控制与决策
Control and Decision

线性时序逻辑引导的安全强化学习

李保罗, 蔡明钰, 阚震

引用本文:
李保罗,蔡明钰,阚震. 线性时序逻辑引导的安全强化学习[J]. 控制与决策, 2023, 38(7): 1835−1844.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2021.1808

线性时序逻辑引导的安全强化学习

李保罗[1], 蔡明钰[2], 阚 震[1†]

(1. 中国科学技术大学 自动化系，合肥 230026；
2. 理海大学 机械工程系，伯利恒 18015)

摘 要: 针对动态不确定环境下机器人执行复杂任务的需求,提出一种线性时序逻辑(linear temporal logic, LTL)引导的无模型安全强化学习算法,能在最大化任务完成概率的同时保证学习过程的安全性. 首先,综合考虑环境中的不确定因素,构建马尔可夫决策过程(Markov decision process, MDP),再用LTL刻画智能体的复杂任务,将其转化为有多接受集的基于转移的有限确定性广义布奇自动机(transition-based limit deterministic generalized Büchi automaton, tLDGBA),并通过接受边界函数构建可记录当前待访问接受集的约束型tLDGBA (constrained tLDGBA, ctLDGBA);其次,构建乘积MDP用于强化学习搜索最优策略;最后,基于LTL对安全性的描述和MDP的观测函数构建安全博弈,并根据安全博弈设计安全盾机制保证系统在学习过程中的安全性. 严格的分析证明了所提出的算法能获得最大化LTL任务完成概率的最优策略. 仿真结果验证了LTL引导的安全强化学习算法的有效性.

关键词: 线性时序逻辑；自动机；马尔可夫决策过程；强化学习；安全博弈；运动规划

中图分类号: TP242 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2021.1808

引用格式: 李保罗,蔡明钰,阚震. 线性时序逻辑引导的安全强化学习[J]. 控制与决策, 2023, 38(7): 1835-1844.

Linear temporal logic guided safe reinforcement learning

LI Bao-luo[1], *CAI Ming-yu*[2], *KAN Zhen*[1†]

(1. Department of Automation，University of Science and Technology of China，Hefei 230026，China；2. Department of Mechanical Engineering，Lehigh University，Bethlehem 18015，USA)

Abstract: This paper presents a linear temporal logic (LTL) guided model-free safe reinforcement learning algorithm to synthesize a control policy that maximizes the satisfaction probability of complex task in an unknown stochastic environment and ensures the safety of agent during learning process. Considering environmental uncertainties, the probabilistic motion of the robot is modeled as a Markov decision process (MDP) with unknown transition probabilities. LTL is applied to describe the complex task, which can be converted to a transition-based limit deterministic generalized Büchi automaton (tLDGBA) with several accepting sets. The accepting frontier function is then designed to record the visited accepting sets, which gives rise to a constrained tLDGBA (ctLDGBA). To ensure the system safety, based on the safety fragment of the LTL formula and the observation function of the MDP, a safety game is constructed to synthesize a shield that ensures the system safety during the learning process. Rigorous analysis shows that the proposed safe reinforcement learning method is guaranteed to obtain the optimal policy that maximizes the probability of satisfying the LTL task while ensuring system safety. The effectiveness of the LTL guided safe reinforcement learning algorithm is demonstrated via simulation results.

Keywords: linear temporal logic；automaton；Markov decision process；reinforcement learning；safety game；motion planning

0 引 言

机器人运动规划的目的是生成满足预定任务的路径. 然而,由于环境干扰、动力学模型不准确等因素引起的运动不确定性给机器人运动规划带来了极大的挑战. 现有研究中,研究者们常用马尔可夫决策过程(Markov decision process, MDP)来建模这种不确定性[1],并且由于环境未知导致MDP的转移概率未知,往往用强化学习(reinforcement learning, RL)方法通

收稿日期: 2021-10-21；**录用日期:** 2022-03-28.
基金项目: 国家自然科学基金面上项目(62173314)；国家自然科学基金联合基金项目(U2013601).
责任编委: 林崇.
†通讯作者. E-mail: zkan@ustc.edu.cn.
*本文附带电子附录文件,可登录本刊官网该文“资源附件”区自行下载阅览.

过对MDP进行采样来逐步获得最优控制策略[2-6]. 当前用于运动规划的各种强化学习方法仍然存在以下两个问题:一是如何恰当地描述运动规划任务并生成合适的回报结构;二是如何确保机器人系统在强化学习过程中的安全性,如避免撞到障碍物和从高处跌落等.

机器人运动规划领域的新趋势是研究完成高层复杂任务的计算框架,不同于仅能实现到达单个目标位置的传统方法,这类新框架能解决包含复杂的逻辑和时序约束的高层规划任务,如序列性任务(依次到达目标区域A、B和C)、持续监视任务(无限次访问目标区域A、B和C)等,以及这些任务复杂的逻辑组合. 线性时序逻辑(linear temporal logic, LTL)是这类新框架中常用的形式化语言,能描述丰富的高层复杂任务[7],近年来将LTL运用于运动规划的研究越来越多[8-16],使用LTL可以解决描述运动规划任务及检验任务完成进度的问题. 当MDP先验未知时,可以根据LTL任务转化得到自动机的接受条件生成相应的回报结构,采用基于模型或无模型强化学习算法获得最优策略. 基于模型的强化学习方法[12-13]能够通过探索状态空间并学习全部的MDP转移概率求出最优策略,但这类方法需要存储整个MDP模型而消耗大量的存储空间,可扩展性不强. 无模型的强化学习方法[14-16]虽然克服了上述缺陷,且能得到最大化回报的策略,但无法严格证明学到的最大化期望回报的策略同时是最大化LTL任务完成概率的策略,即无法保证得到的策略是满足LTL任务的最优策略.

强化学习是探索式的学习方法,机器人需要尽可能充分地探索MDP状态空间才能得到较好的控制策略,无法保证系统在探索过程中的安全性,易导致系统损坏,因此保证机器人在学习过程中的安全性至关重要. 近年来,安全强化学习的研究受到广泛关注[17-23]. 文献[21]通过控制屏障函数(control barrier functions, CBFs)保证机器人系统在强化学习过程中的安全,但该方法只能用于有限路径规划,并且要求系统模型精确已知;文献[22]能根据学到的MDP转移概率来估计系统进入不安全状态的概率,但存储全部转移概率消耗的存储空间过大,可扩展性差;文献[23]基于安全博弈提出了安全盾方法,用非常有限的环境信息来确保系统的安全性,较好地克服了上述方法的缺点,但不能直接用于LTL描述的高层复杂任务中.

针对上述研究方法中存在的问题,本文提出一种新的安全强化学习方法,用于解决未知MDP中完成预定LTL任务的运动规划问题. 首先,用LTL描述高层复杂任务并转化为基于转移的有限确定性广义布奇自动机(transition-based limit-deterministic generalized Büchi automaton,tLDGBA)[24];然后,用自动机和MDP构建乘积MDP,在乘积MDP上求解最优控制策略;最后,将安全盾方法拓展到LTL引导的强化学习中,保障机器人系统安全. 本文的主要工作包括以下4点:1) 提出一种新型增广tLDGBA,并将其用于指引强化学习;2) 将安全盾方法拓展到LTL任务中,提出LTL引导的安全强化学习算法;3) 严格证明通过本文方法求出的策略的最优性;4) 通过仿真实验验证本文方法的有效性.

1 基础理论

1.1 马尔可夫决策过程

定义1[7](带标签的) 马尔可夫决策过程由多元组$M=(S,A,P,s_0,\mathrm{AP},L)$表示. 其中:$S$是有限状态集,$A$是有限动作集,$P:S\times A\times S\to[0,1]$是状态转移概率函数,$s_0\in S$是初始状态,AP是有限原子命题集,$L:S\to 2^{\mathrm{AP}}$是标签函数. 记状态$s$处的可行动作集为

$$A(s)=\{a\in A|\exists s'\in S,P(s,a,s')\neq 0\},$$

对于任意状态$s\in S$,若有$a\in A(S)$,则$\sum\limits_{s'\in S}P(s,a,s')=1$,否则$\sum\limits_{s'\in S}P(s,a,s')=0$.

在MDP M中,从初始状态开始的路径定义为无限状态序列$\sigma=s_0s_1s_2\ldots\in S^{\omega}$,路径$\sigma$对应的标签路径定义为

$$L(\sigma)=L(s_0)L(s_1)L(s_2)\ldots\in(2^{\mathrm{AP}})^{\omega}.$$

简单起见,后文用$\sigma[i]$表示路径上对应的状态s_i,用$\sigma[:i]=s_0s_1\ldots s_i$表示路径的前缀.

定义2[7] MDP上的确定性策略π定义为$\pi:S^*\to A$,其中S^*表示全部有限路径的集合. 无记忆确定性策略为

$$\pi(\sigma[:n])=\pi(\sigma[n]),\ \forall\sigma[:n]\in S^*,a\in A(\sigma[n]).$$

定义3 给定无记忆策略π,MDP在π下生成的马尔可夫链由元组$\mathrm{MC}_{\pi}=(S,P_{\pi},s_0)$表示[7],其中转移概率函数为

$$P_{\pi}(s,s')=P(s,\pi(s),s'),\ \forall s,s'\in S.$$

1.2 线性时序逻辑

本文采用LTL公式描述机器人待完成的任务,并生成相应的回报. LTL公式可由布尔真值(True)、原

子命题($a \in \mathrm{AP}$)、合取($\bigwedge$)、非($\neg$)、接下来($\bigcirc$)和直到($\bigcup$)递归地构成,具体语法如下:

$$\varphi ::= \mathrm{True}|a|\varphi_1 \bigwedge \varphi_2|\neg\varphi|\bigcirc\varphi|\varphi_1 \bigcup \varphi_2,\ a \in \mathrm{AP}, \tag{1}$$

其中φ_1、φ_2和φ均为LTL公式. 其他常用的布尔符号和时序符号定义如下: $\varphi_1 \bigvee \varphi_2 := \neg(\neg\varphi_1 \bigwedge \neg\varphi_2)$(析取); $\varphi_1 \to \varphi_2 := \neg\varphi_1 \bigvee \varphi_2$(若...则...); $\Diamond\varphi := \mathrm{True} \bigcup \varphi$(最终); $\Box\varphi := \neg(\Diamond\neg\varphi)$(总是). 对于MDP M的一条无限路径σ和LTL公式φ,满足关系记为$\sigma \vDash \varphi$[7]. 满足关系可用tLDGBA检验,在定义tLDGBA之前,先给出基于转移的广义布奇自动机(transition-based generalized Büchi automaton, tGBA)[7]的定义.

定义4　tGBA由元组$\mathcal{A}_\varphi = (Q, q_0, \Sigma, \delta, \mathcal{F})$表示. 其中: Q是有限状态集; $q_0 \in Q$是初始状态; $\Sigma = 2^{\mathrm{AP}}$是输入字母表; $\delta: Q \times \Sigma \to 2^Q$是转移集合; $\mathcal{F} = \{F_0, F_1, \ldots, F_{k-1}\}, F_j \subset \delta, \forall j \in \{0, 1, \ldots, k-1\}$,是接受条件.

tGBA的运行记为$\boldsymbol{q} = q_0 l_0 q_1 \ldots \in Q(\Sigma Q)^\omega$, $(q_i, l_i, q_{i+1}) \in \delta, \forall i \in \boldsymbol{N}, l_i \in \Sigma$,若该运行满足tGBA的接受条件: $\inf(\boldsymbol{q}) \bigcap F_j \neq \varnothing, \forall F_j \in \mathcal{F}$,则运行$\boldsymbol{q}$被tGBA接受. 其中, $\inf(\boldsymbol{q})$是运行$\boldsymbol{q}$中无限次出现的转移的集合. 下面给出tLDGBA的定义.

定义5　tGBA $\mathcal{A}_\varphi = (Q, q_0, \Sigma, \delta, \mathcal{F})$称为tLDGBA[24],若tGBA的输入字母表扩充为$\Sigma = 2^{\mathrm{AP}} \bigcup \{\epsilon\}$,且其状态集可被划分为互不相交的两个集合: $Q = Q_I \bigcup Q_D, Q_I \bigcap Q_D = \varnothing$,使得

1) $F_j \subset Q_D \times \Sigma \times Q_D, \forall j \in \{0, 1, k-1\}$;

2) $|\{(q, l, q') \in \delta | l \in \Sigma, q' \in Q_D\}| = 1, \forall q \in Q_D$;

3) $|\{(q, l, q') \in \delta | l \in \Sigma, q' \in Q_I\}| = 0, \forall q \in Q_D$;

4) $\forall (q, \epsilon, q') \in \delta, q \in Q_I \bigwedge q' \in Q_D$.

1.3　安全博弈

定义6　二玩家安全博弈由元组$\mathcal{G} = (G, g_0, \Sigma_1, \Sigma_2, \delta_g, F_g)$表示[23]. 其中: G是博弈状态集; $g_0 \in G$是初始状态; Σ_1和Σ_2分别是玩家1和玩家2的输入字母表; $\delta_g: G \times \Sigma_1 \times \Sigma_2 \to G$是博弈的转移函数; $F_g \subseteq G$是安全状态集. 对于玩家2,一次博弈$\bar{g} = g_0 g_1 \ldots$的结果是获胜,当且仅当$g_i \in F_g, \forall i \geqslant 0$. 玩家2的获胜策略是函数$\rho: G \times \Sigma_1 \to \Sigma_2$,其获胜区域$W \subseteq F_g$是存在获胜策略的安全状态集的子集.

2　问题描述

考虑转移概率未知的MDP M和LTL公式φ描述的任务,本文的目标是自动求出最大化LTL任务满足概率的最优控制策略π^*,同时保证机器人系统在优化策略过程中的安全性.

问题1　给定转移概率未知的MDP $M = (S, A, P, s_0, \mathrm{AP}, L)$和LTL公式$\varphi$,提出一个无模型安全强化学习算法来找到最优策略$\pi^*$. 该策略满足

$$\Pr^{\pi^*}(s \vDash \varphi) = \Pr_{\max}(s \vDash \varphi). \tag{2}$$

其中

$$\Pr^{\pi^*}(s \vDash \varphi) = \Pr^{\pi^*}(\sigma \vDash \varphi | \sigma[0] = s),$$

$$\Pr_{\max}(s \vDash \varphi) = \max_\pi \Pr^\pi(s \vDash \varphi), \forall s \in S.$$

3　基于安全强化学习的控制策略求解

本文方法如图1所示. 先将LTL公式转化为本文设计的自动机;再通过自动机和MDP构建乘积MDP,在乘积MDP上定义相应的回报结构,并从理论上证明使用强化学习算法优化期望回报等价于优化LTL任务满足概率;最后引入安全盾保证学习过程的安全性.

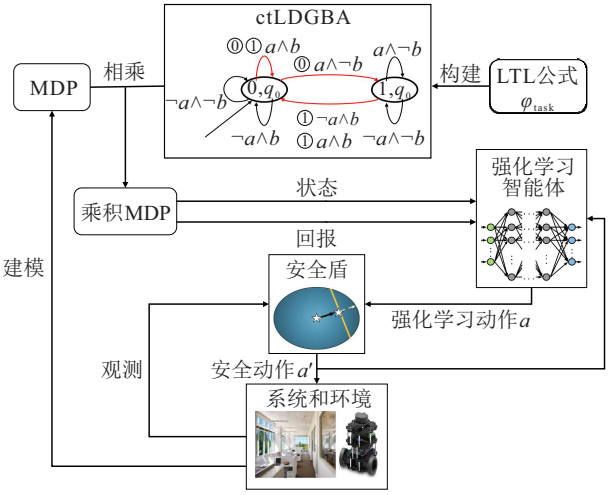


图1　本文方法图示

3.1　约束型tLDGBA

本小节将tLDGBA拓展为约束型tLDGBA(ctLDGBA),拓展原因详见3.2节.

给定一个有k个接受集的tLDGBA $\mathcal{A}_\varphi = (Q, q_0, \Sigma, \delta, \mathcal{F})$,定义集合$V = \{v | v \in \{0, 1, \ldots, k-1\}\}$记录接受集的序号. 给定tLDGBA的一条转移$(q, l, q') \in \delta$,定义函数$\mathrm{idx}: \delta \to 2^V$来记录该转移访问的接受集,即$\mathrm{idx}(q, l, q') = \{j \in V | (q, l, q') \in F_j\}$. 由此,可定义ctLDGBA.

定义7　给定tLDGBA $\mathcal{A}_\varphi = (Q, q_0, \Sigma, \delta, \mathcal{F})$,其对应的ctLDGBA为$\overline{\mathcal{A}}_\varphi = (\bar{Q}, \bar{q}_0, \Sigma, \bar{\delta}, \bar{\mathcal{F}}, f_{\mathcal{A}}, T)$. 其中: $\bar{Q} = V \times Q$是增广状态集. $\bar{q}_0 = (0, q_0)$是初始状态. T是接受边界集[25],记录了未被访问的接受集,初始化为$\mathcal{F}$,并且由接受边界函数更新:

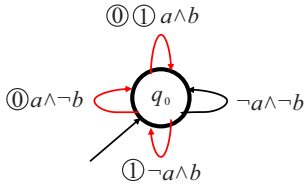
$$f_{\mathcal{A}}((v, q), T) =$$

$$\begin{cases} T\backslash F_j, (v \in \text{idx}(q,l,q')) \bigwedge (F_j \in T); \\ \mathcal{F}\backslash F_j, (v \in \text{idx}(q,l,q')) \bigwedge (T = \varnothing); \\ T, \text{ otherwise.} \end{cases}$$

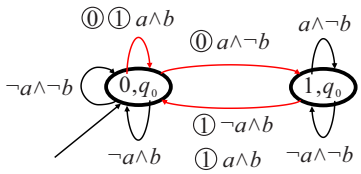
$$\forall (v,q) \in \bar{Q}. \tag{3}$$

新的转移函数定义为$\bar{\delta} = \{((v,q),l,(v',q')) \in \bar{Q} \times \Sigma \times \bar{Q} | (q,l,q') \in \delta, v' = v_\text{next}(v,q,T)\}$, 其中$v_\text{next}(v,q,T)$是ctLDGBA状态分量$v$的转移函数. 若在接受边界集$T$被式(3)更新之后, F_i是T的首个集合, 则$v_\text{next}(v,q,T) = i, F_i = T[0]$; 若$T = \varnothing$, 则$v_\text{next}(v,q,T)$被重置为0, 新的接受条件变为$\bar{\mathcal{F}} = \{\bar{F}_0, \bar{F}_1, \ldots, \bar{F}_{k-1}\}, \bar{F}_j = \{((v,q),l,(v',q')) \in \bar{\delta} | (q,l,q) \in F_j, v \leqslant j\}$.

在定义7中, 用v来增广tLDGBA的状态, v指向接受边界集中的首个集合, 表示当前待访问的接受集. 图2展示了LTL公式$\varphi = \square\Diamond a \bigwedge \square\Diamond b$对应的tLDGBA和ctLDGBA.



(a) tLDGBA



(b) ctLDGBA

图 2 LTL公式对应的自动机

显然, ctLDGBA改变了tLDGBA的结构, 但易证本文设计的ctLDGBA不会改变tLDGBA的接受语言, 有以下定理成立.

定理 1 tLDGBA $\mathcal{A}_\varphi$和对应的ctLDGBA $\bar{\mathcal{A}}_\varphi$具有相同的接受语言, 即$\mathcal{L}(\mathcal{A}_\varphi) = \mathcal{L}(\bar{\mathcal{A}}_\varphi)$.

定理1说明了可用ctLDGBA检验是否完成LTL任务, 限于篇幅略去详细证明过程, 通过分析tLDGBA和ctLDGBA在任意相同输入词$w = l_0 l_1 \ldots$ $(l_i \in \Sigma, \forall i \in \boldsymbol{N})$下生成的运行所访问的对应接受集$\mathcal{F}$和$\bar{\mathcal{F}}$的情况可证.

3.2 乘积MDP

MDP和ctLDGBA可融合为乘积MDP, 用于强化学习搜索最优策略.

定义 8 给定MDP $M = (S, A, P, s_0, \text{AP}, L)$和ctLDGBA $\bar{\mathcal{A}}_\varphi = (\bar{Q}, \bar{q}_0, \Sigma, \bar{\delta}, \bar{\mathcal{F}}, f_{\mathcal{A}}, T)$, 乘积MDP定义为$M^\times = M \times \bar{\mathcal{A}}_\varphi = (S^\times, A^\times, P^\times, s_0^\times, \delta^\times, \mathcal{F}^\times)$. 其中: $S^\times = S \times \bar{Q}$是乘积状态集. $A^\times = A \bigcup A^\epsilon, A^\epsilon := \{\epsilon_{\bar{q}'} | \exists \bar{q}' \in \bar{Q}, \text{s.t.} (\bar{q}, \epsilon, \bar{q}') \in \bar{\delta}\}$是动作集. $s_0^\times = (s_0, \bar{q}_0)$是初始状态. 乘积MDP的状态转移概率函数定义为

$$P^\times((s,\bar{q}),a^\times,(s',\bar{q}')) = \begin{cases} P(s,a^\times,s'), \ \bar{q}' = \bar{\delta}(\bar{q}, L(s)) \bigwedge a^\times \notin A^\epsilon; \\ 1, \ s = s' \bigwedge a^\times = \epsilon_{\bar{q}'} \bigwedge \bar{q}' \in \bar{\delta}(\bar{q}, \epsilon); \\ 0, \text{ otherwise.} \end{cases}$$

$\delta^\times = \{(s^\times, a^\times, (s^\times)') \in S^\times \times A^\times \times S^\times | P^\times(s^\times, a^\times, (s^\times)') > 0\}$是可行转移集; $\mathcal{F}^\times = \{\bar{F}_0^\times, \ldots, \bar{F}_{k-1}^\times\}$, $\bar{F}_j^\times = \{((s,\bar{q}), a^\times, (s',\bar{q}')) \in \delta^\times | (\bar{q}, L(s), \bar{q}') \in \bar{F}_j\}$, $\forall j \in \{0,1,\ldots,k-1\}$, 是接受条件.

乘积MDP融合了MDP和LTL任务的进度信息, 一个满足乘积MDP接受条件的策略能在MDP上生成满足LTL公式的路径, 具体而言, 文献[26]中的引理1对本文定义的乘积MDP仍然成立, 即, 在MDP上找到满足LTL公式的策略等价于在乘积MDP上找到满足乘积MDP接受条件的无记忆策略. 若乘积MDP的极大强连通分量的全部转移与每个接受集的交集均不为空, 则该极大强连通分量称为接受极大强连通分量(accepting maximum end component, AMEC)[7], 一旦路径进入AMEC中, 就存在策略使后续路径全在AMEC中, AMEC中的每条转移都能被无限次访问, 故完成LTL任务可以等价于进入一个AMEC. 此外, 极大强连通分量与任何一个接受集均不相交, 则称为拒绝极大强连通分量(rejecting MEC, RMEC); 仅与部分接受集相交, 则称为中立极大强连通分量(neutral MEC, NMEC)[7].

注 1 定义乘积MDP之后, 本注将说明定义ctLDGBA的原因. 图2给出了LTL公式$\varphi = \square\Diamond a \wedge \square\Diamond b$对应的tLDGBA和ctLDGBA, 该公式要求智能体无限次地访问目标a和b. 给定图3(a)所示的MDP, 其原子命题集为$\text{AP} = \{a, b\}$, 动作集为$A = \{L, R\}$, 通过该MDP和图2(a)的tLDGBA生成图3(b)所示的乘积MDP $M^{\times\prime}$. 考虑$M^{\times\prime}$上的无记忆策略π, 该策略如图3(b)中蓝色箭头所示. 在初始状态(s_0, q_0)处, 策略为$\pi(s_0, q_0) = L$, 转移到状态(s_1, q_0), 访问到目标a, 为了访问目标b, 在该状态的策略只能为$\pi(s_1, q_0) = R$, 采取动作R后到达状态(s_0, q_0). 由于乘积MDP上的策略π是无记忆的, 只能采取动作L, 最终在状态(s_0, q_0)和(s_1, q_0)间循环, 无法访问到目标b, 违背了LTL任务需求, 故tLDGBA不能用于构

建乘积MDP. 容易验证在本例中使用图2(b)所示的ctLDGBA能够得到满足LTL公式的策略,本文提出的ctLDGBA能够用于构建乘积MDP.

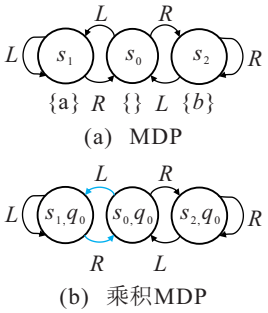


图 3　MDP和乘积MDP示例

假设 1　设乘积MDP $M^\times$ 上至少存在一个满足LTL公式的策略.

假设1是使问题1可解的基础条件,该假设被广泛应用于相关文献[14, 26-27]中. 记乘积MDP $M^\times$ 上的一个无记忆策略 π 生成的马尔可夫链为 $\mathrm{MC}_{M^\times}^{\pi} = (S_\pi^\times, P_\pi^\times, s_0^\times)$,据文献[28]可知,马尔可夫链 $\mathrm{MC}_{M^\times}^{\pi}$ 的状态可以划分为互不相交的一个瞬时类 $\mathcal{T}_\pi$ 和 n 个不可约常返闭集 $R_\pi^i (i \in \{1,2,\dots,n\})$,即 $S_\pi^\times = \mathcal{T}_\pi \bigcup R_\pi^1 \bigcup \dots \bigcup R_\pi^n$. 记一个常返类 R_π^j 中转移的集合为 $\delta_\pi^j = \{(s^\times, a^\times, (s^\times)') \in \delta^\times | s^\times \in R_\pi^j, P^\times(s^\times, a^\times, (s^\times)') > 0\}$. 定义8中的乘积MDP有如下重要特性.

引理 1　给定乘积MDP $M^\times = M \times \bar{\mathcal{A}}_\varphi$,由策略 π 生成的马尔可夫链 $\mathrm{MC}_{M^\times}^{\pi}$ 的任何一个常返类满足以下条件之一:

1) $\delta_\pi^j \bigcap \bar{F}_i^\times \neq \varnothing, \forall i \in \{0,1,\dots,k-1\}$;

2) $\delta_\pi^j \bigcap \bar{F}_i^\times = \varnothing, \forall i \in \{0,1,\dots,k-1\}$.

证明　记 I 为集合 $2^{\{0,1,\dots,k-1\}} \backslash \{\{0,1,\dots,k-1\}, \varnothing\}$ 的一个子集. 假设存在策略 π,其生成的马尔可夫链既不满足条件1)也不满足条件2),即 $\delta_\pi^j \bigcap \bar{F}_i^\times \neq \varnothing, \forall i \in I$. 据文献[28], $\forall (s^\times, a^\times, (s^\times)') \in \delta_\pi^j \bigcap \bar{F}_i^\times$,有 $\sum_{k=0}^{\infty} p^k(s^\times, s^\times) = \infty$,其中 $p^k(s^\times, s^\times)$ 表示从一个常返状态出发,经 k 步返回自身的概率,此等式说明常返状态 $s^\times$ 能被无限次访问. 然而,根据定义7和定义8,接受边界集 T 在全部的接受集被访问之前不会被重置,故当常返类的转移集合仅与部分接受集相交时, T 不会被重置,导致ctLDGBA的状态的 v 部分将永远停留在一个值,这样乘积状态 $s^\times$ 就不可能再被访问到,这与常返状态的性质 $\sum_{k=0}^{\infty} p^k(s^\times, s^\times) = \infty$ 矛盾. □

引理1说明对于任意策略,乘积MDP的全部接受集或位于瞬时类,或位于某个能访问全部接受集的常返类中. 受文献[26]启发,在乘积MDP上定义如下用于强化学习的回报函数和折扣函数:

$$R_{\mathcal{F}^\times}(s^\times) := \begin{cases} 1-\gamma_{\mathcal{F}^\times}, & (s^\times, a^\times, (s^\times)') \in \mathcal{F}^\times; \\ 0, & \text{otherwise}; \end{cases}$$

$$\Gamma_{\mathcal{F}^\times}(s^\times) := \begin{cases} \gamma_{\mathcal{F}^\times}, & (s^\times, a^\times, (s^\times)') \in \mathcal{F}^\times; \\ \gamma, & \text{otherwise}. \end{cases} \tag{4}$$

式(4)中 $\gamma_{\mathcal{F}^\times} = \gamma_{\mathcal{F}^\times}(\gamma)$,满足

$$\lim_{\gamma \to 1^-} \frac{1-\gamma}{1-\gamma_{\mathcal{F}^\times}(\gamma)} = 0. \tag{5}$$

给定乘积MDP的一条路径 $\sigma = s_0^\times s_1^\times \dots$,该路径的累积折扣回报(后文简称回报)为

$$G_t(\sigma) = \sum_{i=0}^{\infty} R_{\mathcal{F}^\times}(\sigma[t+i]) \cdot \prod_{j=0}^{i-1} \Gamma_{\mathcal{F}^\times}(\sigma[t+j]), \tag{6}$$

其中 $\prod_{j=0}^{i-1} := 1$. 根据式(6)可定义路径的期望累积折扣回报(后文简称期望回报)为

$$V^\pi(s_t^\times) = \boldsymbol{E}^\pi[G_t(\sigma) | \sigma(t) = s_t^\times]. \tag{7}$$

问题1中要求的最优策略是最大化LTL任务满足概率的策略,然而强化学习只是尽可能地获得更高的回报,不能保证学到的策略符合需求,即存在"回报黑客(reward hacking)"问题,这是由回报函数定义不合理引起的,下文将严格证明按照式(4)定义回报和折扣函数,能够解决回报黑客问题,这对于解决问题1至关重要. 根据式(4)定义的回报函数和折扣函数,有以下引理成立,由于该引理的证明与文献[26]中的引理2证明类似,此处省略.

引理 2　对于乘积MDP的全部路径 σ 和式(6)定义的回报 $G_t(\sigma)$,有

$$0 \leqslant \gamma G_{t+1}(\sigma) \leqslant G_t(\sigma) \leqslant 1 - \gamma_{\mathcal{F}^\times} + \gamma_{\mathcal{F}^\times} G_{t+1}(\sigma) \leqslant 1.$$

引理2给出了一条路径的回报边界,可用于估计乘积状态的期望回报边界,基于此有如下引理.

引理 3　记策略 π 产生的AMEC为 $\mathrm{AMEC}_{M^\times}^{\pi} = (S_{\mathrm{AMEC}}^{\pi}, A_{\mathrm{AMEC}}^{\pi})$. 其中: S_{AMEC}^{π} 是AMEC的状态集, A_{AMEC}^{π} 是受限动作集. 记 $\delta_{M^\times}^{\pi} = \{(s^\times, a^\times, (s^\times)') \in \delta^\times | (s^\times, a^\times) \in \mathrm{AMEC}_{M^\times}^{\pi}; (s^\times, a^\times, (s^\times)') \in \bar{F}_j^\times, j \in \{0,1,\dots,k-1\}\}$ 为同时位于AMEC和 $M^\times$ 的接受集的全部转移的集合,则 $\forall (s^\times, a^\times, (s^\times)') \in \delta_{M^\times}^{\pi}$,有 $\lim_{\gamma \to 1^-} V^\pi(s^\times) = 1$.

基于引理3,可以得到如下定理.

定理 2　给定乘积MDP $M^\times = M \times \bar{\mathcal{A}}_\varphi$,对于任意乘积状态 $s^\times = (s, (v, q)) \in S^\times$,策略 π 下的期

望回报$V^{\pi}(s^{\times})$满足$\min\limits_{\gamma\to 1^-}V^{\pi}(s^{\times})=\Pr^{\pi}(\Diamond\bar{F}_v^{\times})$,其中$\Pr^{\pi}(\Diamond\bar{F}_v^{\times})$表示从$s^{\times}$出发的路径最终访问接受集$\bar{F}_v^{\times}$的概率.

定理2表明,当折扣函数中的γ趋于1时,状态$s^{\times}=(s,(v,q))$的期望回报等于从该状态出发的路径访问当前需要访问的接受集$\bar{F}_v^{\times}$的概率. 根据引理1、定理2和假设1,可以得到以下定理,该定理证明了通过强化学习方法得到策略的最优性.

定理3 给定乘积MDP $M^{\times}=M\times\bar{\mathcal{A}}_{\varphi}$,存在一个折扣因子$\underline{\gamma}$,使得对于优化期望回报的任意优化算法,若该算法有$\gamma>\underline{\gamma}$和$\gamma_{\mathcal{F}^{\times}}>\underline{\gamma}$,则通过该算法能在$M^{\times}$上得到无记忆策略$\pi^*$,策略$\pi^*$能生成满足$M^{\times}$接受条件的路径$\sigma_{\pi^*}$. 令$\gamma\to 1^-$,得到的最优策略$\pi^*$也是最大化LTL公式满足概率的策略,即$\Pr^{\pi^*}(s^{\times}\vDash\varphi)=\Pr_{max}(s^{\times}\vDash\varphi)$.

证明 设策略π^*是最大化期望回报却不满足乘积MDP接受条件的最优策略. 根据假设1,至少存在一个满足$M^{\times}$接受条件的策略. 接下来将证明对于任何一个满足$M^{\times}$接受条件的策略π,其期望回报均优于策略π^*的期望回报,这样就与π^*是最优策略矛盾,从而证明策略π^*能生成满足$M^{\times}$接受条件的路径σ_{π^*}. 任意策略π'生成的马尔可夫链$\mathrm{MC}_{M^{\times}}^{\pi'}$的状态集可划分为$S_{\pi'}^{\times}=\mathcal{T}_{\pi'}\bigcup R_{\pi'}^1\bigcup\ldots\bigcup R_{\pi'}^n$. 对于策略$\pi^*$,根据引理1有:$\delta_{\pi^*}^j\bigcap\bar{F}_i^{\times}=\varnothing,\forall\delta_{\pi^*}^j,\forall i\in\{0,1,\ldots,k-1\}$. 现在考虑一个乘积状态$s_R^{\times}\in R_{\pi^*}^k$,根据常返类$R_{\pi^*}^k$的定义,从$s_R^{\times}$开始的转移只会发生在$R_{\pi^*}^k$中并且不会访问任何接受集,故$s_R^{\times}$在策略$\pi^*$下的期望回报为

$$V^{\pi^*}(s_R^{\times})=\sum_{n=0}^{\infty}\gamma^n\cdot\sum_{s^{\times}\in R_{\pi^*}^k}P_{\pi^*}^n(s_R^{\times},s^{\times})R_{\mathcal{F}^{\times}}(s^{\times})=0,$$

其中$P_{\pi^*}^n(s_R^{\times},s^{\times})$是在策略$\pi^*$下从$s_R^{\times}$出发经$n$步到达$s^{\times}$的概率. 对于任何一个满足$M^{\times}$接受条件的策略$\pi$,必然存在常返类$R_{\pi}^l$使得$\delta_{\pi}^l\bigcap\bar{F}_i^{\times}\neq\varnothing,\forall i\in\{0,1,\ldots,k-1\}$. 记$\underline{\gamma}$为折扣函数$\Gamma_{\mathcal{F}^{\times}}$的下界.

1) $s_R^{\times}\in R_{\pi}^l$. 根据常返类$R_{\pi}^l$的性质,存在$(s_R^{\times})'$,$(s_R^{\times})''\in R_{\pi}^l$使得$((s_R^{\times})',a^{\times},(s_R^{\times})'')\in\delta_{\pi}^l\bigcap\bar{F}_i^{\times}$,并且从$s_R^{\times}$到达$(s_R^{\times})'$的概率非0,由此可得

$$V^{\pi}(s_R^{\times})=\sum_{n=0}^{\infty}\sum_{s^{\times}\in R_{\pi}^l}P_{\pi}^n(s_R^{\times},s^{\times})R_{\mathcal{F}^{\times}}(s^{\times})\times\prod_{(s^{\times})'\in\mathrm{path}^n(s_R^{\times},s^{\times})}\Gamma_{\mathcal{F}^{\times}}((s^{\times})')\geqslant\sum_{n=0}^{\infty}\underline{\gamma}^nP_{\pi}^n(s_R^{\times},(s_R^{\times})')(1-\gamma_{\mathcal{F}^{\times}})>0.$$

其中:$\mathrm{path}^n(s_R^{\times},s^{\times})$表示从$s_R^{\times}$经$n$步到达$s^{\times}$的路径,当$n=0$时,$\prod\limits_{(s^{\times})'\in\mathrm{path}^n(s_R^{\times},s^{\times})}\Gamma_{\mathcal{F}^{\times}}((s^{\times})'):=1$. 因此,有$V^{\pi}(s_R^{\times})>V^{\pi^*}(s_R^{\times})$.

2) $s_R^{\times}\in\mathcal{T}_{\pi}$. 由于策略$\pi$满足乘积MDP的接受条件,存在$\bar{k}\geqslant 1$使得从$s_R^{\times}$出发的路径经$\bar{k}$步能进入$R_{\pi}^l$,设到达的状态为$(s_R^{\times})'\in R_{\pi}^l$. 此外,对于任意常返态$s^{\times}\in R_{\pi}^j$,总有$\sum\limits_{n=0}^{\infty}\gamma^np^n(s^{\times},s^{\times})>\dfrac{\bar{p}}{1-\gamma^{\bar{n}}}$,其中存在$\bar{n}$使得$p^n(s^{\times},s^{\times})>0$并且其下界为$\bar{p}$[28]. 忽略从$s_R^{\times}$出发经$\bar{k}$步得到的路径的回报,可以得到

$$V^{\pi}(s_R^{\times})\geqslant\prod_{i=1}^{\bar{k}}\Gamma_{\mathcal{F}^{\times}}(s_i^{\times})\cdot P_{\pi}^{\bar{k}}(s_R^{\times},(s_R^{\times})')\times\sum_{n=0}^{\infty}\sum_{s^{\times}\in R_{\pi}^l}\prod_{j=0}^{n-1}\Gamma_{\mathcal{F}^{\times}}(s_j^{\times})\cdot P_{\pi}^n((s_R^{\times})',s^{\times})R_{\mathcal{F}^{\times}}(s^{\times})>\underline{\gamma}^{\bar{k}}P_{\pi}^{\bar{k}}(s_R^{\times},(s_R^{\times})')\cdot\sum_{n=0}^{\infty}\underline{\gamma}^nP_{\pi}^n((s_R^{\times})',(s_R^{\times})')(1-\gamma_{\mathcal{F}^{\times}})>\underline{\gamma}^{\bar{k}}P_{\pi}^{\bar{k}}(s_R^{\times},(s_R^{\times})')\cdot\frac{\bar{p}}{1-\underline{\gamma}^{\bar{n}}}(1-\gamma_{\mathcal{F}^{\times}}).$$

因此,存在$\underline{\gamma}<1$,使得当$\gamma>\underline{\gamma}$和$\gamma_{\mathcal{F}^{\times}}>\underline{\gamma}$时,有$V^{\pi}(s_R^{\times})>\underline{\gamma}^{\bar{k}}P_{\pi}^{\bar{k}}(s_R^{\times},(s_R^{\times})')\cdot\dfrac{\bar{p}}{1-\underline{\gamma}^{\bar{n}}}(1-\gamma_{\mathcal{F}^{\times}})\geqslant 0$.

情况1)和情况2)的结果都证明了$V^{\pi}(s_R^{\times})>V^{\pi^*}(s_R^{\times})$,这与策略$\pi^*$是关于期望回报最优的策略矛盾. 因此,令$\gamma\to 1^-$,就有$\gamma>\underline{\gamma}$和$\gamma_{\mathcal{F}^{\times}}>\underline{\gamma}$,这表明存在满足$M^{\times}$接受条件的最优策略$\pi^*$;又根据定理2,对于策略$\pi^*$有$\lim\limits_{\gamma\to 1^-}V^{\pi^*}(s^{\times})=\Pr^{\pi^*}(\Diamond\bar{F}_v^{\times})$,这表明最大化期望回报就是最大化访问AMEC中的接受集的概率. 因此,可以得出结论:$\Pr^{\pi^*}(s^{\times}\vDash\varphi)=\Pr_{\max}(s^{\times}\vDash\varphi)$. □

3.3 基于安全盾的无模型强化学习

本节提出一种安全强化学习算法来最优化期望回报,用于解决问题1,该算法基于安全盾[23]和Q-learning[29].

Q-learning是一种常用的无模型强化学习算法,动作值Q由以下公式更新:

$$Q(s^{\times},a^{\times})\leftarrow(1-\alpha)Q(s^{\times},a^{\times})+\alpha[R_{\mathcal{F}^{\times}}(s^{\times})+\Gamma_{\mathcal{F}^{\times}}(s^{\times})\cdot\max_{(a^{\times})'}Q((s^{\times})',(a^{\times})')],\quad(8)$$

其中$0<\alpha\leqslant 1$是学习率. Q-learning无法保证智能体在学习过程中的安全性.

安全盾的构建是基于安全规范和环境的抽象,安全规范是用LTL描述的系统需要遵守的安全约束;抽

象是传感器感知到的环境信息的模型,仅能用于确认安全规范是否被违反.

给定MDP $M=(S,A,P,s_0,\mathrm{AP},L)$和观测函数 $f:S\to O$,观测函数是状态空间S到观测集O的映射,抽象定义为$\mathcal{A}^e=(Q^e,q_0^e,\Sigma^e,\delta^e)$. 其中: $\Sigma^e=O\times A$是输入字母表,$\delta^e:Q^e\times\Sigma^e\to Q^e$是转移函数. 用LTL描述系统的安全规范,安全规范可转化为确定性有限自动机(deterministic finite automaton, DFA)[7], DFA为$\mathcal{A}^s=(Q^s,q_0^s,\Sigma^s,\delta^s,F^s)$. 其中:输入字母表同为$\Sigma^s=O\times A$, $F^s\subseteq Q^s$是安全状态集. 给定抽象$\mathcal{A}^e$和安全自动机$\mathcal{A}^s$,二玩家安全博弈可构建为$\mathcal{G}=(G,g_0,\Sigma_1,\Sigma_2,\delta_g,F_g)$. 其中: $G=Q^e\times Q^s$是博弈状态集; $g_0=(q_0^e,q_0^s)$是初始状态; $\Sigma_1=O$和$\Sigma_2=A$分别是环境和智能体的输入字母表; $\delta_g(g,o,a)=(\delta^e(q^e,o\times a),\delta^s(q^s,o\times a)),\forall g=(q^e,q^s)\in G,o\in\Sigma_1,a\in\Sigma_2$,是转移函数; $F_g=Q^e\times F^s$是安全状态集. 安全博弈的获胜区域$W\subseteq F_g$可用标准方法求得[23]. 文献[23]中通过安全博弈$\mathcal{G}$和获胜区域W合成的安全盾不能直接与本文定义的乘积MDP结合,必须修改为能与本文定义的乘积MDP结合的安全盾. 通过安全博弈$\mathcal{G}$、获胜区域W和动作集$A^\times=A\bigcup A^\epsilon$合成安全盾$\mathcal{S}=(Q,q_0,\Sigma_I,\Sigma_O,\delta,\lambda)$. 其中: $Q=G$; $q_0=g_0$; $\Sigma_I=O\times A^\times$; $\Sigma_O=A^\times$; $\delta(g,o,a)=\delta_g(g,o,\lambda(g,o,a)),\forall g\in G,o\in O,a\in A$; $\delta(g,o,\epsilon)=g,\forall g\in G,o\in O,\epsilon\in A^\epsilon$. 输出函数定义为

$$\lambda(g,o,a)=\begin{cases}a,\ a\in A\bigwedge\delta_g(g,o,a)\in W;\\ a,\ a\in A^\epsilon;\\ \bar{a},\ \delta_g(g,o,a)\notin W\bigwedge\delta_g(g,o,\bar{a})\in W.\end{cases}$$

安全盾是LTL引导的强化学习算法的一个独立组件,在学习过程中,智能体根据当前策略选择动作$a_t^\times$,然后将$a_t^\times$输入安全盾检验该动作是否安全. 若该动作不安全,则安全盾输出修改后的安全动作$\bar{a}_t^\times$;否则直接输出$\bar{a}_t^\times=a_t^\times$. 智能体在执行安全动作获得立即回报后,更新状态$(s_t^\times,\bar{a}_t^\times)$的$Q$值,这是因为安全动作$\bar{a}_t^\times$才是智能体实际执行的动作.

加上安全盾后,智能体的可行动作空间$A(s)$缩小为$A'(s)$,这导致假设1可能不成立而使问题1无解,若在$A'(s)$上存在满足LTL公式的策略,则该策略称为安全策略,由此可增强假设1使问题1有解.

假设2 设乘积MDP $M^\times$上至少存在一个满足LTL公式的安全策略.

基于假设2,加上安全盾后定理3仍然成立,这个结论的证明与定理3的证明极其类似,均通过反证法证明,此略. 下面给出本文提出的整个算法.

算法1 LTL引导的安全强化学习算法.

输入: $(M,\bar{\mathcal{A}}_\varphi)$;

输出: 最优策略π^*.

1) 初始化episode $=0$, num_episode(回合总数)和num_step(步数)

2) 构建乘积MDP $M^\times=M\times\bar{\mathcal{A}}_\varphi$

3) 根据$\mathcal{A}^e$、$\mathcal{A}^s$和$A^\times$构建安全盾$\mathcal{S}$

4) 置$\gamma_{\mathcal{F}^\times}=0.99$和$\gamma=0.999$以确定$R_{\mathcal{F}^\times}$和$\Gamma_{\mathcal{F}^\times}$

5) 初始化动作值$Q(s^\times,a)$和接受边界集$T=\mathcal{F}$

6) while episode $<$ num_episode do

7) $\alpha=\max(1-1.5\mathrm{episode}/\mathrm{num\_episode},0.001)$

8) $\epsilon=\max(1-1.5\mathrm{episode}/\mathrm{num\_episode},0.01)$

9) while $t<$ num_step do

10) $s_t^\times=(s_t,(v,q))$

11) 获得观测$\mathrm{sensor}_t=f(s_t)$

12) 根据ϵ贪婪策略选择动作$a_t^\times$

13) 获得安全动作$\bar{a}_t^\times\leftarrow\mathcal{S}(\mathrm{sensor}_t,a_t^\times)$

14) 执行$\bar{a}_t^\times$,按式(3)更新接受边界集T,并观测$s_{t+1}^\times$、$R_{\mathcal{F}^\times}(s_t^\times)$、$\gamma_{\mathcal{F}^\times}(s_t^\times)$

15) 置$r\leftarrow R_{\mathcal{F}^\times}(s_t^\times)$和$\Gamma\leftarrow\gamma_{\mathcal{F}^\times}(s_t^\times)$

16) $Q(s_t^\times,\bar{a}_t^\times)\leftarrow(1-\alpha)Q(s_t^\times,\bar{a}_t^\times)+\alpha[r+\Gamma\cdot\max\limits_{a^\times}Q(s_{t+1}^\times,a^\times)]$

17) $s_t^\times=s_{t+1}^\times$

18) $t++$

19) end while

20) episode $++$

21) end while

22) for all $s^\times\in S^\times$

23) $\pi^*(s^\times)=\arg\max\limits_{a^\times\in A^\times}Q(s^\times,a^\times)$

24) end for

4 仿真实验结果

本节通过仿真实验验证本文所提出算法的可行性,算法1由Python实现. 根据具体的任务定义相应的原子命题集,再根据式(1)给出的LTL语法递归地构建LTL公式,然后将LTL公式通过Rabinizer4[30]转化为tLDGBA,再根据转化结果构建ctLDGBA. 强化学习的参数设置已在算法1中给出. 本文采用LTL文献中常用的网格世界作为仿真实验环境[7,11,15,22,26],

移动机器人可采取“上,下,左,右”4个动作,安全盾以“保持不动”为默认安全动作,当执行某个动作时,移动机器人以概率p向目标方向移动,滑向侧边方向的概率为$1-p$.

4.1 最大化LTL公式满足概率的验证

实验1 验证所提出算法求出的最优策略是最大化LTL公式满足概率的策略.

移动机器人在5×4的网格世界中运动,以概率$p=0.9$向目标方向移动,滑向侧边方向的概率为0.1. 任务设置为:机器人最终到达目标a或目标b,同时避免进入危险区域w. 对应的LTL公式为

$$\varphi_1=(\Diamond\Box a\bigvee\Diamond\Box b)\bigwedge\Box\neg w. \tag{9}$$

本仿真实验中,训练回合设置为100 000个,每个回合最多100步,初始状态为随机状态. 图4显示了每个状态的期望回报,期望回报是从该状态出发满足LTL公式φ_1的概率的估计,正如预期,从标签为a或b的状态出发,满足LTL公式的最大概率为1. 状态$(4,2)$的最大任务满足概率为0.9,这是由移动机器人的运动不确定性引起的. 图5显示了得到的最优策略,从状态$(0,0)$出发到达目标的最短路径是通过区域$(1,0)$和$(2,0)$到达$(3,0)$,然后智能体采取ϵ动作就能访问接受集. 图6显示了学习过程中累积回报的变化曲线,由于初始状态随机,若初始状态是不安全状态,则累积回报必然为0,因此取1 000个回合的平均累积回报来绘制曲线,图6的曲线说明了本文所提出算法的收敛性.

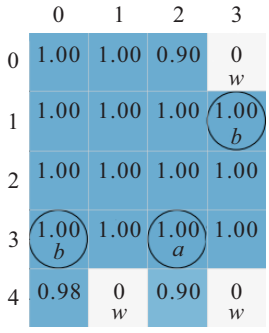


图4 满足LTL公式φ_1的最大概率

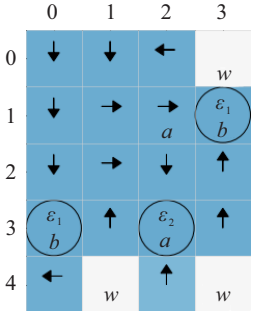


图5 学到的最优策略

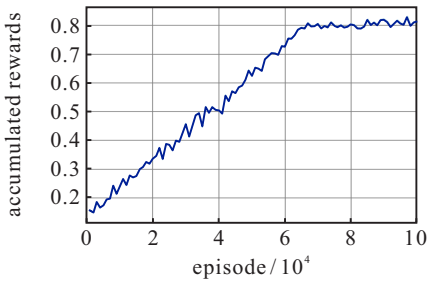


图6 仿真实验1学习曲线

4.2 学习过程的安全性验证

实验2 验证本文所提出的算法是否能保证强化学习过程的安全性. 实验环境如图7所示,并考虑一个更为复杂的LTL任务,这也验证了算法1可用于复杂高层任务规划.

LTL公式为

$$\varphi_2=(\Box\Diamond b_1\bigwedge\Box\Diamond b_2\bigwedge\Box\Diamond b_3)\bigwedge \Box(s\rightarrow\bigcirc((\neg s)\bigcup(b_1\bigvee b_2\bigvee b_3)))\bigwedge\Box\neg\text{obs}. \tag{10}$$

其中:b_1、b_2和b_3是不同的基站,s是补给区,obs是障碍物. LTL公式φ_2要求机器人无限次地访问3个不同基站,同时避免撞到图7中黑色方块表示的障碍物,若机器人到达了补给区,则接下来不要去补给区,直到访问了任一基站. φ_2转化得到的tLDGBA有4个状态和4个接受集,对应的ctLDGBA有16个状态和4个接受集. 本实验设置300 000个训练回合,每个回合最大300步,当机器人撞到障碍物时该回合终止,初始状态设为$(0,0)$.

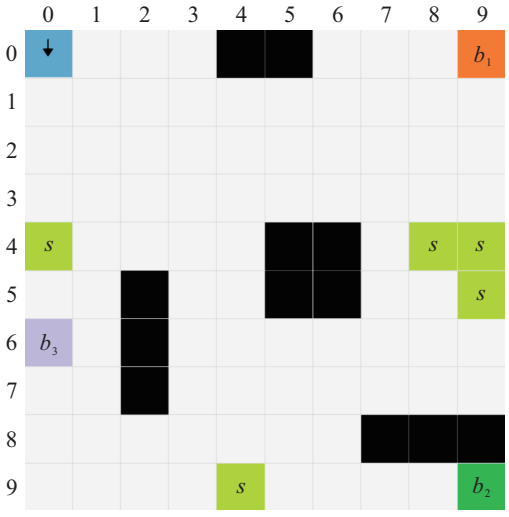


图7 仿真实验2环境

对比3种方法的结果来验证所提出的算法1的安全性和有效性. 如图8所示,红色点划线是LDBA方法[26]的结果,可以看到整个学习过程毫无改进,这是因为LDBA仅有一个接受集,导致强化学习具有稀

疏回报结构;此外,环境中设置了很多障碍物,智能体一旦撞到障碍物该回合立即终止,并且智能体在撞到障碍物时不会获得负回报,无法通过负回报指引智能体在后续的探索过程中排除危险动作,这两个原因使得智能体难以在强化学习过程中获得正回报来改进策略,最终呈现出红色点划线所示的结果. 在本实验中,LDBA 方法不能获得满足 LTL 公式的策略,即使增加回合数和最大时间步也没有作用. 绿色虚线是不加安全盾的 ctLDGBA 方法的结果,得益于 ctLDGBA 的多接受集,智能体每访问一个接受集代表完成一个子任务,就能获得一个正回报,解决了稀疏回报的问题. 绿色虚线最终趋于收敛,表示智能体最终找到了满足 LTL 公式的策略. 蓝色实线是算法 1 的结果,有了安全盾保证学习过程的安全性,智能体在训练初期就能访问接受集,因此蓝色实线迅速上升,最终收敛到比不加安全盾的 ctLDGBA 方法更高的水平.

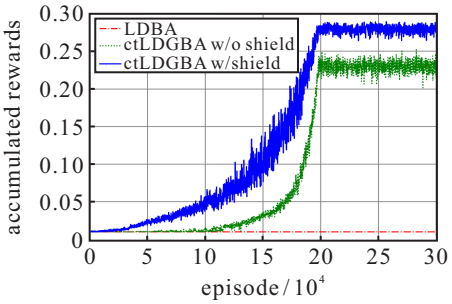


图 8　实验 2 不同方法的学习曲线

表 1 记录了强化学习过程中的障碍物碰撞次数,对比可知,不加安全盾的 ctLDGBA 方法虽然最终能找到满足 LTL 公式的策略,但是在整个学习过程中却发生了 158 250 次碰撞,大幅降低了学习速度并且导致机器人系统不安全. 使用加上了安全盾的算法 1,整个学习过程碰撞次数为 0,这使得智能体在每个回合都能充分地探索状态空间,加快了学习速度. 通过图 8 和表 1 的结果可以得出结论,本文提出的算法能够有效地保证机器人系统在强化学习过程中的安全性,并且加快训练速度.

表 1　300 000 个训练回合碰撞总次数

算法	碰撞次数
LDBA	231 826
无安全盾的 ctLDGBA	158 250
带安全盾的 ctLDGBA	0

5　结　论

本文提出了一种线性时序逻辑引导的无模型安全强化学习算法,并严格证明了该算法求出的最优策略同时是最大化满足 LTL 任务概率的策略. 运动规划仿真实验验证了本文算法的安全性和有效性. 未来的工作将会集中于概率安全盾合成和连续空间的控制策略求解.

参考文献 (References)

[1] Cai M Y, Xiao S P, Li Z J, et al. Optimal probabilistic motion planning with potential infeasible LTL constraints[J]. IEEE Transactions on Automatic Control, 2023, 68(1): 301-316.

[2] Sutton R S, Barto A G. Reinforcement learning: An introduction[J]. IEEE Transactions on Neural Networks, 1998, 9(5): 1054.

[3] 王思鹏, 杜昌平, 郑耀. 基于强化学习的扑翼飞行器路径规划算法[J]. 控制与决策, 2022, 37(4): 851-860.
(Wang S P, Du C P, Zheng Y. Local planner for flapping wing micro aerial vehicle based on deep reinforcement learning[J]. Control and Decision, 2022, 37(4): 851-860.)

[4] Junges S, Jansen N, Dehnert C, et al. Safety-constrained reinforcement learning for MDPs[C]. Tools and Algorithms for the Construction and Analysis of Systems. Berlin: Springer, 2016: 130-146.

[5] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540): 529-533.

[6] Cai M Y, Hasanbeig M, Xiao S P, et al. Modular deep reinforcement learning for continuous motion planning with temporal logic[J]. IEEE Robotics and Automation Letters, 2021, 6(4): 7973-7980.

[7] Baier C, Katoen J P. Principles of model checking[M]. Cambridge: MIT Press, 2008: 1-975.

[8] 肖云涛, 欧林林, 俞立. 基于线性时序逻辑的最优巡回路径规划[J]. 自动化学报, 2014, 40(10): 2126-2133.
(Xiao Y T, Ou L L, Yu L. Optimal patrolling path planning via linear temporal logic[J]. Acta Automatica Sinica, 2014, 40(10): 2126-2133.)

[9] Belta C, Yordanov B, Aydin Gol E. Discrete-time dynamical systems[M]. Cham: Springer International Publishing, 2017: 111-118.

[10] Lacerda B, Faruq F, Parker D, et al. Probabilistic planning with formal performance guarantees for mobile service robots[J]. The International Journal of Robotics Research, 2019, 38(9): 1098-1123.

[11] Cai M, Peng H, Li Z J, et al. Receding horizon control-based motion planning with partially infeasible LTL constraints[J]. IEEE Control Systems Letters, 2020, 5(4): 1279-1284.

[12] Brázdil T, Chatterjee K, Chmelík M, et al. Verification of Markov decision processes using learning algorithms[C]. Automated Technology for Verification and Analysis.

Cham: Springer, 2014: 98-114.

[13] Fu J, Topcu U. Probably approximately correct MDP learning and control with temporal logic constraints[J/OL]. 2014, arXiv: 1404.7073.

[14] Sadigh D, Kim E S, Coogan S, et al. A learning based approach to control synthesis of Markov decision processes for linear temporal logic specifications[C]. The 53rd IEEE Conference on Decision and Control. Los Angeles, 2014: 1091-1096.

[15] Cai M Y, Peng H, Li Z J, et al. Learning-based probabilistic LTL motion planning with environment and motion uncertainties[J]. IEEE Transactions on Automatic Control, 2021, 66(5): 2386-2392.

[16] Hahn E M, Perez M, Schewe S, et al. Omega-regular objectives in model-free reinforcement learning[C]. Tools and Algorithms for the Construction and Analysis of Systems. Cham: Springer, 2019: 395-412.

[17] Garca J, Fernández F. A comprehensive survey on safe reinforcement learning[J]. Journal of Machine Learning Research, 2015, 16(1): 1437-1480.

[18] 朱斐, 吴文, 伏玉琛, 等. 基于双深度网络的安全深度强化学习方法[J]. 计算机学报, 2019, 42(8): 1812-1826.
(Zhu F, Wu W, Fu Y C, et al. A dual deep network based secure deep reinforcement learning method[J]. Chinese Journal of Computers, 2019, 42(8): 1812-1826.)

[19] Bacci E, Parker D. Probabilistic guarantees for safe deep reinforcement learning[C]. Formal Modeling and Analysis of Timed Systems. Cham: Springer, 2020: 231-248.

[20] Berkenkamp F, Turchetta M, Schoellig A P, et al. Safe model-based reinforcement learning with stability guarantees[C]. Proceedings of the 31st International Conference on Neural Information Processing Systems. Red Hook, 2017: 908-919.

[21] Li X, Serlin Z, Yang G, et al. A formal methods approach to interpretable reinforcement learning for robotic planning[J]. Science Robotics, 2019, 4(37): 6276.

[22] Hasanbeig M, Abate A, Kroening D. Cautious Reinforcement Learning with Logical Constraints[C]. Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems. Richland, 2020: 483-491.

[23] Alshiekh M, Bloem R, Ehlers R, et al. Safe reinforcement learning via shielding[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2018, 32(1): 2669-2678.

[24] Sickert S, Esparza J, Jaax S, et al. Limit-deterministic Büchi automata for linear temporal logic[C]. Computer Aided Verification. Cham: Springer, 2016: 312-332.

[25] Hasanbeig M, Kantaros Y, Abate A, et al. Reinforcement learning for temporal logic control synthesis with probabilistic satisfaction guarantees[C]. IEEE 58th Conference on Decision and Control. Nice, 2019: 5338-5343.

[26] Bozkurt A K, Wang Y, Zavlanos M M, et al. Control synthesis from linear temporal logic specifications using model-free reinforcement learning[C]. IEEE International Conference on Robotics and Automation. Paris, 2020: 10349-10355.

[27] Cai M Y, Xiao S P, Li B L, et al. Reinforcement learning based temporal logic control with maximum probabilistic satisfaction[C]. IEEE International Conference on Robotics and Automation. Xi'an, 2021: 806-812.

[28] Durrett R. Essentials of Stochastic Processes[M]. New York: Springer, 2012.

[29] Watkins C J C H, Dayan P. Q-learning[J]. Machine learning, 1992, 8(3/4): 279-292.

[30] Křetínský J, Meggendorfer T, Sickert S, et al. Rabinizer 4: From LTL to your favourite deterministic automaton[C]. Computer Aided Verification. Cham: Springer, 2018: 567-577.

作者简介

李保罗 (1995−), 男, 硕士生, 从事机器人运动规划、线性时序逻辑等研究, E-mail: libaoluo@mail.ustc.edu.cn;

蔡明钰 (1992−), 男, 博士, 从事机器人学、自动驾驶等研究, E-mail: mingyu-cai@lehigh.edu;

阚震 (1983−), 男, 教授, 博士生导师, 从事非线性控制理论、智能机器人控制等研究, E-mail: zkan@ustc.edu.cn.