基于线性时序逻辑规范的样本高效无模型强化学习及其最优性保证

Daqian Shao 和 **Marta Kwiatkowska**
计算机科学系，牛津大学，英国{daqian.shao, marta.kwiatkowska}@cs.ox.ac.uk

arXiv:2305.01381v2 [cs.LG] 3 May 2023

摘要

线性时序逻辑（LTL）广泛用于描述系统策略的高级目标，因此自主系统能够针对这类规范学习最优策略尤为重要。然而，从线性时序逻辑规范中学习最优策略并非易事。我们提出了一种无模型强化学习（RL）方法，能够针对一个未知的随机系统高效地学习最优策略，该系统以马尔可夫决策过程（MDP）建模。我们提出了一种新颖且更通用的乘积马尔可夫决策过程、奖励结构和折扣机制；将它们与现成的无模型强化学习算法结合使用时，能够高效地学习使满足给定线性时序逻辑规范的概率最大化的最优策略，并提供最优性保证。我们还改进了理论结果，阐明如何选择强化学习中的关键参数以确保最优性。为直接评估学习得到的策略，我们采用概率模型检测器 PRISM，计算该策略满足此类规范的概率。在不同线性时序逻辑任务上的多种表格型马尔可夫决策过程环境中进行的若干实验表明，该方法提高了样本效率，并促进了最优策略的收敛。

1 引言

线性时序逻辑（LTL）是一种时序逻辑语言，可用于编写描述逻辑命题无限序列性质的公式。LTL 广泛应用于机器人学和多智能体系统高层目标的形式化规范，因此，系统或系统中的智能体需要学习针对这些高层规范的策略。此类系统被建模为马尔可夫决策过程（MDP）；如果已知 MDP 的状态和转移，则可以采用经典的策略综合技术。然而，当转移事先未知时，必须通过与 MDP 交互来学习最优策略。先验地

无模型强化学习 [Sutton and Barto,2018], 是一种强大的方法，可训练智能体选择动作从而在未知环境中随时间推移最大化奖励，因此非常适合用于线性时序逻辑规范策略学习。然而，由于难以确定何时、何地以及给予智能体多少奖励，利用基于奖励的强化学习来学习能够最大化满足线性时序逻辑规范概率的最优策略并非易事 [Alur 等人， 2021] 。为此，大多数研究采用的方法是先将线性时序逻辑规范转换为自动机，然后利用原始环境马尔可夫决策过程和该自动机构建乘积马尔可夫决策过程，并在其上应用无模型强化学习算法。然而，仍存在一个关键障碍：如何恰当地定义奖励，以引导智能体最优地满足线性时序逻辑规范。已有多种算法被提出用于学习线性时序逻辑规范 [Hahn 等人，2019年；Bozkurt等人，2019年；Hasanbeig 等人， 2019] 这些算法为保证最优性，所选关键参数依赖于对环境马尔可夫决策过程的假设或认知。这些研究通常只证明最优性保证参数的存在，或给出对这些参数不必要地苛刻的界限，这可能导致学习效率低下。此外，即使掌握了对环境马尔可夫决策过程的某些认知，如何明确选择这些参数仍不清楚。再者，实验中对假设参数的评估是间接进行的，要么检查价值函数，要么比较获得的期望奖励，因此难以针对一般马尔可夫决策过程中的线性时序逻辑学习调优最优性参数。

在本文中，我们提出了一种新颖且更通用的乘积马尔可夫决策过程、奖励结构和折扣机制。借助无模型强化学习算法，该方法能够高效地学习最优策略，在保证条件下最大化满足线性时序逻辑规范的概率。我们在乘积马尔可夫决策过程和奖励结构的最优性方面给出了改进的理论结果，并通过更严格的分析得出了更优的最优性参数界限。此外，这项分析还阐明了如何根据环境中的马尔可夫决策过程明确选择最优性参数。我们还采用反事实想象，利用已知的高层线性时序逻辑规范，进一步提升算法性能。最后但同样重要的是，我们提出使用 PRISM 模型检查器 [Kwiatkowska等人， 2011] 直接评估学习所得策略的满足概率，从而提供一个可直接比较算法并调节关键参数的平台。我们在多种常见的马尔可夫决策过程环境中开展了实验，

涉及各种具有挑战性的线性时序逻辑任务，结果表明我们的方法具有更高的样本效率和更好的收敛性。

我们的贡献包括：（i）一种新颖的乘积马尔可夫决策过程设计，其中纳入了接受状态计数器和广义奖励结构；（ii）一种新颖的强化学习算法，可收敛到满足线性时序逻辑规范的最优策略，并提供理论最优性保证以及关于如何选择关键参数的理论分析结果；（iii）采用反事实想象，通过反事实推理生成想象经验，以利用线性时序逻辑规范的已知结构；以及（iv）通过在评估流程中创新性地集成概率模型检测器，直接评估所提出的算法，取得了有力的实证结果，表明其样本效率更高，训练收敛性更好。

相关工作

大多数基于奖励的强化学习线性时序逻辑学习工作都采用乘积马尔可夫决策过程，即环境马尔可夫决策过程与由线性时序逻辑规范转换得到的自动机的乘积。Sadigh 等人 [2014] 最先使用确定性Rabin自动机构建这一乘积，并采用折扣奖励设计来学习线性时序逻辑；之后的工作则采用了一种新的自动机设计——极限确定性 B¨uchi 自动机（极限确定性 Büchi 自动机）[Sickert 等人， 2016]。Hahn 等人 [2019] 采用由极限确定性 Büchi 自动机构成的乘积马尔可夫决策过程，并加入汇状态，将线性时序逻辑满足问题转化为极限平均奖励问题，同时提供最优性保证。Hahn 等人 [2020] 随后对此方法作了修改，加入两个折扣因子，并获得类似的最优性保证。Bozkurt等人 [2019] 提出了一种在乘积马尔可夫决策过程中进行折扣奖励学习的算法，并提供最优性保证；该算法根据对未知环境马尔可夫决策过程的某些假设来选择折扣因子。据我们所知，这些方法是目前仅有的能够为完整的无限时域线性时序逻辑学习提供最优性保证的方法。不过，仍有许多方法展示了线性时序逻辑学习方面的实证结果。Hasanbeig 等人 [2020； 2019] 提出将接受前沿函数作为乘积马尔可夫决策过程的奖励，而 Cai 等人 [2021] 则将这一奖励前沿扩展到了连续控制任务。

由于完整线性时序逻辑难以学习，许多方法着重于学习受限的有限线性时序逻辑变体。Giacomo 等人 [2013； 2019] 提出了有限轨迹线性时序逻辑变体及相应的强化学习算法；Littman 等人 [2017]为广义线性时序逻辑变体制定了一种学习算法；Aksaray 等人 [2016] 提出了学习信号时序逻辑的方法，而 Li 等人 [2016] 则提出了一种用于机器人学应用的截断线性时序逻辑变体。

另一类相关工作利用自动机来学习非马尔可夫奖励。Toro Icarte 等人 [2022； 2018] 定义了一种奖励机自动机，用于表示高层非马尔可夫奖励；而 Camacho 等人 [2019] 提出了一种通过将有限线性时序逻辑规范转换为奖励机来学习它们的方法。然而，奖励机的表达能力严格弱于线性时序逻辑。最后，还有一些工作利用其他高层逻辑规范来促进学习 [Andreas 等人，2016年；Jiang 等人，2021年；Jothimurugan 等人，2020年；Jothimurugan等人， 2021], 但这些工作与基于线性时序逻辑进行强化学习的相关性较低。

2 预备知识

在阐述我们的问题之前，我们先介绍马尔可夫决策过程、线性时序逻辑和强化学习的基础知识。

2.1 马尔可夫决策过程

定义 *1*（马尔可夫决策过程 [*Littman*， 2001]）。马尔可夫决策过程（*MDP*） $\mathcal{M}$ 是一个元组（S， s_0， A， T， $\mathcal{AP}$， L， r， γ），其中 S是有限状态集，$s_0 \in S$ 是初始状态，A 是有限动作集，T： $S \times A \times S \to [0,1]$ 是概率转移函数，$\mathcal{AP}$ 是原子命题集， L： $S \to 2^{\mathcal{AP}}$ 是命题标记函数， r： $S \times A \times S \to \mathbb{R}$ 是奖励函数， γ： $S \to (0,1]$ 是折扣函数。令 $A(s)$ 表示状态 s下可用动作的集合，则对于所有 $s \in S$，均有 $\sum_{s' \in S} T(s, a, s') = 1$，当且仅当 $a \in A(s)$ 成立，否则为 *0*。

无限路径是状态序列 $\sigma = s_0, s_1, s_2 \ldots$，其中存在 $a_{i+1} \in A(s_i)$，使得对所有 $i \geq 0$， $T(s_i, a_{i+1}, s_{i+1}) > 0$；有限路径则是这样的有限序列。我们将 MDP $\mathcal{M}$ 的无限路径集和有限路径集分别记为 $Paths^{\mathcal{M}}$ 和 $FPaths^{\mathcal{M}}$。我们用 $\sigma[i]$ 表示s_i，并用 $\sigma[:i]$ 和 $\sigma[i:]$ 分别表示路径的前缀和后缀。此外，我们假设存在自环：若某个状态 s满足$A(s) = \varnothing$，则令 $T(s, a, s) = 1$，其中存在某个$a \in A$，且 $A(s) = a$，从而所有有限路径都可以扩展为无限路径。

对于 $\mathcal{M}$ ，有限记忆策略 π 是一个函数 π ： $FPaths^{\mathcal{M}} \to D(A)$，满足 $supp(\pi(\sigma)) \subseteq A(\sigma[-1])$，其中 $D(A)$表示在 A上的分布， $supp(d)$表示该分布的支撑集， $\sigma[-1]$ 是有限路径 σ的最后一个状态。若策略 π 仅取决于当前状态，则称其为无记忆策略，即 当 $\sigma[-1] = \sigma'[-1]$ 时， $\pi(\sigma) = \pi(\sigma')$；若对所有 $\sigma \in FPaths^{\mathcal{M}}$， $\pi(\sigma)$都是点分布，则称该策略为确定性策略。对于确定性无记忆策略，我们令 $\pi(s)$表示 $\pi(\sigma)$，其中 $\sigma[-1] = s$。

令 $Paths_{\pi}^{\mathcal{M}} \subseteq Paths^{\mathcal{M}}$ 表示遵循策略 π 的无限路径子集，并按照标准方式在 $Paths_{\pi}^{\mathcal{M}}$ 上定义概率空间（$Paths_{\pi}^{\mathcal{M}}$, $\mathcal{F}_{Paths_{\pi}^{\mathcal{M}}}$, P_{π}）。然后，对于任意函数 f： $Paths_{\pi}^{\mathcal{M}} \to \mathbb{R}$，令 $\mathrm{E}_{\pi}[f]$ 表示在遵循 π 的 $\mathcal{M}$ 的无限路径上对 f 的期望。

由 $\mathcal{M}$ 和确定性无记忆策略 π 诱导的马尔可夫链（MC）是一个元组 $\mathcal{M}_{\pi} = (S, s_0, T_{\pi}, \mathcal{AP}, L)$，其中 $T_{\pi}(s, s') = T(s, \pi(s) s')$。马尔可夫链的底强连通分量（BSCC）是一个状态集合$C \subseteq S$，满足：对于任意状态对 $s_1, s_2 \in C$，存在一条沿转移函数 T_{π} 从 s_1 到 s_2 的路径（强连通）；并且不存在状态 $s' \in S \setminus C$ 使得$T_{\pi}(s, s') > 0$ 对所有 $s \in C$ 均成立（汇）。

2.2 线性时序逻辑

线性时序逻辑（LTL）为系统规范提供了一种高层次的描述方式。LTL 表达能力很强，能够描述具有无限时域的规范。

定义 2([Baier 和 Katoen, 2008])。 由原子命题构成的线性时序逻辑公式 $\mathcal{AP}$ 由以下语法定义：

$\varphi ::= true \mid p \mid \varphi_1 \wedge \varphi_2 \mid \neg\varphi \mid \mathsf{X}\,\varphi \mid \varphi_1 \mathsf{U}\, \varphi_2, \quad p \in \mathcal{AP},$

其中，X 表示下一个，U 表示直到。其他布尔运算符和时序运算符可按如下方式定义：或：$\varphi_1 \vee \varphi_2 = \neg(\neg\varphi_1 \wedge \neg\varphi_2)$；蕴含：$\varphi_1 \rightarrow \varphi_2 = \neg\varphi_1 \vee \varphi_2$；最终：$\mathsf{F}\ \varphi = true \mathsf{U}\ \varphi$；总是：$\mathsf{G}\ \varphi = \neg(\mathsf{F}\ \neg\varphi)$。

无限路径$\sigma \in Paths^{\mathcal{M}}$ 满足线性时序逻辑公式 φ 记作 $\sigma \models \varphi$，其定义通过对 φ的结构进行归纳给出：

$$\begin{aligned} \sigma \text{ satisfies } \varphi \text{ if } \quad & p \in L(\sigma[0]) && \text{for } p \in \mathcal{AP}; \\ & \sigma[1:] \models \varphi && \text{for } \mathsf{X}\,\varphi; \\ \exists i, \sigma[i] \models \varphi_2 \text{ and } \forall j < i, \sigma[j] \models \varphi_1 \quad & && \text{for } \varphi_1 \mathsf{U}\, \varphi_2, \end{aligned}$$

布尔运算符的满足关系按其默认含义定义。

2.3 强化学习

强化学习 [Sutton and Barto, 2018] 教会智能体在未知环境中从其动作空间中选择动作，以便随着时间推移最大化奖励。在大多数情况下，环境被建模为一个 MDP $\mathcal{M}$ =(S, s_0, A, T, $\mathcal{AP}$, L, r, γ)。给定一个确定性无记忆策略 π，在每个时间步 t，设智能体的当前状态为 s_t，则选择动作 $a = \pi(s_t)$，并从环境中获得下一个状态 $s_{t+1} \sim T(s_t, a, \cdot)$以及即时奖励 $r(s_t, a, s_{t+1})$。然后，从 $s \in S$ 和时间步 t开始，遵循 π 的期望折扣奖励为

$$G_t^{\pi}(s) = \mathbb{E}_{\pi}[\sum_{i=t}^{\infty}(\prod_{j=t}^{i-1}\gamma(s_j)) \cdot r(s_i, a_i, s_{i+1}) \mid s_t = s], \quad (1)$$

其中 $\prod_{j=t}^{t-1} := 1$。智能体的目标是学习使期望折扣奖励最大化的最优策略。请注意，我们定义的是折扣函数，而非恒定的折扣因子，因为对于我们提出的 LTL 学习算法而言，根据当前 MDP状态对奖励进行折扣至关重要。

Q学习 [Watkins 和 Dayan, 1992] 是一种广泛应用于无模型强化学习的方法。它采用 Q 函数$Q^{\pi}(s, a)$的概念，该函数表示在状态 s 下执行动作 a 并随后遵循策略 π 所获得的期望折扣奖励。所有最优策略对应的 Q 函数均满足贝尔曼最优性方程：

$$\begin{aligned} Q^*(s,a) = \sum_{s' \in S} T(s,a,s')\big(r(s,a,s') \\ +\gamma(s)\max_{a' \in A} Q^*(s',a')\big)\ \forall a \in A, s \in S. \end{aligned} \quad (2)$$

在 Q学习算法的每次迭代中，智能体的经验，即下一个状态 s' 和即时奖励$r(s, a, s')$，都会用于更新 Q 函数：

$$Q(s,a) \overset{\alpha}{\leftarrow} r(s,a,s') + \gamma(s)\max_{a' \in A} Q(s',a'), \quad (3)$$

其中，α 是学习率，$x \overset{\alpha}{\leftarrow} y$ 表示 $x \leftarrow x + \alpha(y - x)$。此外，可以通过在每个状态 s中选择状态-动作对价值最高的动作a，从最优 Q 函数 $Q^*(s, a)$中恢复最优策略。只要每个状态-动作对都被无限次访问，Q学习就会在极限情况下收敛到最优 Q 函数 [Watkins 和 Dayan, 1992]，从而学得最优策略。

3 我们的方法

我们的目标是提出一种无模型强化学习方法，以高效学习最优策略，在提供保证的同时最大化满足线性时序逻辑规范的概率。下面概述我们的方法。首先，我们将 LTL目标 φ 转换为极限确定性 B¨uchi 自动机。然后，我们引入一种新颖的乘积马尔可夫决策过程，并在其上定义广义奖励结构。基于这一奖励结构，我们提出一种采用折叠Q函数的 Q学习算法，以提供最优性保证的方式学习最优策略。最后，我们通过反事实想象增强该算法，利用自动机结构提升性能，同时保持最优性。

3.1 问题形式化

给定一个 MDP $\mathcal{M}$ =(S, s_0, A, T, $\mathcal{AP}$, L, r, γ)，其状态和转移未知，且具有线性时序逻辑目标 φ。对于该 MDP $\mathcal{M}$的任意策略 π，令 $\mathrm{P}_{\pi}(s \models \varphi)$ 表示从状态 s 出发、遵循 π 的路径满足线性时序逻辑公式 φ的概率：

$$\mathbb{P}_{\pi}(s \models \varphi) = \mathbb{P}_{\pi}\{\sigma \in Paths_{\pi}^{\mathcal{M}} \mid \sigma[0] = s, \sigma \models \varphi\}. \quad (4)$$

然后，我们希望设计一种无模型强化学习算法，学习一个确定性的无记忆最优策略 π_{φ}，使满足 φ的 $\mathcal{M}$的概率最大化：

$$\mathbb{P}_{\pi_{\varphi}}(s \models \varphi) = \max_{\pi} \mathbb{P}_{\pi}(s \models \varphi) \quad \forall s \in S. \quad (5)$$

3.2 极限确定性 B¨uchi 自动机

我们首先将线性时序逻辑规范转换为自动机。常见的自动机选择包括确定性 Rabin 自动机和非确定性 B¨uchi 自动机。在本工作中，我们采用一种称为极限确定性 B¨uchi 自动机（LDBA）的 B¨uchi 自动机变体 [Sickert等人, 2016]。

定义 3. 非确定性 *B¨uchi* 自动机是一个自动机 $\mathcal{A} = (\mathcal{AP}, \mathcal{Q}, q_0, \Delta, \mathcal{F})$，其中 $\mathcal{AP}$ 是原子命题的集合，$\mathcal{Q}$ 是有限状态集合，$q_0 \in \mathcal{Q}$是初始状态，$\mathcal{F} \subseteq \mathcal{Q}$ 是接受状态集合。设 $\Sigma = 2^{\mathcal{AP}} \cup \{\epsilon\}$ 是有限字母表，则转移函数定义为 $\Delta: \quad \mathcal{Q} \times \Sigma \rightarrow 2^{\mathcal{Q}}$。

定义 4 (极限确定性 Büchi 自动机)。 如果 $\mathcal{Q}$ 可以划分为确定性集合和非确定性集合，即 $\mathcal{Q} = \mathcal{Q}_{\mathcal{N}} \cup \mathcal{Q}_{\mathcal{D}}$，其中 $\mathcal{Q}_{\mathcal{N}} \cap \mathcal{Q}_{\mathcal{D}} = \varnothing$，则称 *B¨uchi* 自动机是极限确定性的，使得

1. $\mathcal{F} \subseteq \mathcal{Q}_{\mathcal{D}}$ 和 $q_0 \in \mathcal{Q}_{\mathcal{N}}$；
2. $|\Delta(q, \nu)| \leq 1$ 对所有 $q \in \mathcal{Q}_{\mathcal{N}}$和 $\nu \neq \epsilon \in \Sigma$；
3. $\Delta(q, \nu) \subseteq \mathcal{Q}_{\mathcal{D}}$ 和 $|\Delta(q, \nu)| \leq 1$ 对所有 $q \in \mathcal{Q}_{\mathcal{D}}$ 和$\nu \in \Sigma$;

极限确定性 Büchi 自动机是一种 Büchi 自动机，其中非确定性仅限于自动机的初始部分 $\mathcal{Q}_{\mathcal{N}}$。极限确定性 Büchi 自动机从非确定性初始部分开始，在到达接受状态后，通过 ϵ-转移进入确定性接受部分 $\mathcal{Q}_{\mathcal{D}}$，此后所有转移均为确定性的。我们遵循 Bozkurt等人 [2019]的表述，通过添加一种处理所有非确定性的 ϵ-转移来扩展字母表，这意味着只有 ϵ 能将自动机状态转移到多个状态：$|\Delta(q,\epsilon)| > 1$。这样，马尔可夫决策过程就可以与自动机同步；我们将在第 3.3 节中详细讨论这一点。

无限词 $\mathcal{W} \in \Sigma^\omega$（其中 Σ^ω 是字母表 Σ 上所有无限词的集合）被 B¨uchi 自动机 $\mathcal{A}$ 接受，当且仅当存在一个从 q_0 出发的无限自动机运行 $\theta \in \mathcal{Q}^\omega$，其中 $\theta[t+1] \in \Delta(\theta[t], \mathcal{W}[t])\ \forall t \geq 0$，且 $\inf(\theta) \cap \mathcal{F} \neq \varnothing$；其中 $\inf(\theta)$ 是该运行中被无限次访问的自动机状态集合 θ。

极限确定性 Büchi 自动机与 LTL 语言具有同等表达能力，并且可以在由线性时序逻辑规范 φ 导出的极限确定性 Büchi 自动机上评估任意给定线性时序逻辑规范的满足情况。我们使用 Rabinizer 4 [K˘ret´ınsk´y等人， 2018] 将线性时序逻辑公式转换为极限确定性 Büchi 自动机。图 1a 展示了由线性时序逻辑公式“FG a & G !c”导出的极限确定性 Büchi 自动机示例，其中状态 1 为接受状态。极限确定性 Büchi 自动机不同于奖励机 [Toro Icarte等人， 2022] 因为它们能够表达可由无限路径满足的性质，其表达能力严格强于奖励机，而且它们的接受条件也不同。

3.3 乘积马尔可夫决策过程

本节提出一种新颖的乘积马尔可夫决策过程，由环境马尔可夫决策过程、一个极限确定性 Büchi 自动机和一个整数计数器构成，各组成部分的转移同步进行。与文献中采用的标准乘积马尔可夫决策过程不同 [Bozkurt 等，2019年；Hahn 等，2019年；Hasanbeig 等， 2020], 这种新颖的乘积马尔可夫决策过程引入了一个计数器，用于统计从初始状态出发的路径所经过的接受状态数量。

定义 5 (乘积马尔可夫决策过程)。给定一个马尔可夫决策过程 $\mathcal{M} = (S, s_0, A, T, \mathcal{AP}, L, r, \gamma)$、一个极限确定性 *Büchi* 自动机 $\mathcal{A} = (\mathcal{AP}, \mathcal{Q}, q_0, \Delta, \mathcal{F})$以及 $K \in \mathbb{N}$，我们按如下方式构造乘积马尔可夫决策过程：

$$\mathcal{M}^\times = \mathcal{M} \times \mathcal{A} \times [0..K] = (S^\times, s_0^\times, A^\times, T^\times, \mathcal{F}^\times, r^\times, \gamma^\times),$$

其中，乘积状态 $S^\times = S \times \mathcal{Q} \times [0..K]$、初始状态$s_0^\times = (s_0, q_0, 0)$、乘积动作 $A^\times = A \cup \{\epsilon_q \mid q \in \mathcal{Q}\}$、接受集 $\mathcal{F}^\times = S \times \mathcal{F} \times [0..K]$ 以及乘积转移 $T^\times$：$S^\times \times A^\times \times S^\times \to [0,1]$ 定义如下：

$$T^\times((s,q,n),a,(\hat{s},\hat{q},n)) = \begin{cases} T(s,a,\hat{s}) & \text{if } a \in A \text{ and } \hat{q} \in \Delta(q, L(s)) \setminus \mathcal{F}; \\ 1 & \text{if } a = \epsilon_{\hat{q}}, \hat{q} \in \Delta(q,\epsilon) \text{ and } \hat{s} = s; \\ 0 & \text{otherwise}. \end{cases} \tag{6}$$

$$T^\times((s,q,n),a,(\hat{s},\hat{q},\min(n+1,K))) = \begin{cases} T(s,a,\hat{s}) & \text{if } a \in A \text{ and } \hat{q} \in \Delta(q, L(s)) \cap \mathcal{F}; \\ 0 & \text{otherwise}. \end{cases} \tag{7}$$

其中，所有其他转移均等于 *0*。适用于 *LTL* 学习的乘积奖励$r^\times$： $S^\times \times A^\times \times S^\times \to \mathbb{R}$ 和乘积折扣函数$\gamma^\times$： $S^\times \to (0,1]$ 将在定义 *6* 中给出。

此外，无限路径 σ *of* $\mathcal{M}^\times$ 满足 *Büchi* 条件 $\varphi_{\mathcal{F}}$，当且仅当 $\inf(\sigma) \cap \mathcal{F}^\times \neq \varnothing$。略微滥用一下记号，我们将这一条件用 *LTL* 语言表示为 $\sigma \models \mathsf{GF}\ \varphi_{\mathcal{F}}$，这意味着对于所有 $M \in \mathbb{N}$，总是存在 $s^\times \in \mathcal{F}^\times$，它们将在 $\sigma[M:]$中被访问。

在乘积马尔可夫决策过程中执行 MDP 动作 $a \in A$ $\mathcal{M}^\times$ 时，通过对当前环境 MDP 状态应用标记函数来推导用于 LDBA 转移的字母表：$L(s) \in 2^{\mathcal{AP}}$。在这种情况下，LDBA 转移 $\delta(q, L(s))$ 是确定性的。否则，如果执行了一个 ϵ-动作$\epsilon_{\hat{q}} \in \{\epsilon_q \mid q \in \mathcal{Q}\}$，则 LDBA 通过一个 ϵ-转移进行转移，并通过将自动机状态转移至 $\hat{q}$ 来消除 $\delta(q,\epsilon)$ 的非确定性。初始状态下， K 计数器值为 0；每当到达一个接受状态时，计数器值就增加 1，直至达到上限 K。

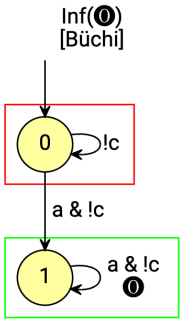
示例 1. 我们通过一个示例来说明定义 *5* 中的乘积马尔可夫决策过程结构。在图 *1b* 中，我们有一个网格环境，智能体可以选择向上、向下、向左或向右移动。任务是无限次访问标记为*“a”*的状态，同时不访问*“c”*，如图 *1a* 中的 *LDBA* 所述。马尔可夫决策过程从 *(1,0)* 开始，墙以实心灰色方块表示。第一行中的状态仅允许执行向右动作，如向右指的三角形所示；该动作会通向 *(0,9)* 处的一个汇。位于 *(2,0)* 的位置还有一道概率门，会随机使智能体向下或向右移动；如果智能体到达 *(2,1)*，则可以到达 *(2,9)* 处的接受汇。因此，最优策略是从起点向下移动；若位于 *(2,0)* 的概率门使智能体向右移动，则留在 *(2,9)*。满足该任务的概率等于概率门将智能体送往右侧的概率。直观而言，这个环境中有一些容易探索的初始接受状态，但它们会通向非接受汇；而真正的最优路径则需要更多探索。如果我们在此任务的乘积马尔可夫决策过程中设置 $K = 10$，就可以为最初访问的接受状态分配非常小的奖励，并随着访问到更多接受状态而逐步提高奖励，从而鼓励探索并引导智能体找到最优策略。

接下来，我们给出一个定理，指出带有 B¨uchi 条件 $\varphi_{\mathcal{F}}$ 的乘积马尔可夫决策过程，在最优策略方面与具有线性时序逻辑规范 φ的原始马尔可夫决策过程等价。该定理的证明见附录 A.1。

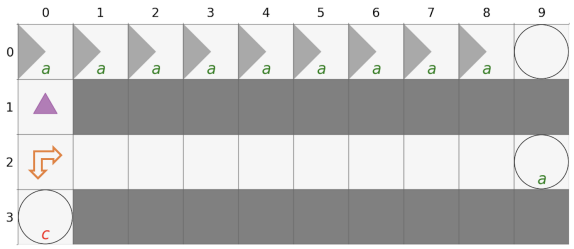
定理 1(可满足性等价)。对于由线性时序逻辑公式 φ诱导的任意乘积马尔可夫决策过程 $\mathcal{M}^\times$，我们有

$$\sup_{\pi} \mathbb{P}_\pi(s_0 \models \varphi) = \sup_{\pi^\times} \mathbb{P}_{\pi^\times}(s_0^\times \models \mathsf{GF}\ \varphi_{\mathcal{F}}). \tag{8}$$

此外，从初始状态出发，在乘积马尔可夫决策过程中使满足 *B¨uchi* 条件 $\varphi_{\mathcal{F}}$ 的概率最大化的确定性无记忆策略，会在原始马尔可夫决策过程 $\mathcal{M}$ 中诱导出一种确定性有限记忆最优策略，使从初始状态出发满足 φ 的概率最大化。



(a) “FG a & G !c”的极限确定性 Büchi 自动机。



(b) 用于说明 K 计数器的一个马尔可夫决策过程示例，其中 (1,0) 处的紫色三角形是起点，(2,0) 处的双向箭头是一个概率门。

图 1：一个用于说明 K 计数器的极限确定性 Büchi 自动机示例（左）和概率门控马尔可夫决策过程（右）（见示例 1）。

LTL 学习的奖励结构

我们首先为 LTL 学习定义一种广义奖励结构 $\mathcal{M}^\times$，然后证明，在该奖励结构下，获得最高期望折扣奖励与达到满足 $\varphi_\mathcal{F}$ 的最高概率是等价的。

定义 6 (奖励结构)。给定一个乘积马尔可夫决策过程 $\mathcal{M}^\times$和一个策略 π，乘积奖励函数 $r^\times$： $S^\times \times A^\times \times S^\times \to \mathbb{R}$ 适用于 *LTL* 学习，当且仅当

$$r^\times((s,q,n),a^\times,(s',q',n')) = \begin{cases} R_n & \text{if } q' \in \mathcal{F}; \\ 0 & \text{otherwise,} \end{cases} \quad (9)$$

其中， $R_n \in (0、 U]$ 是 $n \in [0..K]$ 的常数，而 $U \in (0,1]$ 是奖励的上界。奖励仅在接受自动机状态下非零，并取决于 K 计数器的值。

然后，给定折扣因子 $\gamma \in (0,1]$，，我们将乘积折扣函数 $\gamma^\times : S^\times \to (0,1]$ 定义为

$$\gamma^\times(s_j^\times) = \begin{cases} 1 - r_j^\times & \text{if } r_j^\times > 0; \\ \gamma & \text{otherwise,} \end{cases}$$

策略 π 从 $s^\times$ 开始，在时间步 t 之后的期望折扣奖励为

$$G_t^\pi(s^\times) = \mathbb{E}_\pi[\sum_{i=t}^{\infty}(\prod_{j=t}^{i-1}\gamma^\times(s_j^\times)) \cdot r^\times(s_i^\times, a_i^\times, s_{i+1}^\times) \mid s_t^\times = s^\times]. \quad (10)$$

路径中达到的最高 K 值（即路径中访问过的接受状态数量）可作为衡量该路径对满足 $\varphi_\mathcal{F}$有多大希望的指标。利用这一点，我们可以为接受状态分配不同的奖励来引导智能体，正如第 3.3 节的动机示例中所讨论的那样。接下来，我们给出一个引理，阐述乘积马尔可夫决策过程关于满足 B¨uchi 条件 $\varphi_\mathcal{F}$的性质。

引理 1. 给定一个乘积马尔可夫决策过程 $\mathcal{M}^\times$ 及其对应的线性时序逻辑公式 φ 和一个策略 π，我们用 $\mathcal{M}_\pi^\times$ 表示由 π诱导的马尔可夫链。令 $B_\mathcal{F}^\times$ 表示属于 $\mathcal{M}_\pi^\times$的接受底强连通分量的状态集合， $B_\varnothing^\times$ 表示属于拒绝底强连通分量的状态集合：

$$B_\mathcal{F}^\times := \{s^\times \mid s^\times \in B \in BSCC(\mathcal{M}_\pi^\times)\ B \cap \mathcal{F}^\times \neq \varnothing\}; \quad (11)$$

$$B_\varnothing^\times := \{s^\times \mid s^\times \in B \in BSCC(\mathcal{M}_\pi^\times), B \cap \mathcal{F}^\times = \varnothing\}, \quad (12)$$

其中， $BSCC(\mathcal{M}_\pi^\times)$ 是 $\mathcal{M}_\pi^\times$的所有底强连通分量构成的集合。我们进一步定义更一般的接受集合和拒绝集合：

$$B_\mathcal{F} := \{(s,q,n) \mid \exists n' \in [0..K] : (s,q,n') \in B_\mathcal{F}^\times\}; \quad (13)$$

$$B_\varnothing := \{(s,q,n) \mid \exists n' \in [0..K] : (s,q,n') \in B_\varnothing^\times\}. \quad (14)$$

于是我们有$\mathbb{P}_\pi(s^\times \models \mathsf{GF}\ \varphi_\mathcal{F}) = 1\ \forall s^\times \in B_\mathcal{F}$，$\mathbb{P}_\pi(s^\times \models \mathsf{GF}\ \varphi_\mathcal{F}) = 0\ \forall s^\times \in B_\varnothing$ 以及 $B_\varnothing \cap \mathcal{F}^\times = \varnothing$。此外， $B_\mathcal{F}$和 $B_\varnothing$ 是汇集集，这意味着一旦到达该集合，就无法到达集合外的任何状态。

该引理的证明见附录 A.2。利用该引理，我们现在可以陈述并证明本文的主要定理。

定理 2 (最优性保证). 给定一个线性时序逻辑公式φ 和一个乘积马尔可夫决策过程 $\mathcal{M}^\times$，存在一个上界$U \in (0,1]$，用于奖励；还存在一个折扣因子 $\gamma \in (0,1]$，使得对于所有满足定义 *6* 的乘积奖励 $r^\times$ 和乘积折扣函数 $\gamma^\times$，最大化期望折扣奖励 $G_0^{\pi_r}(s_0^\times)$ 的最优确定性无记忆策略 π_r，同时也是最大化在乘积马尔可夫决策过程 $\mathcal{M}^\times$上满足 *B*¨奇条件$\mathbb{P}_{\pi_\varphi}(s_0^\times \models \mathsf{GF}\ \varphi_\mathcal{F})$ 概率的最优策略 π_φ 。

证明概要。我们现在简要说明证明思路，以帮助理解主要步骤及关键参数的选择。完整证明见附录 A.3。

为确保最优性，给定策略 π 、乘积马尔可夫决策过程 $\mathcal{M}^\times$ 和线性时序逻辑公式 φ，我们希望证明遵循 π 所获得的期望折扣奖励与 π 满足 φ 的概率之间存在紧确界，从而最大化其中一个量等价于最大化另一个量。

概括而言，我们希望选取两个关键参数：奖励上界 $U \in$（$0,1]$ ）和折扣因子 $\gamma \in (0,1],$ ），以充分约束：(i) 对最终到达拒绝底强连通分量的路径所给予的奖励（因此这些路径不满足线性时序逻辑规范）；以及 (ii) 对于最终到达接受底强连通分量的路径，来自拒绝状态的奖励所受到的折扣。

我们非正式地将 $C_\varnothing^\pi$ 记为到达拒绝底层强连通分量之前访问接受状态的期望次数，并且通过选择 $U = 1/C_\varnothing^\pi$ 可以充分地界定（i）。接下来，我们非正式地将 $C_\mathcal{F}^\pi$ 记为到达接受底层强连通分量之前访问拒绝状态的期望次数，并将

算法 1： 基于线性时序逻辑的 KC Q-learning

输入： 环境马尔可夫决策过程 $\mathcal{M}$，线性时序逻辑公式 φ

将 φ 转换为 LBDA $\mathcal{A}$
使用 $\mathcal{M}$ 和 $\mathcal{A}$构建乘积马尔可夫决策过程 $\mathcal{M}^{\times}$
为每个 s 和 q 配对初始化 Q
对于 $l \leftarrow 0$ 至最大回合数 **执行**
  初始化 $(s,\ q\ n) \leftarrow (s_0,\ q_0, 0)$
  对于 $t \leftarrow 0$ 到最大时间步**执行**
    获取由 Q 导出的策略 π （例如， ϵ-贪心）
    在 $\mathcal{M}^{\times}$ 中执行动作 $a^{\times} \leftarrow \pi((s,\ q\ n))$
    获取下一个乘积状态 $(s',\ q',\ n')$
    $r \leftarrow r^{\times}((s,\ q\ n)\ a^{\times},\ (s',\ q',\ n'))$
    $\gamma \leftarrow \gamma'(s,\ q\ n)$
    $Q((s,\ q)\ a^{\times}) \overset{\alpha}{\leftarrow} r + \gamma$最大值$_{a \in A^{\times}}\ Q((s',\ q')\ a)$
    更新 $(s,\ q\ n) \leftarrow (s',\ q',\ n')$
获取贪心策略 π_{φ} 从 Q
返回 $\mathcal{M}$ 上的诱导策略，通过移除 ϵ-动作

到达接受底层强连通分量后，连续访问接受状态之间的期望步数记为 N^{π}。直观而言，对于（ii），我们使用 $C^{\pi}_{\mathcal{F}}$ 界定到达接受底层强连通分量之前的折扣量，并使用 N^{π} 界定到达该底层强连通分量之后的折扣量，从而得到 $\gamma = 1 - 1/(C^{\pi}_{\varnothing} * N^{\pi} + C^{\pi}_{\mathcal{F}})$。

在实践中，改用 $C^{\pi}_{\varnothing}$，$C^{\pi}_{\mathcal{F}}$和 N^{π}的上界同样可以确保最优性，而这些上界可以根据对马尔可夫决策过程的假设或了解推导出来。□

如证明概要所示，选择 $U = 1/C^{\pi}_{\varnothing}$ 和$\gamma = 1 - 1/(C^{\pi}_{\varnothing} * N^{\pi} + C^{\pi}_{\mathcal{F}})$ 足以确保最优性。以图 1b 中的概率门控马尔可夫决策过程为例，我们有 $C^{\pi}_{\varnothing} \approx C^{\pi}_{\mathcal{F}} \leq 10$ 和 $N^{\pi} = 1$，因此选择 $U = 0.1$ 和 $\gamma = 0.95$足以保证最优性。对于更一般的马尔可夫决策过程，在状态数 $|S|$ 和最小非零转移概率 $p_{\min} := \min_{s,\ a,\ s'}\{T(s,\ a,\ s') > 0\}$ 已知这一常见假设下，$C^{\pi}_{\varnothing}$ 和 $C^{\pi}_{\mathcal{F}}$可以由 $|S|/p_{\min}$给出上界，而N^{π} 可以由 $|S|$给出上界。

3.4 使用 Q学习进行线性时序逻辑学习

利用该乘积马尔可夫决策过程 $\mathcal{M}^{\times}$ 及其奖励结构，我们提出算法 1 (KC)，这是一种针对线性时序逻辑规范、利用 K 计数器乘积马尔可夫决策过程的无模型 Q学习算法。随着探索的进行，乘积马尔可夫决策过程会即时构建：对于动作 $a^{\times} \in A$，在环境状态 s下于环境中执行动作 $a^{\times}$，观察下一个环境状态 s'。然后，根据 $n \leq K$ 和自动机状态 $q' \in \mathcal{F}$，利用转移函数 $q\ L(s)$ 和计数器 n，计算下一个自动机状态 q'。如果 $a^{\times} \in \{\epsilon_q \mid q \in \mathcal{Q}\}$，则使用$\epsilon$-转移更新 q'，并保持环境状态 s 和计数器 n 不变。然而，直接在该乘积马尔可夫决策过程中采用 Q学习，会得到一个定义在整个乘积状态空间 $S^{\times}$上的 Q 函数，这意味着智能体需要为每个 K 值学习 Q 函数。为提高效率，我们提出仅在环境状态 S 和自动机状态 $\mathcal{Q}$ 上定义 Q 函数；对于 $\mathcal{M}^{\times}$的路径 σ，其更新

算法 2： 从线性时序逻辑生成 CF+KC Q-learning

输入： 环境马尔可夫决策过程 $\mathcal{M}$、线性时序逻辑公式 φ

将 φ 转换为 LBDA $\mathcal{A}$
使用 $\mathcal{M}$ 和 $\mathcal{A}$构造乘积马尔可夫决策过程 $\mathcal{M}^{\times}$
为每个 s 和q 对初始化 Q
对 $l \leftarrow 0$ 至最大事件处理数**执行**
  初始化 $(s,q\ n) \leftarrow (s_0, q_0, 0)$
  对 $t \leftarrow 0$ 至最大时间步**执行**
    获取由 Q 推导出的策略 π （例如， ϵ-贪心）
    在 $\mathcal{M}^{\times}$ 中获取动作 $a^{\times} \leftarrow \pi((s,q\ n)$
    对 $^{-}q \in \mathcal{Q}$ **执行**
      获取反事实想象 $(s', {}^{-}q', n')$
       通过在 $(a^{\times}, {}^{-}q\ n)$ 处采取动作
      $r \leftarrow r^{\times}((s, {}^{-}q\ n)\ a^{\times}, (s', {}^{-}q', , n'))$
      $\gamma \leftarrow \gamma'(s,\ {}^{-}q\ n)$
      $Q((s,\ {}^{-}q)\ a^{\times}) \overset{\alpha}{\leftarrow}$
       $r + \gamma$最大化$_{a \in A^{\times}}\ Q((s',\ {}^{-}q')\ a)$
    通过动作 $a^{\times}$ 在 $(s,\ q\ n)$ 处获得 $(s',\ q',\ n')$
    更新 $(s,\ q\ n) \leftarrow (s',\ q',\ n')$
从 Q 中获得贪心策略π_{φ}
返回通过移除 ϵ-动作得到的 $\mathcal{M}$ 上的诱导策略

在时间步 t 的 Q 函数更新规则为：

$$Q_{t+1}((s_t, q_t), a^{\times}_t) \overset{\alpha}{\leftarrow} r^{\times}(s^{\times}_t, a^{\times}_t, s^{\times}_{t+1}) + \gamma'(s^{\times}_t) \max_{a^{\times} \in A^{\times}} Q_t((s_{t+1}, q_{t+1}), a^{\times}), \tag{15}$$

其中，$s^{\times}_t = (s_t,\ q_t,\ n_t)$ 是在时间步 t 执行的动作，$s^{\times}_{t+1} = (s_{t+1},\ q_{t+1},\ n_{t+1})$ 是下一个乘积状态，α 是学习率。我们声称，使用这个折叠后的 Q 函数，该算法能够返回满足 φ 的最优策略，因为最优策略与 K 计数器无关，证明见附录 A.4。

定理 3。 给定环境马尔可夫决策过程 $\mathcal{M}$ 和线性时序逻辑规范 φ，以及适当的折扣因子 γ 和满足定理 *2* 的奖励函数 $r^{\times}$，算法 *1* 中描述的用于线性时序逻辑的 *Q*学习收敛到最优策略 π_{φ}，该策略在 $\mathcal{M}$ 上最大化满足 φ 的概率。

3.5 反事实想象

此外，我们提出一种利用乘积马尔可夫决策过程结构（特别是极限确定性 Büchi 自动机）来促进学习的方法。我们采用反事实推理生成合成想象：从环境马尔可夫决策过程中的一个状态出发，想象自己处于每个自动机状态，同时执行相同的动作。

如果智能体处于乘积状态 $(s,\ q\ n)$，并选择一个动作 $a^{\times} \in A$，对于每个 $^{-}q \in \mathcal{Q}$，通过在 $(s,\ {}^{-}q\ n)$ 执行动作 $a^{\times}$，可以计算出下一个状态 $(s',\ {}^{-}q',\ n')$：首先在环境状态中执行该动作，然后计算下一个自动机状态 $^{-}q' = \Delta({}^{-}q\ L(s))$ 和下一个 K值 n'。智能体获得的奖励为 $r^{\times}(s,\ {}^{-}q\ \bar{n})$，因此我们可以利用这组扩充后的经验更新 Q 函数。这些由反事实想象生成的经验

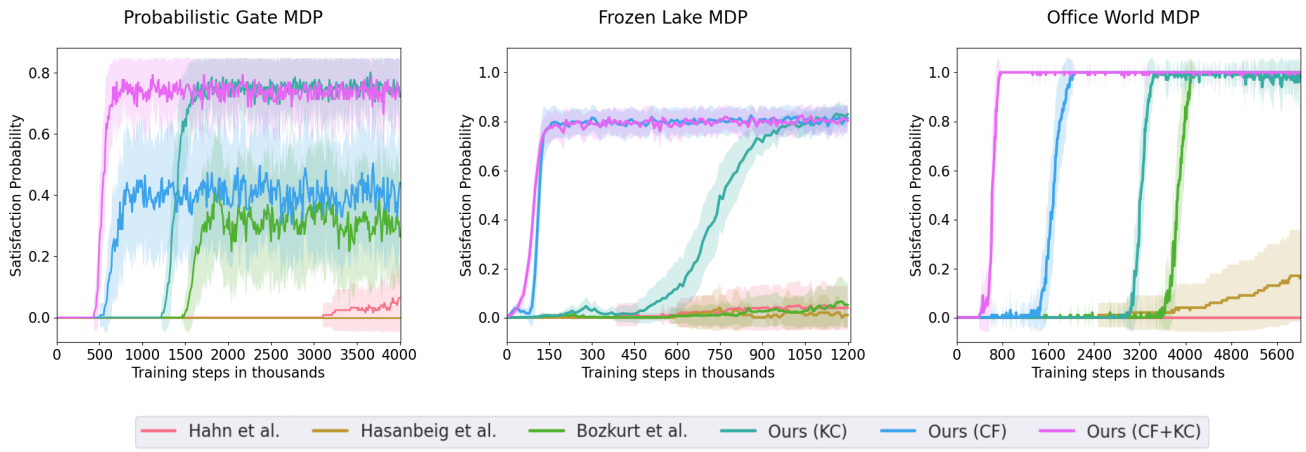


图 2：不同马尔可夫决策过程和任务上的实验结果。

仍然是从 $\mathcal{M}^{\times}$ 依据转移函数 $T^{\times}$采样得到的，因此，与Q学习等任何离策略学习算法结合使用时，算法的最优性保证仍然成立。

如算法2（CF-KC）所示，只需修改几行代码（算法 2 中的第 9–12 行），即可将反事实想象（CF）融入我们的 KC Q-learning 算法；它也可与其他用于线性时序逻辑的自动机乘积马尔可夫决策过程强化学习算法结合使用。请注意，反事实想象的思想与 Toro Icarte 等人 [2022], 提出的方法类似，但我们的方法采用极限确定性 Büchi 自动机来学习线性时序逻辑规范的乘积马尔可夫决策过程。

4 实验结果

我们在多种 MDP 环境上评估了所提出的算法，其中包括更贴近实际且更具挑战性的随机马尔可夫决策过程环境[1]。我们提出了一种方法，通过使用概率模型检测器 PRISM [Kwiatkowska 等人,2011]直接评估满足线性时序逻辑规范的概率。我们根据环境马尔可夫决策过程和策略构建诱导马尔可夫链 $\mathcal{M}_{\pi}$ 的 PRISM格式模型，并使用 PRISM 计算给定线性时序逻辑规范的精确满足概率。我们采用表格型Q学习作为核心离策略学习方法，实现了三种算法：使用 K 计数器奖励结构的 Q学习（KC）、使用 K 计数器奖励结构和反事实想象的Q学习（CF+KC），以及仅使用反事实想象的 Q学习（CF）；其中我们设置 $K = 0$。我们将所提方法的性能与 Bozkurt 等人 [2019], 、Hahn 等人 [2019] 和 Hasanbeig 等人 [2020]提出的方法进行比较。请注意，在 $K = 0$且不使用反事实想象这一特殊情形下，我们的 KC 算法在算法上等价于 Bozkurt 等人 [2019]的方法，只需将其参数设为 $\gamma_B = 1 - U$。据我们所知，我们的方法与所有其他现有方法均不同。实验的详细信息和设置见附录B。

我们将学习率设为 $\alpha = 0.1$，并将 $\epsilon = 0.1$ 用于探索。我们还为奖励设置了一个相对宽松的上界

[1]我们的算法和实验的实现可在 GitHub 上找到: https://github.com/shaodaqian/rl-from-ltl

所有实验均采用$U = 0.1$ 和折扣因子 $\gamma = 0.99$，以确保最优性。需要注意的是，我们的算法对于定义 6 中定义的一类奖励结构均具有最优性；而在实验中，为了便于训练和探索，我们选用了一种特定的奖励函数，即随着 K 的值增加，接受状态的奖励线性增加，具体为 $r_n = U \cdot n/K \; \forall n \in [0..K]$。通过将所有可用状态-动作对的 Q 值设为 $2U$，对 Q 函数进行乐观初始化。所有实验均运行 100 次，并绘制平均满足概率，阴影区域表示半个标准差。

首先，我们在示例 1 中描述的概率门控马尔可夫决策过程上进行实验，任务为“FG a & G !c”，其含义是未来只能到达标记为“a”的状态，同时永远不能到达标记为“c”的状态。我们将该任务设为 $K = 10$。在图 2（左）中，与其他三种方法相比，我们的方法 KC 具有更高的样本效率和更快的收敛速度；CF 则展现出更高的样本效率，但训练稳定性仍不足。CF+KC 的性能最佳，而其他方法要么收敛较慢（Bozkurt 等 [2019] 和 Hahn等 [2019]），要么由于缺乏理论最优性保证而无法收敛（Hasanbeig 等 [2020]）。

第二个MDP环境是 OpenAI Gym 中的 8×8 冰湖环境 [Brockman 等， 2016]。该环境由冰湖地块组成，智能体有 1/3 的概率朝预定方向移动，向两侧移动的概率则各为 1/3，详情见附录B.2。任务是“(GF a | GF b) & G !h”，这意味着总是到达湖泊营地“a”或湖泊营地“b”，同时永不掉入洞“h”。我们为该任务设置 $K = 10$，在图 2（中）中可以观察到，我们所有方法的样本效率都显著更高，尤其是 CF+KC 和 CF，它们在大约 150k 训练步数时收敛到最优策略。另一方面，其他三种方法在 1200k 训练步数时才刚开始收敛。CF 在这项任务中表现尤为出色，因为每个时间步都可以同时考虑总是到达“a”或“b”的选择，从而降低探索环境的样本复杂度。

最后，我们在以下内容的轻微修改版本上进行了实验：

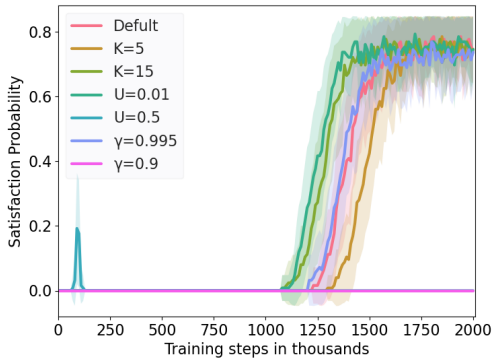


图 3：关键参数的敏感性分析。

Toro Icarte 等人 [2022], 提出的更具挑战性的办公室世界环境，详情见附录 B.3。我们在办公室世界中加入了冰面，任务是巡逻“a”和“b”之间的走廊，或先在“l”处写信，再在取茶处“t”和工作场所“w”之间巡逻，同时避开障碍物“o”。针对这一任务，$K = 5$被设定为逐步增加奖励，以实现更陡峭的增长，因为巡逻状态之间距离较远，每个事件处理中访问多个接受状态会耗费较多时间。图 2（右）再次展示了我们方法的性能优势，其中 CF+KC表现最佳，CF和KC分别位居第二和第三。对于这个大型环境中的挑战性任务，Hahn等人 [2019] 的方法需要最多的训练步数才能收敛。

总体而言，结果表明，KC 提升了样本效率和训练稳定性，尤其是在具有挑战性的任务中。此外，CF 大幅提高了样本效率，与 KC 结合后取得了最佳结果。

4.1 运行时间分析和敏感性分析

值得一提的是，在进行反事实想象时，环境中的每一步都会对 Q 函数进行多次更新。这会增加计算复杂度，但如果环境步的计算开销较大，针对 Q 函数额外执行的内循环更新只会对总计算时间产生轻微影响。以办公室世界任务为例，完成 600 万训练步所需的平均时间分别为 KC 的 170.9 秒、CF 的 206.3 秒和 CF+KC 的 236.1 秒。然而，收敛到最优策略所需的时间分别为 KC 的 96.8 秒、CF 的 70.2 秒和 CF+KC 的 27.5 秒。

为对关键参数进行敏感性分析，我们在概率门MDP任务上运行实验，将不同参数与默认值 $U = 0.1$，$\gamma = 0.99$ 和 $K = 10$进行比较。如图 3 所示，如果 $U(= 0.5)$ 设得过高，或 $\gamma(= 0.9)$ 设得过低，算法就无法如预期收敛到最优策略。然而，较宽松的参数 $U = 0.01$ 和 $\gamma = 0.995$ 不会损害性能，这意味着，即使对底层 MDP 的了解有限，我们的算法在参数设置较宽松时仍能表现良好。最优性不受 K 值的影响，而性能仅会受到不同 K值的轻微影响。

5 结论

我们提出了一种新颖的无模型强化学习算法，用于在未知的随机 MDP 中学习满足线性时序逻辑规范的最优策略，并提供最优性保证。我们提出了一种新颖的乘积马尔可夫决策过程、一种广义奖励结构，以及一种在参数设置适当时能够确保收敛到最优策略的强化学习算法。此外，我们引入了反事实想象，利用线性时序逻辑规范生成想象体验。最后，我们借助 PRISM [Kwiatkowska 等人， 2011],直接评估了所提方法的性能，并展示了这些方法在各种 MDP 环境和线性时序逻辑任务上的优越表现。

未来的工作包括：在我们的广义奖励结构框架下探索其他特定的奖励函数；解决开放问题 [Alur 等提出的 2021] 移除所有关于底层 MDP 的假设；以及将理论框架扩展到具有连续状态和动作的环境中的 MDP，这或许可以通过对状态空间进行抽象来实现。此外，利用基于潜在函数的奖励塑形 [Ng 等， 1999][Devlin 和 Kudenko，2012] 发掘线性时序逻辑规范的语义类别结构，并在不同环境之间迁移智能体的类似时序逻辑知识 [Xu 和 Topcu， 2019] 也值得探究。

致谢

本研究获得 EPSRC Prosperity Partnership FAIR 项目资助（资助编号 EP/V056883/1）。DS 感谢图灵研究所与埃森哲合作项目提供的资助。MK 获得欧洲研究理事会通过欧盟地平线2020研究与创新计划提供的资助（ FUN2MODEL，资助协议编号 834115）。

参考文献

[Aksaray等， 2016] Derya Aksaray、Austin Jones、Zhao- dan Kong、Mac Schwager 和 Calin Belta。用于稳健满足信号时序逻辑规范的 Q学习。*2016*年 *IEEE* 第*55*届决策与控制会议，*CDC 2016*，第 6565–6570 页，2016年 9 月。

[Alur等， 2021] Rajeev Alur、Suguman Bansal、Osbert Bastani 和 Kishor Jothimurugan。强化学习中规范转换框架。*arXiv:2111.00272*，2021年。

[Andreas等人， 2016] Jacob Andreas、Dan Klein 和 Sergey Levine。基于策略草图的模块化多任务强化学习。第*34*届国际机器学习会议，*ICML 2017*，1:229–239，2016年11月。[Baier 和 Katoen， 2008] Christel Baier 和 Joost-Pieter Katoen。模型检验原理，第 950 卷。麻省理工学院出版社，2008年。[Bozkurt等人， 2019] Alper Kamil Bozkurt、Yu Wang、Michael M. Zavlanos 和 MiroslavPajic。利用模型无关强化学习，从线性时序逻辑规范中进行控制合成。*IEEE*机器人与自动化国际会议论文集，第10349–10355页，2019年。

[Brockman等人， 2016] Greg Brockman、Vicki Cheung、Ludwig Pettersson、Jonas Schneider、John Schulman、Jie Tang 和 Wojciech Zaremba，OpenAI。OpenAI Gym。*arXiv:1606.01540*，2016年。

[Cai等， 2021] Mingyu Cai、Mohammadhosein Hasan-beig、Shaoping Xiao、Alessandro Abate 和 Zhen Kan。基于时序逻辑的连续运动规划模块化深度强化学习。*IEEE*机器人与自动化快报，6(4):7973–7980，2021年。

[Camacho等， 2019] Alberto Camacho、Rodrigo Toro Icarte、Toryn Q. Klassen、Richard Valenzano 和 Sheila A. McIlraith。LTL及其扩展：强化学习中奖励函数规范的形式语言。*IJCAI*国际人工智能联合会议，2019年-8月：6065–6073，2019年。

[De Giacomo 和 Vardi， 2013] Giuseppe De Giacomo 和 Moshe Y Vardi。有限轨迹上的线性时序逻辑和线性动态逻辑。*AAAI*人工智能会议论文集，2013年。

[Devlin 和 Kudenko， 2012] Sam Devlin 和 Daniel Kudenko。《基于动态势函数的奖励塑造》。第 *11* 届自主智能体与多智能体系统国际会议论文集，2012年。

[Giuseppe De Giacomo等人， 2019] Giuseppe De Giacomo、Luca Iocchi、Marco Favorito 和 Fabio Patrizi。《约束规范的基础：采用 LTLf/LDLf 约束规范的强化学习》。载于第二十九届自动规划与调度国际会议论文集，2019年。

[Hahn等人， 2019] Ernst Moritz Hahn、Mateo Perez、Sven Schewe、Fabio Somenzi、Ashutosh Trivedi 和 Dominik Wojtczak。《无模型强化学习中的 ω 正则目标》。系统构建与分析工具和算法国际会议论文集，11427 LNCS:395–412，2019年。

[Hahn等， 2020] Ernst Moritz Hahn、Mateo Perez、Sven Schewe、Fabio Somenzi、Ashutosh Trivedi 和 Dominik Wojtczak。面向 ω-正规目标的无模型强化学习的忠实且有效的奖励方案。自动化验证与分析技术，12302 LNCS：108–124，2020年。

[Hasanbeig等， 2019] M. Hasanbeig、Y. Kantaros、Alessandro Abate、D. Kroening、G. J. Pappas 和 I. Lee。具有概率满足保证的时序逻辑控制综合强化学习。*IEEE*决策与控制会议论文集，2019年12月：5338–5343，92019。

[Hasanbeig等， 2020] Mohammadhosein Hasanbeig、D. Kroening 和 Alessandro Abate。结合时序逻辑的深度强化学习。国际定时系统形式化建模与分析会议论文集，12288LNCS：1–22，2020年。

[Jiang等， 2021] Yuqian Jiang、Suda Bharadwaj、Bo Wu、Rishi Shah、Ufuk Topcu 和 Peter Stone。基于时序逻辑的奖励塑造，用于持续型强化学习任务。*AAAI* 人工智能会议论文集，35(9):7995–8003，2021年 5 月。[Jothimurugan等， 2020] Kishor Jothimurugan、Rajeev Alur 和 Osbert Bastani。用于强化学习任务的可组合规范语言。神经信息处理系统进展，32，2020 年 8 月。[Jothimurugan等， 2021] Kishor Jothimurugan、Suguman Bansal、Osbert Bastani 和 Rajeev Alur。基于逻辑规范的组合式强化学习。神经信息处理系统进展，12:10026–10039，2021年 6 月。

[K˘ret´ınsk´y 等, 2018] Jan K˘ret´ınsk´y, Tobias Meggendorfer, Salomon Sickert 和 Christopher Ziegler。Rabinizer 4：从线性时序逻辑到您偏好的确定性自动机。国际计算机辅助验证会议论文集，10981 LNCS:567–577，2018年。

[Kwiatkowska等， 2011] Marta Kwiatkowska、Gethin Norman 和 David Parker。PRISM 4.0：概率实时系统的验证。计算机辅助验证，6806 LNCS:585–591，2011。

[Li 等, 2016] Xiao Li、Cristian Ioan Vasile 和 Calin Belta。带有时序逻辑奖励的强化学习。*IEEE*智能机器人与系统国际会议，2017-September:3834–3839，12 2016年。

[Littman等，2017] Michael L. Littman、Ufuk Topcu、Jie Fu、Charles Isbell、Min Wen 和 James MacGlashan。基于广义线性时序逻辑的环境无关任务规范。*arXiv:1704.04341*，2017年4月。

[Littman, 2001] M.L. Littman。马尔可夫决策过程。《国际社会与行为科学百科全书》，第9240–9242页，2001年。

[Majeed 和 Hutter, 2018] Sultan Javed Majeed 和 Marcus Hutter。关于非马尔可夫决策过程中的Q学习收敛。*IJCAI*国际人工智能联合会议，2018年7月：2546–2552，2018年。

[Ng 等, 1999] Andrew Y. Ng、Andrew Y. Ng、Daishi Harada 和 Stuart Russell。奖励变换下的策略不变性：理论及其在奖励塑形中的应用。第十六届国际机器学习会议论文集，第278–287页，1999年。

[Sadigh 等人, 2014] Dorsa Sadigh、Eric S. Kim、Samuel Coogan、S. Shankar Sastry 和 Sanjit A. Seshia。基于学习的方法：针对线性时序逻辑规范的马尔可夫决策过程控制综合。*IEEE*决策与控制会议论文集，2015年2月（2月）：1091–1096，2014年。

[Sickert等人, 2016] Salomon Sickert、Javier Esparza、Stefan Jaax 和 Jan K˘ret´ınsk´y。用于线性时序逻辑的极限确定性 B¨uchi 自动机。计算机辅助验证，9780：312–332，2016年。

[Sutton and Barto， 2018] Richard S Sutton 和 Andrew G Barto。强化学习：导论，第3卷。麻省理工学院出版社，2018年。

[Toro Icarte等， 2018] Rodrigo Toro Icarte、Toryn Q Klassen、Richard Valenzano 和 Sheila A Mcllraith。运用奖励机进行强化学习中的高层任务规范与分解。第*35*届国际机器学习会议论文集，2018年。[Toro Icarte等， 2022] Rodrigo Toro Icarte、Toryn Q. Klassen、Richard Valenzano 和 Sheila A. McIlraith。奖励机：利用强化学习中的奖励函数结构。《人工智能研究杂志》，73:173–208，2022年。[Watkins 和 Dayan， 1992] Christopher J. C. H. Watkins 和 Peter Dayan。Q学习。《机器学习》*1992*年*8(3)*，8(3):279–292，1992年5月。[Xu 和 Topcu， 2019] Zhe Xu 和 Ufuk Topcu。强化学习中的时序逻辑公式迁移。第二十八届国际人工智能联合会议论文集，2019年。

A 理论结果：证明

A.1 定理 1 的证明

证明。我们将证明等式　　通过验证不等式两侧并随后构造诱导策略来实现最优性　　cy 策 $\mathcal{M}$。

对于 $\leq$，Sickert等人已证明 [，对于环境马尔可夫决策过程 $\mathcal{M}$ 中遵循某一策略 π的每条接受路径，总存在一条对应的极限确定性 Büchi 自动机 θ 接受运行来解决非确定性。将已解决的非确定性作为 ϵ-动作附加到原路径上，会得到 $\mathcal{M}^{\times}$的一条接受路径，因为计数器最终会达到 K，且接受状态 $\mathcal{F}^{\times}$ 会被无限多次访问。因此，我们为 $\mathcal{M}^{\times}$ 构造了一个策略，其在 $\mathcal{M}$上的表现至少与 π 一样好。

对于 $\geq$，显然， π 在 $\mathcal{M}^{\times}$ 上的任何策略都可以通过消除 ϵ-转移并移除自动机投影和 K 计数器，为 $\mathcal{M}$ 导出一个策略。因此，遵循 π 且满足 B¨uchi 条件 $\varphi_{\mathcal{F}}$ 的任何路径，都会导出一条 $\mathcal{M}$ 的接受路径；该路径由 φ导出的 $\mathcal{A}$ 给出，其中 $\mathcal{A}$ 的非确定性由 π的 ϵ-转移解决，从而满足 φ。

□

A.2 引理1的证明

证明。 首先，我们回顾马尔可夫链（MC）中底强连通分量的性质：对于 MC 的任意无限路径 σ，最终都会到达某个底强连通分量；一旦到达，就无法到达该底强连通分量之外的任何状态，并且以概率 1 无限多次访问其中的所有状态。因此，由于所有 $s^{\times} \in B^{\times}_{\mathcal{F}}$ 都属于包含接受状态的底强连通分量，从 $s^{\times}$出发的所有路径都会以概率 1 无限多次到达接受状态，因此 $\mathrm{P}_{\pi}(s^{\times} \models \mathsf{GF}\ \varphi_{\mathcal{F}}) = 1\ \forall s^{\times} \in B^{\times}_{\mathcal{F}}$。类似地，所有 $s^{\times} \in B^{\times}_{\varnothing}$都属于不包含接受状态的底强连通分量，因此 $\mathrm{P}_{\pi}(s^{\times} \models \mathsf{GF}\ \varphi_{\mathcal{F}}) = 0\ \forall s^{\times} \in B^{\times}_{\varnothing}$。

现在，我们注意到，对于 $\mathcal{M}^{\times}_{\pi}$中的状态，环境状态和自动机状态的马尔可夫链转移实际上与计数器值无关：

$$T^{\times}_{\pi}((s,q,n_1),(s',q',\min(n_1+1,K))) = T^{\times}_{\pi}((s,q,n_2),(s',q',\min(n_2+1,K))) \tag{16}$$

对所有 $n_1, n_2 \in [0..K], s, s' \in S$ 以及 $\mathsf{q}, \mathsf{q}' \in \mathcal{Q}$。基于这一观察，对于所有 $(s, \mathsf{q}\ n) \in B_{\mathcal{F}}$，都存在 $(s, \mathsf{q}\ n') \in B^{\times}_{\mathcal{F}}$，根据第 13 行，它属于某个底强连通分量。因此，由公式 16 可达的任意状态 $(s_1, \mathsf{q1}, n_1)$，都意味着存在 n_2，使得 $(s_1, \mathsf{q1}, n_2)$ 可从 $(s, \mathsf{q}\ n')$ 到达。根据底强连通分量的定义，这意味着 $(s_1, \mathsf{q1}, n_2) \in B^{\times}_{\mathcal{F}}$，而根据第 13 行，这又意味着 $(s_1, \mathsf{q1}, n_1) \in B_{\mathcal{F}}$，由此可见 $B_{\mathcal{F}}$ 是一个汇集集。此外，由于 $(s, \mathsf{q}\ n') \in B^{\times}_{\mathcal{F}}$ 属于某个接受底强连通分量，它以概率 1 能到达该分量中的一个接受状态 $(s_{\mathcal{F}}, \mathsf{q}_{\mathcal{F}}, n_{\mathcal{F}}) \in B^{\times}_{\mathcal{F}}$。根据公式 16，存在一个 $(s_{\mathcal{F}}, \mathsf{q}_{\mathcal{F}}, n_3) \in B_{\mathcal{F}}$，它以概率 1 可从 $(s, \mathsf{q}\ n)$ 到达；并且，根据乘积马尔可夫决策过程的定义 5 中接受状态的定义，$(s_{\mathcal{F}}, \mathsf{q}_{\mathcal{F}}, n_3)$ 也属于接受状态，这意味着从 $B_{\mathcal{F}}$ 中的任意状态出发，都能以概率 1 到达接受状态。结合 $B_{\mathcal{F}}$ 是一个汇集集这一事实，我们得出结论：从 $B_{\mathcal{F}}$ 出发，以概率 1 能无限次到达接受状态，即 $\mathrm{P}_{\pi}(s^{\times} \models \mathsf{GF}\ \varphi_{\mathcal{F}}) = 1\ \forall s^{\times} \in B_{\mathcal{F}}$。另一个观察是，由于计数器值沿所有路径递增，且在接受底强连通分量中会无限次到达接受状态，因此接受底强连通分量中的所有状态的计数器值都必须等于 K。

根据相同的论证，我们得出 $B_{\varnothing}$ 也是一个汇集集。此外，对于 $(s, q\ n) \in B_{\varnothing}$，存在属于某个拒绝底层强连通分量的 $(s, q\ n') \in B^{\times}_{\varnothing}$，从该分量无法到达任何接受状态。根据方程 16 和乘积马尔可夫决策过程的接受状态定义（定义 5），我们可以看到，从 $(s, q\ n)$ 也无法到达任何接受状态，这意味着 $\mathrm{P}_{\pi}(s^{\times} \models \mathsf{GF}\ \varphi_{\mathcal{F}}) = 0\ \forall s^{\times} \in B_{\varnothing}$ 和 $B_{\varnothing} \cap \mathcal{F}^{\times} = \varnothing$，由此完成证明。

□

A.3 定理 2 的证明

证明。 为概述证明，我们希望说明，对于任意确定性无记忆策略 π，所有计数器值为 0 的乘积状态对应的期望折扣奖励 $G^{\pi}_0(s^{\times})$，即 $s^{\times} = (s,\ q,\ 0)$，与从 $s^{\times}$ 开始并遵循 π 时满足 $\varphi_{\mathcal{F}}$ 的概率 $\mathrm{P}_{\pi}(s^{\times} \models \mathsf{GF}\ \varphi_{\mathcal{F}})$ 接近。我们通过分别给出这两个量之差的上界和下界来证明这一点。

根据 $s^{\times} = (s,\ q,\ 0)$，我们考虑以遵循策略π 的无限路径是否满足 B¨üchi 条件 $\varphi_{\mathcal{F}}$为条件的期望折扣奖励。

$$G^{\pi}_0(s^{\times}) = G^{\pi}_0(s^{\times} \mid s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}})\mathbb{P}_{\pi}(s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}}) \tag{17}$$

$$+ G^{\pi}_0(s^{\times} \mid s^{\times} \not\models \mathsf{GF}\varphi_{\mathcal{F}})\mathbb{P}_{\pi}(s^{\times} \not\models \mathsf{GF}\varphi_{\mathcal{F}}) \tag{18}$$

我们首先考虑第 17 行的各个组成部分。令从 $s^{\times}$ 首次到达接受集 $B_{\mathcal{F}}$ 的停止时刻为 $\tau^{\pi}_{\mathcal{F}} := \inf\{t > 0 \mid s^{\times}_t \in B_{\mathcal{F}}\}$，并令在 $B_{\mathcal{F}}$ 中首次到达的状态为 $(s, q\ n)$。接着，令包含 $(s, q\ n)$ 的接受底层强连通分量 B 中接受状态之间的命中时刻为 τ^{π}_{BSCC}。 此外，令 $c^{\pi}_{\mathcal{F}acc} := \left|t \leq \tau^{\pi}_{\mathcal{F}} \colon s^{\times}_t \in \mathcal{F}^{\times}\right|$，并

$c^{\pi}_{\mathcal{F}rej} := \left|t \le \tau^{\pi}_{\mathcal{F}} : s^{\times}_t \notin \mathcal{F}^{\times}\right|$ 为从 $s^{\times}$ 到达 $B_{\mathcal{F}}$ 之前到达的接受状态和非接受状态的数量，分别对应两者，其中 $c^{\pi}_{\mathcal{F}acc} + c^{\pi}_{\mathcal{F}rej} = \tau^{\pi}_{\mathcal{F}}$。注意，上述所有量都取决于起始状态 $s^{\times}$，但为方便起见，我们在符号中省略了该状态。于是，我们有

$$G^{\pi}_0(s^{\times} \mid s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}}) \tag{19}$$

$$= \mathbb{E}_{\pi}\left[\sum_{i=0}^{\infty}(\prod_{j=0}^{i-1}\gamma(s^{\times}_j)) \cdot r^{\times}_i \;\middle|\; s^{\times}_0 = s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}}|\right] \tag{20}$$

$$\ge \mathbb{E}_{\pi}\left[\gamma^{c^{\pi}_{\mathcal{F}rej}}\left(\sum_{n=0}^{c^{\pi}_{\mathcal{F}acc}}(\prod_{j=0}^{n-1}(1-R_j))R_n\right) + \sum_{i=\tau^{\pi}_{\mathcal{F}}}^{\infty}(\prod_{j=0}^{i-1}\gamma(s^{\times}_j)) \cdot r^{\times}_i \;\middle|\; s^{\times}_0 = s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}})\right] \tag{21}$$

$$\ge \mathbb{E}_{\pi}\left[\gamma^{c^{\pi}_{\mathcal{F}rej}}\left(\sum_{n=0}^{c^{\pi}_{\mathcal{F}acc}}(\prod_{j=0}^{n-1}(1-R_j))R_n + \sum_{i=\tau^{\pi}_{\mathcal{F}}}^{\infty}(\prod_{j=0}^{c^{\pi}_{\mathcal{F}acc}}(1-R_j)\prod_{k=\tau^{\pi}_{\mathcal{F}}}^{i-1}\gamma(s^{\times}_k)) \cdot r^{\times}_i\right) \;\middle|\; s^{\times}_0 = s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}})\right] \tag{22}$$

$$\ge \mathbb{E}_{\pi}\left[\gamma^{c^{\pi}_{\mathcal{F}rej}}\left(\sum_{n=0}^{c^{\pi}_{\mathcal{F}acc}}(\prod_{j=0}^{n-1}(1-R_j))R_n + \gamma^{\tau^{\pi}_{BSCC}/U}\sum_{n=c^{\pi}_{\mathcal{F}acc}}^{\infty}(\prod_{j=0}^{n-1}(1-R_j))R_n\right) \;\middle|\; s^{\times}_0 = s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}})\right] \tag{23}$$

$$\ge \gamma^{C^{\pi}_{\mathcal{F}}+N^{\pi}/U}\left(\sum_{n=0}^{\infty}(\prod_{j=0}^{n-1}(1-R_j))R_n\right) \tag{24}$$

$$= \gamma^{C^{\pi}_{\mathcal{F}}+N^{\pi}/U} \cdot 1 \tag{25}$$

其中 $C^{\pi}_{\mathcal{F}} = \mathrm{E}_{\pi}[c^{\pi}_{\mathcal{F}rej} \mid s^{\times}_0 = s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}})]$, $N^{\pi} = \mathrm{E}_{\pi}[\tau^{\pi}_{BSCC} \mid s^{\times}_0 = s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}})]$, R_n 是定义 6 中的非零奖励，并且我们稍微滥用了记号，将 $R_n = R_K\ \forall n > K$ 设为.....。第 21 行的不等式成立，是因为对于求和中的前 $\tau^{\pi}_{\mathcal{F}}$ 个元素，乘积中至多有 $c^{\pi}_{\mathcal{F}rej}$ 个零奖励折扣因子项；将它们从求和中提出后，只剩下非零奖励项。第 22 行同理：先将求和其余部分的乘积中的 $c^{\pi}_{\mathcal{F}rej}$ 个折扣因子项提出，留下 $(1-R_j)$ 项以及其余的折扣因子 $\gamma(s^{\times}_k)$。根据引理 1，以及以下观察，第 23 行成立：到达接受底层强连通分量后，每个非零奖励在接受状态之间会额外获得 $\gamma^{\tau^{\pi}_{BSCC}}$ 个折扣因子。此外，对无限序列求和可得 $U/(1-\gamma^{\tau^{\pi}_{BSCC}}(1-U)) \le \gamma^{\tau^{\pi}_{BSCC}/U}$，其中 $\gamma \in (0,1)$ 且 $U \in (0,1)$，这给出了额外折扣的上界，并只留下非零奖励的折扣因子 $1-R_j$。最后，根据归纳法，第 25 行成立，因为无限等比级数和 $\sum_{n=0}^{\infty}(1-R_K)^n R_K = 1$，且 $R_n + (1-R_n) * 1 = 1$ 对所有 $n < K$ 均成立。

直观而言，仅包含接受状态的路径的期望折扣奖励为 1；对于满足 $\varphi_{\mathcal{F}}$ 的一般路径，我们通过界定奖励从非零奖励状态获得的折扣量，为其期望折扣奖励给出下界。

接下来，我们考虑第 18 行中的各个部分。令从 $s^{\times}$ 首次到达拒绝集合 $B_{\varnothing}$ 的停止时刻为 $\tau^{\pi}_{\varnothing} := \inf\{t > 0 \mid s^{\times}_t \in B_{\varnothing}\}$，并令 $c^{\pi}_{\varnothing acc} := \left|t \le \tau^{\pi}_{\varnothing} :\ s^{\times}_t \in \mathcal{F}^{\times}\right|$ 表示从 $s^{\times}$ 到达 $B_{\varnothing}$ 之前经过的接受状态数量。我们同样在记号中省略对 $s^{\times}$ 的依赖。于是，我们有

$$G^{\pi}_0(s^{\times} \mid s^{\times} \not\models \mathsf{GF}\varphi_{\mathcal{F}}) = \mathbb{E}_{\pi}\left[\sum_{i=0}^{\infty}(\prod_{j=0}^{i-1}\gamma(s^{\times}_j)) \cdot r^{\times}_i \;\middle|\; s^{\times}_0 = s^{\times} \not\models \mathsf{GF}\varphi_{\mathcal{F}})\right] \tag{26}$$

$$\le \mathbb{E}_{\pi}\left[\sum_{n=0}^{c^{\pi}_{\varnothing acc}}(\prod_{j=0}^{n-1}(1-R_j)) \cdot R_n \;\middle|\; s^{\times}_0 = s^{\times} \not\models \mathsf{GF}\varphi_{\mathcal{F}})\right] \tag{27}$$

$$= \sum_{n=0}^{C^{\pi}_{\varnothing}}(\prod_{j=0}^{n-1}(1-R_j)) \cdot R_n \tag{28}$$

$$\le \sum_{n=0}^{C^{\pi}_{\varnothing}}(1-U)^n \cdot U \tag{29}$$

$$= 1 - (1-U)^{C^{\pi}_{\varnothing}} \tag{30}$$

其中 $C^{\pi}_{\varnothing} = \mathrm{E}_{\pi}[c^{\pi}_{\varnothing acc} \mid s^{\times}_{0} = s^{\times} \not\models \mathsf{GF}\varphi_{\mathcal{F}}]$。第 27 行的不等式成立，是因为根据引理 1，一旦到达 $B_{\varnothing}$，便无法再到达具有非零奖励的接受状态；而总奖励的上界可通过省略非接受状态的折扣因子 γ，仅对到达 $B_{\varnothing}$ 之前经过的 $c^{\pi}_{\varnothing acc}$ 个接受状态的折扣奖励求和来得到。第 28 行通过对 $\tau^{\pi}_{\varnothing}$ 取期望成立；第 29 行根据归纳法成立，因为对于所有 $n \in [0..K]$，若 $X_1 \leq X_2$，则 $R_n + (1 - R_n) * X_1 \leq U + (1-U) * X_2$，且 $U \geq R_n$ 是所有奖励的上界。直观而言，我们已经界定了不满足 $\varphi_{\mathcal{F}}$ 的路径所获得的非零奖励总量。

Therefore, f来自方程17 以及这些技术结果，我们得到了期望折扣奖励的下界

$$G^{\pi}_{0}(s^{\times}) \geq \gamma^{C^{\pi}_{\mathcal{F}} + N^{\pi}/U} \cdot \mathbb{P}_{\pi}(s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}}) \tag{31}$$

假设 $G^{\pi}_{0}(s^{\times} \mid s^{\times} \not\models \mathsf{GF}\varphi_{\mathcal{F}}) = 0$，并得到了期望折扣奖励的上界：

$$G^{\pi}_{0}(s^{\times}) \leq \mathbb{P}_{\pi}(s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}}) + (1 - (1-U)^{C^{\pi}_{\varnothing}}) \cdot \mathbb{P}_{\pi}(s^{\times} \not\models \mathsf{GF}\varphi_{\mathcal{F}}) \tag{32}$$

$$= \mathbb{P}_{\pi}(s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}}) + (1 - (1-U)^{C^{\pi}_{\varnothing}}) \cdot (1 - \mathbb{P}_{\pi}(s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}})) \tag{33}$$

$$= 1 - (1-U)^{C^{\pi}_{\varnothing}} + \mathbb{P}_{\pi}(s^{\times} \models \mathsf{GF}\varphi_{\mathcal{F}}) \cdot (1-U)^{C^{\pi}_{\varnothing}} \tag{34}$$

最后但同样重要的是，由于我们考虑的是有限马尔可夫决策过程中的确定性无记忆策略，因此策略集合是有限的。我们令最优策略与最佳次优策略满足 $\varphi_{\mathcal{F}}$ 的概率之差为 $\delta := \mathrm{P}^{\pi_{\varphi}}(s^{\times}_{0} \models \mathsf{GF}\ \varphi_{\mathcal{F}}) - \max_{\{\pi \neq \pi_{\varphi}\}} \mathrm{P}_{\pi}(s^{\times}_{0} \models \mathsf{GF}\ \varphi_{\mathcal{F}})$。我们可以令奖励上界 $U \in (0,1)$ 足够小，并令 $\gamma \in (0,1)$ 足够大，使得根据第 31 行和第 34 行的界，我们有

$$G^{\pi}_{0}(s^{\times}) \leq \max_{\pi \neq \pi_{\varphi}} \{\mathbb{P}_{\pi}(s^{\times}_{0} \models \mathsf{GF}\ \varphi_{\mathcal{F}})\} + \delta = \mathbb{P}_{\pi_{\varphi}}(s^{\times}_{0} \models \mathsf{GF}\ \varphi_{\mathcal{F}}) \leq G^{\pi_{\varphi}}_{0}(s^{\times}) \quad \forall \pi \neq \pi_{\varphi} \tag{35}$$

这意味着，最优策略 π_{φ} 的期望折扣奖励必须大于任何次优策略所获得的期望折扣奖励。因此，我们可以得出结论：在乘积马尔可夫决策过程 $\mathcal{M}^{\times}$ 上，使满足 B¨uchi 条件 $\varphi_{\mathcal{F}}$ 的概率最大化的策略 π_r maximizes the total expected discounted reward $G^{\pi_r}_{0}(s^{\times})$，也是使满足该条件的概率最大化的最优策略 π_{φ}，至此证明完成。

为选择关键参数 $U \in (0,1)$ 和 $\gamma \in (0,1)$ 以确保最优性，若假设最优策略与次优策略之间存在合理的差距 $\delta \approx 0.5$，通常只需令 $U = 1/C^{\pi}_{\varnothing}$ 和 $\gamma = 1 - 1/(C^{\pi}_{\varnothing} * N^{\pi} + C^{\pi}_{\mathcal{F}})$。例如，对于图 1b 中的概率门控马尔可夫决策过程示例，有 $C^{\pi}_{\varnothing} \approx C^{\pi}_{\mathcal{F}} \approx 10$ 和 $N^{\pi} = 1$，因此选择 $U = 0.1$ 和 $\gamma = 0.95$ 即可。若无法获得命中时间和停止时间，在马尔可夫决策过程的常见假设下，已知 $\mathcal{M}$ 中的状态数量 $|S|$ 以及最小非零转移概率 $p_{\min} := \min_{s,a,s'}\{T(s,a,s') > 0\}$，则 $C^{\pi}_{\varnothing}$ 和 $C^{\pi}_{\mathcal{F}}$ 可由 $|S|/p_{\min}$ 上界，N^{π} 可由 $|S|$ 上界。 □

A.4 定理 3 的证明

证明。 首先，回顾 定理 1 和定理 2 可知，最大化期望折扣奖励的策略 π_r of $\mathcal{M}^{\times}$ 会导出满足$\mathcal{M}^{\times}$ 的最优策略，方法是在 $\mathcal{M}$ 中移除 ϵ-动作。

接下来注意，尽管算法 1 中 Q学习的奖励是非马尔可夫的，但按照 Majeed 和 Hutter [关于非马尔可夫 Q-learning 收敛到最优 Q 值的证明，采用定义 5 中的 K 计数器和定义 6 中的奖励，我们验证算法 1 中的 Q 函数对所有 $(s,\ q)$ 和动作 $a^{\times}$ 都收敛到最优 Q 值；该值等于从 $(s,\ q,\ 0)$ 采取动作 $a^{\times}$ 所获得的期望折扣奖励。

由此，回顾一下，根据方程 16，环境状态和自动机状态的转移不依赖于计数器值；根据定义 6，对于具有相同计数器值的接受状态，非零奖励相同。我们论证，对于最优折扣奖励策略 π_r，新策略 $\pi'((s,\ q\ n)) = \pi_r((s,\ q,\ 0))\ \forall n \in [0..K]$ 仍然是最优策略，其中每个计数器值对应的策略均设为相同。这是因为，如果 π' 不是最优的，也就是说，$\varphi_{\mathcal{F}}$ 的可满足性低于最优策略，那么必然是 π' 在某个计数器值下做出了错误决策，导致到某个接受底层强连通分量 B 的可达概率不是最优的。这意味着 π_r 也不是最优的，因为当计数器值等于 0 时，采取与 π' 相同的动作，到该底强连通分量 B 的可达概率不是最优的，从而产生矛盾。最后，由于 π' 独立于计数器值，忽略 K 计数器后，由算法 1 学得的折叠后的最优 Q 函数导出的策略与最优 π' 相同，这意味着算法 1 收敛到最优策略，从而完成证明。

□

B 补充实验细节与设置

实验在一台 Linux 服务器（Ubuntu 18.04.2）上进行，该服务器配备两颗 Intel Xeon Gold 6252 CPU 和六块 NVIDIA GeForce RTX 2080 Ti GPU。我们所有的算法均使用 Python 实现，完整源代码可在补充材料中找到。我们选择了下文所述的三个随机环境（概率门、冰湖和办公室世界），并设置了若干具有挑战性的线性时序逻辑任务。对于所有环境和任务，我们将学习率设为 $\alpha = 0.1$，探索率设为 $\epsilon = 0.1$，奖励上限设为 $U = 0.1$，折扣因子设为 $\gamma = 0.99$，以确保最优性。对于 Hahn等人 [2019], 提出的方法，我们采用了他们在 Mungojerrie 工具中的实现[2]。我们将默认参数值设为 $\alpha = 0.1$，并

[2]https://plv.colorado.edu/wwwmungojerrie/docs/v1 0/index.html

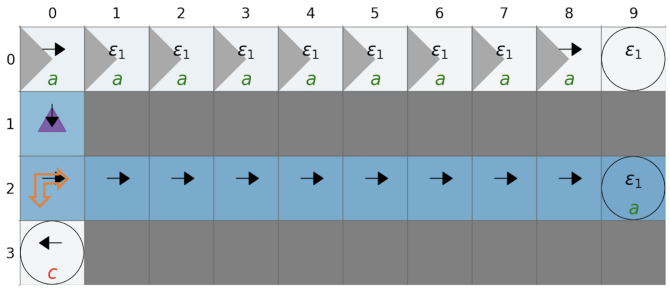
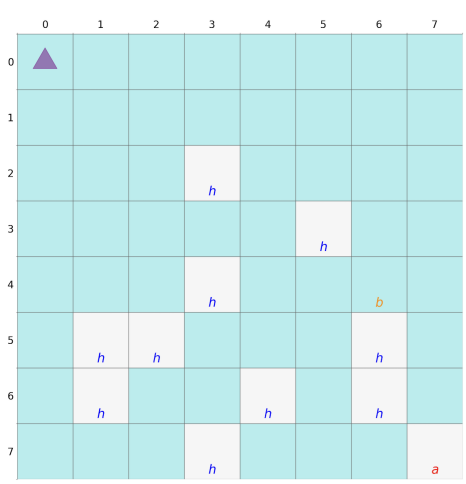
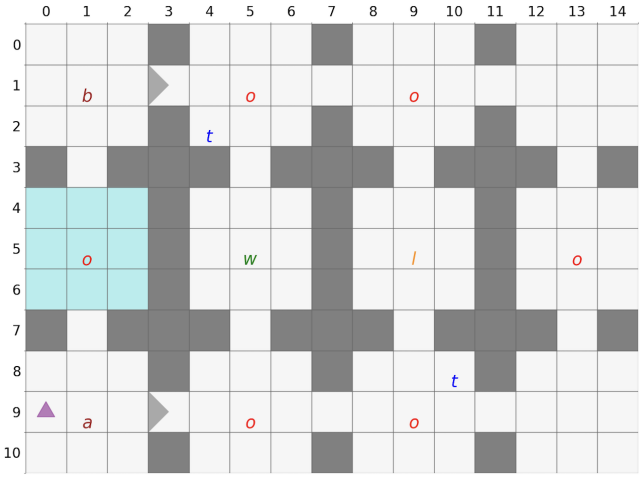


图 4：示例概率门MDP任务的最优策略1



(a) 冰湖任务的MDP环境。蓝色代表冰，“h”表示洞，“a”和“b”表示湖泊营地，紫色三角形表示起点。



(b) 办公室世界任务的MDP环境。蓝色表示冰，“o”表示障碍物，“w”、“l”和“t”分别表示工作场所、信件和茶，紫色三角形表示起点。

图 5：实验中使用的 MDP 环境。

$\epsilon = 0.1$，其中概率门任务采用 $\zeta = 0.995$，冰湖和办公室世界任务采用 $\zeta = 0.99$。对于 Hasanbeig等人 [2020], 提出的方法，我们使用其 GitHub代码库[3] 中的实现，参数设为 $\gamma = 0.99$、$\alpha = 0.1$ 和 $\epsilon = 0.1$。对于 Bozkurt等人 [2019], 提出的方法，我们也采用其 GitHub代码库[4]，参数设为 $\gamma = 0.99$、$\gamma_B = 0.9$、$\alpha = 0.1$ 和 $\epsilon = 0.1$。

对于实验所使用的环境和任务，我们特意选择了常用的、具有随机转移的环境，这些环境更加贴近现实，也更具挑战性，尤其适用于无限时域LTL任务。我们使用的线性时序逻辑任务是机器人学和自主系统中的标准任务规范。冰湖和办公室世界环境中的任务更具挑战性，因为它们要求智能体在两个子任务之间进行选择，这能更好地展示 K 计数器和反事实想象的优势。

B.1 概率门

概率门控马尔可夫决策过程在示例 1 中进行了描述，其中任务是无限次访问标记为“a”的状态，同时不访问标记为“c”的状态。我们将回合长度设置为 100（这是该 MDP 中路径的最大长度），并将回合数设置为 40000。在图 4 中，黑色箭头表示最优策略：智能体从起点向下移动，然后在到达右侧尽头之前前往 (2,9)；(2,9) 是一个接受汇状态。

B.2 冰湖

冰冻湖泊环境 [Brockman 等，2016] 如图 5a 所示。蓝色状态表示冰冻湖泊，在这些状态下，智能体有 1/3 的概率沿预定方向移动，向左或向右侧向移动的概率各为 1/3。标记为“h”的白色状态是洞，标记为“a”和“b”的状态是湖泊营地。任务为“(GF a | GF b) & G !h”，这意味着总是到达湖泊营地“a”或“b”，同时绝不掉入洞“h”。对于此任务，我们将回合长度设为 200，回合数设为 6000。

[3] https://github.com/grockious/lcrl [4] https://github.com/alperkamil/csrl

B.3 办公室世界

办公室世界环境 [Toro Icarte 等，2022] 如图5b所示。其中包含一块以蓝色标记的冰块，其定义与冰冻湖泊环境中的冰块相同；还包含位于 (1, 3) 和 (9,3) 的两扇单向门，智能体只能向右穿过，如灰色三角形所示。灰色方块是墙，标记为“o”“l”“t”和“w”的状态分别表示障碍物、信件、茶和工作场所。此环境的任务同样颇具挑战：“(GF a & GF b) | (F l & X (GF t & GF w)) & G !o”。这意味着要么在“a”和“b”之间的走廊中巡逻，要么前往“l”写信，随后在取茶“t”和工作场所“w”之间巡逻，同时绝不碰到障碍物“o”。对于此任务，我们将回合长度设为 1000，回合数设为 6000。

B.4 PRISM

我们使用 PRISM 4.7 [Kwiatkowska 等人， 2011] 根据线性时序逻辑任务评估学得的策略。PRISM 工具和安装文档可从其官方网站[5]获取。每次需要在环境马尔可夫决策过程上评估策略时，我们的 Python 代码都会自动根据马尔可夫决策过程和策略构建其诱导的马尔可夫链，将其表示为 PRISM 语言中的离散时间马尔可夫链，并通过 Python 调用针对线性时序逻辑任务规范评估该 PRISM 模型。PRISM 的最大迭代次数参数设为 100000；在所有实验中，我们每训练 10000 步评估一次当前策略，以绘制训练曲线。

PRISM 模型示例

我们提供一个示例 PRISM 模型，展示由概率门控马尔可夫决策过程及其最优策略诱导的马尔可夫链。

```
离散时间马尔可夫链

模块 ProductMDP
    m : [0..40] init 10;
    a : [0..3] init 0;
    [事件处理 ](m=0)&(a=0) -> (m'=0)&(a'=1);
    [交流 ](m=0)&(a=1) -> 1 : (m'=1)&(a'=1);
    [交流 ](m=0)&(a=2) -> 1 : (m'=1)&(a'=2);
    [事件处理 ](m=1)&(a=0) -> (m'=1)&(a'=1);
    [交流 ](m=1)&(a=1) -> 1 : (m'=2)&(a'=1);
    [交流 ](m=1)&(a=2) -> 1 : (m'=2)&(a'=2);
    [事件处理 ](m=2)&(a=0) -> (m'=2)&(a'=1);
    [交流 ](m=2)&(a=1) -> 1 : (m'=3)&(a'=1);
    [交流 ](m=2)&(a=2) -> 1 : (m'=3)&(a'=2);
    [事件处理 ](m=3)&(a=0) -> (m'=3)&(a'=1);
    [交流 ](m=3)&(a=1) -> 1 : (m'=4)&(a'=1);
    [交流 ](m=3)&(a=2) -> 1 : (m'=4)&(a'=2);
    [事件处理 ](m=4)&(a=0) -> (m'=4)&(a'=1);
    [交流 ](m=4)&(a=1) -> 1 : (m'=5)&(a'=1);
    [交流 ](m=4)&(a=2) -> 1 : (m'=5)&(a'=2);
    [事件处理 ](m=5)&(a=0) -> (m'=5)&(a'=1);
    [交流 ](m=5)&(a=1) -> 1 : (m'=6)&(a'=1);
    [交流 ](m=5)&(a=2) -> 1 : (m'=6)&(a'=2);
    [事件处理 ](m=6)&(a=0) -> (m'=6)&(a'=1);
    [交流 ](m=6)&(a=1) -> 1 : (m'=7)&(a'=1);
    [交流 ](m=6)&(a=2) -> 1 : (m'=7)&(a'=2);
    [事件处理 ](m=7)&(a=0) -> (m'=7)&(a'=1);
    [交流 ](m=7)&(a=1) -> 1 : (m'=8)&(a'=1);
    [交流 ](m=7)&(a=2) -> 1 : (m'=8)&(a'=2);
    [事件处理 ](m=8)&(a=0) -> (m'=8)&(a'=1);
    [交流 ](m=8)&(a=1) -> 1 : (m'=9)&(a'=1);
    [交流 ](m=8)&(a=2) -> 1 : (m'=9)&(a'=2);
    [事件处理 ](m=9)&(a=0) -> (m'=9)&(a'=1);
    [交流 ](m=9)&(a=1) -> 1.0 : (m'=9)&(a'=2);
    [交流 ](m=9)&(a=2) -> 1.0 : (m'=9)&(a'=2);
    [交流 ](m=10)&(a=0) -> 1 : (m'=20)&(a'=0);
    [交流 ](m=10)&(a=1) -> 1 : (m'=20)&(a'=2);
    [交流 ](m=10)&(a=2) -> 1 : (m'=20)&(a'=2);
    [交流 ](m=11)&(a=0) -> 1.0 : (m'=11)&(a'=0);
    [交流 ](m=11)&(a=1) -> 1.0 : (m'=11)&(a'=2);
    [交流 ](m=11)&(a=2) -> 1.0 : (m'=11)&(a'=2);
    [交流 ](m=12)&(a=0) -> 1.0 : (m'=12)&(a'=0);
```

[5] https://www.prismmodelchecker.org/download.php

```
[交流](m=12)&(a=1) -> 1.0 : (m'=12)&(a'=2);
[交流](m=12)&(a=2) -> 1.0 : (m'=12)&(a'=2);
[交流](m=13)&(a=0) -> 1.0 : (m'=13)&(a'=0);
[交流](m=13)&(a=1) -> 1.0 : (m'=13)&(a'=2);
[交流](m=13)&(a=2) -> 1.0 : (m'=13)&(a'=2);
[交流](m=14)&(a=0) -> 1.0 : (m'=14)&(a'=0);
[交流](m=14)&(a=1) -> 1.0 : (m'=14)&(a'=2);
[交流](m=14)&(a=2) -> 1.0 : (m'=14)&(a'=2);
[交流](m=15)&(a=0) -> 1.0 : (m'=15)&(a'=0);
[交流](m=15)&(a=1) -> 1.0 : (m'=15)&(a'=2);
[交流](m=15)&(a=2) -> 1.0 : (m'=15)&(a'=2);
[交流](m=16)&(a=0) -> 1.0 : (m'=16)&(a'=0);
[交流](m=16)&(a=1) -> 1.0 : (m'=16)&(a'=2);
[交流](m=16)&(a=2) -> 1.0 : (m'=16)&(a'=2);
[交流](m=17)&(a=0) -> 1.0 : (m'=17)&(a'=0);
[交流](m=17)&(a=1) -> 1.0 : (m'=17)&(a'=2);
[交流](m=17)&(a=2) -> 1.0 : (m'=17)&(a'=2);
[交流](m=18)&(a=0) -> 1.0 : (m'=18)&(a'=0);
[交流](m=18)&(a=1) -> 1.0 : (m'=18)&(a'=2);
[交流](m=18)&(a=2) -> 1.0 : (m'=18)&(a'=2);
[交流](m=19)&(a=0) -> 1.0 : (m'=19)&(a'=0);
[交流](m=19)&(a=1) -> 1.0 : (m'=19)&(a'=2);
[交流](m=19)&(a=2) -> 1.0 : (m'=19)&(a'=2);
[交流](m=20)&(a=0) -> 0.8 : (m'=21)&(a'=0) + 0.2 : (m'=30)&(a'=0);
[交流](m=20)&(a=1) -> 0.8 : (m'=21)&(a'=2) + 0.2 : (m'=30)&(a'=2);
[交流](m=20)&(a=2) -> 0.8 : (m'=21)&(a'=2) + 0.2 : (m'=30)&(a'=2);
[交流](m=21)&(a=0) -> 1 : (m'=22)&(a'=0);
[交流](m=21)&(a=1) -> 1.0 : (m'=21)&(a'=2);
[交流](m=21)&(a=2) -> 1.0 : (m'=21)&(a'=2);
[交流](m=22)&(a=0) -> 1 : (m'=23)&(a'=0);
[交流](m=22)&(a=1) -> 1.0 : (m'=22)&(a'=2);
[交流](m=22)&(a=2) -> 1.0 : (m'=22)&(a'=2);
[交流](m=23)&(a=0) -> 1 : (m'=24)&(a'=0);
[交流](m=23)&(a=1) -> 1.0 : (m'=23)&(a'=2);
[交流](m=23)&(a=2) -> 1.0 : (m'=23)&(a'=2);
[交流](m=24)&(a=0) -> 1 : (m'=25)&(a'=0);
[交流](m=24)&(a=1) -> 1.0 : (m'=24)&(a'=2);
[交流](m=24)&(a=2) -> 1.0 : (m'=24)&(a'=2);
[交流](m=25)&(a=0) -> 1 : (m'=26)&(a'=0);
[交流](m=25)&(a=1) -> 1.0 : (m'=25)&(a'=2);
[交流](m=25)&(a=2) -> 1.0 : (m'=25)&(a'=2);
[交流](m=26)&(a=0) -> 1 : (m'=27)&(a'=0);
[交流](m=26)&(a=1) -> 1.0 : (m'=26)&(a'=2);
[交流](m=26)&(a=2) -> 1.0 : (m'=26)&(a'=2);
[交流](m=27)&(a=0) -> 1 : (m'=28)&(a'=0);
[交流](m=27)&(a=1) -> 1.0 : (m'=27)&(a'=2);
[交流](m=27)&(a=2) -> 1.0 : (m'=27)&(a'=2);
[交流](m=28)&(a=0) -> 1 : (m'=29)&(a'=0);
[交流](m=28)&(a=1) -> 1 : (m'=27)&(a'=2);
[交流](m=28)&(a=2) -> 1 : (m'=27)&(a'=2);
[事件处理](m=29)&(a=0) -> (m'=29)&(a'=1);
[交流](m=29)&(a=1) -> 1.0 : (m'=29)&(a'=1);
[交流](m=29)&(a=2) -> 1.0 : (m'=29)&(a'=2);
[事件处理](m=30)&(a=0) -> (m'=30)&(a'=1);
[交流](m=30)&(a=1) -> 1.0 : (m'=30)&(a'=2);
[交流](m=30)&(a=2) -> 1.0 : (m'=30)&(a'=2);
[交流](m=31)&(a=0) -> 1.0 : (m'=31)&(a'=0);
[交流](m=31)&(a=1) -> 1.0 : (m'=31)&(a'=2);
[交流](m=31)&(a=2) -> 1.0 : (m'=31)&(a'=2);
[交流](m=32)&(a=0) -> 1.0 : (m'=32)&(a'=0);
[交流](m=32)&(a=1) -> 1.0 : (m'=32)&(a'=2);
[交流](m=32)&(a=2) -> 1.0 : (m'=32)&(a'=2);
[交流](m=33)&(a=0) -> 1.0 : (m'=33)&(a'=0);
[交流](m=33)&(a=1) -> 1.0 : (m'=33)&(a'=2);
[交流](m=33)&(a=2) -> 1.0 : (m'=33)&(a'=2);
```

```
    [交流](m=34)&(a=0) -> 1.0 : (m'=34)&(a'=0);
    [交流](m=34)&(a=1) -> 1.0 : (m'=34)&(a'=2);
    [交流](m=34)&(a=2) -> 1.0 : (m'=34)&(a'=2);
    [交流](m=35)&(a=0) -> 1.0 : (m'=35)&(a'=0);
    [交流](m=35)&(a=1) -> 1.0 : (m'=35)&(a'=2);
    [交流](m=35)&(a=2) -> 1.0 : (m'=35)&(a'=2);
    [交流](m=36)&(a=0) -> 1.0 : (m'=36)&(a'=0);
    [交流](m=36)&(a=1) -> 1.0 : (m'=36)&(a'=2);
    [交流](m=36)&(a=2) -> 1.0 : (m'=36)&(a'=2);
    [交流](m=37)&(a=0) -> 1.0 : (m'=37)&(a'=0);
    [交流](m=37)&(a=1) -> 1.0 : (m'=37)&(a'=2);
    [交流](m=37)&(a=2) -> 1.0 : (m'=37)&(a'=2);
    [交流](m=38)&(a=0) -> 1.0 : (m'=38)&(a'=0);
    [交流](m=38)&(a=1) -> 1.0 : (m'=38)&(a'=2);
    [交流](m=38)&(a=2) -> 1.0 : (m'=38)&(a'=2);
    [交流](m=39)&(a=0) -> 1.0 : (m'=39)&(a'=0);
    [交流](m=39)&(a=1) -> 1.0 : (m'=39)&(a'=2);
    [交流](m=39)&(a=2) -> 1.0 : (m'=39)&(a'=2);
模块结束

标签 "一个" = (m=0) | (m=1) | (m=2) | (m=3) | (m=4) | (m=5) | (m=6) | (m=7) | (m=8) | (m=29);
label "c" = (m=30);
```

列表 1：由概率门控马尔可夫决策过程及其最优策略诱导的马尔可夫链的 PRISM 模型。

B.5 Rabinizer

我们使用 Rabinizer 4 [Křetínský 等人, 2018] 将线性时序逻辑公式转换为极限确定性 Büchi 自动机。该工具可从其网站下载[6]，我们使用带有“-e”和“-d”选项的“ltl2ldba”脚本，从输入的线性时序逻辑公式构造具有 ϵ-转移的极限确定性 Büchi 自动机。

[6] https://www7.in.tum.de/ kretinsk/rabinizer4.html