PPO×Family

第四讲：解密稀疏奖励空间

主办 中国计算机学会 CCF

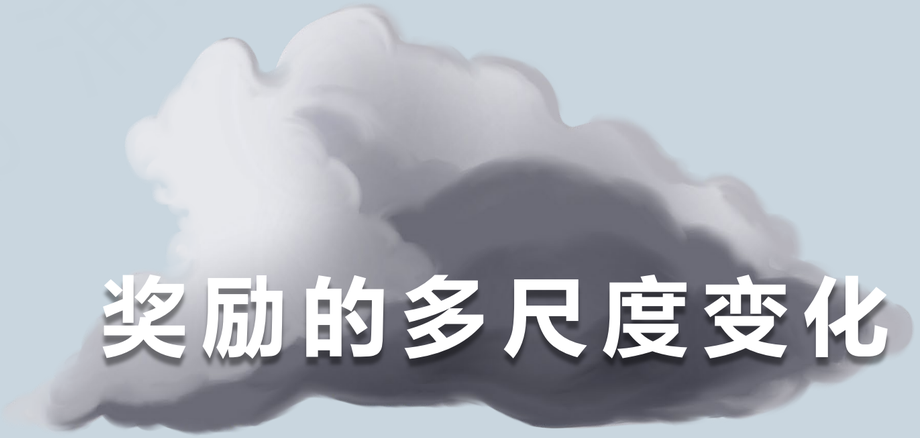
承办 上海人工智能实验室 Shanghai Artificial Intelligence Laboratory

协办 GAIAA 北京大学人工智能研究院 INSTITUTE FOR ARTIFICIAL INTELLIGENCE, PEKING UNIVERSITY 浙江大学上海高等研究院 SHANGHAI INSTITUTE FOR ADVANCED STUDY ZHEJIANG UNIVERSITY 商汤 sensetime

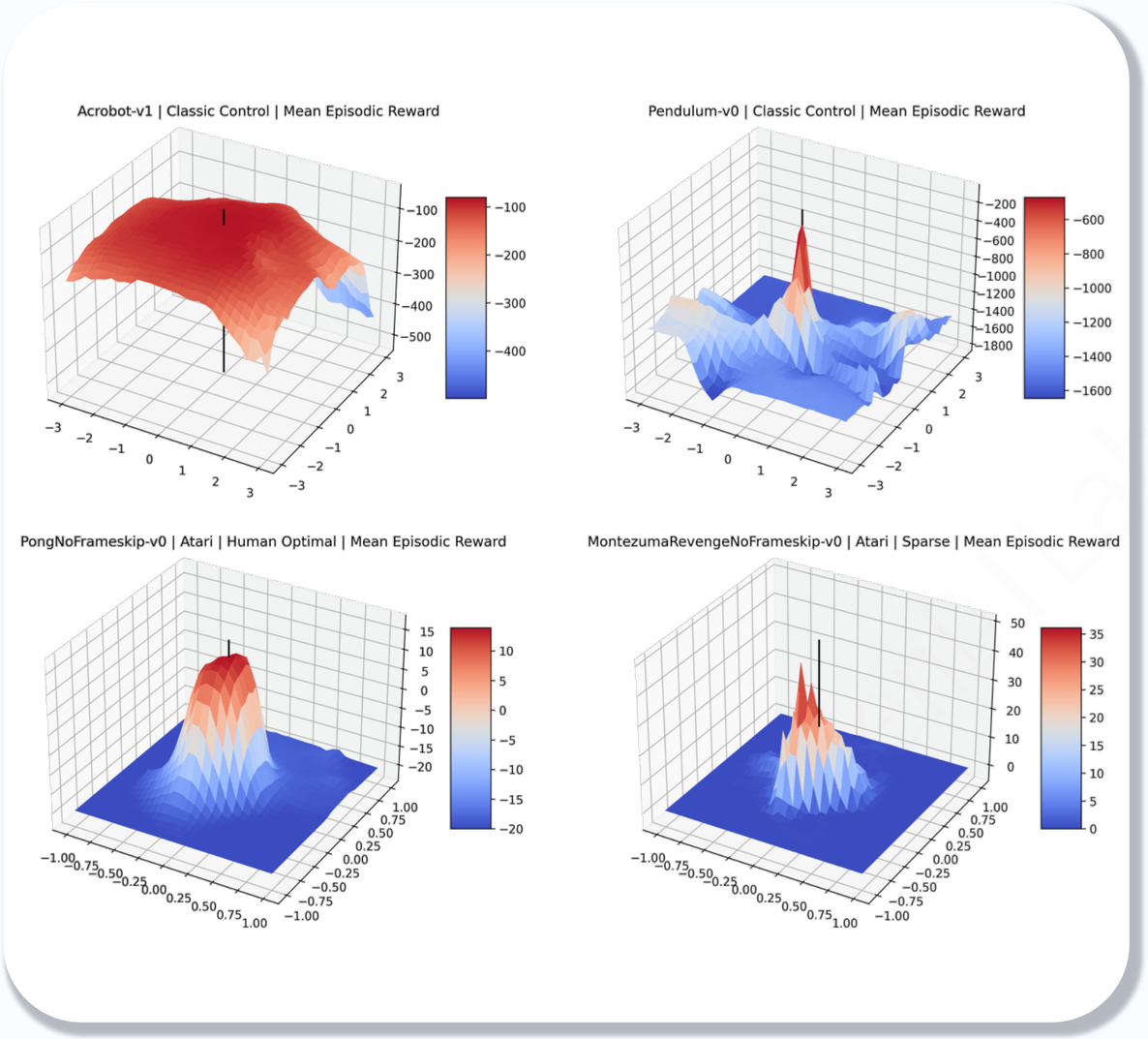
支持 知乎 机器之心 智海 新一代人工智能科教平台 paperweekly

奖励空间上的两朵乌云

01



02

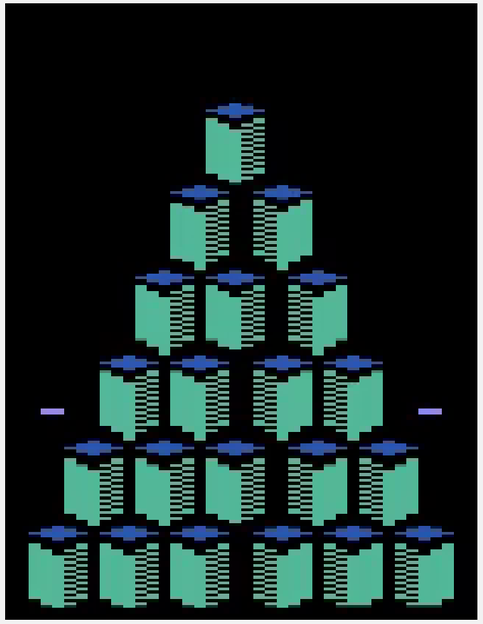


Reward Space Overview

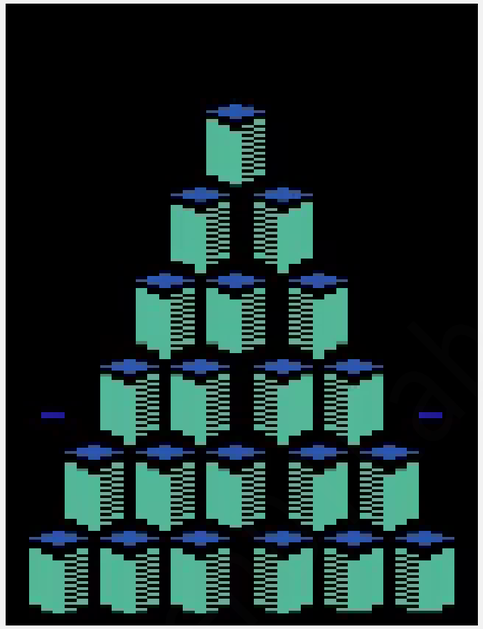
奖励空间概述

Ref: http://aaai-rlg.mlanctot.info/papers/AAAI22-RLG_paper_38.pdf

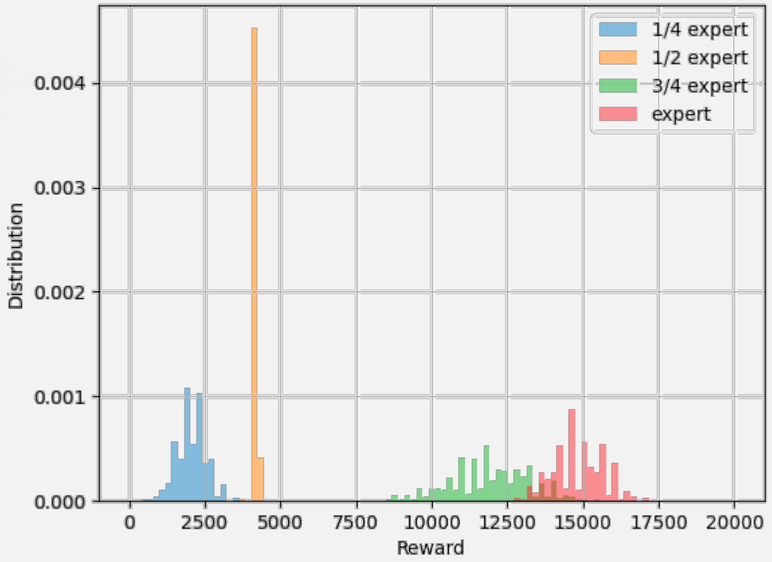
千差万别的奖励空间



训练前期



训练后期



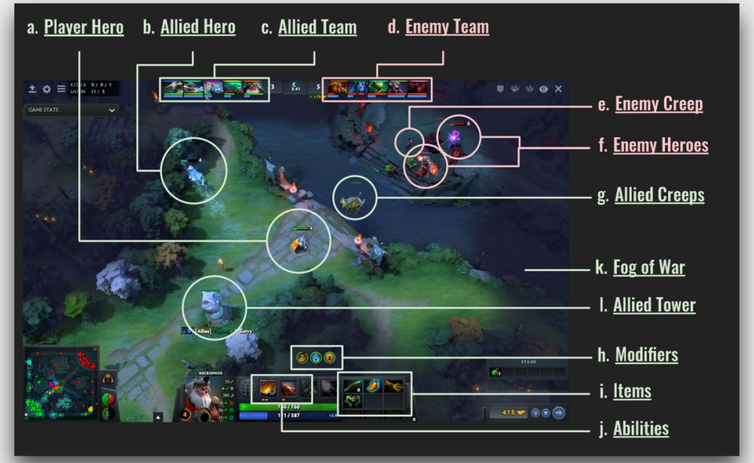
不同策略对应的奖励分布

特点

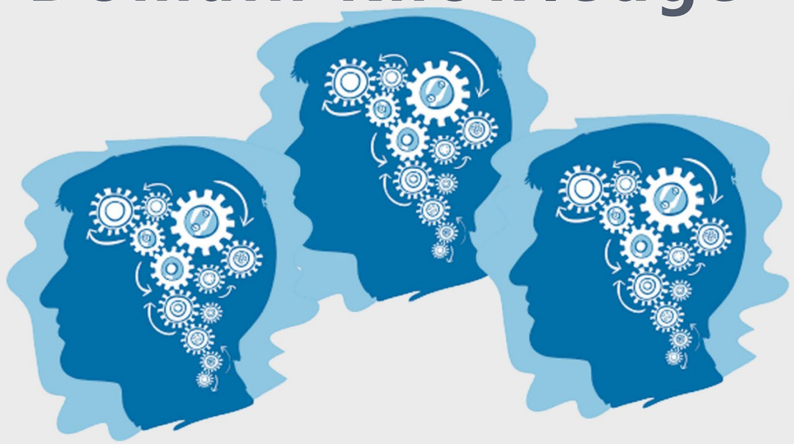
- **不同环境的奖励取值分布不同，**具体决策问题的定义相关
- **同一环境不同训练阶段，episode的长度、奖励的分布都可能不同**

  RL训练的不同阶段探索到不同的环境空间子集

Game Mechanism



Domain Knowledge



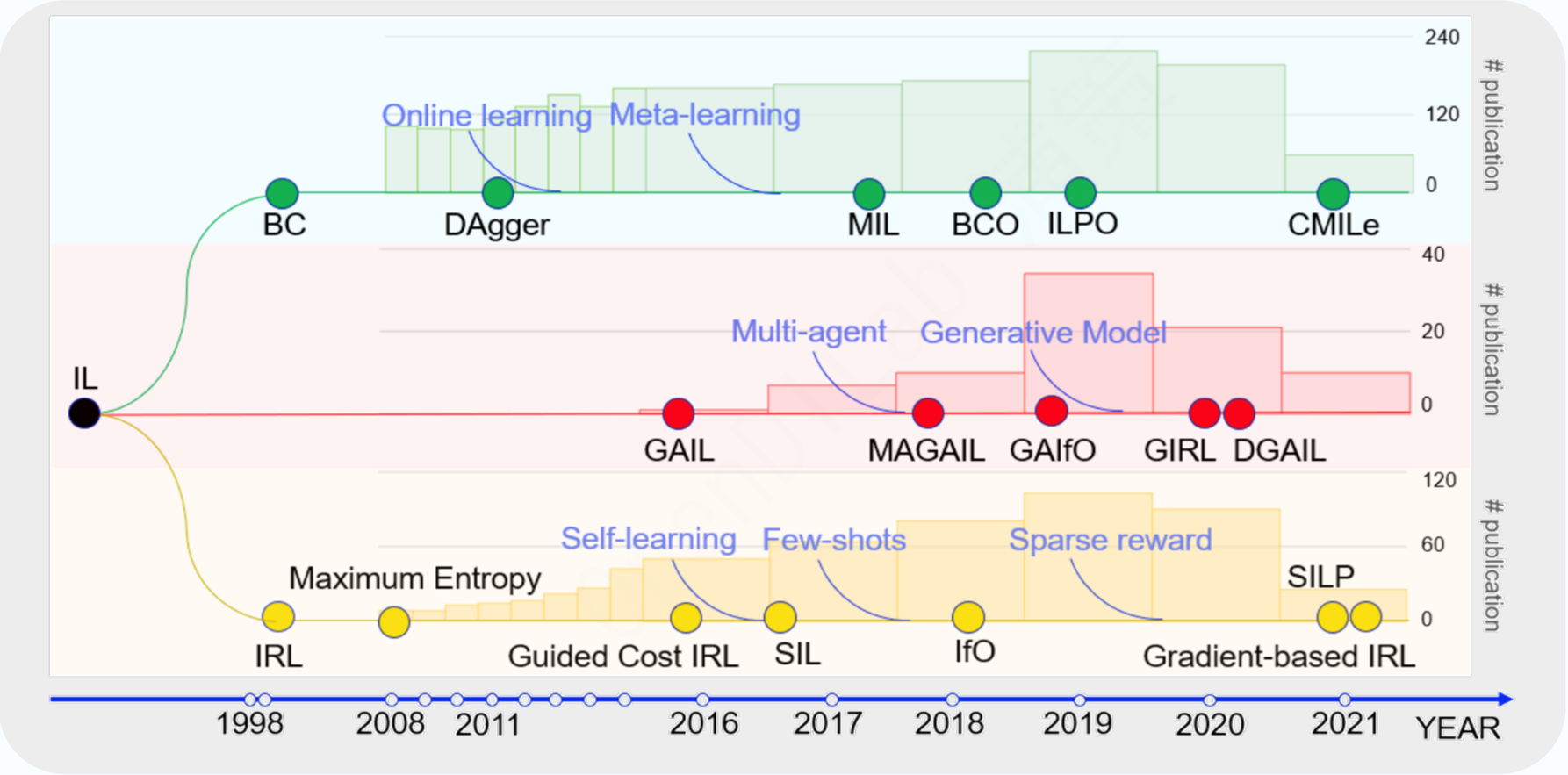
OpenAI Five: https://openai.com/five/

Reward Shaping

奖励塑形

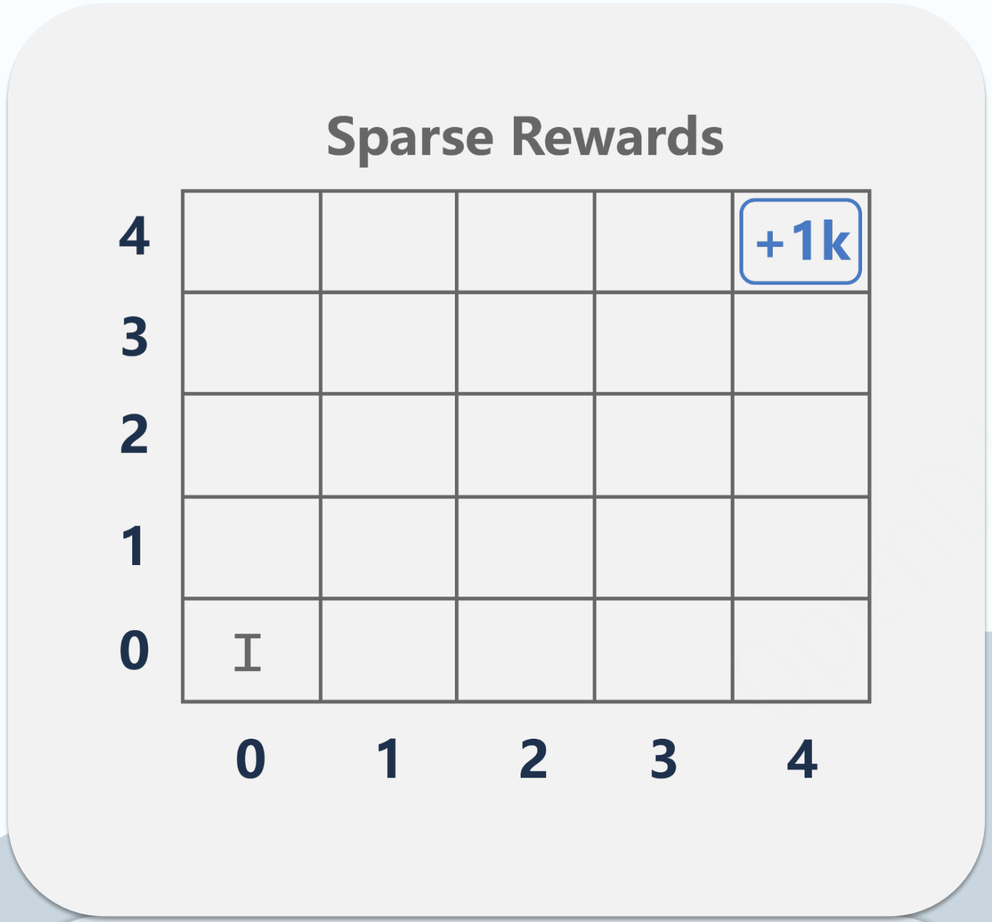
Name	Reward	Heroes	Description
Win	5	Team	
Hero Death	-1	Solo	
Courier Death	-2	Team	
XP Gained	0.002	Solo	
Gold Gained	0.006	Solo	For each unit of gold gained. Reward is not lost when the gold is spent or lost.
Gold Spent	0.0006	Solo	Per unit of gold spent on items without using courier.
Health Changed	2	Solo	Measured as a fraction of hero's max health.‡
Mana Changed	0.75	Solo	Measured as a fraction of hero's max mana.
Killed Hero	-0.6	Solo	For killing an enemy hero. The gold and experience reward is very high, so this reduces the total reward for killing enemies.
Last Hit	-0.16	Solo	The gold and experience reward is very high, so this reduces the total reward for last hit to ~ 0.4.
Deny	0.15	Solo	
Gained Aegis	5	Team	
Ancient HP Change	5	Team	Measured as a fraction of ancient's max health.
Megas Unlocked	4	Team	
T1 Tower*	2.25	Team	
T2 Tower*	3	Team	
T3 Tower*	4.5	Team	
T4 Tower*	2.25	Team	
Shrine*	2.25	Team	
Barracks*	6	Team	
Lane Assign†	-0.15	Solo	Per second in wrong lane.

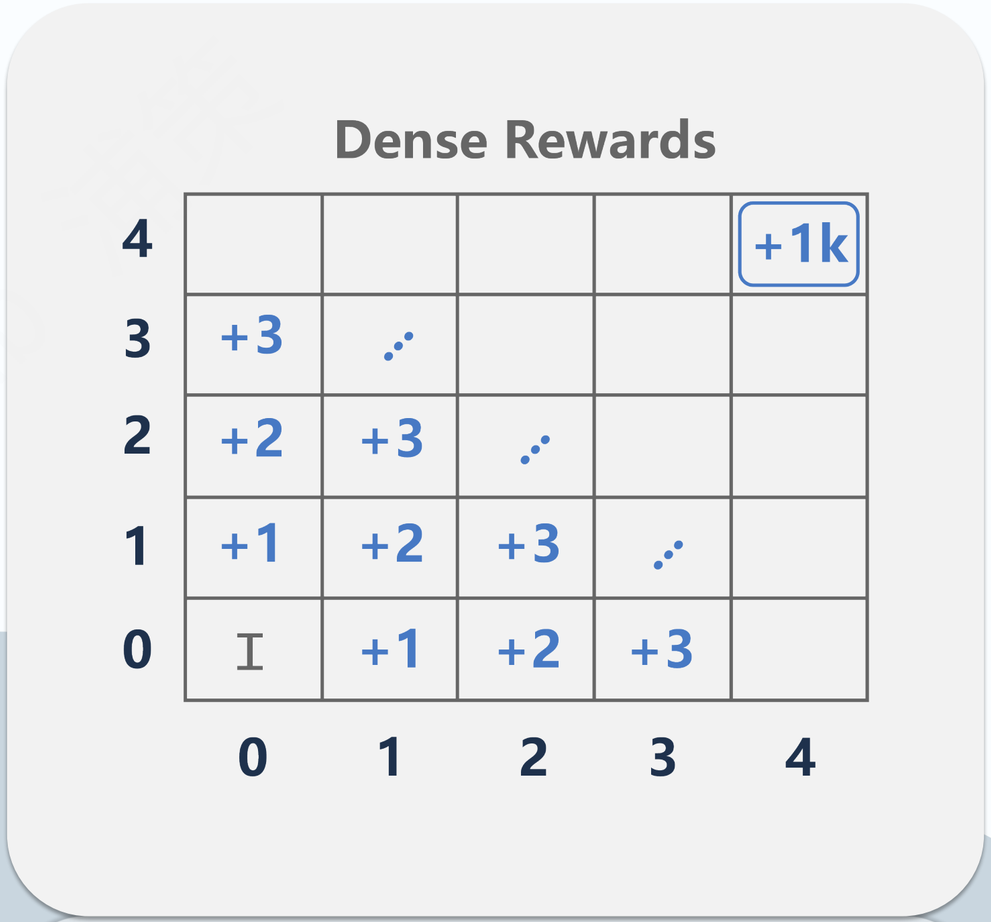
模仿学习



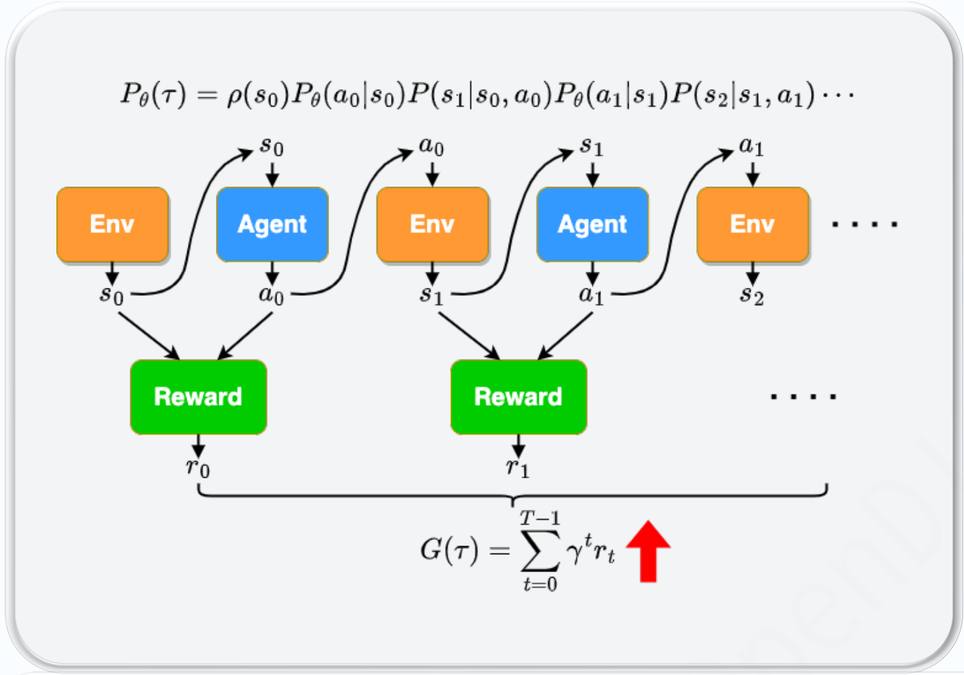
Imitation Learning: Progress, Taxonomies and Challenges: https://arxiv.org/pdf/2106.12177.pdf
逆强化学习补充材料：https://github.com/opendilab/PPOxFamily/tree/main/chapter4_reward/chapter4_supp_irl.pdf
行为克隆补充材料：https://github.com/opendilab/PPOxFamily/tree/main/chapter4_reward/chapter4_supp_bc.pdf

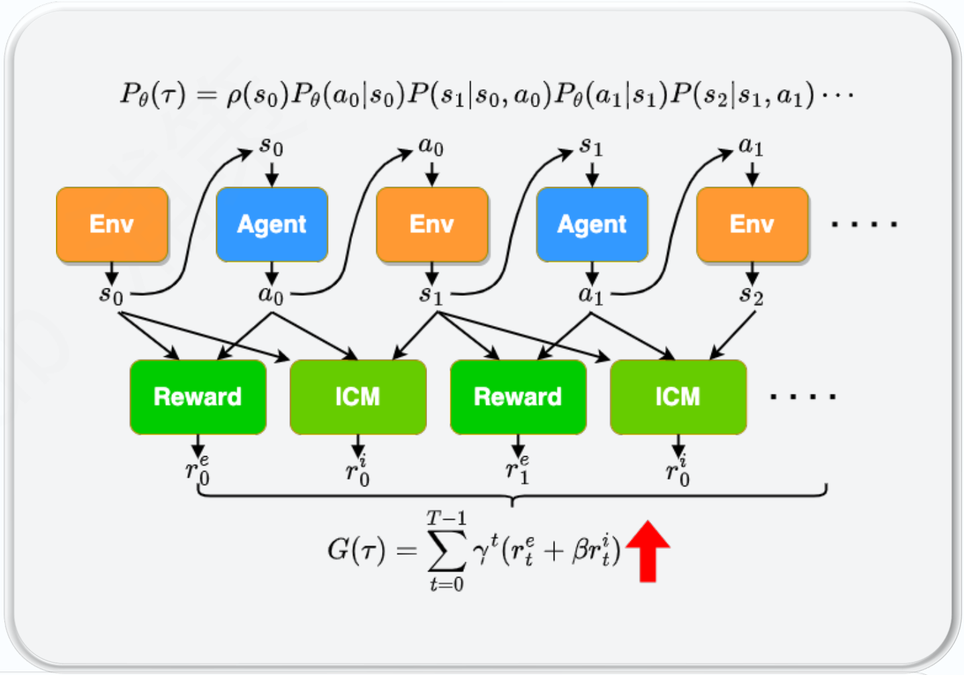
奖励的稀疏性


Sparse Rewards
4
3
2
1
0
+1k
I
0
1
2
3
4


Dense Rewards
4
3
2
1
0
+1k
+3
+2
+3
+1
+2
+3
I
+1
+2
+3
0
1
2
3
4

理论：PPO+好奇心机制●动机



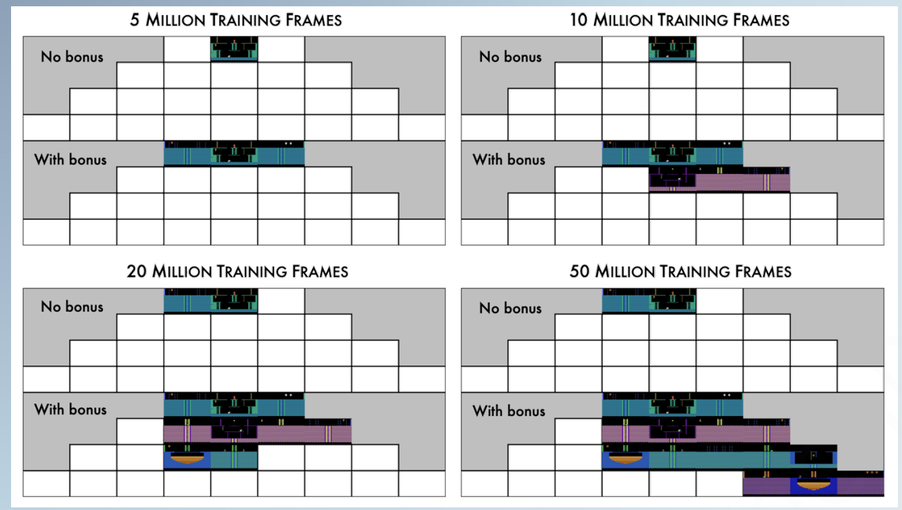


设计新的内在（Intrinsic）奖励，激发智能体的好奇心

- 原则：激励智能体执行有利于减少环境不确定性的动作
- 原理：根据“状态-动作”的新颖性给出内在奖励，越新颖越大

理论：PPO+好奇心机制●尝试

朴素做法



- 计数or伪计数，越小越新颖
- 神经网络预测误差，越大越新颖

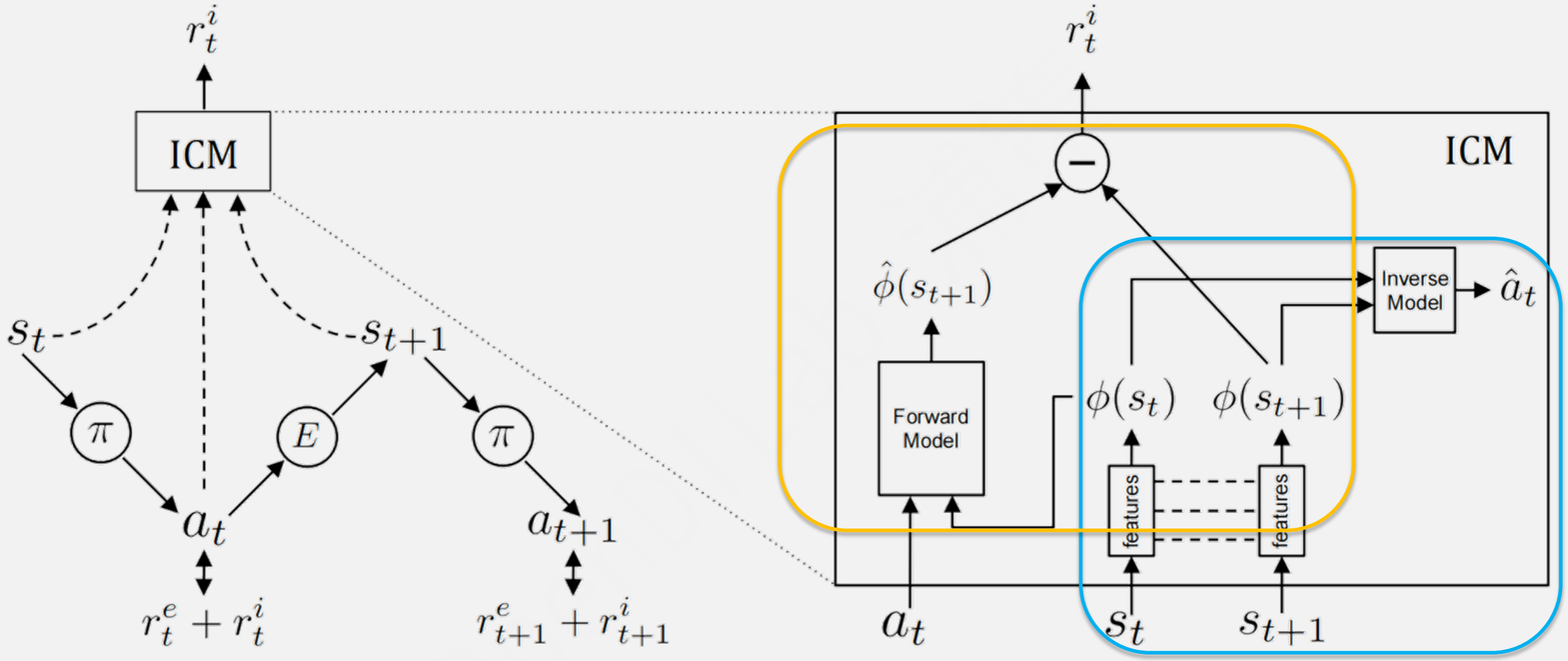
现存问题

- 高维表征难以计数or预测
- 表征信息中的噪声会带来副作用

ICM: https://arxiv.org/pdf/1705.05363.pdf

Methodology

理论：PPO+好奇心机制●方法



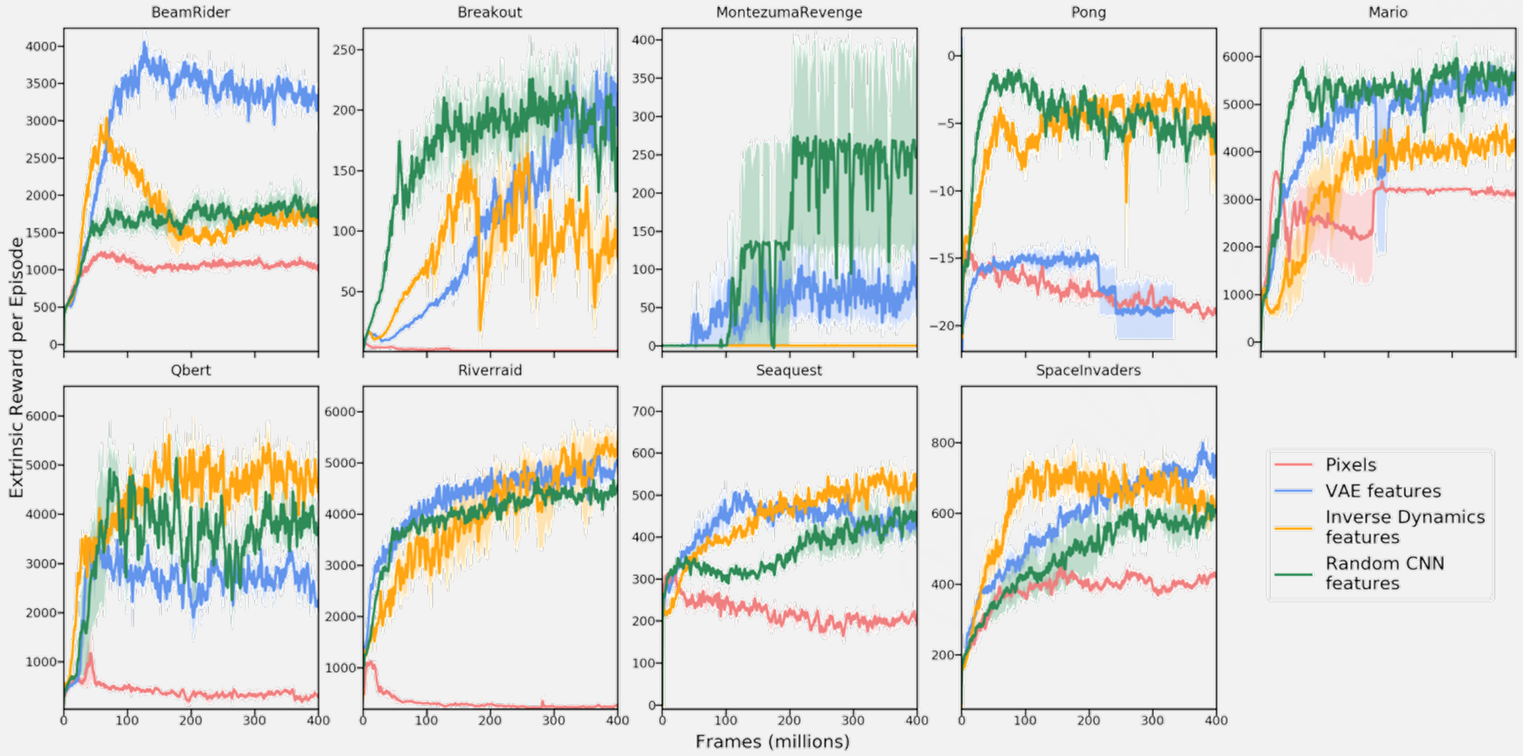
只有好奇心是不够的，还要知道什么对决策是重要的

- （关注）智能体动作直接控制的内容（例如：智能体开枪射出子弹）
- （关注）虽然不受智能体动作控制，但是会对智能体决策造成影响的内容（例如：怪物的移动）
- （忽略）本质上对决策无效的信息，如背景，噪声等

ICM: https://arxiv.org/pdf/1705.05363.pdf

Extension

理论：PPO+好奇心机制●延伸



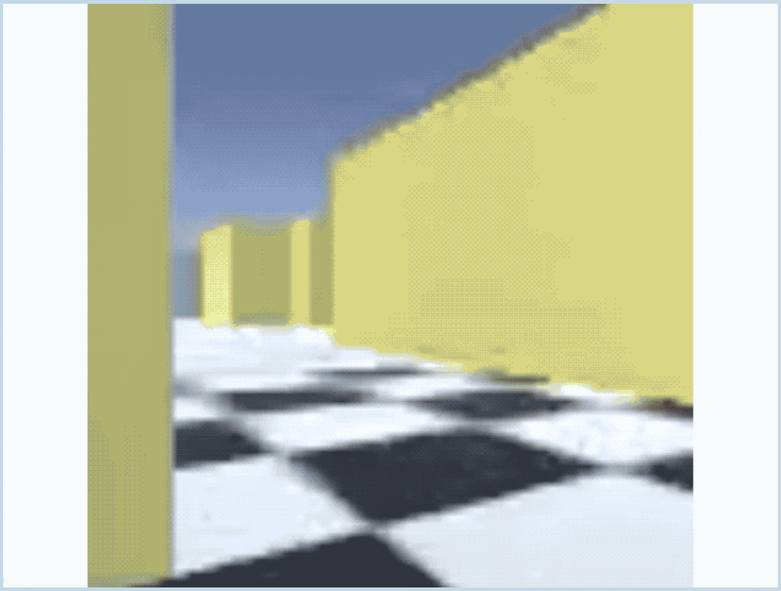
什么是适于探索的表征

- 紧凑：特征应该易于建模，最好是低维的，并且过滤掉了原始状态空间的和探索无关的部分。
- 充分：特征应该包含用于决策的所有重要信息。否则，智能体可能由于缺乏信息，做出错误决策。
- 稳定：（Non-stationary）非平稳的奖励使RL学习过程变得困难

Large-Scale Study of Curiosity-Driven Learning: https://arxiv.org/pdf/1808.04355.pdf
ICM中文解读博客：https://zhuanlan.zhihu.com/p/473676311

Motivation

理论：PPO+RND 动机



RND
(Random Network Distillation)

- 原则：激励智能体探索更多的新颖状态
- 原理：构建只和观测状态相关的随机蒸馏问题，
  预测误差越大，近似说明智能体之前访问的次数少，
  从而该状态新颖性较大

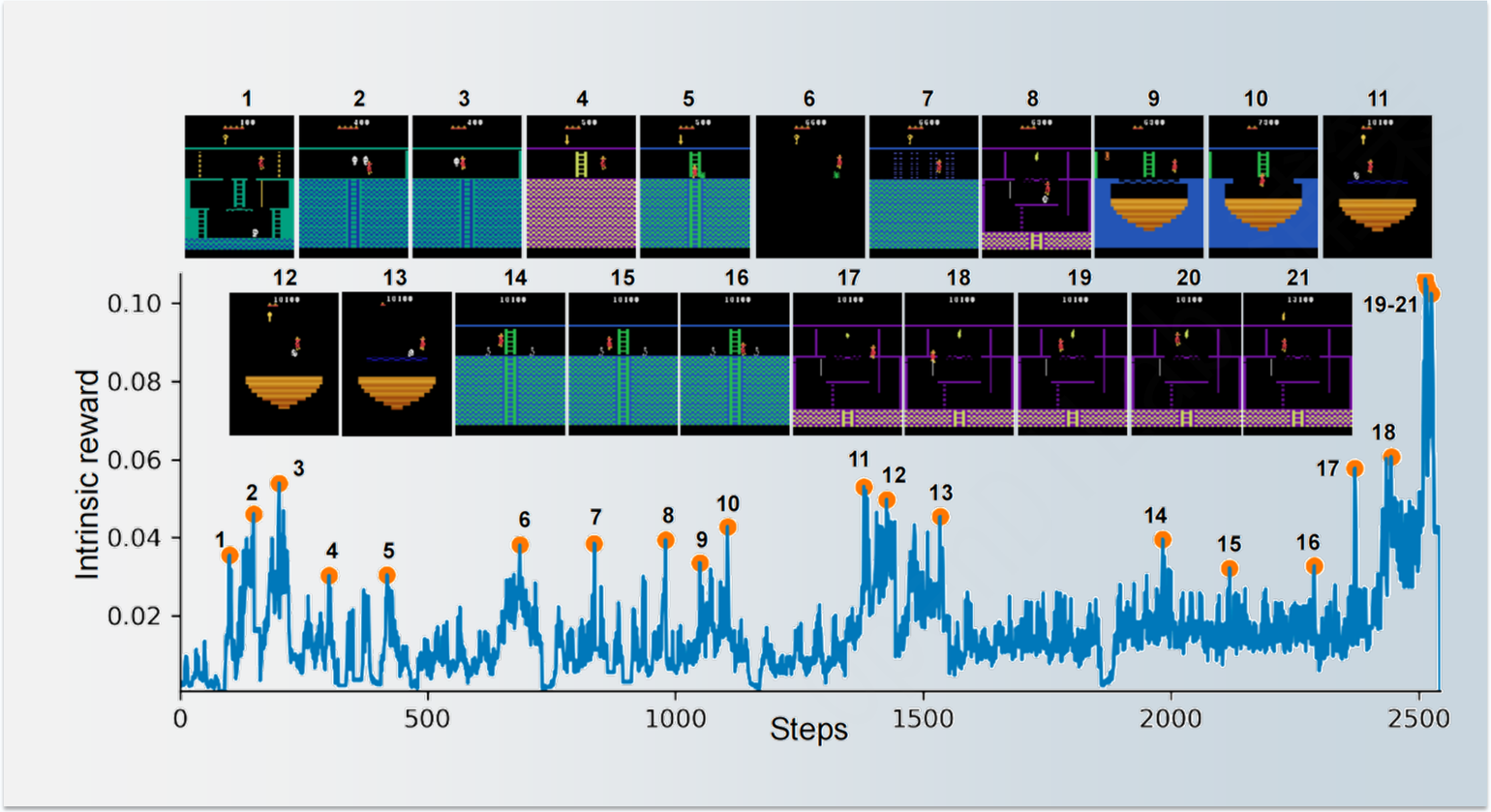
ICM可能失败的场景：

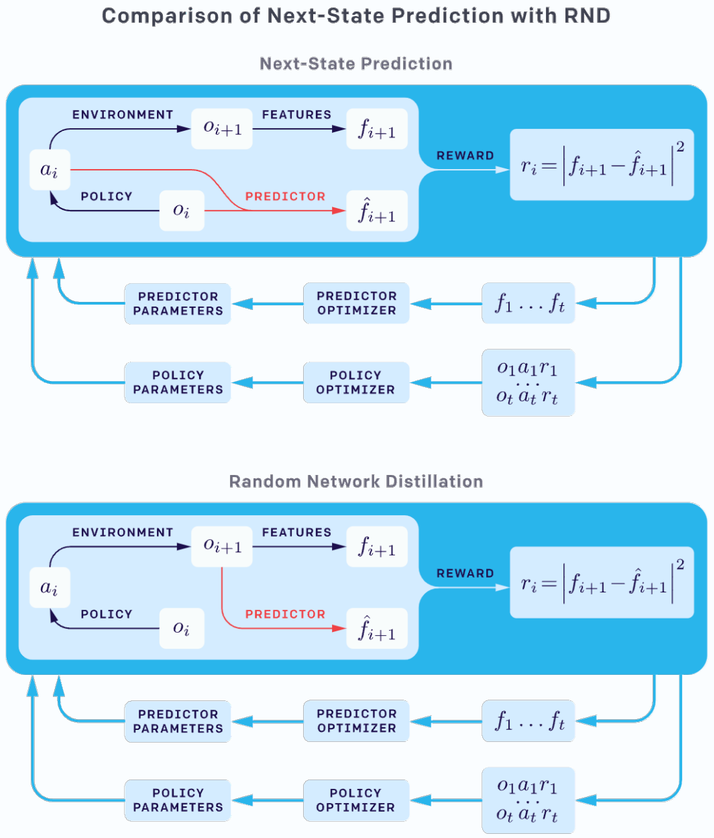
- 状态转移非常复杂，神经网络容量有限
- 状态转移是随机函数，例如 noisy-TV 问题

RND: https://arxiv.org/pdf/1810.12894.pdf

Methodology

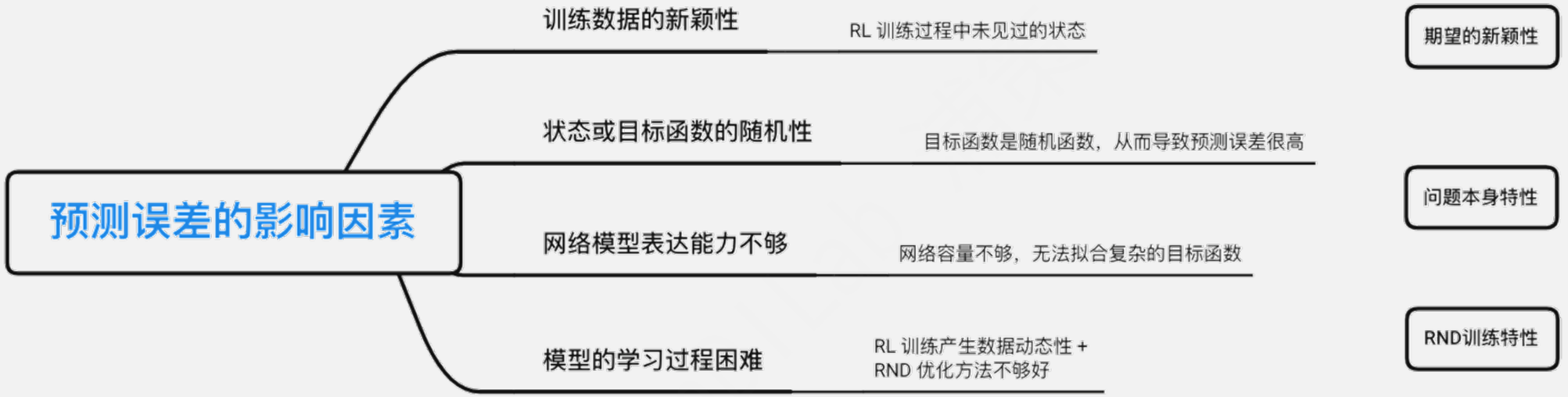
理论：PPO+RND●方法





RND: https://arxiv.org/pdf/1810.12894.pdf

理论：PPO+RND 思考



如果我们通过一个自监督学习任务，比如AutoEncoder，
将状态映射到隐空间上，并根据重建误差，来给出内在奖励，
这样会有什么问题呢？

RND中文解读博客: https://zhuanlan.zhihu.com/p/485476646

代码：奖励模型训练技巧面面观

Algorithm 1 RND + PPO (RND related is highlighted in blue)

Input: initialize policy parameters θ and value parameters ϕ. Training epochs per collect E, max number of episodes K, trajectory length T, batch size B. RND: initialize prediction network parameters $\hat{\chi}$ and fixed target network parameters χ, number of prediction optimization steps per collect N_{predict},
for $k = 0, 1, 2, ..., K$ **do**
  Collect a set of trajectories $\mathcal{D}_k = \{\{s_t, s_{t+1}, a_t, e_t\}\}$ by running policy π_{θ_k} interaction with environment.
  Normalize the observation in $\mathcal{D}_k$, then obtain the normalized counterpart: $\hat{s}_t, \hat{s}_{t+1}$.
  for $j = 1$ **to** N_{predict} **do**
    Optimize χ_f wrt distillation loss $\|\hat{f}_{\hat{\chi}}(\hat{s}_t) - f_{\chi}(\hat{s}_t)\|^2$ using Adam.
  end for
  Calculate intrinsic reward $i_t = \|\hat{f}_{\hat{\chi}}(\hat{s}_t) - f_{\chi}(\hat{s}_t)\|^2$. Then normalize the intrinsic reward and obtain the normalized counterpart: $\hat{i}_t$.
  Calculate augmented reward $r_t = e_t + \beta \hat{i}_t$ and obtain the augmented trajectories $\hat{\mathcal{D}}_k = \{\{s_t, s_{t+1}, a_t, r_t\}\}$.
  Compute trajectory target return estimates $\hat{R}_t$ on $\hat{\mathcal{D}}_k$. (e.g. n-step TD or other methods)
  Compute advantage estimates $\hat{A}^{\theta_k}$ with value V_{ϕ_k} on $\hat{\mathcal{D}}_k$. (e.g. GAE or other methods)
  for $e = 0, 1, ..., E-1$ **do**
    for minibatch:$b \in \hat{\mathcal{D}}_k$ **do**
      Update the policy by maximizing the PPO-Clip objective with Adam:

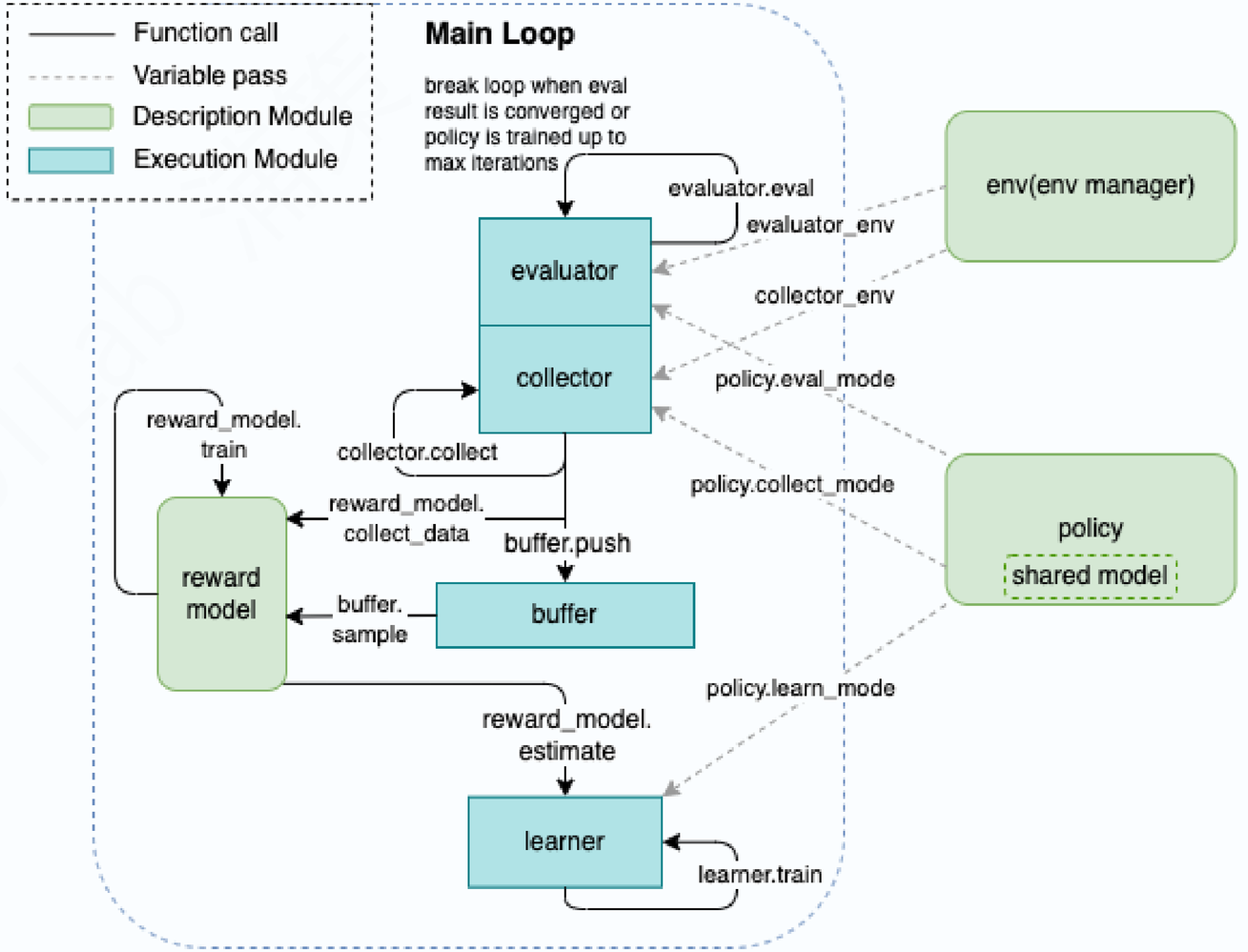
$$L_\theta = \frac{1}{B \cdot T} \sum_{\tau \in \mathrm{b}} \sum_{t=0}^{T-1} \min(r(\theta)\hat{A}^{\theta_k}(s_t, a_t), \mathrm{clip}(r(\theta), 1-\epsilon, 1+\epsilon)\hat{A}^{\theta_k}(s_t, a_t))$$

$$r(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_k}(a_t|s_t)}$$

      Fit the value by regression on mean-squared error with Adam:

$$L_\phi = \frac{1}{B \cdot T} \sum_{\tau \in \mathrm{b}} \sum_{t=0}^{T-1} (V_\phi(s_t) - \hat{R}_t)^2$$

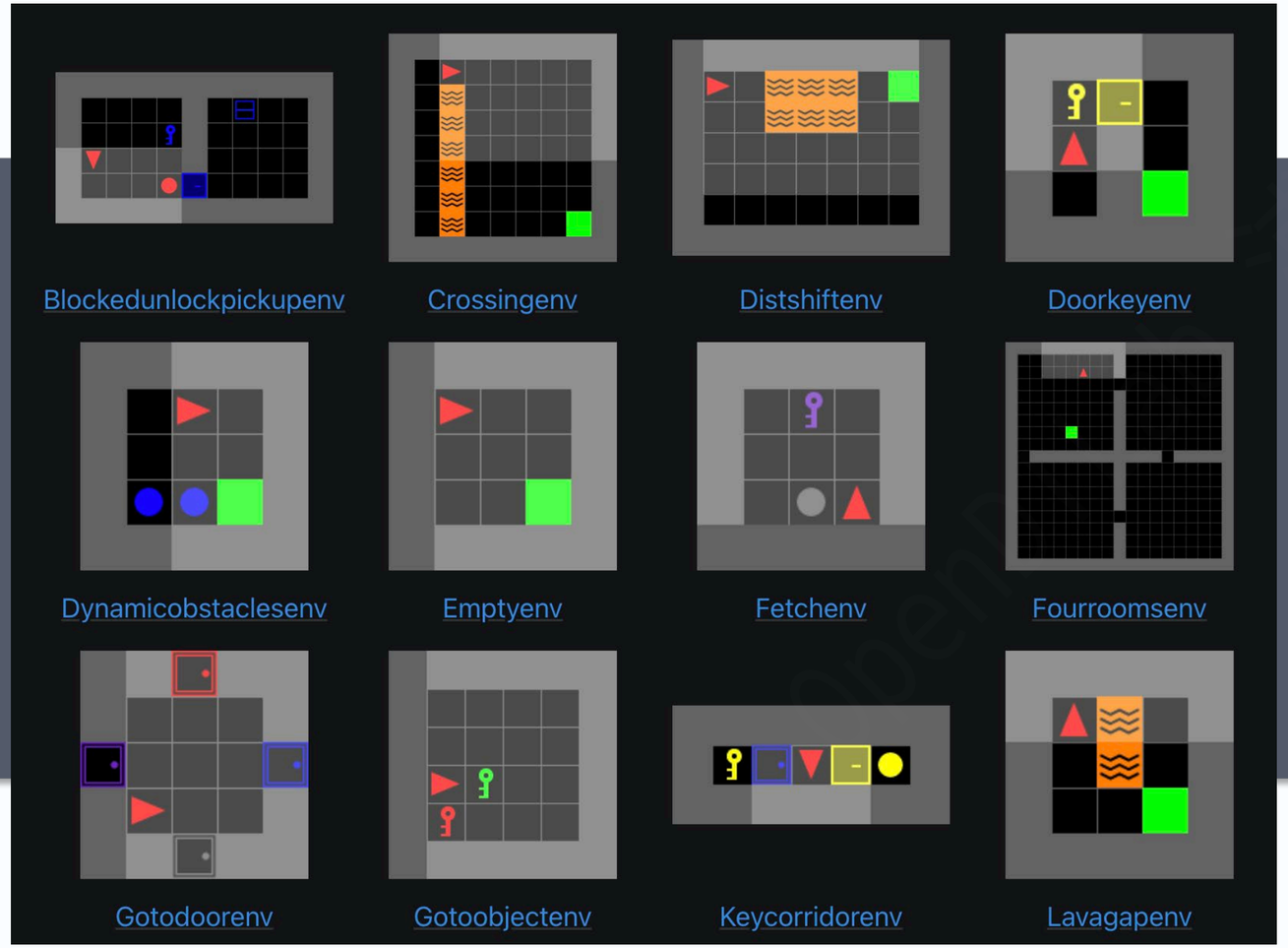
    end for
  end for
end for



完整示例：https://opendilab.github.io/PPOxFamily/

Practice

实践：PPO+minigrid（迷宫）



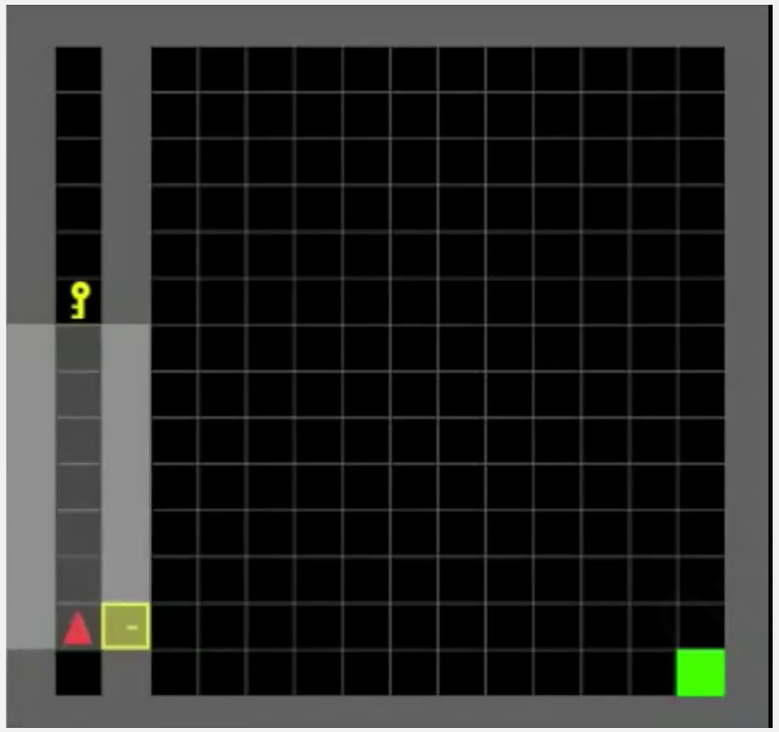
特点

- 稀疏奖励环境，只有在智能体（红色）到达目标点（绿色）时才获得大于零的奖励，具体的数值由达到目标所用的总步数决定，在这之前奖励都是0

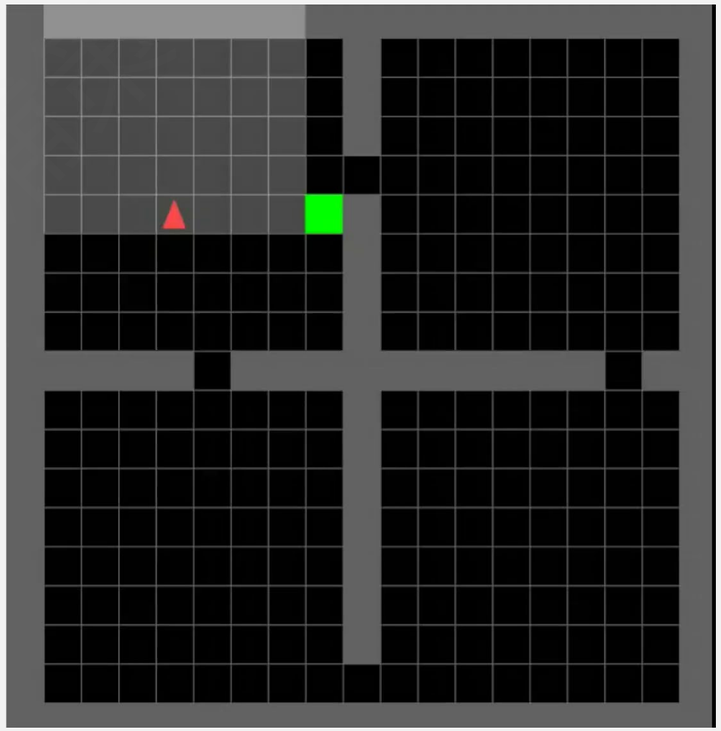
minigrid: https://github.com/Farama-Foundation/Minigrid
DI-engine + minigrid doc: https://di-engine-docs.readthedocs.io/zh_CN/latest/13_envs/minigrid_zh.html

Practice

实践：PPO+minigrid



door-key

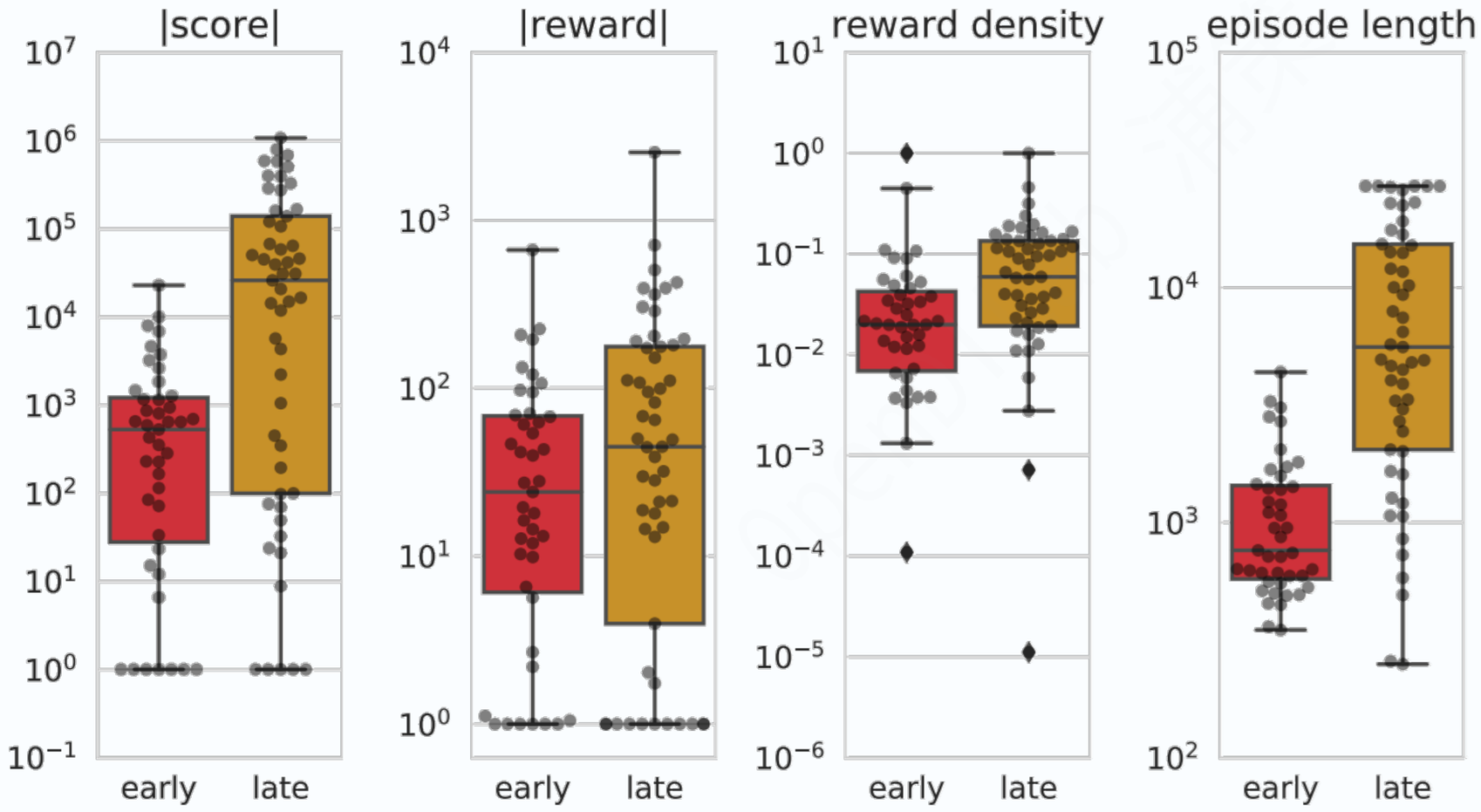


four-room

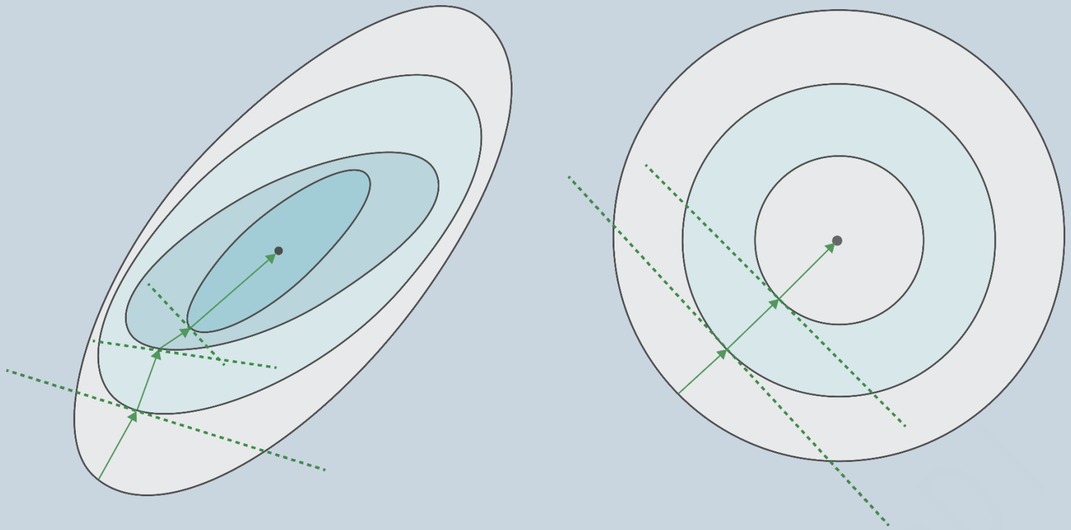
视频样例：https://github.com/opendilab/PPOxFamily/issues/44

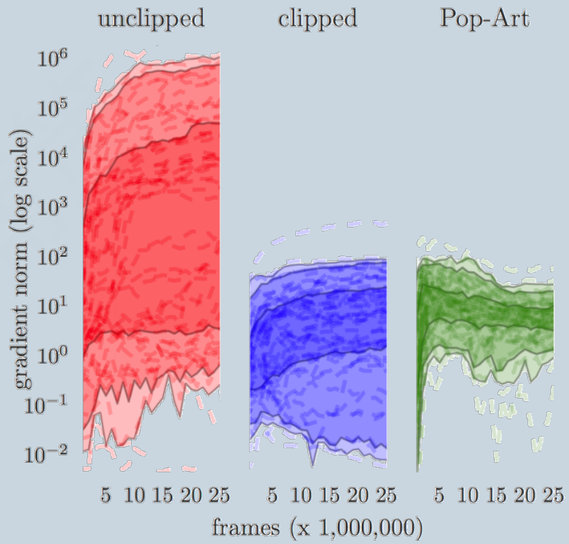
Multi Magnitude Reward

奖励的多尺度变化



理论：PPO+奖励剪裁

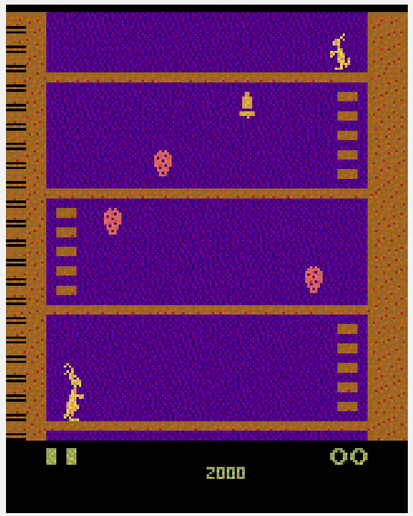


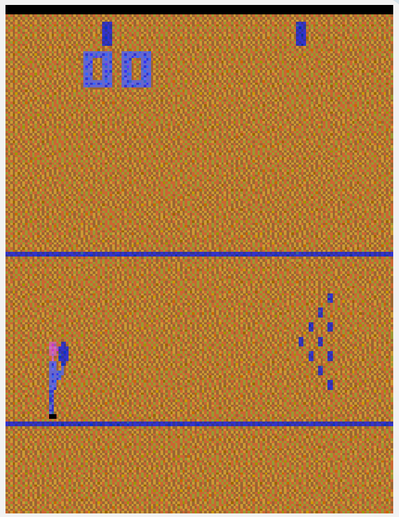


- **深度学习角度**：在更新参数时，如果奖励函数变化范围很大（进而导致损失函数变化大），则很难设定合理的学习率。
- **强化学习角度**：智能体需要先在奖励较低的范围里表现出色，才能抵达奖励较高的范围。并且较高的损失会主导训练过程

$$\text{sign}(\text{reward}) = \begin{cases} 1 & \text{reward} > 0 \\ 0 & \text{reward} = 0 \\ -1 & \text{reward} < 0 \end{cases}$$

理论：PPO+Pop-Art 动机





- 奖励裁剪可能改变决策问题的含义，无法区分不同水准的决策行为

暴力裁剪的问题

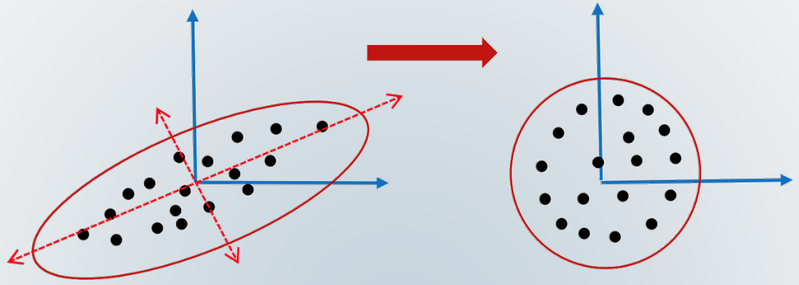
$$\tilde{Y}_t = \Sigma_t^{-1}(Y_t - \mu_t)$$

- 标准化使用的均值和方差难以确定
- 可以使用指数加权平均（EMA）来自适应计算，但这样标准化会引入新的非平稳问题

简单标准化的问题

PopArt: https://arxiv.org/pdf/1602.07714.pdf

理论：PPO+Pop-Art 原理

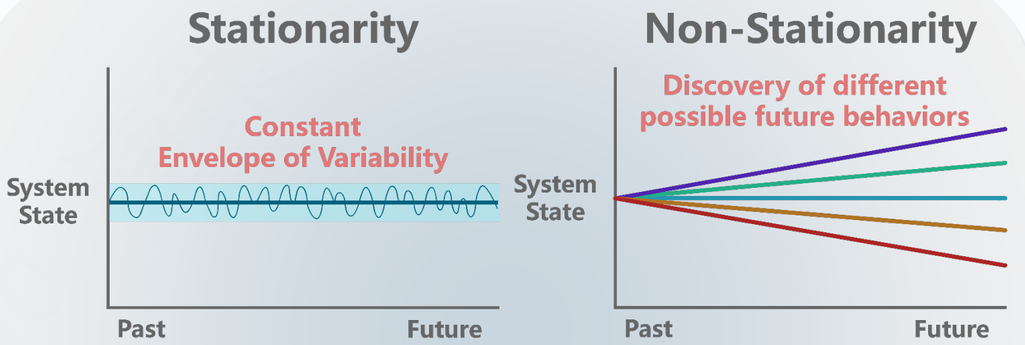


$$\sum_{i=1}^{t}(Y_i-\mu_t)/\sigma_t=0 \quad \text{and} \quad \frac{1}{t}\sum_{i=1}^{t}(Y_i-\mu_t)^2/\sigma_t^2=1$$

$$\mu_t=\frac{1}{t}\sum_{i=1}^{t}Y_i \quad \text{and} \quad \sigma_t=\frac{1}{t}\sum_{i=1}^{t}Y_i^2-\mu_t^2\,.$$

$$\mu_t=(1-\beta_t)\mu_{t-1}+\beta_t Y_t \qquad \nu_t=(1-\beta_t)\nu_{t-1}+\beta_t Y_t^2$$

Art
(Adaptively rescaling targets)



$$f_{\theta,\Sigma,\mu,W,b}(x)=f_{\theta,\Sigma_{new},\mu_{new},W_{new},b_{new}}(x)$$

$$W_{new}=\Sigma_{new}^{-1}\Sigma W$$

$$b_{new}=\Sigma_{new}^{-1}(\Sigma b+\mu-\mu_{new})$$

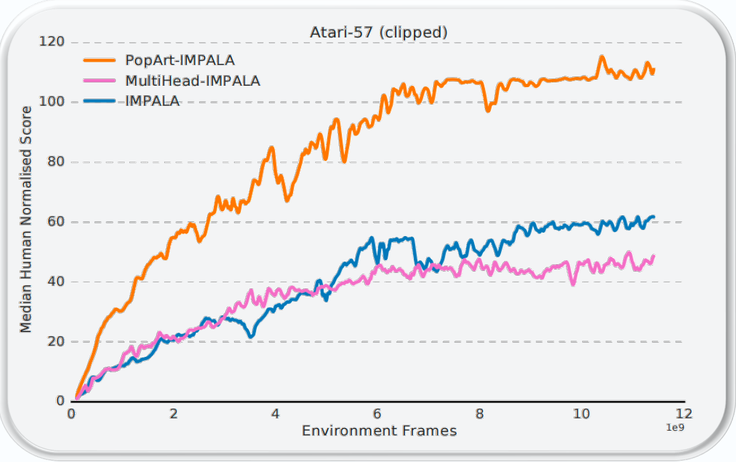
Pop
(Preserving outputs precisely)

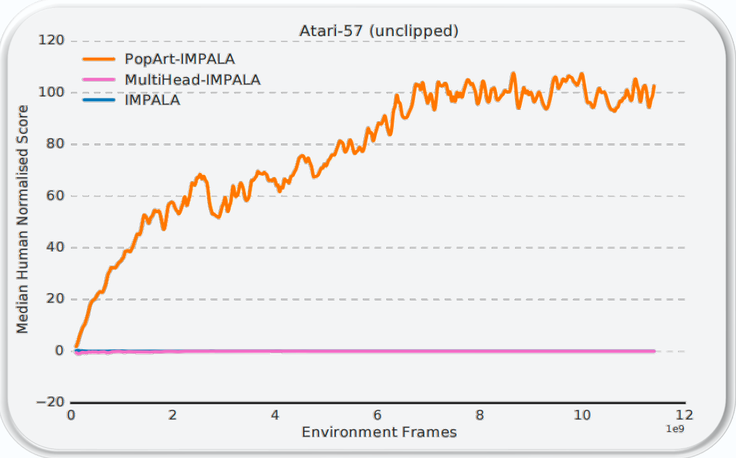
注：$f_{\theta,\Sigma,\mu,W,b}(x)=\Sigma(Wh_\theta(x)+b)+\mu$

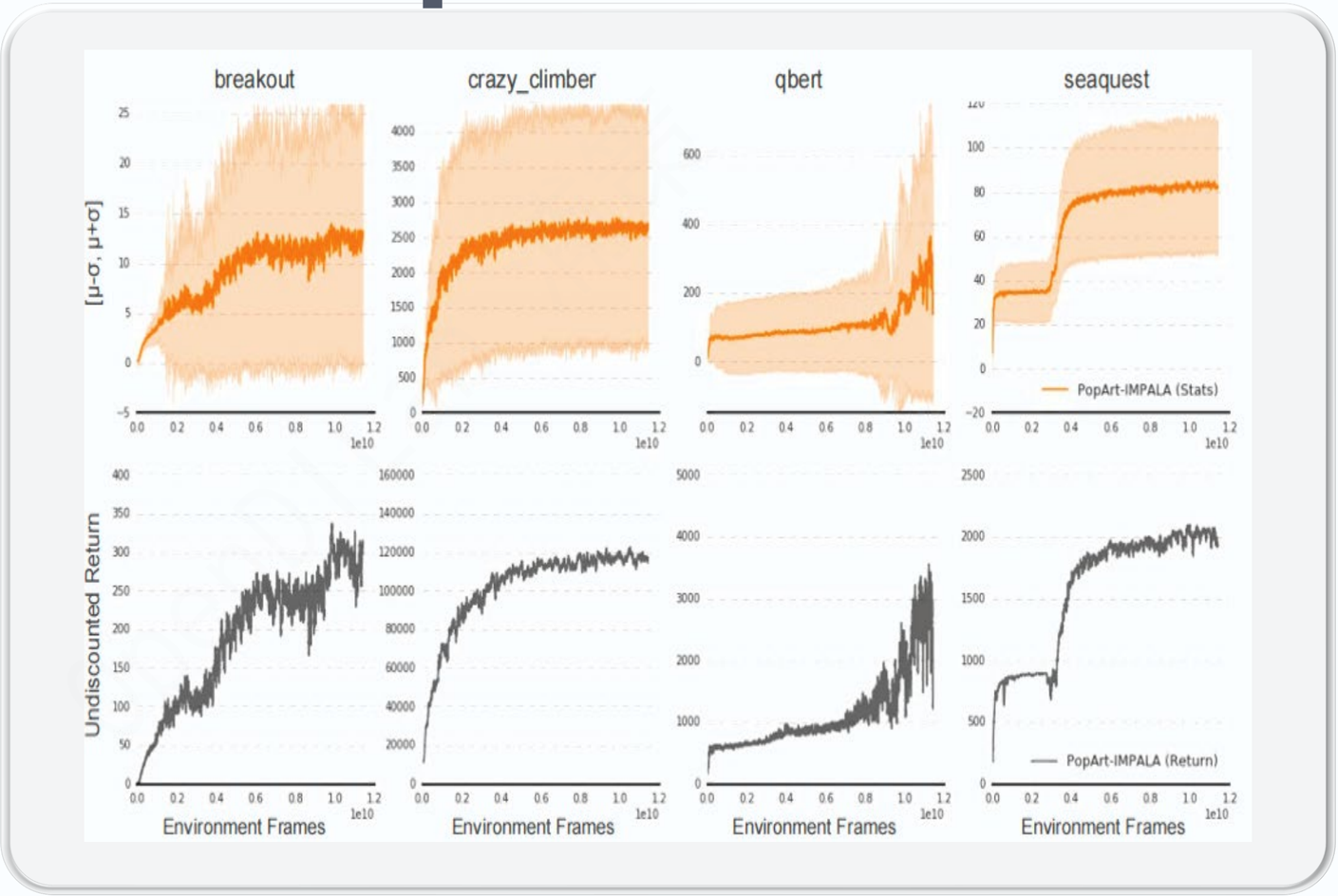
PopArt: https://arxiv.org/pdf/1602.07714.pdf

理论：PPO+PopArt

分析





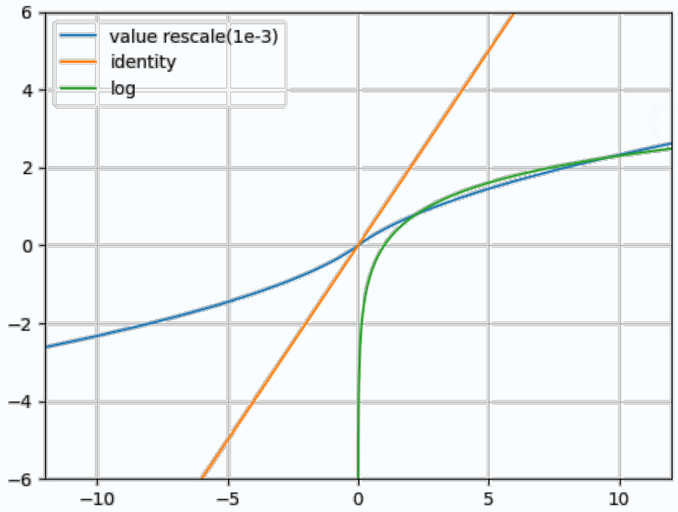


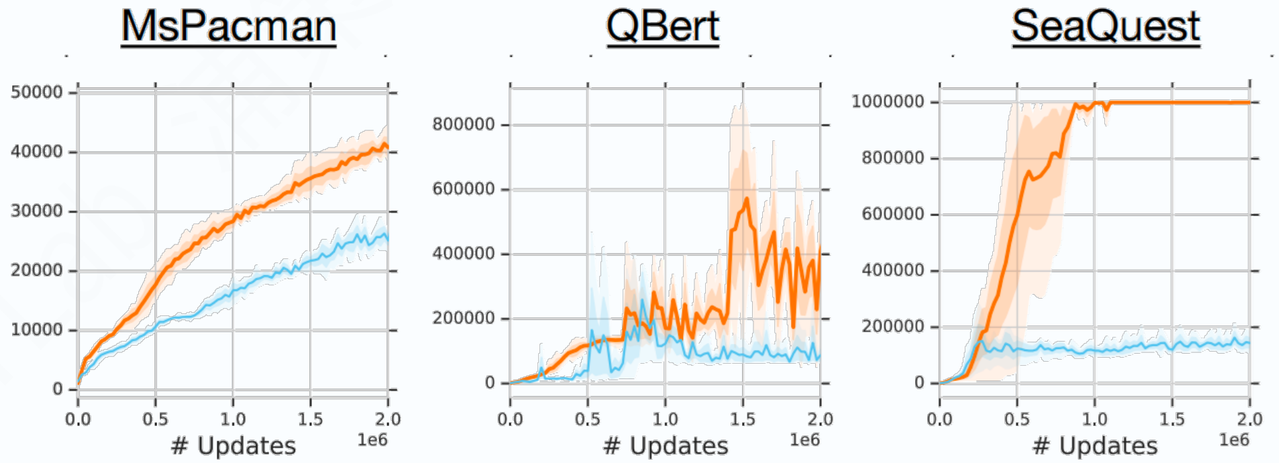
PopArt: https://arxiv.org/pdf/1602.07714.pdf

理论：PPO+Value Rescale

$$h : z \mapsto \mathrm{sign}(z)(\sqrt{|z|+1}-1)+\epsilon z$$

- h 是一个严格单调递增的函数
- 存在闭式的（closed form）逆函数
- 末尾项保证逆函数Lipschitz连续
- 确定性的 MDP





Observe and Look Further: https://arxiv.org/pdf/1805.11593.pdf

代码：reward clip/norm 操作的实现

HOME

Stars 1.1k | bilibili video course | Follow @opendilab

View code on GitHub

PyTorch implementation of reward clipping and normalization operations to suppress multi-magnitude reward problem,
which can be easily integrated in PPO or other RL algorithms.
This document mainly includes:
- value rescale transform and its inverse transform Related Link
- exponential moving average (EMA) mean and std
- Pop-Art Related Link

Overview
Implementation of value rescale operation, eps can be selected in [1e-2, 1e-3, 1e-4] .
h(x) is a strictly monotonically increasing function, and the final additive regularization term ensures that the inverse transform is Lipschitz continuous.
This transformation form has the desired effect of reducing the scale of the targets while being Liptschitz continuous and admitting a closed form inverse.

```python
import numpy as np
import torch


def value_rescale_transform(x: torch.Tensor, eps: float = 1e-2) -> torch.Tensor:
```

Reduce value scale:

$$h(x) = \text{sign}(x)(\sqrt{|x|+1} - 1) + \epsilon x$$

```python
    return torch.sign(x) * (torch.sqrt(torch.abs(x) + 1) - 1) + eps * x


```

Overview
Implementation of value rescale inverse operation, eps can be in [1e-2, 1e-3, 1e-4] .

```python
def value_inv_transform(x: torch.Tensor, eps: float = 1e-2) -> torch.Tensor:
```

Recover value scale:

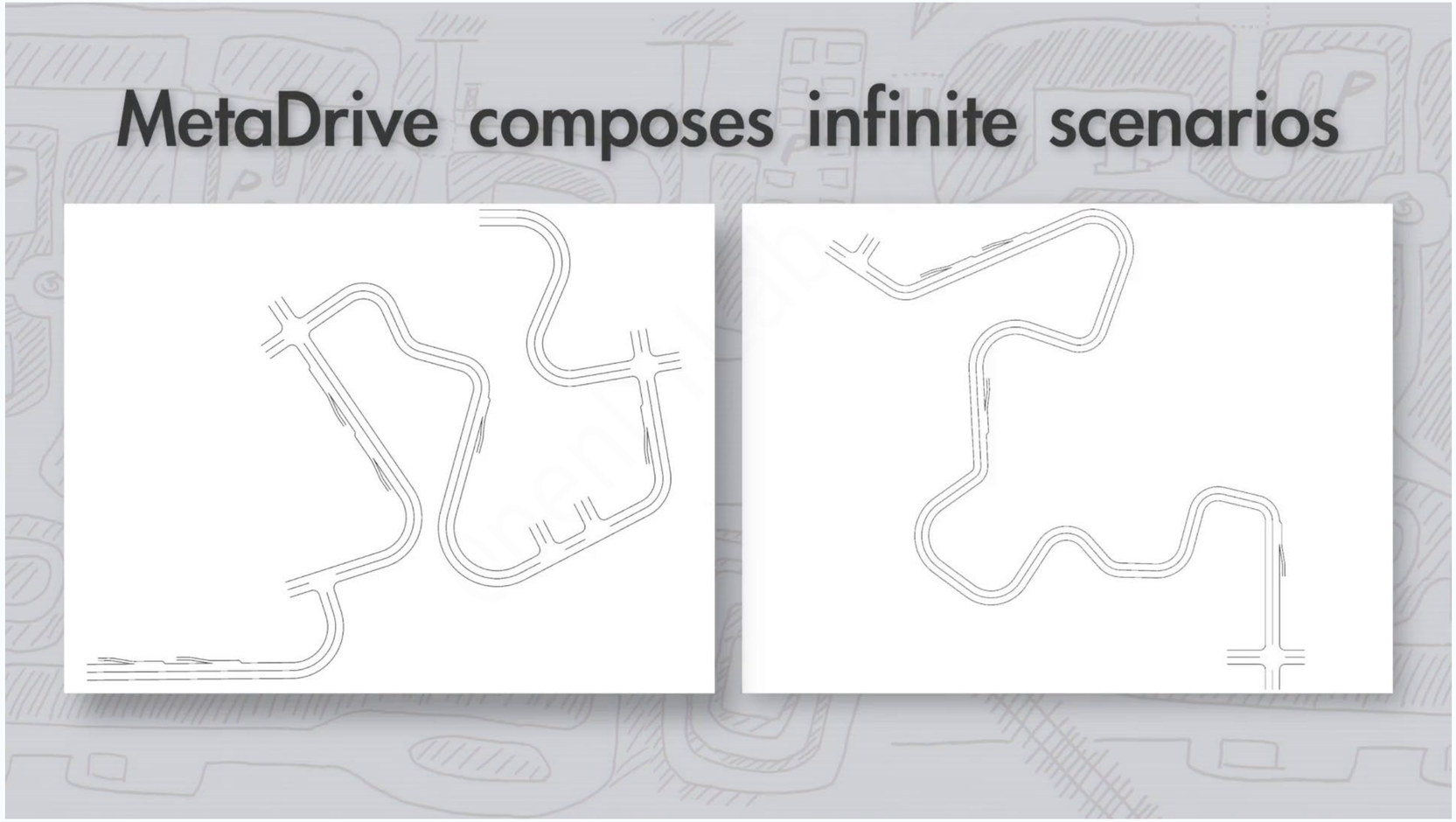
$$h^{-1}(x) = \text{sign}(x)((\frac{\sqrt{1+4\epsilon(|x|+1+\epsilon)}-1}{2\epsilon})^2 - 1)$$

```python
    return torch.sign(x) * (((torch.sqrt(1 + 4 * eps * (torch.abs(x) + 1 + eps)) - 1) / (2 * eps)) ** 2 - 1)


```

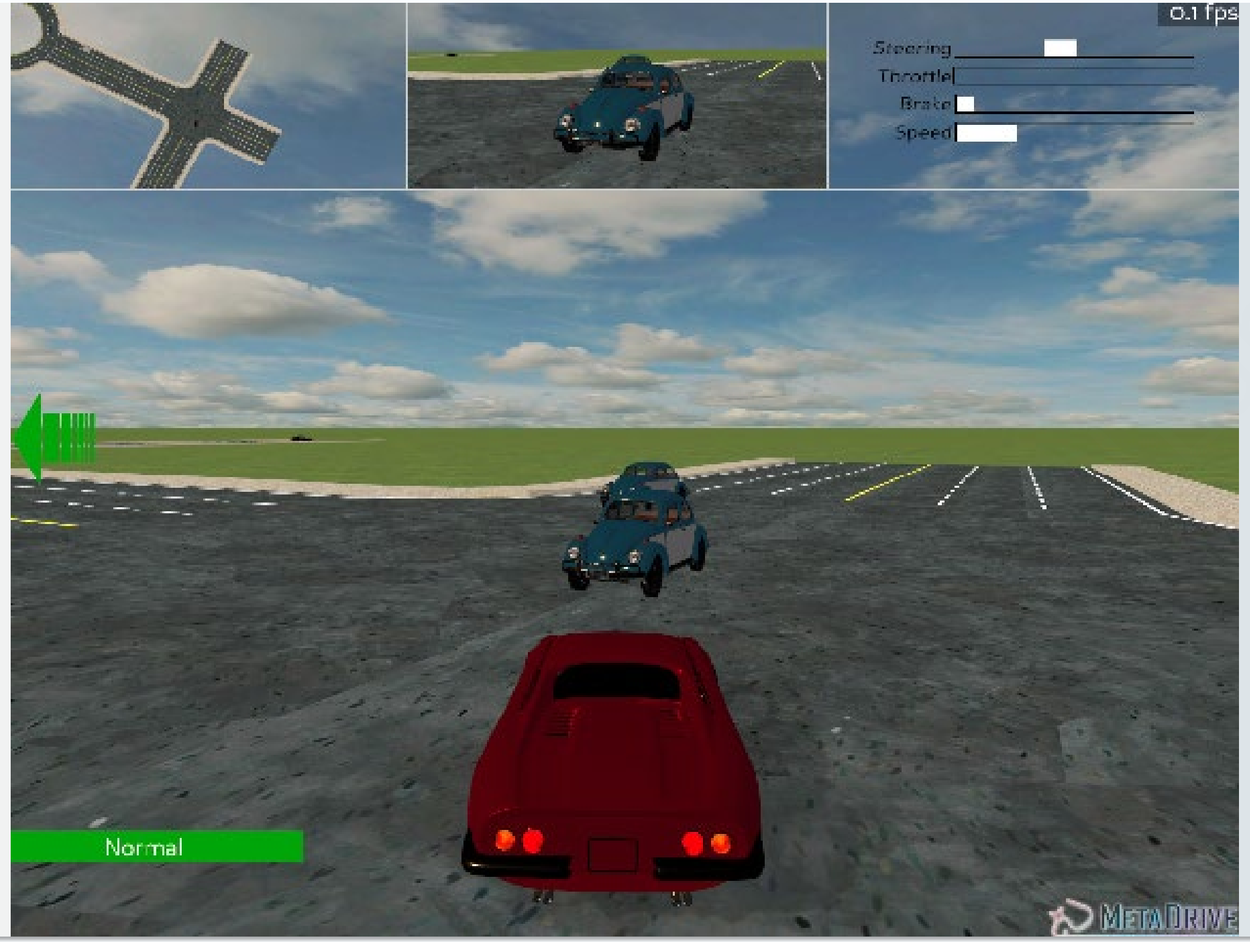
完整示例：https://opendilab.github.io/PPOxFamily/

实践：PPO+metadrive



metadrive: https://github.com/metadriverse/metadrive

实践：PPO+metadrive



动作

- 2维连续动作，转向角和加速度

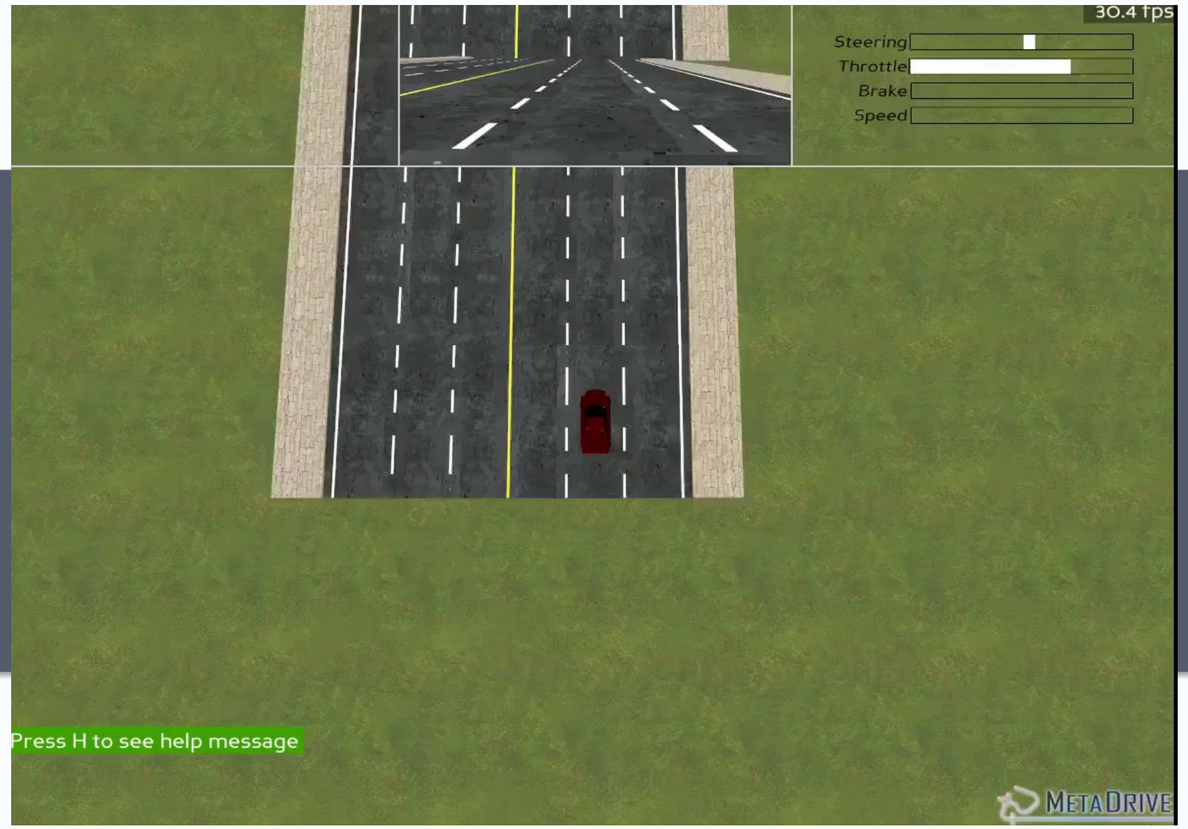
奖励

- 驾驶奖励（行驶距离）
- 速度奖励（驾驶高效且稳定）
- 横向比例（车道控制）
- 终止奖励（终点 or 撞车）

DI-engine + metadrive: https://di-engine-docs.readthedocs.io/zh_CN/latest/13_envs/metadrive_zh.html

实践：PPO+metadrive

失败样例



成功样例

视频样例：https://github.com/opendilab/PPOxFamily/issues/44

下节预告

（五）决策之链：探索时序建模

- POMDP 的定义与剖析
- RL + RNN 的前世今生
- RL + Transformer 的起承转合