arXiv:2303.02135v1 [cs.LG] 3 Mar 2023

# 最终折扣 时序逻辑反事实经验回放

**Cameron Voloshin** [1] **Abhinav Verma** [2] **Yisong Yue** [1 3]


## 摘要

线性时序逻辑（LTL）为策略优化提供了一种简化的任务描述方式，而这些任务原本可能难以通过标量奖励函数来描述。然而，标准的强化学习框架可能过于短视，难以找到最大程度满足 LTL 的策略。本文作出两项贡献。首先，我们提出一种新的基于价值函数的代理指标，并采用一种称为最终折扣的技术，据此可以找到以最高可实现概率满足线性时序逻辑规范的策略。其次，我们提出一种新的经验回放方法，通过对满足线性时序逻辑规范的不同方式进行反事实推理，从在策略轨迹采样中生成离策略数据。在离散和连续状态—动作空间中进行的实验，证实了我们提出的反事实经验回放方法的有效性。


## 1. 引言

在标准的强化学习（RL）框架中，目标是在未知环境中制定一种能够最大化奖励函数的策略。在许多强化学习应用中，实践者负责设计奖励函数，使智能体在学习过程结束后能够采取期望的行为。然而，要通过标量奖励传达现实世界任务的规范，可能颇具挑战（Randløv& Alstrøm，1998年；Toromanoff et al.，2019年；Ibarzetal.，2018年；Zhanget al.，2021年；Nget al.，1999）。这种做法俗称奖励塑形，实践者往往会借助启发式“面包屑”（Sorg，2011年）来引导智能体采取预期行为。尽管存在“奖励函数足矣”的假设（Sutton & Barto，2018年；Silveret al.，2021年），有些任务可能无法化约为标量奖励（Abel et al.，2021年）。

为应对这些挑战，研究人员探讨了使用线性时序逻辑（LTL）来规范智能体行为的替代性强化学习框架——（见第 7 节）。线性时序逻辑可以表达系统未来路径所需的特性（Baier 和 Katoen，2008），从而能够精确、灵活地规范任务和行为。例如，我们可以要求系统按照特定顺序反复完成一组目标（更多示例见第 2 节）。

在没有重大假设的情况下，无法准确说明策略满足线性时序逻辑目标的概率。现有工作几乎无一例外地使用Q学习与稀疏强化学习启发式方法（Bozkurt 等人，2020年；Cai 等人，2021a），旨在促使智能体生成看起来能够满足任务的轨迹。首先，这些启发式方法包含复杂的技术假设，因而难以获得非渐近保证，即使在有限状态-动作空间中也是如此。其次，稀疏奖励函数对任何基于梯度的强化学习算法都构成了重大挑战，因为它们提供的信号不足以妥善解决信用分配问题。解决稀疏性需要采用分层方法（Bozkurt 等人，2020年）或重新引入奖励塑形（Hasanbeig 等人，2018年）。关于既有工作的进一步阐述，请参见第7 节。

**我们的贡献**。本文聚焦于通过与环境进行在线交互，对由线性时序逻辑指定的目标进行无模型策略学习。我们做出了两项技术贡献。首先，我们使用一种称为最终折扣的技术，利用经过修改的价值函数代理重新表述强化学习问题。其核心思想是考虑到，最优地满足线性时序逻辑规范可能与满足该规范所需的时间长短无关（例如，“最终始终到达目标”）。我们在第4节证明，最终折扣下的最优策略能够最大化满足线性时序逻辑规范的概率。

其次，我们提出一种经验回放方法来解决奖励稀疏性问题。具体而言，任何 LTL公式都可以转换为一个完全已知的专用有限状态自动机，据此，我们可以从一条同策略轨迹生成多条反事实轨迹。我们将该方法称为*LTL* 引导的反事实经验回放。我们通过有限状态/动作空间和连续状态/动作空间，并分别采用 Q学习和策略梯度方法，对反事实经验回放方法带来的性能提升进行了实证验证。

[1]加州理工学院 [2]宾夕法尼亚州立大学 [3]Latitude AI。通信联系人：Cameron Voloshin <cvoloshin@caltech.edu>。

初步工作。正在由国际机器学习会议（ICML）审阅。请勿传播。

## 2. 预备知识

我们介绍必要的背景知识并给出示例，以帮助理解问题陈述和解决方案。原子命题 是一个取真值的变量。在一组原子命题 AP 上定义的字母表 为 $\Sigma = 2^{\mathrm{AP}}$。例如，如果 AP $= \{x, y\}$，那么$\Sigma = \{\{\}, \{x\}, \{y\}, \{x, y\}\}$。 $\Delta(A)$ 表示集合 $A$上的概率分布集合。

### 2.1. 运行示例

我们将使用图 1（First）所示的环境作为运行示例。智能体用绿色圆点表示，环境中有 3 个分别着黄色、蓝色和红色的圆形区域。原子命题由 $\{y, b, r\}$给出，分别指代相应颜色的区域。 $\Sigma$中的元素表示智能体所在的区域。

#### 2.2. 带标签状态空间的马尔可夫决策过程

沿用 Voloshin 等人（2022）的类似记号，我们假设环境遵循由元组 $\mathcal{M} = (\mathcal{S}^{\mathcal{M}}, \mathcal{A}^{\mathcal{M}}, P^{\mathcal{M}}, d_0^{\mathcal{M}}, \gamma, L^{\mathcal{M}})$ 给出的折扣标记马尔可夫决策过程（马尔可夫决策过程）框架，该元组由状态空间 $\mathcal{S}^{\mathcal{M}}$、动作空间 $\mathcal{A}^{\mathcal{M}}$、未知的转移函数 $P^{\mathcal{M}}: \mathcal{S}^{\mathcal{M}} \times \mathcal{A}^{\mathcal{M}} \to \Delta(\mathcal{S}^{\mathcal{M}})$、初始状态分布 $d_0^{\mathcal{M}} \in \Delta(\mathcal{S}^{\mathcal{M}})$以及标记函数$L^{\mathcal{M}}: \mathcal{S}^{\mathcal{M}} \to \Sigma$组成。令 $A^{\mathcal{M}}(s)$ 表示状态 $s$下可用动作的集合。

与传统的马尔可夫决策过程不同， $\mathcal{M}$ 具有一个标记函数 $L^{\mathcal{M}}$，它会返回该状态下为真的原子命题。例如，在图1 （第一幅图）中，当智能体进入状态 $s \in \mathcal{S}^{\mathcal{M}}$，且该状态同时位于黄色区域和蓝色区域内时， $L^{\mathcal{M}}(s) = \{y, b\}$。

### 2.3. 线性时序逻辑（LTL）

这里我们将对线性时序逻辑作基本介绍。若要了解更全面的概述，请参阅Baier&Katoen(2008)。

**定义 2.1** （线性时序逻辑规范， $\varphi$）。 线性时序逻辑规范 $\varphi$ 是对任务的完整描述，由原子命题与逻辑联结词组合构成：非（$\neg$）、与（&）和蕴含（$\to$）；以及时序算子：下一步（$X$）、反复/始终/全局（$G$）、最终（$F$）和直到（$U$）。

**示例**。对于 $AP = \{x, y\}$，一些基本任务规范包括安全性（$G\neg x$）、可达性（$Fx$）、稳定性（$FGx$）、响应（$x \to Fy$）和进展（$x \,\&\, XFy$）。
再来看看图中的环境1（第一张），其中$AP = \{y, r, b\}$。如果任务是最终到达黄色区域并留在那里（称为稳定化），那么我们写作$\varphi = FGy$。或者，如果我们希望智能体无限循环于黄色区域和红色区域之间，同时避开蓝色区域，则有 $\varphi = GF(y \,\&\, XFr) \,\&\, G\neg b$，这是安全性、可达性和进展的组合。

### 2.4. LTL 满足性

LTL 具有递归语义，用于定义逻辑连接词满足性的含义。在不失一般性的情况下，我们将使用一种专用自动机，即极限确定性 Büchi 自动机 $\mathcal{B}_\varphi$（Sickert 等人，2016年），其定义如下，用于跟踪 $\varphi$ 满足性的进展。关于构造极限确定性 Büchi 自动机的更多细节，参见 Hahn 等人 （2013年）；Baier 和 Katoen （2008）；K˘ret´ınsk`y 等人 （2018年）。为简洁起见，我们省略 $\varphi$来自 $\mathcal{B}_\varphi$。

定义 2.2。（极限确定性 B¨uchi 自动机，LDBA（Sickert 等人，2016年））LDBA 是一个五元组 $\mathcal{B} = (\mathcal{S}^{\mathcal{B}}, \Sigma \cup \mathcal{A}_{\mathcal{B}}, P^{\mathcal{B}}, \mathcal{S}^{\mathcal{B}^*}, b_{-1}^{\mathcal{B}})$，由以下部分组成：(i) 有限状态集 $\mathcal{S}^{\mathcal{B}}$；(ii) 有限字母表 $\Sigma = 2^{\mathrm{AP}}$； $\mathcal{A}_{\mathcal{B}}$ 是一组带索引的跳转转移；(iii) 转移函数$P^{\mathcal{B}}$： $\mathcal{S}^{\mathcal{B}} \times (\Sigma \cup \mathcal{A}_{\mathcal{B}}) \to \mathcal{S}^{\mathcal{B}}$；(iv) 接受状态 $\mathcal{S}^{\mathcal{B}^*} \subseteq \mathcal{S}^{\mathcal{B}}$；以及 (v) 初始状态 $b_{-1}^{\mathcal{B}}$。存在对 $\mathcal{S}^{\mathcal{B}} = \mathcal{S}_D^{\mathcal{B}} \cup \mathcal{S}_N^{\mathcal{B}}$ 的互斥划分，使得 $\mathcal{S}^{\mathcal{B}^*} \subseteq \mathcal{S}_D^{\mathcal{B}}$；对于 $b \in S_D^{\mathcal{B}}$， $a \in \Sigma$ 则 $P^{\mathcal{B}}(b, a) \subseteq \mathcal{S}_D^{\mathcal{B}}$ 是闭合的。 $\mathcal{A}_{\mathcal{B}}(b)$ 仅在 $b \in \mathcal{S}_N^{\mathcal{B}}$ 时（可能）非空，并允许 $\mathcal{B}$ 在不读取 AP 的情况下转移到 $\mathcal{S}_D^{\mathcal{B}}$。路径 $\varrho = (b_0, b_1, \ldots)$ 是 $\mathcal{B}$ 中状态的序列，这些状态在 $P^{\mathcal{B}}$ 下通过连续转移到达。

**定义2.3**。 ($\mathcal{B}$ 接受) **$\mathcal{B}$ 接受** 一条路径 $\varrho$，如果路径中存在某个状态 $b \in \mathcal{S}^{\mathcal{B}^*}$ 被无限次访问。

示例。再次考虑图1（左）中的环境，其中 $AP = \{y, r, b\}$。如果我们想要为 $\varphi = FGy$ （到达并稳定在 $y$）构造一个极限确定性 Büchi 自动机，那么就会得到图1（右）所示的状态机。在这个状态机中，智能体从状态 0 开始。接受集为 $\mathcal{S}^{\mathcal{B}^*} = \{1\}$。状态 0 和状态 1 之间的转移在形式上称为跳转转移： $\mathcal{A}_{\mathcal{B}}(0) = \{\epsilon\}$；如果 $\mathcal{A}_{\mathcal{B}}(\cdot)$，则 $= \varnothing$。每当智能体处于极限确定性 Büchi 自动机的状态 0 时，它可以选择留在状态 0，也可以立即转移到状态 1。这个选择意味着智能体认为自己已经满足了线性时序逻辑规范中“最终”这一部分。当智能体执行这次跳转后，此后必须满足 $y$，才能留在状态 1。智能体可以自行决定何时认为自己能够在此后满足 $y$。执行跳转后，如果智能体未能留在 $y$，就会立即转移到汇点，即状态 2。当状态 1 被无限次到达时，极限确定性 Büchi 自动机接受，这意味着智能体最终满足了“始终 $y$”，正如所期望的那样。

另一个例子是，这次不涉及跳转转移： $\varphi = GF(y \,\&\, XFr) \,\&\, G\neg b$(在 $y$和 $r$ 之间永远振荡，同时避开 $b$)。极限确定性 Büchi 自动机如图 1 （第三幅）所示。在这个状态机中，智能体从状态 1 出发，接受集为 $\mathcal{S}^{\mathcal{B}^*} = \{1\}$。要返回状态 1，智能体必须同时访问 $r$ 和 $y$。

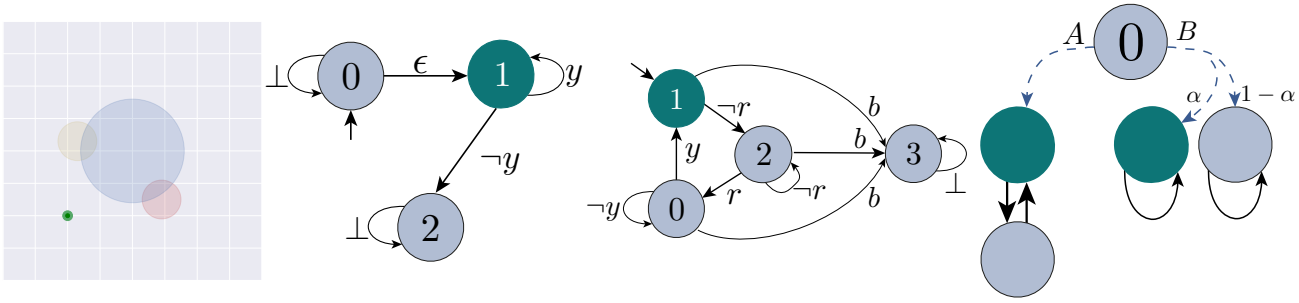


图*1.* 示例。 首先： Flatworld 环境示意图。智能体是一个绿色圆点，其中有3个区域：黄色、蓝色和红色。其次： “$FGy$”对应的极限确定性 Büchi 自动机 $\mathcal{B}$。 $\mathcal{S}^{\mathcal{B}^*} = \{1\}$，以绿色圆圈表示。初始状态为 $b_{-1} = 0$。第三： “$GF(y \,\&\, XFr) \,\&\, G\neg b$”对应的极限确定性 Büchi 自动机 $\mathcal{B}$。 $\mathcal{S}^{\mathcal{B}^*} = \{1\}$，以绿色圆圈表示。初始状态为 $b_{-1} = 1$。第四： 对于这个示例，从状态0开始的智能体在求解 $\arg\max_{\pi\in\Pi} \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}[\sum_{i=0}^{\infty} \gamma^i \mathbf{1}_{\{b_i\in\mathcal{S}^{\mathcal{B}^*}\}}]$，其中 $\mathcal{S}^{\mathcal{B}^*}$ 如绿色圆圈所示时，如果对于任意 $\gamma \in [0,1]$， $\alpha \in (1/2, 1)$，它会以概率1选择动作 $B$。这样的策略具有 $\mathbb{P}[\pi \models \varphi] = \alpha < 1$。然而，概率最优策略会确定性地选择动作 $A$，其 $\mathbb{P}[\pi^* \models \varphi] = 1$。这说明了灾难性的短视行为。

无限次执行此操作即可满足极限确定性 Büchi 自动机条件，因此也满足该规范。如果在任何时候遇到 $b$，智能体就会转移到汇点，即状态3。

## 3. 问题表述

我们首先引入一些额外的记号。令 $\mathcal{Z} = \mathcal{S}^{\mathcal{M}} \times \mathcal{S}^{\mathcal{B}}$。令 $\Pi : \mathcal{Z} \times \mathcal{A} \to \Delta([0,1])$ 为定义于 MDP 与 LDBA （如下所定义）的乘积空间上的一个（随机性）策略类，其中 $\mathcal{A}((s, b)) = \mathcal{A}^{\mathcal{M}}(s) \cup \mathcal{A}^{\mathcal{B}}(b)$，以考虑 $\mathcal{B}$ 中的跳转转移。

**将 MDP 与 LDBA 同步**。 对于任意 $(s, b) \in \mathcal{Z}$，策略 $\pi \in \Pi$ 能够在 $\mathcal{A}^{\mathcal{M}}(s)$ 中选择一个动作，或在 $\mathcal{A}^{\mathcal{B}}(b)$ 中选择一个动作（如果可用）。因此，我们可以根据新的概率转移关系，将 **轨迹** 生成为序列 $\tau = (s_0, b_0, a_0, s_1, b_0, a_1, \ldots)$，其定义如下：

$$P(s', b'|s, b, a) = \begin{cases} P^{\mathcal{M}}(s, a, s') & a \in A^{\mathcal{M}}(s), b' \in P^{\mathcal{B}}(b, L(s')) \\ 1, & a \in A^{\mathcal{B}}(b), b' \in P^{\mathcal{B}}(b, a), s = s' \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

令 $\tau$ 的 LDBA 投影为子序列 $\tau_{\mathcal{B}} = (b_0, b_1, \ldots)$。 $\tau_{\mathcal{B}}$ 中的元素可被视为跟踪智能体对线性时序逻辑规范的满足情况：

**定义 3.1** （运行满足性， $\tau \models \varphi$）。 我们说一条轨迹满足 $\varphi$，如果 $\mathcal{B}$ 接受 $\tau_{\mathcal{B}}$；当 $\exists b \in \tau_{\mathcal{B}}$ 与 $b \in \mathcal{S}^{\mathcal{B}^*}$ 一起无限次发生时，即可满足这一条件。

令 $\mathrm{T}_\pi^P = \mathbb{E}_{z\sim d_0^{\mathcal{M}}\times\{b_{-1}\}}[\mathrm{T}_\pi^P(z)]$ 表示从任意初始状态出发的所有可能轨迹上的分布 $z \in d_0^{\mathcal{M}} \times \{b_{-1}\}$ 其中 $\mathrm{T}_\pi^P(z)$ 表示从 $z \in \mathcal{Z}$ 出发、由 $\pi$ 在关系 $P$ （见（1））下生成的所有可能轨迹上的（条件）分布。LTL 满足性的概率通过计算满足线性时序逻辑规范的轨迹数量得出：

**定义 3.2**（状态满足， $z \models \varphi$）。 $\mathbb{P}_\pi[z \models \varphi] = \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P(z)}[\mathbf{1}_{\{\tau\models\varphi\}}] = \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}[\mathbf{1}_{\{\tau\models\varphi\}}|z_0 = z]$**定义 3.3** （策略满足， $\pi \models \varphi$）。 $\mathbb{P}[\pi \models \varphi] = \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}[\mathbf{1}_{\{\tau\models\varphi\}}]$ 其中 $\mathbf{1}_X$ 是 $X$的指示函数。

理想情况下，我们希望找到一个具有最高线性时序逻辑规范满足概率的策略：即生成最多满足线性时序逻辑的运行的策略。形式化地，

$$\pi^* \in \arg\max_{\pi\in\Pi} \mathbb{P}[\pi \models \varphi]. \quad (2)$$

我们注意到，公式（2）是构建线性时序逻辑满足性策略优化问题的标准起点（Yang 等人，2021年；Bozkurt 等人，2020年；Cai 等人，2021a；Hasanbeig 等人，2018年；2020年；Voloshin 等人，2022年）。

## 4. 强化学习友好形式：最终折扣

遗憾的是，公式（2）中的最大化问题并不容易优化，因为我们没有关于 $\mathbb{P}[\pi \models \varphi]$ 的直接信号。在没有任何额外假设（例如对马尔可夫决策过程的结构化知识）的情况下，任何有限子序列只能提供关于 $\tau \models \varphi$的证据，却无法给出具体证明。

**最终折扣**。 为了解决上述问题，我们开发了一种基于价值函数的改进代理方法，如下所示。给定一条轨迹 $\tau = (s_0, b_0, a_0, \ldots)$，我们会跟踪 $b_i \in \mathcal{S}^{\mathcal{B}^*}$ 的访问次数，并鼓励智能体尽可能多次访问 $\mathcal{S}^{\mathcal{B}^*}$。具体而言，在最终折扣下，当智能体处于状态 $b_i \in \mathcal{S}^{\mathcal{B}^*}$ 时，价值函数会给予其 1 的奖励，且不会对两次访问 $\mathcal{S}^{\mathcal{B}^*}$之间的时间间隔进行折扣。形式上，我们将寻求

$$\pi_\gamma^* \in \arg\max_{\pi\in\Pi} \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}[\sum_{i=0}^{\infty} \Gamma_i \mathbf{1}_{\{b_i\in\mathcal{S}^{\mathcal{B}^*}\}}] \quad (\equiv V_\pi^\gamma), \quad (3)$$

其中 $\Gamma_0 = 1$和

$$\Gamma_i = \prod_{t=0}^{i-1} \gamma(b_t), \quad \gamma(b_t) = \begin{cases} \gamma, & b_t \in \mathcal{S}^{\mathcal{B}^*} \\ 1, & \text{otherwise} \end{cases}. \quad (4)$$

**直觉** $\Gamma_i$。乍看之下，将 $\Gamma_i = \gamma^i$ 设为传统强化学习指数折扣率似乎是合理的。遗憾的是，$\nexists \gamma \in [0,1]$ 通过 $\Gamma_i = \gamma^i$ 可以避免灾难性的短视行为。具体来说，来看图 1（第四幅）。智能体从状态 0 开始，且只有两个动作 $A$ 和 $B$。采取动作 $A$ 会直接转移到一个接受状态，此后每隔 2 步就会访问一次该接受状态。另一方面，动作 $B$ 以概率 $\alpha$ 转移到一个接受状态，并以概率 $1-\alpha$转移到一个汇状态。通过动作$B$ 到达的接受状态会在每一步都被重新访问。假设 $\beta = \pi(A) = 1 - \pi(B)$那么我们可以计算:

$$\mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}[\sum_{i=0}^{\infty} \gamma^i \mathbf{1}_{\{b_i \in \mathcal{S}^{\mathcal{B}^*}\}}] = \frac{\beta}{1-\gamma^2} + \frac{(1-\beta)\alpha}{1-\gamma}. \quad (5)$$

对于 $\alpha > 1/2$，最优选择 $\beta$ 是 $\beta = 0$，这意味着$P(\pi \models \varphi) = \alpha$。当 $\alpha \in (1/2,\ 1)$ 时，这意味着$\pi$ 不是概率最优。事实上，$P(\pi \models \varphi) = \alpha < 1$当 $\beta = 0$，但通过选择 $\beta = 1$，$P(\pi^* \models \varphi) = 1$。这里的直觉可以通过采用 $\gamma \to 1$形式化：执行动作 $A$ 的平均奖励为 $1_{2}$，而执行动作 $B$ 的平均奖励以概率 $\alpha$为 $\mathbf{1}$；当 $\alpha > 1/2$较大时，这种选择值得冒险。

为了避免这种短视行为，我们必须避免区分在良好状态之间的返回时间。返回到 $\mathcal{S}^{\mathcal{B}^*}$ 所需的步数（平均而言）无关紧要：我们只要求系统确实会返回。因此，在系统不处于 $\mathcal{S}^{\mathcal{B}^*}$时，我们在定义 $\Gamma_i$ 时不计算时间（因此有 $\gamma = 1$）。我们将此称为最终折扣。

### 4.1. 对 $\pi_\gamma^*$ 的分析

本节分析满足 $\varphi$ 的 $\pi_\gamma^*$ 的概率与最佳可能情况 $\pi^*$ 的概率之间的比较。

令集合 $O(\tau) = \{i : b_i \in \mathcal{S}^{\mathcal{B}^*}\}$ 表示到达良好状态的时刻（时间步）。这个量很自然，因为当且仅当 $\tau \models \varphi$ 时，$|O(\tau)| = \infty$。

**引理 4.1.** 对于任意 $\pi \in \Pi$ 和 $\gamma \in (0,1)$，有

$$|(1-\gamma)V_\pi^\gamma - \mathbb{P}[\pi \models \varphi]| \le \log(\frac{1}{\gamma}) O_\pi$$

其中 $O_\pi = \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}\left[|O(\tau)| \middle| \tau \not\models \varphi\right]$ 是不满足 $\varphi$ 的轨迹中到达接受状态的预期访问次数。

证明。固定某个状态 $z = (s,b) \in \mathcal{Z}$。

$$\begin{aligned} V_\pi^\gamma(z) &= \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}[\sum_{i=0}^{\infty} \Gamma_i \mathbf{1}_{\{b_i \in \mathcal{S}^{\mathcal{B}^*}\}} | z_0 = z] \\ &= \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}\left[\sum_{j=0}^{|O(\tau)|} \gamma^j | z_0 = z\right] \end{aligned}$$

利用 $\sum_{j=0}^{k} \gamma^j = \frac{1-\gamma^k}{1-\gamma}$ 这一事实，我们得到

$$\begin{aligned} V_\pi^\gamma(z) = \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}\left[\frac{1-\gamma^{|O(\tau)|}}{1-\gamma} \middle| \substack{\tau\models\varphi \\ z_0=z}\right] \mathbb{P}_\pi[z \models \varphi] \\ + \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}\left[\frac{1-\gamma^{|O(\tau)|}}{1-\gamma} \middle| \substack{\tau\not\models\varphi \\ z_0=z}\right] \mathbb{P}_\pi[z \not\models \varphi] \quad (6) \end{aligned}$$

由于对于任意 $\tau \models \varphi$，$|O(\tau)| = \infty$，

$$\mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}\left[\frac{1-\gamma^{|O(\tau)|}}{1-\gamma} \middle| \substack{\tau\models\varphi \\ z_0=z}\right] = \frac{1}{1-\gamma} \quad (7)$$

连同 $\mathbb{P}_\pi[z \not\models \varphi] \ge 0$ 可得

$$V_\pi^\gamma(z) \ge \frac{1}{1-\gamma} \mathbb{P}_\pi[z \models \varphi]. \quad (8)$$

对初始状态取期望，可得

$$V_\pi^\gamma \ge \frac{1}{1-\gamma} \mathbb{P}[\pi \models \varphi]. \quad (9)$$

现在我们来求一个上界。设 $M_\pi(t) = \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}\left[e^{t|O(\tau)|} \middle| \tau \not\models \varphi\right]$。再次从公式 (6) 出发，并利用公式 (7)，可得

$$V_\pi^\gamma(z) \le \frac{\mathbb{P}_\pi[z \models \varphi]}{1-\gamma} + \frac{1 - \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}\left[e^{\log(\gamma)|O(\tau)|} \middle| \substack{\tau\not\models\varphi \\ z_0=z}\right]}{1-\gamma} \quad (10)$$

这里用到了，对于任意 $z \in \mathcal{Z}$，都有 $\mathbb{P}_\pi[z \not\models \varphi] \le 1$。随后对初始状态分布取期望，可得

$$(1-\gamma)V_\pi^\gamma \le \mathbb{P}[\pi \models \varphi] + 1 - M_\pi(\log(\gamma)) \quad (11)$$

特别地，$M_\pi(t)$ 是凸函数，因此其值不低于其切线:

$$\begin{aligned} M_\pi(t) \ge M_\pi(0) + tM'_\pi(0) &= 1 + t\mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}\left[|O(\tau)| \middle| \tau \not\models \varphi\right] \\ &= 1 + tO_\pi \end{aligned}$$

将这个不等式代入公式 (11)，并结合公式 (9)，

$$\mathbb{P}[\pi \models \varphi] \le (1-\gamma)V_\pi^\gamma \le \mathbb{P}[\pi \models \varphi] + \log(\frac{1}{\gamma}) O_\pi \quad (12)$$

两边同时减去 $\mathbb{P}[\pi \models \varphi]$ 并取绝对值，即完成证明。□

**定理 4.2.**（非渐近保证）对于任意 $\gamma \in (0,\ 1)$，

$$\sup_{\pi\in\Pi} \mathbb{P}[\pi \models \varphi] - \mathbb{P}[\pi_\gamma^* \models \varphi] \le 2\log(\frac{1}{\gamma}) \sup_{\pi\in\Pi} O_\pi \quad (13)$$

*where* $O_\pi = \mathbb{E}_{\tau\sim \mathrm{T}_\pi^P}\left[|O(\tau)| \middle| \tau \not\models \varphi\right]$.

证明. 考虑任意满足 $\mathbb{P}[\pi_i \models\varphi] \to \sup_\pi \mathbb{P}[\pi \models \varphi]$ as $i \to \infty$的序列 $\{\pi_i\}_{i=1}^{\infty}$。那么对于任意 $\pi_i$,

$$\begin{aligned}\mathbb{P}[\pi_i \models \varphi] - \mathbb{P}[\pi^*_\gamma \models \varphi] &= \mathbb{P}[\pi_i \models \varphi] - (1-\gamma)V^\gamma_{\pi_i} \\ &\quad + (1-\gamma)V^\gamma_{\pi_i} - (1-\gamma)V^\gamma_{\pi^*_\gamma} \\ &\quad + (1-\gamma)V^\gamma_{\pi^*_\gamma} - \mathbb{P}[\pi^*_\gamma \models \varphi] \\ &\overset{(a)}{\leq} |\mathbb{P}[\pi_i \models \varphi] - (1-\gamma)V^\gamma_{\pi_i}| \\ &\quad + |\mathbb{P}[\pi^*_\gamma \models \varphi] - (1-\gamma)V^\gamma_{\pi^*_\gamma}| \\ &\overset{(b)}{\leq} \log(\frac{1}{\gamma})(O_{\pi_i} + O_{\pi^*_\gamma}) \\ &\overset{(c)}{\leq} 2\log(\frac{1}{\gamma}) \sup_{\pi\in\Pi} O_\pi\end{aligned}$$

其中$(a)$ 由三角不等式并去掉项$(1-\gamma)V^\gamma_{\pi_i} - (1-\gamma)V^\gamma_{\pi^*_\gamma}$构成，因为根据 $\pi^*_\gamma$的定义，该项非正；$(b)$ 是引理4.1的应用，而$(c)$是对所有策略取上确界。当 $i \to \infty$ 时，对两边取极限，即完成证明。 □

**推论 4.3**。 如果 $\Pi$ 中的策略数量有限，那么$\sup_{\pi\in\Pi} O_\pi = m < \infty$ 可以取到，且是有限常数，并且

$$\sup_{\pi\in\Pi} \mathbb{P}[\pi \models \varphi] - \mathbb{P}[\pi^*_\gamma \models \varphi] \leq 2m\log(\frac{1}{\gamma})$$

**推论 4.4**。 若 $\mathcal{S}^{\mathcal{M}}$ 和 $\mathcal{A}^{\mathcal{M}}$ 均为有限值，则$\mathcal{Z}$ 和 $\mathcal{A}$ 均为有限值。已知最优策略是确定性的（*Puterman, 2014*年），因此我们只需考虑确定性策略，而这类策略的数量有限。因此$sup_{\pi\in\Pi} O_\pi = m < \infty$ 可以取到，且是有限常数，并且

$$\sup_{\pi\in\Pi} \mathbb{P}[\pi \models \varphi] - \mathbb{P}[\pi^*_\gamma \models \varphi] \leq 2m\log(\frac{1}{\gamma})$$

### 4.2. 解释

定理 4.2 要求量 $\sup_{\pi\in\Pi} O_\pi$ 有限，才能使该界有意义。实际上，我们只需对 $M_\pi(\log(\gamma))$ 提出要求，但对 $O_\pi$ 提出要求更容易理解。顺便一提， $M_\pi(\log(\gamma))$ 可以解释为随机变量的矩生成函数，该随机变量表示访问 $\mathcal{S}^{\mathcal{B}^*}$的次数。我们转而考虑同样自然的量 $O_\pi$。 $O_\pi$ 它表示一条不满足规范的轨迹访问良好状态的次数（平均值）。理想情况下，这个数值应当较小，这样就容易区分良好策略与不良策略。

**坏消息**。 当 $\Pi$ 是一个无限类时，确保 $\sup_{\pi\in\Pi} O_\pi$ 有限的条件并非显而易见，而且取决于马尔可夫决策过程的转移函数 $P$和 $\Pi$的具体情况。

**算法 1**使用 LCER 学习

**参数：** 最大时域 $T$。经验回放缓冲区 $D = \{\}$。

1: **对于** $k = 1, 2, \cdots$ **执行**
2: 在马尔可夫决策过程中运行 $\pi_{k-1}$，持续 $T$ 个时间步并收集 $\tau = (s_0, b_0, a_0, \ldots, s_{T-1}, b_{T-1}, a_{T-1}, s_T, b_T)$
3: $D_k \leftarrow \text{LCER}(D_{k-1}, \tau)$
4: $\pi_k \leftarrow$ 更新$(\pi_{k-1}, D_k)$ // Q学习/策略梯度。
5: **结束循环**

假设 $\sup_{\pi\in\Pi} O_\pi$ 是无限的。这意味着存在一些策略，它们会产生最终无法到达 $\mathcal{S}^{\mathcal{B}^*}$的不良轨迹，但在此过程中会访问 $\mathcal{S}^{\mathcal{B}^*}$ 任意多次（但次数有限）。换句话说，在宇宙热寂之前，这些策略与真正的概率最优策略无法区分。

考虑图1 （右）中的规范：无限多次地在红色和黄色之间循环，同时避开蓝色。一种看似不错、实则不佳的策略是频繁完成任务，但在失败的情况下，会在最终失败之前在红色和黄色之间循环很多次。上确界$_{\pi\in\Pi} O_\pi$ 为无穷大意味着，存在一些策略，可以在失败之前循环任意多次。

**好消息是，推论**4.4 揭示，离散化足以生成概率最优策略，其次优性以$\log(\frac{1}{\gamma})$的速率缩小。这表明， $P$和 $\Pi$ 的紧性以及 $P$的连续性或许就足够了，但我们将这些条件留待未来工作。最后，由于所有计算机都以有限精度进行运算，策略数量有限，因此推论4.3同样适用。

## 5. LTL反事实经验回放

可以使用任意 Q 学习或策略梯度方法优化式（3）中的公式，如算法1 （第 4 行）所示。然而，这样做颇具挑战，因为它会受到奖励稀疏性的影响：智能体只有到达良好状态时才能获得信号。

我们通过利用极限确定性 Büchi 自动机来应对奖励稀疏性： $P^{\mathcal{B}}$ 是完全已知的。通过了解 $P^{\mathcal{B}}$，我们可以通过修改在极限确定性 Büchi 自动机中开始的状态，仅凭一条同策略轨迹生成多条离策略轨迹。值得注意的是，这并不需要访问马尔可夫决策过程的转移函数 $P^{\mathcal{M}}$。我们将这种方法称为线性时序逻辑引导的反事实经验回放，即LCER （算法1，第 3 行），因为它在标准经验回放（Lin，1992年；Mnih 等人，2013年；2015年）的基础上进行了修改，将极限确定性 Büchi 自动机中其他位置的反事实经验纳入其中。LCER 通过 Q 学习最容易理解，而应用于策略梯度方法时则需要谨慎修改。

**采用 LCER进行 Q学习**。 算法 2 概述了

**算法2**用于Q学习的LCER

**参数：** 数据集 $D$。长度为 $T$ 的轨迹 $\tau$。

**对于** $(s_t,\ a_t,\ s_{t+1}) \in \tau$ **执行**
  **对于** $b \in \mathcal{S}^{\mathcal{B}}$ **执行**
    Set $\tilde{b} \leftarrow P^{\mathcal{B}}(b, L^{\mathcal{M}}(s_{t+1}))$
    $D \leftarrow D \cup (s_t, b, a_t, \mathbf{1}_{\tilde{b} \in B^*}, s_{t+1}, \tilde{b})$
    **对于** $\epsilon \in \mathcal{A}^{\mathcal{B}}(s)$ **执行**
      Set $\tilde{b} \leftarrow P^{\mathcal{B}}(b, \epsilon)$
      $D \leftarrow D \cup (s_t, b, \epsilon, \mathbf{1}_{\tilde{b} \in B^*}, s_t, \tilde{b})$
    **结束循环**
  **结束循环**
**结束循环**
**返回** $D$

用于 Q学习的 LCER。无论智能体处于什么状态 $s \in \mathcal{S}^{\mathcal{M}}$，我们都可以假设智能体处于任何$b \in \mathcal{S}^{\mathcal{B}}$。这样，对于智能体执行的任何动作，我们都可以存储经验元组：

$$\{(s, b, a, r, s', \tilde{b}') \mid \forall b \in \mathcal{S}^{\mathcal{B}}\} \tag{14}$$

其中，$\tilde{b}' = P^{\mathcal{B}}(b,\ L^{\mathcal{M}}(s'))$ 是在状态 $(s,\ b)$ 中观察到带标签状态 $L(s')$ 后，本应发生的转移，且 $r = \mathbf{1}_{\tilde{b}' \in B^*}$。此外，我们可以添加所有跳转转移：

$$\{(s, b, \epsilon, r, s, \tilde{b}') \mid \forall b \in \mathcal{S}^{\mathcal{B}}, \forall \epsilon \in \mathcal{A}^{\mathcal{B}}(b)\} \tag{15}$$

由于跳转也不会影响马尔可夫决策过程。请注意，当我们添加跳转转移时，$s' = s$，因为跳转时只有极限确定性 Büchi 自动机的状态会发生变化。

**使用 LCER 进行策略梯度**。策略梯度的 LCER概要见算法 3。与 Q学习不同，策略梯度需要计算未来累计奖励：$R_k(\tau) = \sum_{i=k}^{T} \Gamma_i \mathbf{1}_{\{b_i \in \mathcal{S}^{\mathcal{B}^*}\}}$。因此，我们必须生成与 $P^{\mathcal{B}}$ 一致的完整轨迹，而不是像式（14）中那样生成独立的转移元组。下面将说明如何生成所有可行轨迹。

假设已收集到一条轨迹 $\tau = (s_0,\ b_0,\ a_0,\ \ldots\ldots,\ s_T,\ b_T)$。让我们移除跳转转移 $(s_i,\ b_i,\ a_i)$，其中$a_i \in \mathcal{A}^{\mathcal{B}}(b_i)$，并考虑该轨迹在马尔可夫决策过程 $\tau_{\mathcal{M}} = (s_0,\ s_1,\ \ldots\ldots,\ s_T)$ 上的投影。我们只能控制初始LDBA状态 $b_0$，因为轨迹序列中的所有其他自动机状态 $(b_1,\ \ldots\ldots,\ b_T)$ 都由 $\tau_{\mathcal{M}}$和 $b_{i+1} = P^{\mathcal{B}}(b_i,\ L^{\mathcal{M}}(s_i))$ 决定。

因此我们加入

$$\tilde{\mathcal{T}}(\tau) = \{(s_0, \tilde{b}_0, a_0, \ldots, s_T, \tilde{b}_T) \mid \forall \tilde{b}_0 \in \mathcal{S}^{\mathcal{B}}, \tilde{b}_i = P^{\mathcal{B}}(\tilde{b}_{i-1}, L^{\mathcal{M}}(s_i))\} \tag{16}$$

其中，轨迹之间只有极限确定性 Büchi 自动机状态不同。

**算法 3**用于策略梯度的 LCER

**参数：** 数据集 $D$。长度为 $T$ 的轨迹 $\tau$。

设定 $\tilde{\mathcal{T}}_0 \leftarrow \tilde{\mathcal{T}}(\tau)$
**对于** $k = 1, \ldots, T-1$ **执行**
  $\tilde{\mathcal{T}}_k \leftarrow \mathcal{E}(\tilde{\mathcal{T}}_{k-1})$
  **if** $\tilde{\mathcal{T}}_k == \mathcal{T}_{k-1}$ **然后**
    Set $\tilde{\mathcal{T}}_{T-1} \leftarrow \mathcal{T}_k$
    **中断** 结束条件判断
**结束循环**
设置 $D \leftarrow D \cup \tilde{\mathcal{T}}_{T-1}$
**返回** $D$

现在我们处理跳转转移。考虑某个 $\tilde{\tau} \in \mathcal{T}(\tau)$。回想一下，只要 $\mathcal{A}^{\mathcal{B}}(\tilde{b}_i)$ 非空，就可能发生跳转转移。这需要添加一条轨迹，该轨迹在跳转发生之前始终与 $\tilde{\tau}$完全相同。跳转发生后，后续的动作序列和马尔可夫决策过程状态序列相同，但极限确定性 Büchi 自动机状态不同。具体而言，假设 $b_i$ 有可用的跳转转移，$\epsilon \in \mathcal{A}^{\mathcal{B}}(\tilde{b}_i)$。那么：

$$\tilde{\tau}_{i,\epsilon} = (s_0, \tilde{b}'_i, a_0, \ldots, s_i, \tilde{b}'_i, \epsilon, s_i, \tilde{b}'_{i+1}, a_i, \ldots, s_T, \tilde{b}'_T) \tag{17}$$

其中 $\tilde{b}'_k = \tilde{b}_i$ 对于 $k \le i$ 和 $\tilde{b}'_k = P^{\mathcal{B}}(\tilde{b}'_{k-1},\ L^{\mathcal{M}}(s_k))$ 否则。

我们必须添加所有可能存在的 $\tilde{\tau}'_{i,\ \epsilon}$。令 $\mathcal{E}$ 为向现有序列中添加跳转的算子：

$$\mathcal{E}(\tilde{\mathcal{T}}(\tau)) = \tilde{\mathcal{T}}(\tau) \cup \{\tilde{\tau}_{i,\epsilon} \text{ from Eq (17)} \mid \forall \tilde{\tau} \in \tilde{\mathcal{T}}(\tau), \exists b_i \in \tilde{\tau} \text{ s.t. } \exists \epsilon \in \mathcal{A}^{\mathcal{B}}(b_i)\}. \tag{18}$$

由于 $\tau$的原始长度为 $T$，我们最多只能应用 $\mathcal{E}(\mathcal{E}(\ldots(\mathcal{E}(\mathcal{T}(\tau)))))$ $T$次。

**注 5.1**。 $\tau$ 的长度必须足够大，以确保极限确定性 *Büchi* 自动机有机会到达 $\mathcal{S}^{\mathcal{B}^*}$。一个充分条件是 $T \ge |\{b | \mathcal{A}^{\mathcal{B}}(b) \neq \varnothing\}|$，即具有跳转转移的极限确定性 *Büchi* 自动机状态数。

可以在策略的轨迹展开过程中构造出可行轨迹，而不是事后构造，详见附录 B。

## 6. 实验

我们在四个领域中进行实验，这些领域的线性时序逻辑公式、状态空间、动作空间和环境随机性各不相同，具体将在下一节总结。我们的目标是回答以下两个问题：(1) 我们能否获得符合预期、表现出满足线性时序逻辑策略行为的策略？(2) LCER 对学习性能有何影响？

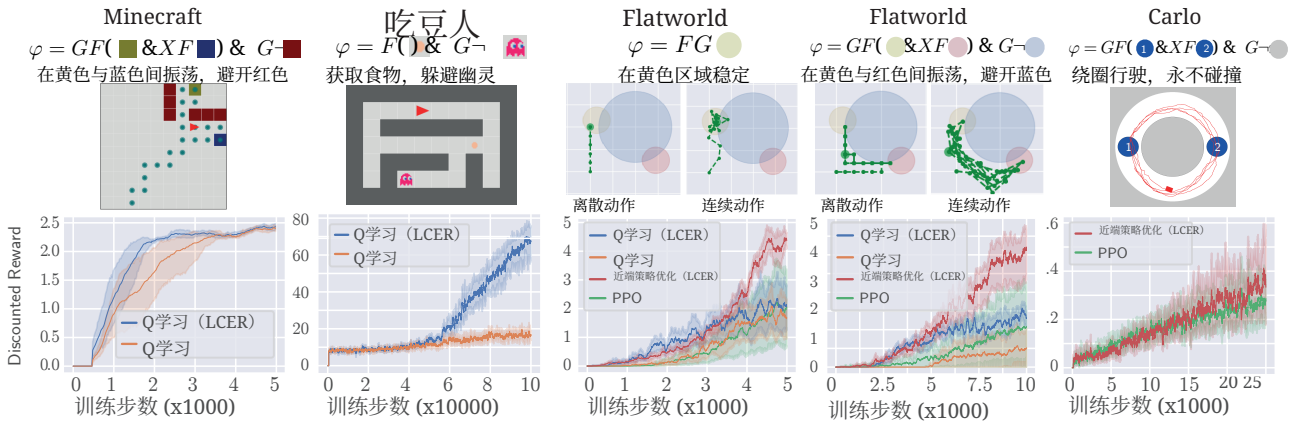


图2。 结果。 每一列对应一个环境以及一个我们希望智能体满足的LTL公式。每列的中间部分展示了环境和最终策略生成的一条轨迹（吃豆人除外，此处展示的是初始状态）。每列底部的学习曲线表明，使用LCER添加离策略数据能够显著提升经验性能。第一列：Minecraft，其中智能体应访问黄色和蓝色区域，同时避开红色区域。最终策略以蓝色点表示。第二列： 吃豆人，其中智能体应收集食物，同时避开鬼魂。第三列： Flatword，其中智能体最终应稳定在黄色区域内。当动作是离散的时，我们使用Q学习；当动作是连续的时，我们使用近端策略优化。第四列： 与第三列相同，但智能体应在黄色和红色区域之间振荡，同时避开蓝色区域。第五列： Carlo，其中智能体应通过无限次访问标记为1和2的蓝色区域，在不发生碰撞的情况下绕圈行驶。

### 6.1. 环境详情

**Minecraft** Minecraft 环境是一个 $10 \times 10$ 确定性网格世界，包含 5 个可用动作：左、右、上、下和不动。由红色三角形表示的智能体从单元格(9,2)出发。环境以及智能体的最终行为（以蓝色圆点表示）见图2（第一幅）。

**吃豆人** 吃豆人环境是一个 $5 \times 8$ 确定性网格世界，有 5 个可用动作：左、右、上、下、不动。由红色三角形表示的智能体从格子$(0,\ 3)$开始。幽灵以 0.8 的概率追逐智能体，并以 0.2 的概率随机采取动作，因此该环境具有随机性。环境的起始位置见图2（第二幅）。

**Flatworld** Flatworld 环境（见图 2第三幅和第四幅）是一个二维连续世界。由绿色圆点表示的智能体从 $(-1,\ -1)$开始。该世界的动力学由 $x' = x + a/10$ 给出，其中 $x \in \mathbb{R}^2$ 和 $a \in [0,1]^2$。我们还可以通过设置 5 个动作（右、上、左、下、不动）使动作空间离散化，智能体在相应方向上执行全油门动作。

**Carlo** Carlo 环境（如图 2 Fifth 所示）是一个简化的自动驾驶模拟器，使用自行车模型描述动力学。智能体可以观测自身的位置、速度和以弧度表示的航向，共计 5 个维度。智能体可以控制航向和油门，动作空间为 $[-1,1]^2$。对于该领域，我们选择使用环形赛道，智能体从道路中央以角度 $\{\pi(1+2i)/4\}_{i=0}^{3}$ 起步，并沿逆时针方向绕圈行驶，且不发生碰撞。

### 6.2. 方法与基线

当动作空间为离散时，我们采用搭配 LCER 的 Q 学习；否则，我们采用搭配 LCER 的近端策略优化。作为对比的基线是采用相同方法但不使用 LCER。这样我们就可以验证 LCER 对性能的影响程度。我们还在图 2的每列中间绘制了各环境下最终策略的一条轨迹；吃豆人环境除外，因为若不使用视频，很难直观展示幽灵与吃豆人之间的交互。

**处理** $\mathcal{A}$。 对于近端策略优化，智能体的策略是在连续动作空间上定义的高斯分布（与标准实现相同）。为了处理马尔可夫决策过程中连续动作空间下的极限确定性 Büchi 自动机中的跳转转移，我们首先让智能体决定是否执行跳转转移（即进行一次概率性抛硬币）。如果智能体选择不执行，则按照高斯分布采样动作。抛硬币的概率和高斯分布均通过学习得到。对于近端策略优化的重要性采样项，我们会调整 $\pi$ 的密度，以计入抛硬币操作。更多详情请参见附录 A。

### 6.3. 结果

**我们能否实现预期行为?** 答案是响亮的肯定。对于每个环境（吃豆人除外）

我们在图中每条学习曲线上方展示了最终策略的轨迹2。确定最终策略满足规范的概率目前仍是一个具有挑战性的开放问题（有限状态动作空间除外）。尽管如此，在每个环境中，智能体都能定性地完成任务。即使面对具有连续动作空间的困难任务，智能体也能够学会完成线性时序逻辑规范。

**LCER 有助于学习过程吗?** 根据图2最后一行的学习曲线，LCER 显然加快了学习速度。除 Carlo 之外的所有环境中，LCER 相较于不使用经验回放，带来了显著提升。

**为什么LCER 会有帮助?** 理解极限确定性 Büchi 自动机的一种方式，是把它看作完成任务所需步骤的课程。通过将智能体的极限确定性 Büchi 自动机状态替换为另一个设想的极限确定性 Büchi 自动机状态，我们就能让智能体“假装”自己已经完成了任务的一部分。

例如，考虑图2中的 Flatworld 示例，其中包含 $\varphi = GF(y \,\&\, XF(r)) \,\&\, (G\neg b)$。基准智能体（不使用LCER）必须完成整个任务才能获得任何奖励。然而，使用反事实数据的智能体只需从极限确定性 Büchi 自动机的状态 $y$ 0 出发，到达 0 （极限确定性 Büchi 自动机见图1 ）。智能体熟练掌握如何到达 $y$后，还需要学会如何从状态$r$ 2 到达 2。独立完成这两项任务后，智能体就解决了整个任务。将智能体置于极限确定性 Büchi 自动机的状态0 ，实际上就是让智能体假装自己已经到达了 $r$。从这个意义上说，任务的一部分已经完成。

## 7. 相关工作

**寻找满足 LTL 的策略**。在寻找满足 LTL 的策略的尝试中，当动力学未知时，Q 学习方法是主要选择；当动力学已知时，则主要采用线性规划方法（Sadigh 等人，2014年；Hasanbeig 等人，2018年；Bozkurt 等人，2020年；Cai 等人，2021b；Ding 等人，2014年）。Q 学习方法主要局限于有限状态-动作空间。在扩展到连续动作空间的研究中（Hasanbeig 等人，2020年），研究者采用深度确定性策略梯度（DDPG），并将其用于分层强化学习；众所周知，这种方法可能会找到短视的策略（Toro Icarte 等人，2022年）。

处理一部分线性时序逻辑规范（包括那些可表示为有限表达式的规范）也可以通过奖励机来实现（Toro Icarte 等人，2022年；Camacho 等人，2019年；Vaezipoor 等人，2021年）。我们的工作处理 $\omega$-正则表达式，并涵盖正则表达式。许多问题是 $\omega$-正则问题，但并非正则问题，例如活性问题——性（某件好事最终会发生）和安全性（坏事永远不会发生）。

**关于公式(3)的表述**。在将价值函数定义为访问次数$\mathcal{S}^{\mathcal{B}}$和状态依赖 $\Gamma_i$ 的函数方面，值得注意的先前工作包括 Bozkurt 等人(2020年)；Cai 等人(2021a)。尤为重要的是，这些作者使用多个彼此关系复杂的状态依赖折扣率，并且这些关系需要在 $\gamma \to 1^{-}$时满足。我们的工作大幅简化了这一问题，去除了技术性假设，同时强化了理论保证。这使我们能够得到策略的 LTL 满足概率相对于 $\gamma$的次优性的非渐近依赖关系。

**离策略数据**。可以将Toro Icarte 等人(2022年) 中的反事实样本视为 LCER 的一种实例，但其仅适用于有限LTL表达式和离散动作空间。扩展到连续动作空间和完整LTL需要谨慎处理。在连续动作和完整LTL设置下，(Wang 等人，2020年) 通过让智能体从不同的初始LDBA状态（而非 $b_{-1}$）开始来实现这一点；不过，这仍然是从不同起始状态进行的在策略方法，并未充分利用整个LDBA结构。这项工作可视为对我们工作的补充。

**理论**。具有策略满足强理论保证的研究包括Fu 和 Topcu(2014年)；Wolff 等人(2012年)；Voloshin 等人(2022年)，但它们同样仅适用于离散状态/动作空间。将这些工作扩展到连续状态空间并非易事，因为它们大量依赖离散马尔可夫链结构。

## 8. 讨论

据我们所知，我们的工作首次充分利用极限确定性 Büchi 自动机作为一种经验回放形式，也是首次使用策略梯度来学习满足LTL的策略。与大多数以往工作不同，我们提出的最终折扣形式不局限于 LTL的有限片段。

尽管最终折扣为我们提供了保证，但一般而言，式（2）中的问题并不是 PAC可学习的（Yang 等人，2021年）。不过，与 SAT求解器类似，对于难以解决的问题，寻找合理的启发式方法仍然是有益的。我们将表明，在特定情况下，最终折扣能够对（2）中所关注的量提供信号；即使它失效，也会选择一个与成功策略难以区分的策略。此外，我们推测，第4.2 节讨论的坏消息对于一般的线性时序逻辑规范而言是不可避免的，除非对马尔可夫决策过程作出重要假设。例如，对于线性时序逻辑中的稳定性问题，若假设系统具有控制仿射动力学，则李雅普诺夫函数可以作为策略满足线性时序逻辑规范的证书。对此，一个合理的放宽方式是要求系统在一段较长但有限的时间内以某种特定方式运行。

## 参考文献

Abel、D.，Dabney、W.，Harutyunyan、A.，Ho、M.K.，Littman、M.，Precup、D. 和 Singh、S.，《马尔可夫奖励的表达能力》。载于 神经信息处理系统进展，2021年。网址：
h
t
t
p
s://
proceedings.neurips.cc/paper/2021/file/4079016d940210b4ae9ae7d41c4a2065-Paper

. pdf.

Baier, C. 和 Katoen, J.-P. 模型检测原理。麻省理工学院出版社，剑桥，马萨诸塞州，2008年。ISBN 978-0- 262-02649-9。

Bozkurt, A. K., Wang, Y., Zavlanos, M. M. 和 Pajic, M. 使用无模型强化学习，根据线性时序逻辑规范进行控制综合。载于 *2020*年 *IEEE*机器人与自动化国际会议 *(ICRA)*，第 10349–10355 页，2020年。doi: 10.1109/ICRA40945.2020.9196796。

Cai, M., Hasanbeig, M., Xiao, S., Abate 和 Kan, Z. 将模块化深度强化学习用于结合时序逻辑的连续运动规划。*IEEE*机器人与自动化快报，6(4):7973–7980，2021a。

Cai, M., Xiao, S., Li, Z. 和 Kan, Z. 带有潜在不可行线性时序逻辑约束的最优概率运动规划。*IEEE*自动控制汇刊，第 1–1 页，2021年b。doi: 10.1109/TAC.2021.3138704。

Camacho, A., Toro Icarte, R., Klassen, T. Q., Valenzano, R. 和 McIlraith, S. A. 线性时序逻辑及其他：用于强化学习奖励函数规范的形式语言。载于第二十八届国际人工智能联合会议论文集（*IJCAI-19*），第 6065–6073 页。国际人工智能联合会，2019年7月。doi: 10.24963/ijcai.2019/840。网址：https://doi.org/10.24963/ijcai.2019/840。

Ding, X., Smith, S. L., Belta, C. 和 Rus, D. 具有线性时序逻辑约束的马尔可夫决策过程最优控制。*IEEE*自动控制汇刊，59(5):1244–1257，2014年5月。ISSN 0018-9286, 1558-2523。doi: 10.1109/TAC.2014.2298143。

Fu, J. 和 Topcu, U. 基于时序逻辑约束的可能近似正确马尔可夫决策过程学习与控制。载于 Fox, D.、Kavraki, L. E. 和 Kurniawati, H.（编），机器人：科学与系统*X*，加利福尼亚大学伯克利分校，美国，*2014*年*7*月*12*日至*16*日，2014年。doi: 10.15607/RSS.2014.X. 039。
URLhttp://www.roboticsproceedings.org/rss10/p39.html。

Hahn, E. M., Li, G., Schewe, S., Turrini, A°，一个d Zhang, L。无需确定化的惰性概率模型检测。*arXiv*预印本 *arXiv:1311.2928*，2013年。

Hasanbeig, M., Abate, A. 和 Kroening, D. 逻辑约束强化学习，2018年。URL https://arxiv.org/abs/1801.08099。

Hasanbeig, M., Kroening, D. 和 Abate, A. 基于时序逻辑的深度强化学习。发表于形式化建模与定时系统分析国际会议，第1–22页。Springer，2020年。

Hasselt, H. v., Guez, A. 和 Silver, D. 使用双重Q学习进行深度强化学习。载于第三十届*AAAI*人工智能会议论文集，AAAI'16，第 2094–2100 页。AAAI 出版社，2016年。

Ibarz, B., Leike, J., Pohlen, T., Irving, G., Legg, S. 和 Amodei, D. 在 Atari 中根据人类偏好和示范学习奖励。神经信息处理系统进展，第 31 卷，2018年。

Kingma, D. P. 和 Ba, J. Adam：一种随机优化方法。载于 Bengio, Y. 和 LeCun, Y.（编），第三届国际学习表征会议（*ICLR 2015*），美国加利福尼亚州圣迭戈，*2015*年*5*月*7*日至*9*日，会议论文集，2015年。网址：http://arxiv.org/abs/1412.6980。

Kˇret´ınsk`y, J.、Meggendorfer, T. 和 Sickert, S. Owl：一个用于 $\omega$-词、自动机和线性时序逻辑的库。载于自动化验证与分析技术国际研讨会，第 543–550 页。Springer，2018年。

Lin, L.-J. 基于强化学习、规划和教学的自我改进反应式智能体。机器学习，8：293–321，1992年。

Mnih, V.、Kavukcuoglu, K.、Silver, D.、Graves, A.、Antonoglou, I.、Wierstra, D. 和 Riedmiller, M. 使用深度强化学习玩 Atari。*arXiv* 预印本 *arXiv:1312.5602*，2013年。

Mnih, V.、Kavukcuoglu, K.、Silver, D.、Rusu, A. A.、Veness, J.、Bellemare, M. G.、Graves, A.、Riedmiller, M.、Fidje- land, A. K.、Ostrovski, G. 等。通过深度强化学习实现人类水平控制。《自然》，518（7540）：529–533，2015年。

Ng，A. Y.、Harada，D. 和 Russell，S. 奖励变换下的策略不变性：奖励塑形的理论与应用。载于*Icml*，第 99 卷，第 278–287 页，1999 年。

Puterman，M. L.《马尔可夫决策过程：离散随机动态规划》。约翰·威立父子公司，2014年。

Randløv，J. 和 Alstrøm，P. 使用强化学习和塑形学习骑自行车。载于*ICML*，1998年。

Sadigh, D., Kim, E. S., Coogan, S., Sastry, S. S., 和 Seshia, S. A. 一种基于学习的方法，用于针对线性时序逻辑规范进行马尔可夫决策过程的控制综合。载于第*53*届*IEEE*决策与控制会议，第1091–1096页，2014年。doi：10.1109/CDC.2014. 7039527。

Sickert, S., Esparza, J., Jaax, S., 和 Kˇret´ınsk´y, J. 用于线性时序逻辑的极限确定性 b¨uchi 自动机。载于 Chaudhuri, S. 和 Farzan, A.（编），计算机辅助验证，第 312–332 页，Cham，2016年。Springer 国际出版公司。ISBN 978-3-319-41540-6。

Silver, D., Singh, S., Precup, D. 和 Sutton, R. S. 奖励已足够。人工智能，299:103535，2021年。Sorg, J. 最优奖励问题：为受限智能体设计有效奖励。博士论文，密歇根大学，美国，2011年。URLhttps://handle. net/2027.42/89705。Sutton, R. S. 和 Barto, A. G. 强化学习：导论。麻省理工学院出版社，第二版，2018年。URL http://incompleteideas.net/book/the-book-2nd.html。

Toro Icarte, R., Klassen, T. Q., Valenzano, R. 和 McIlraith, S. A. 奖励机：利用强化学习中的奖励函数结构。*J. Artif. Int. Res.*，73，2022年5月。ISSN 1076-9757。doi: 10.1613/jair.1. 12440。URLhttps://doi.org/10.1613/jair.1.12440。

Toromanoff, M., Wirbel, E. 和 Moutarde, F. 深度强化学习在Atari上真的达到超人水平了吗？拉平竞争环境。*arXiv*预印本*arXiv:1908.04683*，2019年。Vaezipoor, P., Li, A. C., Icarte, R. T. 和 McIlraith, S. A. Ltl2action：为多任务强化学习泛化线性时序逻辑指令。载于第*38*届国际机器学习会议论文集，*ICML*，第139卷机器学习研究论文集，第10497–10508页，2021年。URL http://proceedings.mlr.press/v139/vaezipoor21a.html。

Voloshin, C., Le, H. M., Chaudhuri, S. 和 Yue, Y. 具有线性时序逻辑约束的策略优化。载于 Oh, A. H., Agarwal, A., Belgrave, D. 和 Cho, K.（编），神经信息处理系统进展，2022年。URLhttps://openreview.net/forum?id=yZcPRIZEwOG。

Wang, C., Li, Y., Smith, S. L. 和 Liu, J. 使用深度神经网络进行满足时序逻辑规范的连续运动规划，2020年。URL https://arxiv.org/abs/2004.02610。

Wolff, E. M., Topcu, U. 和 Murray, R. M. 具有时序逻辑规范的不确定马尔可夫决策过程的鲁棒控制。载于*2012*年*IEEE*第*51*届决策与控制会议（*CDC*），第3372–3379页，2012年。doi: 10.1109/CDC.2012.6426174。

Yang, C., Littman, M. L. 和 Carbin, M. 针对一般线性时序逻辑目标的强化学习是不可处理的。*CoRR*，abs/2111.12679，2021年。URL https://arxiv.org/abs/2111.12679。

Zhang, B., Rajan, R., Pineda, L., Lambert, N., Biedenkapp, A., Chua, K., Hutter, F. 和 Calandra, R. 关于超参数优化对基于模型的强化学习的重要性。在人工智能与统计国际会议，2021年。

# A. 实验

## A.1. 环境详情

环境和实验详情总结于表1。

表1. 环境详情

| 环境 | 实验 | $\mathcal{S}^{\mathcal{M}}$ | $\mathcal{A}^{\mathcal{M}}$ | 动力学 | LTL公式 |
|---|---|---|---|---|---|
| Minecraft | Q学习 | 离散 | 离散 | 确定性 | $GF(y \,\&\, XF(b))\ \&\ (G\neg r)$ |
| 吃豆人 | Q学习 | 离散 | 离散 | 随机性 | $F$(食物) & ($G\neg$幽灵) |
| Flatworld 1 | Q学习 | $\mathbb{R}^2$ | 离散 | 确定性 | $FGy$ |
| Flatworld 2 | Q学习 | $\mathbb{R}^2$ | 离散 | 确定性 | $GF(y \,\&\, XF(r))\ \&\ (G\neg b)$ |
| Flatworld 3 | PPO | $\mathbb{R}^2$ | $[0,1]^2$ | 确定性 | $FGy$ |
| Flatworld 4 | PPO | $\mathbb{R}^2$ | $[0,1]^2$ | 确定性 | $GF(y \,\&\, XF(r))\ \&\ (G\neg b)$ |
| Carlo | PPO | $\mathbb{R}^5$ | $[-1,1]^2$ | 确定性 | $GF$(区域1 & $XF$(区域2)) & ($G\neg$崩溃) |

## A.2. 实验设置

每个实验 均使用10 个随机种子运行。图 2 中的结果是对这些随机种子的平均值。

**Q-learning 实验**。 令 $k$ 为某个极限确定性 Büchi 自动机状态中可用的跳转转移数量的最大值 $k = \max_{b\in\mathcal{S}^{\mathcal{B}}} |\mathcal{A}^{\mathcal{B}}(b)|$。令 $m = \max_{s\in\mathcal{S}^{\mathcal{M}}} |\mathcal{A}^{\mathcal{M}}(s)|$。神经网络 $Q_\theta(s)$以 $s \in \mathcal{S}^{\mathcal{M}}$ 为输入，并为每个 $b \in \mathcal{S}^{\mathcal{B}}$输出一个$m+k$维向量$\mathbb{R}^{(m+k)\times|\mathcal{S}^{\mathcal{B}}|}$。对我们的目的而言，我们将 $Q_\theta(s,b)$视为与极限确定性 Büchi 自动机当前特定状态相对应的单个$m+k$维向量 $b$。

当 $\mathcal{S}^{\mathcal{M}}$ 为离散时，我们将 $Q_\theta(s,b)$参数化为一个表。否则， $Q_\theta(s,b)$由 3 个线性层参数化，隐藏维度为 128，中间使用 ReLU 激活，且没有最终激活函数。在根据 $b$中存在的跳转转移数量进行掩码处理后，我们可以选择 $\arg\max_{i\in[0,\dots,|\mathcal{A}^{\mathcal{B}}(b)|]} Q_\theta(s,b)_i$，以概率 $1-\eta$选择最高的 $Q$-值，并以 $\eta$ 概率均匀选择。这里， $\eta$ 初始化为 $\eta_0$，并以指定的频率线性（或指数）衰减（见表2）。

每个回合（在长度为 $T$ 的轨迹展开后），我们执行 $K$ 个梯度步数，每次使用不同的批次，批次大小见表 3。我们使用 Adam 优化器（Kingma & Ba，2015年），学习率也由表中指定。
在连续状态空间中，我们采用带目标网络的双重深度Q网络（Hasselt 等人，2016年）（而非深度Q网络），目标网络以表 3 所指定的频率进行更新。

表2。 Q学习实验的超参数（离散动作空间）

| | $\eta$ | | $\eta$ 衰减 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 实验 | $\eta_0$ | 最小 $\eta$ | Type | Rate | Freq | 批大小 | $K$ （批次） | LR | 目标更新 | T | $\gamma$ |
| Minecraft | .3 | 0 | 指数 | .9 | 100 | 128 | 20 | - | - | 100 | .99 |
| 吃豆人 | .4 | 0 | 线性 | .05 | 400 | 512 | 200 | - | - | 100 | .999 |
| Flatworld 1 | .8 | .15 | 指数 | .9 | 100 | 128 | 5 | .001 | 15 | 20 | .95 |
| Flatworld 2 | .8 | .15 | 指数 | .9 | 100 | 128 | 5 | .001 | 15 | 50 | .95 |

**PPO 实验**。 设 $k$ 为某个极限确定性 Büchi 自动机状态 $k = \max_{b\in\mathcal{S}^{\mathcal{B}}} |\mathcal{A}^{\mathcal{B}}(b)|$中可用的跳转转移数量的最大值。神经网络 $f_\theta(s)$以 $s \in \mathcal{S}^{\mathcal{M}}$ 作为输入，并为每个 $b \in \mathcal{S}^{\mathcal{B}}$输出一个 $\mathbb{R}^{(k+2)\times|\mathcal{S}^{\mathcal{B}}|}$ $(k+2)$ 维向量。就我们的目的而言，我们将 $f_\theta(s,\ b)$ 视为与极限确定性 Büchi 自动机当前特定状态对应的单个 $(k+2)$ 维向量$b$。

$f_\theta(s,b)$由 3 个线性层参数化，隐藏维度为 64，中间使用 ReLU 激活。第一个维度对应于采样高斯动作 $a \sim \mathcal{N}(f_\theta(s,b)[0], \mathrm{diag}(\sigma^2))$，其中 $\sigma$ 初始化为 $\sigma_0$（见表 3），并且每 10 个回合按指数衰减（衰减速率见表中所示）。其余的 $k+1$ 个维度（经过适当掩码以适应 $|\mathcal{A}^{\mathcal{B}}(b)|$ 的大小并应用 Softmax 后）表示采取马尔可夫决策过程动作 $a$ 或某个跳转转移 $\epsilon_i$的概率 $p = [p_a, p_{\epsilon_0}, \dots, p_{\epsilon_k}]$。我们从分类分布($p$)变量中采样，以选择是否返回

$a \sim \mathcal{N}(\text{Tanh}(f_\theta(s,\ b)[0]),\ \text{diag}(\sigma^2))$，或针对某个 $i$ 的 $a = \epsilon_i$。当选择 $a$ 时，可通过将 $p_a$ 与高斯密度相乘来计算密度；否则为 $p_{\epsilon_i}$。

对于评论家网络，我们使用参数化网络 $f_\phi(s,\ b) \to \mathbb{R}$，该网络由 3 个线性层组成，隐藏维度为 64，中间使用 Tanh 激活函数，且没有最终激活函数。

每个回合中（在长度为 $T$ 的轨迹展开之后），我们会进行 5 个梯度步数，每次使用的批次大小如表 3 所示。PPO 中的重要性采样项会裁剪至 $1 \pm .4$。评论家学习率为 。01。行动者网络和评论家网络均使用 Adam 优化器（Kingma &Ba，2015年）。

表 *3*。 PPO 实验的超参数（连续动作空间）

| 实验 | $\sigma_0$ | $\sigma$ 衰减速率 | 最小 $\sigma$ | 批大小 | 行动者学习率 | T |
|---|---|---|---|---|---|---|
| Flatworld 3 | 1.8 | .98 | .3 | 128 | .001 | 20 |
| Flatworld 4 | 1.8 | .99 | .1 | 128 | .001 | 50 |
| Carlo | .5 | .999 | .3 | 16 | 。0001 | 500 |

## B. 在轨迹展开过程中为策略梯度构造可行轨迹

假设我们想在策略进行轨迹展开的同时实时生成可行轨迹。也就是说，通过在 $P$ 中运行 $\pi$，我们得到如下形式的部分轨迹：$\tau_t = (s_0,\ b_0,\ a_0,\ \ldots\ldots,\ s_t,\ b_t)$。令 $a_t = a \in \mathcal{A}$ 表示 $\pi$ 执行的第 $t$ 个动作，令 $s_{t+1} = s' \in \mathcal{M}$ 表示在马尔可夫决策过程中观测到的下一个状态。

令 $\mathcal{T}_t$ 为时间步 $t$ 处当前的可行（部分）轨迹集合。元素 $\tau_k = (s_0,\ b_0,\ a_0,\ \ldots\ldots,\ s_k,\ b_k) \in \mathcal{T}_t$ 表示长度为 $k$ 步的（部分）轨迹，但不一定属于在 $\pi$ 的轨迹展开过程中观察到的轨迹。这里，$k \geq t$。因此，对于每个 $\tau_k \in \mathcal{T}_t$，以下 4 种情况之一成立：

**情况 1.** 动作 $a$ 不是跳转转移（即 $a \in \mathcal{A}^{\mathcal{M}}(s_k)$），并且在 $b_k$ ($\mathcal{A}^{\mathcal{B}}(b_k) = \varnothing$) 中没有可用的跳转转移。此时，我们可以构造拼接序列：$\tau_{k+1} = \tau_k \cup (a,\ s',\ b_{k+1})$，其中 $b_{k+1} = P^{\mathcal{B}}(b_k,\ L^{\mathcal{M}}(s'))$。我们设定 $\mathcal{T}_\epsilon = \varnothing$。

**情况 2**。动作 $a$ 是一个跳转转移，并且当前在 $b_k$（即 $a \in \mathcal{A}^{\mathcal{M}}(b_k)$）中可行。于是，我们可以构造拼接轨迹 $\tau_{k+1} = \tau_k \cup (a,\ s',\ b_{k+1})$，其中 $b_{k+1} = P^{\mathcal{B}}(b_k,\ a)$。我们设定 $\mathcal{T}_\epsilon = \varnothing$。

**情况 3**。动作 $a$ 不是跳转转移（即 $a \in \mathcal{A}^{\mathcal{M}}(s_k)$），但在 $b_k$ 中至少存在一个可行的跳转转移（即 $\mathcal{A}^{\mathcal{B}}(b_k) \neq \varnothing$）。因此，除了根据情况 1 构造 $\tau_{k+1}$ 之外，我们还拥有所有可能的跳转：

$$\mathcal{T}_\epsilon = \{\tau_k \cup (\epsilon, s_k, b_{k+1}, a, s', b_{k+2}) | \forall \epsilon \in \mathcal{A}^{\mathcal{B}}(b_k), b_{k+1} = P^{\mathcal{B}}(b_k, \epsilon), b_{k+2} = P^{\mathcal{B}}(b_{k+1}, a_t)\}$$

**情况 4**。动作 $a$ 是一个跳转转移，但在 $b_k$ 中不可行（即 $a \notin \mathcal{A}^{\mathcal{B}}(b_k)$）。在这种情况下，我们直接跳过这条轨迹。设定$\tau_{k+1} = \tau_k$ 和 $\mathcal{T}_\epsilon = \varnothing$。

遍历完 $\tau_k \in \mathcal{T}_t$ 中的每个元素并构建 $\tau_{k+1}$ 和 $\mathcal{T}_\epsilon$ 后，我们可以更新当前的可行轨迹集合：

$$\mathcal{T}_{t+1} = \cup_{\tau_k \in \mathcal{T}_t} \Big( (\mathcal{T}_t \setminus \{\tau_k\}) \cup \{\tau_{k+1}\} \cup \mathcal{T}_\epsilon \Big) \tag{19}$$

简单来说，这个过程会将 $\tau_k$ 替换为 $\tau_{k+1}$，并在存在跳转转移时将其一并添加。该算法见算法4。

**算法 4**用于策略梯度的 LCER（选项 2）

**参数：** 数据集 $D$。长度为 $T$ 的轨迹 $\tau$。

1: 设置 $\mathcal{T}_0 \leftarrow \{(s_0,\ b) | b \in \mathcal{B}\}$2:
**对于** $(s_t,\ a_t,\ s_{t+1}) \in \tau$ **执行**3:
根据公式（19）构建 $\mathcal{T}_t$ 4: **结束**
**循环**5: 设置 $D \leftarrow D \cup \mathcal{T}_T$6: **返回** $D$