ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.19, No.11, November 2008, pp.2957−2967
DOI: 10.3724/SP.J.1001.2008.02957

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

基于后悔值的多Agent冲突博弈强化学习模型*

肖 正+, 张世永

(复旦大学 计算机与信息技术系,上海 200433)

Reinforcement Learning Model Based on Regret for Multi-Agent Conflict Games

XIAO Zheng+, ZHANG Shi-Yong

(Department of Computer and Information Technology, Fudan University, Shanghai 200433, China)

+ Corresponding author: E-mail: xiaozheng206@163.com, http://www.fudan.edu.cn

Xiao Z, Zhang SY. Reinforcement learning model based on regret for multi-agent conflict games. *Journal of Software*, 2008,19(11):2957−2967. http://www.jos.org.cn/1000-9825/19/2957.htm

Abstract: For conflict game, a rational but conservative action selection method is investigated, namely, minimizing regret function in the worst case. By this method the loss incurred possibly in future is the lowest under this very policy, and Nash equilibrium mixed policy is obtained without information about other agents. Based on regret, a reinforcement learning model and its algorithm for conflict game under multi-agent complex environment are put forward. This model also builds agents' belief updating process on the concept of cross entropy distance, which further optimizes action selection policy for conflict games. Based on Markov repeated game model, this paper demonstrates the convergence property of this algorithm, and analyzes the relationship between belief and optimal policy. Additionally, compared with extended *Q*-learning algorithm under MMDP (multi-agent markov decision process), the proposed algorithm decreases the number of conflicts dramatically, enhances coordination among agents, improves system performance, and helps to maintain system stability.

Key words: Markov game; reinforcement learning; conflict game; conflict resolving

摘 要: 对于冲突博弈,研究了一种理性保守的行为选择方法,即最小化最坏情况下 Agent 的后悔值.在该方法下,Agent 当前的行为策略在未来可能造成的损失最小,并且在没有任何其他Agent信息的条件下,能够得到Nash均衡混合策略.基于后悔值提出了多 Agent 复杂环境下冲突博弈的强化学习模型以及算法实现.该模型中通过引入交叉熵距离建立信念更新过程,进一步优化了冲突博弈时的行为选择策略.基于 Markov 重复博弈模型验证了算法的收敛性,分析了信念与最优策略的关系.此外,与 MMDP(multi-agent markov decision process)下 *Q* 学习扩展算法相比,该算法在很大程度上减少了冲突发生的次数,增强了 Agent 行为的协调性,并且提高了系统的性能,有利于维持系统的稳定.

关键词: Markov 对策;强化学习;冲突博弈;冲突消解

中图法分类号: TP18 **文献标识码**: A

对于单个 Agent,需要选择合适的行为以适应环境的变化,以最高效的方式完成任务.而在一个多 Agent 系

* Received 2007-06-28; Accepted 2007-08-24

统中,Agent之间的关系常常是复杂多样的,多个自治的Agent共享同一个环境,其他Agent的行为也会对环境产生影响,因此,决策时必须考虑其他Agent可能采取的动作.资源的有限或Agent之间目标的不同,都会导致各种冲突,如竞争有限的资源.即使共享同一个目标,在分工合作时也会发生冲突.此时,单个Agent的最优行动方案将不再最优,甚至可能成为最差的.因此,多Agent行为决策需要能够在消除多方冲突的基础上产生最优的策略,达到彼此行为的充分协调,提高系统性能.

MDP(Markov decision process)认为,Agent决策仅取决于当前的状态,而与历史无关,被用来解决随机序贯决策问题.对于单Agent与环境交互的系统,该形式化框架对Agent决策是一个较好的表示[1,2].该方法扩展到多Agent环境下被称为Markov对策(Markov game)[3].Markov对策框架下,Agent在制订决策时不仅考虑了Agent与环境之间的交互,而且考虑了与其他Agent之间的交互.将多个Agent之间的交互作为一个随机博弈(stochastic game),为多Agent系统中冲突的解决提供了有效的理论框架.本文即建立在Markov对策框架下,对Agent冲突的消解进行了研究.

强化学习是从动物学习、参数扰动自适应控制等理论发展而来的,如果Agent的某个行为策略导致环境正的奖赏,那么Agent以后产生这个行为策略的趋势便会加强.强化学习为多Agent之间的协作提供了鲁棒的学习方法,在没有外界指导的情况下,Agent通过与不确定环境的不断交互获得最优解.这种无监督试错型学习方法已经被用来解决MDP[4]或Markov对策问题[3,5−9].我们将多方利益冲突下的Agent决策问题定义为一个Markov对策问题,利用强化学习的方法找到协调各个Agent行为的最优策略.

本文针对多Agent环境下经常发生的由冲突博弈所描述的一类冲突进行了研究.该类冲突即使允许Agent之间相互交互协商,有时候也难以达成一致,从而无法避免冲突的发生,阻碍了Agent目标的实现,导致整个系统性能的下降.我们通过对这类冲突的博弈矩阵的分析,从Agent进行理性行为选择的角度出发,借鉴定性决策理论中的最小化最大后悔值的决策判据,找到了冲突博弈下Agent的最优策略,且在一定条件下,该策略略亦为Nash均衡策略.进一步从实际应用的角度出发,在Markov对策框架下提出了该方法的强化学习模型.不同于以往的研究,此学习算法专门针对多Agent系统中经常发生的冲突博弈,将定性决策理论中的方法引入到强化学习算法中,为其提供了一种有效的Agent行为决策方法.同时,利用信息论中的交叉熵的概念定义了Agent信念交叉熵距离,缓解频繁信念变化状况下学习算法动荡的缺点;建立了Agent信念更新过程,进一步优化Agent的策略.将Agent信念逻辑上的差异定量地外化为交叉熵距离,有助于算法的设计和实现.仿真实验表明了基于后悔值的冲突博弈强化学习算法在Agent没有协商的条件下,能够大大降低冲突发生的频率;此外,其策略的Nash均衡特性使得Agent能够公平地、尽可能多地完成各自的任务,保持系统处于均衡稳定状态.

1 Markov对策

定义 1(马尔可夫决策过程 MDP). 设存在四元组$\langle S,A,T,R\rangle$,其中S为离散状态集,A为行为集,状态转移函数$T:S\times A\to \Pr(S)$,报酬函数$R:S\times A\to \Re$. MDP的目标是找到最优策略,使得折扣期望报酬和$E\left(\sum_{j=0}^{\infty}\gamma^j r_{t+j}\right)$最大($\gamma$为折扣因子,$r_{t+j}=R(s_j,a_j)$为$t+j$时刻的立即报酬).

对于以上的MDP问题,可以用强化学习方法中的Q学习算法求解.Q值取决于Agent所处状态和采取的动作,具体计算表达式如下:

$$Q(s,a)\leftarrow(1-\alpha)Q(s,a)+\alpha[R(s,a)+\gamma V(s')],$$

$$V(s)\leftarrow\max_{a\in A}Q(s,a).$$

根据Q值得到Agent在状态s的最优策略为

$$\pi^*(s)=\arg\max_{a\in A}Q(s,a).$$

在多Agent系统中,最优策略是状态和其他Agent行为的函数.只需将MDP模型稍加扩展,即可得到Markov对策模型.Markov对策模型可以表示为一个五元组$\langle N,S,\vec{A},T,R\rangle$,其中$N$为Agent集合,$\vec{A}\in A_1\times A_2\times\ldots\times A_n$为联合行动集.相应的状态转移函数$T$,报酬函数$R$均为状态与联合行动$\vec{a}$的函数.$Q$值的更新必然与其他Agent的行

为选择有关,这时 Agent i 的 Q 值迭代式为式(1),V 值取决于 Agent 的策略.不同的对弈情境下 Agent 的行为策略也不同,因此,研究者对其中的一些博弈情境进行了研究,在下一节我们将给出简单介绍.

$$Q_i(s,\vec{a}) \leftarrow (1-\alpha)Q_i(s,\vec{a}) + \alpha[R_i(s,\vec{a}) + \gamma V_i(s')] \tag{1}$$

2 相关工作

利用强化学习的方法来解决 Markov 对策问题,关键是如何定义 V 值,使得 Q 值收敛到最优策略对应的值.最简单的方法 MMDP(multi-agent markov decision process)[5]直接从单 Agent MDP 强化学习方法扩展得来.在 MMDP 中,用 Q 的最大值更新 V 值,最优策略为使得 Q 值最大的联合行动.当 Agent 之间为存在优势策略均衡的博弈时,该方法能够收敛到均衡对应的策略.但正如文章开始所指出的,在多 Agent 环境下,个体之间的关系是复杂的,各种冲突的存在导致 Agent 之间的博弈是多样的,可能没有优势策略均衡,或者存在多个 Nash 均衡,或者不存在均衡解等.

在多 Agent 环境下,Littman,Claus,Boutilier,高阳等人对多种多样的博弈中某一类特殊博弈进行了研究.针对零和博弈(zero-sum game),Littman 提出了极大极小 Q 学习算法(maxmin-Q learning)[3].零和博弈是指在任何策略下,所有 Agent 的报酬之和为 0.当 Agent 的目标完全相反时,可以用该博弈模型表示.以两个 Agent 为例,在状态 s 下,Agent A 最优策略为对手 O 选择最坏动作时最有利的动作.定义如下:

$$V(s) = \max_{\pi \in PD(A)} \min_{o \in O} \sum_{a \in A} \pi_a Q(s,(a,o)).$$

从以上公式可以看出,极大极小 Q 学习算法支持混合策略,能够找到唯一 Nash 均衡的混合策略解.此外,Claus 和 Boutilier 考虑了协作 Agent 之间的协调博弈(如图 1 所示)[7].由于 Agent 共享同一个目标,因此对于任一联合行动,它们的支付(payoff)是相等的.协调博弈具有以下特点:存在两个纯策略均衡,所有 Agent 对均衡的排序是相同的.在 Agent 行为可能相冲突的情况下,可以通过 Pareto 最优或事前协商的方式达成一致.但文献[1]通过改进强化学习的探索策略,无须通信即可使各个 Agent 一致收敛到最优均衡解.高阳等人[10]利用强化学习得到元对策平衡解,从而解决了非零和对策中组合谬误(fallacy of composition)的矛盾.

	a_0	a_1
b_0	(2,2)	(0,0)
b_1	(0,0)	(2,2)

(a_0,b_0) (a_1,b_1) are pure policy equilibriums

Fig.1 Coordination game

图 1 协调博弈

以上讨论的几类博弈都有特殊的性质,为了对于任意随机博弈都能得到最优均衡策略,Hu 和 Wellman[8,11]提出了 Nash-Q 学习算法(Nash-Q learning).该算法根据某一 Nash 均衡更新 V 值.

$$V_i(s) = NashQ_i(s) = \pi_1(s)...\pi_n(s)Q_i(s).$$

其中,$(\pi_1(s),...,\pi_n(s))$ 为一个 Nash 均衡策略.由于 Nash 均衡可能不止一个,因此 $V_i(s)$可能并不唯一,这使得 Nash-Q 算法成为一个不确定的过程.因此,Nash-Q 算法要付诸实践,还存在许多问题.此外,Littman 认为该算法是对 MMDP 和极大极小 Q 算法的整合,适用于优势策略均衡和存在鞍点的博弈.因此,Littman 基于 Nash-Q 算法提出了一种 FFQ 算法[6].在两人对策中,当对方是 Friend 时,将二者之间看作纯合作的 Team 形式,所以 Agent 只需追求自身的最大回报即可实现整体的最大回报;当对方为 Foe 时,则可将对策的 n 人划分为两个对立方,并运用极大极小 Q 学习算法.Bowling[12]指出,多 Agent 学习具有的两个期望性能——理性和收敛.Nash-Q 满足理性,但在很多情况下不能收敛;而 FFQ 收敛,却通常不够理性.因此,在这两种算法的基础上,Greenwald 提出了一种既收敛又具有整体理性的算法 CEQ[13].该算法以相关均衡(correlated equilibrium)的解概念定义 Nash 平衡.相关均衡是联合动作空间上的一个概率分布,Agent 根据其他 Agent 对它的条件概率进行最优化,其值函数 V 定义为

$$V(s) \in CE(Q_1(s),Q_2(s),...,Q_n(s)).$$

对于多人一般和博弈,Nash 均衡的计算是一个 NP 难问题,所以很多学者致力于利用博弈中的最好回应来学习 Nash 均衡策略.最近常见的是策略梯度方法[14,15],通过迭代的策略梯度上升求得收敛于 Nash 均衡的最好回应策略.虽然该类算法不直接依赖于均衡,但因为最好回应与均衡之间存在着隐含的依赖关系,若学习最好回

应的算法收敛,则收敛点一定是一个 Nash 均衡点.针对具体的应用,Daniel 等人[16]利用多 Agent 学习在无限拍卖过程中逼近不完全信息情况下的最好回应和平衡.

本文延续 Littman,Claus,Boutilier,高阳等人的研究思路,研究一类特殊的冲突博弈(将在下一节给出详细描述).它类似于协调博弈,存在两个纯策略 Nash 均衡,但不同 Agent 对各个均衡的排序是相反的.对于这种冲突,即使允许 Agent 可以通信,也难以避免.这类冲突在非合作 Agent 争夺有限的资源时比较常见.下一节分析了此类博弈的最优策略,并建立了相应的强化学习模型.

3 基于后悔值强化学习模型

3.1 冲突博弈

在一个多 Agent 环境下,Agent 大致可分为两类:合作 Agent 与非合作 Agent.合作 Agent 彼此共享同一个目标,具有相同的支付函数,或者对支付的偏序结构一致.它们之间的冲突主要是对均衡的选择不一致(如上述的协调博弈).它们需要在两个纯策略均衡(a_0,b_0),(a_1,b_1)之间选择一个.这类冲突的本质是由于 Agent 无法知道其他 Agent 的信念和喜好,不知道其他 Agent 可能会采取什么行动,因而难以选择自己的最优动作.如果所有 Agent 都具有完美信息,则这类冲突可以在一定程度上加以避免.而对于非合作 Agent,它们具有不同的目标,都希望自己的利益最大化,能够最快地实现自己的目标.在多 Agent 系统中,常常会发生多 Agent 为了达成自己的目标而竞争同一种有限资源的情况,这时 Agent 之间就会发生严重的冲突.对于这类冲突,即使 Agent 能够交互,事先了解到对方的意图和喜好,也仍然无法避免这类冲突的发生.下面我们以两个 Agent 为例,他们驾驶汽车沿同一车道相向行驶.相遇时,如果双方继续各自的方向,将会发生碰撞,给双方造成严重损失;而如果一方避让,则会延后其到达目的地的时间,使其收益减小.该问题的博弈模型如图 2 所示.

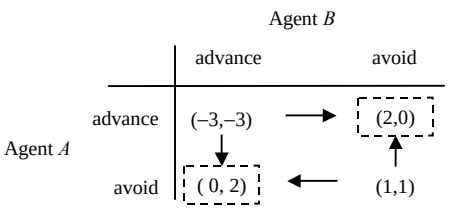

Fig.2 Conflict game

图 2 冲突博弈

冲突博弈有两个纯策略 Nash 均衡(advance,avoid)和(avoid, advance),但是它们存在非对称的缺陷,两个 Agent 对均衡的排序是相反的.Agent A 偏好于均衡(advance,avoid),而 Agent B 偏好于(avoid,advance),任何一个纯策略均衡都不是最佳预测.在没有通信的情况下,两个 Agent 该如何独立地进行决策,确定的行为策略显然不能在非合作 Agent 之间达成共识.此时,Agent 最佳的行为选择方式是依概率选择.该方式下的 Nash 均衡即为混合策略 Nash 均衡.混合策略下行为的不可预见性有时对 Agent 也是大有好处的.在机器人足球比赛(RoboCup)中,带球的 Agent 必须决定直接向前冲还是传球.一般而言,传球可以向前推进得更快,但是选择出乎对手意料之外的行动才是最重要的.因此,Agent 最佳的策略可能看似随机的,但却是理性考虑的结果.

3.2 最优策略

在冲突博弈的混合策略均衡中,Agent A 与 Agent B 在坚持与避让之间必须是无差异的,否则其中一方有偏离该策略的倾向.根据这一特点,Agent 的两个行为期望报酬相等,这时对应的策略满足 Nash 均衡.不妨设 A 坚持的概率为θ,而 B 选择坚持的概率为λ,则有以下方程:

$$\begin{cases} E_A(avoid)=\lambda\times(0)+(1-\lambda)\times(1)=\lambda\times(-3)+(1-\lambda)\times(2)=E_A(advance) \\ E_B(avoid)=\theta\times(0)+(1-\theta)\times(1)=\theta\times(-3)+(1-\theta)\times(2)=E_B(advance) \end{cases}.$$

由于图 2 所示冲突博弈恰好具有对称性,因此两个 Agent 的混合策略均衡相同且为(0.25,0.75).这时,最糟糕的情况下,Agent 相碰撞发生的概率为$(\theta\cdot\theta)$=0.0625.

对于均衡混合策略,按照上述支付均等化方法[17]在计算各自策略时需要知道对方的报酬函数,而由于通信开销过大等原因,知道所有 Agent 的报酬函数是不现实的.而当面临这种无法调和的冲突时,一个理性的 Agent 往往希望此刻的行为在将来看来是最不后悔的选择,后悔值越小越好,但由于其他 Agent 行为不确定,一种保守

的做法即最坏情况下期望后悔值最小.若 B 选择了动作 b_j,则 A 选择行为 a_i 的后悔值 $reg^{b_j}(a_i)$ 为 b_j 下 A 的最大报酬函数与当前行为 a_i 报酬的差.在已知 B 的策略时,我们称 Agent A 的最优策略为最佳响应策略,定义如下:

定义 2(最佳响应策略(optimal response policy)). 若 B 的策略为 π_B,A 的报酬函数为 $reg^{b_j}(a_i)$,则 A 的最佳响应策略为

$$\pi_A^* = \arg \min_{\pi_A \in \Pr(A)} \max_{b_j \in B} [\pi_B(b_j) \sum_{a_i \in A} \pi_A(a_i) reg^{b_j}(a_i)].$$

其中,后悔值函数 $reg^{b_j}(a_i)$ 计算如下:

$$reg^{b_j}(a_i) = \max_{a_i \in A} rew^{b_j}(a_i) - rew^{b_j}(a_i).$$

在已知 B 的策略 π_B 时,可以通过以上定义得到 A 的最佳响应策略 π_A^*.以图 2 的两个 Agent 冲突博弈为例,A 的后悔值矩阵如图 3 所示,假设 A 与 B 之间无法通信,且 A 没有任何关于 B 的信息,这时在 A 看来,B 的两个行为应该是等概率的,即 A 预测 B 的策略为(π_B(advance),π_B(avoid))=(0.5,0.5),则 A 在最坏情况下的后悔值为 min(1.5π_A(advance),0.5π_A(avoid)).当 1.5π_A(advance)=0.5π_A(avoid)=0.5(1−π_A(advance))时,使得最坏情况下期望后悔值最小(如图 4 所示),所以 A 的策略(π_A(advance),π_A(avoid))=(0.25,0.75),该策略也正是均衡混合策略.因此,两个独立封闭的 Agent 基于以上最佳响应策略的定义独自决策时能得到均衡混合策略.而若 A 拥有关于 B 的部分知识时,最佳响应策略在 A 对 B 的预测策略下使得 A 未选择最优动作而可能遭受最大损失最小,就后果平均严重程度而言,该策略风险是最低的.

		Agent B	
		advance	avoid
Agent A	advance	3	0
	avoid	0	1

Fig.3 Regret value matrix

图 3 后悔值矩阵

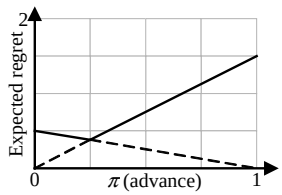

Fig.4 Regret value curve in worst case

图 4 最坏情况后悔值曲线

不难得出,在一般多 Agent 情况下,Agent i 的最佳响应策略为

$$\pi_i^*(A_i) = \arg \max_{\pi_i \in \Pr(A_i)} \min_{a_{-i}} \left[\pi(a_{-i}) \sum_{a_i \in A_i} \pi_i(a_i) reg^{a_{-i}}(a_i) \right].$$

其中,a_{-i} 表示除 Agent i 之外其他 Agent 的联合行动, $\pi(a_{-i})$为其余 Agent 各自策略下行动组合 a_{-i} 的概率.该式可以通过线性规划的方法求解.

3.3 Q学习模型

在上一节最优策略的计算过程中,需要事先获得 Agent 的博弈效用矩阵,即联合行动的回报.现实中这一信息往往难以预先得到.因为在实际应用中,Agent 往往需要执行一系列的动作才能实现某个目标,Agent 的行为选择是一个序贯决策问题,当前行为的回报受到未来若干行为的影响.立即报酬无法准确反映当前状态下多个行为之间的偏序关系,即立即报酬大的行为并不一定是达到目标最佳的策略.要获得 Agent 行为的回报函数,一种解决方法是采用动态规划的方法,但其高昂的计算开销不适合 Agent 的实时行为选择,且难以应用到开放的动态的多 Agent 环境中.在线不断地学习 Agent 联合行动报酬是一个可行的方法,即 Q 学习.Q 函数反映了某个状态下联合行动的期望回报.

从上文以及式(1)不难发现,如何迭代更新整体回报值函数 $V(s)$是 Q 学习的一个关键问题.在 Q 学习算法中经常被使用的一种整体回报定义是折算累积回报,把未来的回报相对于立即回报进行折算,因为在许多情况下,我们希望获得更快的回报.在多 Agent 环境下,Agent 之间的关系繁多、复杂,因此相应地衍生出了许多有关 $V(s)$

计算的研究,在第 2 节已作介绍.上一节我们对一类特殊的 Agent 关系——冲突博弈进行了分析,并找到了 Agent 之间发生冲突博弈时的最优策略,该计算过程为冲突博弈强化学习提供了折算累积回报 $V(s)$迭代更新的一种方法.根据过去的经验,采用策略迭代,利用当前的 Q 值得到最优策略,并同时可以计算出该策略下折算累积回报 $V(s)$,进而产生新的 Q 函数值,反复迭代更新,最终收敛到一个稳定的最佳响应策略.本节以下部分详细描述了一种基于上一节定义的最佳响应策略的冲突博弈 Q 学习模型.

在第 1 节介绍 Markov 对策时已经给出了利用 Q 学习方法时 Q 值的更新表达式.为了模型描述的完整性,这里重复给出,以 Agent i 为例.

$$Q_i(s,\vec{a}) \leftarrow (1-\alpha)Q_i(s,\vec{a}) + \alpha[R_i(s,\vec{a}) + \gamma V_i(s')].$$

α为学习速率,γ为折扣因子,Agent 完成任务越快,其得到的报酬也将越大.联合行动的报酬用当前 Q 值替代,则 Agent i 在状态 s 下的后悔值为

$$reg^{a_{-i}}(s,a_i) = \max_{a_i \in A} Q(s,a_i,a_{-i}) - Q(s,a_i,a_{-i}).$$

通过对冲突博弈的分析,理性 Agent 此时较好的决策思路是最小化最坏情况下的后悔值,以免在未来遭受较大的损失.这一做法虽然看似有些保守,但在冲突博弈环境下则不失为一种理性的抉择方法.这将在第 5.2 节得到证实.根据不同行为的后悔值,按照定义 2 更新策略:

$$\pi_i(s,A_i) \leftarrow \arg \min_{\pi_i \in \Pr(A_i)} \max_{a_{-i}} \left[\pi(s,a_{-i}) \sum_{a_i \in A_i} \pi_i(s,a_i) reg^{a_{-i}}(s,a_i) \right].$$

Agent 策略发生变化,必然会影响到状态 s 下折算累积回报值 $V(s)$.在一般的 Q 学习算法中,状态 s 下回报值 $V(s)$取所有行为中报酬 Q 最大的,而在我们的模型中得到的策略是一个概率分布,因此整体回报 $V(s)$依各行为概率按照期望报酬和计算更为合理.根据上式得到 i 最新的策略π_i更新状态 s 下的整体回报值 V:

$$V_i(s) \leftarrow \sum_{a_{-i}} \pi(s,a_{-i}) \sum_{a_i \in A_i} \pi_i(s,a_i) Q(s,a_i,a_{-i}).$$

在上式中,$\pi(s,a_{-i})$几乎是不可能事先知道的,本文采用假想对策(fictitious play)[18]中使用的方法:其他 Agent 的策略估计为 Agent i 基于当前对其他 Agent j($j \neq i$)的信念得出的概率预测值,即

$$\pi(s,a_{-i}) = \Pr(s,a_{-i}) = \prod_{j \neq i} \Pr\left(s,a_j \middle| Bel_i(j)\right).$$

将上式代入到π_i和 V 的更新式中,反复迭代,最终收敛得到 Agent i 的最优策略:

$$\pi_i^*(s) = \arg \min_{\pi_i \in \Pr(A_i)} \max_{a_{-i}} \left[\Pr(s,a_{-i}) \sum_{a_i \in A_i} \pi_i(s,a_i) \left(\max_{a_i \in A} Q(s,a_i,a_{-i}) - Q(s,a_i,a_{-i}) \right) \right].$$

当 Agent 的信念不变时,该模型得到的最优策略是静态的,不会随着时间而改变.但从 $\Pr(a_j|Bel_i(j))$可以看出,若 Agent i 的信念改变,则 Agent i 预测的 j 的策略也将改变,从而使得 i 的最优策略 $\pi_i^*(s)$ 发生变化.为了适应这一不确定性,需要重新启动 Q 学习过程,若信念没有错误,则会得到与其他 Agent 行为更匹配的策略.频繁再学习又会使策略不断动荡,而无法收敛.下面给出了何时进行再学习的决策方法.

定义 3(交叉熵(cross entropy)). p 和 q 是两个概率分布,二者的交叉熵定义为

$$D(p\|q) = \sum_{x \in \xi} p(x) \log \frac{p(x)}{q(x)}.$$

交叉熵的直观含义是对于随机变量 X,若开始认为其概率分布为 $q(x)$,则采用另一种概率分布 $p(x)$,这种变化导致观察者获得信息增量.设 Agent i 对 j 的信念更新后为 $Bel_i'(j)$,该信念下 i 预测的 j 的行为概率分布为 $\Pr(a_j|Bel_i'(j))$,简写为 $\Pr_i'(a_j)$,则信念更新前后的交叉熵为

$$D\left(\Pr'(a_j)\middle\|\Pr_i(a_j)\right) = \sum_{a_j \in A_j} \Pr_i'(a_j) \log \frac{\Pr'(a_j)}{\Pr_i(a_j)}.$$

根据 KL 距离(Kullback-Leibler divergence)的概念,定义 Agent i 对 j 的信念的变化程度(或信念交叉熵距离):

$$Diff(Bel_i(j), Bel'_i(j)) = D(\Pr_i(j)\|\Pr'_i(j)) + D(\Pr'_i(j)\|\Pr_i(j)) = \sum_{a_j \in A_j} (\Pr_i(a_j) - \Pr'_i(a_j))(\log \Pr_i(a_j) - \log \Pr'_i(a_j)) \quad (2)$$

只有当 Agent i 对 j 的信念改变大于一定的阈值δ时,才开始新一轮的策略学习过程.从另一个角度说,Agent i 能够容忍对信念改变不大于δ的小错误,且能够防止一部分 Agent 恶意欺骗.

4 算法实现

在经典 Q 学习算法中,假设 Agent 采取行动 a 使得状态由 s 转变为 s',并得到立即报酬 r,则 Agent 按照 $Q(s,a)=r+\gamma V(s')$更新 Q 值.但是,由于奖赏具有延迟回报的特点,导致存在时间信用分配(temporal credit assignment)的问题.一般采用时间差分法(temporal difference,简称 TD)来解决时间信用分配问题.因此,Q 学习算法中 Q 值的更新一般如第 1 节中式(1)所示.

要保证收敛到最优的策略,学习速率α不能太快.α随着时间慢慢减缓,衰落因子为μ.此外,学习过程中任意状态下任何动作都要被频繁访问足够多次,这就是 Q 学习算法中的探索控制机制.常用的探索方法有随机策略;ε贪婪策略;Boltzmann 探索等.由于探索方法不是本文的重点,本文采用较为简单的随机探索机制:以概率 P_e 进入探索过程,按照均匀分布随机选择行为,以保证所有状态和行为都可能被访问到;否则,根据当前策略π作出行为选择.详细算法如下:

算法 1. 基于后悔值的冲突博弈 Q 学习算法.

基于 Markov 对策框架$\langle N,S,\vec{A},T,R\rangle$,各符号的含义不变.

(1) 初始化:

for $\forall s\in S,\forall \vec{a}\in \vec{A}$,

$Q(s,\vec{a})=1$;

$V(s)=1$;

for $\forall s\in S$

$\pi_i(s,a_i)=1/|A_i|$;

for $\forall$ Agent $j\in N(j\neq i),\forall s\in S$

$\pi_j(s,a_j)=\Pr(s,a_j|Bel_i(j))$;

(2) 行为选择

if explore with probability P_e

return joint action $\vec{a}$ uniformly at random;

else

return joint action $\vec{a}$ according to individual policy π;

(3) 学习

Update Q: $Q_i(s,\vec{a}) \leftarrow (1-\alpha)Q_i(s,\vec{a}) + \alpha[R_i(s,\vec{a}) + \gamma V_i(s')]$;

Update π_i: $\pi_i(s) \leftarrow \arg \min_{\pi_i \in \Pr(A_i)} \max_{a_{-i}} \left[\Pr(s,a_{-i}) \sum_{a_i \in A_i} \pi_i(s,a_i) reg^{a_{-i}}(s,a_i) \right]$;

Update V: $V_i(s) \leftarrow \sum_{a_{-i}} \pi(s,a_{-i}) \sum_{a_i \in A_i} \pi_i(s,a_i) Q(s,a_i,a_{-i})$;

Update α: $\alpha \leftarrow \alpha \times \mu$;

(4) 信念更新

for $\forall j\in N(j\neq i)$

Retrieve $Bel'_i(j)$;

if $Diff(Bel_i(j),Bel'_i(j))>\delta$

$\pi_j(s,a_j)=\Pr(s,a_j|Bel'_i(j))$;

goto (2);

5 实验与结果分析

为了说明冲突博弈 Q 学习算法的有效性,而又不失一般性,本文在简单的 Markov 重复博弈模型上进行了实验.重复博弈假设每一个决策时刻,Agent 之间重复相同的博弈,其效用矩阵不变.系统状态转换是确定的,只取决于系统当前的状态以及所有 Agent 的动作.该模型下的学习算法试图找到适应多个 Agent 行为策略的最优策略.重复博弈模型是多 Agent 环境的一个简化和极端情况.本节首先对我们提出的冲突博弈 Q 学习算法的收敛性进行验证;然后说明收敛策略的有效性;最后描述 Agent 信念与最优策略之间的关系.

对现实中以下场景(如图 5 左图所示)建立 Markov 重复博弈模型.有两个 Agent,它们各自的任务是将处于河两岸 5 箱货物分别运送到河的另一边,一个 Agent 一次只能载重 1 箱货物,且连接两岸的桥一次只允许 1 个人通过,否则两个人有掉入河中的危险并且受到 3 个单位的惩罚.运送一箱货物得到 2 个单位的立即回报,仅当全部货物运送到对岸才算完成了一个任务,并得到额外 2 个单位的奖赏.系统状态定义为两边所剩货物数,系统的初始状态 S_0 为{5,5}.此时,若联合行动为(a_0,b_0),则两个 Agent 将会发生碰撞,均不能将货物成功送达,状态保持不变;若 A 选择通过,B 选择等待避让,则 A 将能完成一次货物运送,系统状态转变为(4,5).以此类推,系统状态转换如图 5 右图所示,m 和 n 分别表示 A 和 B 尚未搬运过河的货物数量.任何一方首先完成任务,系统将从初始状态开始新一轮的对弈.实验中,算法 1 的各参数设置如下:探索概率 P_e=0.35,学习时长为 10^4 个单位时间,初始学习速率α=1,为了使学习速率在学习过程结束时减小为 0.01,学习速率衰减因子$\mu=10^{\log 0.01/10^4}$=0.9995,Q 值更新式中折扣因子γ=0.9,立即回报函数 R 如图 5 状态转换箭头上方所给数值.

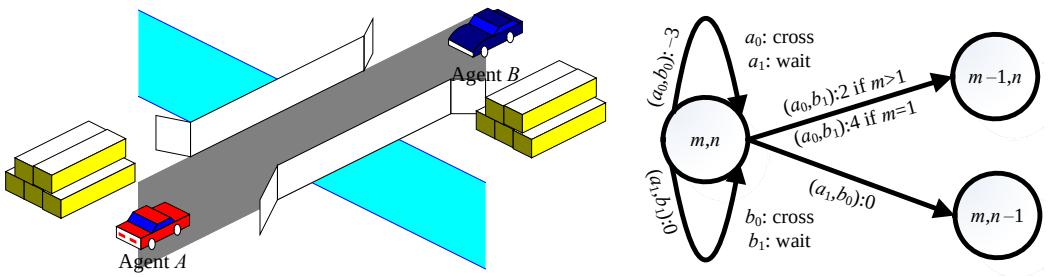

Fig.5 Fictitious experiment scene (state transforming shown right)

图 5 实验假设场景(右图为状态转换示例)

5.1 可收敛性

算法的可收敛性是迭代算法有效性的重要前提条件之一.基于上文描述的实验场景和参数设置,为了简化实验编程实现,我们增加了以下条件:假设 Agent B 的策略π_B为符合均匀分布的随机策略.Agent A 与 B 之间没有交互,A 没有对 B 的背景知识,也无法观测到 B 的行为或从其行为推测 B 的偏好,即 A 对 B 的信念始终是空,有$\Pr(s,b_0|bel_A(B))=\Pr(s,b_1|bel_A(B))=0.5(\forall s\in S)$.在以上假设下,Agent 通过算法 1 学习 B 的随机策略,并在 10^4 个单位时间内在状态(5,5)下 A 的 Q 值和策略π_A变化曲线如图 6 和图 7 所示.

从图 6 可以看出,当迭代次数达到 6 000 时,4 种可能联合行动的 Q 值变化缓慢,逐渐收敛到稳定的数值,并且联合行动(a_0,b_1)能够使 A 获得最大的报酬,联合行动(a_0,b_0)下由于两个 Agent 发生碰撞,而使 A 的收益最小.4 个 Q 值偏序关系满足冲突博弈时的效用矩阵.图 7 所示的 A 的最佳响应策略随着 Q 值的收敛也达到收敛.下一节将对算法 1 得到最佳响应策略的有效性进行说明,并验证冲突博弈下该算法的冲突消解性质.

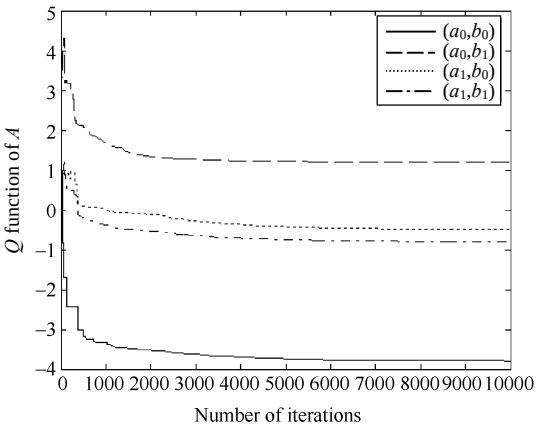

Fig.6 Four Q functions of A under state (5,5)

图 6 状态(5,5)下 A 的 4 个 Q 函数

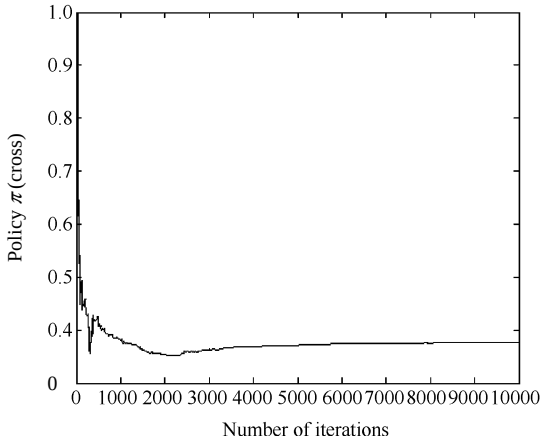

Fig.7 A's policy π(cross) in state (5,5)

图 7 状态(5,5)下 A 的策略π(通过)

5.2 有效性

多个 Agent 在面对可能发生的冲突时,首先往往希望尽可能避免冲突,然而当 Agent 之间无法事先协商或无法达成一致时,有些情况下冲突难以避免,如本文讨论的冲突博弈就属于这种情况.这时 Agent 希望减小冲突发生的频率,降低由于冲突带来的损失.然而另一方面,从整个系统的角度来看,希望各个 Agent 能够公平地实现各自的任务,系统稳定运转.例如,本实验中,希望两个 Agent 能够平等地利用有限资源,使得它们都能够充分完成各自的任务,而不偏向于任何一方,使得另一方有意图偏离当前策略.对于前一个指标,我们通过冲突发生的次数进行分析;而对于后一个指标,通过分析两个 Agent 完成任务的百分比进行验证.

实验中假设有 3 种 Agent:使用均匀分布随机策略的 Agent(Rand-Agt)、使用 MMDP 扩展 Q 学习算法的 Agent(Q-Agt)和使用本文提出的基于后悔值的强化学习算法的 Agent(RegretQ-Agt).实验场景仍然基于图 5,并且参数设置不变.两个 Agent 之间彼此不具有对方的知识,且没有任何信息交互.对其中 4 种 Agent 组合进行了实验,4 组实验运行 10 000 个单位时间,结果见表 1.

Table 1 Game results for Agents of different type

表 1 不同类型 Agent 组合博弈结果

A		Regret$Q$-Agt				Q-Agt			
		Games	A's task (%)	B's task (%)	Conflict	Games	A's task (%)	B's task (%)	Conflict
B	Rand-Agt	656	58.08	41.31	1 990	565	43.01	0.0	7 669
	RegretQ-Agt	643	51.17	48.52	1 987	–	–	–	–
	Q-Agt	–	–	–	–	1 000	0.0	0.0	100 000

当 A 使用 MMDP 扩展 Q 学习算法时,A 与 B 之间发生冲突的次数远远高于 A 使用本文的基于后悔值的强化学习算法.特别是,当 B 也为 Q-Agent 时,由于两方均自利地选取回报最大的行为(坚持通过),而使 A 与 B 之间永远处于冲突,而双方均无法实现各自目标.相比而言,当 Agent 为 RegretQ-Agt 时,可以充分减小发生冲突的次数,而降低冲突造成的损失.在本实验中,冲突次数降低了大约 75%.

此外,从表 1 第 1 行可以看出,当 B 采用随机策略时,无论 A 采用何种算法,A,B 分别完成任务的百分比都不能达成均等.这种情况下,任务完成数较少的一方有改变策略的动机,使得能够尽可能多地完成自己的任务.尤其当 A 为 Q-Agent 时,由于 A 的自利,而使较为仁慈(一定程度合作)的 B 永远无法实现自己的目标.从系统的角度来说,此时系统中的资源无法做到公平合理的分配和使用,系统资源利用率低,系统处于非平稳态.当 A,B 均采用基于后悔值的强化学习算法时,各自完成的任务百分比维持在 50%左右,从而充分利用了资源,并且保证了系统的稳定.

5.3 信念与最优策略

在本文提出的基于后悔值的强化学习模型中,Agent 的信念决定了其对对方行为的概率预测,因此其具有的信念的准确度影响了算法最终收敛策略的质量.图 8 展示了在不完美信念水平下,算法 1 得到策略的与完美信念下策略的交叉熵距离(方块虚线所示)以及此时冲突发生的次数(三角虚线所示),其中策略的交叉熵距离计算类似于式(2),仅将 $\Pr(s,a_j|Bel_i(j))$用$\pi_i(s,a_i)$替换.当信念交叉熵距离为 0 时, Agent 具有完美信念,对 B 的行为策略有准确的了解. Agent 预测 B 的行为概率分布与 B 实际的概率分布差异(图中 X 轴表示)越大,则当前信念下的最佳响应策略与完美信念下的最佳响应策略之间的差异也越大,此时的策略与完美信念下的最佳响应策略相比,发生冲突的总数也相应有所增加,策略质量下降.若 Agent A 能够了解 B 的相关行为决策知识,并观察 B 在线行为选择,而及时调整更新自己的信念,则能够趋向最优的最佳响应策略,使得冲突发生次数减少.

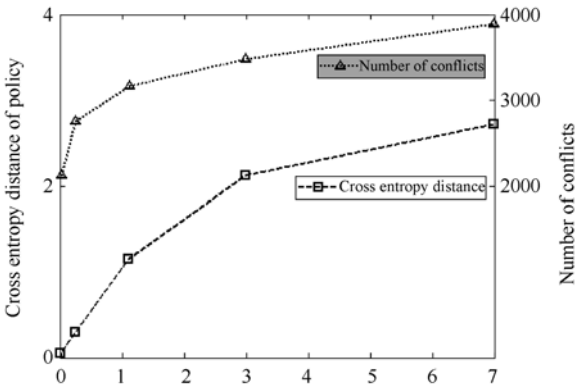

Fig.8 Influence on policy quality for belief

图 8 信念对策略质量的影响

在图 8 中,A 预测的 B 的策略与 B 实际策略的交叉熵距离越大,得到的最佳响应策略的质量也越差.因此,当 A 获得足够多的关于 B 的信念时,需要将这些信念应用到本文的基于后悔值的学习算法中,达到提高策略质量的目的.正如第 3 节所说,频繁的信念更新使得策略收敛减慢或无法收敛.所以,算法中当信念提高超过一定的阈值δ时,才使用新的信念进行学习.图 8 为阈值δ的设定提供了一定的参考.当信念交叉熵距离小于 1 时,策略交叉熵距离与信念交叉熵距离近似成正比例关系,即当信念逐渐接近完美信念时,最佳响应策略以同样的速率逼近最优最佳响应策略.因此,通过计算两个信念交叉熵距离,可以估计出最佳响应策略较之前一个策略的改善程度.反之,知道可接受策略的波动范围,可以得到学习算法中信念可暂时不更新的容忍范围.

6 结论及下一步工作

在 Markov 对策下,本文针对多 Agent 系统中经常出现的冲突博弈问题,基于最小化最坏情况下后悔值得到了该博弈下的最优策略,并提出了该方法的强化学习模型及算法实现.此外,引入交叉熵距离概念建立了 Agent 学习中信念更新过程,进一步提高了最优策略的质量.通过该模型得到的策略,能够减少冲突发生的次数,增强 Agent 之间行为的协调性,从而提高了系统性能.

本文提出的模型引入了 Agent 信念,得到的策略相应受到 Agent 当前信念的影响.恶意的 Agent 可能会发布虚假消息,或者采取误导行为,使得其他 Agent 错误地更新自己的信念,从而得到有利于恶意 Agent 的策略,使得本算法失效.因此,下一步将研究相关策略,防止恶意 Agent 进行欺骗,增强本文提出的算法的安全性和鲁棒性.此外,分析 Agent 信念到对行为概率预测之间的映射关系也有重要意义,有利于作出更准确的信念更新决策.

致谢 在此,我们向对本文的工作给予支持和建议的同行,尤其是复旦大学计算机与信息技术系张世永教授、

钟亦平教授领导的讨论班上的同学和老师,以及本文的审稿人表示感谢.

References:

[1] Tan M. Multi-Agent reinforcement learning: Independent vs. cooperative agents. In: Utgoff PE, ed. Proc. of the 10th Int'l Conf. on Machine Learning. San Francisco: Morgan Kaufmann Publishers, 1993. 330−337.

[2] Sen S, Sekaran M, Hale J. Learning to coordinate without sharing information. In: Proc. of the National Conf. on Artificial Intelligence. Menlo Park: AAAI Press, 1994. 426−431.

[3] Littman ML. Markov games as a framework for multi-agent reinforcement learning. In: Cohen WW, Hirsh H, eds. Proc. of the 11th Int'l Conf. on Machine Learning. San Francisco: Morgan Kaufmann Publishers, 1994. 157−163.

[4] Zhang SM, Shi CY. An efficient solution algorithm for factored MDP using feature vector extraction. Journal of Software, 2005, 16(5):733−743 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/16/733.htm

[5] Pappachan PM. An MDP-based policy for stochastic multi-agent domains. In: Proc. of the Int'l Conf. on Systems, Man, and Cybernetics (SMC'99). Piscataway: IEEE Press, 1999. 464−468.

[6] Littman ML. Friend-or-foe *Q*-learning in general-sum games. In: Brodley CE, Danyluk AP, eds. Proc. of the 18th Int'l Conf. on Machine Learning. San Francisco: Morgan Kaufmann Publishers, 2001. 322−328.

[7] Claus C, Boutilier C. The dynamics of reinforcement learning in cooperative multi-agent systems. In: Proc. of the 15th National Conf. on Artificial Intelligence. Menlo Park: AAAI Press, 1998. 746−752.

[8] Hu JL, Wellman MP. Multi-Agent reinforcement learning: Theoretical framework and an algorithm. In: Proc. of the 15th Int'l Conf. on Machine Learning. San Francisco: Morgan Kaufmann Publishers, 1998. 242−250.

[9] Shoham Y, Powers R, Grenager T. Multi-Agent reinforcement learning: A critical survey. Technical Report, Stanford: Stanford University, 2003.

[10] Gao Y, Zhou ZH, He JZ, Chen SF. Research on Markov game-based multi-agent reinforcement learning model and algorithm. Journal of Computer Research and Development, 2000,37(3):257−263 (in Chinese with English abstract).

[11] Hu JL, Wellman MP. Nash-*Q* learning for general-sum stochastic games. Journal of Machine Learning Research, 2003,4(6): 1039−1069.

[12] Bowling M, Veloso M. Rational and convergent learning in stochastic games. In: Proc. of the 17th Int'l Joint Conf. on Artificial Intelligence. Menlo Park: AAAI Press, 2001. 1021−1026.

[13] Greenwald A, Hall K. Correlated *Q*-learning. In: Proc. of the 20th Int'l Conf. on Machine Learning. San Francisco: Morgan Kaufmann Publishers, 1998. 242−250.

[14] Bowling M, Veloso M. Multi-Agent learning using a variable learning rate. Artificial Intelligence, 2002,136(2):215−250.

[15] Banerjee B, Peng J. Adaptive policy gradient in multi-agent learning. In: Proc. of the Autonomous Agent and Multi-Agent System (AAMAS). New York: ACM, 2003. 686−692.

[16] Reeves DM, Wellman MP. Computing best response strategies in infinite games of incomplete information. In: Proc. of the 20th Conf. on Uncertainty in Artificial Intelligence. Virginia: AUAI Press, 2004. 470−478.

[17] Rasmusen E, Wrote; Wang H, Bai JH, Wu RH, Trans. Games and Information: An Introduction to Game Theory. 2nd ed., Beijing: Beijing University Press, 2003 (in Chinese).

[18] Brown GW. Iterative solution of games by fictitious play. In: Koopmans TC, ed. Activity Analysis of Production and Allocation. New York: John Wiley and Sons, 1951. 374−376.

附中文参考文献:

[4] 张双民,石纯一.一种基于特征向量提取的 FMDP 模型求解方法.软件学报,2005,16(5):733−743. http://www.jos.org.cn/1000-9825/16/733.htm

[10] 高阳,周志华,何加洲,陈世福.基于 Markov 对策的多 Agent 强化学习模型及算法研究.计算机研究与发展,2000,37(3):257−263.

[17] Rasmusen E,著;王晖,白金辉,吴任昊,译.博弈与信息:博弈论概论.北京:北京大学出版社,2003.

肖正(1981−),男,湖南怀化人,博士生,主要研究领域为计算机网络,分布式系统,多 Agent 系统.

张世永(1950−),男,教授,博士生导师,CCF 高级会员,主要研究领域为计算机网络,信息安全,无线通信,移动计算.