基于 PyTorch 的深度强化学习平台设计与实现

翁家翌

清华大学计算机科学与技术系

2020 年 6 月 5 日

图 1: 使用 DQN [?] 玩 Atari 打砖块游戏

图 2: AlphaGo [?] 人机对战

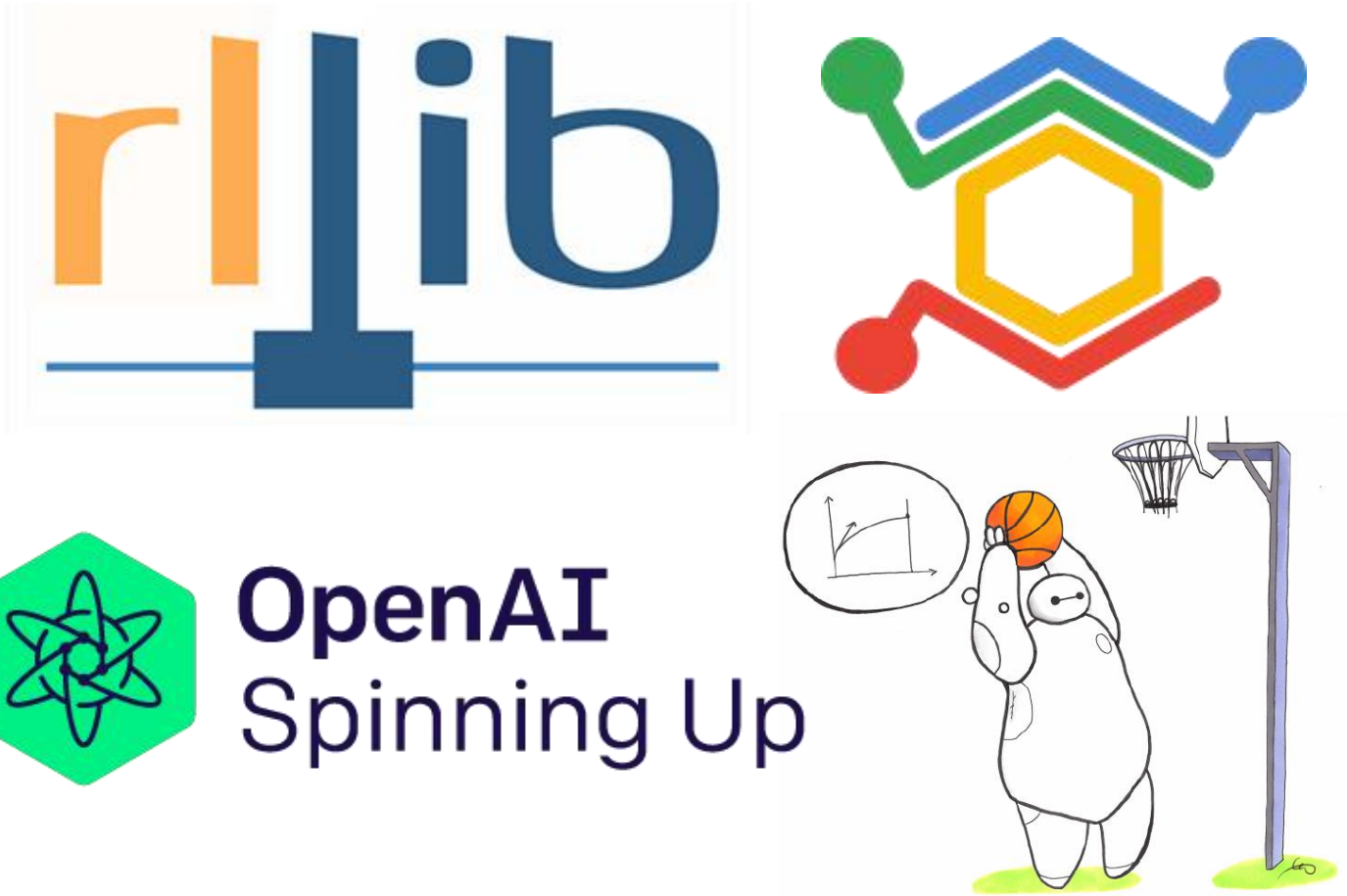


图 3: 目前较为主流的深度强化学习算法平台 [?, ?, ?, ?]

现有框架不足之处

- 算法模块化不足
- 实现算法种类有限
- 代码实现复杂度过高
- 文档不完整
- 平台性能不佳
- 缺少完整单元测试
- 环境定制支持不足

平台名称	星标数	后端框架	模块化	文档	代码质量	单元测试	上次更新
Ray/RLlib[?]	11460	TF/PyTorch	√	较全	10 / 24065	√	2020.5
Baselines[?]	9764	TF	×	无	2673 / 10411	√	2020.1
Dopamine[?]	8845	TF1	√	较全	180 / 2519	√	2019.12
SpinningUp[?]	4630	TF1/PyTorch	×	全面	1656 / 3724	×	2019.11
keras-rl[?]	4612	Keras	√	不全	522 / 2346	√	2019.11
Tensorforce[?]	2669	TF	√	全面	3834 / 13609	√	2020.5
PyTorch-DRL[?]	2424	PyTorch	√	无	2144 / 4307	√	2020.2
Stable-Baselines[?]	2054	TF1	×	全面	2891 / 10989	√	2020.5
天授	1529	PyTorch	√	全面	0 / 2141	√	2020.5
rlpyt[?]	1448	PyTorch	√	较全	1191 / 14493	×	2020.4
rlkit[?]	1172	PyTorch	√	不全	275 / 7824	×	2020.3
B-suite[?]	975	TF2	×	无	220 / 5353	×	2020.5
Garage[?]	709	TF1/PyTorch	√	不全	5 / 17820	√	2020.5

表 1: 深度强化学习平台总览，按照 GitHub 星标数从大到小排序，截止 2020/05/12

注：代码质量一栏数据格式为“PEP8 不符合规范数 / 项目 Python 文件行数”。

图 4: 天授平台标志

tianshou

An elegant, flexible, and superfast PyTorch deep Reinforcement Learning platform.

reinforcement-learning deep-learning gae pytorch dqn policy-gradient ddpg

Python MIT 235 1,594 13 0 Updated 2 hours ago

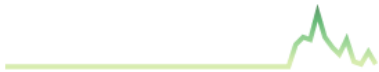


图 5: 天授平台 GitHub 信息

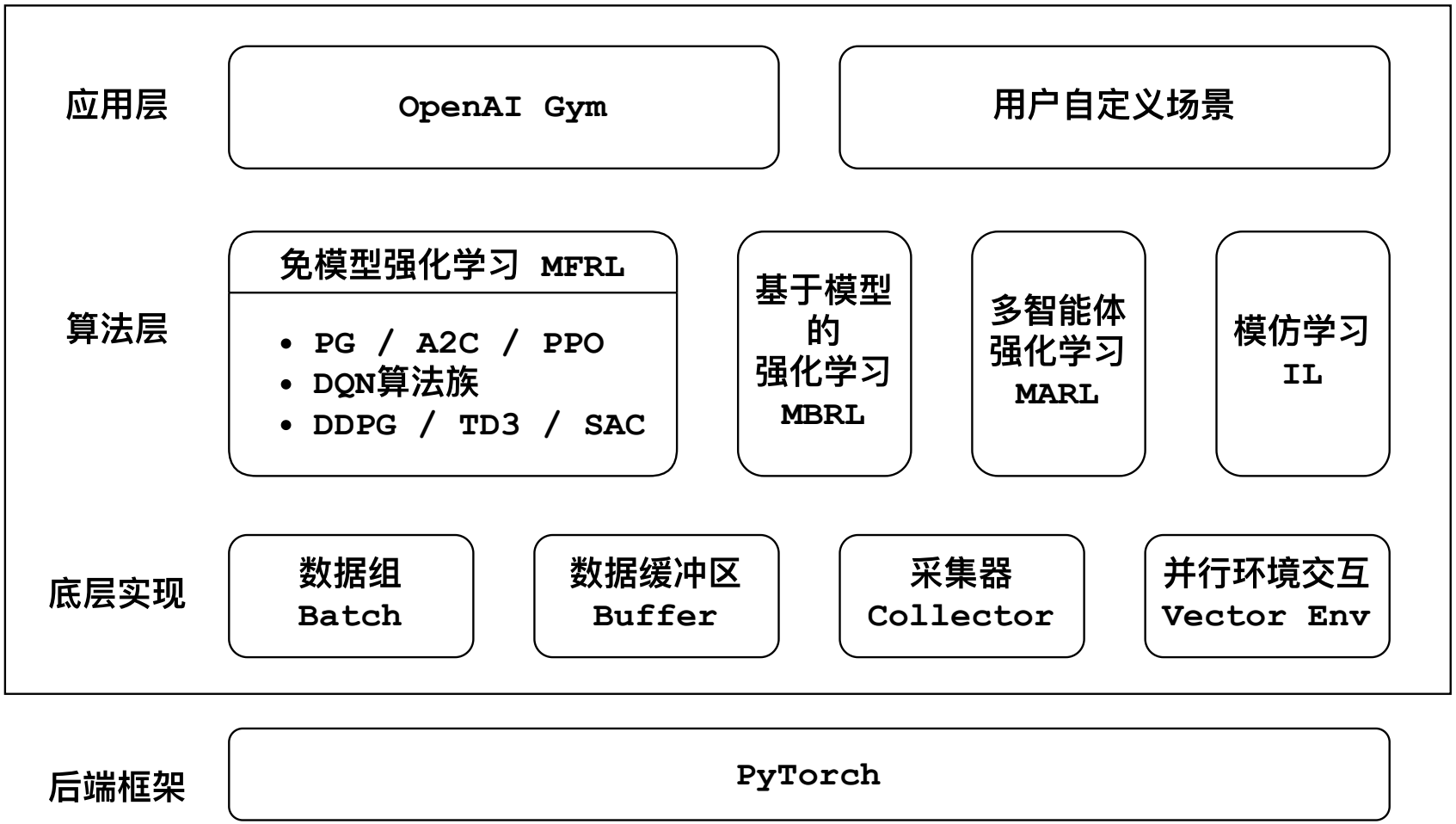

应用层
OpenAI Gym
用户自定义场景
算法层
免模型强化学习 MFRL
• PG / A2C / PPO
• DQN算法族
• DDPG / TD3 / SAC
基于模型的强化学习 MBRL
多智能体强化学习 MARL
模仿学习 IL
底层实现
数据组 Batch
数据缓冲区 Buffer
采集器 Collector
并行环境交互 Vector Env
后端框架
PyTorch

图 6: 天授平台总体架构

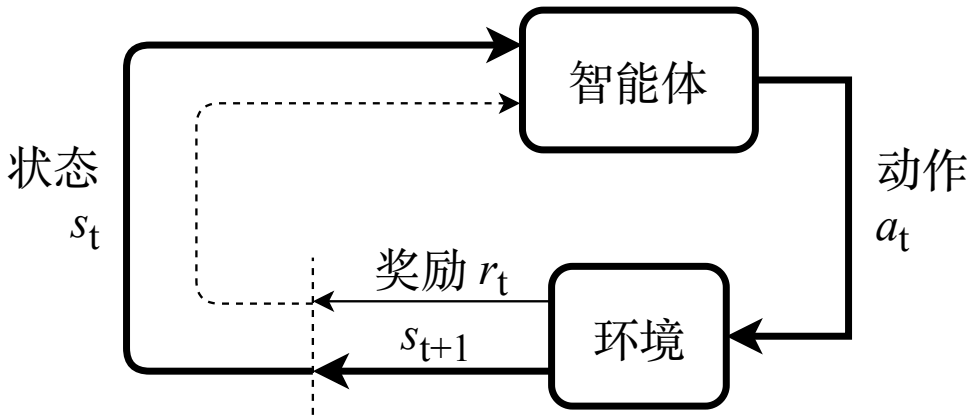


图 7: 强化学习算法中智能体与环境循环交互的过程

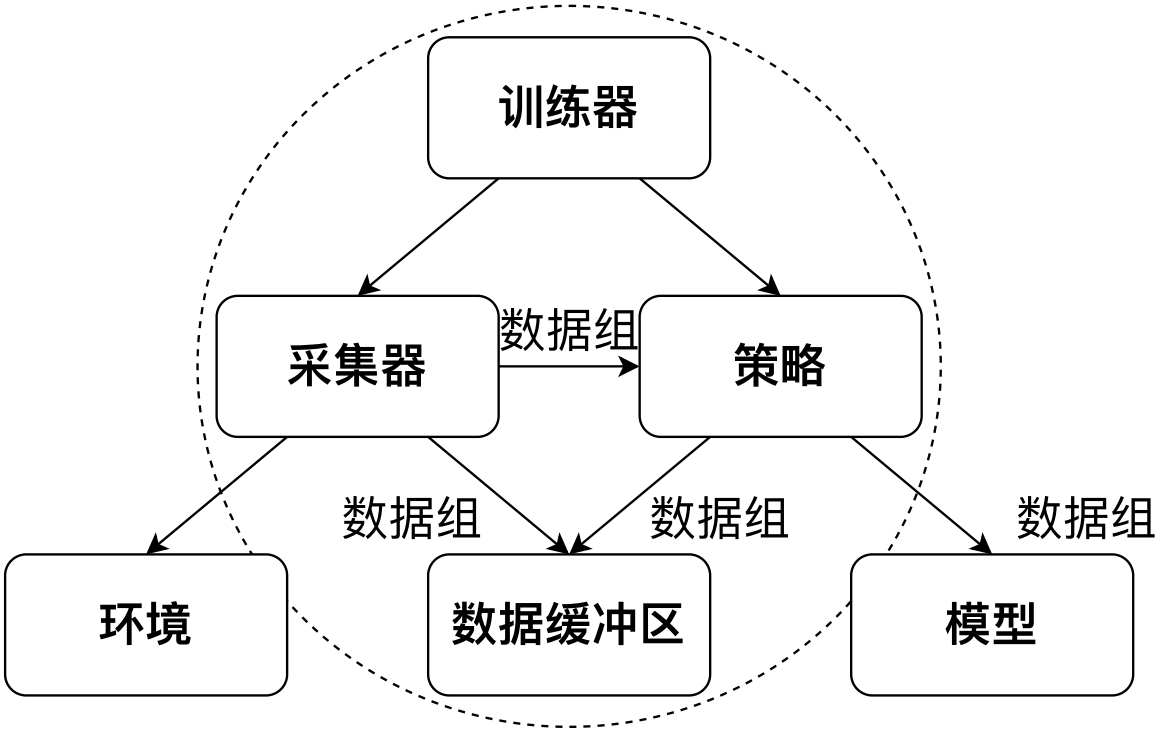


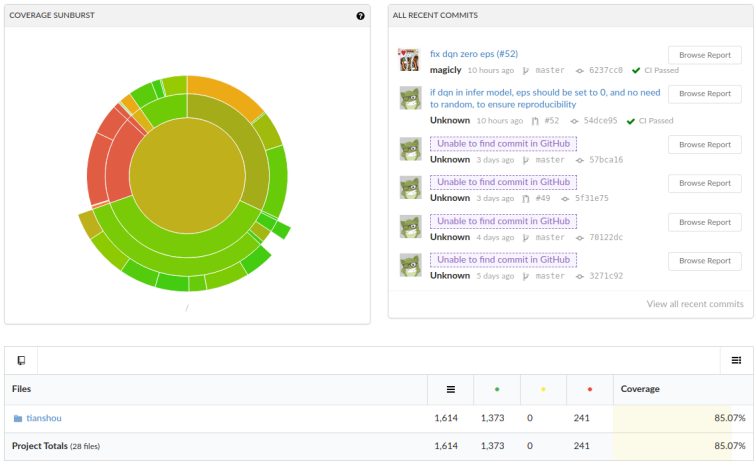
图 8: 深度强化学习算法模块抽象凝练

算法核心模块

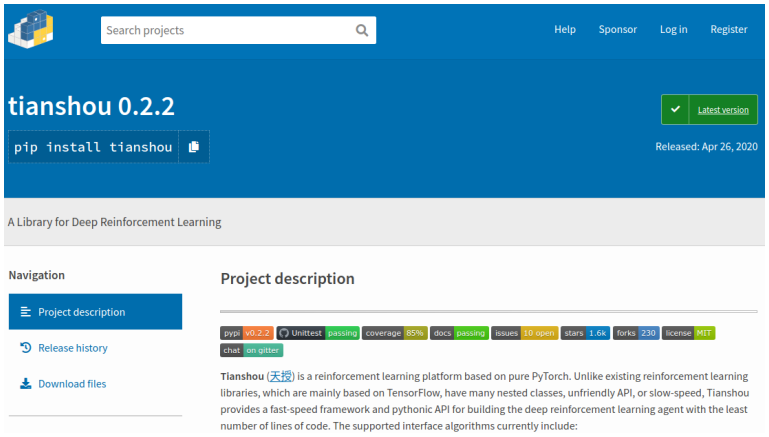
- 策略（Policy）被拆分为四部分：
  - ___init___：初始化
  - forward：计算动作
  - process_fn：训练之前与缓冲区交互
  - learn：训练策略
- 使用如上接口，可在平均 100 行代码内实现 DQN [?]、DDQN [?]、PG [?]、A2C [?]、PPO [?]、DDPG [?]、TD3 [?]、SAC [?] 等强化学习算法。

其他亮点

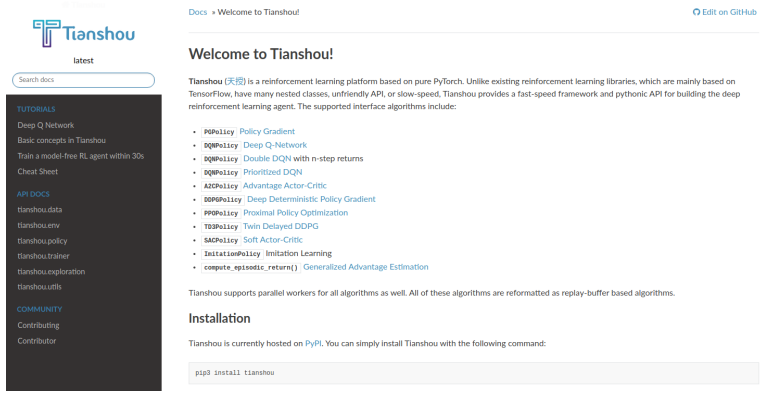
- 实现简洁，总代码行数仅两千行，方便二次开发
- 支持所有算法的并行环境采样
- 支持 RNN 在 POMDP 问题上的训练
- 支持所有 Q 学习算法的 n 步估计
- 支持任意环境状态表示



(a) 天授单元测试结果



(b) 天授在 PyPI 平台的发布界面



(c) 天授文档页面

图 9: 天授平台外围支持

算法支持

平台与算法	DQN	DDQN	PDQN	PG	A2C	PPO	DDPG	TD3	SAC	总计
RLlib	√	√	√	√	√	√	√	√	√	**9**
Baselines	√	×	√	×	√	√	√	×	×	5
PyTorch-DRL	√	√	√	√	√	√	√	√	√	**9**
SB	√	√	√	×	√	√	√	√	√	8
rlpyt	√	√	√	√	√	√	√	√	√	**9**
天授	√	√	√	√	√	√	√	√	√	**9**

表 2: 各平台支持的免模型深度强化学习算法一览

并行环境采样

平台与算法	DQN	DDQN	PDQN	PG	A2C	PPO	DDPG	TD3	SAC
RLlib	√	√	√	√	√	√	√	√	√
Baselines	×	-	×	-	√	√	√	-	-
PyTorch-DRL	×	×	×	×	√	√	×	×	×
SB	×	√	×	-	√	√	√	×	×
rlpyt	√	√	√	√	√	√	√	√	√
天授	√	√	√	√	√	√	√	√	√

表 3: 各平台各免模型深度强化学习算法支持并行环境采样情况一览

注："-" 表示算法未实现

其他类型算法

平台与算法类型	MBRL	MARL	MetaRL	IL
RLlib	√	√	×	×
Baselines	×	×	×	√
PyTorch-DRL	×	×	×	×
SB	×	×	×	√
rlpyt	×	×	×	×
天授	×	×	×	√

表 4: 各平台支持的其他类型强化学习算法一览

平台	RNN
RLlib	√
Baselines	×
PyTorch-DRL	×
SB	×
rlpyt	√
天授	√

表 5: 各平台对 RNN 的支持

模块化

平台与模块化	算法实现	数据处理	训练策略
RLlib	√	√	√
Baselines	×	×	×
PyTorch-DRL	部分模块化	×	√
SB	×	×	×
rlpyt	√	√	√
天授	√	√	部分模块化

表 6: 各平台模块化功能实现一览，其中：(1) 算法实现模块化，指实现强化学习算法的时候遵循一套统一的接口；(2) 数据处理模块化，指将内部数据流进行封装存储；(3) 训练策略模块化，指由专门的类或函数来处理如何训练强化学习智能体。

易用性

平台与易用性	代码复杂度	环境定制化	文档	教程
RLlib	250/24065	√	×	√
Baselines	110/10499	×	×	×
PyTorch-DRL	55/4366	×	×	×
SB	100/10989	×	√	√
rlpyt	243/14487	×	√	×
天授	**29/2141**	√	√	√

表 7: **各平台易用性一览，代码复杂度一栏数据格式为 Python 文件数/代码行数**

单元测试

平台与单元测试	PEP8 代码风格	基本功能	训练过程	代码覆盖率
RLlib	√	√	部分	暂缺 *
Baselines	√	√	部分	53% **
PyTorch-DRL	不遵循 + 无测试	√	完整	62% **
SB	√	√	部分	85%
rlpyt	×	部分	部分	22%
天授	√	√	完整	85%

表 8: 各平台单元测试情况一览

注：*：由于 RLlib 平台单元测试过于复杂，代码覆盖率并未集成至单元测试中，因此无法获取代码覆盖率；
**：手动在其单元测试脚本中添加代码覆盖率开启选项，并在 Travis CI 第三方测试平台中获取测试结果。

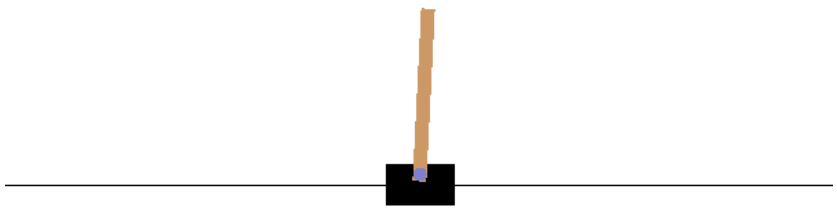


图 10: CartPole-v0 任务可视化

图 11: Pendulum-v0 任务可视化

平台与算法	PG	DQN	A2C	PPO
RLlib	19.26 ± 2.29	28.56 ± 4.60	57.92 ± 9.94	44.60 ± 17.04
Baselines	-	×	×	×
PyTorch-DRL *	×	31.58 ± 11.30	×	**23.99 ± 9.26**
SB	-	93.47 ± 58.05	57.56 ± 12.87	34.79 ± 17.02
rlpyt	**	**	**	**
天授	**6.09 ± 4.60**	**6.09 ± 0.87**	**10.59 ± 2.04**	31.82 ± 7.76

表 9: CartPole-v0 测试结果，运行时间单位为秒

注：“-”表示算法未实现；“×”表示五组实验完成任务平均时间超过 1000 秒或未完成任务；
*：由于 PyTorch-DRL 中并未实现专门的评测函数，因此适当放宽条件为“训练过程中连续 20 次完整游戏的平均总奖励大于等于 195”；
**：rlpyt 对于离散动作空间非 Atari 任务的支持不友好，可参考 https://github.com/astooke/rlpyt/issues/135。

平台与算法	PPO	DDPG	TD3	SAC
RLlib	123.62 ± 44.23	314.70 ± 7.92	149.90 ± 7.54	97.42 ± 4.75
Baselines	745.43 ± 160.82	×	-	-
PyTorch-DRL *	**	59.05 ± 10.03	57.52 ± 17.71	63.80 ± 27.37
SB	259.73 ± 27.37	277.52 ± 92.67	99.75 ± 21.63	124.85 ± 79.14
rlpyt	***	123.57 ± 30.76	113.00 ± 13.31	132.80 ± 21.74
天授	**16.18 ± 2.49**	**37.26 ± 9.55**	**44.04 ± 6.37**	**36.02 ± 0.77**

表 10: Pendulum-v0 测试结果，运行时间单位为秒

注："-" 表示算法未实现；"×" 表示五组实验完成任务平均时间超过 1000 秒或未完成任务；
*：由于 PyTorch-DRL 中并未实现专门的评测函数，因此适当放宽条件为 "训练过程中连续 20 次完整游戏的平均总奖励大于等于-250"；
**：PyTorch-DRL 中的 PPO 算法在连续动作空间任务中会报异常错误；
***：rlpyt 并未提供使用 PPO 算法的任何示例代码，经尝试无法成功跑通。

- 深度强化学习算法平台“天授”
  - 支持了诸多主流的强化学习算法
  - 支持各种不同的环境的并行采样、数据存储、定制化
  - 模块化、实现简洁、可复现性、接口灵活、速度快
- 未来工作
  - 添加更多强化学习算法
  - 加入更多种类的环境并行接口
  - 完善教程
  - 提供更多任务上调优过的示例代码

Thanks!