文章编号:2095-476X(2013)03-0042-04

基于粗糙集的改进Q学习算法

过金超, 刘杰, 崔光照

(郑州轻工业学院 电气信息工程学院, 河南 郑州 450002)

摘要:针对Q学习算法容易出现错误的时间间隔重叠和高估Q值的情况,进而导致收敛速度慢、学习性能下降的问题,提出了一种改进的Q学习算法,即粗糙集Q学习算法.该算法通过有效处理不完备信息和不确定性知识,使Q值所引起的误差最小化,进而减少Q值的高估,提高学习性能.基于2种算法的机器人自主导航实验结果表明,粗糙集Q学习算法有更高的学习效率和更强的避障能力.

关键词:Q学习算法;粗糙集;机器人导航

中图分类号:TP242.6;TP181 **文献标志码**:A DOI:10.3969/j.issn.2095-476X.2013.03.010

An improved Q-learning algorithm based on rough set

GUO Jin-chao, LIU Jie, CUI Guang-zhao

(*College of Electric and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou* 450002, *China*)

Abstract: Q-learning algorithm has a fundamental flaw, that is, prone to error intervals overlap, and thus overestimation of the correct Q-value. These are likely to lead to low convergence speed and continuous decline in the performance of Q-learning, an improved Q-learning algorithm was proposed, that was rough sets Q-learning algorithm. The algorithm can be able to minimize the overestimation caused by Q-values and improve performance of learning through effectively deal with incomplete information and uncertain knowledge. Navigation experiments based on these two algorithms were conducted, the results showed that rough sets Q-learning algorithm had higher efficiency of learning and stronger ability of obstacle avoidance than Q-learning algorithm.

Key words: Q-learning algorithm; rough set; robot navigation

0 引言

强化学习是一种非常有效的机器学习方法,在智能机器人系统应用非常广泛[1].它是一种无监督学习模式,通过试探行动-评价的迭代从环境中获得知识,进而在该环境中选择最优动作.因为外部环境信息提供得较少,强化学习的学习效率比较低,因此学者们一直致力于提出新的算法以对其进行改进.

Watkins于1989年提出了Q学习算法,它在强化学习的发展过程中具有里程碑的意义[2].文献[3]利用Q学习算法进行了移动机器人自主导航实验,证明该算法具有可行性,但是标准Q学习算法收敛速度慢,学习效率较低.在Watkins之后,1996年J. Peng提出了多步Q学习算法[4],利用将来无限多步的信息进行更新当前Q函数,因此要对大量的

收稿日期:2013-01-20

作者简介:过金超(1978—),男,河南省开封市人,郑州轻工业学院副教授,博士,主要研究方向为机器人智能控制及优化.

状态 - 动作对进行更新. 当状态 - 动作对空间规模较大时,计算量很大,学习效率并不高.

本文在 Q 学习算法基础上,提出一种改进的 Q 学习算法,即粗糙集 Q 学习算法,将该算法在机器人导航中实验验证,以期为强化学习开创新的发展空间.

1　Q 学习算法

1.1　Q 学习算法流程

Q 学习算法流程如图 1 所示[5]. Q 学习算法是无模型强化学习方法,机器人在系统中不需预测将来的未知状态. 策略是指在状态 s 中采取行动 a 的概率的函数. 回报预测是行动和状态的一种函数,其表达式为

$$R(s,a) = E[\sum \gamma^t r_t + 1 \mid s_0 = s, a_0 = a]$$

其中,$\gamma(0 \leqslant \gamma \leqslant 1)$ 是折扣因子,r_t 为 t 时刻的立即回报.

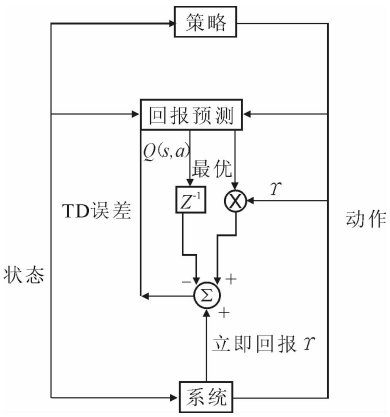

图 1　Q 学习算法流程

从图 1 可以看出,回报预测有预测状态和预测行动 2 个输出. 前者与 γ 相乘,图中用 ⊗ 来表示,得到的结果与 $Q(s,a)$ 相减得到 $[\gamma\max\ Q(s',a') - Q(s,a)]$,其结果再与立即回报 r 相加,最终得到 $r + [\gamma\max\ Q(s',a') - Q(s,a)]$. 回报预测和当前状态会影响策略模块,从而影响下一个状态,最终得到

$$Q(s,a) = Q(s,a) + \alpha[r + \gamma\max_{a'} Q(s',a') - Q(s,a)]$$

1.2　Q 学习算法存在的问题

机器人在初始状态 s 下,Q 学习算法会根据某种策略选择一个动作 a,得到下一个状态 s',同时获得立即回报 r,于是有

$$Q(s,a) = r + \gamma \max_a (s',a') \qquad ①$$

在某一状态 Q 值提出一个动作选择策略,在其对应状态选择一个 $Q(s,a)$ 最大时的动作 a. 反复应用 ① 式将产生最大化预期的累积折扣奖励的动作策略,但是此结果只有当 Q 值被精确地存储时才成立. 若设当前被储存的 Q 值为 Q^a,它意味着某种隐含的目标值 Q^t 被一个噪声项 Y_s^a 破坏,Y_s^a 的近似值由 ② 式给出

$$Q^a(s',a') = Q^t(s',a') + Y_s^a \qquad ②$$

该噪声被一系列均值为 0 的随机变量模式化,它会导致 ① 式左边的某种错误,设随机变量

$$Z_r = r + \gamma \max_a Q^a(s',a') - [r + \gamma \max_a Q^t(s',a')] = \gamma[\max_a Q^a(s',a') - \max_a Q^t(s',a')]$$

有关以上分析的一个关键观察是:零均值噪声 Y_s^a 可能很容易导致正平均值 Z_r. 此情况可能导致一些 Q 值太小,一些太大. 但是 Q 学习算法总是采用最大的 Q 值,如果一些 Q 值大小接近,并且错误的时间间隔有重叠现象,那么其中的一个很可能高估某个行动正确的 Q 值[6]. 这种高估会使 Q 学习算法的性能不断下降,以致智能机器人系统不能学到最佳策略.

2　粗糙集 Q 学习算法

粗糙集理论由 Pawlak 在 1982 年提出,它能对各种不完备信息和不确定性知识进行有效的分析,并对其进行推理,从而发现隐含的知识和规律,为智能机器人系统在未知环境下的导航及其他应用提供较为完善的理论基础[7]. 粗糙集理论包含 2 方面的内容,即有决策分析与无决策分析. 前者包括决策分析、规则提取、故障诊断;后者包括从大量信息中发现知识,离散信息系统的数据约简以及对得到的不确定性知识和不完备信息的处理.

在 Q 学习算法中,Q 值容易出现高估的情况,会严重的影响学习性能. 因而,本文基于粗糙集理论的无决策分析和有决策分析对 Q 学习算法做了 2 点改进以提高其学习性能.

1) 减去来自最大 $Q(s',a')$ 的 $(1-\gamma)v'$. 这有助于使高估值接近标准值,减少 Q 值的误差. $\gamma(0 \leqslant \gamma \leqslant 1)$ 是折扣因子,v' 是粗糙包含的平均值,其计算公式为

$$v' = \frac{1}{|B|} \sum_{B_j(x) \in B} v(B_j(x), B * D)$$

其中,$v(B_j(x), B * D) = \dfrac{|B_j(x) \cap B * D|}{|B * D|}$.

2) 用 γ^n 代替 γ. 这有助于减少 Q 值的高估.

上述 2 点均有助于提高算法的学习性能,从而得到粗糙集 Q 学习算法,其变换式为

$$Q(s,a) = Q(s,a) + v'[r + \gamma^n[\max_{a'} Q(s',a') - (1-\gamma)v'] - Q(s,a)]$$

式中,n 是情节的数量.

以下给出粗糙 Q 学习算法的主要描述.

输入:状态 $s \in S$,动作 $a \in A(s)$,初始化的 $Q(s,a)$,α,γ,π 为任意一个策略(非贪婪型).

输出:每个状态 - 动作对的最优动作值 $Q(s,a)$.

```
for(i = 0; i ≤ # of episodes; i ++) do
    Initialize s
  Choose a from s,using policy derived from Q
  Repeat(for each step of episode):
  Take action a; observe reward,r,and next state,s'
  Q(s,a) = Q(s,a) + v'[r + γ^n[max_a' Q(s',a') -
(1 -γ) v'] - Q(s,a)]
  s ← s'; a ← a';
  until s is terminal
 end
end
```

3 仿真实验及分析

TeamBots 是基于 Java 的一款机器人仿真软件,TBSim 为 TeamBots 中的仿真模块,本文采用 TBSim 来进行机器人导航实验. 实验环境如图2 所示,R(机器人所在位置) 为起始点,D(正方框) 为目标点,圆点部分为环境中的障碍物.

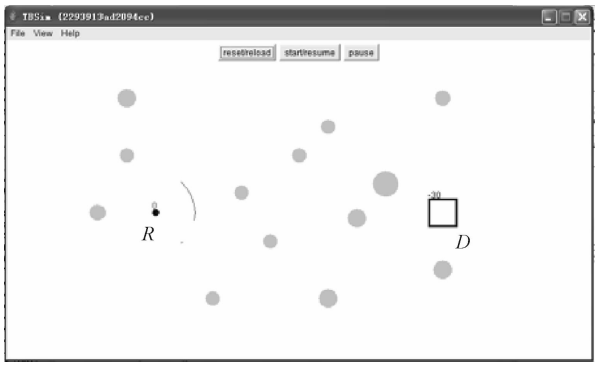

图 2 实验环境

实验环境建好之后,分别将 Q 学习算法和粗糙集 Q 学习算法在平台上进行实现,然后进行导航实验. 对于机器人来说,环境是未知的,障碍物亦是未知的,机器人需要通过这 2 种算法的学习来实现从起始点到目标点的无碰撞路径的寻找. 定义机器人从起始点到目标点的过程为机器人的一个学习周期. 因 2 种算法的学习效率是不一样的,为了测试 2 种算法的学习效率,实验设定同样的参数以更好地比较其性能,即 $\alpha = 0.6$,$\gamma = 0.8$,学习周期均为 100,并且设置同样的实验环境. 2 种不同算法下,机器人经过 100 个学习周期得到不同的实验结果,如图 3 和图 4 所示,其中曲线(目标点左侧) 为机器人的路径.

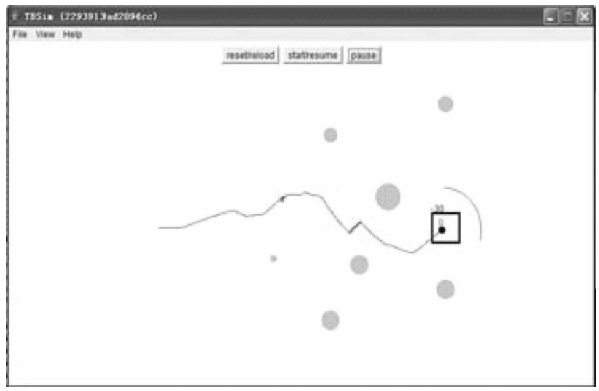

图 3 Q 学习算法导航示意图

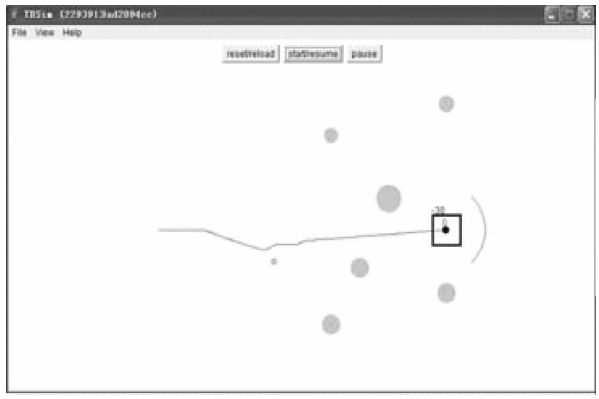

图 4 粗糙集 Q 学习算法导航示意图

图 3 的路线比较弯曲,路径较长一些;图 4 的路线接近直线,路径较短一些. 这说明粗糙集 Q 学习算法较 Q 学习算法能进行更好的自主学习导航,所以粗糙集 Q 学习算法比 Q 学习算法收敛速度更快,学习效率更高.

为了测试 2 种算法在导航实验中机器人的避障能力,实验设置 2 种相同的仿真环境,学习周期设置为 300. 将 2 种算法在仿真平台上实现,得到如图 5 所示的实验结果.

由图 5 可以看出,在学习周期均为 300 的情况

下,2 种算法的避障能力差异很大. 在 0—300 周期内,粗糙集 Q 学习算法的碰撞次数一直远远低于 Q 学习算法. 这说明粗糙集 Q 学习算法较 Q 学习算法有更好的避障能力.

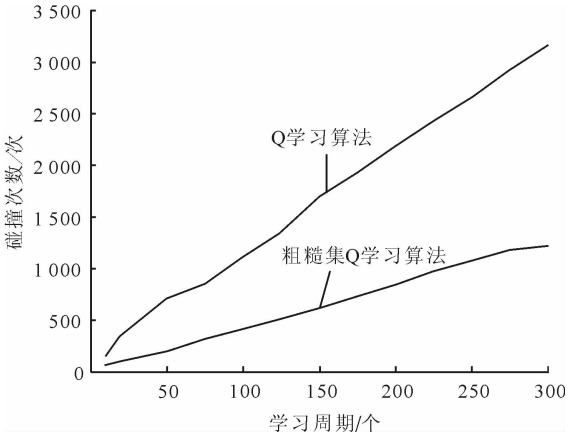

图 5　机器人碰撞曲线图

4　结论

本文针对Q学习算法容易高估 Q 值进而导致收敛速度慢这一问题,将 Q 学习算法与粗糙集理论结合,提出了一种改进的Q学习算法,即粗糙集Q学习算法. 粗糙集Q学习算法充分利用了粗糙集理论,它能有效地处理不完备信息和不确定性知识,减小了 Q 值的高估,使高估值更接近标准值,进而加快了收敛速度,提高了学习效率. 将 2 种算法在仿真实验中进行机器人自主导航的性能比较,结果表明,粗糙集 Q 学习算法较 Q 学习算法有更高的学习效率,更好的避障能力.

参考文献:

[1] Yu H Z, Bertsekas D P. Q-learning and policy iteration algorithms for stochastic shortest path problems [EB/OL]. (2012 - 04 - 18) [2013 - 01 - 05]. http://link.springer.com/article/10.1007%2Fs10479 - 012 - 1128 - z/fulltext.html.

[2] 王雪松,程玉虎. 机器学习理论、方法及应用[M]. 北京:科学出版社,2009.

[3] James F Peters, Christopher Henry. Approximation spaces in off-policy Monte Carlo learning[J]. Engineering Applications of Artificial Intelligence, 2007(20):667.

[4] Peng J, Williams R J. Incremental multi-step Q-learning [J]. Machine Leaning, 1996, 22(1/3):283.

[5] Pandey D, Pandey P. Approximate Q-learning: An introduction[C]//2010 Second International Conference on Machine Learning and Computing, Washington DC: IEEE Computer Society, 2010.

[6] 邱玉霞. 进化计算与粗糙集研究及应用[M]. 北京:冶金工业出版社,2009.

[7] 高庆吉. 基于粗糙集理论的移动机器人自主导航研究[D]. 哈尔滨:哈尔滨工业大学,2006:15 - 16.