第38卷 第9期 北 京 工 业 大 学 学 报 Vol. 38 No. 9
2012年 9月 JOURNAL OF BEIJING UNIVERSITY OF TECHNOLOGY Sep. 2012

基于CMAC网络Sarsa(λ)学习的RoboCup守门员策略

刘云龙，吉国力
(厦门大学 自动化系，厦门 361005)

摘 要：针对RoboCup仿真组足球比赛场上状态复杂多变、同时供决策的信息大多为连续变量、智能体利用现有信息通常无法判断当前状态下最优动作的问题，以守门员为例，首先利用CMAC神经网络对连续状态空间泛化，然后在泛化后的状态上，采用Sarsa(λ)学习算法获取守门员的最优策略. 通过在RoboCup仿真平台上进行仿真，实验结果表明，采用基于CMAC的Sarsa(λ)学习算法的守门员，经过一定时间的学习后，防守时间显著增长，防守效果明显优于其他算法，验证了本文所提方案的有效性.

关键词：RoboCup仿真组足球比赛；CMAC神经网络；泛化；Sarsa(λ)学习算法；最优策略
中图分类号：TP 181 **文献标志码**：A **文章编号**：0254－0037(2012)09－1348－05

CMAC-based Sarsa(λ) Learning Algorithm for RoboCup-soccer Goalkeeper

LIU Yun-long, JI Guo-li
(Department of Automation, Xiamen University, Xiamen 361005, China)

Abstract: RoboCup simulated soccer has a large and complex state space, at the same time the variables used for decision are usually continuous, that make it difficult for the agent to choose the optimal action. This paper presents the goalkeeper as a case study, based on CMAC neural network, the continuous state space is firstly generalized, and then the Sarsa (λ) learning algorithm is employed to find the optimal policy. The author empirically evaluated and compared the defending effect of the goalkeepers with different strategies. Simulation results show that the goalkeeper with the learning algorithm has better defending effect and its defending time increases obviously after a period of time.

Key words: RoboCup simulated soccer; CMAC neural networks; generalization; Sarsa (λ) learning algorithm; optimal policy

RoboCup仿真组机器人足球比赛是一个完全分布式控制、实时异步的多智能体环境. 作为当前人工智能研究的热点，该环境模拟了真实的机器人足球比赛，场上状态复杂多变，智能体的感知信息和动作执行效果均存在噪声，从而为人工智能和机器学习领域的研究提供了绝佳的实验平台[1].

RoboCup仿真组比赛中，供决策的信息大多为连续变量，状态空间复杂，难以用人为方式确定场上每个状态并选择每个状态对应的动作；同时，智能体的感知和动作是异步的，传统的利用感知到的信息直接选择采取的动作往往无法达到预期效果. 以守门员为例，守门员在机器人足球比赛中扮演着重要的角色. 传统的守门员策略通常是通过预编码的方式来确定什么态势下应该采取什么样的动作，这类

收稿日期：2010-07-20.
基金项目：福建省自然科学基金资助项目(2010J05140)；高等学校博士学科点专项科研基金资助项目(20100121120022).
作者简介：刘云龙(1977—)，男，讲师，主要从事强化学习、预测状态表示、智能决策等方面的研究，E-mail：ylliu@xmu.edu.cn.

方法实现简单、对某些特定态势能有效地防守，例如 Freiburg 队的守门员策略[2]. 但如上所述，机器人足球比赛环境状态空间复杂，供决策的信息是连续变量，仅靠预编写代码的方式确定当前状态并采取相应的动作，往往无法包含所有可能发生的状态. 而在未考虑到的状态上，不容易判断采取何种动作才能最有效地防守，导致此类算法缺乏灵活性和自适应能力. 因此，使得守门员具有学习能力成为解决此类问题很好的途径.

CMAC 神经网络是在 1972 年由 Albus 提出[3]，仿照小脑如何控制肢体运动的原理而建立的神经网络模型，可以有效地对状态空间泛化. 该模型有固有的局部泛化能力，即在输入空间中相近的输入向量给出相近的输出. 即使不针对输入进行训练，只要输入落入该状态空间范围，输出就保持相同. 而在诸多学习方法中，强化学习[4]由于不需要环境模型，同时不需要专门的训练样本以及可以通过和环境的动态交互在线学习相应的策略，能较好地满足机器人足球比赛这种环境下的要求，近年来在机器人足球比赛中得到了广泛的应用[5-7].

在本文中，首先利用 CMAC 神经网络对连续状态空间泛化，然后在泛化后的状态上，采用强化学习算法中的 Sara(λ) 学习算法获取守门员的最优策略，并在 RoboCup 仿真平台上进行仿真.

1　守门员防守策略

守门员在球场上的位置可参考图 1，其中：$R_A(x_A, y_A, v_A)$ 表示进攻方机器人；$B(x_B, y_B, v_B)$ 表示球；$R_G(x_G, y_G, v_G)$ 表示本方守门员；x、y、v 分别表示横坐标、纵坐标和速度. 球门线的横坐标为 G，机器人颜色较浅的一面朝向机器人前进的方向. 在当前时刻 t，机器人 R_A 有多种选择，可以选择以不同的角度带球前进或者射门. 如果射门，当 R_A 踢球后，球将做直线运动，在 t' 时刻到达球门，与球门线交于 $K(G, y_K)$. 此时对于机器人 R_G 而言，可以选择以下 3 种动作中的一种.

1) 出击：根据球的速度、朝向，守门员本身的速度、朝向，守门员在球的运动轨线上找一个点 N，使得守门员移动到点 N 的时间不大于球滚动到点 N 的时间，守门员移动到点 N 去抢球.

2) 接球：根据球的速度、朝向，判断球与本方球门线的交点，如图 1 中的 K 点，守门员移动到这个交点——防守点进行防守.

3) 等待：守门员移动到本方球门线的中点——$M(G, 0)$.

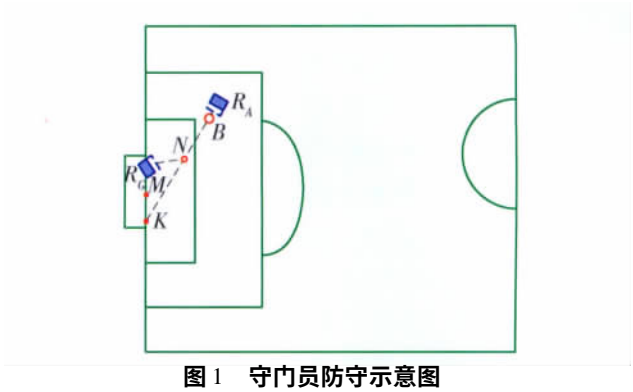

图 1　守门员防守示意图

Fig. 1　Defending diagram of the goalkeeper

假定机器人 R_A、R_G、球 B 做的均是匀速直线运动，如果 R_G 选择接球，要保证防守成功，需满足

$$R_G K / v_G \leqslant t' - t \tag{1}$$

这样才能保证在球到达防守点之前，守门员已经到达防守点防守. 如果 R_G 选择出击，假定 N 为抢球点，要保证防守成功，需满足

$$t_\theta + R_G N / v_G \leqslant BN / v_B \tag{2}$$

式中 t_θ 表示的是 R_G 抢球过程中转向需要的时间.

在当前态势下，如果 R_G 一直选择等待，显然不能成功防守，但如果 R_G 在对方射门之前选择等待，则更有可能满足式(1)，使得防守成功率增加。如果 R_A 在当前时刻选择带球，由于在带球的过程中，R_A 可能不停地变换带球前进的方向，使得 R_G 难以确定抢球点的确切位置，并且如果 R_G 出击过早，在不能抢到球的情况下，会导致 R_A 带球过了 R_G 后直接面对空门. 如果出击过晚，R_A 可以从容地选择射门时机和角度，守门员 R_G 即使判断对了球的运动轨迹，也很难满足式(1)或式(2)，导致失球. 显然对于出击时机的把握和不出击时选择接球还是等待决定了防守的成功率.

在本文中，通过将 CMAC 神经网络和 Sarsa(λ) 学习算法引入守门员的策略选择中，学习确定守门员在某一时刻应该采取的动作，提高了守门员的防守成功率.

2　基于 CMAC 网络 Sarsa(λ) 学习的守门员策略

强化学习算法中的状态评估函数 $Q(s, a)$ 被表示为一个显式的查找表时，只有每个可能的状态-动作对被无限频繁地访问，学习过程才会收敛. 在状态空间和动作空间都不是很大的情况下，这种方法得到了很好的应用. 但对于机器人足球比赛而言，

状态空间复杂多变,供决策的信息是连续变量,显然无法用 $Q(s,a)$ 表示所有可能的状态. 同时,如果状态太多,采用查找表的方式,$Q(s,a)$ 不但要占用大量的内存空间,在学习的过程中更新这个表格也变得很困难. 为了克服这些缺陷,本文采用 CMAC 神经网络对状态空间泛化,将表示当前状态的状态变量作为 CMAC 神经网络的输入,从而相近的输入向量有相近的输出,并进一步将 CMAC 神经网络和强化学习算法相结合,实现守门员动作的自动选择. 其中,CMAC 神经网络的作用在于将输入的状态变量影射到对应的地址,具体地址中的权值更新则是通过 Sarsa(λ) 学习方法完成.

2.1 Sarsa(λ)学习算法

Sarsa(λ) 学习算法是一种在线强化学习方法,可在学习过程中,根据智能体当前执行的策略更新 $Q(s,a)$. 其中,$Q(s,a)$ 是状态评估函数,指的是在状态 s 执行动作 a 可获取的奖励值. 这里假定 $Q(s,a)$ 被表示为一个显式的查找表,每个不同输入值(即状态-动作对)对应一个表项. 具体学习算法如下[8].

对所有的状态 s 和动作 a,任意初始化 $Q(s,a)$,并令 $e(s,a)\leftarrow 0$.

步骤 1 初始化状态 s.

步骤 2 根据从 Q 得到的策略(例如 ε-贪婪算法),从状态 s 中选择动作 a.

步骤 3 对学习周期的每一步重复以下步骤.

1) 执行动作 a,观察立即奖励值 r 和下一个状态 s'.

2) 根据从 Q 得到的策略(例如 ε-贪婪算法),从状态 s' 中选择动作 a'.

3) $\delta\leftarrow r+\gamma Q(s',a')-Q(s,a)$.

4) $e(s,a)\leftarrow e(s,a)+1$.

5) 对于所有的状态 s、动作 a:

① $Q(s,a)\leftarrow Q(s,a)+\alpha\delta e(s,a)$.

② $e(s,a)\leftarrow\gamma\lambda e(s,a)$.

6) $s\leftarrow s'$; $a\leftarrow a'$.

7) 继续执行 1) 直到 s 为结束状态.

其中: α 为学习速率参数; γ 为折算因子,它确定了延迟回报与立即回报的相对比例; $e(s,a)$ 存储的是针对当前的奖励以前的动作应该收到的奖励的信誉度; 参数 λ 决定了应该给予以前的动作多少信誉.

2.2 基于 CMAC 神经网络的 Sarsa(λ)学习算法在守门员策略中的应用

学习的目的是每一步在出击、接球、等待 3 个动作中选择一个动作,使得本方不失球的时间最长. 在 RoboCup 仿真平台中,1 步是 100 ms,机器人在 1 步内只能执行 1 个动作. 在比赛开始的时候任意初始化存储向量 $\boldsymbol{\theta}$ 并使得踪迹向量 $\boldsymbol{e}=\boldsymbol{0}$,踪迹向量 $\boldsymbol{e}$ 中的元素和存储向量 $\boldsymbol{\theta}$ 中的元素一一对应. 然后将每个学习周期分为以下 3 个过程: 初始阶段,中间阶段,最后阶段. 其中,本文中 1 个学习周期指的是从比赛开始到对方进球. 各个阶段的具体内容如下.

1) 初始阶段

步骤 1 守门员通过传感器感知当前状态 s,作为 CMAC 神经网络的输入,将其映射到实际存储器中,得到各个动作在当前状态 s 下对应的存储向量 $\boldsymbol{\theta}$ 中的地址 F_a.

步骤 2 将每个动作对应的地址的内容相加得到当前状态下该动作的 Q 值: $Q_a=\sum_{i\in F_a}\theta(i)$.

步骤 3 根据得到的 Q 值和 ε-贪婪算法确定本步要执行的动作 a,同时令 lastSelectAction$\leftarrow a$. 将动作 a 在当前状态下对应的踪迹向量赋 1: $e(i)\leftarrow 1$,其中 $i\in F_a$; 将其他动作 $\bar{a}$ 在当前状态下对应的踪迹向量赋 0: $e(i)\leftarrow 0$,其中 $i\in F_{\bar{a}}$.

步骤 4 执行动作 a. 转入中间阶段.

2) 中间阶段

步骤 1 获取上一个动作执行后的奖励值 r,令 $\delta\leftarrow r-Q_{\text{lastSelectAction}}$.

步骤 2 将通过传感器获得的当前状态变量 s 作为 CMAC 神经网络的输入,将其映射到实际存储器中,得到各个动作在当前状态下对应的存储向量 $\boldsymbol{\theta}$ 中的地址 F_a.

步骤 3 将每个动作对应的地址的内容相加,得到当前状态下该动作的 Q 值: $Q_a=\sum_{i\in F_a}\theta(i)$,根据得到的 Q 值和 ε-贪婪算法确定本步要执行的动作 a',同时令 lastSelectAction$\leftarrow a'$.

步骤 4 更新存储向量 $\boldsymbol{\theta}$: $\delta\leftarrow\delta+\gamma Q_{\text{lastSelectAction}}$,$\boldsymbol{\theta}\leftarrow\boldsymbol{\theta}+\alpha\delta\boldsymbol{e}$. 重新计算各个动作对应的 Q 值: $Q_a\leftarrow\sum_{i\in F_a}\theta(i)$,并衰减踪迹向量 $\boldsymbol{e}$ 中所有元素: $\boldsymbol{e}\leftarrow\gamma\lambda\boldsymbol{e}$. 将动作 lastSelectAction 在当前状态下对应的踪迹向量赋 1: $e(i)\leftarrow 1$,其中 $i\in F_a$; 将其他动作 $\bar{a}$ 在当前状态下对应的踪迹向量赋 0: $e(i)\leftarrow 0$,其中 $i\in F_{\bar{a}}$.

步骤 5 执行动作 lastSelectAction.

步骤 6 射门队员进球,执行最后一步; 否则,执行步骤 1.

3) 最后阶段

获得上一个动作执行后的奖励值 r，令 $\delta \leftarrow r - Q_{\text{lastSelectAction}}$，更新存储向量 $\boldsymbol{\theta}$: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha\delta \boldsymbol{e}$.

3　实验

3.1　实验设计

在基于 UvA trilearn 队提供 RoboCup 仿真组平台[8] 上，以一对一(守门员对射门队员)的守门实验来对守门员进行学习. 在奖励值的选择上为了避免牵涉过多的人为因素，本文仅采用和目标直接相关的不失球的步数作为奖励值: 每 1 步给予 1 的奖励. 在状态变量的选择上，为了尽量减少射门策略对守门员策略的影响，CMAC 神经网络的输入采用以下 9 个和具体的射门策略无关的状态变量: 球的速率; 球和守门员的相对距离; 守门员和球的相对角度; 守门员离本方底线的相对距离; 球和射门队员的相对距离; 守门员的纵坐标; 以球为顶点的守门员、球、进攻队员的夹角; 射门队员的速率; 射门队员的纵坐标.

本文采用的 CMAC 泛化参数(generalization parameter)为 64，输入的维数为 9. Sarsa(λ) 学习中各个参数的取法如下: $\alpha = 0.125$，$\lambda = 0.9$，$\gamma = 1$，并且仅保留大于 0.001 的踪迹(trace).

为了验证经过学习后的守门员的防守效果，采用学习策略的守门员分别与采用以下几种策略的守门员在防守效果上做了比较:

1) 出击策略　始终采用出击动作;

2) 接球策略　始终采用防守动作;

3) 等待策略　始终采用等待动作;

4) 随机策略　在每一步随机采用以上 3 种动作中的一种.

几种不同的守门员策略均对应相同的射门策略，即射门队员在无球状态时，抢球获得球的控制权，获得球的控制权后，根据当前态势的不同采取不同的方式带球或者射门.

3.2　实验分析

图 2 显示了经过学习后的守门员的一次成功拦截对方射门的过程.

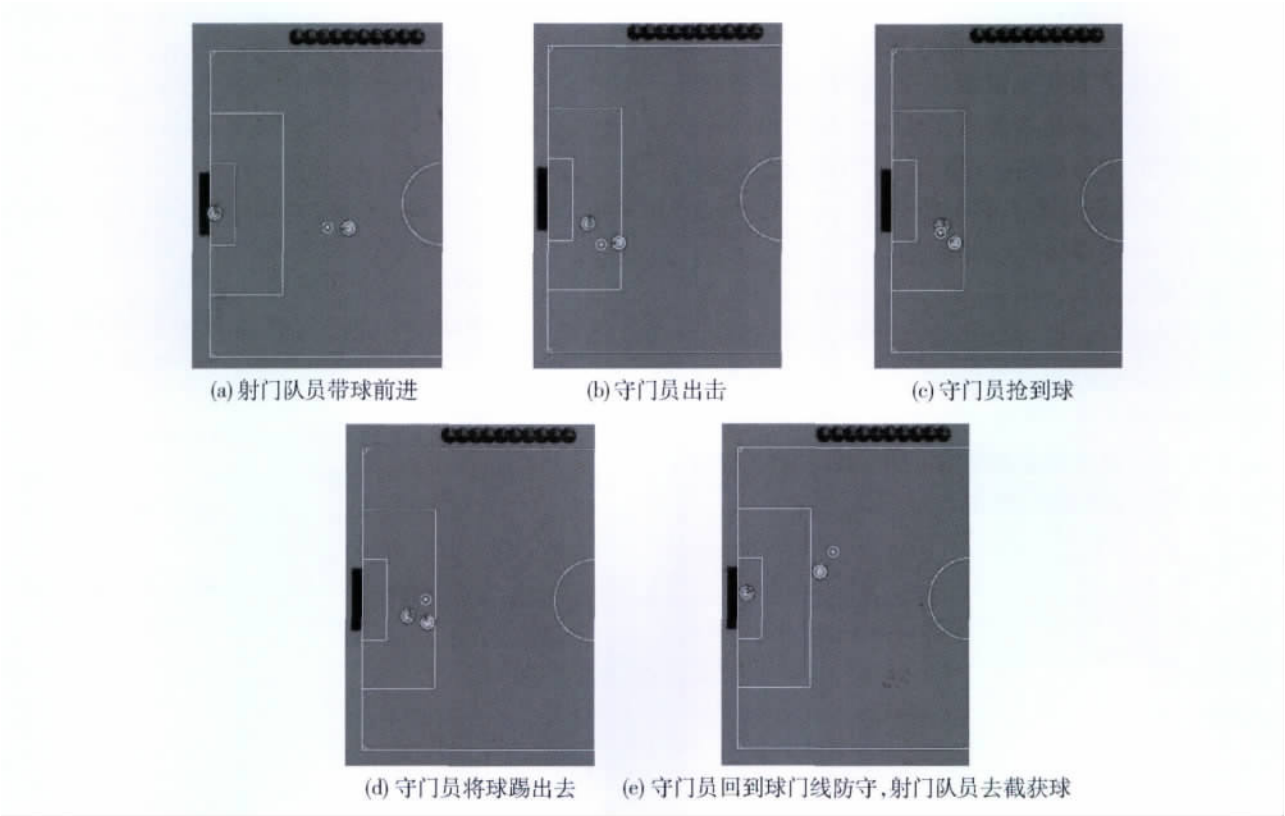

图 2　守门员防守过程

Fig. 2　Defending process of the goalkeeper

从图中可以看出，守门员准确判断了射门队员及球的运行轨迹，成功地对射门队员的射门进行了拦截. 射门队员进球后比赛将重新开始，根据防守时间的长短判断防守效果的好坏. 其中

$$T_{\text{防守时间}} = T_{\text{比赛开始时间}} - T_{\text{对方进球时间}}$$

采用不同防守策略的守门员在防守时间上的比

较如图 3 所示.

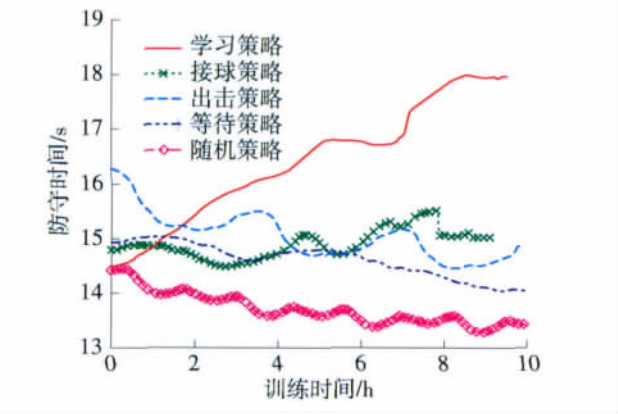

图 3 采用各个策略的守门员的防守时间

Fig. 3 Defending time of the goalkeepers with different strategies

从图 3 可以看出,经过一定时间的训练后,采用学习策略的守门员的防守时间增至 18 s 左右. 考虑到未采用学习算法时,射门队员从开球到进球所需时间通常在 14 s 左右,可以认为采用学习策略的守门员的防守时间显著增长. 同时,从图 3 可以看出,经过 2 h 左右的学习,采用学习策略的守门员的防守时间已超过采用其他策略的守门员的防守时间,并且随着训练时间的增多有继续增长的趋势,而采用其他策略的守门员在防守时间上没有什么大的变化,验证了将基于 CMAC 神经网络的 Sarsa(λ) 算法应用于守门员动作选择的有效性.

从实验结果可以看出,机器人通常需要较长时间才能学习得到较优的动作选择策略,但机器人足球比赛往往在短时间内结束,机器人没有足够的时间进行相应的学习. 因此,在实际应用中,可考虑将预编码方式同学习算法相结合;同时,如何改进现有算法,提高学习的效率,将是以后研究的重点.

4 结论

1) 通过在守门员的动作选择问题中引入 Sarsa(λ) 学习算法,并用 CMAC 神经网络对状态空间泛化,实现了守门员动作的自主选择,避免了在采用预编码方式确定守门员动作时存在的缺乏自适应能力等问题.

2) 实验结果表明经过一段时间的学习后,采用学习算法的守门员的防守时间显著增长,明显超过采用其他策略的守门员的防守时间,验证了算法的有效性.

参考文献:

[1] NODA I, MATSUBARA H, HIRAKI K, et al. Soccer server: a tool for research on multiagent systems [J]. Applied Artificial Intelligence, 1998, 12(2/3): 233-250.

[2] WEIGEL T, GUTMANN S, DIETL M, et al. CS Freiburg: coordinating robots for successful soccer playing [J]. IEEE Transactions on Robotics and Automation, 2002, 18(5): 685-699.

[3] 李人厚. 智能控制理论和方法 [M]. 西安: 西安电子科技大学出版社, 1999: 116-120.

[4] SUTTON S, BRATO G. Reinforcement learning: an introduction [M]. Cambridge: A Bradford Book, 1998: 4-6.

[5] STONE P, SUTTON S. Scaling reinforcement learning toward RoboCup soccer [C] // The Eighteenth International Conference on Machine Learning, Williams College. Williamstown: Morgan Kaufmann, 2001: 537-544.

[6] WHITESON S, TAYLOR E, STONE P. Empirical studies in action selection with reinforcement learning [J]. Adaptive Behavior, 2007, 15(1): 33-50.

[7] 段勇, 杨淮清, 崔宝侠, 等. 强化学习在足球机器人基本动作学习中的应用 [J]. 机器人, 2008, 30(5): 453-459.
DUAN Yong, YANG Huai-qing, CUI Bao-xia, et al. Application of reinforcement learning to basic action learning of soccer robot [J]. Robto, 2008, 30(5): 453-459. (in Chinese)

[8] KOK J. UvA trilearn 2003-soccer simulation team [EB/OL]. [2010-07-18]. http://staff. science. uva. nl/~jellekok/robocup/2003/.

(责任编辑 张 蕾)