强化学习“探索与利用”方法研究

报告人　洪星星
导　师　李文新

北京大学博士生综合考试

2020年10月24日

目录

1. 背景知识
2. 强化学习算法
3. 研究动机
4. “探索与利用”概述
5. 强化学习中的“探索与利用”方法
6. 总结与展望

背景知识

MDP

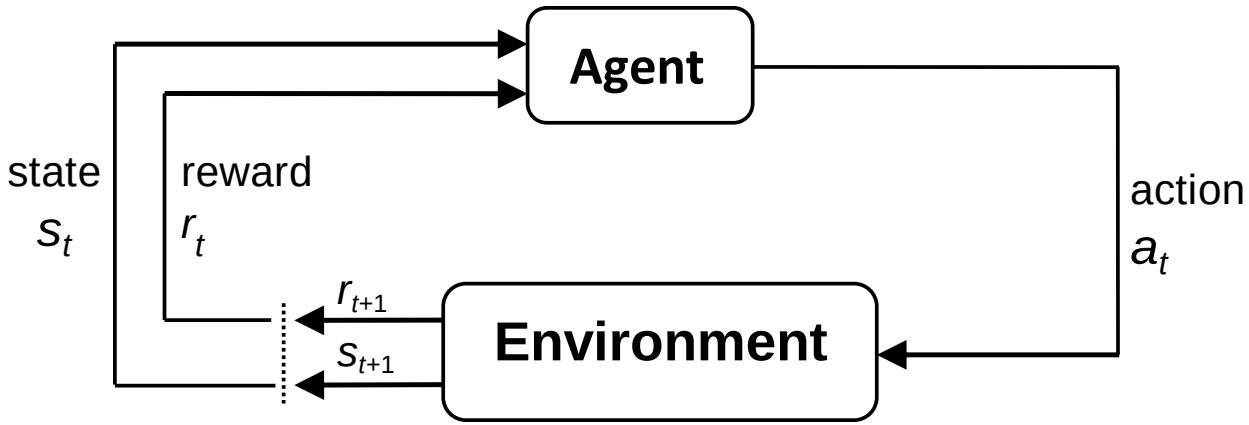


图 1: 智能体与环境交互示意

- MDP定义为$\langle S, A, P, R, \gamma \rangle$
- S和A分别为状态和动作集合
- $P(s, a, s')$是转移概率函数
- $R(s, a)$是即时收益函数
- γ为折扣因子$(0 < \gamma < 1)$
- 依据策略$\pi(s, a)$与环境交互

MDP

- 轨迹 $\boldsymbol{h} = [s_0, a_0, r_1, s_1, a_1, r_2, \ldots, s_t, a_t, r_{t+1}, \ldots]$
- 累计收益 $G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$
- 状态价值函数 $V^{\pi}(s) = \mathbb{E}_{\pi}[G_t | s_t = s]$
- 状态-动作价值函数 $Q^{\pi}(s, a) = \mathbb{E}_{\pi}[G_t | s_t = s, a_t = a]$
- 策略评估 评估策略π下的期望累积收益$\mathbb{E}[G_t]$
- 策略优化 找最优策略π^*以最大化期望累积收益$\mathbb{E}[G_t]$

$$\pi^* = \arg\max_{\pi} \mathbb{E}[G_t]$$

策略评估

- 贝尔曼期望等式

$$V^{\pi}(s) = \Sigma_a \pi(a|s) \Sigma_{s'} P\left(s, a, s'\right)\left(R\left(s, a, s'\right) + \gamma V^{\pi}\left(s'\right)\right)$$

$$Q^{\pi}(s, a) = \Sigma_{s'} P\left(s, a, s'\right)\left(R\left(s, a, s'\right) + \gamma \Sigma_{a'} \pi\left(a'|s'\right) Q^{\pi}\left(s', a'\right)\right)$$

- $R^{\pi}_{ave}(s) = \Sigma_a \pi(a|s) \Sigma_{s'} P\left(s, a, s'\right) R\left(s, a, s'\right)$
- $T^{\pi}(s \to s') = \Sigma_a \pi(a|s) P\left(s, a, s'\right)$

Theorem

基于$V^{\pi}(s) = \Sigma_a \pi(a|s) \Sigma_{s'} P\left(s, a, s'\right)\left(R\left(s, a, s'\right) + \gamma V^{\pi}\left(s'\right)\right)$定义算子$\mathcal{T}$，可证明$\mathcal{T}$为*Banach*空间中的压缩算子

$$\mathcal{T}V \equiv R^{\pi}_{ave} + \gamma T^{\pi} V$$

策略评估

Theorem

贝尔曼期望等式递推展开，得$V^{\pi}(\vec{S})$与$Q^{\pi}(s,a)$表达式

$$
\begin{aligned}
V^{\pi}(\vec{S}) &= R^{\pi}_{ave}(\vec{S}) + (\gamma T^{\pi})^{1} \cdot R^{\pi}_{ave}(\vec{S}) + (\gamma T^{\pi})^{2} \cdot R^{\pi}_{ave}(\vec{S}) + \dots \\
&= \sum_{k=0}^{\infty} (\gamma T^{\pi})^{k} \cdot R^{\pi}_{ave}(\vec{S}) \\
&= \sum_{k=0}^{\infty} (\gamma T^{\pi})^{(k)} \cdot R^{\pi}_{ave}(\vec{S}) \\
&= (\boldsymbol{I} - \gamma T^{\pi})^{-1} \cdot R^{\pi}_{ave}(\vec{S}) \\
Q^{\pi}(s,a) &= R(s,a) + \gamma P(s,a,\vec{S})(\boldsymbol{I} - \gamma T^{\pi})^{-1} R^{\pi}_{ave}(\vec{S})
\end{aligned}
$$

策略评估

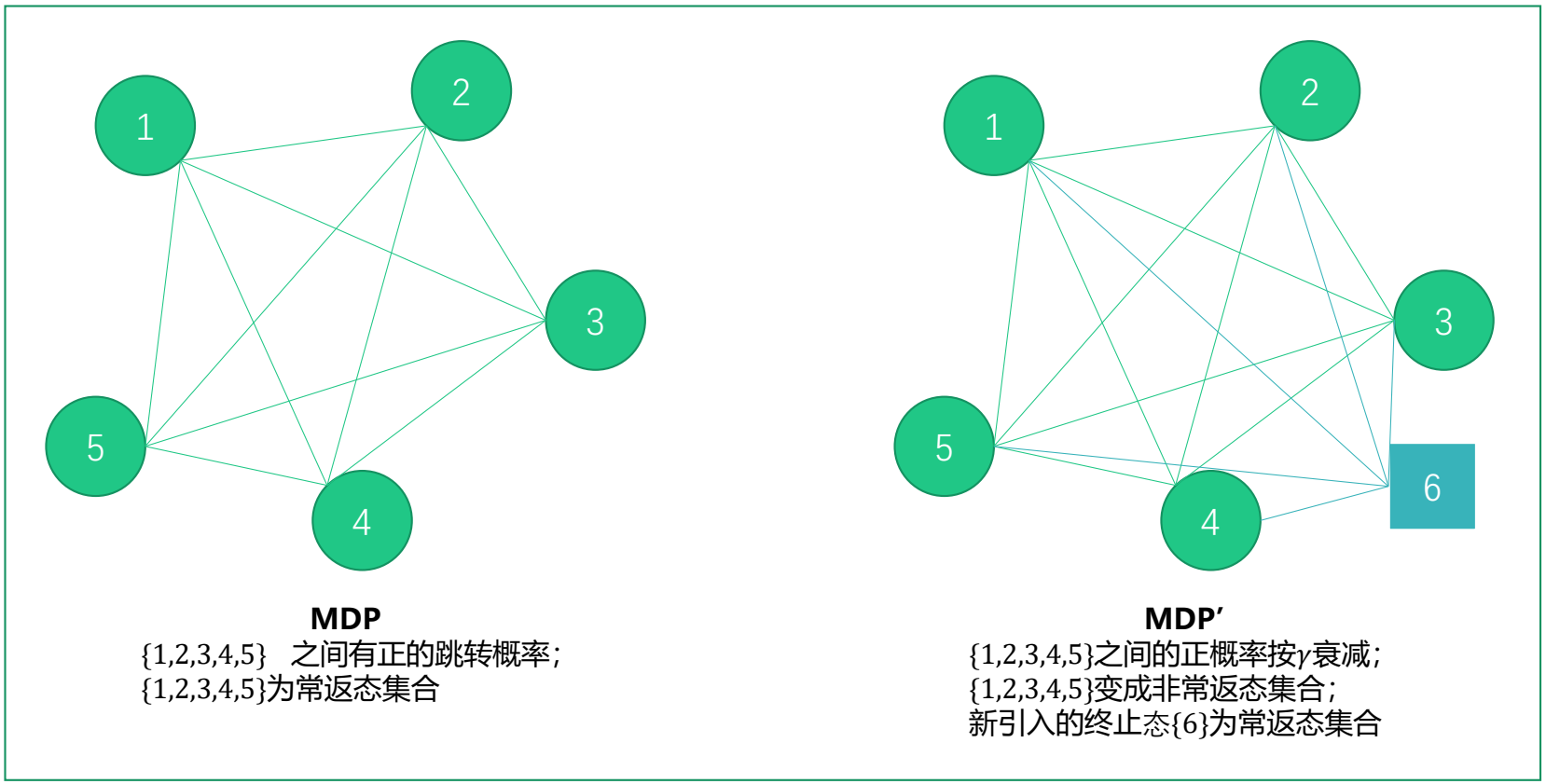


图 2: 引入$\gamma(0 < \gamma < 1)$可视作把MDP转换为MDP'的示例

策略优化

- 贝尔曼最优化等式

$$V^*(s) = \max_{\pi} V^{\pi}(s) = \max_{a} \Sigma_{s'} P(s,a,s')\left(R(s,a,s') + \gamma V^*(s')\right)$$

$$Q^*(s,a) = \max_{\pi} Q^{\pi}(s,a) = \Sigma_{s'} P(s,a,s')\left(R(s,a,s') + \gamma \max_{a'} Q^*(s',a')\right)$$

- 最优策略必为贪心选择动作导出的确定性策略，因此下式成立

$$V^*(s) = \max_{a} Q^*(s,a)$$

- 类似定义贝尔曼最优化算子具有压缩性以证明存在唯一不动点

策略优化

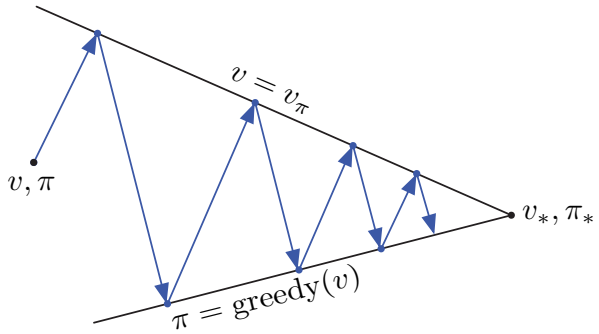


图 3: 广义策略提升框架GPI中策略评估与优化交替

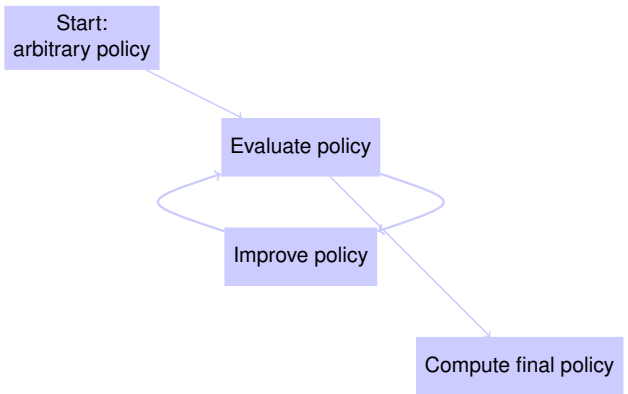


图 4: 策略迭代

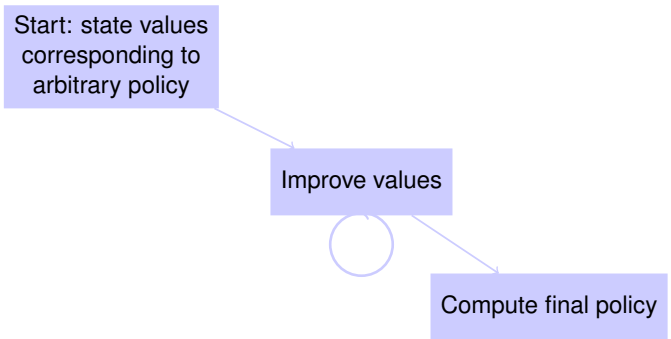


图 5: 值迭代

策略优化

策略从π提升到π'常采用greedy, $\epsilon-$greedy和softmax几种方式

- greedy

$$\pi'(a|s) = \begin{cases} 1 & a = \arg\max_a Q^\pi(s,a) \\ 0 & \text{otherwise} \end{cases}$$

- $\epsilon-$greedy

$$\pi'(a|s) = \begin{cases} \frac{\epsilon}{|A|} + 1 - \epsilon, & a = \arg\max_a Q^\pi(s,a) \\ \frac{\epsilon}{|A|}, & \text{otherwise} \end{cases}$$

- softmax

$$\pi'(a|s) = \frac{e^{\beta Q^\pi(s,a)}}{\sum_{a \in A} e^{\beta Q^\pi(s,a)}}$$

深度强化学习

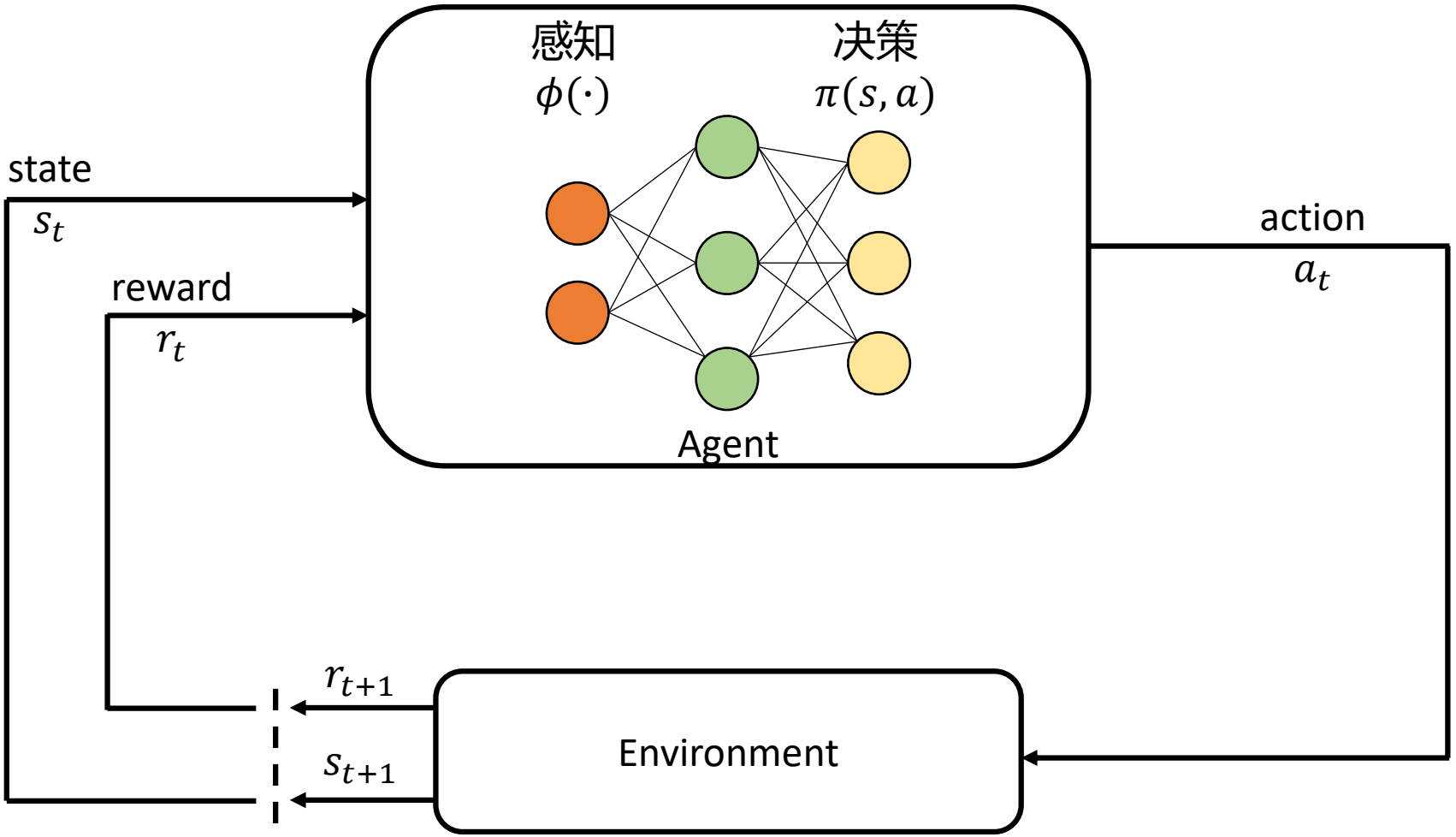
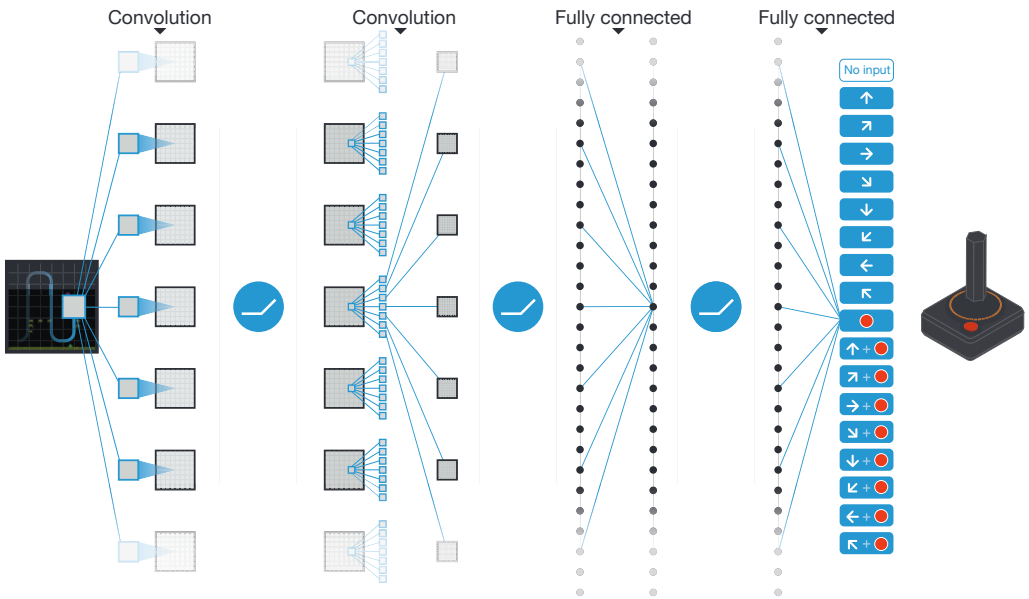


图 6: 深度强化学习智能体与环境交互示意

强化学习算法

DQN



- 深度卷积网络
- 经验回放
- 目标网络

图 7: DQN架构(Mnih et al., 2015)

$$L_i(\theta_i) = \mathbb{E}_{(s,a,r,s')\sim \mathrm{U}(D)}\left[\left(r + \gamma \max_{a'} Q\left(s', a'; \theta_i^-\right) - Q(s, a; \theta_i)\right)^2\right]$$

$$\nabla_{\theta_i} L(\theta_i) = \mathbb{E}_{s,a,r,s'}\left[\left(r + \gamma \max_{a'} Q\left(s', a'; \theta_i^-\right) - Q(s, a; \theta_i)\right) \nabla_{\theta_i} Q(s, a; \theta_i)\right]$$

DQN变种

表 1: DQN及其变种算法

算法	特点
DQN (Mnih et al., 2015)	CNN网络，卷积层感知，全连接层输出动作概率； 经验回放池机制； 目标网络以稳定训练； $Y^{DQN} \equiv r + \gamma \max_{a'} Q\left(s', a'; \theta_i^-\right)$
Double DQN (van Hasselt et al., 2016)	将选动作和评估值函数解耦，用两个网络 $Y^{DoubleDQN} \equiv r + \gamma Q\left(s', \arg\max_{a' \in A} Q(s', a', \theta_i); \theta_i^-\right)$
优先经验回放 (Schaul et al., 2016)	将相对更重要的记忆以更高频率被回放
Dueling DQN (Wang et al., 2016)	定义优势函数$A^{\pi}(s,a)$，将$Q^{\pi}(s,a)$拆解为$V^{\pi}(s)$与$A^{\pi}(s,a)$两部分
Rainbow (Hessel et al., 2018)	结合上述所有特点的方案，还包括多步、随机参数等技巧

随机策略梯度定理

$$
\begin{aligned}
\nabla V^{\pi}(s) &= \nabla\left[\sum_{a}\pi(s,a;\theta)Q^{\pi}(s,a)\right], \quad \text{对所有} s\in S \\
&= \sum_{a}\left[\nabla\pi(s,a;\theta)Q^{\pi}(s,a)+\pi(s,a;\theta)\nabla Q^{\pi}(s,a)\right] \\
&= \sum_{a}\left[\nabla\pi(s,a;\theta)Q^{\pi}(s,a)+\pi(s,a;\theta)\nabla\sum_{s'}P\left(s,a,s'\right)\left[r(s,a,s')+\gamma V^{\pi}\left(s'\right)\right]\right] \\
&= \sum_{a}\left[\nabla\pi(s,a;\theta)Q^{\pi}(s,a)+\gamma\pi(s,a;\theta)\sum_{s'}P\left(s,a,s'\right)\underbrace{\nabla V^{\pi}\left(s'\right)}_{\text{递推展开}}\right] \\
&= \sum_{a}\Bigg[\nabla\pi(s,a;\theta)Q^{\pi}(s,a)+\gamma\pi(s,a;\theta)\sum_{s'}P\left(s,a,s'\right)\times \\
&\quad \underbrace{\sum_{a'}\left[\nabla\pi(s',a';\theta)Q^{\pi}(s',a')+\gamma\pi(s',a';\theta)\sum_{s''}P(s',a',s'')\nabla V^{\pi}(s'')\right]}\Bigg] \\
&= \dots \\
&= \sum_{x}\sum_{k=0}(\gamma T^{\pi})^{(k)}_{s\to x}\sum_{a}\left[\nabla\pi(x,a;\theta)Q^{\pi}(x,a)\right] \\
&= \sum_{x}\sum_{k=0}(\gamma T^{\pi})^{(k)}_{s\to x}\sum_{a}\left[\frac{\nabla\pi(x,a;\theta)}{\pi(x,a;\theta)}\pi(x,a;\theta)Q^{\pi}(x,a)\right] \\
&= \sum_{x}\sum_{k=0}(\gamma T^{\pi})^{(k)}_{s\to x}\mathbb{E}_{\pi}\left[Q^{\pi}(x,a)\nabla\log\pi(x,a;\theta)\right]
\end{aligned}
$$

随机策略梯度定理

$$\sum_{k=0}(\gamma T^{\pi})^{(k)}_{s\to x} \approx \bar{\text{freq}}(x) = \frac{\sum_{k=0}\text{FREQ}^k(x)}{m}$$

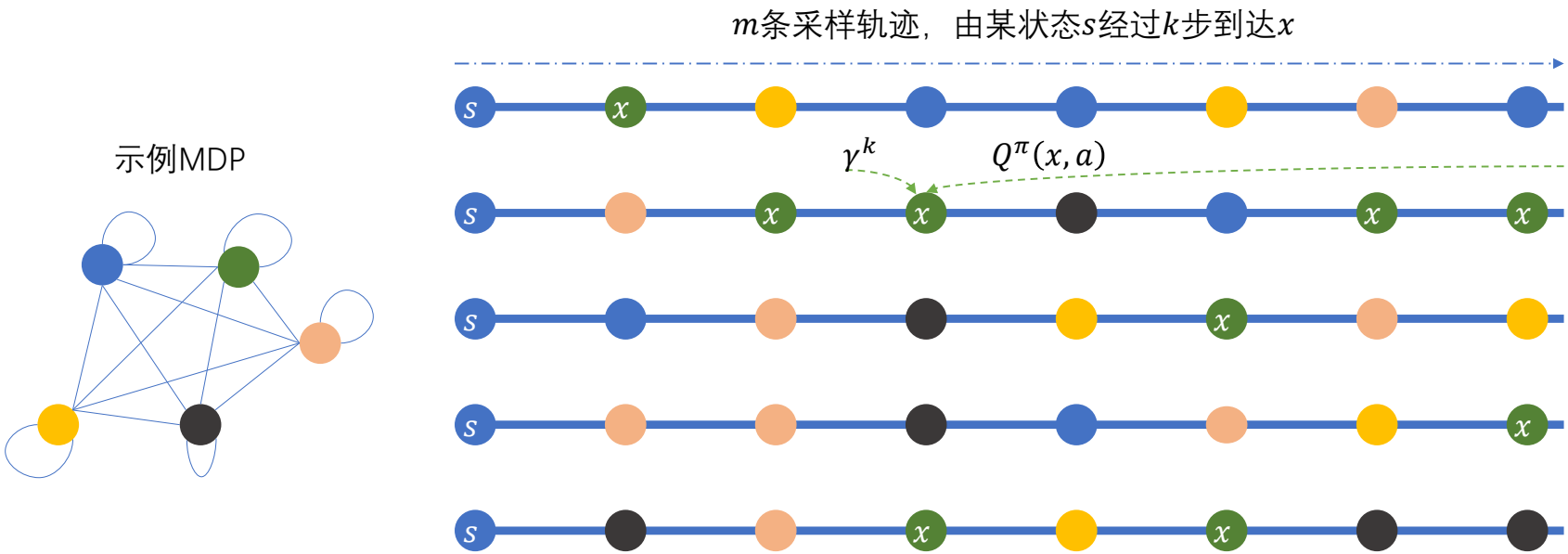


图 8: 从某一状态出发游走产生若干条轨迹

随机策略梯度定理

$$
\begin{aligned}
\nabla V^{\pi}(s) &= \sum_{x}\sum_{k=0}(\gamma T^{\pi})^{(k)}_{s\to x}\mathbb{E}_{\pi}\left[Q^{\pi}(x,a)\nabla\log\pi(x,a;\theta)\right] \\
&\approx \sum_{x}\underbrace{\bar{\text{freq}}(x)}_{\text{依据}\gamma T^{\pi}}\mathbb{E}_{\pi}\left[Q^{\pi}(x,a)\nabla\log\pi(x,a;\theta)\right] \\
&= \sum_{x}\underbrace{\frac{\sum_{k=0}\text{FREQ}^{k}(x)}{m}}_{\text{依据}\gamma T^{\pi}}\mathbb{E}_{\pi}\left[Q^{\pi}(x,a)\nabla\log\pi(x,a;\theta)\right] \\
&= \frac{1}{m}\sum_{x}\underbrace{\sum_{k=0}\text{FREQ}^{k}(x)}_{\text{依据}\gamma T^{\pi}}\mathbb{E}_{\pi}\left[Q^{\pi}(x,a)\nabla\log\pi(x,a;\theta)\right] \\
&= \frac{1}{m}\sum_{x}\sum_{k=0}\gamma^{k}\underbrace{\frac{1}{\gamma^{k}}\text{FREQ}^{k}(x)}_{\text{依据}T^{\pi}\text{从}s\text{出发经}k\text{步到}x\text{的频次}}\mathbb{E}_{\pi}\left[Q^{\pi}(x,a)\nabla\log\pi(x,a;\theta)\right] \\
&\approx \underbrace{\frac{1}{m}\sum_{i=1}^{m}\sum_{k=0}^{H_i}\left[\gamma^{k}Q(x,a)\nabla\log\pi(x,a;\theta)|s\xrightarrow{(k)}x\right]}_{\text{依据}T^{\pi}\text{且从}s\text{出发}}
\end{aligned}
$$

随机策略梯度定理

- 初始状态分布

$$
\begin{aligned}
\nabla J(\theta) &= \nabla V^{\pi}(x_0) \\
&= \sum_{x}\sum_{k=0}(\gamma T^{\pi})^{(k)}_{x_0\to x}\mathbb{E}_{\pi}\left[Q^{\pi}(x,a)\nabla \log \pi(x,a;\theta)\right] \\
&\approx \dots \\
&\approx \underbrace{\frac{1}{m}\sum_{i=1}^{m}\sum_{k=0}^{H_i}\left[\gamma^k Q(x,a)\nabla \log \pi(x,a;\theta)|x_0 \xrightarrow{(k)} x\right]}_{\text{依据}T^{\pi}\text{且}x_0\sim\text{uniform}(S)}
\end{aligned}
$$

- $\mathbb{E}_{x\sim\rho^{\pi_\theta},a\sim\pi}\left[Q^{\pi}(x,a)\nabla \log \pi(x,a;\theta)\right]$

REINFORCE等随机策略梯度算法

表 2: REINFORCE等随机策略梯度算法

算法	特点
REINFORCE	基于随机梯度定理，梯度为 $\sum_x \sum_{k=0} (\gamma T^\pi)^{(k)}_{s\to x} \mathbb{E}_\pi \left[Q^\pi(x,a) \nabla \log \pi(x,a;\theta) \right]$ 用蒙特卡洛方法估计梯度 $\frac{1}{m} \sum_{i=1}^{m} \sum_{k=0}^{H_i} \left[\gamma^k Q(x,a) \nabla \log \pi(x,a;\theta) \vert x_0 \xrightarrow{(k)} x \right]$
引入基线的REINFORCE	引入常数基数b有利于减小方差 且对策略梯度保持为无偏估计
Actor-Critic	策略函数拟合之外增加对状态-动作价值函数拟合
A2C	把Critic网络原始的累计收益替换成优势函数
A3C	多线程异步运行 全局有一套Actor神经网络参数 各环境中也存有一副本

确定性策略梯度定理

$$
\begin{aligned}
\nabla_\theta V^{\mu_\theta}(s) &= \nabla_\theta Q^{\mu_\theta}\left(s, \mu_\theta(s)\right) \\
&= \int_{\mathcal{S}} \sum_{t=0}^{\infty} \gamma^t p\left(s \to s', t, \mu_\theta\right) \nabla_\theta \mu_\theta\left(s'\right) \nabla_a Q^{\mu_\theta}\left(s', a\right)\Big|_{a=\mu_\theta(s')} \mathrm{d}s' \\
\nabla_\theta J\left(\mu_\theta\right) &= \nabla_\theta \int_{\mathcal{S}} p_1(s) V^{\mu_\theta}(s) \mathrm{d}s \\
&= \int_{\mathcal{S}} p_1(s) \nabla_\theta V^{\mu_\theta}(s) \mathrm{d}s \\
&= \int_{\mathcal{S}} \int_{\mathcal{S}} \sum_{t=0}^{\infty} \gamma^t p_1(s) p\left(s \to s', t, \mu_\theta\right) \nabla_\theta \mu_\theta\left(s'\right) \nabla_a Q^{\mu_\theta}\left(s', a\right)\Big|_{a=\mu_\theta(s')} \mathrm{d}s' \mathrm{d}s \\
&= \int_{\mathcal{S}} \rho^{\mu_\theta}(s) \nabla_\theta \mu_\theta(s) \nabla_a Q^{\mu_\theta}(s, a)\Big|_{a=\mu_\theta(s)} \mathrm{d}s
\end{aligned}
$$

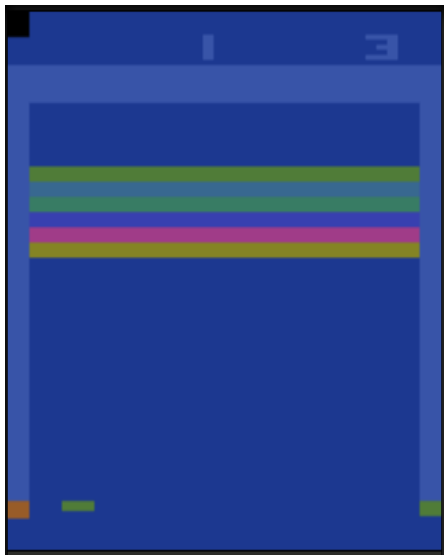
DDPG等确定性策略梯度算法

表 3: DDPG等确定性策略梯度算法

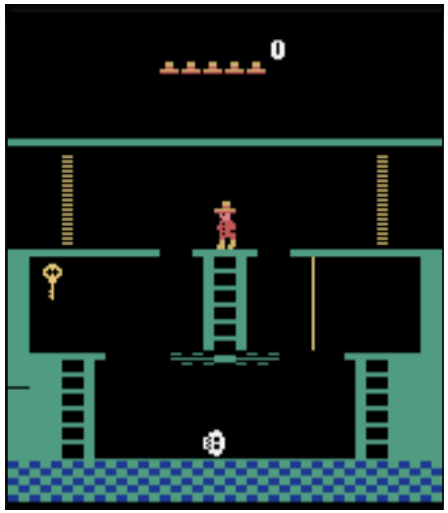
算法	特点
DPG	基于确定性策略梯度定理 状态映射到动作值而非动作概率分布$a = \mu_\theta(s)$ $\int_{\mathcal{S}} \rho^{\mu_\theta}(s) \nabla_\theta \mu_\theta(s) \nabla_a Q^{\mu_\theta}(s,a)\big\|_{a=\mu_\theta(s)} \mathrm{d}s$ $\theta^{k+1} = \theta^k + \alpha \mathbb{E}_{s\sim\rho^\mu}\left[\nabla_\theta \mu_\theta(s) \nabla_a Q^{\mu^k}(s,a)\big\|_{a=\mu_\theta(s)}\right]$
异策略DPG	利用行为策略的样本辅助训练 解决确定性策略探索性不足的问题
DDPG	借用DQN两个机制：经验回放和独立目标网络 目标网络的参数采用soft更新模式

研究动机

硬探索(Hard Exploration Problem)



Breakout



Montezuma's revenge

Pitfall

图 9: 三种Atari游戏

样本效率(Sample Inefficiency)

- 1000万游戏帧训练DQN

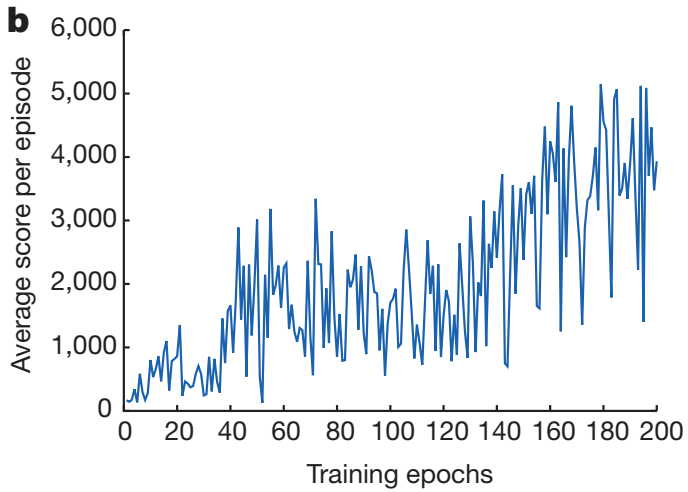


图 10: Seaquestc(Mnih et al., 2015)

- 490万局围棋对局

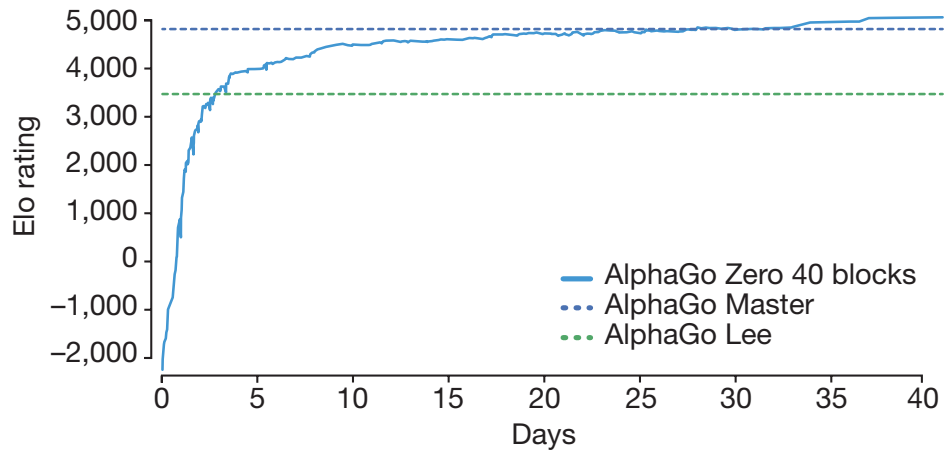
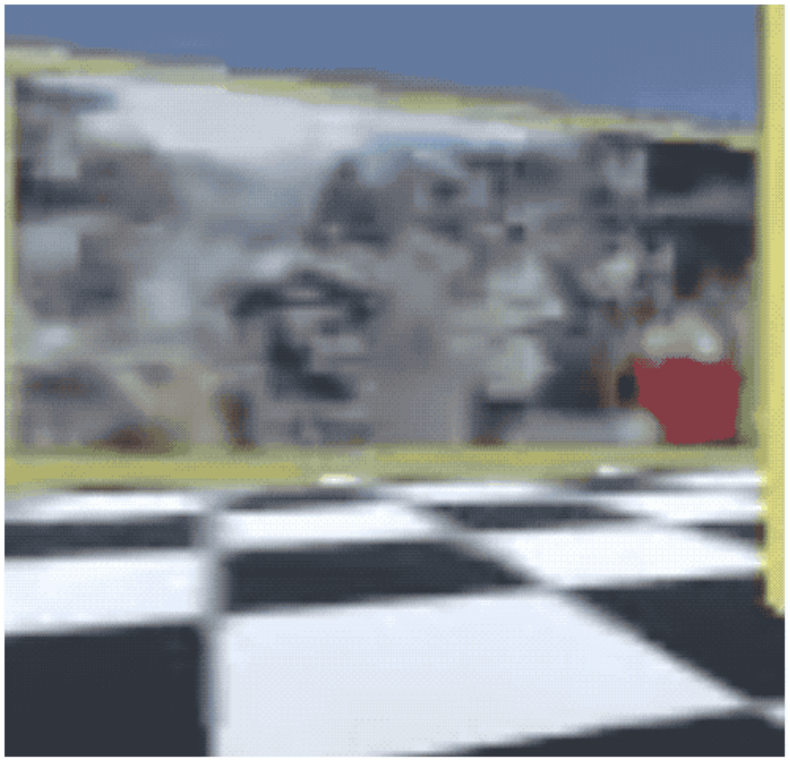
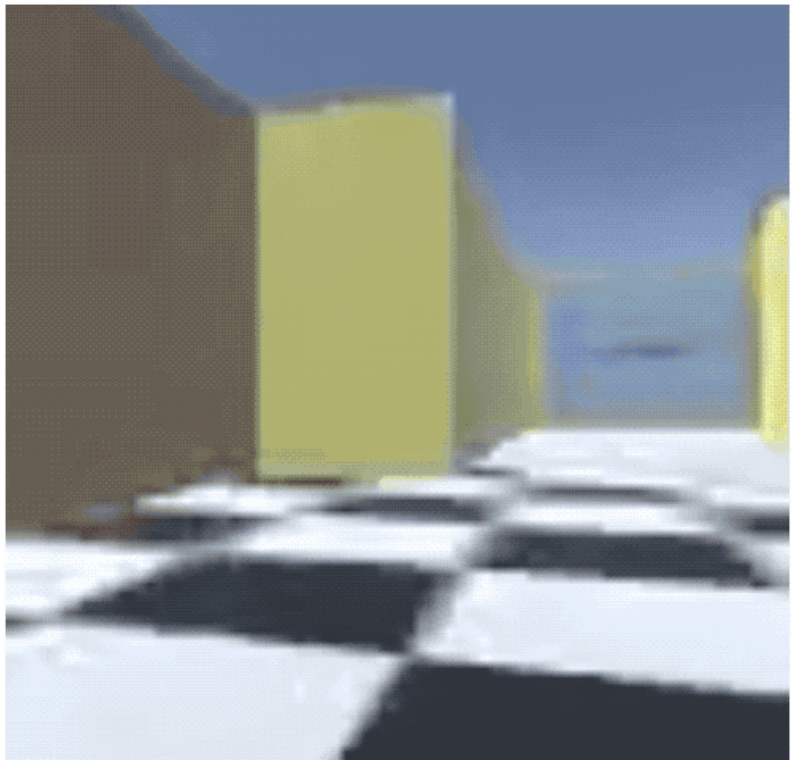


图 11: AlphaGo(Silver et al., 2017)

环境噪声(Noisy-TV Problem)



Agent in a maze with a noisy TV



Agent in a maze without a noisy TV

图 12: 迷宫墙面上的噪声电视(Noisy-TV Problem)(Burda et al., 2019)

“探索与利用”概述

探索与利用

- 探索问题
  - 智能体如何发现高收益策略，往往最优策略由时序上一系列复杂行动构成，且很可能单独考察每个行动并无正收益?
  - 智能体该如何决定：是该尝试新行动来发现更高收益，抑或用已知能够获取高收益的行动来继续应对?
- 两种权衡
  - 探索与利用困境(Exploration and Exploitation Dilemma)
  - 为将来利用而探索(Exploration for Future Exploitation)
- 探索策略
  - 随机探索
  - 系统性探索

多臂老虎机

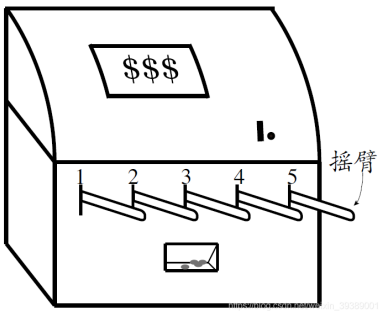


图 13: MAB

Algorithm 1: Multi-Armed Bandit (MAB) Learning

for $t = 1; \ldots; T$: **do**

Algorithm selects an arm $i_t \in [n]$;

Simultaneously, environment selects a reward vector $\mathbf{x}^t = (x_1^t, \ldots, x_n^t) \in [0, 1]^n$;

Algorithm observes reward $x_{i_t}^t$;

end

图 14: MAB Learning

基于乐观探索的方法

- 平均遗憾值函数$\mathrm{Reg}(T) = T\mu^* - \sum_{t=1}^{T} \mathbb{E}\left[\mu_{i_t}\right]$
- UCB1算法
  - $i_t \leftarrow \mathrm{argmax}_{i\in[n]} \left(\hat{\mu}_i^{t-1} + \sqrt{\frac{\alpha \ln t}{2N_i^{t-1}}} \right)$
  - $\mathrm{Reg}(T)$上界为$O(\log T)$

基于汤普森采样的方法

- 使用后验概率进行更有针对性的探索
- Beta分布为Bernoulli分布的共轭先验
  - 在先验分布为Beta分布而似然函数为Bernoulli分布时，后验概率分布仍是Beta分布

$$\theta_i \sim B(S_i + \alpha, F_i + \beta)$$

$$S_{\hat{i}} = S_{\hat{i}} + 1, \text{如果} r_{i \sim \text{Beta}} = 1$$

$$F_{\hat{i}} = F_{\hat{i}} + 1, \text{如果} r_{i \sim \text{Beta}} = 0$$

基于信息增益的方法

- 从信息增益角度提出探索策略IDS(Information Directed Sampling)

$$\begin{array}{ccccc} \mathscr{F}_t = (A_1, Y_{1,A_1}, \ldots, A_{t-1}, Y_{t-1,A_{t-1}}) \\ \downarrow \qquad \cdots \qquad \downarrow \\ R_{1,A_1} \qquad \cdots \qquad R_{t-1,A_{t-1}} \end{array}$$

- 信息增益$g_t(a) = \mathbb{E}\left[H(\alpha_t) - H(\alpha_{t+1}) \,|\mathscr{F}_t, A_t = a\right]$
  - $\alpha_t(a) = \mathbb{P}(A^* = a|\mathscr{F}_t)$为关于$A^*$的后验概率分布
- $\Delta_t(a) := \mathbb{E}\left[R_{t,A^*} - R_{t,a}|\mathscr{F}_t\right]$表示在$t$时选择$a$的即时遗憾期望
- $\Delta_t(\pi) = \sum_{a\in\mathscr{A}} \pi(a)\Delta_t(a)$
- 平均信息增益 $\pi \in \mathscr{D}(\mathscr{A}), g_t(\pi) := \Sigma_{a\in\mathscr{A}}\pi(a)g_t(a)$

基于信息增益的方法

$$\pi_t^{\mathrm{IDS}} \in \operatorname*{arg\,min}_{\pi \in \mathcal{D}(\mathscr{A})} \left\{ \Psi_t(\pi) := \frac{\Delta_t(\pi)^2}{g_t(\pi)} \right\}$$

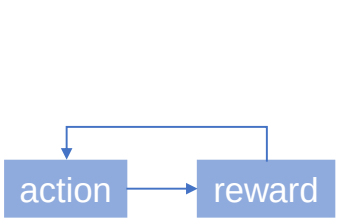
- 尽量避免那些几乎获取不到新信息的选择
- 且避免那些明显次优的选择。

探索方法的设计原则

- 在策略中增加随机扰动或考虑策略熵以增大探索性
- 对未知与不确定抱以乐观心态，选不确定性大的选项
- 认为“新”等同于“好”，给“好”状态赋以额外奖励
- 推断出概率分布，并依概率选最佳动作
- 选那些能带来新鲜感和惊讶，即能带来较大信息增益的选项等

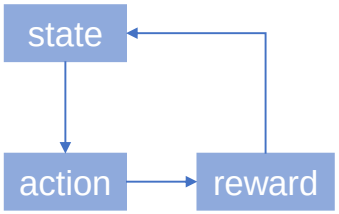
探索方法的理论分析

multi-armed bandits



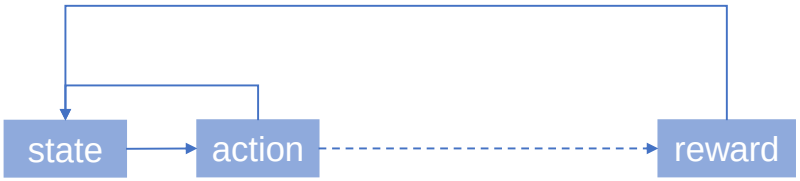
multi-armed bandits
(1-step stateless
RL problems)

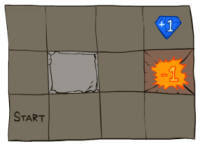
contextual bandits



contextual bandits
(1-step RL problems)

Full Reinforcement Learning Problems





small, finite MDPs
(e.g., tractable planning,
model-based RL setting)

large, infinite MDPs,
continuous spaces

theoretically tractable

theoretically intractable

图 15: 探索问题的理论分析难度随状态空间规模而变化(Levine, 2018)

强化学习中的“探索与利用”方法

强化学习中的“探索与利用”方法

- 基于好奇心驱动的探索
- 基于熵正则的探索
- 用带噪声神经网络实现探索
- 高效利用样本经验
- 其他方法及理论工作

Intrinsic Curiosity Module(ICM)

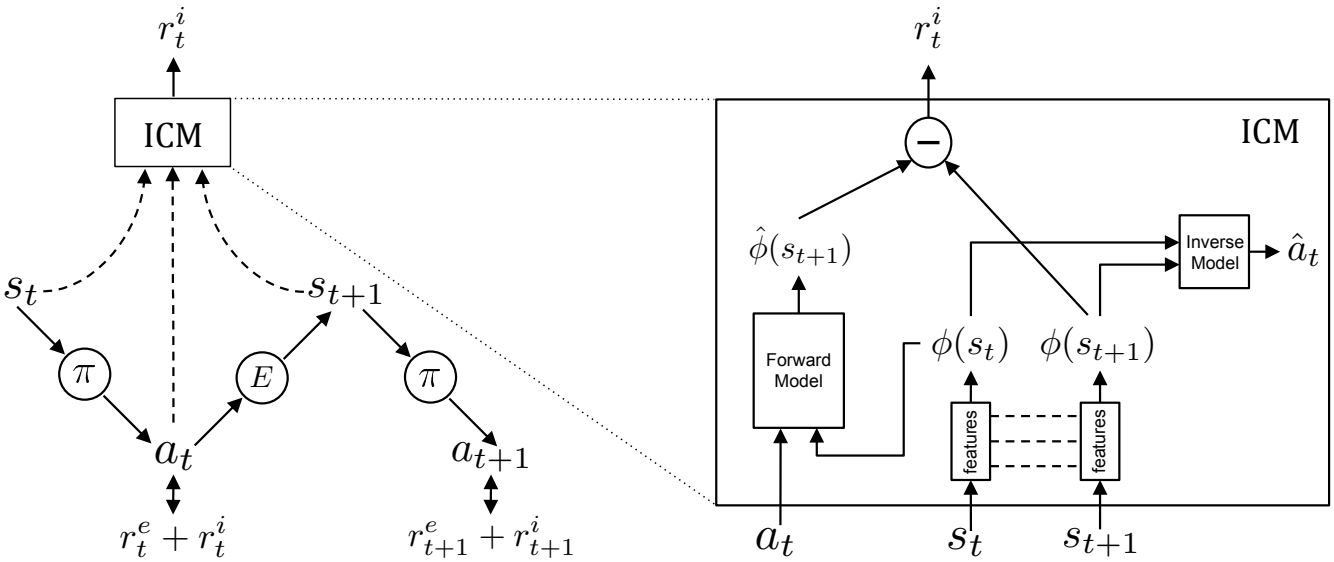


图 16: Intrinsic Curiosity Module(ICM)框架(Pathak et al., 2017)

$$L_F\left(\phi\left(s_t\right), \hat{\phi}\left(s_{t+1}\right)\right) = \frac{1}{2}\left\|f\left(\phi\left(s_t\right), a_t; \theta_F\right) - \phi\left(s_{t+1}\right)\right\|_2^2 \quad L_I\left(\hat{a}_t, a_t; \theta_I\right)$$

$$\min_{\theta_P, \theta_I, \theta_F}\left[-\lambda \mathbb{E}_{\pi\left(s_t; \theta_P\right)}\left[\Sigma_t r_t\right] + (1-\beta) L_I + \beta L_F\right]$$

Disagreement

- 多ICM集成，用模型间的不一致表达探索的额外收益

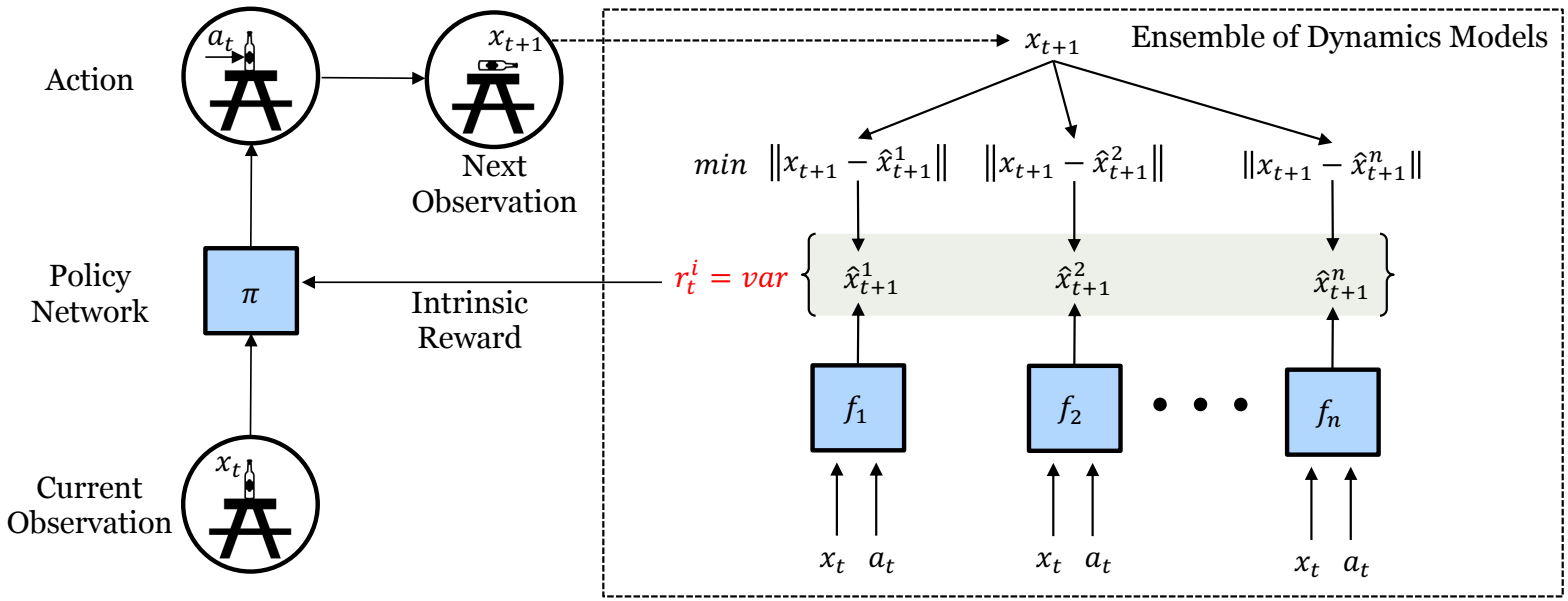


图 17: 基于不一致意见的探索模型框架(Pathak et al., 2019)

DORA

- 构造新MDP M'指引探索方向(Choshen et al., 2018)
  - 基于原MDP $M = (S, A, R, P, \gamma)$构造$M' = (S, A, 0, P, \gamma_E)$
  - 在M'上学习得到的Q值函数记做E值
  - 将M'中所有状态动作对的E值初始化为1，即$E(s, a) = 1$
  - 由于$E^*(s, a) = 0$，在训练过程中，状态动作对的$E(s, a)$将从1逐步变为0，接近真实的值函数

$$E(s, a) \leftarrow (1 - \alpha)E(s, a) + \alpha\left(0 + \gamma_E E\left(s', a'\right)\right)$$

  - 借助$\log_{1-\alpha} E$构造额外奖赏，指导动作选择

Episodic Curiosity

- 用可达性描述新颖度
- 用神经网络度量可达
- 基于可达定义增强奖赏

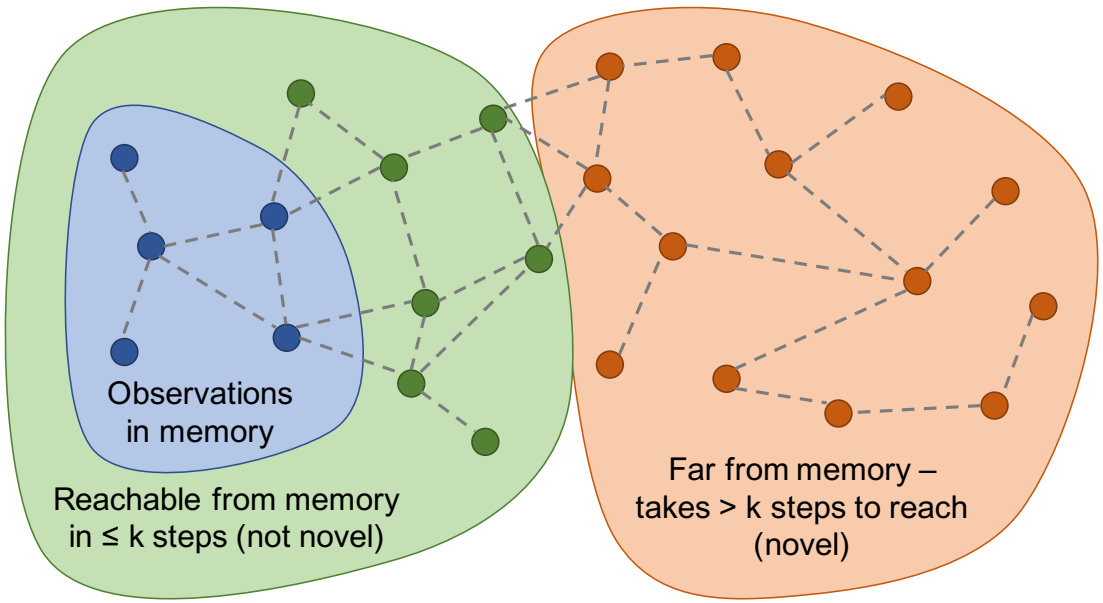


图 18: Episodic Curiosity中的新颖度(novelty)(Savinov et al., 2019)

Episodic Curiosity

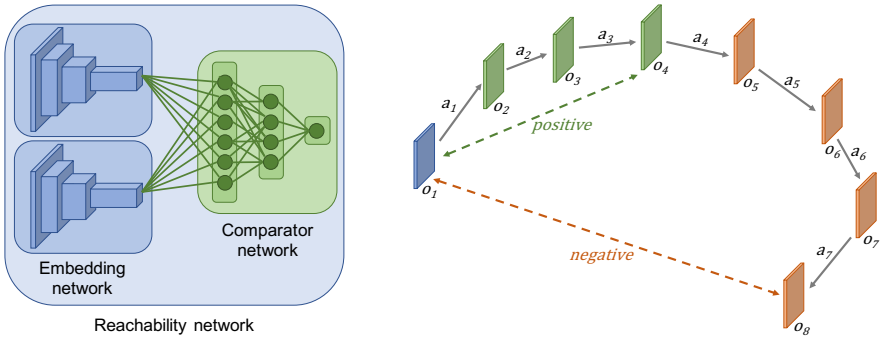


图 19: 可达性度量R-网络

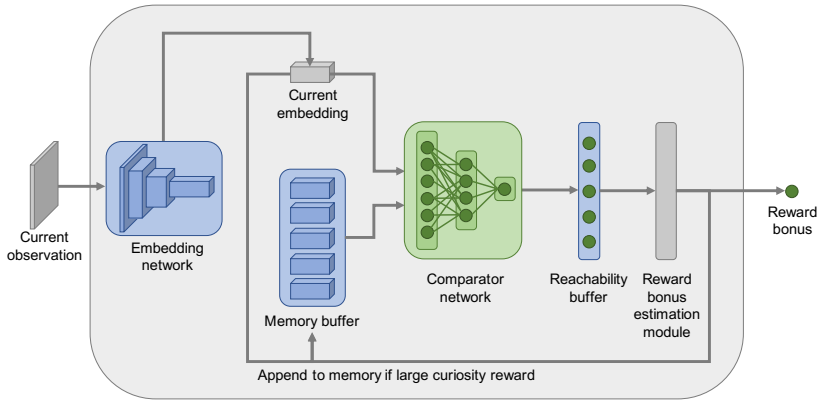


图 20: EC模型整体架构

$$\mathbf{e} = E(\mathbf{o}) \quad \text{embedding vector of } \mathbf{o}$$

$$\mathbf{M} = \langle \mathbf{e}_1, \ldots, \mathbf{e}_{|\mathbf{M}|} \rangle \quad \text{memory buffer stores } |\mathbf{M}| \text{ elements}$$

$$c_i = C(\mathbf{e}_i, \mathbf{e}), \quad i = 1, |\mathbf{M}| \quad \text{outputs of comparator network}$$

$$C(\mathbf{M}, \mathbf{e}) = F(c_1, \ldots, c_{|\mathbf{M}|}) \in [0, 1] \quad \text{aggregation function F, could be max}$$

$$b = B(\mathbf{M}, \mathbf{e}) = \alpha(\beta - C(\mathbf{M}, \mathbf{e})) \quad \text{bonus calculator}$$

Hash Based Pseudo-Counts

- 探索增强奖赏为$r^i : \mathcal{S} \mapsto \mathbb{R}$。

$$r^+(s) = \frac{\beta}{\sqrt{n(\phi(s))}}$$

- 通过$\phi(\mathbf{s})$将s压缩为k位编码，再计数$n(\phi(\mathbf{s}))$

$$\phi(s) = \text{sgn}(Ag(s)) \in \{-1, 1\}^k$$

- AutorEncoder编码更复杂或连续空间的观测

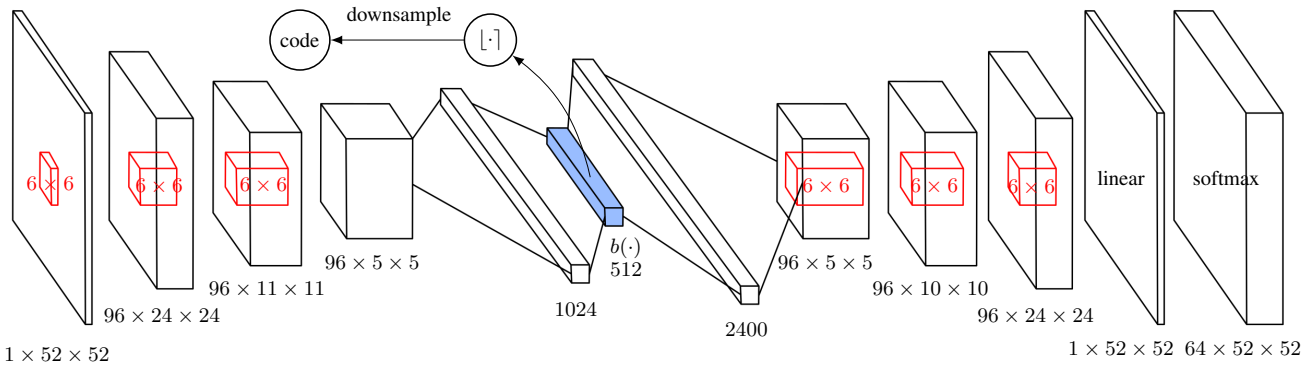


图 21: 基于自编码器的哈希编码计数方法架构(Tang et al., 2017)

Soft Q-Learning

- Q学习策略弊端
  - 连续状态、动作空间
  - 任务自适应性差

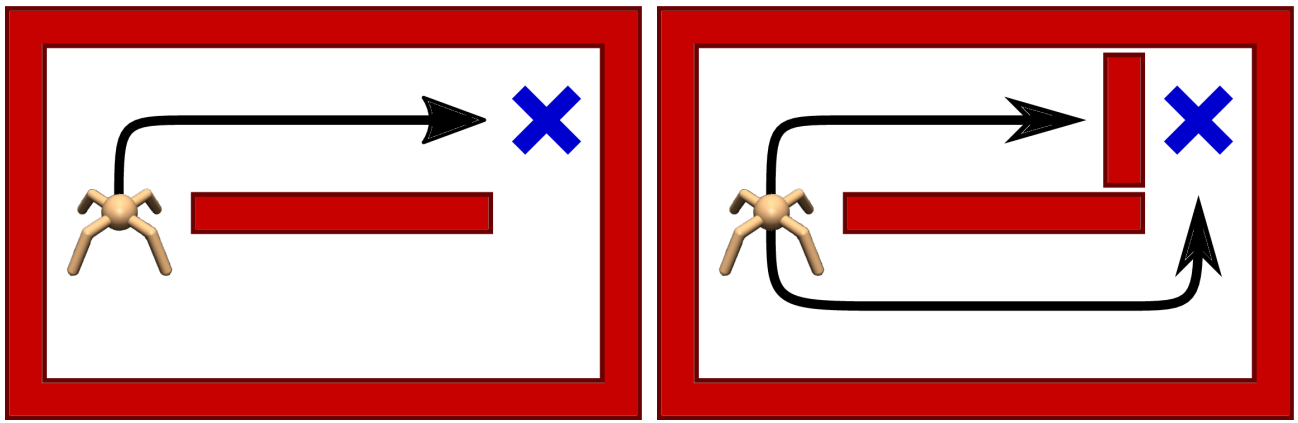


图 22: 巡航任务微调环境致最优策略彻底失效(Haarnoja et al., 2017)

Soft Q-Learning

- 引入熵正则化项并重构贝尔曼方程

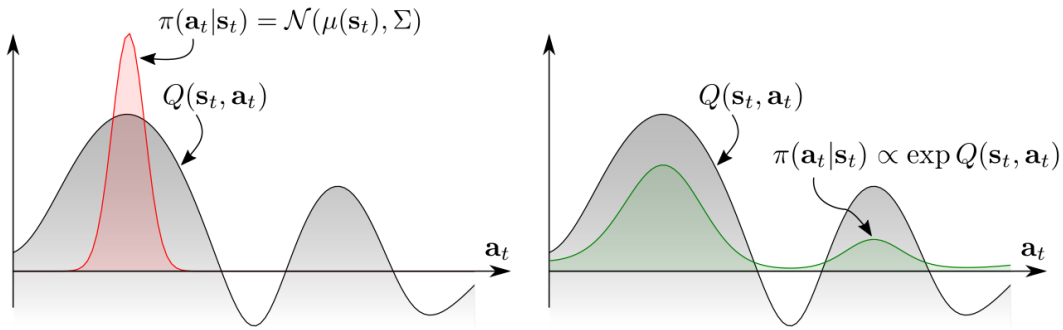


图 23: 将最优策略表达为玻尔兹曼分布

$$\pi^*_{\text{MaxEnt}} = \arg\max_{\pi} \sum_t \mathbb{E}_{(\mathbf{s}_t,\mathbf{a}_t)\sim\rho_\pi} \left[r\left(\mathbf{s}_t, \mathbf{a}_t\right) + \alpha \mathcal{H}\left(\pi\left(\cdot|\mathbf{s}_t\right)\right)\right]$$

$$Q^*_{\text{soft}}\left(\mathbf{s}_t, \mathbf{a}_t\right) = r_t + \mathbb{E}_{(\mathbf{s}_{t+1},\ldots)\sim\rho_\pi} \left[\sum_{l=1}^{\infty} \gamma^l \left(r_{t+l} + \alpha \mathcal{H}\left(\pi^*_{\text{MaxEnt}}\left(\cdot|\mathbf{s}_{t+l}\right)\right)\right)\right]$$

$$V^*_{\text{soft}}\left(\mathbf{s}_t\right) = \alpha \log \int_{\mathcal{A}} \exp\left(\frac{1}{\alpha} Q^*_{\text{soft}}\left(\mathbf{s}_t, \mathbf{a}'\right)\right) d\mathbf{a}'$$

Soft Actor-Critic

- (Haarnoja et al., 2018)引入熵正则项，重构造贝尔曼方程
  - **策略评估**

$$\mathcal{T}^{\pi} Q\left(\mathbf{s}_t, \mathbf{a}_t\right) = r\left(\mathbf{s}_t, \mathbf{a}_t\right) + \gamma \mathbb{E}_{\mathbf{s}_{t+1} \sim p}\left[V\left(\mathbf{s}_{t+1}\right)\right]$$

$$V\left(\mathbf{s}_t\right) = \mathbb{E}_{\mathbf{a}_t \sim \pi}\left[Q\left(\mathbf{s}_t, \mathbf{a}_t\right) - \log \pi\left(\mathbf{a}_t | \mathbf{s}_t\right)\right]$$

  - $Q^{k+1} = \mathcal{T}^{\pi} Q^{k}$反复迭代求得$Q$与$V$值收敛点
  - **策略提升**

$$\pi_{\text{new}} = \arg\min_{\pi' \in \Pi} \mathrm{D}_{\mathrm{KL}}\left(\pi'\left(\cdot | \mathbf{s}_t\right) \| \frac{\exp\left(Q^{\pi_{\text{old}}}\left(\mathbf{s}_t, \cdot\right)\right)}{Z^{\pi_{\text{old}}}\left(\mathbf{s}_t\right)}\right)$$

  - 引入V值目标网络、采用重参数化技巧处理梯度更新

NoisyNet

- NoisyNet(Fortunato et al., 2018)在神经网络参数中掺杂噪声以增强策略探索性

$$y = wx + b$$

$$y \overset{\text{def}}{=} \left(\mu^w + \sigma^w \odot \varepsilon^w\right) x + \mu^b + \sigma^b \odot \varepsilon^b$$

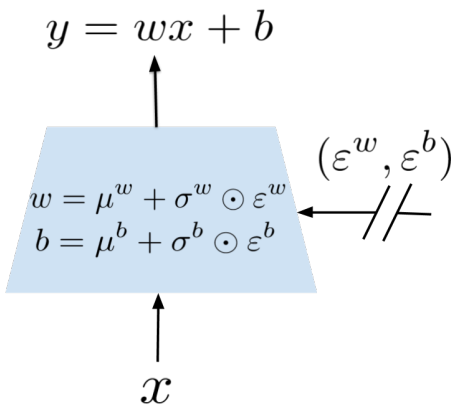


图 24: 引入随机噪声参数的线性层模型

Parameter Space Noise

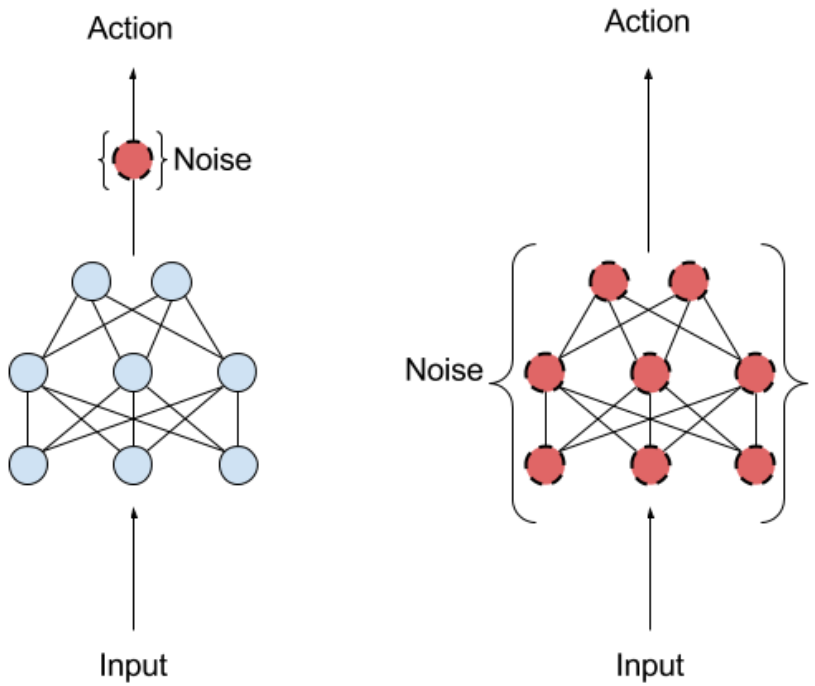


图 25: 动作空间和策略参数空间引入噪声对比(Plappert et al., 2017)

$$a_t = \pi(s_t) + \mathcal{N}\left(0, \sigma^2 I\right)$$

$$\widetilde{\theta} = \theta + \mathcal{N}\left(0, \sigma^2 I\right)$$

RND

- RND(Burda et al., 2019)引入网络蒸馏技术，Target Network有固定的随机初始化的权重，在训练中不改变，它能够为每一个状态观测输出一个特征表示
- Predictor Network将Target Network的输出结果作为label 并根据MSE损失更新参数θ

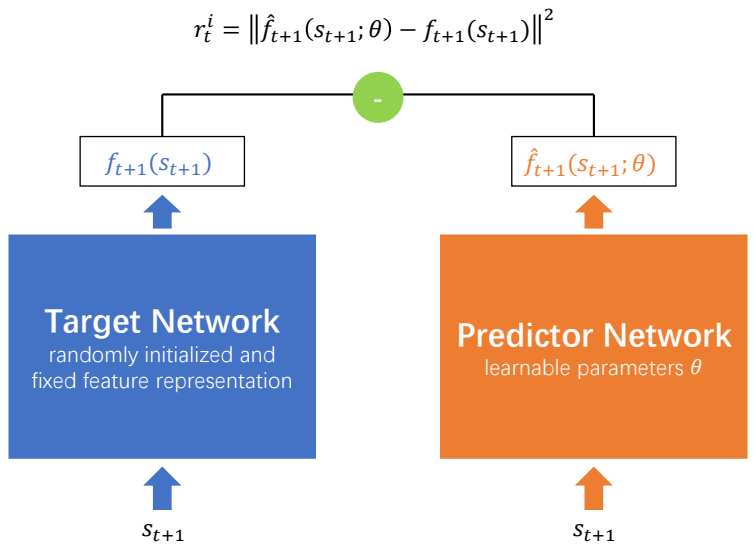


图 26: Random Network Distillation(RND)模型产生内在增强奖赏

RND

- 与ICM不同，RND框架中前向网络易于实现和计算，不以s_t,a_t作输入预测下一状态特征表示

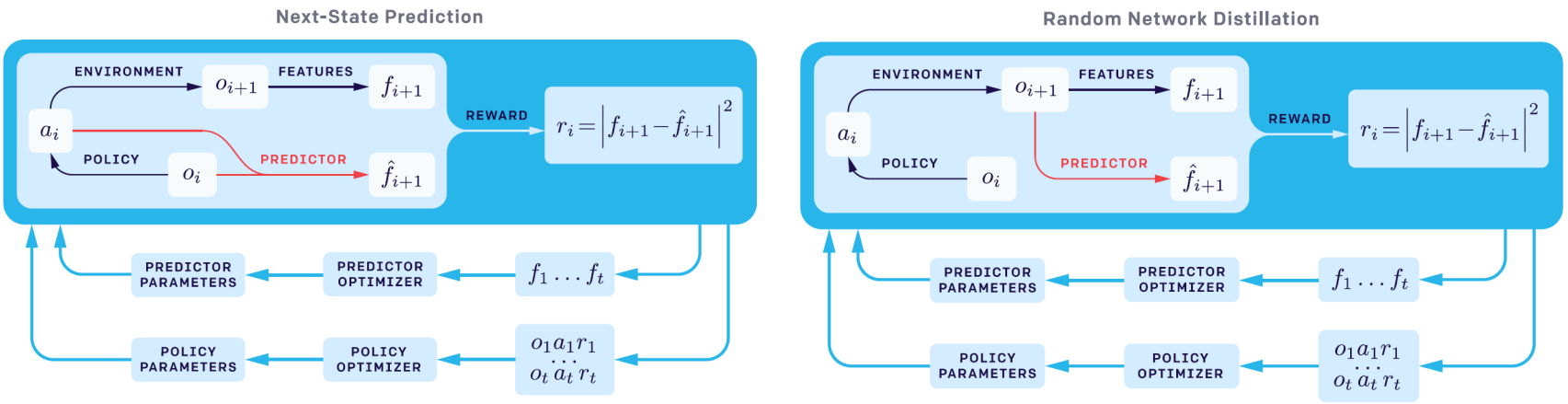


图 27: RND与ICM中前向预测模块对比

Hindsight Experience Replay

- HER(Andrychowicz et al., 2017)解决稀疏的二元奖赏任务
  - 预设目标g，轨迹序列为$s_0, s_1, s_2, \cdots, s_T \neq g$，将目标重设为$s_T$，未成功完成任务变为成功
  - 扩展目标集合$\mathcal{G}(g \in \mathcal{G})$，有利于扩充经验池中正例采样轨迹

$$a_t \leftarrow \pi_b\left(s_t \| g\right) \quad \|\text{意味着将}s_t\text{和}g\text{串接起来}$$

$$r_t := r\left(s_t, a_t, g\right) \quad g\text{随机取自}\mathcal{G}$$

$$\text{Replay Buffer} \leftarrow \left(s_t \| g, a_t, r_t, s_{t+1} \| g\right) \quad \text{串接}s_t\text{和}g\text{存入经验回放池}$$

- 课程学习(Curriculum Learning)(Bengio et al., 2009)

Go-Explore

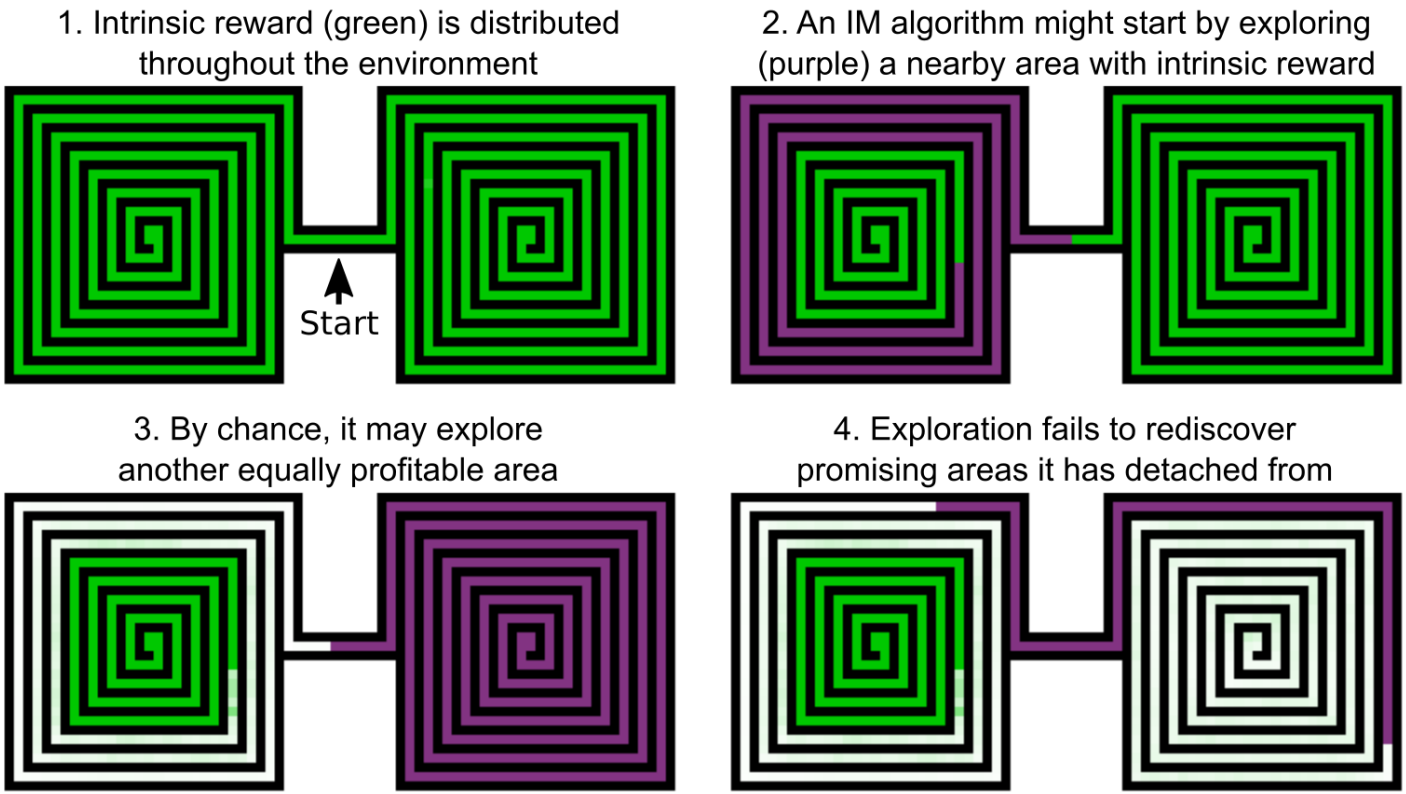


图 28: Intrinsic Motivation(IM)类方法存在的问题(Ecoffet et al., 2019)

Go-Explore

- Uber提出Go-Explore解决门特祖玛复仇问题
  - 存档探索过的路径
  - 如有必要对存档路径模仿学习
  - 图像粗粒化

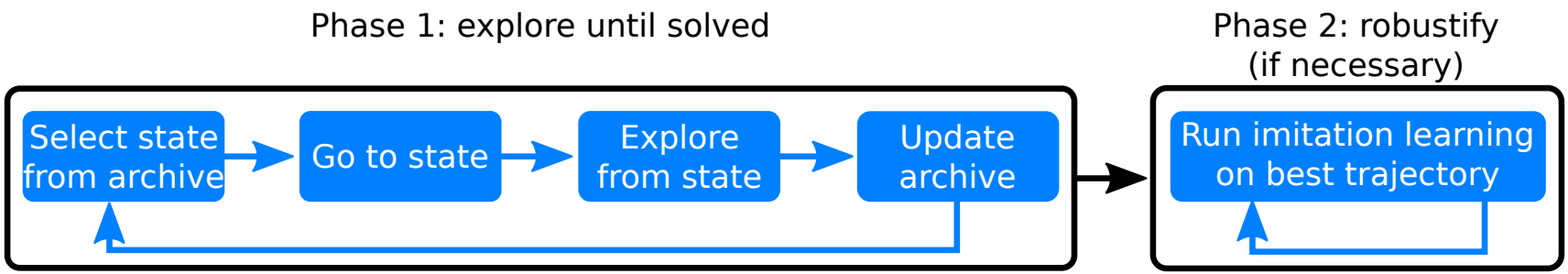
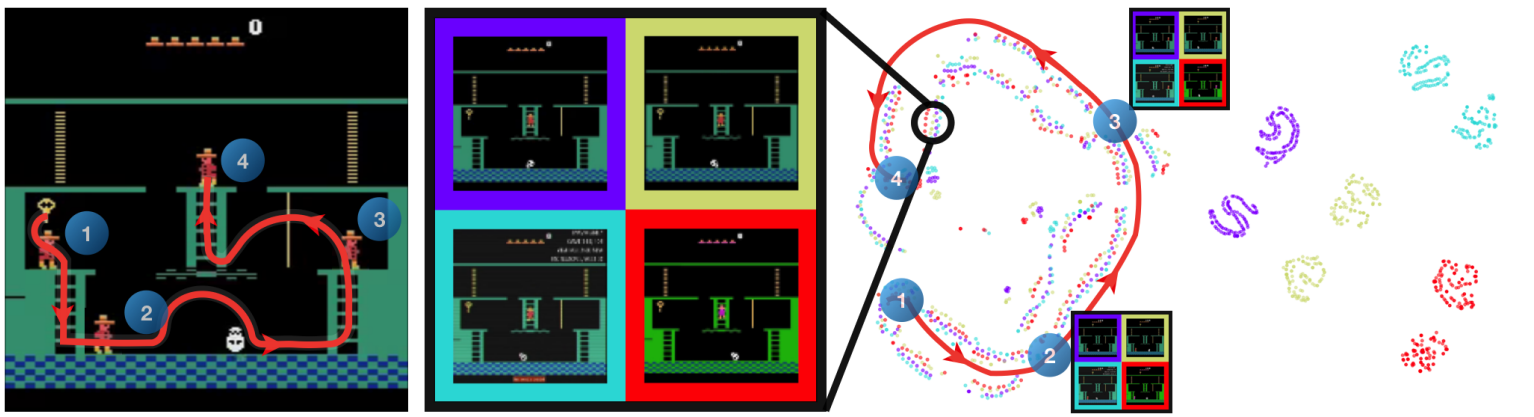


图 29: Go-Explore方法

TDC+CMC

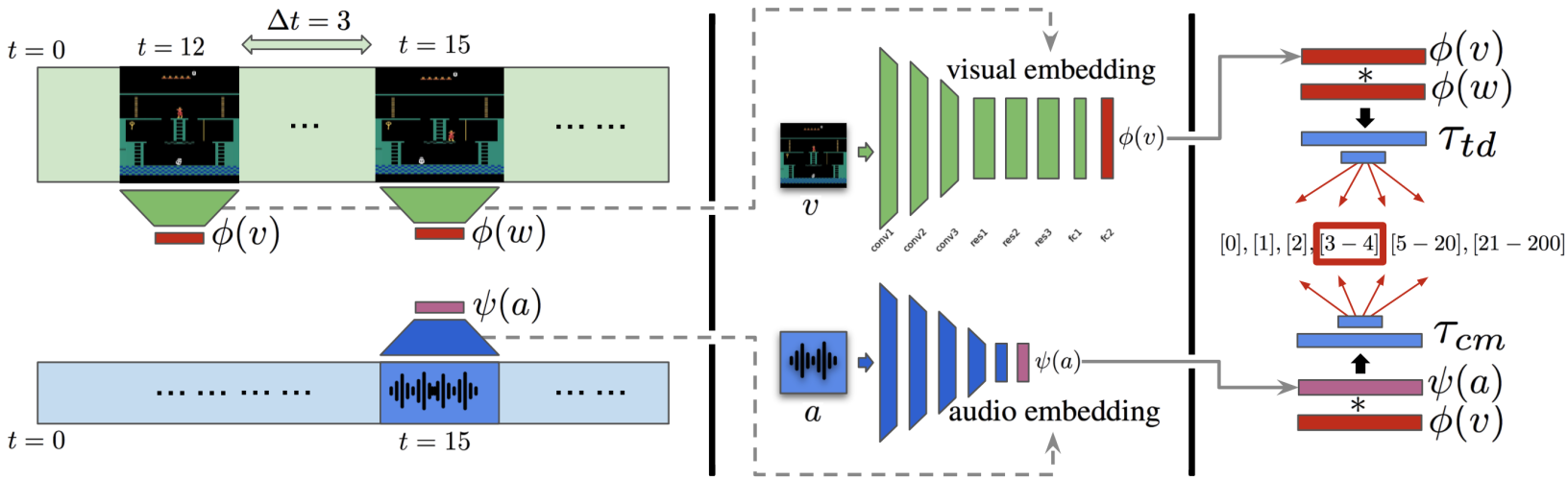
- Google团队利用人类玩家玩蒙特祖玛复仇的YouTube多版本在线录像来训练AI(Aytar et al., 2018)
  - 自监督学习方式对齐不同录像
  - TDC 画面时间差预测(**T**emporal **d**istance **c**lassification)
  - CMC 画面与声音对齐的跨模态时间差预测(**C**ross-**m**odal temporal distance **c**lassification)



(a) An example path (b) Aligned frames (c) Our embedding (d) Pixel embedding

图 30: 学习表征将不同画面在表征空间中对齐

TDC+CMC



(a) Temporal and cross-modal pair selection (b) Embedding networks (c) Classification networks

图 31: 基于YouTube示例录像的模仿学习神经网络模型

LfSD

- LfSD(Salimans and Chen, 2018)不断重新设置出发点，将长期规划任务拆分成多个子任务

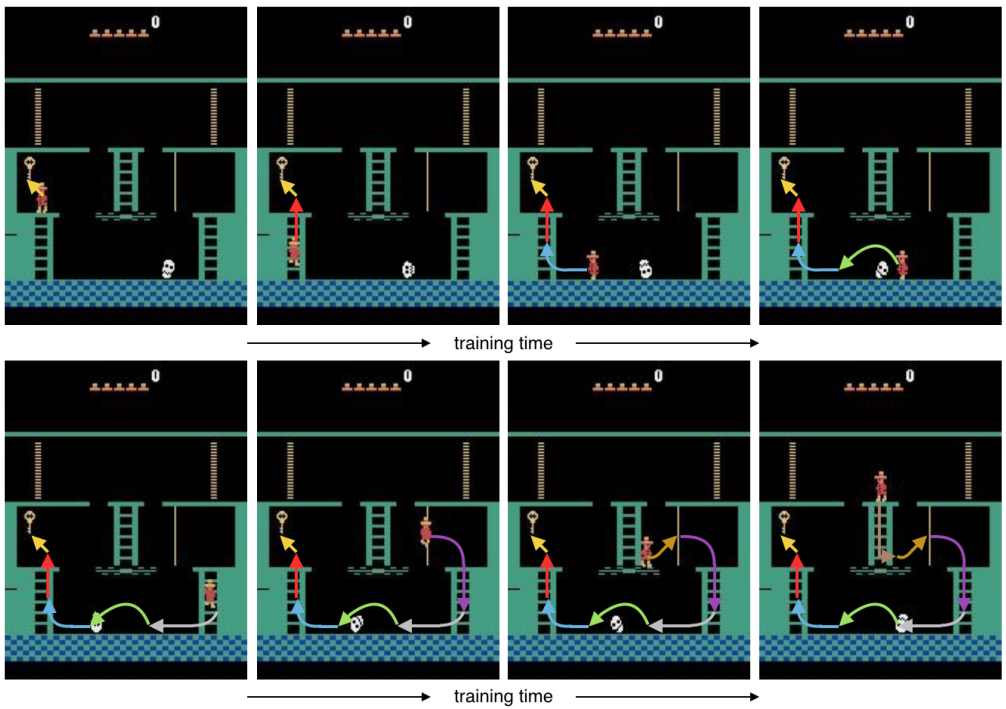


图 32: 逐步后置起始点从而引导智能体学习到更长的行动路径

R2D3

- R2D3(Paine et al., 2020)
  - 多个actor进程并行采样
  - 所有actor进程共享经验缓冲池
  - 全局learner运行Double DQN网络
  - 示例经验和actor采样经验按比例(ρ)混合

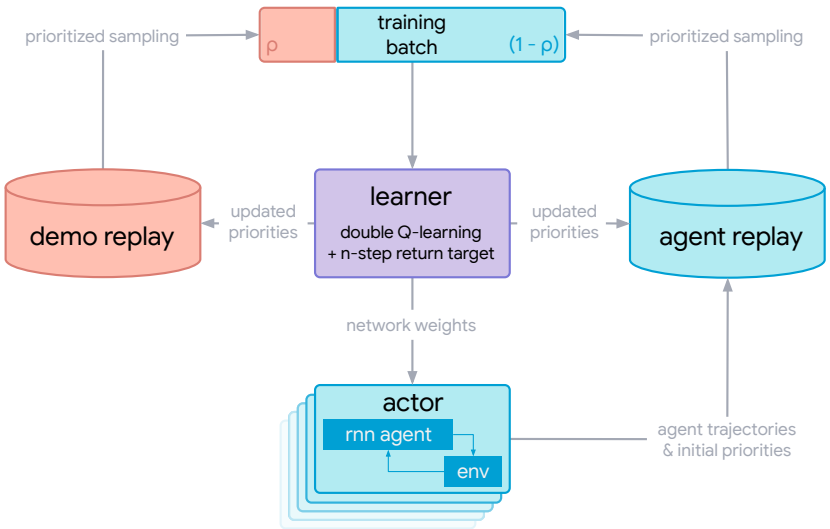


图 33: Recurrent Replay Distributed DQN from Demonstrations(R2D3)框架

Bootstrapped DQN

- Bootstrapped DQN(Osband et al., 2016)
  - 同时学习关于(s, a)的$K \in \mathbb{N}$个Q值来近似建模出Q值分布
  - 多个Q值网络共享一部分参数
  - 动作选取呈现出受Q值分布影响的探索效果

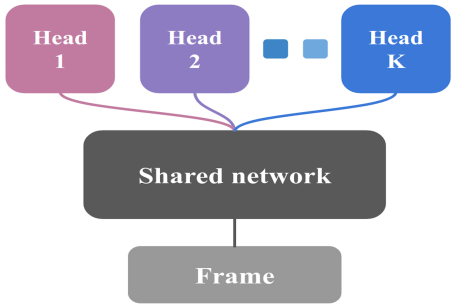


图 34: Bootstrapped DQN架构

VIME

- VIME(Houthooft et al., 2016)是基于变分信息最大化的探索方法
- 智能体连续决策并降低环境的不确定性，形式化为最大化在$\{a_t\}$上的累计熵减

$$\sum_t \left(H\left(\Theta|\xi_t, a_t\right) - H\left(\Theta|s_{t+1}, \xi_t, a_t\right)\right)$$

$$H\left(\Theta|\xi_t, a_t\right) - H\left(\Theta|s_{t+1}, \xi_t, a_t\right) = \mathbb{E}_{s_{t+1}\sim\mathcal{P}(.|\xi_t,a_t)}[D_{\mathrm{KL}}(p(\theta|\xi_t, a_t, s_{t+1})\|p$$

- 增强奖赏r^i

$$\eta D_{\mathrm{KL}}\left[p\left(\theta|\xi_t, a_t, s_{t+1}\right)\|p\left(\theta|\xi_t\right)\right]$$

总结与展望

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin A. Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen. King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, 2015.

Hado van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *AAAI*, 2016.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *CoRR*, abs/1511.05952, 2016.

Ziyu Wang, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, and Nando de Freitas. Dueling network architectures for deep reinforcement learning. *ArXiv*, abs/1511.06581, 2016.

Matteo Hessel, Joseph Modayil, H. V. Hasselt, T. Schaul, Georg Ostrovski, W. Dabney, Dan Horgan, B. Piot, Mohammad Gheshlaghi Azar, and D. Silver. Rainbow: Combining improvements in deep reinforcement learning. *ArXiv*, abs/1710.02298, 2018.

D. Silver, Julian Schrittwieser, K. Simonyan, Ioannis Antonoglou, Aja Huang, A. Guez, T. Hubert, L. Baker, Matthew Lai, A. Bolton, Yutian Chen, T. Lillicrap, F. Hui, L. Sifre, George van den Driessche, T. Graepel, and Demis Hassabis. Mastering the game of go without human knowledge. *Nature*, 550:354–359, 2017.

Yuri Burda, Harrison A Edwards, Amos J. Storkey, and Oleg Klimov. Exploration by random network distillation. *ArXiv*, abs/1810.12894, 2019.

Sergey Levine. Deep reinforcement learning. http://rail.eecs.berkeley.edu/deeprlcourse-fa18/, 2018.

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 488–489, 2017.

Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. Self-supervised exploration via disagreement, 2019.

Leshem Choshen, Lior Fox, and Yonatan Loewenstein. Dora the explorer: Directed outreaching reinforcement action-selection. *ArXiv*, abs/1804.04012, 2018.

Nikolay Savinov, Anton Raichuk, Raphael Marinier, Damien Vincent, Marc Pollefeys, Timothy P. Lillicrap, and Sylvain Gelly. Episodic curiosity through reachability. *ArXiv*, abs/1810.02274, 2019.

Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Exploration: A study of count-based exploration for deep reinforcement learning. *ArXiv*, abs/1611.04717, 2017.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. *ArXiv*, abs/1702.08165, 2017.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *ICML*, 2018.

Meire Fortunato, Mohammad Gheshlaghi Azar, Bilal Piot, Jacob Menick, Ian Osband, Alex Graves, Vlad Mnih, Rémi Munos, Demis Hassabis, Olivier Pietquin, Charles Blundell, and Shane Legg. Noisy networks for exploration. *ArXiv*, abs/1706.10295, 2018.

Matthias Plappert, Rein Houthooft, Prafulla Dhariwal, Szymon Sidor, Richard Y. Chen, Xi Chen, Tamim Asfour, Pieter Abbeel, and Marcin Andrychowicz. Parameter space noise for exploration. *ArXiv*, abs/1706.01905, 2017.

Marcin Andrychowicz, Dwight Crow, Alex Ray, Jonas Schneider, Rachel H Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *ArXiv*, abs/1707.01495, 2017.

Yoshua Bengio, J茅r么me Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 41–48, 2009.

Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth O. Stanley, and Jeff Clune. Go-explore: a new approach for hard-exploration problems. *ArXiv*, abs/1901.10995, 2019.

Yusuf Aytar, Tobias Pfaff, David Budden, Thomas Paine, Ziyu Wang, and Nando de Freitas. Playing hard exploration games by watching youtube. In *NeurIPS*, 2018.

Tim Salimans and Richard Chen. Learning montezuma's revenge from a single demonstration. *ArXiv*, abs/1812.03381, 2018.

Tom Le Paine, Caglar Gulcehre, Bobak Shahriari, Misha Denil, Matthew D. Hoffman, Hubert Soyer, Richard Tanburn, Steven Kapturowski, Neil C. Rabinowitz, D.V.J. Williams, Gabriel Barth-Maron, Ziyu Wang, Nando de Freitas, and Worlds Team. Making efficient use of demonstrations to solve hard exploration problems. *ArXiv*, abs/1909.01387, 2020.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. In *Advances in neural information processing systems*, pages 4026–4034, 2016.

Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. In *NIPS*, 2016.