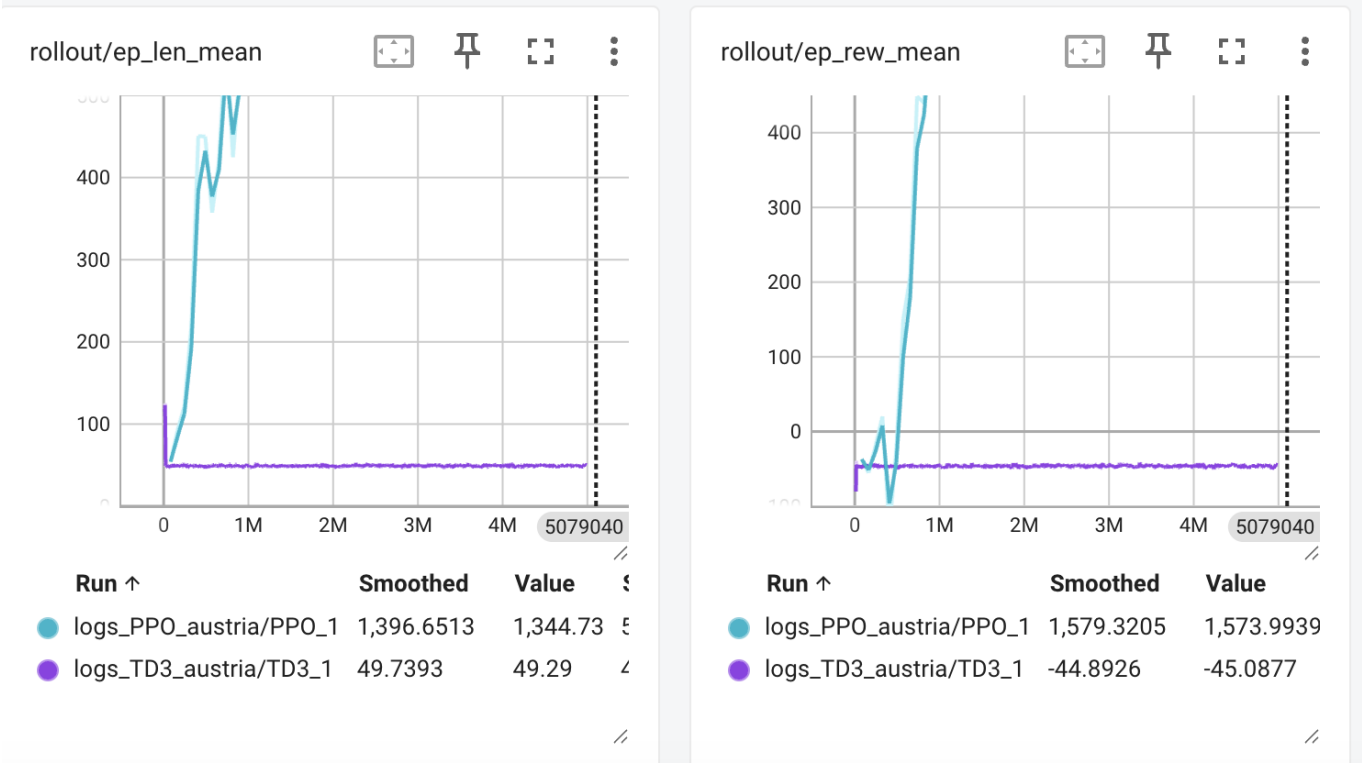
- Conv2d(32, kernel=8, stride=4) + ReLU
- Conv2d(64, kernel=4, stride=2) + ReLU
- Conv2d(64, kernel=3, stride=1) + ReLU
- Flatten + MLP heads (policy / value)

2.5 Packages and Tools

- `stable-baselines3`
- `gymnasium`
- `racecar_gym` (course environment)
- `opencv-python`
- `numpy`

---

3. Method Comparison and Evaluation



- **PPO (Austria, custom)**: 成功率約60%，受 reward shaping 影響較大。
- Austria 在 reward shaping 後能通過髮夾彎，速度控制更平順。

```
Episode 1: score=3.5513 reward=3.23
Episode 2: score=1.2838 reward=4.03
Episode 3: score=0.6576 reward=0.48
Episode 4: score=3.2996 reward=3.25
Episode 5: score=1.6526 reward=0.97
==== Summary ====
Mean score: 2.0890 | Min score: 0.6576
```

- **TD3 (Austria, custom)**: 早期探索容易撞牆，`ep_len_mean` 近乎不成長，整體表現明顯低於 PPO。 因 early-stage action noise 導致大量 collision，episode 太短，學不到有效策略。

```
Episode 1: score=0.2687 reward=0.03
Episode 2: score=0.5752 reward=0.01
Episode 3: score=0.0441 reward=0.01
Episode 4: score=0.3417 reward=0.00
Episode 5: score=0.5972 reward=0.01
==== Summary ====
Mean score: 0.3654 | Min score: 0.0441
```

Observation

- Reward scaling 對 austria 成功率影響極大。
- 高 n_env 平行化能加速收斂，但也讓更新頻率降低，需要配合較小 n_steps。
- 圖像輸入下，PPO 對超參數敏感度較低，較適合作為 baseline。

- TD3因 early-stage action noise 導致大量 collision，episode 太短，學不到有效策略。
- PPO 在影像輸入下較穩定，搭配 reward shaping 與 frame stack 能形成有效策略；TD3 前期探索噪聲造成頻繁撞牆，replay buffer 被負樣本主導，導致難以學到可行控制。

---

4. Challenges and Learning Points

1. **Reward Design**: 預設 reward 太小，critic 幾乎無法學習；透過 progress 放大與 checkpoint reward 解決。
2. **Collision Handling**: collisionStop 對模型影響很大，若設錯容易導致策略過度保守。
3. **Stability**: TD3 在影像輸入下不易收斂，需要更精細的探索與 replay buffer 設計。

---

5. Future Work

1. **更穩健的 Reward Shaping**: 引入 racing line 或 curvature-based reward。
2. **多場景訓練**: 結合 circle/austria 混合訓練，提升泛化。
3. **資料增強**: 將 DrQ-style augmentation 併入 TD3 以提升穩定性。
