一种最大集合期望损失的多目标 Sarsa(λ)算法

刘 全[1,2],李 瑾[1],傅启明[1],崔志明[1],伏玉琛[1]
(1.苏州大学计算机与科学学院,江苏苏州 215000;2.符号计算与知识工程教育部重点实验室(吉林大学),吉林长春 130012)

摘 要: 针对 RoboCup 这一典型的多目标强化学习问题,提出一种基于最大集合期望损失的多目标强化学习算法 LRGM-Sarsa(λ)算法.该算法预估各个目标的最大集合期望损失,在平衡各个目标的前提下选择最佳联合动作以产生最优联合策略.在单个目标训练的过程中,采用基于改进 MSBR 误差函数的 Sarsa(λ)算法,并对动作选择概率函数和步长参数进行优化,解决了强化学习在使用非线性函数泛化时,算法不稳定、不收敛的问题.将该算法应用到 RoboCup 射门局部策略训练中,取得了较好的效果,表明该学习算法的有效性.

关键词: 多目标;自适应 Sarsa (λ);最大集合期望损失;强化学习;机器人足球
中图分类号: TP181 **文献标识码:** A **文章编号:** 0372-2112 (2013) 08-1469-05
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2013.08.003

A Multiple-Goal Sarsa(λ) Algorithm Based on Lost Reward of Greatest Mass

LIU Quan[1,2], LI Jin[1], FU Qi-ming[1], CUI Zhi-ming[1], FU Yu-chen[1]
(1. *Institute of Computer Science and Technology, Soochow University, Suzhou, Jiangsu* 215006, *China*;
2. *Key Laboratory of Symbol Computation and Knowledge Engineering (Jilin University), Ministry of Education, Changchun, Jilin* 130012, *China*)

Abstract: For solving the multiple-goal problem in RoboCup, a novel multiple-goal Reinforcement Learning algorithm, named LRGM-Sarsa(λ), is proposed. The algorithm estimates the lost reward of the greatest mass of every sub goal and trades off the long term reward of the sub goals to get a composite policy. In the single learning module, B error function, which is based on MSBR error function is proposed. B error function has guaranteed the convergence of the value prediction with the non-linear function approximation. The probability funciton of selecting actions and the parameter α are also improved with respect to B error function. This algorithm is applied to the training of shooting in Robocup 2D. The experimental results show that the proposed algorithm is more stable and converges faster.

Key words: multiple-goal; adaptive Sarsa(λ); lost reward of greatest mass; reinforcement learning; robocup 2D

1 引言

多目标强化学习是强化学习研究中非常重要的研究方向之一,被广泛应用到各个领域,例如游戏、网络路由、无线传感网络以及机器人设计等.由于单个 agent 只能感知自身的行为,会产生状态的部分感知,即 POMDP 问题.多个目标之间,如何协调通信以达到整体目标最优是目前的研究热点之一[1~6].

解决多目标强化学习问题的核心思想是每个 MDP 模型都拥有自己的学习模块.Karlsson 提出了基于最大集合的 GM-Q 算法,将所有学习模块的 Q 值求和,在最大集合中选择具有最大和值的动作来执行[7].Humphrys 认为具有最大和值的动作,对所有的学习模块来说不一定是最好的,有可能导致所有的学习模块都无法到达自己的目标.并提出了 winner-take-all 法,要求每次选择的至少是一个学习模块的最优动作,即在所有学习模块的最优动作集合中进行选择[8].但是该算法也存在缺陷,具有最大 Q 值的模块所选择的动作不一定是最优的,而其他模块因为没有执行自己的最优动作而会丢失很多有用的信息.该算法有时会有较好的性能,但是过于依赖奖赏函数[8].

文献[9,10]对算法探索时使用的动作选择概率函数进行了改进,提高了算法的收敛速度和收敛效果.文献[11]提出了一种新的多智能体 Q 学习算法,通过联合动作的统计学习其他智能体的行为策略,并利用策略

收稿日期:2012-11-09;修回日期:2013-03-06;责任编辑:蓝红杰
基金项目:国家自然科学基金(No.61070223,No.61103045,No.61272005,No.61170020);江苏省自然科学基金(No.BK2012616);江苏省高校自然科学研究项目(No.09KJA520002,No.09KJB520012);吉林大学符号计算与知识工程教育部重点实验室项目(No.93K172012K04)

向量的全概率分布保证了对联合最优动作的选择.文献[12]针对石油生产日常维护作业调度问题,提出了 MCSR 结构,采用经典的 GM-Q 算法,加入预测机制和协作通信,使算法有较强的有效性和实用性.文献[13,14]证明了满足一定约束条件的多目标强化学习算法能够保证收敛到最优值.但是在实际问题的求解过程中,较难满足这些约束条件,算法的收敛性还是没有从理论上得到根本保证.

本文提出了一种新的多目标强化学习算法(Lost-Reward-based Greatest Mass Sarsa(λ),LRGM-Sarsa(λ)).该算法预估各个目标的最大集合期望损失,在平衡协调的前提下选择动作,产生最优联合策略.在单个学习模块的函数逼近过程中,优化了误差函数、动作选择概率函数和步长参数,使得函数逼近过程趋于稳定,加快了收敛速度,并且从理论上证明算法的收敛性.将 LRGM-Sarsa(λ)应用到 RoboCup 局部射门策略训练中,实验表明该算法具有较快的收敛速度和较好的收敛效果.

2 LRGM-Sarsa(λ)算法

2.1 多目标问题形式化

强化学习问题本质上可以形式化为 MDP 模型.MDP 模型可以用一个四元组(S,A,T,R)表示:S 为可能的状态集合,A 为可能的动作集合,$T: S\times A\to T$ 是状态转移函数,$R: S\times A\to R$ 是奖赏函数.在每一个时间步 k,环境处于状态集合 S 中的某个状态 s,agent 选择动作集合 A 中的一个动作 a,收到立即奖赏 r,并转移至下一状态 s'.状态转移函数 $T(s,a,s')$表示在状态 s 执行动作 a 转移到状态 s'的概率,可以用 $P^a_{ss'}$表示.而奖赏函数 $R(s,a)$表示在状态 s 执行动作 a 之后所获得的立即奖赏值.Agent 目标就是寻求一个最优控制策略 π^*,使得长期折扣奖赏回报最大.

定义 1 N 个相互关联的 MDP 模型集合表示为 $\{M_i\}_1^N$——MDP 联合模型.M_i 表示第 i 个 MDP 模型.

单个 MDP 模型都有自己的状态集合,但是所有的 MDP 模型共享一个动作集合,并且在每个时间步都要执行相同的动作.联合模型的目的是让单个 agent 同时面临不同的子目标.这个 MDP 联合模型形式化的表示了一个多目标 MDP 模型,最终目标是要寻找多目标模型的最优控制策略,使得 N 个 MDP 模型中的折扣奖赏回报总和最大.

MDP 联合模型的状态集合是所有单个 MDP 模型的状态集合的笛卡尔集,即 $S=S_1\times S_2\times\cdots\times S_N$.而联合奖赏函数定义为:$R(s,a)=\sum_{i=1}^{N}R_i(s_i,a)$.在联合模型中,每个 MDP 模型都是独立的,所以联合状态转移函数定义为:$T(s,a,s')=\prod_{i=1}^{N}T_i(s_i,a,s_i')$.

2.2 算法描述

针对 Humphrys 提出的 GM-Q 算法中存在的问题,Sprague 提出了 W-Learning 算法[15],W-Learning 算法是一种基于平衡解思想的方法,算法思想是在平衡各个目标的前提下得到问题的求解.

本文根据文献[15]提出了一种新的多目标强化学习算法 LRGM-Sarsa(λ).该算法利用 G 值函数对各个模块进行动作选择,在每个学习模块中利用公式(1)进行 G 值学习:

$$G_i(s_i)\leftarrow\max_a Q_i(s_i,a)-(r_i+\gamma\max_b Q_i(s_i',b)) \quad (1)$$

其中,s_i'是其他模块在执行动作 a 之后转移到的状态,$Q_i(s_i,a)$表示在状态 s_i 执行动作 a,及采取后续策略的折扣奖赏和的期望.

根据公式(1),在每个时间步具有最高 G 值的模块执行其所选择的动作,而其它没有被选中的模块则更新其 G 值.在给定状态下,没有被选中的模块就可以根据公式(1)预测将要损失的长期奖赏回报,而在下次选择动作时,损失最大的模块将具有动作优先选择权.

而与 W-Learning 算法最大的不同是 LRGM-Sarsa(λ)算法使用了资格迹(eligibility traces),以此来提高算法的效率,并且在设计算法时,不只是单步的计算过程得到每一个状态下的最优动作,而是设计了整体的算法流程得到求解问题的最终策略.在加入资格迹时,选取替代迹以加快算法收敛速度.同时,本文选择了在策略的 Sarsa 算法,Sarsa 算法使用的是执行动作后实际得到的 Q 值.在联合策略控制下,每个使用 Sarsa 算法的学习模块所得到的 Q 值比使用 Q 学习得到的 Q 值更接近真实的期望回报.假设对每个模块的估计都是真实可靠的,那么算法就能够从各个模块中选择出使得回报总和最大的动作,这样就能保证整体算法的收敛性.

LRGM-Sarsa(λ)算法描述如下:

LRGM-Sarsa(λ)算法

Step1:对于起始状态 s,任意选择一个模块作为优先选择模块 l 进行初始化,$G_l\leftarrow 0$,$a_l\leftarrow\text{argmax}_a Q_l(s_l,a)$,$i\leftarrow 0$.

Step2:对每个模块内部进行 Q 值更新,针对状态动作对(s,a)得到下一状态 s',立即奖惩值 r.根据动作选择概率函数选择状态 s'的动作 a'.

Step3:每个模块内部更新资格迹 $e(s,a)$:$\delta\leftarrow r+\gamma Q(s',a')-Q(s,a)$,$e(s,a)\leftarrow e(s,a)+1$.

Step4:对于每个模块内的所有状态动作对(s,a)进行更新:$Q(s,a)\leftarrow Q(s,a)+\alpha\delta e(s,a)$,$e(s,a)\leftarrow\gamma\lambda e(s,a)$.

Step5:除模块 l 外,对模块 i 根据公式(1)计算 G_i 值.

Step6:判断如果 $G_i>G_l$,则转至 Step7;否则转至 Step8.

Step7:更新最优模块 l,$G_l\leftarrow G_i$,$a_l\leftarrow\text{argmax}_a Q_i(s_i,a)$,$l\leftarrow i$,转至 Step6.

Step8:判断除模块 l 外其余模块 i 的 G_i 值是否都更新,若是则获得状态 s_l 的最优动作 a_l,即最优动作状态对(s_l,a_l),转至 Step9;否则,$i\leftarrow i+1$ 转至 Step5.

Step9:判断策略 π 中所有最优动作状态对(s_l,a_l)是否改变,若是则转至 Step10;否则转至 Step11.

Step10:$s \leftarrow s'$,$a \leftarrow a'$,判断状态 s 是否为终止状态,若是则转至 Step1;否则转至 Step2.

Step11:算法终止,产生最优策略 π^*.

3　单个模块的强化学习算法

为了解决单个模块的学习过程中使用非线性函数泛化空间时可能导致算法不收敛的问题,本文采用误差函数——Bellman 剩余均方误差(Mean-Squared Bellman Residual,MSBR)来进行函数逼近.设 $E(s_t)$ 表示 MSBR 误差函数,如式(2)所示.依据 MSBR 误差函数,近似函数权重 W 的调整仍使用梯度下降方法,如式(3)所示.

$$E(s_t) = \frac{1}{n}\sum_s [V^\pi(s_t) - V_t(s_t)]^2 \tag{2}$$

$$\Delta w_r = \alpha[V^\pi(s_t) - V_t(s_t)][\nabla_w V^\pi(s_t) - \nabla_w V_t(s_t)] \tag{3}$$

其中,$V^\pi(s_t)$表示状态 s_t 的值,是指 agent 在状态 s_t 根据策略 π 执行动作 a_t 及采取后续策略所得到的积累奖赏的期望,$V_t(s_t)$则表示使用近似函数逼近得到的 V 值.

对于状态数量有限的 MDP 模型来说,当 $E(s_t)$趋于零时,值函数趋于最优.使用梯度下降方法至少能保证 $E(s_t)$收敛到局部最小值,算法也能保证收敛.但是,MSBR 误差函数也有缺陷,使用 MSBR 误差收敛有一个前提假设,就是训练样本分布是不变的,这个前提假设是很多实际问题不具备的.策略更新大多使用贪心方法,从而导致训练分布是实时变化的.为了解决这一问题,本文结合 MSBR 误差函数采用 B 误差函数[16].

考虑一个单独 MDP 模型,在一个随机策略下产生一个训练片段.

定义 2　设 $u_t = \{s_0, a_0, R_0; s_1, a_1, R_1; \cdots; s_{t-1}, a_{t-1}, R_{t-1}; s_t, a_t, R_t\}$表示从起始状态 s_0,经过时间步 t 所产生的训练片段.其中在状态 s_i 下选择动作 a_i 产生回报值 R_i,并且转移到状态 s_{i+1}.

随机策略可以用权重向量为 W 的函数表示.假设 MDP 模型的起始状态为 s_0,如果 MDP 模型有终结状态,并且 s_i 为终结状态之一,那么 $s_{i+1} = s_0$.

定义 3　设 U_t 表示从时刻 0 到时刻 t 的所有可能的训练片段.

使用 $E(u_t)$表示在 t 时刻的 MSBR 误差.$P(end \mid u_t)$表示从 0 时刻开始,在片段 u_t 发生的前提下,到达终结状态的概率,即在一个完整的情节(episode)训练中,同时训练长度有限的前提下,产生 u_t 片段的概率.

定义 4　设 B 表示在完整训练中总的期望误差.此处的期望值是由给定策略下的状态访问频率加权获得的.B 误差函数公式如式(4)所示.

$$B = \sum_{T=0}^{\infty}\sum_{u_T \in U_T} P(\text{episode ends at time } T \text{ after trajectory } u_T) \cdot \sum_{i=0}^{T} E(u_i) = \sum_{t=0}^{\infty}\sum_{u_t \in U_t} E(u_t)P(u_t) \tag{4}$$

其中,$P(u_t)$如式(5)展开.

$$P(u_t) = P(a_t \mid u_t)\cdot P(R_t \mid u_t)\prod_{i=0}^{t-1} P(a_t \mid u_i)\cdot P(R_i \mid u_i)\cdot P(u_{i+1} \mid u_i)\cdot[1 - P(end \mid u_i)] \tag{5}$$

式(5)是 u_t 产生的概率,$P(a_i \mid s_i)$可以看作以权重向量 $\boldsymbol{W}$ 为变量的函数,且是以权重向量 $\boldsymbol{W}$ 为变量的、非零的平滑函数.B 误差对于 $\boldsymbol{W}$ 的偏导数如式(6)所示.式(6)给出了完整训练情节下总的期望误差 B 的无偏估计,对 B 误差函数使用随机梯度下降方法,更新近似函数的权重,使得 B 误差趋近于 0.

$E_{SARSA}(u_t)$误差函数和权重 $\boldsymbol{W}$ 调整分别如式(7)、式(8)所示.在式(7)和式(8)中,$E_{SARSA}(u_t)$函数和 $P(a_t \mid u_t)$都必须是关于权重 $\boldsymbol{W}$ 的平滑函数,而且保证 $E_{SARSA}(u_t)$函数是有界函数.如果是在当前策略下选择动作,从而得到训练片段,那么 Δw 的总和就是实际梯度的无偏估计.在学习过程中,如果在每次学习之后进行权重更新,同时权重 $\boldsymbol{W}$ 是在有限区域内的,并且学习步长逐步趋近于 0,那么 B 误差函数以概率 1 收敛.

$$\begin{aligned}\frac{\partial}{\partial w}B &= \sum_{t=0}^{\infty}\sum_{u_t \in Ut}[(\frac{\partial}{\partial w}E(u_t))P(u_t) + E(u_t)P(u_t) \cdot \sum_{j=1}^{t}\frac{\frac{\partial}{\partial w}[P(a_{j-1} \mid u_{j-1})]}{P(a_{j-1} \mid u_{j-1})}] \\ &= \sum_{t=0}^{\infty}\sum_{u_t \in U_t} P(u_g)[\frac{\partial}{\partial w}E(u_t) + E(u_g) \cdot \sum_{j=1}^{t}\frac{\partial}{\partial w}\ln(P(a_{j-1} \mid u_{j-1}))]\end{aligned} \tag{6}$$

$$E_{SARSA}(u_t) = \frac{1}{2}E^2[R_{t-1} + \gamma Q(s_t, a_t) - Q(s_{t-1}, a_{t-1})] \tag{7}$$

$$\Delta w_t = -\alpha[\frac{\partial}{\partial w}E(u_t) + E(u_t)\frac{\partial}{\partial w}\ln(P(a_{t-1} \mid u_{t-1}))] \tag{8}$$

4　LRGM-Sarsa(λ)算法在 RoboCup 射门策略中的应用

RoboCup 机器人仿真足球比赛是 MAS 系统研究的标准问题,同时包含了动态、实时、不确定环境中的合作和对抗.本文考虑的是局部战术的学习任务——射门局部策略的训练,因为球队的射门能力对球队来说是十分重要的,关系到比赛的最终成绩.

射门局部策略是一个典型的多目标问题,常见的多目标问题均属于 MAS 系统,由多个 agent 完成多个目标的实现.射门局部策略同样是基于 MAS 系统的多目标问题,因为在考虑射门的同时,还要防止对方后卫截

球,保持球权在己方,提高己方控球时间以保持对敌方的压迫,显然这两个目标的实现靠单个球员是很难实现的.射门局部策略的训练包括攻守两方,进攻方为两名前锋球员,防守方为一名后卫和一名守门员,共有四名队员进行 2 对 2 训练.射门局部策略可视为如何选择最优动作以实现多目标的策略.对于射门决策问题,进攻方的控球队员的动作选择如下:

Shoot():射门,尝试直接射门得分;*Dribble*():带球,控球移动寻找机会;*Pass*():传球,将球传给队友,制造射门机会;*Hold*():持球,持球静止并尽可能远离对手;*GotoP*():跑位,跑到一些特定点,比如能够传球或者接到队友传球的点.

而防守方队员的动作则采用固定策略,跑向球(*GotoBall*()),然后截球(*Intercep*()).防守方的守门员也是采用固定策略,首先根据进攻方位置预估射门路线进行站位调整(*GotoP*()),然后对于射门选择扑球动作(*SaveBall*()).

在射门局部策略的训练中,我们主要训练的是进攻方的射门能力,而进攻方队员的动作集合一致,表示为 $a\in\{Shoot(), Dribble(), Pass(), Hold(), GotoP()\}$.

状态描述考虑需要表示球员的位置和身体的朝向.所以每个球员使用三个变量表示,一共需要 12 个变量.训练场景如图 1 所示.状态描述时以球门中心点 *C* 为参照,简化状态描述.同时,在训练中使用 *BP* 神经网络作为近似函数泛化 $(s_t, a_t)\rightarrow Q(s_t, a_t)$.

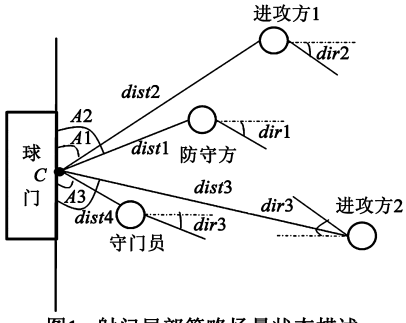

图1 射门局部策略场景状态描述

训练中动作奖赏值设定如下:对于带球动作,如果带球成功突破对手并在半径 1.5 米内没有对手,则给予奖赏值 1;对于传球动作,如果队友成功接到球并在其半径 1.5 米内没有对手,则给予奖赏值 1;对于射门动作,射门成功则给予奖赏值 2;其它动作,奖赏值均为 0.训练场景起始时,球在进攻方手中并且从球场边路带球.如果进攻方射门成功,或者被守方得球,或者球出界,或者训练时间超过 100 个仿真周期,则训练结束.

实验中,参数 $\varepsilon=0.05$, $\tau=0.1$, $\beta=3.5$.图 2 是不同算法对失误动作数量的比较,失误动作指失败的传球、带球和射门.图 2 中纵坐标为 100 次试验的平均失误的动作数量,横坐标为学习步数,每 200 步计算一次平均失误动作数.

本文将 LRGM-Sarsa(λ)算法分别与文献[11]New Q-Learning 算法和文献[12]的 MCSR-Sarsa 算法进行了比较.从图 2 中可以看到,在收敛速度上,本文算法快于文献[11]和文献[12]的算法;从图 3 中可以看到在收敛稳定性上,本文算法要优于文献[11]和文献[12]的算法.从图 2 中可以看到本文算法整体的收敛曲线最为平滑,同时从图 3 中可知在学习步数 2650 左右,本文算法就基本收敛到最优值,之后的收敛曲线也很稳定,相对的,文献[11]和文献[12]的算法的收敛曲线在收敛之后还是有一定的波动.可以说本文算法在收敛速度和收敛的稳定性上优于文献[11]和文献[12]的算法.

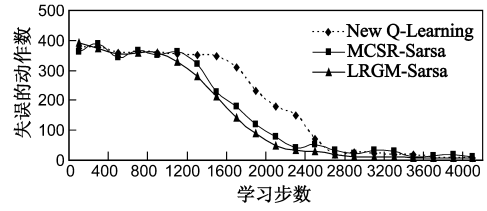

图2 整个训练过程中不同算法失误动作数量比较

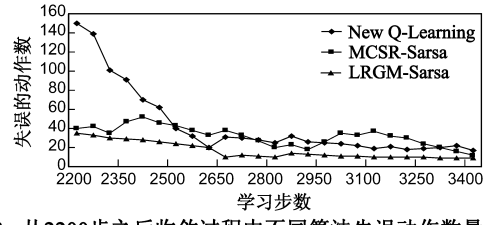

图3 从2200步之后收敛过程中不同算法失误动作数量比较

图 4 为 3 种算法在每次情节训练后所得到的平均奖惩值的比较.从图 4 中可知,文献[11]算法收敛后的平均奖惩值约为 1.49,文献[12]算法收敛后的平均奖惩值约为 1.72,而 LRGM-Sarsa(λ)算法收敛后的平均奖惩值约为 1.8,在 3 种算法中 LRGM-Sarsa(λ)算法获得的平均奖惩值最高.同时,从图 4 中也可以看到 LRGM-Sarsa(λ)算法的收敛性能优于另两种算法.

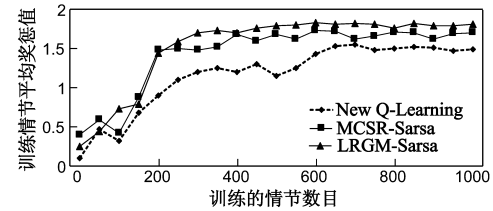

图4 不同算法的平均奖惩值比较

表 1 是不同算法的多目标性能比较,包括进球个数和训练场景的持续时间,即攻方的持球时间比较.表 1 的数据是在算法收敛训练结束后,继续运行训练场景

100 次获得的.进球数目是 100 次累积进球数,持球时间是取 100 次持球时间的平均值,以 RoboCup 仿真周期为单位,一个仿真周期为 100ms.表 2 是进行过射门局部策略训练的球队与未进行过训练的球队对抗结果数据.

5　结论

本文针对 RoboCup 这一多目标问题,提出一种新的多目标强化学习算法 LRGM-Sarsa(λ).该算法利用最大集合概念,预估各个目标的期望损失,在平衡协调的前提下对联合动作进行选择以产生最优联合策略.在单个目标的训练过程,优化 MSBR 误差函数、动作选择概率函数和步长参数,解决了使用非线性函数近似泛化时,算法不稳定、不收敛的问题.在实验中,将该算法应用到 RoboCup 射门局部策略训练中,取得了较好的效果,实验结果表明,该学习算法具有较快的收敛速度和较好的收敛效果.

由于 RoboCup 这一问题的复杂性和强化学习算法自身的缺陷,在很多方面可以做进一步的研究.本文算法在局部策略训练中有较好的效果,而在全局策略训练中算法的通信协作机制仍需改进以适应多 agent 训练;在训练中,敌方采用固定策略以简化问题模型,而当敌方不采用固定策略时,算法中如何对敌方策略进行预判,以加强球队对抗能力也是待解决的问题.

参考文献

[1] Feng Wu, Shlomo Zhilberstein, Xiaoping Chen. Online planning for multi-agent systems with bounded communication[J]. Artifical Intelligence, 2011, 175(2): 487 – 511.

[2] Zongzhang Zhang, Xiaoping Chen. Accelerating point-based POMDP algorithms via greedy strategies[A]. Proceedings of the 2nd International Conference on Simulation, Modeling and Programming for Autonomous Robots[C]. Darmstadt, Germany: Computer Science, 2010. 545 – 556.

[3] Feng Wu, Shlomo Zilberstein, XiaoPing Chen. Multi-agent online planning with communication[A]. Proceedings of ICAPS-09[C]. Thessaloniki, Greece: AAAI, 2009.

[4] Constantin A, Dana H. Credit assignment in multiple goal embodied visuomotor behavior[J]. Frontiers in Psychology, 2010, 173(1): 1 – 13.

[5] Marek Grzes, Daniel Kuadenko. Multigrid reinforcement learning with reward shaping[A]. Proceedings of ICANN 2008[C]. Verlag, Berlin: Springer, 2008. 357 – 366.

[6] 王雪松,张依阳,程玉虎.基于高斯过程分类器的连续空间强化学习[J].电子学报,2009,39(6):1153 – 1158.
Wang Xue-song, et al. Reinforcement learning for continuous spaces based on Gaussian process classifier[J]. Acta Electronic Sinica, 2009, 39(6): 1153 – 1158. (in Chinese)

[7] Karlsson J. Learning to Solve Multiple Goals[D]. Rochester: University of Rochester, 1997.

[8] Humphrys M. Action selection methods using reinforcement learning[A]. Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior[C]. Cambridge, MA: MIT, 1996. 135 – 144.

[9] Daw N, Doherty J, et al. Cortical substrates for exploratory decisions in humans[J]. Nature, 2006, 441(7095): 876 – 879.

[10] Rangel A, Hare T. Neural computations associated with goal-directed choice[J]. Current opinion in neurobiology, 2010, 20(6): 262 – 270.

[11] 郭锐,吴敏,彭军,彭娇,曹卫华.一种新的多智能体 Q 学习算法[J].自动化学报,2007,33(4):367 – 372.
Guo Rui, Wu Min, et al. A new multi-agent Q-learning[J]. Acta Automatica Sinica, 2007, 33(4): 367 – 372. (in Chinese)

[12] Aissani N, Beldjilali B, Beldjilali B. Dynamic scheduling of maintenance tasks in the petroleum industry: A reinforcement approach[J]. Engineering Applications of Artificial Intelligence, 2009, 22(7): 1083 – 1103.

[13] Sprague N, Ballard D, Robinson A. Modeling embodied visual behaviors[J]. ACM Transactions on Applied Perception, 2007, 4(2): 1 – 23.

[14] Sprague N, Ballard D, Robinson A. Modeling embodied visual behaviors[J]. ACM Transactions on Applied Perception, 2007, 4(2): 1 – 23.

[15] Ana M, Timothy C, Hal P. Taxing executive processes does not necessarily increase impulsive decision making[J]. Experimental Psychology, 2010, 57(3): 193 – 201.

[16] Baird C. Residual algorithms: reinforcement learning with function approximation[A]. Proceedings of the Twelfth International Conference of Machine Learning[C]. San Francisco, CA: Morgan Kaufman, 1995.

作者简介

刘　全　男,1969 年生于内蒙古牙克石,博士,教授,博士生导师.主要研究方向为强化学习、无线传感器网络、智能信息处理.
E-mail: quanliu@suda.edu.cn

李　瑾　女,1986 年生于江苏苏州,硕士,主要研究方向为强化学习和仿真机器人足球.