文章编号：1001-0920 (2011) 02-0227-06

基于 ART 2 的 Q 学习算法研究

姚明海, 瞿心昱, 李佳鹤, 顾勤龙, 汤丽平

(浙江工业大学 信息工程学院，杭州 310023)

摘 要： 为了解决 Q 学习应用于连续状态空间的智能系统所面临的“维数灾难”问题, 提出一种基于 ART 2 的 Q 学习算法. 通过引入 ART 2 神经网络, 让 Q 学习 Agent 针对任务学习一个适当的增量式的状态空间模式聚类, 使 Agent 无需任何先验知识, 即可在未知环境中进行行为决策和状态空间模式聚类两层在线学习, 通过与环境交互来不断改进控制策略, 从而提高学习精度. 仿真实验表明, 使用 ARTQL 算法的移动机器人能通过与环境交互学习来不断提高导航性能.

关键词： Q 学习；ART 2；增量式学习；两层在线学习；移动机器人导航

中图分类号： TP273 **文献标识码：** A

Study on Q-learning algorithm based on ART 2

YAO Ming-hai, *QU Xin-yu*, *LI Jia-he*, *GU Qin-long*, *TANG Li-ping*

(College of Information Engineering，Zhejiang University of Technology，Hangzhou 310023，China．Correspondent: YAO Ming-hai，E-mail：ymh@zjut.edu.cn)

Abstract: In order to solve the problem of dimension disaster which may be produced by applying Q-learing to intelligent system of continuous state-space, this paper proposes a Q-learning algorithm based on ART 2 and gives the specific steps. Through introducing the ART 2 neural network in the Q-learning algorithm, Q-learning Agent in view of the duty learns an appropriate incremental clustering of state-space model, so Agent can carry out decision-making and a two-tiers online learning of state-space model cluster in unknown environment without any priori knowledge. Through the interaction with the environment unceasingly alternately to improve the control strategies, the learning accuracy is increased. Finally, the mobile robot navigation simulation experiments show that, using the ARTQL algorithm, motion robot can improve its navigation performance continuously by interactive learning with the environment.

Key words: Q-learning；ART 2；incremental learning；two-tiers online learning；mobile robot navigation

1 引 言

强化学习[1]是一种无监督的在线学习, 它采用“试错法”学习机制, 通过感知环境状态和从环境中获得不确定奖赏值来学习动态系统的最优行为策略. 因此, 强化学习被广泛应用于智能系统的控制中[2-3], 并被证明是一种通过经验逐步提高智能系统自主学习能力的有效计算方法[4], 其中 Q-Learning 算法[5]是一种应用最广的强化学习算法. 由于经典的 Q 学习算法主要用于求解具有不完整的、离散信息的马尔可夫行为问题, 而智能体所处的环境通常是连续变化的状态空间, 导致将 Q 学习应用于连续状态空间的智能系统将面临着动态规划的“维数灾难”问题[4]. 针对维数灾难目前主要有两种解决途径: 1) 将连续状态空间进行离散化; 2) 函数逼近法. 其中离散化方法的核心思想是将连续状态空间量化为离散区域, 同一区域的状态值函数相等, 即任务分解. 文献 [6] 用区格法先将整个状态空间划分成若干区域, 在同一区域认为其值函数相等; [7] 根据当前行为价值估计函数, 采用矢量化的方法对状态空间进行自适应划分; [8] 研究了一种多线性插值法来缓解 Q 学习的维数问题. 但这些方法泛化能力有限, 会造成区域划分过于简单和运算复杂性增加的矛盾. 函数逼近法是采用参数化的函数来逼近强化学习要学习的各种映射关系, 而神经网络具有很好的泛化能力和非线性函数逼近能力, 适用于函数逼近法. 已有学者采用神经网络解决连续空间强化学习问题, 如: BP 网络、自适

收稿日期： 2009-11-24；**修回日期：** 2010-06-18.

基金项目： 国家自然科学基金项目(61070113)；浙江省自然科学基金项目(20080376).

作者简介： 姚明海(1963−), 男, 教授, 博士生导师, 从事人工智能与智能控制、模式识别等研究；瞿心昱(1985−), 男, 博士生, 从事人工智能的研究.

应共振理论(ART)网络、径向基函数网络等. 这些基于神经网络的强化学习能很大程度提高学习速度. 以上这些传统方法要求Agent具备一定的环境模型先验知识, 且当外部环境发生改变时, Agent需要重新进行状态划分和行为策略的学习. 而Q学习Agent最显著的特征是自主性, 在完成任务的过程中应能适应其所处未知环境的变化, 需要一种函数逼近效果好的在线学习方式学习所需映射关系. 因此, 本文提出一种基于ART 2的Q学习算法. ART 2[9]是一种自组织竞争神经网络结构, 是无监督的学习网络, 它克服了BP网络、反馈网络等对已学知识的忘却问题, 在与环境交互中, 其记忆容量可随样本空间的增加而自适应地增加, 并且在不破坏网络先前所学知识的情况下学习新的知识. 与同为自组织网络的SOM不同的是, ART 2可在线地学习而无需像SOM一样重新学习新类别, 这对于机器人实时在线路径规划和避障是非常必要的. 通过在Q学习算法中引入ART 2神经网络, 让Agent在学习过程中, 针对需要完成的任务学习一种适当的增量式的状态空间模式聚类, 使Agent无需任何先验知识即可在实际运行环境中进行行为决策和状态空间模式聚类的两层在线学习, 通过与环境交互来不断改进控制策略, 从而提高学习精度. 此后, 在移动机器人导航仿真实验中表明ARTQL学习算法是合理有效的.

2 ART2 神经网络

Carpenter[10-11]提出了ART. 之后, 他与Grossberg合作提出了ART网络, 该网络是一种模拟人类心理和认知活动机理的人工神经网络模型, 其重要特点是可解决现有的网络模型难以兼顾“适用于稳定性”和“弹性”的缺陷. 目前, 基于ART理论已建立了3种重要的ART神经网络模型, 其中ART2网络采用竞争学习和自稳机制原理实现稳定的无监督分类, 能在未知数据类别和类别数的情况下进行实时的在线学习, 并对已学习过的模式快速响应和自动识别, 是一种理想的自组织神经网络分类器模型[12-13]. ART 2的结构如图1所示.

ART 2由注意子系统和调整子系统两部分组成. 注意子系统由F_1, F_2两个STM层以及两者之间的LTM层构成, 主要完成由底向上模式矢量的竞争选择及矢量间相似度的比较. 其中: F_1为输入比较层; F_2为整个系统的核心, 是识别层, 用来完成各神经元的竞争学习. 调整子系统则由图1左边的重置机构组成, 用于检查相似度能否达到满意的标准, 并作出相应的动作, 决定成功还是重置. 设输入向量X是一个N维的模拟量$X=(x_0,x_1,\cdots,x_{N-1})$. 在$F_1$中含有$N$个处理单元, 每个处理单元由3层6个神经元构成, 分别为: z_i,s_i,v_i,u_i,p_i,q_i, 构成两个正反馈滤波回路, 用于对输入信号x_i进行特征增强和噪声抑制, 如图1所示. 输入信号经F_1层处理后, 通过与自下而上的LTM权系数w_{ij}的加权组合传递到F_2层. 在F_2层利用竞争机制产生获胜神经元j^*, 之后通过j^*对应的自上而下的LTM权系数w_{j^*i}, 将信号反馈回F_1层的p_i节点. 此时F_1层的p_i节点是去噪后的归一化输入u_i和获胜节点模式中心w_{j^*i}的线性组合, 通过比较p_i与p_i的相似度$\|R\|$与警戒门限$\rho\,(0<\rho<1)$的大小来决定是否对F_2层进行重置. 若$\|R\|<\rho$, 则调整子系统向F_2层发出重置信号抑制当前获胜单元, 在F_2层的其他单元中重新进行竞争选择; 若$\|R\|>\rho$, 则对获胜单元的长期记忆LTM权值进行调整, 以加深对输入模式的记忆. 如果输入模式对F_2层所有已存在模式的相似度均不能满足警戒门限, 则将输入模式作为一类新的模式加入到F_2层中, 并保存其LTM权值.

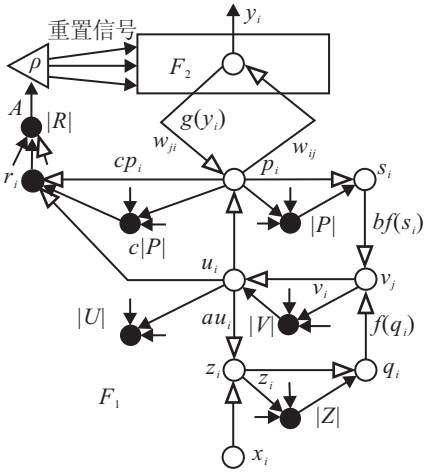

图 1 ART2 网络结构示意图

3 基于 ART 2 的 Q 学习算法

3.1 Q-Learning 算法

强化学习[4]是指Agent通过从环境状态到动作映射的学习, 以使动作从环境中获得的累积强化信号最大. 当Agent处于某一环境状态时, 强化学习方法并不告知采取相应的正确动作, 而是由环境提供的强化信号对所选动作的优劣进行评价, 通过不断地试错来得到最优策略(状态-动作的最佳映射). 强化学习结构与状态-动作序列过程如图2所示.

强化学习中应用较广泛的是1-step Q-Learning算法[5], Q学习可看作一种增量式动态规划, 它通过

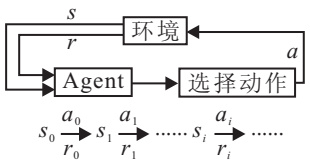

图 2 强化学习系统结构图

直接优化一个可迭代计算的行为价值函数来找到一个策略使得期望折扣回报总和最大, 使 Agent 在每一次的迭代中都需要考察每一个行为, 可确保学习过程收敛.

$Q(s_t, a_t)$ 表示 Agent 在 s_t 状态下执行动作 a_t 的价值函数, 它的值为从状态 s_t 开始执行动作 a_t 的立即回报加上以后遵循最优策略所产生的延时回报, 可表示为

$$Q(s_t, a_t) = r(s_t, a_t) + \gamma V^*\delta(s_t, a_t) = (s_t, a_t) + \gamma \max_a Q(s_{t+1}, a).$$

一旦求出最优行为价值函数 $Q^*(s_t, a_t)$, 则优化策略为

$$\pi^*(s_t) = \arg\max_{a_t} Q^*(s_t, a_t).$$

采用一步 Q 学习算法来迭代逼近实际的 Q 函数, 可在线增量式估计 Q 值, 同时学习策略和行为价值函数. 在学习过程中, $Q(s_t, a_t)$ 的更新迭代公式为

$$Q(s_t, a_t) = (1-\alpha)Q(s_t, a_t) + \alpha(r(s_t, a_t) + \gamma \max_a Q(s_{t+1}, a)).$$

其中: α 为学习率, γ 为折扣因子, $r(s_t, a_t)$ 为在状态 s_t 执行动作 a_t 的立即回报. 在学习过程的早期步骤中, $Q(s_t, a_t)$ 不一定能准确地反映其隐式含义的策略. 但通过所有状态重复所有动作, 由长期的强化信号约束可确保 Agent 学到整体最优策略.

3.2　基于 ART 2 的 Q 学习算法

通常, Agent 所处的环境是连续变化的状态空间, 而强化学习算法主要求解具有不完整的、离散信息的马尔可夫行为问题, 将强化学习应用于连续状态空间的智能系统将面临着动态规划的“维数灾难”问题. 为此, 在 Q 学习算法中引入 ART2 神经网络, 让 Q 学习 Agent 针对需要完成的任务, 学习一个适当的增量式的状态空间模式聚类, 使 Agent 无需任何先验知识即可在实际运行环境中进行行为决策和状态空间模式聚类的两层在线学习, 通过与环境交互来不断改进控制策略, 从而提高学习精度. ARTQL 算法的体系结构如图 3 所示. 图 3 可看作一个 6 层拓扑结构的神经网络, 其中: A 层为输入层, 表示 Agent 所能探测到的状态空间; B, C, D 层对输入信号进行噪声抑制和特征增强处理; E 层各神经元对应于通过竞争学习划分后的各状态空间模式类; F 层是输出层, 表示 Agent 能执行的 L 种行为. E 层的每个模式类都与 F 层的 L 个神经元 $A = (a_1, a_2, \cdots, a_L)$ 相连, 其连接权值 z_{jk} 代表每个模式-行为对的值函数估计 $Q(y_j, a_k)$.

ARTQL 算法流程如图 4 所示.

1) 当外界状态进入 ARTQL 输入层, 输入矢量经过 B, C, D 层的噪声抑制和特征增强以后, 进入 E 层进行竞争学习以选择相应的状态空间模式类. 同时, 根据输入矢量的短期记忆模式与竞争获胜单元的长期记忆模式之间的匹配度是否满足警戒门限来决定是否对获胜单元抑制和重新进行竞争学习. 对通过警戒门限测试的竞争获胜单元所对应的长期记忆 LTM 权向量进行调整, 以加深对输入模式的记忆. 若输入模式对所有已存在模式类均不能匹配, 则在 E 层添加一个新的状态空间模式类, 并保存它的 LTM 权值.

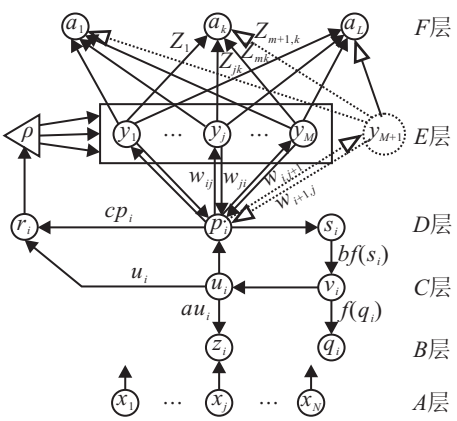

图 3　ARTQL 体系结构示意图

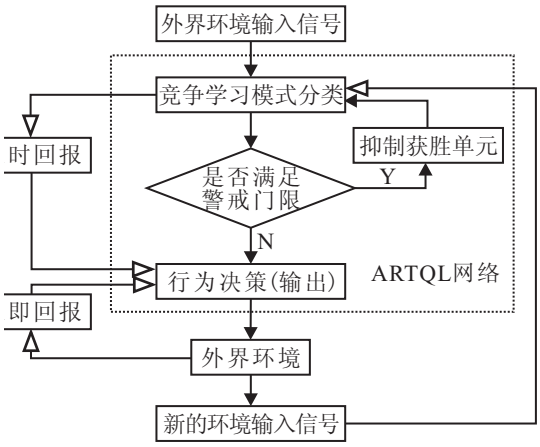

图 4　ARTQL 学习算法流程

2) E 层竞争获胜单元根据其对应于 F 层各个神经元的权值(即每种行为的估计 Q 值), 采用 Boltzmann 探索机制选择一个行为作用于外部环境.

3) 由强化学习与环境交互得到该选择行为的运行效果, 以及采取该行为后 Agent 所进入的新的状态空间模式类.

4) 根据选择行为的立即回报和遵循最优策略所产生的延时回报, 更新 E, F 层之间相应的网络权值, 以使下次 Agent 进入该状态空间模式类时, 能有更优的行为决策能力. Q 学习算法在每一步只更新一个估计 Q 值, 学习效率不高, 因此引入资格迹机制使每一步学习都能更新 E, F 层之间所有的网络权值, 从而提高学习效率.

5) 通过这种不断与环境的交互学习, 使Agent在提高其行为决策能力的同时, 能学习到一个适当的增量式的状态空间模式聚类, 当经过两层在线学习足够时间和次数后, Agent能较好地完成任务.

4 ARTQL算法实现

ARTQL学习算法的具体步骤如下:

Step 1: 参数初始化. 初始化A, B, C, D, E, F层各神经元的特征值为0; 初始化B, C, E层系数, 滤波因子$\theta = 1/\sqrt{N}$; 初始化E层模式类总数$M = 1$, N为输入向量的维数; D, E层之间内外星权向量初始化为$(1-d)\sqrt{N}$. 取E, F层之间权向量$z_{jk} = 10$, 所有模式类的资格迹$w_{ji} = 0$, 定义衰减因子λ, 强化学习系数α, 折扣因子γ, 权重学习系数d以及警戒门限ρ.

Step 2: 重复执行Step 3~Step 18的学习循环过程, 直到满足停止条件.

Step 3: 将Agent重置到初始状态空间X_0, 重置计时器t.

Step 4: 时刻Agent进入状态空间$X(t)$. 若X_0为终止状态空间X_{end}, 则Agent完成本轮的在线学习任务, 返回执行Step 3; 否则, 继续执行Step5.

Step 5: 网络由A层接收输入信号x_i, 按照Step 6~Step 17执行一次学习.

Step 6: B层各神经元的特征值更新算法为

$$\begin{cases} z_i = x_i + au_i, \\ q_i = z_i/(e + \|Z\|). \end{cases}$$

Step 7: C层各神经元的特征值更新算法为

$$\begin{cases} v_i = f(q_i) + bf(s_i), \\ u_i = v_i/(e + \|V\|), \end{cases}$$

$$f(x) = \begin{cases} 0, 0 \leqslant x < \theta; \\ x, x \geqslant \theta. \end{cases}$$

Step 8: D层各神经元的特征值更新算法为

$$\begin{cases} p_i = u_i + dw_{ji}, \\ s_i = p_i/(e + \|P\|). \end{cases}$$

Step 9: 按下式计算E层M个神经元的特征值:

$$y_j = \sum_{i=1}^{N} p_i w_{ij}.$$

Step 10: 根据竞争机制得到E层的获胜神经元

$$y_{j*} = \max\{y_j | j = 1, 2, \cdots, M\},$$

并向D层的节点发出反馈调整信号

$$g(y_j) = \begin{cases} d, \ j = j^*; \\ 0, \ j \neq j^*. \end{cases}$$

$$p_i = u_i + \sum_{j=1}^{M} g(y_j) w_{ji}.$$

Step 11: 警戒门限测试. 由节点u_i和调整后的节点p_i计算D层神经元r_i的特征值

$$r_i = \frac{u_i + cp_i}{e + \|U\| + c\|P\|}, \ \|R\| = \sqrt{\sum_{i=1}^{N} r_i^2}.$$

若$\|R\| \leqslant \rho$, 则抑制当前获胜单元$y_{j^*} = 0$, 并取消其竞争资格, 继续执行Step 12; 若$\|R\| \geqslant \rho$, 则执行Step 13.

Step 12: 若E层所有神经元均被抑制, 即

$$y_j = 0, \ \forall j = 1, 2, \cdots, M,$$

则在E层增加一个新的神经元y_{M+1}, 并作为获胜单元, 初始化对应的网络权值, 继续执行Step 13; 否则, 返回执行Step 9.

Step 13: 按下式更新获胜单元y_{j^*}所对应的E层内外星权向量, 以加深对当前输入模式的记忆:

$$w_{j^*i} = du_i + [1 - d(1-d)]w_{j^*i}, \ \forall i;$$

$$w_{ij^*} = du_i + [1 - d(1-d)]w_{ij^*}, \ \forall i.$$

Step 14: 基于Boltzmann分布选择F层节点α_k作为行为输出$a_{k^*} = 1$. 当E层节点y_{j^*}获胜时选择F层节点a_{k^*}的概率为

$$P(y_{j^*}, a_{k^*}) = \frac{\exp(z_{j^*k^*}/T)}{\sum_{k=1}^{L} \exp(z_{j^*k}/T)}.$$

其中: T为温度系数, 决定动作选择的随机性. 应随着学习次数的增加逐步减小T值, 以使算法逐步收敛于最优行为选择策略.

Step 15: 更新所有模式-行为对的资格迹

$$e(y_j, a_k) = \begin{cases} \gamma\lambda e(y_j, a_k) + 1, \ y_j = y_{j^*}, a_k = a_{k^*}; \\ \gamma\lambda e(y_j, a_k), \ \text{otherwise}. \end{cases}$$

Step 16: 执行动作α_k后, Agent进入下一状态空间X_{t+1}, 由强化学习评价函数得到t时刻Agent收到的立即回报$r(t)$.

Step 17: 返回执行Step 6~Step 12, 得到$t+1$时刻E层的获胜单元$y_{j^*}(t+1)$以及其所对应的最大行为估计值$\max\limits_{a_k}(z_{j^*k}(t+1))$, 按下式更新$t$时刻$F$层的所有网络权值

$$z_{jk}(t) = z_{jk}(t) + \alpha[r(t) + \gamma \max\nolimits_{a_k}(z_{j^*k}(t+1)) - z_{jk}(t)]e(y_j, a_k), \ \forall j, k.$$

Step 18: Agent进入下一时刻的学习, $t = t+1$, 返回执行Step 4.

5 仿真实验分析

为了验证ARTQL学习算法的有效性, 本文在移动机器人避碰撞导航仿真实验中引入ARTQL学习算法, 用ARTQL网络中的长期记忆LTM权向量来存储机器人探索过的所有状态空间模式类以及相应的强

化学习行为估计值. 移动机器人 R 在完全未知的环境中运行, 通过传感器获取环境信息后, 将当前状态矢量输入 ARTQL 网络, 在网络内部通过竞争学习得到当前输入矢量所属的状态空间模式类, 由该模式类所对应的所有状态行为估计值选择一个导航动作作用于外部环境. 根据强化学习函数评价导航效果, 当效果不理想时, 惩罚相应的模式行为估计值, 以降低下次同一模式类获胜时该导航动作被选择的概率; 否则, 增大网络中相应的模式行为估计值, 使效果好的导航动作能被再次选择. 若当前输入矢量对所有已存在模式类均不能匹配, 则机器人认为探索到一类新的环境类型, 同时扩大 ARTQL 网络的拓扑结构和保存新模式类的 LTM 权值. 仿真流程如图 4 所示. 对仿真实验的状态空间维数控制以及增量式的在线学习情况进行分析比较. 仿真实验在 Microsoft Visual C++6.0 软件平台上实现. 仿真环境如图 5 所示. 坐标系中随机分布着各种形状的障碍物, 标号 0 为机器人的起始位置, 标号 1 为目标区域. 仿真机器人从起始位置出发, 直到到达终点或执行动作的次数超过预先设置的阈值, 结束一次学习过程, 将机器人重置回起始位置重新进行下一轮的学习. 实验中设定机器人的控制周期 $T = 0.02$ s, 动作执行阈值为 2 000 次. ARTQL 算法的参数设置如下: $a = b = 10, c = 0.1$, $d = 0.9, e = 0.05, \rho = 0.99, \alpha = 0.2, \gamma = 0.8, \lambda = 0.05$.

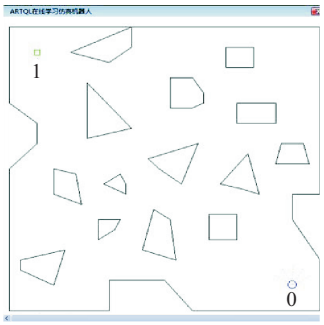

图 5　仿真环境设置

ARTQL 网络的输入包括模拟机器人 5 个方向上声纳传感器的距离信息 $d = \{d_{l1}, d_{l2}, d_c, d_{r2}, d_{r1}\}$, 以及当前机器人运动方向与目标方向之间的夹角 θ_t, 输入矢量维数 $V = 6$. 输出动作由移动机器人左、右轮的线速度 v_l 和 v_r 组成, 规定机器人的可选动作有 6 种, 如表 1 所示.

表 1　仿真机器人的可选动作

可选动作	v_l	v_r
直行	0.4	0.4
倒退	−0.4	−0.4
向左大转	0.0	0.8
向左小转	0.2	0.6
向右大转	0.8	0.0
向右小转	0.6	0.2

根据机器人动力学, 机器人在 t 时刻的线速度和角速度可以分别表示为

$$v_t = \frac{v_l + v_r}{2},\ \omega_t = \frac{v_l - v_r}{L},$$

其中 L 为机器人的宽度. 机器人所要学习的导航动作包括避障和接近目标, 综合两种行为, 确定强化学习评价函数如下:

$$r_t = \begin{cases} 1.0,\ D_t \leqslant D_R; \\ -1.0,\ d_t \leqslant D_S; \\ k_1(\sum d_t - \sum d_{t-1}),\ D_S \leqslant d_t \leqslant D_A; \\ k_2(\theta_{t-1} - \theta_t),\ d_t > D_A. \end{cases}$$

从而验证了本文方法的泛化能力和导航精度, 如图 6 所示.

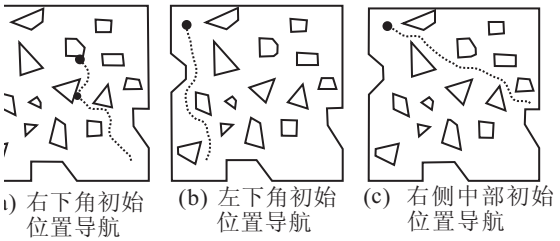

图 6　3 个不同初始位置的导航结果

图 6 中分别比较分析了在学习过程中 ARTQL 方法的模式总数和碰撞次数的变化趋势, 其中 ARTQL 对模式总数的控制体现了其对维数灾难问题的解决方法. 模式总数比较分析曲线如图 7 所示, 碰撞次数分析曲线如图 8 所示.

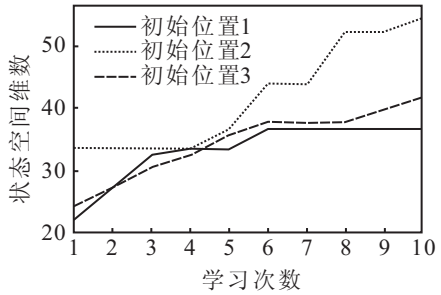

图 7　状态空间维数分析

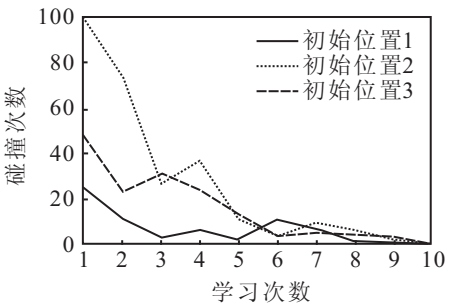

图 8　碰撞次数分析

通过实验, 得到如下结论: 1) 在导航实验中, 机器人能根据当前环境类型选择合适的动作进行在线学习. 机器人在一轮学习中, 与障碍物碰撞的次数越少, 到达目标区域的耗时越短, 导航性能越强. 2) 随着在

线学习次数的增加, 机器人通过与环境的不断交互, 利用ARTQL学习算法, 逐步优化行为决策和状态空间模式分类的能力, 导航性能不断提高. 3) 在不同的初始位置下, 通过先前学到的知识, 在新的环境中经过较短时间的学习, 机器人就能迅速完成导航任务. 实验证明, 机器人在未知环境探索中, 其记忆容量可随外界状态样本空间的增加而自适应地增加, 并且不会破坏先前所学的知识.

6 结 论

本文提出一种基于ART 2的Q学习算法. 利用ART 2网络中的LTM权值存储已探索到的所有状态空间模式类特征, 当Agent在未知环境中运行时, ART 2通过竞争学习对当前输入状态进行自适应地归类, 其记忆容量可随输入样本空间的增加而自适应地增加, 并且不会破坏先前所学的知识. 通过在Q学习算法中引入ART 2神经网络, 让Q学习Agent针对需要完成的任务学习一个适当的增量式的状态空间模式聚类, 使Agent无需任何先验知识即可在实际运行环境中进行行为决策和状态空间模式聚类的两层在线学习, 通过与环境交互来不断改进控制策略, 从而提高学习精度. 最后, 在移动机器人避碰撞导航仿真实验中引入ARTQL学习算法. 实验结果表明, 仿真机器人在未知环境中采用ARTQL进行在线学习, 通过与环境的不断交互, 逐步优化机器人行为决策和状态空间模式聚类的能力, 实现导航性能的不断提高. 由此可知, ARTQL学习算法是合理有效的. 另外, ARTQL中所使用的强化学习方法也需进一步完善以适应更加复杂的学习任务.

参考文献(References)

[1] Tom M Mitchell. Machine learning[M]. Beijing: Machine Press, 2004.

[2] Xiao N F, Nahavandi S. A reinforcement learning approach for robot control in an unknown environment[C]. Proc of the IEEE Int Conf on Industrial Technology. Bangkok, 2002: 1096-1099.

[3] Wang Y C, John M Usher. Application of reinforcement learning for Agent-based production scheduling[J]. Engineering Application of Artificial Intelligence, 2005, 18(1): 73-82.

[4] 高阳. 强化学习研究进展—-机器学习及其应用[M]. 北京: 清华大学出版社, 2006: 116-134.
(Gao Y. Progress of reinforcement learning research — Machine learning and its application[M]. Beijing: Tsinghua University Press, 2006: 116-134.)

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

[6] Singh S, Jaakkola T, Jordan M I. Reinforcement learning with soft state aggregation[C]. Advances in Neural Information Processing Systems. Morgan Kaufmann: MIT Press, 1995: 361-368.

[7] Lau H Y K, Mak K L, Lee I S K. Adaptive vector quantization for reinforcement learning[C]. Proc of the l5th World Congress of Int Federation of Automatic Control. Barcelona, 2002: 21-26.

[8] Davies S. Multidimensional triangulation and interpolation for reinforcement learning[C]. Advance in Neural Information Processing Systems. Cambridge: MIT Press, 1997: 1005-1010.

[9] Karthikeyan B, Gopal S, Venkatesh S. ART-2: An unsupervised neural network for PD pattern recognition and classification[J]. Expert System Application, 2006, 31 (2): 345-350.

[10] 韩力群. 人工神经网络理论、设计及应用[M]. 北京: 化学工业出版社, 2007.
(Han L Q. Theory, design and application of artificial neural network[M]. Beijing: Beijing Chemical Industry Press, 2007.)

[11] Carpenter G A, Grossberg S. ART2: Stable self-organization of stable category recognition codes for analogue input patterns[J]. Applied Optics, 1987, 26(23): 4919-4930.

[12] Luo J H, Chen D Z. An enhanced ART2 neural network for clustering analysis[C]. Proc of the 1st Int Workshop on Knowledge Discovery and Data Mining. Adelaide, 2008: 81-85.

[13] Qian X D, Wang Z O, Wang Y. A method of data clustering based on improved algorithm of ART2[C]. Proc of the 4th Int Conf on Machine Learning and Cybernetics. Guangzhou, 2005: 2021-2026.