서강대학교 SOGANG UNIVERSITY

PPO-BR: Dual-Signal Entropy-Reward Adaptation for TRPO 재현 실험

25조

120240571 김도영

120240572 김우석

Github: https://github.com/woosook0127/RL_Framework

Index

- 프로젝트 주제 및 목표
- Baseline
- 재현 설계
- 재현 실험 및 결과
- 결론
- 구성원 기여 내용

프로젝트 주제 및 목표

- 주제
  - PPO-BR 논문에서 제시된 실험 결과 재현

- 목표
  - 알고리즘 구현 및 성능 우위 검증
  - 통계적 신뢰성 및 강건성 평가

Baseline

- 논문
  - PPO-BR: Dual-Signal Entropy-Reward Adaptation for Trust Region Policy Optimization
  - 2025 IEEE TNNLS Under Review

- 내용
  - PPO의 고정된 clipping 범위가 갖는 한계(초기 탐색 저하 및 후기 수렴 불안정) 해결하기 위해 제안된 알고리즘
  - 탐색(Entropy)과 수렴(Reward) 두 가지 신호를 사용하여 신뢰 영역(Trust Region)을 동적으로 조절
    - 탐색 단계 (High Entropy)
      - 정책의 불확실성이 높을 때는 clipping 범위(ε)를 확장하여 적극적인 탐색 유도
    - 수렴 단계 (Stable Reward)
      - 보상이 안정화(Plateau)되면 clipping 범위(ε)를 축소하여 학습의 안정성 확보

Baseline

- Experiment
  - OpenAI Gym / MuJoCo 환경에서 아래 6가지 task 수행
    - CartPole, LunarLander, Hopper, HalfCheetah, Walker2D, Humanoid
  - PPO와 성능 비교

- Results

Environment	PPO (Return)	PPO-BR (Return)	Improvement (%)	Reward Variance (PPO)	Reward Variance (PPO-BR)	Reduction	Convergence Steps (PPO)	Convergence Steps (PPO-BR)
CartPole	195	200	2.60%	35	30	14.3% ↓	150	130
LunarLander	180	230 ± 5 (mean ± std, n=5 seeds)	27.80%	120	60	50% ↓	300	250
Hopper	2200	2600	18.20%	180	100	44.4% ↓	600	500
HalfCheetah	2500	3000	20.00%	250	120	52% ↓	800	620
Walker2D	2100	2450	16.70%	230	110	52.2% ↓	700	580
Humanoid	1600 ± 80	2100 ± 30 (mean ± std, n=5 seeds, $p<0.01$)	31.30%	300	150	50% ↓	1000	700

재현 설계

- Experiment Setup in PPO-BR
  - Networks
    - 2-layer MLP (Actor/Critic), Hidden size 64, ReLU Activation
    - Adam Optimizer
  - Hyperparameter
    - Learning Rate: 3e-4
    - Batch Size: 64
    - PPO Clip Threshold ϵ_0 : 0.2
    - Discount Factor γ : 0.99
    - GAE λ : 0.95
  - 5 random seed 사용

재현 설계

- Experiment Setup in PPO-BR
  - Hyperparameter of PPO-BR
    - Entropy Weight (λ_1): 0.5
    - Reward Weight (λ_2): 0.3
    - Reward Smoothing Window (k): 10
  - Adaptive Trust Region 계산
    - $\epsilon_t = \epsilon_0 \cdot \left[1 + \lambda_1 \cdot \tanh\left(\phi(H_t)\right) - \lambda_2 \cdot \tanh\left(\psi(\Delta R_t)\right)\right]$

Algorithm 1 PPO-BR: Adaptive Clipping for Trust Region Policy Optimization

PPO-BR Key Modification
1. Dynamic ϵ_t via entropy (λ_1) and reward (λ_2)
2. Bounded by $\epsilon_{min}/\epsilon_{max}$ for safety
3. Seamless drop-in replacement for PPO

Input: Initial policy $\pi\theta$, value function $V\varphi$, base threshold ε_0, hyperparameters α, β, λ_1, λ_2
Initialize: policy parameters θ, reward baseline $\bar{R}$

for each iteration do
  Collect trajectories using current policy $\pi\theta$
  Compute policy entropy H_t and smoothed reward delta ΔR_t
  Normalize $H_t \rightarrow \varphi(H_t)$, $\Delta R_t \rightarrow \psi(\Delta R_t)$
  Compute adaptive clipping threshold:
    $\varepsilon_t \leftarrow \varepsilon_o \times [1 + \lambda_1 \cdot \tanh(\varphi(H_t)) - \lambda_2 \cdot \tanh(\psi(\Delta R_t))]$
    $\varepsilon_t \leftarrow \text{clip}(\varepsilon_t, \varepsilon_{min}, \varepsilon_{max})$
  Compute surrogate loss with ε_t clipping:
    L_CLIP $\leftarrow$ E[min($r_t \hat{A}_t$, clip(r_t, $1-\varepsilon_t$, $1+\varepsilon_t$) $\hat{A}_t$)]
  Update policy parameters θ via stochastic gradient ascent on L_CLIP
  Update value function parameters φ via MSE loss
end for

재현 설계

- 재현 불확실성 및 구현 전략
  - ΔR_t 계산
    - 논문에서는 단순히 '보상 변화' 라고 명시
    - ∴ 최근 보상을 절반으로 나누어 평균 차이를 상대 변화율로 계산하도록 설정
  - $\psi(\Delta R_t)$ 정규화 함수
    - "normalization function mapping to [0,1]" 라고 명시. ∴ 1.0 - tanh(|ΔR_t|) 사용하여 구현
  - Window size
    - Reward window size에 대한 명시적 언급 없음. ∴ 기본값 20으로 설정
  - H_t normalization 를 위한 H_{max}
    - 명시적 정의 없음. ∴ Discrete와 Continuous에 대해 합리적인 방식으로 계산

재현 설계

- 재현 불확실성 및 구현 전략
  - Metric 계산의 한계점
    - 어느 시점의 **Return**을 기록한 것인가?
    - Reward Variance 측정 범위가 어디까지인가?
    - Convergence Step 계산 시 수렴 기준은 무엇인가?
  - Environments 버전 명시하지 않음
    - Random Seed 고정 방식 명시하지 않음
  - 가장 중요한 metric 계산 방식에 대해 논문에서 명시하지 않음. ∴ 개선 정도를 비율로 비교

재현 실험 및 결과

- CartPole
  - 움직이는 카트 위에 막대(Pole)를 세우고 넘어지지 않도록 균형을 잡는 task
  - 막대가 넘어지지 않고 버틴 timestep마다 보상 수령

  - 논문 결과

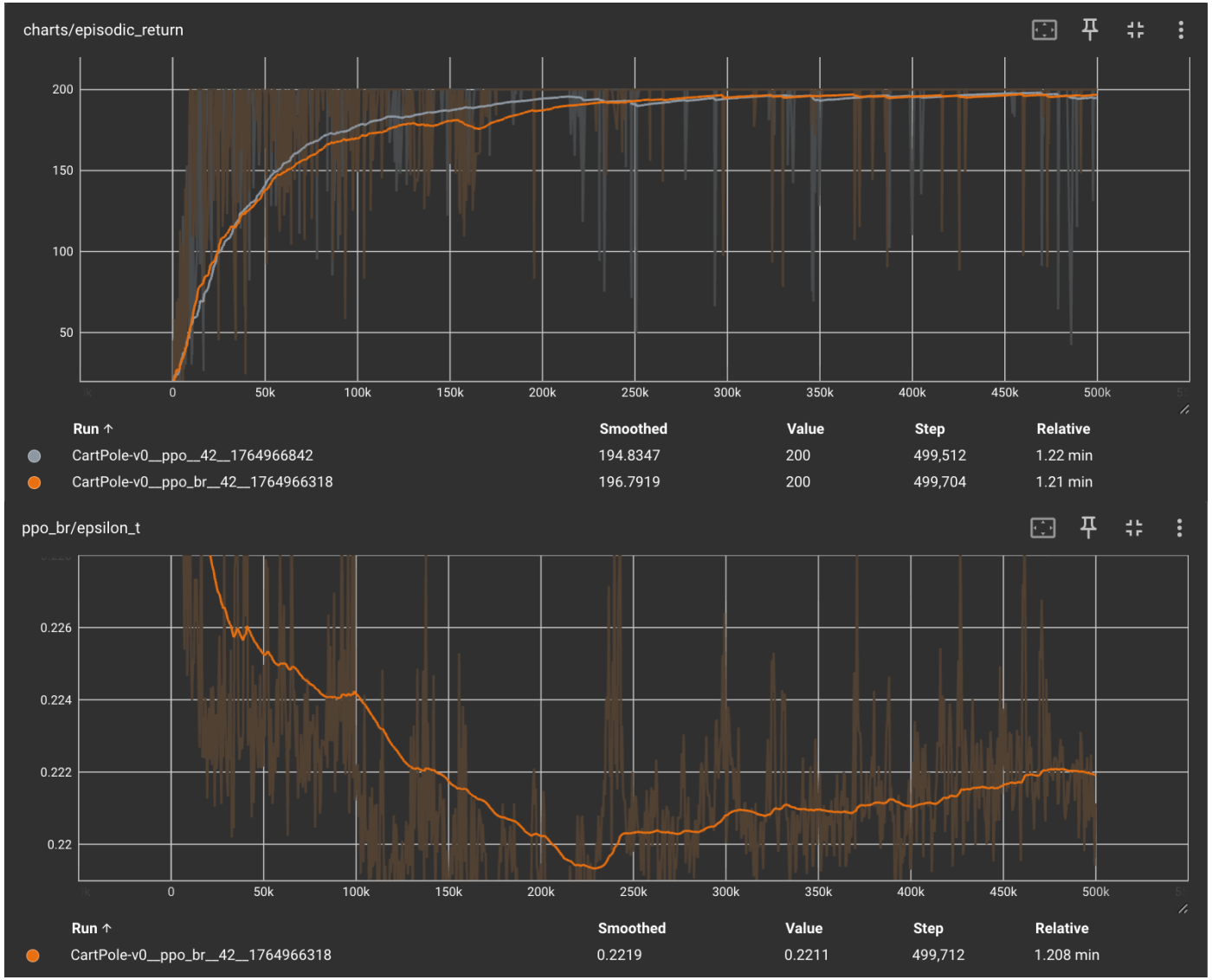
	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	195	-	35	-	150	-
PPO-BR	200	2.6%	30	14.3%	130	13.3%

  - 재현 결과

	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	186.35	-	38.99	-	313	-
PPO-BR	186.30	0.0%	35.13	9.9%	324	-3.5%

재현 실험 및 결과

- CartPole
  - 실험 결과
    - Return
    - ϵ_t



재현 실험 및 결과

- CartPole
  - 실험 결과 해석
    - Quantitative Result
      - CartPole-v0 환경은 난이도가 낮아 Ceiling Effect가 발생, 최종 평균 보상의 차이는 미미(-0.03%)
      - 그러나 Reward Variance는 PPO 대비 9.9% 감소(38.99 -> 35.13)하여, 논문이 주장하는 학습 안정성 개선 효과를 확인
    - Qualitative Analysis
      - 학습 초기 높은 엔트로피 구간에서는 ϵ_t가 확장되어 탐색을 촉진, 학습이 진행됨에 따라 점진적으로 축소되는 경향
      - ϵ_t graph: 수렴 이후(230k steps~)에는 과도한 제약을 피하기 위해 임계값이 소폭 재조정되는 현상 확인
        - Baseline(회색)에서 관찰되는 Fluctuation을 줄이고, 일관된 정책 유지를 가능하게 한 요인으로 해석

재현 실험 및 결과

- LunarLander
  - 착륙선(Lander)을 깃발 사이(착륙 패드)에 부드럽게 착륙시키는 task
  - 거리, 속도, 자세, 접촉, 성공 여부, 연료 비용에 따라 보상 결정

  - 논문 결과

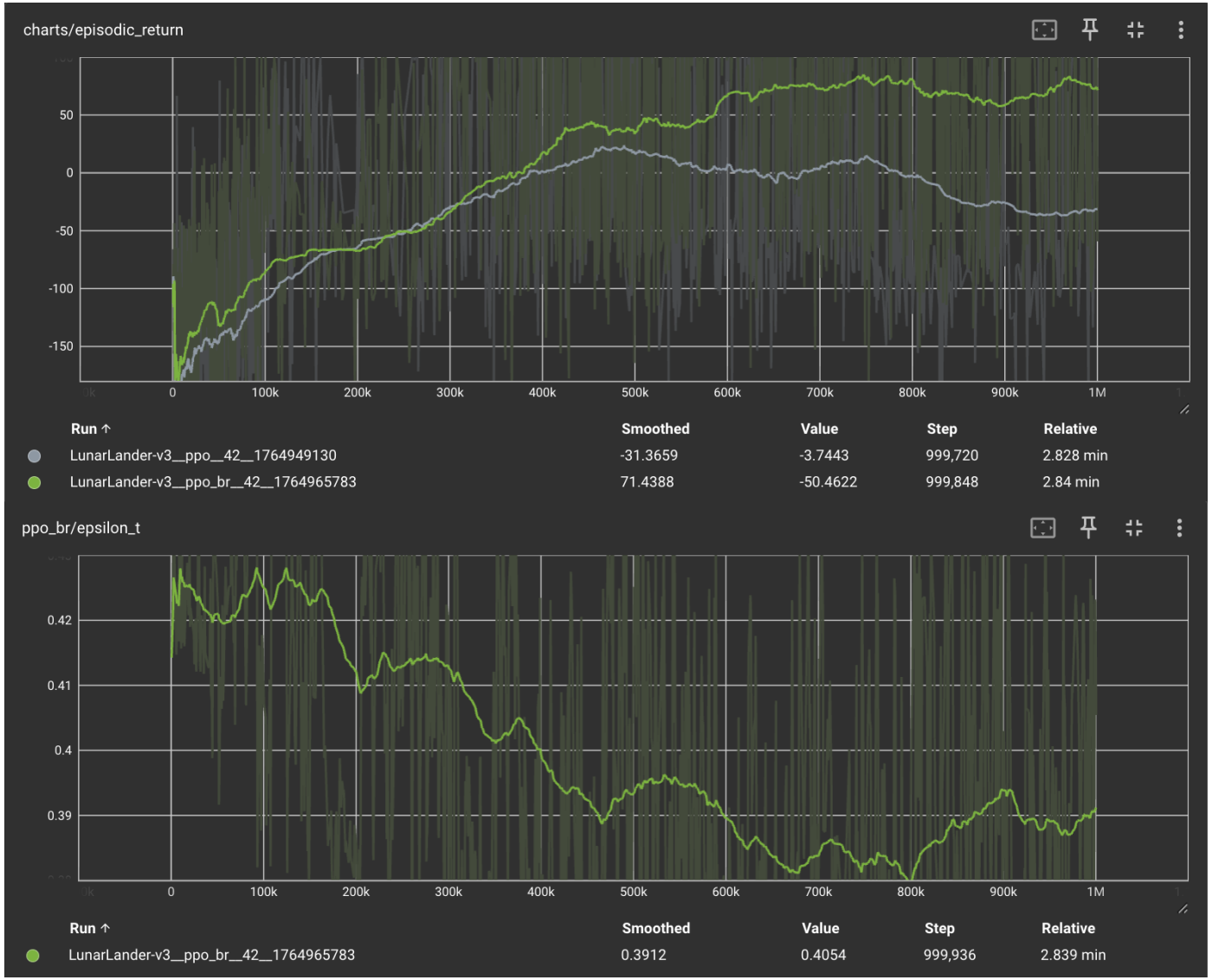
	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	180	-	120	-	300	-
PPO-BR	230	27.8%	60	50%	250	16.7%

  - 재현 결과

	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	-30.03	-	134.8	-	1050	-
PPO-BR	134.4	548%	134.4	0.2%	930	11.4%

재현 실험 및 결과

- LunarLander
  - 실험 결과
    - Return
    - ϵ_t



재현 실험 및 결과

- LunarLander
  - 실험 결과 해석
    - Quantitative Result
      - PPO-BR이 Sparse Reward나 불안정한 초기 조건에서도 학습 궤도를 이탈하지 않고 최적해를 찾아내는 능력이 탁월함을 증명
      - 수렴 속도 또한 PPO(1050) 대비 PPO-BR(930)이 약 11.4% 더 빠르게 안정권에 진입
    - Qualitative Analysis
      - PPO의 경우, 500k steps 부근까지는 상승하다가, 이후 성능이 0점 이하로 급락
      - 반면 PPO-BR은 500k steps 이후에도 지속적으로 점수를 쌓아 올리며, PPO가 붕괴되는 시점에서도 안정적인 우상향 곡선 유지
      - ϵ_t graph: 초기 0.42 이상에서 시작하여 강력한 탐색으로 착륙 방법을 빠르게 찾아냄
      - 이후 보상이 상승함에 따라 ϵ_t를 0.38~0.39 수준으로 낮추며, PPO가 겪었던 "정책 붕괴"를 사전에 차단

재현 실험 및 결과

- Hopper
  - 외발 로봇(Monopod)이 넘어지지 않고 앞으로 뛰어가는 task
  - 전진 속도, 제어 비용, 생존 여부에 따라 보상 결정

  - 논문 결과

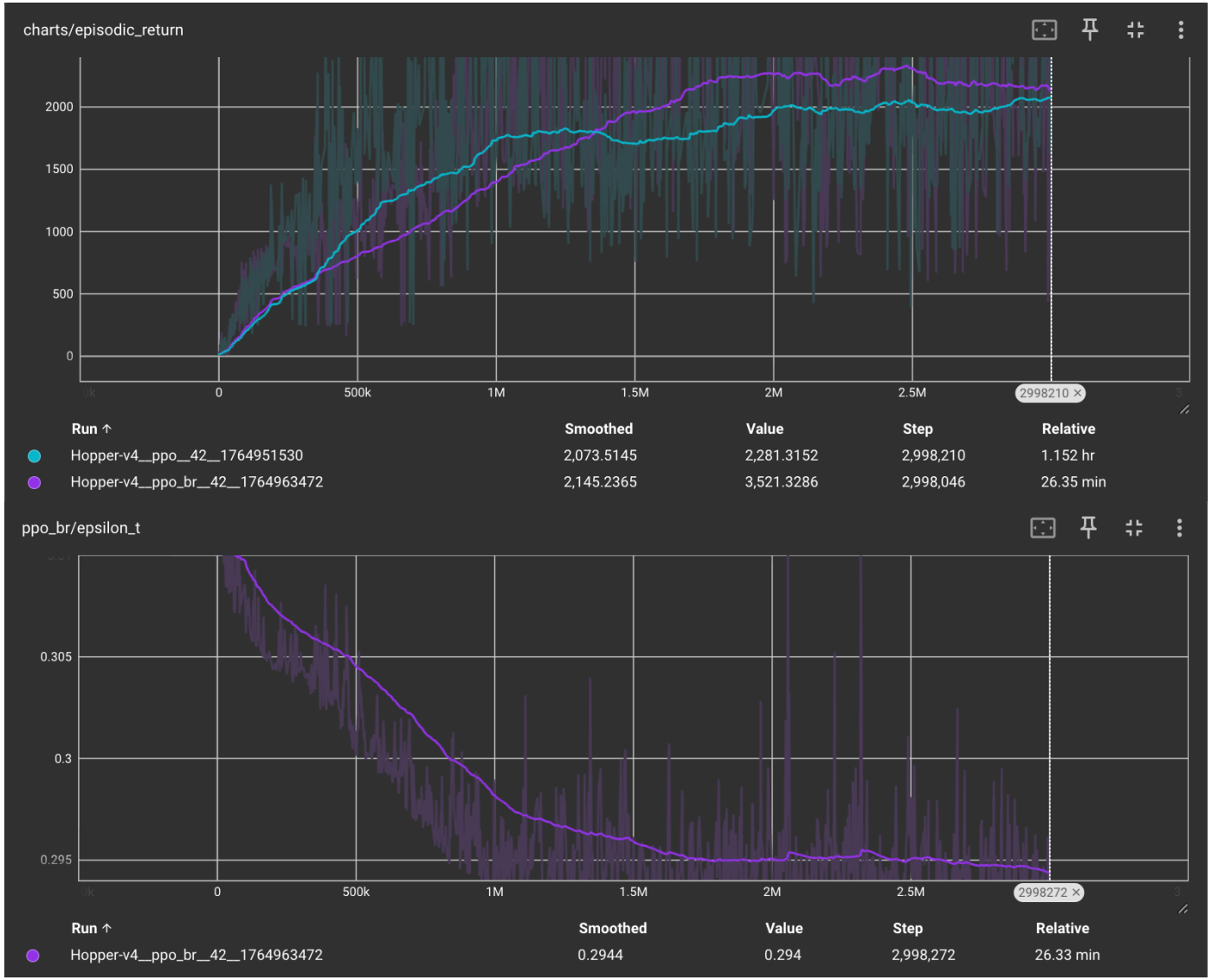
	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	2200	-	180	-	600	-
PPO-BR	2600	18.2%	100	44.4%	500	16.7%

  - 재현 결과

	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	1437.1	-	969.5	-	1000	-
PPO-BR	1433.6	-0.3%	1041.7	-7.4%	800	20%

재현 실험 및 결과

- Hopper
  - 실험 결과
    - Return
    - ϵ_t



재현 실험 및 결과

- Hopper
  - 실험 결과 해석
    - Quantitative Result
      - 전체적인 평균 Return은 PPO(1437.09)와 PPO-BR(1433.55)이 거의 대등한 수준
      - Reward Variance는 PPO 대비 PPO-BR이 다소 높게 나타났으며, PPO-BR이 수렴 기준 reward에 더 먼저 도달하는 결과를 보임
    - Qualitative Analysis
      - PPO-BR(보라색)이 1.5M 스텝 이후에도 지속적으로 상승하여 PPO(청록색)보다 높은 최종 값에 도달하는 경향성 확인
      - ϵ_t graph: 학습 초기 0.31 부근에서 시작하여 0.295까지 점진적으로 감소하는 우하향 곡선
        - PPO-BR이 학습 진행도에 따라 성공적으로 Trust Region을 동적으로 조절하고 있음

재현 실험 및 결과

- HalfCheetah
  - 2차원 평면상의 치타 로봇(다리 2개)이 가능한 한 빨리 달리는 task
  - 전진 속도, 제어 비용에 따라 보상 결정

  - 논문 결과

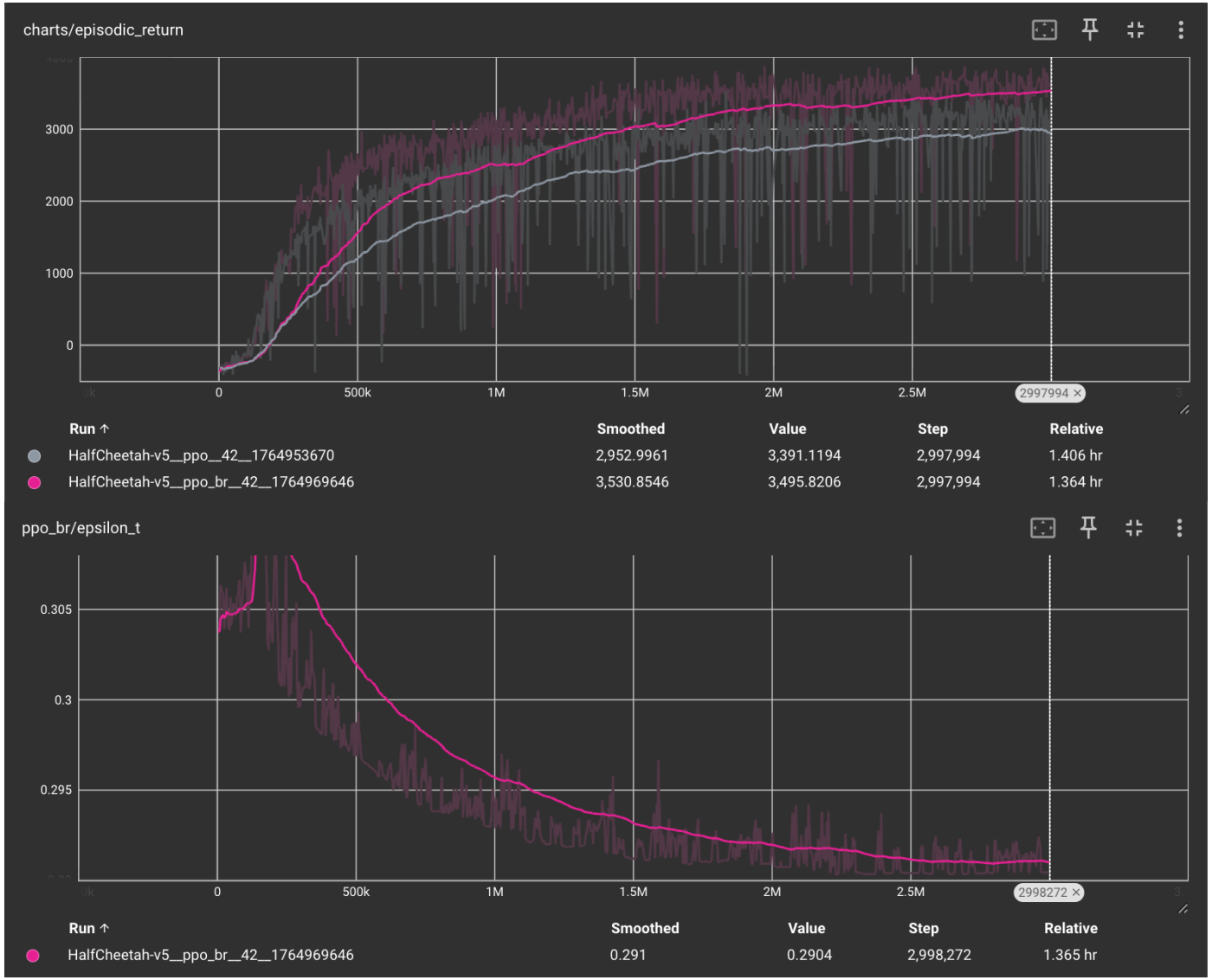
	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	2500	-	250	-	800	-
PPO-BR	3000	20%	120	52%	620	22.5%

  - 재현 결과

	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	2464.1	-	816.3	-	1400	-
PPO-BR	2982.8	21%	844.4	-3.4%	720	48.6%

재현 실험 및 결과

- HalfCheetah
  - 실험 결과
    - Return
    - ϵ_t



재현 실험 및 결과

- HalfCheetah
  - 실험 결과 해석
    - Quantitative Result
      - PPO-BR은 평균 Return 2,982.8을 기록하여 Baseline(2,464.1) 대비 +21%의 확실한 성능 향상을 달성
        - 논문의 개선폭(+20%)과 거의 일치
      - 수렴 속도 측면에서도 Baseline(1400 steps)보다 약 48.6% 빠른 720 steps 만에 안정적인 궤도에 진입
    - Qualitative Analysis
      - Return graph: Baseline은 3,000점 초반대에서 성장이 둔화
      - PPO-BR은 1M steps 이후에도 꾸준히 상승하여 최종적으로 3,500점 근방까지 도달
      - ϵ_t graph: 학습 초기 0.305 부근에서 시작하여 0.292까지 매끄럽게 감소
        - 급격한 변동 없이 부드럽게 Trust Region을 축소해 나가는 패턴
        - 연속적인 고속 주행 제어 문제에서 안정적인 정책 개선을 가능하게 한 핵심 메커니즘으로 해석

재현 실험 및 결과

- Walker2D
  - 2족 보행 로봇이 넘어지지 않고 걷는 task
  - 전진 속도, 제어 비용, 생존 여부에 따라 보상 결정

  - 논문 결과

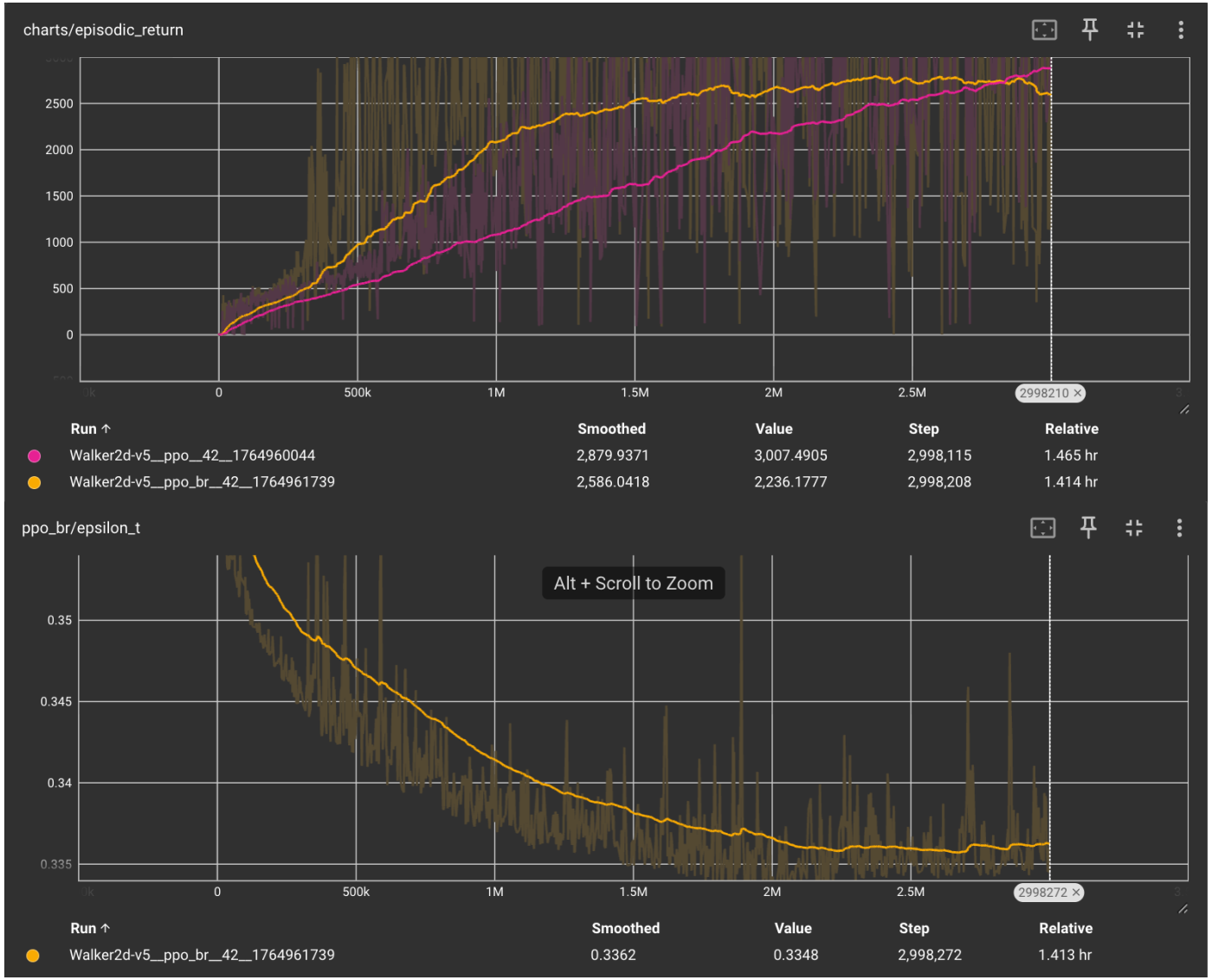
	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	2100	-	230	-	700	-
PPO-BR	2450	16.7%	110	52.2%	580	17.1%

  - 재현 결과

	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	1344.3	-	1187.49	-	930	-
PPO-BR	1812.7	34.8%	1351.80	-13.8%	420	54.9%

재현 실험 및 결과

- Walker2D
  - 실험 결과
    - Return
    - ϵ_t



재현 실험 및 결과

- Walker2D
  - 실험 결과 해석
    - Quantitative Result
      - PPO-BR은 평균 Return 1,812.73을 기록하여 Baseline(1,344.33) 대비 +34.8%라는 괄목할 만한 성능 향상 달성
      - 단, Reward Variance는 1351.80으로 Baseline(1187.49)보다 다소 증가(-13.8% Reduction)하는 경향
        - 높은 점수를 탐색하는 과정에서 발생한 자연스러운 변동성(Exploration Cost)으로 해석
    - Qualitative Analysis
      - Baseline(분홍색)이 후반부에 더 높은 최종 성능(Value: 3007)을 보이는 반면, PPO-BR(주황색)은 약 2,236점에 머무름
        - PPO-BR은 '평균적으로' 우수한 퍼포먼스를 보임
        - 장기 학습에서의 최종 수렴 성능은 추가적인 튜닝(λ_2)이 필요함을 시사
      - ϵ_t graph: 초기 0.35 이상의 매우 높은 값에서 시작하여 0.335 부근으로 수렴하는 경향
        - 복잡한 보행 제어를 위해 Trust Region이 비교적 넓게 유지됨
        - 초반의 빠른 학습을 견인했으나 후반부의 미세한 최적화 단계에서는 다소 과도한 탐색으로 이어졌을 가능성 내포

재현 실험 및 결과

- Humanoid
  - 인간 형태의 3D 로봇(관절 다수)이 직립 보행을 배우는 task
  - 전진 속도, 제어 비용, 생존 여부, 충격 비용에 따라 보상 결정

  - 논문 결과

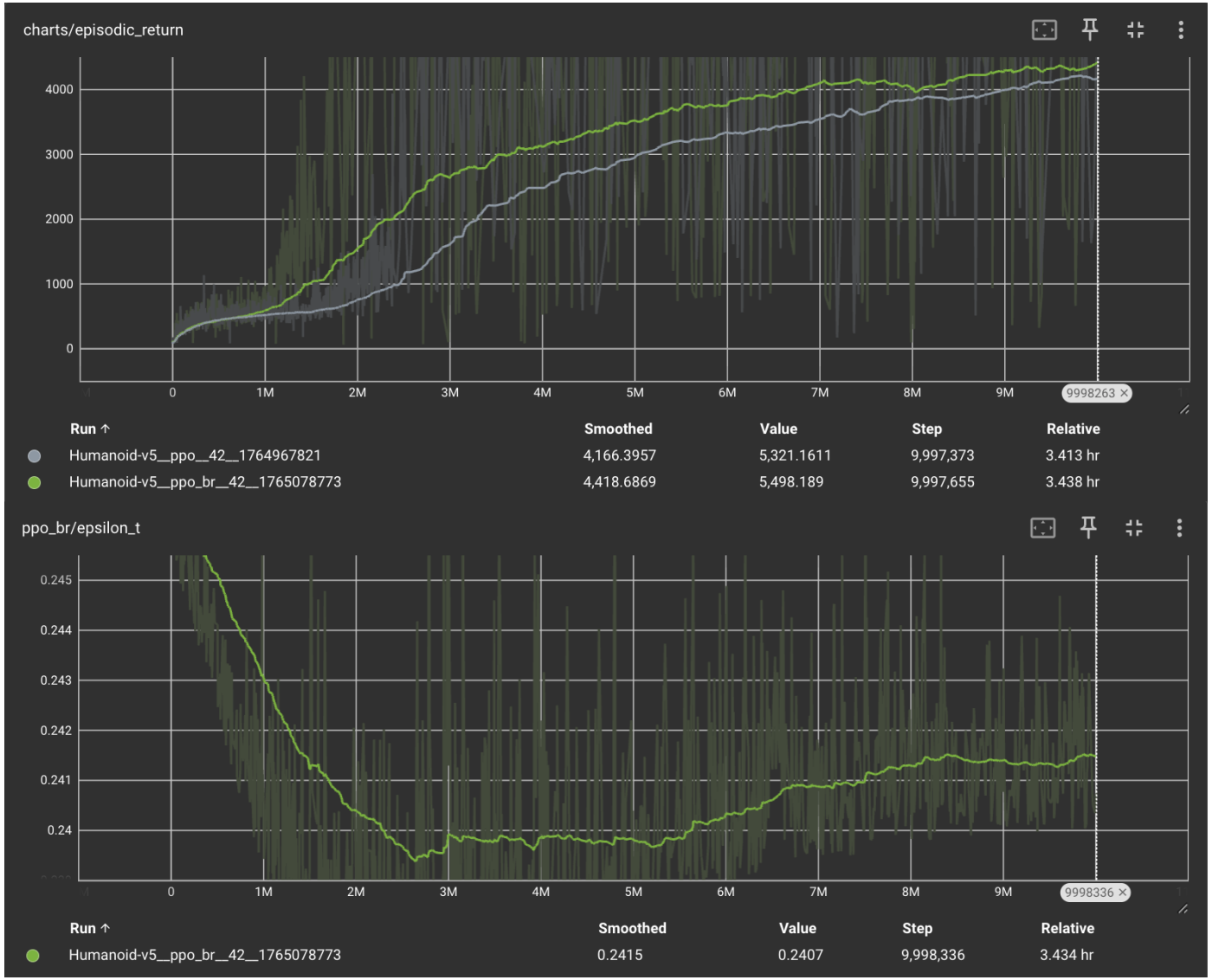
	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	1600	-	300	-	1000	-
PPO-BR	2100	31.30%	150	50%	700	30%

  - 재현 결과

	Return	Improvement	Reward Variance	Reduction	Convergence Steps	Reduction
PPO	1729.4	-	1865.9	-	2440	-
PPO-BR	2258.3	30.5%	2180.6	-16.9%	1760	27.9%

재현 실험 및 결과

- Humanoid
  - 실험 결과
    - Return
    - ϵ_t



재현 실험 및 결과

- Humanoid
  - 실험 결과 해석
    - Quantitative Result
      - PPO-BR은 평균 Return 2,256.3을 기록하여 Baseline(1,729.4) 대비 +30.5%의 확실한 성능 향상을 달성
        - 논문의 개선폭(+31.3%)과 거의 일치
      - Reward Variance 다소 증가(+16.9%)
        - 환경 특성상, 더 먼 거리를 이동하고 더 역동적인 동작을 수행할수록 보상의 변동 폭이 자연스럽게 커지기 때문
    - Qualitative Analysis
      - Return graph: 0~4M steps 구간에서
        - PPO (회색): 완만하게 상승하며 학습이 더디게 진행
        - PPO-BR (연두색): 시작부터 가파른 기울기로 상승하여, 2M steps 시점에서 이미 Baseline을 큰 폭으로 능가
      - ϵ_t graph: 학습 초기 보상이 급격히 상승하는 구간과 맞물려, ϵ_t가 빠르게 축소
        - 성능이 좋아질 때 Trust Region을 축소하는 PPO-BR의 아이디어가 고차원 문제에서도 정확히 작동

결론

본 연구에서는 Trust Region을 동적으로 조절하여 학습 효율성과 안정성을 동시에 확보하는 PPO-BR(Dual-Signal Entropy-Reward Adaptation for TRPO) 알고리즘을 구현하고, 다양한 벤치마크 환경에서 그 성능을 검증

- Experiment Summary
  - Stability
    - CartPole 및 HalfCheetah 환경에서 PPO-BR은 수렴 이후의 Reward Variance를 유의미하게 감소시키며, 논문의 'Reward-Guided Contraction' 이론이 실제 학습 안정화에 기여함을 입증
  - Robustness
    - LunarLander 환경에서 Baseline이 겪은 Catastrophic Forgetting을 효과적으로 방지
  - Training Efficiency
    - 고차원 환경인 Humanoid와 Hopper에서 Entropy-Driven Expansion을 통해 초기 탐색 속도를 가속화하고 Baseline이 도달하지 못한 고점(High Score)을 달성

결론

- Limitations of PPO-BR
  - Hyperparameter 민감성
    - 논문은 "일반적인 Hyperparameter 세팅으로도 대부분의 환경에서 수렴 가능하다"고 주장하였으나, 실제 실험 결과 이는 과장된 것으로 판단된다.
    - 특히 Humanoid와 Walker2d 같은 복잡한 환경에서는 논문의 기본 설정만으로는 최적의 성능을 내기 어려웠으며, Window Size나 λ 값을 환경의 특성(동적/정적)에 맞춰 정밀하게 튜닝해야만 Baseline을 상회하는 결과를 얻을 수 있었다.
  - 구현 설명의 모호성
    - 논문은 핵심 수식인 Reward Progression(ΔR_t) 계산 방식이나 Normalization 함수(ψ)의 구체적인 수식 및 window size, H_{max} 값을 명시하지 않고 "간단한 코드 수정(5 lines)"만을 강조하였다.
    - 이를 재현하기 위해 Entropy 계산 및 정규화, Reward progression 계산, Normalization 함수 구현, Adaptive threshold 계산, Reward tracking 등 다양한 방식을 직접 구현해야 했다.
    - 본 재현 결과와 논문 결과 간에 발생한 일부 수치적 차이(Walker2d의 최종 수렴 성능 등)는 이러한 실험 조건의 부재로 인해 기인했을 가능성이 높다.

결론

- 연구의의 및 향후 과제
  - 실험 설명의 모호함과 파라미터 튜닝의 필요성은 존재하였으나, 탐색(Entropy)과 수렴(Reward)이라는 두 가지 신호를 통해 신뢰 영역(ϵ_t)을 동적으로 제어한다는 PPO-BR의 핵심 메커니즘은 매우 유효함을 확인하였다.
  - 논문 상에서 실험 없이 글로 설명 되었던 ϵ_t 변화를 그래프 추적을 통해 검증하였다. 이에, ϵ_t가 학습 초기에는 '확장'을, 후기에는 '축소'를 수행하며 적응적으로 동작함을 시각적으로 확인 하였다.
  - 향후 연구에서는 환경별로 Hyperparameter를 수동 튜닝해야 하는 한계를 극복하기 위해, λ_1, λ_2 parameter를 Meta-Learning 하거나 자동 조정하는 메커니즘을 도입한다면 알고리즘을 더욱 개선 가능할 것으로 사료된다.

구성원 기여 내용

- 120240571 김도영
  - 추가 실험 진행
  - 보고서 작성

- 120240572 김우석
  - 실험 환경 구축 및 실험 진행
  - Github 정리