增强学习算法的性能测试与对比分析*

李兆斌，徐 昕，吴 军，连传强
（国防科学技术大学 机电工程与自动化学院 自动化研究所，长沙 410073）

摘 要： 研究了几类典型增强学习算法的性能评估问题，包括 Q-学习算法、最小二乘策略迭代（LSPI）和基于核的最小二乘策略迭代（KLSPI）算法等，重点针对 Markov 决策问题（MDP）的值函数平滑特性对算法性能的影响进行了研究。分别利用值函数非平滑的组合优化问题——旅行商问题（TSP）和值函数平滑的 Mountain-Car 运动控制问题，对不同增强学习算法的性能进行了测试和比较分析。分析了三种算法针对不同类型问题的各自特点，通过实验对比，验证了近似策略迭代算法，特别是 KLSPI 算法在解决值函数平滑的序贯决策问题时性能更优。通过分析实验结果表明，MDP 值函数的平滑程度是影响近似策略迭代算法性能表现的重要因素。

关键词： 增强学习；值函数；近似策略迭代；平滑性
中图分类号： TP309 **文献标志码：** A **文章编号：** 1001-3695（2010）10-3662-04
doi：10.3969/j.issn.1001-3695.2010.10.014

Performance test and comparative analysis for reinforcement learning algorithms

LI Zhao-bin, XU Xin, WU Jun, LIAN Chuan-qiang
(*Institute of Automation, College of Mechatronics & Automation, National University of Defense Technology, Changsha* 410073, *China*)

Abstract: This paper studied the performance evaluation problem for reinforcement learning (RL) algorithms, including Q-learning, least-squares policy iteration (LSPI) and kernel based least-squares policy iteration (KLSPI). Investigated the performance influence of the smoothness of value functions in Markov decision processes in detail. Tested the RL algorithms on a combinatorial optimization problem—the traveling salesman problem (TSP), which had non-smooth value functions and the Mountain-Car motion control problem with smooth value functions. Analyzed the characteristics of different RL algorithms and demonstrated that approximate policy iteration algorithms, especial KLSPI, had better performance when solving sequential decision-making problems with smooth value functions. Furthermore, it verifies that whether is the sequential decision-making problems with smooth value functions or not will play an important role in the performance of approximate policy iteration.

Key words: reinforcement learning; value function; approximate policy iteration; smoothness

0 引言

增强学习（RL），又称为强化学习或者再励学习，是求解序贯优化决策问题的一类机器学习方法。在增强学习理论和算法研究中，通常将序贯优化决策问题建模为 Markov 决策过程（Markov decision process，MDP）[1]。但与运筹学中研究的求解 MDP 优化策略的动态规划方法不同，增强学习算法不要求已知 MDP 的状态转移模型，因此在不确定的优化决策与控制问题中具有更广泛的应用前景。近年来增强学习在算法和实际应用方面都取得了重要的研究进展，在解决一些复杂的优化控制问题中已经有了若干成功应用，如西洋双陆棋[2]、生产调度[3]、电梯优化控制[4]以及直升机的飞行控制[5]等。文献[6]充分论证了增强学习模型在反复选择中可以提高技能，并提出将增强学习与认知体系相结合的研究方向。

Watkins 在 1989 年提出的 Q-学习算法被认为是增强学习领域的重要突破[7]。Q-学习通过学习状态行为值函数使其直接逼近 Q^*。对于状态与行为集合有限的 MDP，只要充分遍历所有状态行为对，Q-学习在理论上可以收敛到最优值函数。针对具有连续或大规模状态行为空间的序贯优化决策问题，2003 年 Lagoudakis 等人[8]提出的最小二乘策略迭代算法（LSPI）极大地改善了之前其他值函数以及策略逼近算法稳定性不高的问题。2007 年文献[9]提出的基于核的最小二乘策略迭代算法（KLSPI）通过在基函数中引入核函数，使得算法的稳定性、收敛性相对于 LSPI 更进一步。

尽管增强学习算法，特别是近似策略迭代算法在解决连续的或者大规模状态行为空间序贯决策问题，如倒立摆学习控制[8,9]等问题上，算法性能较以前的学习控制算法有较大程度的提高，但在处理一些组合优化问题时，结果又往往很难令人满意。面对增强学习算法在处理各种实际问题时的不同性能表现，以往的研究仅局限于对算法的基本思想、收敛性和学习效率进行分析，或者只是具体分析问题的某个特性对算法性能产生的影响，而缺乏从问题的一般性入手对增强学习算法进行对比分析和性能研究，难以对增强学习算法在处理实际问题上的应用产生广泛的指导作用。为此，本文从 MDP 值函数平滑程度的角度出发，用旅行商问题（TSP）和值函数平滑的 Moun-

收稿日期：2010-03-22；修回日期：2010-04-26 基金项目：国家自然科学基金资助项目（60774076，90820302）；湖南省自然科学基金资助项目（07JJ3122）；霍英东青年教师基金资助项目（114005）

作者简介：李兆斌（1983-），男，陕西咸阳人，硕士研究生，主要研究方向为模式识别、机器学习（zgtongshuai_1@163.com）；徐昕（1974-），男，湖北宜昌人，研究员，博士，主要研究方向为模式识别、机器学习；吴军（1980-），男，湖南醴陵人，博士，主要研究方向为模式识别、智能系统；连传强（1986-），男，辽宁大连人，硕士研究生，主要研究方向为多机器人学习.

tain-Car 运动控制问题对上述的三种典型增强学习算法的性能进行了测试和比较分析,分析了各种算法针对两种不同问题的各自特点。

1　增强学习的基本原理和算法

增强学习在与环境相互作用的过程中,通过极大化或极小化累积回报来选择策略,即学习的目标函数是学习一个控制策略,以此建立从状态 s 到动作 a 的映射,如图 1 所示。

许多实际问题一般可以建模为 Markov 过程: 用$\{S,A,R,P\}$表示。其中: S 是状态空间; A 是动作空间; R 是回报函数; P 是状态转移概率; $p(s,a,s')$ 和 $r(s,a,s')$ 代表从状态 s 采取动作 a 到达 s' 的概率和回报。状态动作对(s,a)的期望回报定义为

$$R(s,a)=\sum_{s'}p(s,a,s')\,r(s,a,s') \tag{1}$$

状态行为值函数 $Q^{\pi}(s,a)$ 为在状态 s 执行动作 a,而且以后也依据策略 π 选择动作的期望折扣回报。其中 $\pi(s)$ 是动作策略。

$$Q^{\pi}(s,a)=E_{s_t,P,a_t,\pi}\Big(\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)\mid s_0=s,a_0=a\Big) \tag{2}$$

状态行为值函数 $Q^{\pi}(s,a)$ 满足下面的 Bellman 等式:

$$Q^{\pi}(s,a)=R(s,a)+\gamma\sum_{s'\in S}P(s,a,s')\sum_{a'\in A}\pi(a';s')\,Q^{\pi}(s',a') \tag{3}$$

其中 s'是从状态 s 采取动作 a 后的下一个状态。

对一个 MDP,存在一个最优策略 $\pi^{*}(s)$,它可以最大化累积回报:

$$\pi^{*}(s)=\arg\max_{a}Q^{\pi^{*}}(s,a) \tag{4}$$

尽管传统的表格型增强学习在理论和算法研究方面已取得了许多结果,并成为求解序贯优化决策问题的一类有效方法,但主要针对的是小规模离散状态空间,其状态和动作都认为是有限的集合。然而在实际工程应用中遇到的许多控制问题所涉及的状态变量,如温度、流量、位置和速度等一般取值于连续区间,此时传统的增强学习算法如 Q-学习算法、Sarsa-算法等就难以适应。它们不仅对大规模和连续空间的优化决策问题难以保证算法的收敛性,而且存在学习效率不高的缺点。为提高增强学习的泛化能力,一个基本的方法是通过逼近值函数或者策略来研究增强学习理论或算法。到目前为止,近似增强学习主要可以分为三类,即值函数逼近(VFA) 、策略搜索和执行器—评价器方法。在这三类逼近增强学习方法中,值函数逼近得到了广泛的研究。根据逼近的基本性质,函数逼近方法可以分为两类,即线性和非线性。其中线性逼近方法的基函数容易构造,但存在逼近能力有限的问题; 非线性的逼近方法如神经网络,与线性逼近方法相比,具有很强的逼近能力,但是非线性特征选择困难,网络构造缺乏可靠的理论依据,并且实验结果缺乏严格的理论证明。KLSPI 通过在策略迭代的过程中引入和应用核函数,使之成为解决大规模和连续空间马尔可夫决策问题中的一种有效方法。

1.1　Q-学习算法

Q-学习算法是一种 TD 学习算法,其基于一步观察的学习规则,可以表示为

$$Q(s,a)\leftarrow Q(s,a)+\alpha\left[R(s,a)+\gamma\max_{a'}\hat{Q}(s',a')-Q(s,a)\right] \tag{5}$$

只要系统可以被建模为一个确定性 Markov 过程,回报函数有界,并且动作的选择可使每个状态动作对被无限地访问,那么经过充分训练以后,信息会从回报非零的状态向后传播到整个状态行为空间,最终得到一个 Q 表。文献[10]证明了 Q-学习算法的收敛性:

a) 对每个 s、a,初始化表项 $\hat{Q}(s,a)$ 为 0
b) 观察当前状态 s
c) 循环:
　(a) 选择一个动作 a 并执行它
　(b) 接收到立即回报 r
　(c) 观察新状态 s'
　(d) 对 $\hat{Q}(s,a)$ 按照下式更新表项:
　$\hat{Q}(s,a)\leftarrow r+\gamma\max_{a'}\hat{Q}(s',a')$
　(e) $s\leftarrow s'$

1.2　最小二乘策略迭代算法

最小二乘法(LS) 本质是欧氏空间的最优化估计方法。式(1) 写成矩阵形式可以表示为

$$Q^{\pi}=R+\gamma P\prod_{\pi}Q^{\pi} \tag{6}$$

LSPI 目标是用一组线性基函数$\{\Phi_i(s,a)\}$来逼近最优策略下的期望总回报 $Q(s,a)$。引入系数 w 的 $Q(s,a)$ 估计值可以表示为

$$\hat{Q}(s,a,w)=\sum_{i=1}^{L}\phi_i(s,a)\,w_i \tag{7}$$

由于状态行为值函数 Q^{π} 是 Bellman 算子 T_{π} 的不动点:

$$(T_{\pi},Q)(s,a)=R(s,a)+\gamma\sum_{s'\in S}P(s,a,s')\sum_{a'\in A}\pi(a';s')\,Q^{\pi}(s',a') \tag{8}$$

能高精度逼近 $Q(s,a)$ 的 $\hat{Q}(s,a,w)$ 也应满足 Bellman 算子 T_{π} 的不动点条件,即

$$T_{\pi}\hat{Q}^{\pi}\approx\hat{Q}^{\pi} \tag{9}$$

令

$$\phi(s,a)=\begin{pmatrix}\phi_1(s,a)\\ \phi_2(s,a)\\ \vdots\\ \phi_L(s,a)\end{pmatrix}\qquad \Phi=\begin{pmatrix}\phi(s_1,a_1)^{\mathrm{T}}\\ \vdots\\ \phi(s,a)^{\mathrm{T}}\\ \vdots\\ \phi(s_{|S|},a_{|A|})^{\mathrm{T}}\end{pmatrix}$$

$$\hat{Q}(w)=\begin{pmatrix}\hat{Q}(s_1,a_1,w)^{\mathrm{T}}\\ \vdots\\ \hat{Q}(s,a,w)^{\mathrm{T}}\\ \vdots\\ \hat{Q}(s_{|S|},a_{|A|},w)^{\mathrm{T}}\end{pmatrix} \tag{10}$$

则 $R+\gamma P\prod_{\pi}\hat{Q}(w)$ 在$\{\hat{Q}(w)\}$空间上的投影为 $\phi(\phi^{\mathrm{T}}\phi)^{-1}\phi^{\mathrm{T}}(R+\gamma P\prod_{\pi}\hat{Q}(w))$,所以基函数$\{\Phi_i(s,a)\}$的线性最小二乘不动点(fixed point) 解应满足:

$$\hat{Q}(w)=\phi(\phi^{\mathrm{T}}\phi)^{-1}\phi^{\mathrm{T}}(R+\gamma P\prod_{\pi}\hat{Q}(w)) \tag{11}$$

解得

$$w^{\pi}=(\phi^{\mathrm{T}}(\phi-\gamma P\prod_{\pi}\phi))^{-1}\phi^{\mathrm{T}}R \tag{12}$$

对于期望总回报估计值 $\hat{Q}(s,a,w)$,可以用贪婪算法求取最优策略:

$$\pi(s)=\arg\max_{a\in A}\hat{Q}(s,a,w) \tag{13}$$

由此可得 LSPI 的算法流程,如图 2 所示。

1.3　基于核的最小二乘策略迭代算法

基于核的 LSPI,通过在 LSPI 中引入核函数,使得近似策略迭代算法的非线性逼近能力大大加强。在 KLSPI 算法中,基函数是用基于核的特征表示:$\{\phi_i(s)=k(s,s_j)\}(0\leqslant j\leqslant d,i=1,2,\cdots,m)$。其中 $k(s,s_j)$ 是一个 Mercer 的核函数。对于有限的状态集合$\{s_1,s_2,\cdots,s_n\}$,核函数矩阵 $K=[k(s_i,s_j)]$ 是正定

的。根据 Mercer 理论，存在一个 Hilbert 空间 H 和一个从 S 到 H 的映射 ϕ，使得

$$k(s_i,s_j)=\langle\phi(s_i),\phi(s_j)\rangle \tag{14}$$

其中 $\langle\cdot,\cdot\rangle$ 是空间 H 上的内积。

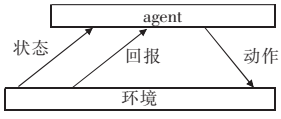

图 1 一个与环境交互的 agent

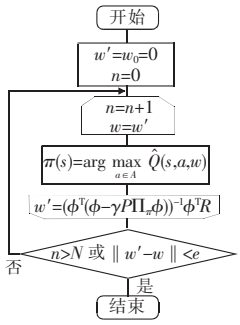

图 2 LSPI 算法流程图

2 在组合优化问题中的性能测试

旅行商问题(TSP)是一个约束性很强的组合优化问题，指的是从给定的城市集合中某一个城市从发，以最短的路径走遍所有的城市，要求每个城市必须走而且只能走一遍。传统的 TSP 解决方法以精确算法和各类启发式搜索算法为主，如最近邻启发式算法，贪婪启发式算法等。另外模拟退火算法[11]、遗传算法[12]等可以跳出局部最优，以便找到更优的路径。

但是随着问题规模的增加，以往的搜索算法可能存在时间代价、存储代价和计算代价太大等问题。文献[13]将神经网络与增强学习中的 Sarsa 算法相结合，用神经网络动态规划的方法对 TSP 进行了求解。实验结果在收敛速度等性能方面与以往传统算法相比有较大程度的改进。此外，将增强学习算法本身运用于解决约束性很强的组合优化问题，并且实验结果令人鼓舞，这在推动机器学习算法的工程运用特别是组合优化问题中的应用具有重要的理论价值和积极的探索意义。尽管如此，应该看到神经网络方法本身在参数选择、隐层节点配置等方面缺乏理论依据，存在一定的缺陷，而且也很难从理论上对其收敛性加以证明。LSPI 和 KLSPI 基函数便于依据模型特点进行选择，其收敛性也得到了充分论证[8,9]。本章分别将 Q-学习算法、LSPI 和 KLSPI 用于 TSP 的仿真研究和分析。

将问题构建为一个 MDP 模型如下：

a) 状态空间及状态。用一个 $n+1$ 维数组表示当前状态。其中前 n 维用二进制数表示，0 表示数组中其索引对应的城市已走过，1 表示还没有走或者当前正处于该索引对应的城市；第 $n+1$ 维用十进制数表示当前所在的城市编号[13]。

b) 动作空间。初始动作空间为 $A=\{1,2,\cdots,n\}$，action $=I$ $(i\in A)$ 表示当前状态下采取的动作。尔后每执行一个动作，下一状态可执行的动作集合总数减 1，直至走完所有的城市，对动作空间及状态进行重置。状态和动作表示以及状态转移关系如图 3 所示。

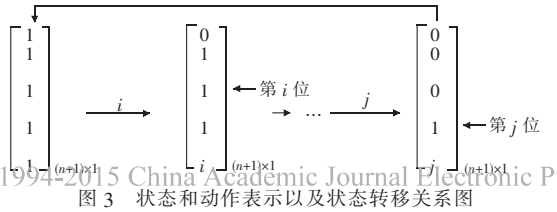

图 3 状态和动作表示以及状态转移关系图

本文针对 10 个城市的 TSP 进行了测试研究。城市编号分别为 1 ~ 10。为便于分析，固定从编号为 1 的城市出发。用 0 ~ 200的随机数表示每个城市的横坐标和纵坐标。

实验参数设置如下：

a) Q-学习。当前状态 s 下所在的城市到下一个城市(城市编号对应动作之间的距离表示一步立即回报 $R=R(s,a)$；折扣因子 $\gamma=1$；学习率为 0.02；最大周期数为 5 000；一个周期指的是从 1 号城市出发走完一条完整路径；贪婪因子为当前周期/最大周期。

b) LSPI。仿真实验中，在状态空间和动作空间的联合空间上均匀采样，共产生了 20 000 条路径样本(一条路径样本指从 1 号城市出发，按要求走完所有城市的一条完整路径)。LSPI 算法参数设定：最大迭代次数为 10，迭代终止误差 1e-005，折扣因子为 0.9，基函数的选择为状态多项式的线性组合，即 $\{1,s,s^2,s^3\}$。

c) KLSPI。KLSPI 算法的参数设置与 LSPI 相同，只是基函数的选择不同。本实验中选择的核函数为

$$k(s,s_0)=e^{-\frac{(s-s_0)^2}{400^2}} \tag{15}$$

核辞典的稀疏化过程如下：由于问题的状态空间规模上限为 $2^{(n-1)}\times(n-1)$，动作空间规模为 $n-1$(n 为城市总数)，将核辞典在状态空间中进行均匀配置，使每两个径向基函数(RBF)中心间隔 200 个状态，那么 10 个城市的 TSP 问题就可以得到 23 个核函数。Q-学习中学习周期对应的路程代价曲线如图 4 所示。从图 4 可以看出，Q-学习算法用于解决小规模的组合优化问题具有很好的收敛性，存储代价和计算消耗可以忍受，经过足够多的训练总可以收敛到最优策略或次优策略。但此类问题随着规模的增大，时间代价和存储代价增长得很快：本实验中 10 个城市的 TSP 问题最优路径搜索需进行3 000 ~ 4 000个学习周期，耗时 3 s 左右。Q-学习结束以后搜索到的最终路径如图 5 所示。

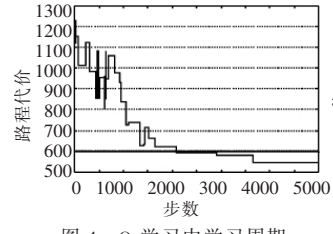

图 4 Q-学习中学习周期对应的路程代价曲线

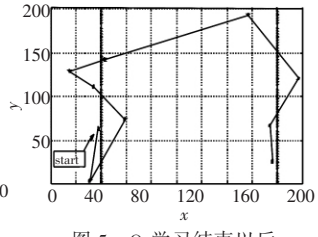

图 5 Q-学习结束以后搜索到的最终路径

图 6 分别采用 4 次多项式作为基函数的 LSPI 算法和高斯核函数作为基函数的 KLSPI 算法对上述的 10 个城市 TSP 进行求解，每种算法都用完全随机策略收集路径样本。可以看出，近似策略迭代算法可以只经过少量迭代较快地收敛到次优解。近似策略迭代算法得到的最终策略与值函数的逼近程度有很大关系。但是由于 TSP 本身状态行为值函数 $Q(s,a)$ 关于状态 s 和动作 a 不平滑，无论是多项式还是核函数作为基函数，都很难以满足要求的精度逼近这种情况下的状态行为值函数 Q，进而导致了近似迭代策略算法得到的最终策略较其他算法得到的策略效果差；而且值函数不平滑也导致了在每一次新的迭代中对最优值函数的逼近程度有所提高，但在相同状态下得到的策略依然不变的问题，如表 1 所示。当然，也应该看到在其他条件相同的情况下，由于核函数相对多项式函数具有更强的逼近能力，经过相同次数的策略迭代，KLSPI 算法得到的解总

是优于 LSPI 算法得到的解。

表 1　LSPI 和 KLSPI 算法在迭代中策略变化与最终的路程代价关系

算法		迭代次数					
		1	2	3	4	5	6
LSPI	$\|w_{t+1}-w_t\|_2$	6729.39	3217.90	76.80	6.00	0.36	0
	总路程代价	937.1440	928.8255	928.8255	928.8255	928.8255	928.8255
KLSPI	$\|w_{t+1}-w_t\|_2$	100564286	46267354	500500	177619	46258	0
	总路程代价	858.7286	823.4959	823.4959	823.4959	747.3246	747.3246

3　在运动控制问题中的性能测试

Mountain-Car 问题[14] 是增强学习中测试算法性能的标准问题之一[9,15]。问题描述为: 初始时刻,动力不足的小车位于波谷中,由于重力的影响,小车无法直接达到波峰上的目标点。要想达到目标点,小车必须在波谷中来回运动,积蓄足够的能量再冲到波峰,如图 7 所示。

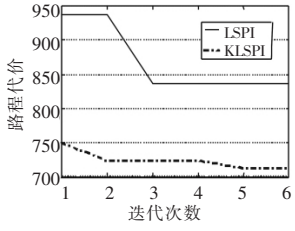

图 6　近似策略迭代算法的迭代次数与路程代价曲线

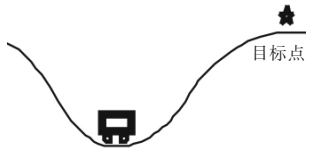

图 7　Mountain-Car 问题

Mountain-Car 问题有两个连续的状态分量: $s_t = [p_t, v_t]$。其中 p_t 为小车的位置,v_t 为小车的速度。动作定义为 $a \in \{-1, 1\}$,分别对应为向左加速和向右加速。山坡的地形模型简化为 $h = \sin(3p_t)$。其中 h 为高度。车体的简化动力学模型为

$$\begin{cases} v_{t+1} = \text{bound}\left[v_t + 0.001a_t + g\cos(3p_t)\right] \\ p_{t+1} = \text{bound}\left[p_t + v_t\right] \end{cases} \tag{16}$$

其中: $p \in [-1.5, 0.5]$; $v \in [-0.07, 0.07]$; bound$(\cdot)$ 是范围限制函数; 重力加速度 $g = -0.0025$。当位置 $p_{t+1} = -1.5$ 时,速度 $v_{t+1} = 0$。位置 $p = -0.5$ 对应于山谷的最低点,小车需要从这一点以初速度 0 出发,以最短时间到达图 7 中的目标点处,此时 $p^* = 0.45$。设计 Mountain-Car 问题的回报信号为

$$r_t(p_t, a_t, p_{t+1}) = \begin{cases} -1 & \text{if } p_{t+1} < p^* \\ 100 & \text{if } p_{t+1} \geqslant p^* \end{cases} \tag{17}$$

增强学习使得累计折扣回报达到最大,也就是使小车达到目标点所花费的时间最短。

实验参数设置如下:

a) Q 学习。在 Mountain-Car 问题中,虽然动作空间是离散的两个值 -1 和 1,但是两个状态分量却是连续的,要想建立 Q 表进行学习,必须进行状态离散划分。实验中选定一些状态空间中的点,它们的个数也就是离散化后状态的数目。对于某一状态 $s_t = [p_t, v_t]$,计算它与这些点的欧式距离,并把它归入到与它最近的那个点所属的状态。这些点的选择为 p 和 v 的组合,共 60 个状态。

$$p \in \{-1.5, -1.295, -1.09, -0.885, -0.68, -0.475, -0.27, -0.065, 0.14, 0.345\}$$

$$v \in \{-0.07, -0.042, -0.014, 0.014, 0.042, 0.07\}$$

其他的学习参数设置如下: 最大周期数为 201,每个周期内最大的运行步数为 1 000,学习率为 0.5,折扣因子为 1.0,随机动作选择的概率为 0.01。

b) LSPI。仿真实验中,在状态空间和动作空间的联合空间上均匀采样,共产生了 5 000 个样本。LSPI 算法参数设定: 最大迭代次数为 10,迭代终止误差 1e-007,折扣因子为 0.9,基函数的选择为两个状态分量三次多项式的组合,即$\{1, p, p^2, p^3\}$和$\{1, v, v^2, v^3\}$的组合,共计 16 个基函数。

c) KLSPI。该算法参数设置与 LSPI 算法相同,只是基函数的选择不同。在 KLSPI 中,核函数的选择为

$$k(s, s_0) = e^{-\left[\frac{(p-p_0)^2}{0.5^2} + \frac{(v-v_0)^2}{0.05^2}\right]} \tag{18}$$

其中: $s = [p, v]$, $s_0 = [p_0, v_0]$。稀疏化参数为 0.05,共得到 23 个核函数。

在 Mountain-Car 的运动控制问题中,尽管 Q-学习可以收敛到最终 Q^*,但是时间代价太大,需要经过 100 次学习周期以上才可以收敛到最终策略,如图 8 所示。LSPI 和 KLSPI 算法由于使用基函数$\{\Phi_i(s, a)\}$对状态行为值函数进行逼近,经过少量迭代均可收敛到次优策略对应的值函数,收敛效率也较 Q-学习算法大大提高,如图 9 所示。从图 8、9 学习过程曲线对比中可以看出,近似策略迭代算法得到的结果比 Q-学习算法得到的最终策略还优,这主要是因为此类运动控制问题状态行为值函数比第 2 章中提到的组合优化问题的状态行为值函数平滑,这就为算法的高效率逼近最优 Q^* 提供了保证。此外,由于核函数构成的逼近器的逼近能力较多项式逼近器的逼近能力有很大程度提高,KLSPI 算法与 LSPI 算法相比,在收敛速度、稳定性等方面表现出更好的性能,如图 10 所示。

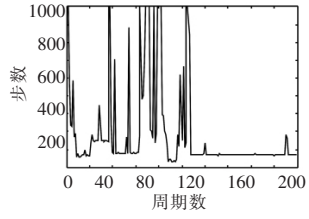

图 8　Q-学习的学习过程曲线

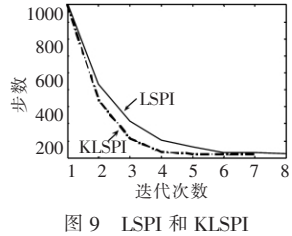

图 9　LSPI 和 KLSPI 迭代过程曲线

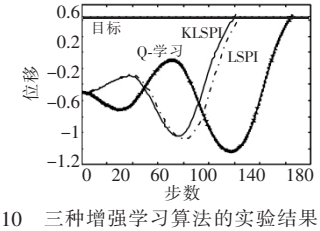

图 10　三种增强学习算法的实验结果比较

4　结束语

由三种典型算法在两类问题上分别进行的性能测试以及第 2 和 3 章进行的实验对比,可以得出以下结论:

a) 对于离散状态和行为空间问题,Q-学习只要经过足够多的训练,在理论上可以收敛到最优解,但是时间和空间代价往往较大,而且在面对大规模状态行为空间问题时,Q 表难以解决维数灾难问题。

b) 近似策略迭代算法具有较好的稳定性和收敛性,但是 LSPI 算法的线性基函数存在逼近能力有限的问题,针对一些大规模状态行为空间问题很难找到最优解或次优解。KLSPI 由于引入了核函数,可以较好地逼近最优策略对应的状态行为值函数,改善了 LSPI 算法存在的问题。

c) 通过对状态行为值函数非平滑的 TSP 和值函数平滑的 Mountain-Car问题进行测试,由不同算法得到的最终策略比较可以看出,MDP 值函数的平滑程度是影响　(下转第 3669 页)

行比较,文中给出了对 10 000 个实例运行摘叶算法预处理程序后得到的等价实例变量缩减的规模在不同约束方程密度取值情况下的变化情况(图 2)以及 LGC 算法与去掉摘叶预处理算法模块和高斯消去预处理算法模块后子完全算法(CB 算法)的平均执行时间的对比(表 2)。

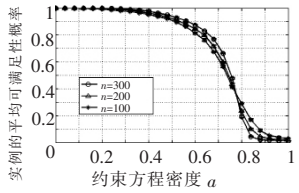

图 1　不同规模随机实例平均可满足概率相图

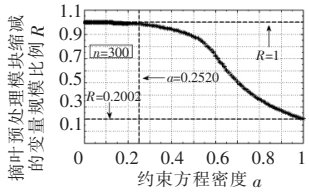

图 2　LF procedure所消减的变量规模比例

表 2　LGC 算法与 CB 算法计算 1 000 个实例消耗时间比值

实例变量规模 n	$a=0.3$	$a=0.5$	$a=0.7$
$n=100$	211.335 7	522.358 1	73.750 2
$n=200$	268.159 0	680.784 9	64.318 7

5　结束语

本文研究了等概率混合线性和非线性布尔方程条件下 MAS 模型的可满足概率随着约束方程密度增加而出现的可满足性相变现象。在构建高斯消去预处理算法和摘叶预处理算法的基础上,提出了一种新的针对 MAS 模型的完全求解算法,并进一步用该算法计算了在不同约束方程密度参数条件下大量随机实例中可满足实例所占的比例,通过不同参数条件下大量随机实例求解的数值实验,得到了可满足性阈值的算法估计。

参考文献:

[1] HARTMANN A K, WEIGT M. Phase transitions in combinatorial optimization problems: basics, algorithms and statistical mechanics [M]. [S. l.]: Wiley-VCH, 2005.

[2] FRIEDGUT E. Sharp thresholds of graph properties, and the k-SAT problem [J]. Journal of the American Mathematical Society, 1999, 12(4): 1017-1054.

[3] MONASSON R, ZECCHINA R. Statistical mechanics of the random k-SAT model [J]. Physical Review E, 1997, 56(2): 1357-1370.

[4] MEZARD M, RICCI-TERSENGHI F, ZECCHINA R. Two solutions to diluted p-spin models and XORSAT problems [J]. Journal of Statistical Physics, 2003, 111(3-4): 505-533.

[5] CHEESEMAN P, KANEFSKY B, TAYLOR W M. Where are the really hard problems [C] //Proc of the 12th International Joint Conference on Artificial Intelligence. San Francisco: Morgan Kaufmann, 1991.

[6] KIRKPATRICK S, SELMAN B. Critical behavior in the satisfiability of random Boolean expressions [J]. Science, 1994, 264(5163): 1297-1301.

[7] MONASSON R, ZECCHINA R, KIRKPATRICK S, *et al*. Determining computational complexity from characteristic "phase transitions" [J]. Nature, 1999, 400(6740): 133-137.

[8] MEZARD M, PARISI G, ZECCHINA R. Analytic and algorithmic solutions of random satisfiability problems [J]. Science, 2002, 297(5582): 812-815.

[9] KRZAKALA F, MONTANARI A, RICCI-TERSENGHI F, *et al*. Gibbs states and the set of solutions of random constraint satisfaction problems [J]. Proceedings of the National Academy of Sciences of the United States of America, 2007, 104(25): 10318-10323.

[10] BULIRSCH R, STOER J. Introduction to numerical analysis [M]. New York: Wiley, 1989.

[11] WEI Wei, GUO Bing-hui, ZHENG Zhi-ming. Statistical and algebraic analysis of a family of random Boolean equations [J]. Journal of Statistical Mechanics: Theory and Experiment, 2009(2): 2010.

[12] GUO Bing-hui, WEI Wei, SUN Y, *et al*. Algebraic characteristics and satisfiability threshold of random Boolean equations [J]. Physical Review E, 2010, 81(031122): 1-10.

(上接第 3665 页) 近似策略迭代算法性能表现的重要因素。针对状态行为值函数平滑的控制问题,使用近似策略迭代算法特别是 KLSPI 算法能够以较高的速度和精度逼近最优值函数或者策略,达到对目标的决策控制;但是对于值函数不平滑的问题,基函数很难对样本点邻近的点(s,a)的值函数进行较为精确的描述和表达,所以在依据贪心策略求取最优动作策略时就很难保证以较好的时间代价收敛到最优策略或次优策略。

本文的后续工作将针对一些值函数非平滑的组合优化问题,特别是问题的模型信息不完备的一些规划问题,考虑将增强学习算法与其他启发式算法相结合,这样既可以提高学习效率,又可以通过学习积累知识,不断改进策略,实现决策或控制的优化目标。

参考文献:

[1] KAELBLING L P, LITTMAN M L, MOORE A W. Reinforcement learning: a survey [J]. Journal of Artificial Intelligence Research, 1996, 4: 237-285.

[2] TESAURO G. TD-Gammon: a self-teaching back-Gammon program, achieves master-level play [J]. Neural Computation, 1994, 6(2): 215-219.

[3] ZHANG Wei, DIETTERICH T G. A reinforcement learning approach to job-shop scheduling [C] //Proc of the 14th International Joint Conference on Artificial Intelligence. San Francisco, CA: Morgan Kaufmann, 1995: 1114-1120.

[4] CRITES R H, BARTO A G. Elevator group control using multiple reinforcement learning agents [J]. Machine Learning, 1998, 33(2-3): 235-262.

[5] NG A Y, KIM H J, JORDAN M I, *et al*. Autonomous helicopter flight via reinforcement learning [C] //Proc of NIPS 16. 2004: 799-806.

[6] FU W T, ANDERSON J R. From recurrent choice to skill learning: a reinforcement-learning model [J]. Journal of Experimental Psychology, 2006, 135(2): 184-206.

[7] SUTTON R S, BARTO A G. Reinforcement learning: an introduction [M]. Cambridge, MA: MIT Press, 1998.

[8] LAGOUDAKIS M G, PARR R. Least-squares policy iteration [J]. Journal of Machine Learning Research, 2003, 4: 1107-1149.

[9] XU Xin, HU De-wen, LU Xi-cheng. Kernel based least-squares policy iteration [J]. IEEE Trans on Neural Networks, 2007, 18(4): 973-992.

[10] WATKINS C J C H, DAYAN P. Q learning [J]. Machine Learning, 1992, 8(3-4): 279-292.

[11] LAARHOVEN P, AARTS E. Simulated annealing: theory and applications [M]. Kluwer, Norwelt: [s. n.], 1987.

[12] HOMAIFAR A, GUAN S, LIEPINS G E. A new approach on the traveling salesman problem by genetic algorithms [C] //Proc of the 5th International Conference on Genetic Algorithms. San Francisco: Morgan Kaufmann, 1993: 460-466.

[13] MA Jia, YANG Tao, HOU Zeng-guang, *et al*. Nerodynamic programming: a case study of the traveling salesman problem [J]. Neural Comput & Applic, 2008, 17(4): 347-355.

[14] 文锋. 基于自适应评价者设计方法的学习控制研究 [D]. 合肥: 中国科学技术大学, 2005.

[15] SINGH S P, SUTTON R S. Reinforcement learning with replacing eligibility traces [J]. Machine Learning, 1996, 22(1-3): 123-158.