DOI:10.11992/tis.201604008
网络出版地址:http://kns.cnki.net/kcms/detail/23.1538.TP.20170301.1147.002.html

基于事件驱动的多智能体强化学习研究

张文旭,马磊,王晓东
(西南交通大学 电气工程学院,四川 成都 610031)

摘 要:本文针对多智能体强化学习中存在的通信和计算资源消耗大等问题,提出了一种基于事件驱动的多智能体强化学习算法,侧重于事件驱动在多智能体学习策略层方面的研究。在智能体与环境的交互过程中,算法基于事件驱动的思想,根据智能体观测信息的变化率设计触发函数,使学习过程中的通信和学习时机无需实时或按周期地进行,故在相同时间内可以降低数据传输和计算次数。另外,分析了该算法的计算资源消耗,以及对算法收敛性进行了论证。最后,仿真实验说明了该算法可以在学习过程中减少一定的通信次数和策略遍历次数,进而缓解了通信和计算资源消耗。

关键词:事件驱动;多智能体;强化学习;分布式马尔科夫决策过程;收敛性
中图分类号: TP181 **文献标志码**:A **文章编号**:1673-4785(2017)01-0082-06

中文引用格式:张文旭,马磊,王晓东. 基于事件驱动的多智能体强化学习研究[J]. 智能系统学报, 2017, 12(1): 82-87.
英文引用格式:**ZHANG Wenxu, MA Lei, WANG Xiaodong. Reinforcement learning for event-triggered multi-agent systems**[J]. **CAAI transactions on intelligent systems**, 2017, 12(1): 82-87.

Reinforcement learning for event-triggered multi-agent systems

ZHANG Wenxu, MA Lei, WANG Xiaodong
(School of Electrical Engineering, Southwest Jiaotong University, Chengdu 610031, China)

Abstract: Focusing on the existing multi-agent reinforcement learning problems such as huge consumption of communication and calculation, a novel event-triggered multi-agent reinforcement learning algorithm was presented. The algorithm focused on an event-triggered idea at the strategic level of multi-agent learning. In particular, during the interactive process between agents and the learning environment, the communication and learning were triggered through the change rate of observation. Using an appropriate event-triggered design, the discontinuous threshold was employed, and thus real-time or periodical communication and learning can be avoided, and the number of communications and calculations were reduced within the same time. Moreover, the consumption of computing resource and the convergence of the proposed algorithm were analyzed and proven. Finally, the simulation results show that the number of communications and traversals were reduced in learning, thus saving the computing and communication resources.

Keywords: event-triggered; multi-agent; reinforcement learning; decentralized Markov decision processes; convergence

近年来,基于事件驱动的方法在多智能体研究中得到广泛关注[1-3]。在事件驱动的思想中,智能体可以根据测量误差间歇性的更新状态,减少通信次数和计算量。文献[4]首次在多智能体系统的协作中运用事件驱动的策略,并设计了基于事件驱动机制的状态反馈控制器。随后,文献[5-7]将基于事件驱动的控制器扩展到非线性系统,以及复杂网

收稿日期:2016-04-05. 网络出版日期:2017-03-01.
基金项目:国家自然科学基金青年项目(61304166).
通信作者:张文旭.Email: wenxu_zhang@163.com.

络等领域。但是,目前事件驱动与强化学习的结合还相对不足[8-9],并主要集中在对多智能体的控制器设计上,较少有学者关注其在学习策略层的应用。在现有的多智能体强化学习算法中,由于智能体携带的通信设备和微处理器性能有限,其学习过程中通常存在两个问题:1)智能体间的信息交互需占用较大的通信带宽;2)在学习的试错和迭代过程中,消耗了大量的计算资源。以上问题都将减少智能体的工作时间,或增加设计上的复杂性。本文区别于传统的多智能体学习算法,侧重于事件驱动在多智能体学习策略层的研究,首先从自触发和联合触发两个方面定义触发函数,然后在分布式马尔可夫模型中设计了基于事件驱动的多智能体强化学习算法,最后对算法的收敛性进行了论证。

1　问题描述

1.1　分布式马尔可夫模型

考虑一个分布式马尔可夫模型(decentralized markov decision processes, DEC-MDPs),是由一个五元组$\langle I,\{S\},\{A_i\},P,R\rangle$构成的,其中,$I$ 表示有限的智能体集合;$\{S\}$表示一个有限的系统状态集合;$\{A_i\}$表示智能体 i 可采取的动作的集合;P 表示系统的转移;R 表示回报函数。DEC-MDPs 与多智能体的马尔可夫模型(multi agent-MDPs, M-MDPs)的唯一区别在于,在 M-MDPs 中系统的全局信息被所有智能体完全获得,而在 DEC-MDPs 中,每一个智能体仅具有局部的观测,或者说是全局信息的一个子集,当所有的子集放在一起求并集时,这些局部信息能够合成一个完整的全局信息,在完全通信的情况下,DEC-MDPs 可以被简化为 M-MDPs 模型。求解 DEC-MDPs 的目的是找到一个联合策略 $\vec{\pi}=(\pi_1,\pi_2,\cdots,\pi_n)$来最大化回报函数 R。求解 DEC-MDPs 问题的计算复杂度为 NEXP[10]难度,即问题的状态随着步数增加呈现双指数增长。

1.2　Q-学习

文献[11]提出了一类通过引入期望的延时回报,求解无完全信息的 MDPs 类问题的方法,称为 Q-学习(Q-learning)。Q-学习是一种模型无关的强化学习方法,通过对状态-动作对的值函数进行估计,以求得最优策略。Q-学习算法的基本形式如下:

$$Q^*(s,a)= R(s,a) +\gamma\sum_{s'\in S}P(s,a,s')\max Q^*(s',a')$$

式中:$Q^*(s,a)$表示智能体在状态 s 下采用动作 a 所获得的奖赏折扣总和;γ 为折扣因子;$P(s,a,s')$表示概率函数;最优策略为智能体在状态 s 下选用 Q 值最大的策略。Q-学习存在的最大问题为,智能体需要通过试错的方式找到最优策略,这样的方式使得 Q-学习需要考虑所有的可能策略,从而需要消耗大量计算资源。

2　触发规则设计

在事件驱动思想中,智能体把从环境中得到的观测误差作为重要的评判标准,当它超过一个预设的阈值时事件被触发,智能体更新状态并计算联合策略,而事件触发的关键在于对触发函数的设计。

2.1　自事件触发设计

DEC-MDPs 模型中,每一个智能体通过独立的观测获取局部信息,然后广播到全队,所以每一个智能体首先需要自触发设计。在时刻 t,当每一个智能体观测结束后,其根据上一刻观测与当前观测的变化率,进行一次自触发过程,智能体用自触发方式来判断是否需要广播自身的观测信息。智能体 i 从 $t-1$ 时刻到 t 时刻的观测变化率定义为

$$e_i(t)=|o_i(t)-o_i(t-1)|/o_i(t-1) \tag{1}$$

式中:$o_i(t)$为在 t 时刻的观测值。定义 $0<e<1$ 为自事件触发函数阈值,当智能体 i 观测信息的变化率 $e_i(t)$大于 e 时进行通信。此时,不一定所有的智能体都被驱动,没有采集到新观测信息的智能体仅接收信息。在自事件触发过程,智能体无需每一时刻进行通信,因此减少智能体的通信消耗。

2.2　联合事件触发设计

联合事件触发的对象是智能体团队,考虑的是一个联合观测的变化情况。假设在时刻 t 智能体团队获得当前的联合观测 $O(t)=(O_1(t),O_2(t),\cdots,O_n(t))$。此时,智能体团队从 $t-1$ 时刻到 t 时刻的联合观测变化率定义为

$$E(t)=(e_1(t),e_2(t),\cdots e_n(t)) \tag{2}$$

式中:$e_i(t)=|o_i(t)-o_i(t-1)|/o_i(t-1)$。

利用方差计算两个时刻的误差偏移程度,令联合观测变化率期望为 $F(t)=\sum_{i=1}^{n}e_i(t)/n$,方差为

$$D(t)=\sum_{i=1}^{n}(e_i(t)-F(t))^2\cdot p \tag{3}$$

式中:$p=1/n$ 为 $e_i(t)$的分布律,令

$$H(t)=|D(t)-F(t)|/F(t) \tag{4}$$

定义 $0<G<1$ 为团队的联合事件触发函数阈值,当 $H(t)$大于 G 时,认为智能体团队的状态已经发生较大改变,需要对 Q 值表进行遍历,并计算一个新的联合策略,否则智能体直接延用上一刻的联合策略。

自事件触发和联合事件触发的区别在于:

1)自事件触发的对象是单个智能体,对应的事件由智能体自身的观测变化率所触发,触发后的行动为进行广播式通信,自事件触发的目的是为了减少通信资源消耗;而联合事件触发针对的是智能体团队的联合观测变化率,触发后的行动是计算联合策略,目的在于减少计算资源消耗。

2)当单个智能体的观测发生变化时,并不一定导致团队的联合观测变化率发生较大改变。即当环境整体发生变化时,虽然每一个智能体的观测都发

生了变化,但对联合观测而言,所有智能体在两个时刻的变化率相对无变化,所以制定的联合策略可能无明显变化,此时也认为智能体团队不需要被触发。比如在机器人足球问题中,$t-1$ 时刻机器人团队的联合策略为,机器人 A 带球行动且其他队友跑位行动。到 t 时刻后,机器人 A 和其他机器人的观测(双方机器人的站位和距离)都发生了较大变化,机器人团队在通过广播通信获得全局观测信息后,根据观测信息进行判断,两个时刻双方机器人的相对站位和相对距离可能无大变化。此时,如果团队计算新的联合策略,也将是机器人 A 带球且其他队友跑位,与 $t-1$ 时刻的联合策略相同。所以,认为团队在 t 时刻无需计算新的联合策略,可以直接使用上一刻的策略。图 1 为事件触发流程图。

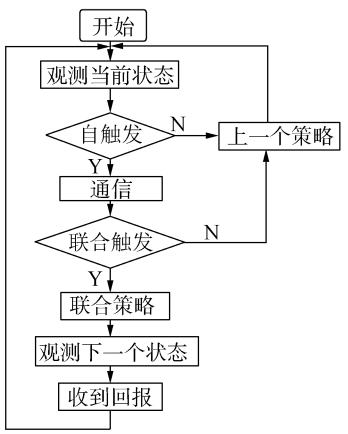


图 1 事件触发流程图

Fig.1 The flow chart of event-triggered

3 基于事件驱动的强化学习

本节介绍了基于事件驱动的强化学习算法,以及对事件驱动下计算资源消耗进行了分析,同时对算法的收敛性进行了论证。

3.1 基于事件驱动的强化学习设计

在完全通信情况下,DEC-MDPs 被简化为 M-MDPs 模型,所以直接考虑基于事件驱动的多智能体马尔可夫模型(event-triggered M-MDPs),其由一个六元组$\langle I,\{S\},\{A_i\},P,R,e\rangle$构成,其中 e 表示事件触发函数,当团队的触发函数大于阈值时,团队被触发并执行联合行动策略,同时发生状态转移,转移函数为 $P=\{s_{t+1}\mid s_t,a,e\}$。基于事件驱动的强化学习过程不同于经典的强化学习,如图 2 所示,智能体需要首先根据触发函数来判断事件是否被触发,如果被触发才执行一个联合行动并影响环境。

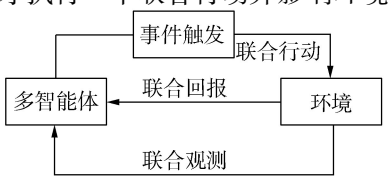


图 2 基于事件驱动的强化学习框架

Fig.2 The frame of reinforcement learning with event-triggered

强化学习的目的是找到一个策略使团队获得最大的奖励信号。如果在所有状态下,策略 π 的期望回报值都大于或等于策略 π'的,那么称之为最优策略。最优策略可能有多个,都将其称作最优策略,记 π^*,而最优策略对应的状态-联合动作对$(s,\vec{a})$也有相同的最优值函数,记作 Q^*。

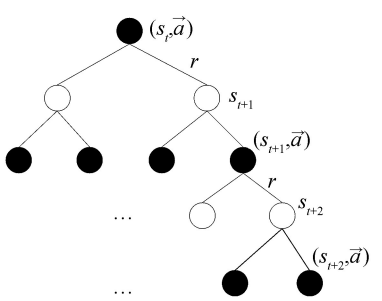
基于事件驱动的 Q-学习算法,类似于经典 Q-学习,均是不去估计环境模型,而是直接优化一个可迭代计算的 Q 函数。区别在于,经典 Q-学习中,智能体在每一个时刻都需要对 Q 值进行迭代计算,而基于事件驱动的 Q-学习,仅在智能体被触发的情况下,Q 值才进行迭代计算。此时,定义 Q 函数为在状态 s_t 时被触发并执行联合动作 $\vec{a}_t$,表达式为

$$Q_{t+1}(s_t,\vec{a}_t,e)=r_t\cdot\max_{a_t}\{Q_{t+1}(s_t,\vec{a}_t,e)\mid\vec{a}_t\in A\} \tag{5}$$

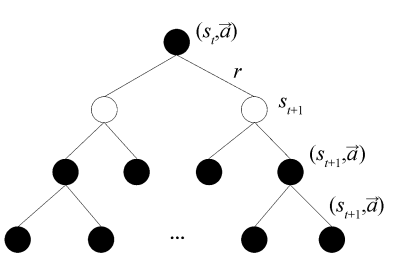
对于任意一个策略和下一个状态,在状态 s 的值和后继状态值之间存在如下关系:

$$Q^*=E\{r_{t+1}+\gamma Q_{t+1}(s_t,\vec{a},e)\mid s_t=s,\vec{a}_t=\vec{a}_t,e_t=e\}=\sum_{s'}P_{ss'}^{\vec{a}}(R_{ss'}^{\vec{a}}+\gamma\max_{\vec{a}}Q^*(s',\vec{a},e)) \tag{6}$$

式(6)为贝尔曼公式,它表示了当前状态和其后继状态之间的联系。图 3 表示了强化学习中 Q 值迭代与状态转移的回溯关系。图 3(a)中,每一个实心点表示一个状态-联合动作对,每一个空心点表示一个状态,智能体从一个状态-联合动作对出发,依次到达下一个状态。在图 3(b)中,智能体团队在 s_{t+1}状态下得到最优策略$(s_{t+1},\vec{a})$,假设团队在下一状态没有被事件触发,则不进行状态转移,直接延续上一时刻的最优策略$(s_{t+1},\vec{a})$。



(a)传统的 Q-学习



(b)基于事件驱动的 Q-学习

图 3 两种方式回溯图

Fig.3 The backtracking of two methods

根据贝尔曼迭代,Q 值逐渐收敛到一个最优 Q 值,在传统的强化学习中,每一个学习步智能体都需要通过查表方式找到最大的 Q 值,其迭代表达式为

$$\Delta Q(s_t,\vec{a}_t)=r_t+\gamma\max_{\vec{a}\in A}Q_t(s_{t+1},\vec{a}_t)-Q_t(s_t,\vec{a}_t) \tag{7}$$

$$Q_t(s_t,\vec{a}_t)=Q_t(s_t,\vec{a}_t)+\alpha\Delta Q_t(s_t,\vec{a}_t)=$$
$$Q_t(s_t,\vec{a}_t)+\alpha(r_t+\gamma\max_{\vec{a}\in A}Q_t(s_{t+1},\vec{a}_t)-Q_t(s_t,\vec{a}_t))=$$
$$(1-\alpha)Q_t(s_t,\vec{a}_t)+\alpha(r_t+\gamma\max_{\vec{a}\in A}Q_t(s_{t+1},\vec{a}_t)) \tag{8}$$

事件驱动的思路则不同,当智能体没有被触发情况下,将直接选用上一个 Q 值作为当前的 Q 值,在基于事件驱动的 Q-学习中,Q 值迭代过程可以表示为

$$Q_t(s_t,\vec{a}_t,e)=(1-\alpha)Q_{t-k}(s_t,\vec{a}_t,e)+$$
$$\alpha(r_t+\gamma\max_{\vec{a}\in A}Q_t(s_{t+1},\vec{a}_t,e)) \tag{9}$$

式中 k 表示上次触发时刻和当前时刻的差值。

3.2　计算资源消耗

Q-学习中的计算资源消耗,主要体现在智能体需要对所有策略进行试错。从决策树角度看,树根和树枝对应着智能体团队的状态与观测,其在每一次观测后,根据不同的观测都会转移到不同的下一刻状态,即$\{s_{t+1}\leftarrow s_t|(\vec{o},\vec{a})\}$。在每一个树层中,智能体团队需要通过遍历 Q 值表,查找得到一个最优策略。Q 值表的实现采用 Lookup 表格来表示 Q 函数。设 $Q(s,\vec{a})$ 为一个 Lookup 表格,$s\in S$ 和 $\vec{a}\in\vec{A}$ 为有限集合,表的大小等于 $S\times\vec{A}$ 的笛卡尔乘积中的元素的个数。举例说明,假设存在 i 个智能体,每一个智能体有 m 个动作,每一时刻有 n 个状态,Q 值表的大小为 $n^i\times m^i$,在第 t 步,智能体共需遍历 $(n^i\times m^i)\times t$ 次 Q 值,当参与学习的智能体数量较多,以及每一个智能体的动作和状态集合较大时,查表需要占用极大的计算资源。

对于基于事件驱动的决策树,在智能体不被驱动的树层中,下一刻状态将直接等于当前状态,即 $s_{t+1}=s_t$,状态转移概率为

$$P^{\vec{a}}_{s_ts_{t+1}}=Pr\{s_{t+1}=s_t|\vec{a}_{t+1}=\vec{a}_t\}=1$$

状态转移概率 $Pr=1$ 意味着,此时整棵决策树中不被驱动的树层不生成树枝,进而也减少下一层中树枝对应的树根。同理,不生成新的树枝,智能体也无需对当前树层里所有的 Q 值进行遍历。上述例子中,假设 t 步中存在 k 次不被驱动,那么在 t 步学习过程中,遍历 Q 值的次数为$(n^i\times m^i)\times(t-k)$次。

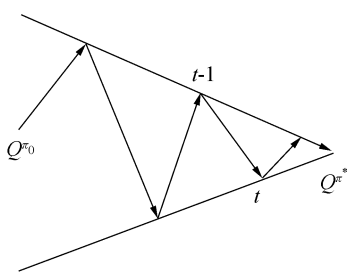
3.3　算法收敛性分析

智能体每次的策略评估,即策略迭代,都是从前一个策略的值函数开始。在事件驱动的强化学习中,智能体只有在观测信息变化情况下,才更新信念空间并进行策略评估,否则直接使用上一时刻的策略。假设在 t 时刻,智能体没有被事件所触发,那么智能体在 t 时刻不参与式(9)的迭代,直接使用 $t-1$ 时刻迭代后的 Q 值。此时,在达到最优策略的过程中,Q 值的迭代计算过程由每一时刻都计算,减少为事件触发时刻才计算。

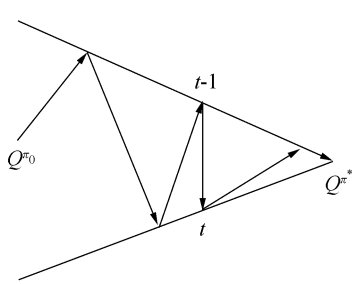
$$\pi_0\rightarrow Q^{\pi_0}\rightarrow\pi_1\rightarrow Q^{\pi_1}\rightarrow\pi_2\rightarrow Q^{\pi_2}\rightarrow\pi_3\rightarrow Q^{\pi_3}\rightarrow\cdots\pi^* \tag{10}$$

$$\pi_0\rightarrow Q^{\pi_0}\rightarrow\pi_1\rightarrow Q^{\pi_1}\rightarrow\pi_2\rightarrow Q^{\pi_2}\rightarrow\pi_2\rightarrow Q^{\pi_2}\rightarrow\cdots\pi^* \tag{11}$$

如图 4(a)和式(10)所示,Q 值从初始到收敛至最优 Q^* 的过程,是一个渐进收敛的过程,Q 值通过迭代,从 $t-1$ 时间到 t 时刻逐渐接近最优;如图4(b)和式(11)所示,在智能体不被驱动的情况下,Q 值不进行迭代,在 $t-1$ 时刻直接使用 t 时刻的 Q 值,减少了 Q 值的迭代计算。



(a)经典的 Q-学习策略迭代



(b)基于事件驱动的 Q-学习策略迭代

图 4　两种方式策略迭代

Fig.4　Policy iteration of two methods

推论 1　基于事件驱动的 Q-学习算法,不会影响算法的收敛性。

引理 1　收敛引理[12]。令 χ 为一个任意的集合,假设 B 是 χ 中一个空间有界的集合,即 $B(\chi)$,$T:B(\chi)\rightarrow B(\chi)$。$v^*$ 为 T 的一个固定点,令 $\tau=(T_0,T_1,\cdots)$ 为来自 $F_0(v^*)$ 的初值,τ 在 v^* 点逼近 T,假设 F_0 为 τ 中一个不变式。令 $V_0\in F_0(v^*)$,定义 $V_{t+1}=T_t(V_t,V_t)$。如果存在随机函数 $0\leqslant F_t(x)\leqslant 1$ 和 $0\leqslant G_t(x)\leqslant 1$ 以概率 1 满足以下条件,那么在 $B(\chi)$ 中 V_t 以概率 1 收敛到 v^*:

1)对所有的 U_1 和 $U_2\in F_0$,对所有的 $x\in\chi$,

$$|T_t(U_1,v^*)(x)-T_t(U,V)(x)|\leqslant$$
$$G_t(x)|U_1(x)-U_2(x)| \tag{12}$$

2)对所有的 U 和 $V \in F_0$,对所有的 $x \in X$,

$$|T_t(U_1, v^*)(x) - T_t(U, V)(x)| \leqslant F_t(x)(\|v^* - V\| + \lambda_t) \quad (13)$$

式中:当 $t \to \infty$ 时,λ_t 以概率 1 收敛到 0。

3)对所有的 $k>0$,当 $t \to \infty$ 时,$\prod_{t=k}^{n} G_t(x)$ 收敛到 0。

4)当 $t \to \infty$ 时,存在 $0 \leqslant \gamma < 1$ 对所有的 $x \in X$ 有

$$F_t(x) \leqslant \gamma(1 - G_t(x)) \quad (14)$$

证明 在事件驱动的强化学习中,令 $T = (T_0, T_1, \cdots, T_k, T_{k+1} = T_k, T_t, \cdots)$ 为一个动作序列,表示智能体执行行动后从当前状态到下一个状态的映射,其中 $(\cdots T_k, T_{k+1} \cdots)$ 指当智能体在没有被事件驱动的情况下智能体的第 T_{k+1} 个行动等于第 T_k 个行动,同时,迭代过程为

$$f_{t+2} = T_{k+1}(f_{t+1}, f_{t+1}) = T_k(f_t, f_t) \quad (15)$$

令 $V, U_0, V_0 \in B(X)$, $U_{t+1} = T_t(U_t, V)$, $V_{t+1} = T_t(V_t, v^*)$, $\delta_t(x) = |U_t(x) - V_t(x)|$。根据收敛引理有

$$\begin{aligned}
\delta_{t+1}(x) &= |U_{t+1}(x) - V_{t+1}(x)| = \\
&|T_t(U_t, v^*)(x) - T_t(V_t, V_t)(x)| \leqslant \\
&|T_t(U_t, v^*)(x) - T_t(V_t, v^*)(x)| + \\
&|T_t(V_t, v^*)(x) - T_t(V_t, V)(x)| \leqslant \\
&G_t(x)|U_t(x) - V_t(x)| + F_t(x)(\|v^* - V_t\| + \lambda_t) = \\
&G_t(x)\delta_t(x) + F_t(x)(\|v^* - V_t\| + \lambda_t) \leqslant \\
&G_t(x)\delta_t(x) + F_t(x)(\|v^* - V_t\| + \|U_t - V_t\| + \lambda_t) = \\
&G_t(x)\delta_t(x) + F_t(x)\|v^* - V_t\| + \lambda_t \quad (16)
\end{aligned}$$

在满足条件 1)和 2)的情况下,虽然基于事件驱动的动作序列 T 中有相同的动作 $T_k = T_{k+1}$,但仍然满足李普西斯条件,所以不会影响 Q-学习的收敛,证毕。

4 仿真结果及分析

考虑一个多智能体覆盖问题,2 个智能体随机出现在一个大小为 10×10 的格子世界中,如图 5 所示。每一个智能体都有上下左右 4 个行动,且观测范围为自身周围一圈共 8 个格子,观测到的格子分为“没走过”“走过”和“障碍物”3 个状态,分别对应着 30、-5 和-10 的回报值,世界的边界对智能体作为障碍物;且每一个智能体可以进行广播式通信。在这个场景中,每一个智能体获得的是一个局部观测,当它们进行广播通信后,对于整个世界,获得的仍然是一个局部的观测。但考虑到对整个世界的全局观测需要极大的计算量,所以实验设定每一时刻当两个智能体通信后,所获得的信息对它们而言是一个全局观测。

智能体团队的任务为尽快走完所有的格子,即完成对格子世界的覆盖,当走过的格子超过 90%以上,认为此次覆盖任务成功,当智能体在 1 000 步仍不能完成 90%的覆盖时,认为此次任务失败。其中定义学习率为 0.6,折扣因子为 0.2。

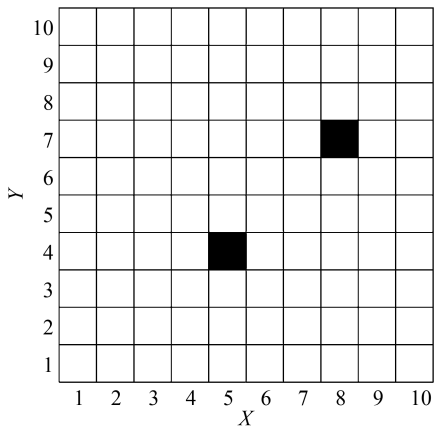


图 5 多智能体覆盖问题

Fig.5 The coverage problem of multi-agent

图 6 比较了事件驱动与传统 Q-学习任务成功率,可以看出两种算法成功率一致,但是由于 Q 值迭代次数减少,使得事件驱动 Q-学习的收敛速度变慢。

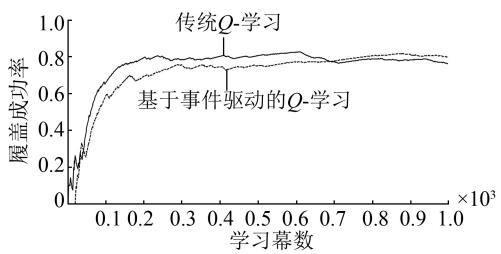


图 6 事件驱动与传统 Q-学习的成功率

Fig.6 The success rate of event-triggered Q and classical Q

图 7 说明了联合触发函数与算法收敛速度的关系,可以看出联合触发函数选取越小,算法收敛性越慢。因为联合触发函数越小,事件触发的次数就越少,从而导致 Q 值迭代次数减少,收敛速度变慢。

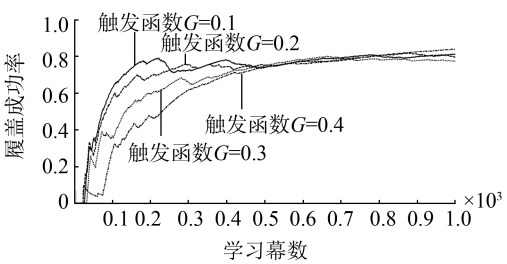


图 7 联合触发函数与收敛速度

Fig. 7 The joint event-triggered function and convergence speed

在学习过程中,智能体团队在每一步需要遍历 Q 值数量为 $(3^8 \times 4)^2 \approx 2^{29.3}$ 次,由表 1 可以看出,随着学习步数的增加,事件驱动将大量减小 Q 值的遍历次数,继而减少计算资源占用,相比较传统的 Q-学习存在明显的优势。

表 1　事件驱动传统 Q-学习遍历次数

Table 1　The number of traverse of event-triggered and classical Q

步数	Q-学习	事件驱动 Q-学习	减少总遍历次数
50	$\approx 2^{29.3}\times 50$	$\approx 2^{29.3}\times 42$	$\approx 2^{32.3}$
100	$\approx 2^{29.3}\times 100$	$\approx 2^{29.3}\times 79$	$\approx 2^{33.6}$
200	$\approx 2^{29.3}\times 200$	$\approx 2^{29.3}\times 153$	$\approx 2^{34.9}$
300	$\approx 2^{29.3}\times 300$	$\approx 2^{29.3}\times 221$	$\approx 2^{35.6}$
500	$\approx 2^{29.3}\times 500$	$\approx 2^{29.3}\times 386$	$\approx 2^{36.2}$

表 2 比较了在一次成功的任务中,事件驱动与传统 Q-学习的通信次数。可以看出,事件驱动减少了智能体间的通信次数。同时与表 1 比较,可以看出自事件触发和联合事件触发次数的区别。

表 2　事件驱动与传统 Q-学习通信次数

Table 2　The number of communication of event-triggered and classical Q

步数	Q-学习	事件驱动 Q-学习	减少通信次数
50	50	45	5
100	100	89	11
200	200	172	28
300	300	258	42
500	500	410	90

5　结束语

本文提出了一种基于事件驱动的多智能体强化学习算法,侧重于多智能体在学习策略层的事件驱动研究。智能体在与环境的交互中,可以根据观测的变化来触发通信和学习过程。在相同时间内,采用事件驱动可以降低数据传输次数,节约通信资源;同时,智能体不需要每一时刻进行试错和迭代,进而减少计算资源。最后,对算法的收敛性进行了论证,仿真结果表明事件驱动可以在学习过程中减少一定的通信次数和策略遍历次数,进而缓解通信和计算资源消耗。进一步工作主要基于现有的研究,将事件驱动的思想应用于不同类的强化学习方法中,并结合事件驱动的特点设计更合理的触发函数。

参考文献:

[1] ZHU Wei, JIANG ZhongPing, FENG Gang. Event-based consensus of multi-agent systems with general linear models [J]. Automatica, 2014, 50(2): 552-558.

[2] FAN Yuan, FENG Gang, WANG Yong, et al. Distributed event-triggered control of multi-agent systems with combinational measurements[J]. Automatica, 2013, 49(2): 671-675.

[3] WANG Xiaofeng, LEMMON M D. Event-triggering in distributed networked control systems[J]. IEEE transactions on automatic control, 2011, 56(3): 586-601.

[4] TABUADA P. Event-triggered real-time scheduling of stabilizing control tasks[J]. IEEE transactions on automatic control, 2007, 52(9): 1680-1685.

[5] ZOU Lei, WANG Zidong, GAO Huijun, et al. Event-triggered state estimation for complex networks with mixed time delays via sampled data information: the continuous-time case[J]. IEEE transactions on cybernetics, 2015, 45(12): 2804-2815.

[6] SAHOO A, XU Hao, JAGANNATHAN S. Adaptive neural network-based event-triggered control of single-input single-output nonlinear discrete-time systems[J]. IEEE transactions on neural networks and learning systems, 2016, 27(1): 151-164.

[7] HU Wenfeng, LIU Lu, FENG Gang. Consensus of linear multi-agent systems by distributed event-triggered strategy [J]. IEEE transactions on cybernetics, 2016, 46(1): 148-157.

[8] ZHONG Xiangnan, NI Zhen, HE Haibo, et al. Event-triggered reinforcement learning approach for unknown nonlinear continuous-time system[C]//Proceedings of 2014 International Joint Conference on Neural Networks. Beijing, China, 2014: 3677-3684.

[9] XU Hao, JAGANNATHAN S. Near optimal event-triggered control of nonlinear continuous-time systems using input and output data[C]//Proceedings of the 11th World Congress on Intelligent Control and Automation. Shenyang, China, 2014: 1799-1804.

[10] BERNSTEIN D S, GIVAN R, IMMERMAN N, et al. The complexity of decentralized control of Markov decision processes[J]. Mathematics of operations research, 2002, 27(4): 819-840.

[11] WATKINS C J C H, DAYAN P. Q-learning[J]. Machine learning, 1992, 8(3/4): 279-292.

[12] SZEPESVÁRI C, LITTMAN M L. A unified analysis of value-function-based reinforcement-learning algorithms [J]. Neural computation, 1999, 11(8): 2017-2060.

作者简介:

张文旭,男,1985 年生,博士研究生,主要研究方向为多智能体系统、机器学习。发表论文 4 篇,其中被 EI 检索 4 篇。

马磊,男,1972 年生,教授,博士,主要研究方向为控制理论及其在机器人、新能源和轨道交通系统中的应用等。主持国内外项目 14 项,发表论文 40 余篇,其中被 EI 检索 37 篇。

王晓东,男,1992 年生,硕士研究生,主要研究方向为机器学习。获得国家发明型专利 3 项,发表论文 4 篇。