文章编号: 1000-0887(2007)03-0369-10

风险敏感度激励学习的广义平均算法*

殷苌茗[1,2], 王汉兴[2], 赵 飞[2]

(1. 长沙理工大学 计算机与通信工程学院, 长沙 410076;
2. 上海大学 理学院 数学系, 上海 200444)

(郭兴明推荐)

摘要: 提出了一种新的算法. 这个算法通过潜在地牺牲控制策略的最优性来获取其鲁棒性. 这是因为,如果在理论模型与实际的物理系统之间存在不匹配,或者实际系统是非静态的,或者控制动作的可使用性随时间的变化而变化时,那么鲁棒性就可能成为一个十分重要的问题. 主要工作是给出了一组逼近算法和它们的收敛结果. 利用广义平均算子来替代最优算子 max(或 min),对激励学习中的一类最重要的算法——动态规划算法——进行了研究,并讨论了它们的收敛性,目的就是为了提高激励学习算法的鲁棒性. 同时使用了更具一般性的风险敏感度性能评价体系,发现基于动态规划的学习算法中的一般结论在这种体系之下并不完全成立.

关 键 词: 激励学习; 风险敏感度; 广义平均; 算法; 收敛性
中图分类号: O23; TP182 文献标识码: A

引 言

目前智能体(Agent)的激励学习研究取得了很大的进展[1-5],已提出了许多以坚实数学理论为基础的各种学习与规划算法[6-8]. 激励学习的目标是找到一个最优控制策略,也就是要找到一个函数,为系统的某个状态指定一个动作(或决策),以优化目标函数,如最小时间、最大奖赏和最小费用等. 因此策略是一个动作序列[3],记所有这样的动作序列的集合为 F. 但是在许多控制问题中,最优解可能不是永久的,也就是说,Agent 系统在某时刻某状态下获得了一个计算最优动作的解,但随着时间与状态的不断变化,最优解可能不再是最优的了[9]. 在本文中,我们提出了一种新的算法,这个算法通过潜在地牺牲最优性来获取鲁棒性. 这是因为,如果在理论模型与实际的物理系统之间存在不匹配的情况,或者实际系统是非静态的,或者控制动作的可使用性随时间的变化而变化时,那么鲁棒性就可能成为一个十分重要的问题[2],[10]. 本文的主要工作是给出了一组逼近算法和它们的收敛结果. 利用广义平均算子来替代最优算子 max(或 min),对激励学习中的一类最重要的算法——动态规划算法——进行了研究,并讨论了它们的收敛性,目的就是为了提高激励学习算法的鲁棒性. 同时使用了更具一般性的风险敏感度性能评价体系. 关于风险敏感度问题有许多研究者进行过探讨,如 Cavazos-Cadena R 和 Montes-de-Oca R 对风险敏感度 Markov 决策过程问题进行了长期的研究,并且得到了一系

* 收稿日期: 2006-02-20; 修订日期: 2007-01-16
基金项目: 国家自然科学基金资助项目(10471088; 60572126)
作者简介: 殷苌茗(1964—),男,湖南人,副教授,博士(联系人. Tel: +86-731-5542939; E-mail: yinchm@csust.edu.cn).

列的好结果[11]. 他们得到的这些结果对激励学习算法的理论研究非常有作用. SatinderSingh 利用广义平均算子对动态规划的求解问题进行过研究[10]. 他的这种思想对本文中所讨论的问题非常有益. 在本文中, 我们发现基于动态规划的学习算法中的一般结论在这种体系之下并不完全成立. 这是我们使用风险敏感度性能评价体系所付出的代价. 但在逼近算法中, 并不影响对逼近最优解的求解. 另外, 我们没有试图对算法的实现过程给出具体的程序伪代码, 也没有对算法进行仿真实验. 这些问题将是我们下一步的关于本算法的应用研究工作.

本文除引言外, 分为 4 节. 在第 1 节, 我们将给出险敏感度动态规划以及广义平均算子等一些基本概念, 同时给出了所要讨论的模型. 把主要的结果放在了第 2 节. 在这一节中, 给出了一组新的算法并讨论了它们的收敛性质. 第 3 节给出了策略空间最优问题的解决方法. 在文章的的最后, 对本文所涉及的内容进行了总结并提出了下一步将要解决的相关问题.

1 概念与模型

1.1 风险敏感度动态规划算法与 Bellman 方程

动态规划(DP) 提供了一种解决最优控制问题的方法. 动态规划方法的理论基础是 Bellman 提出的最优性原理[3]. 它把整体最优控制策略中的瞬时决策, 转换为一个决策序列[12]. 在寻求最优控制策略的过程中, 基于 DP 算法中所使用的是 max 算子[3]. 但是, max 算子只考虑了在某一状态下, 执行最佳动作后的最终结果, 而忽略了在决策过程中, 所有可能带来灾难的其它动作的影响[3]. 我们在这里介绍了一族递归逼近算法, 在这组算法里, 我们把 DP 算法中的 max 算子, 替换为 p 阶广义平均(即一个非线性加权 l_p 范数). 这种 DP 算法将会收敛于某一个解, 而且这个解比通常的 DP 解更具有鲁棒性. 我们证明了, 对于每个参数 $p \geqslant 1$, 相应的递归算法收敛于一个唯一的不动点. 当 p 递增时, 有更好的一致逼近性质, 而且当 $p \rightarrow \infty$ 时, 它会收敛于用 DP 方法求得的解. 本文的主要工作是给出了一组逼近算法和它们的收敛结果.

马尔科夫决策过程(MDP), 是关于最优控制问题的一种非常好的描述. 它对于激励学习(reinforcement learning) 中的探索求解过程来说非常重要. 这是因为许多激励学习方法, 都可以通过多步 MDP 来建立数学模型[3],[13]. 设 S 为系统的一组状态, A 为控制器可以使用的一组动作. 在状态 $x \in S$ 执行动作 $a \in A$, 所获奖赏为 $R(x, a)$, 并用 $P(x, a)(y)$ 表示从状态 x, 执行动作 a, 然后转移到下一个状态 y 的概率. 激励学习的目标函数为下面的基于风险敏感度的动态规划算子 J_n:

$$J_n = \frac{1}{\lambda}\lg\left(E\left(\exp\left(\lambda\sum_{n=0}^{n}\gamma^t R_t\right)\right)\right), \tag{1}$$

其中 $0 < \gamma < 1$ 为折扣因子, n 为学习所设计的步数, E 为数学期望算子, R_t 为在时间步 t 时的瞬时奖赏, λ 为风险敏感度系数, 其取值范围可以是整个非 0 实数集合[11]. 如果算子 J_n 的定义域为状态空间 A 或状态-动作空间 $A \times S$, 则分别为一般的值函数和 Q-函数[4],[12]. 这里使用折扣因子的 t 次幂 γ^t, 在于说明未来奖赏对总奖赏的重要程度. 在本文中, 我们将改进逼近 DP 算法, 来解决状态有限但步数无限的 MDP 问题.

对于静态策略 $\pi: S \rightarrow A$, 其目标函数是:

$$V_\lambda^*(x) = \frac{1}{\lambda}\lg\left(E^\pi\left(\exp\left(\lambda\sum_{t=0}^{\infty}\gamma^t R_t(x, a)\right)\right)\right), \tag{2}$$

其中 $R_t(x, a)$ 为当初始状态为 x, 执行动作 a, 在策略 π 的驱动下, 在 t 时刻 Agent 所获得的瞬

时奖赏. 要获得一个最优策略 π^*, 就是要对在每一状态所获得的奖赏进行最大化. 所有策略的集合用字母 F 表示. 下面简单地用 $V_\lambda^*(x)$ 来表示 $V_\lambda^{\pi^*}(x)$.

对于 MDP, Bellman 最优原理表示如下: 对所有 $x \in S$, 有

$$U_\lambda(V_\lambda^*(x)) = \max_{a\in A}\left[e^{\lambda R(x,a)}\sum_{y\in S}P(x,a)(y)\,U_\lambda(\gamma V_\lambda^*(y))\right], \qquad x \in S. \tag{3}$$

其中 U_λ 是风险敏感度 λ 有关的性能评价函数, 或称其为效用函数(utility function), 其具体的定义可以参见文献[11]的说明. 其计算方法常用的是值递归算法. 这是一种改进的递归算法, 它通过递归公式

$$U_\lambda(V_{\lambda,t+1}(x)) = W(U_\lambda(V_{\lambda,t}(x))) \tag{4}$$

来计算性能评价函数 U_λ, 进而达到计算最优值函数的目的. 其中 $V_{\lambda,t}(x)$ 是在第 t 次递归时 $V_\lambda^*(x)$ 的估计值, 而这里的动态规划算子 W 定义为如下函数 $W: \mathbf{R}^+ \to \mathbf{R}^+$:

$$W(V_{\lambda,t+1}(x)) = \max_{a\in A}\left[\frac{1}{\lambda}\lg\left(E_x^\pi\left[\exp\left(\lambda\sum_{t=0}\gamma^t R_t(s_t,a_t)\right)\Big|_{s_0=x}\right]\right)\right], \qquad x \in S, \tag{5}$$

或者改写为

$$W(U_\lambda(V_{\lambda,t+1}(x))) = \max_{a\in A}\left[e^{\lambda R(x,a)}\sum_{y\in S}P(x,a)(y)\,U_\lambda(\gamma V_{\lambda,t}(y))\right], \qquad y \in S, \tag{6}$$

其中 $\mathbf{R}^+$ 为正实数集, $|S|$ 为状态集 S 中元素的个数. 注意, 根据风险敏感度和性能评价函数的定义以及 Bellman 方程可知, 上面的两个式子是等价的[11]. 而在状态 x 处的最优(贪婪)策略 π^* 按下面的公式给出:

$$\pi^*(x) = \arg\max_{a\in A}\left[\frac{1}{\lambda}\lg\left(E_x^\pi\left[\exp\left(\lambda\sum_{t=0}\gamma^t R(s_t,a_t)\right)\Big|_{s_0=x}\right]\right)\right], \qquad x \in S, \tag{7}$$

为了对基于 DP 的算法更一般化, 保证算法的鲁棒性, 这里将把 max 算子用广义平均来代替. 首先我们给出广义平均的定义及其一些基本性质.

1.2 广义平均的定义及其一些基本性质

为了后面有关结论证明的需要, 在这里以引理的形式列出了关于广义平均的一些结论. 设有两个 n 维向量

$$\boldsymbol{A} = \{a_1, a_2, \cdots, a_n\},\ \boldsymbol{A}' = \{a_1', a_2', \cdots, a_n'\}, \tag{8}$$

定义

$$\boldsymbol{A}_{\max} = \max\{a_1, a_2, \cdots, a_n\},\ \|\boldsymbol{A}\|_\infty = \max\{|a_1|, |a_2|, \cdots, |a_n|\}, \tag{9}$$

还定义

$$A_p = \left[\frac{1}{n}\sum_{i=1}^{n}(a_i)^p\right]^{1/p}. \tag{10}$$

我们称 A_p 为 p 阶广义平均. 在 $1 \leqslant i \leqslant n$, a_i、$a_i' \in \mathbf{R}^+$ 的假设条件下, 下面的一些结论可参见一般的函数论书籍.

引理 1.1(收敛性) $\lim_{p\to\infty} A_p = \boldsymbol{A}_{\max}$.

引理 1.2(一致性) 若 $0 < p < q$ 则有 $A_p \leqslant A_q \leqslant \boldsymbol{A}_{\max}$; 而且若存在 i、j, 使得 $a_i \neq a_j$, $0 < p < q$, 则 $A_p < A_q < \boldsymbol{A}_{\max}$.

引理 1.3(单调性) 若对所有 i, $a_i \leqslant a_i'$, 则 $A_p \leqslant A_p'$; 另外若存在 i, 使得 $a_i < a_i'$, 则 $A_p < A_p'$.

引理 1.4(有界性) 若 $p \geqslant 1$, 且 $\|\boldsymbol{A} - \boldsymbol{A}'\| \leqslant M$, 则 $|A_p - A_p'| \leqslant M$; 另外, 若 $p > 1$ 且 $\boldsymbol{A} \neq \boldsymbol{A}'$, 若 $\|\boldsymbol{A} - \boldsymbol{A}'\| \leqslant M$, 则有 $|A_p - A_p'| < M$.

引理 1.5(线性性) 对任意的常数 c, 有$(cA)_p = cA_p$.

2 基于动态规划风险敏感度的递归不动点

改进的广义平均递归算法是用一个标量参数 p 作下标, 其递归计算公式定义为: 对所有的状态 $x \in S$,

$$U_\lambda(V_{\lambda,t+1}(x)) = W_p(U_\lambda(V_{\lambda,t}(x))), \tag{11}$$

其中这里的更新算子为 $W_p: \mathbf{R}^+ \to \mathbf{R}^+$: 对所有状态 $x \in S$,

$$W_p(U_\lambda(V_{\lambda,t}(x))) = \left\{\frac{1}{|A|}\sum_{a\in A}\left[e^{\lambda R(x,a)}\sum_{y\in S}P(x,a)(y)U_\lambda(\gamma V_{\lambda,t-1}(y))\right]^p\right\}^{1/p}. \tag{12}$$

这个算子就是我们所说的基于风险敏感度的广义平均算子. 因此通过这种递归计算, 即可得到一个关于值函数的序列. 我们将证明, 这个序列在一定的条件下将具有收敛性质. 由此收敛性质就可以知道, 最优逼近策略是存在的, 并且可以通过这种递归方法得到它. 这种松弛的(relaxed) 动态规划方法和一般的动态规划方法比较, 具有一些不同的性质. 下面是关于这个算子的一些收敛性质.

性质 2.1 $\lim_p W_p = W$.

证明 利用广义平均算子的收敛性质(见引理 1.1) 以及算子 W_p 的定义, 对所有 $x \in S$, 有

$$\begin{aligned}
&\lim_p(W_p(U_\lambda(V_{\lambda,t+1}(x)))) = \\
&\quad\lim_p\left\{\frac{1}{|A|}\sum_{a\in A}\left[e^{\lambda R(x,a)}\sum_{y\in S}P(x,a)(y)U_\lambda(\gamma V_{\lambda,t}(y))\right]^p\right\}^{1/p} = \\
&\quad\lim_p\left[e^{\lambda R(x,a)}\sum_{y\in S}P(x,a)(y)U_\lambda(\gamma V_{\lambda,t}(y))\right] = \\
&\quad W(U_\lambda(V_{\lambda,t+1}(x))).
\end{aligned}$$

因此结论成立.

这个性质说明了本文定义的广义动态规划算子和一般的动态规划算子的关系. 下面的性质说明广义动态规划算子具有线性性质.

性质 2.2 对任意的参数 $p \geqslant 1$ 以及任意的正常数 M, 基于风险敏感度的广义平均算子 W_p 有下面的结论成立: 对所有的状态 $x \in S$, 以及任意的策略 $\pi \in F$,

$$W_p(U_\lambda(V^\pi_{\lambda,t}(x)+M)) = e^{\gamma\lambda M}W_p(U_\lambda(V^\pi_{\lambda,t}(x))). \tag{13}$$

证明 可以直接证明这个结论. 这是因为

$$\begin{aligned}
&W_p(U_\lambda(V^\pi_{\lambda,t+1}(x)+M)) = \\
&\quad\left\{\frac{1}{|A|}\sum_{a\in A}\left[e^{\lambda R(x,a)}\sum_{y\in S}P(x,a)(y)U_\lambda(\gamma(V^\pi_{\lambda,t}(y)+M))\right]^p\right\}^{1/p} = \\
&\quad\left\{\frac{1}{|A|}\sum_{a\in A}\left[e^{\lambda R(x,a)}\sum_{y\in S}P(x,a)(y)U_\lambda(\gamma M+\gamma V^\pi_{\lambda,t}(y))\right]^p\right\}^{1/p} = \\
&\quad\left\{\frac{1}{|A|}\sum_{a\in A}\left[e^{\lambda R(x,a)}\sum_{y\in S}P(x,a)(y)e^{\gamma\lambda M}U_\lambda(\gamma V^\pi_{\lambda,t}(y))\right]^p\right\}^{1/p} = \\
&\quad e^{\gamma\lambda M}\left\{\frac{1}{|A|}\sum_{a\in A}\left[e^{\lambda R(x,a)}\sum_{y\in S}P(x,a)(y)U_\lambda(\gamma V^\pi_{\lambda,t}(y))\right]^p\right\}^{1/p} = \\
&\quad e^{\gamma\lambda M}W_p(U_\lambda(V^\pi_{\lambda,t+1}(x))).
\end{aligned}$$

因此结论成立.

引理 2.1 对任意的策略 $\pi \in F$, 当风险敏感度系数 $\lambda \neq 0$ 时, 对所有状态 $x \in S$, 性能评价函数 $U_\lambda(V_\lambda^\pi(x))$ 是关于策略 π 的一个单调函数.

证明 因为对所有 $x \in S$, 当 $V_\lambda^\pi(x) \geqslant V_\lambda^{\pi'}(x)$ 时, 有

$$V_\lambda^\pi - V_\lambda^{\pi'} = \frac{1}{\lambda}\lg E^\pi\left[\exp\left(\lambda\sum_{t=0}\gamma^t R(s_t, a_t)\right)\Big|_{s_0=x}\right] -$$
$$\frac{1}{\lambda}\lg E^{\pi'}\left[\exp\left(\lambda\sum_{k=0}\gamma^k R(s_k', a_k')\right)\Big|_{s_0=x}\right] =$$
$$\frac{1}{\lambda}\lg\frac{E^\pi\left[\exp\left(\lambda\sum_{t=0}\gamma^t R(s_t, a_t)\right)\Big|_{s_0=x}\right]}{E^{\pi'}\left[\exp\left(\lambda\sum_{k=0}\gamma^t R(s_t', a_t')\right)\Big|_{s_0=x}\right]} =$$
$$\frac{1}{\lambda}\lg\frac{e^{\lambda R(x, a_0)}E^\pi\left[\exp\left(\lambda\sum_{t=1}\gamma^t R(s_t, a_t)\right)\right]}{e^{\lambda R(x, a_0')}E^{\pi'}\left[\exp\left(\lambda\sum_{t=1}\gamma^t R(s_t', a_t')\right)\right]} \geqslant 0.$$

因此当 $\lambda > 0$ 时, 有下面的不等式

$$\frac{e^{\lambda R(x, a_0)}E^\pi\left[\exp\left(\lambda\sum_{t=1}\gamma^t R(s_t, a_t)\right)\right]}{e^{\lambda R(x, a_0')}E^{\pi'}\left[\exp\left(\lambda\sum_{t=1}\gamma^t R(s_t', a_t')\right)\right]} \geqslant 1, \tag{14}$$

$$E^\pi\left[\exp\left(\lambda\sum_{t=1}\gamma^t R(s_t, a_t)\right)\right] \geqslant E^{\pi'}\left[\exp\left(\lambda\sum_{t=1}\gamma^t R(s_t', a_t')\right)\right]. \tag{15}$$

即分别对于任意的策略 π 和策略 π', 有下面的不等式成立

$$\exp\left(\lambda\sum_{t=1}\gamma^t R(s_t, a_t)\right) \geqslant \exp\left(\lambda\sum_{t=1}\gamma^t R(s_t', a_t')\right). \tag{16}$$

因此, 可以进行下面的推导过程:

$$U_\lambda(V_\lambda^\pi(x)) - U_\lambda(V_\lambda^{\pi'}(x)) =$$
$$E^\pi\left[U_\lambda\left(\sum_{t=0}\gamma^t R(s_t, a_t)\right)\Big|_{s_0=x}\right] - E^{\pi'}\left[U_\lambda\left(\sum_{t=0}\gamma^t R(s_t', a_t')\right)\Big|_{s_0=x}\right] =$$
$$e^{\lambda R(x, a_0)}E^\pi\left[U_\lambda\left(\sum_{t=1}\gamma^t R(s_t, a_t)\right)\right] - e^{\lambda R(x, a_0')}E^{\pi'}\left[U_\lambda\left(\sum_{t=1}\gamma^t R(s_t', a_t')\right)\right] =$$
$$e^{\lambda R(x, a_0)}\left\{E^\pi\left[U_\lambda\left(\sum_{t=1}\gamma^t R(s_t, a_t)\right)\right] - E^{\pi'}\left[U_\lambda\left(\sum_{t=1}\gamma^t R(s_t', a_t')\right)\right]\right\} =$$
$$e^{\lambda R(x, a_0)}\left[\sum_{i=0}P(s_i, a_i)(s_{i+1})\,U_\lambda\left(\sum_{t=i+1}\gamma^t R(s_t, a_t)\right) -\right.$$
$$\left.\sum_{j=0}P(s_j, a_j)(s_{j+1})\,U_\lambda\left(\sum_{t=j+1}\gamma^t R(s_t', a_t')\right)\right] =$$
$$e^{\lambda R(x, a_0)}\sum_{i=0}P(s_i, a_i)(s_{i+1})\left[U_\lambda\left(\sum_{t=i+1}\gamma^t R(s_t, a_t)\right) - U_\lambda\left(\sum_{t=i+1}\gamma^t R(s_t', a_t')\right)\right] \leqslant$$
$$e^{\lambda R(x, a_0)}\left[U_\lambda\left(\sum_{t=1}\gamma^t R(s_t, a_t)\right) - U_\lambda\left(\sum_{t=1}\gamma^t R(s_t', a_t')\right)\right] =$$
$$e^{\lambda R(x, a_0)}\,\mathrm{sign}(\lambda)\left[\exp\left(\lambda\sum_{t=1}\gamma^t R(s_t, a_t)\right) - \exp\left(\sum_{t=1}\gamma^t R(s_t', a_t')\right)\right].$$

因此当 $\lambda > 0$ 时, 此函数单调递增. 同理可以得到, 当 $\lambda < 0$ 时, 函数 $U_\lambda(V_\lambda^\pi(x))$ 关于策略 π 单调递减, 因此最后结论成立.

定理 2.1 对所有的 $p \geqslant 1$, 基于风险敏感度的广义平均算子 W_p 有下面的结论成立:

(a) (单调性) 对所有 $x \in S$, 任意 $V_\lambda(x)$, $V_\lambda'(x) \in \mathbf{R}^+$, 若 $V_\lambda(x) \leqslant V_\lambda'(x)$, 则有

$$W_p(U_\lambda(V_\lambda^\pi(x))) \leqslant W_p(U_\lambda(V_\lambda^{\pi'}(x)));$$ (17)

(b) (压缩定理) 对所有 $x \in S$, 任意有限的 $V_\lambda(x)$, $V'_\lambda(x) \in \mathbf{R}^+$, 当 $\lambda < 0$ 时有

$$\| W_p(U_\lambda(V_\lambda^\pi(x))) - W_p(U_\lambda(V_\lambda^{\pi'}(x))) \|_\infty \leqslant \alpha \| U_\lambda(V_\lambda^\pi(x)) - U_\lambda(V_\lambda^{\pi'}(x)) \|_\infty, \quad \forall x \in S, \tag{18}$$

其中 $0 < \alpha < 1$, 即当 $\lambda < 0$ 时 W_p 是一个压缩算子.

证明

(a) 直接从引理 2.1 和广义平均的单调性可以得出;

(b) 显然, 根据假设可知存在 $M: 0 \leqslant M < \infty$, 使得

$$\| U_\lambda(V_\lambda^\pi(x)) - U_\lambda(V_\lambda^{\pi'}(x)) \|_\infty \leqslant M. \tag{19}$$

因此对所有 $x \in S$, 有

$$-M \leqslant U_\lambda(V_\lambda^\pi(x)) - U_\lambda(V_\lambda^{\pi'}(x)) \leqslant M, \tag{20}$$

$$U_\lambda(V_\lambda^\pi(x)) \leqslant U_\lambda(V_\lambda^{\pi'}(x)) + M; \quad U_\lambda(V_\lambda^\pi(x)) \geqslant U_\lambda(V_\lambda^{\pi'}(x)) - M. \tag{21}$$

进而根据风险敏感度的性能评价函数的定义可知, 对于任意的常数 σ, 有

$$\begin{cases} U_\lambda(\sigma V_\lambda^\pi(x)) \leqslant U_\lambda(\sigma V_\lambda^{\pi'}(x)) + \mathrm{e}^{\lambda\sigma} M, \\ U_\lambda(\sigma V_\lambda^\pi(x)) \geqslant U_\lambda(\sigma V_\lambda^{\pi'}(x)) - \mathrm{e}^{\lambda\sigma} M. \end{cases} \tag{22}$$

在方程(12)中用 $U_\lambda(\sigma V_\lambda^\pi(y)) + \mathrm{e}^{\lambda\sigma} M$ 代替 $U_\lambda(\gamma V_\lambda^\pi(y))$, 可得

$$\begin{aligned} & W_p(U_\lambda(V_\lambda^\pi(x))) \leqslant \\ & \left\{ \frac{1}{|A|} \sum_{a \in A} \left[\mathrm{e}^{\lambda R(x,a)} \sum_{y \in S} P(x,a)(y) U_\lambda(\sigma V_\lambda^{\pi'}(y)) + \mathrm{e}^{\lambda\sigma} M \right]^p \right\}^{1/p} = \\ & \left\{ \frac{1}{|A|} \sum_{a \in A} \left[\mathrm{e}^{\lambda R(x,a)} \sum_{y \in S} \mathrm{e}^{\lambda\sigma} M P(x,a)(y) + \right. \right. \\ & \left. \left. \mathrm{e}^{\lambda R(x,a)} \sum_{y \in S} P(x,a)(y) U_\lambda(\sigma V_\lambda^{\pi'}(y)) \right]^p \right\}^{1/p} = \\ & \left\{ \frac{1}{|A|} \sum_{a \in A} \left[M \mathrm{e}^{\lambda R(x,a) + \lambda\sigma} \sum_{y \in S} P(x,a)(y) + \right. \right. \\ & \left. \left. \mathrm{e}^{\lambda R(x,a)} \sum_{y \in S} P(x,a)(y) U_\lambda(\sigma V_\lambda^{\pi'}(y)) \right]^p \right\}^{1/p}. \end{aligned}$$

又由于状态转换矩阵 $\boldsymbol{P}$ 是一个随机矩阵, 因此有

$$\begin{aligned} & W_p(U_\lambda(V_\lambda^\pi(x))) \leqslant \\ & \left\{ \frac{1}{|A|} \sum_{a \in A} \left[M \mathrm{e}^{\lambda R(x,a) + \lambda\sigma} + \mathrm{e}^{\lambda R(x,a)} \sum_{y \in S} P(x,a)(y) U_\lambda(\sigma V_\lambda^{\pi'}(y)) \right]^p \right\}^{1/p} \leqslant \\ & \left\{ \frac{1}{|A|} \sum_{a \in A} \left[\mathrm{e}^{\lambda R(x,a)} \sum_{y \in S} P(x,a)(y) U_\lambda(\sigma V_\lambda^{\pi'}(y)) \right]^p \right\}^{1/p} + M \mathrm{e}^{\lambda R(x,a) + \lambda\sigma} = \\ & W_p(U_\lambda(V_\lambda^{\pi'}(x))) + M \mathrm{e}^{\lambda R(x,a) + \lambda\sigma}. \end{aligned}$$

再根据对称性有

$$W_p(U_\lambda(V_\lambda^{\pi'}(x))) \leqslant W_p(U_\lambda(V_\lambda^\pi(x))) + M \mathrm{e}^{\lambda R(x,a) + \lambda\sigma}. \tag{23}$$

利用广义平均的有界性, 再由(17)式和(23)式以及性质 2.2, 可以导出: 对所有 $x \in S$, 有

$$\| W_p(U_\lambda(V_\lambda^\pi(x))) - W_p(U_\lambda(V_\lambda^{\pi'}(x))) \|_\infty \leqslant \mathrm{e}^{\lambda R(x,a) + \lambda\sigma} \| U_\lambda(V_\lambda^\pi(x)) - U_\lambda(V_\lambda^{\pi'}(x)) \|_\infty. \tag{24}$$

由于 $\lambda < 0$, 因此上式右边的系数在 0 和 1 之间, 因此只要令 $\alpha = \mathrm{e}^{\lambda R(x,a) + \lambda\sigma}$ 即可. 这样就证明了结论(b).

注意到这个定理的第 2 部分只对当 $\lambda < 0$ 时成立, 当 $\lambda > 0$ 时不具有压缩性质, 这说明当

风险追求时, 系统的激励学习算法将可能没有收敛性质. 尽管如果实际应用中的激励学习在这种情况下(当 $\lambda > 0$ 时) 不具有算法的收敛性, 但可以去讨论其稳定性. 其稳定性的研究将是我们将来的工作. 下面的定理解决了最优值函数的存在性问题.

定理 2.2 设折扣因子 $0 < \gamma < 1$, 且对所有状态 $x \in S$ 及动作 $a \in A$, $R(x, a) \geqslant 0$, 当 $\lambda < 0$ 时, 如果有初始估值 $V_{\lambda,0}(x) \in \mathbf{R}^+$, 则对于所有 $p \geqslant 1$, 当 t 趋于无穷大时, 递归

$$U_\lambda(V_{\lambda,t+1}(x)) = W_p(U_\lambda(V_{\lambda,t}(x))) \tag{25}$$

收敛于一个唯一的不动点 $U_\lambda(V_p^*(x))$.

证明 由定理 2.1(b) 知, 对所有 $x \in S$,

$$\| W_p(U_\lambda(V_{\lambda,t+1}(x))) - W_p(U_\lambda(V_{\lambda,t}(x))) \|_\infty =$$
$$\left\| \left\{ \frac{1}{|A|} \sum_{a \in A} \left[e^{\lambda R(x,a)} \sum_{y \in S} P(x,a)(y) U_\lambda(\gamma V_{\lambda,t}(y)) \right]^p \right\}^{1/p} - \left\{ \frac{1}{|A|} \sum_{a \in A} \left[e^{\lambda R(x,a)} \sum_{y \in S} P(x,a)(y) U_\lambda(\gamma V_{\lambda,t-1}(y)) \right]^p \right\}^{1/p} \right\|_\infty \leqslant$$
$$e^{\lambda R(x,a)} \| U_\lambda(V_{\lambda,t+1}(x)) - U_\lambda(V_{\lambda,t}(x)) \|_\infty .$$

由于 $\lambda < 0$, 因此满足压缩映像原理的条件, 从而存在唯一的不动点, 并自然地记这个唯一的不动点为 $U_\lambda(V_p^*(x))$. 于是由不动点收敛定理可知, 当 t 趋于无穷大时, 递归算法

$$U_\lambda(V_{\lambda,t+1}(x)) = W_p(U_\lambda(V_{\lambda,t}(x))) \tag{26}$$

收敛于这个唯一不动点 $U_\lambda(V_p^*(x))$. 证毕.

注意我们在这个不动点的表示中仍然使用了参数 p 作为下标, 是为了说明这个不动点是从算子 W_p 所导出来的, 并且是与参数 p 有关的函数.

这个定理的条件仍然要求 $\lambda < 0$, 这是把性能指标扩展到风险敏感度情况下所付出的代价, 因此其应用会受到一定的限制, 即它只适用于风险规避的情况. 当 $\lambda = 0$ 时, 即传统意义上的求解问题, 结论当然是成立的. 问题是在什么样的条件下, 当 $\lambda > 0$ 时结论也成立, 这也是一个值得研究的问题.

推论 2.1 对所有 $x \in S$, 设 $V_\lambda^*(x)$ 为最优值函数, 且当 $\lambda < 0$ 时, 则有

$$\lim_{p \to \infty} U_\lambda(V_p^*(x)) = U_\lambda(V_\lambda^*(x)), \tag{27}$$

即不动点将随 p 的增大无限接近于最优值函数.

证明 由定理 2.2, 由于 $U_\lambda(V_p^*(x))$ 是由

$$U_\lambda(V_{\lambda,t+1}(x)) = W_p(U_\lambda(V_{\lambda,t}(x))), \tag{28}$$

所定义的递归算法收敛的唯一不动点, 因此必定有

$$U_\lambda(V_p^*(x)) = W_p(U_\lambda(V_p^*(x))). \tag{29}$$

所以对所有 $x \in S$, 推导出

$$U_\lambda(V_p^*(x)) = \lim_t U_\lambda(V_{\lambda,t+1}(x)) = \lim_t W_p(U_\lambda(V_{\lambda,t}(x))) =$$
$$\lim_t \left\{ \frac{1}{|A|} \sum_{a \in A} \left[e^{\lambda R(x,a)} \sum_{y \in S} P(x,a)(y) U_\lambda(\gamma V_{\lambda,t-1}(y)) \right]^p \right\}^{1/p} =$$
$$\left\{ \frac{1}{|A|} \sum_{a \in A} \left[e^{\lambda R(x,a)} \sum_{y \in S} P(x,a)(y) \left(\lim_t U_\lambda(\gamma V_{\lambda,t-1}(y)) \right) \right]^p \right\}^{1/p} =$$
$$\left\{ \frac{1}{|A|} \sum_{a \in A} \left[e^{\lambda R(x,a)} \sum_{y \in S} P(x,a)(y) U_\lambda(\gamma V_\lambda^*(y)) \right]^p \right\}^{1/p} =$$
$$W_p(U_\lambda(V_p^*(x))).$$

所以, 上面两边对 p 取极限, 有

$$\lim_{p\to\infty} U_\lambda(V_p^*(x)) = \lim_{p\to\infty} W_p(U_\lambda(V_\lambda^*(x))) = W(U_\lambda(V_\lambda^*(x))) = \max_{a\in A}\left[e^{\lambda R(x,a)}\sum_{y\in S} P(x,a)(y)\,U_\lambda(V_\lambda^*(y))\right] = U_\lambda(V_\lambda^*(y)).$$

因此结论成立.

下面的定理给出了在不同算法之下不动点之间的关系. 从这个定理可以看出最优值函数能够通过逐次估值来逼近获得.

定理 2.3 若 $1 \leqslant p \leqslant q$, 二者都是实常数. 则对所有 $x \in S$, 有下面的结论成立:

$$U_\lambda(V_p^*(x)) \leqslant U_\lambda(V_q^*(x)) \leqslant U_\lambda(V_\lambda^*(x)). \tag{30}$$

证明 对任意的状态 $x \in S$, 下面我们通过对递归算法

$$U_\lambda(V_{p,t+1}(x)) = W_p(U_\lambda(V_{p,t}(x))) \tag{31}$$

以及 $$U_\lambda(V_{q,t+1}(x)) = W_p(U_\lambda(V_{q,t}(x))) \tag{32}$$

的估值的并行实现, 来考虑本定理的递归问题. 这里的逐次估值可以通过添加的下标 p 和 q 看出, 用 p 和 q 两个下标的目的是为了区别这两个算法, 同时省略了风险敏感度系数这个下标. 假设

$$U_\lambda(V_{p,0}(x)) = U_\lambda(V_{q,0}(x)) \leqslant U_\lambda(V_p^*(x)) \leqslant U_\lambda(V_\lambda^*(x)), \tag{33}$$

且 $$U_\lambda(V_{p,0}(x)) = U_\lambda(V_{q,0}(x)) \leqslant U_\lambda(V_q^*(x)) \leqslant U_\lambda(V_\lambda^*(x)). \tag{34}$$

由压缩性、单调性和一致性并作用于每个状态, 当 $1 \leqslant p \leqslant q$ 时, 易知有如下的结论:

$$U_\lambda(V_{p,0}(x)) = U_\lambda(V_{q,0}(x)); \quad U_\lambda(V_{p,1}(x)) \leqslant U_\lambda(V_{q,1}(x)),$$
$$U_\lambda(V_{p,2}(x)) \leqslant U_\lambda(V_{q,2}(x)), \ \cdots, \ U_\lambda(V_{p,t}(x)) \leqslant U_\lambda(V_{q,t}(x)).$$

由于 $$U_\lambda(V_{p,\infty}(x)) = U_\lambda(V_p^*(x)); \quad U_\lambda(V_{q,\infty}(x)) = U_\lambda(V_q^*(x)), \tag{35}$$

故存在 $V_{\lambda,0}(x) \in \mathbf{R}^+$, 使得对所有 $t > 0$, 有

$$U_\lambda(V_{p,t}(x)) \leqslant U_\lambda(V_{q,t}(x)) < U_\lambda(V_\lambda^*(x)). \tag{36}$$

因此由定理 2.2 知, 此不动点与初始估值 $V_{\lambda,0}(x)$ 无关, 所以有

$$U_\lambda(V_p^*(x)) \leqslant U_\lambda(V_q^*(x)) \leqslant U_\lambda(V_\lambda^*(x)) \tag{37}$$

(对任意的初始估值 $V_{\lambda,0}(x) \in \mathbf{R}^+$. 证毕.

推论 2.2 所有 $x \in S$, 对于任意的 $\varepsilon > 0$, 存在 $p \geqslant 1$, 使得对所有参数 $q > p$, 有

$$\| U_\lambda(V_q^*(x)) - U_\lambda(V_\lambda^*(x)) \|_\infty < \varepsilon. \tag{38}$$

证明 由定理 2.2 以及定理 2.3 的结论可知, 效用(值)函数序列 $\{U_\lambda(V_p^*(x))\}$ 有界并且收敛于 $U_\lambda(V_\lambda^*(x))$, 结论即可得证.

由此可知, 广义平均算子序列 $\{W_p\}$ 定义了一组递归逼近的值递归的算法. 由于参数 p 是递增的, 因此, 它逼近到最优值函数. 但是在实际的估值计算中我们可以不去考察最优值函数是怎样的逼近程度, 而是去考察由逼近导出的贪婪策略是否为最优的. 下面即考虑了贪婪策略的最优性问题.

3 策略空间的最优性

下面讨论系统所采取的策略的最优性问题. 一般的非风险敏感度情况可参见文献[10].

性质 3.1 对于任意的 Markov 决策过程, 且设给定的动作空间 A 有限, 对所有状态 $x \in S$, 如果存在 $\delta > 0$, 使得对任意 $V_\lambda(x) \in \mathbf{R}^+$, 有

$$\| U_\lambda(V_\lambda(x)) - U_\lambda(V_\lambda^*(x)) \|_\infty < \delta/2. \tag{39}$$

则关于 $V_\lambda(x)$ 的任何贪婪策略都是最优的.

证明　对所有 $x \in S$, 设 $A - \{\pi^*(x)\}$ 为系统处于状态 x 时的非最优动作集合, 其中 $\{\pi^*(x)\}$ 为最优动作的集合. 如果 $A - \{\pi^*(x)\}$ 不为空集, 那么

$$m(x) = \min_{a \in A - \{\pi^*(x)\}} | U_\lambda(V^*_\lambda(x, \pi^*(x))) - U_\lambda(V^*_\lambda(x, a)) | =$$
$$\min_{a \in A - \{\pi^*(x)\}} \Big| e^{\lambda R(x, \pi^*(x))} \sum_{y \in S} P(x, \pi^*(x))(y)\, U_\lambda(\gamma V^*_\lambda(y)) -$$
$$e^{\lambda R(x, a)} \sum_{y \in S} P(x, a)(y)\, U_\lambda(\gamma V^*_\lambda(y)) \Big| \tag{40}$$

必定为一个大于 0 的有限数. 令 $\delta = \min_{x \in S}\{m(x)\}$. 注意到仅当在系统的每个状态下, 动作都为最优时才有 $\delta = 0$, 因此可以把 δ 看作是一个任意的非0值. 于是由上式可知最优值函数小于 $\delta/2$, 而且在任意状态的贪婪动作仍然是最优的. 证毕.

由性质 3.1 可知, 只要被估值的值函数的值, 比最优值函数的值小 $\delta/2$, 那么由逼近导出的策略就会是最优的. 定义 F_p 为关于 $V^*_p(x)$ 的静态贪婪策略的集合, 如果 $\pi \in F_p$, 对所有 $x \in S$, $a \in A$, 那么执行动作 $\pi(x)$ 的立即奖赏, 加上系统在下一个状态时的期望折扣值, 大于系统其它任何动作的相关值, 也就是有下面的式子成立:

$$e^{\lambda R(x, \pi(x))} \sum_{y \in S} P(x, \pi(x))(y)\, U_\lambda(\gamma V^*_p(y)) \geqslant$$
$$e^{\lambda R(x, a)} \sum_{y \in S} P(x, a)(y)\, U_\lambda(\gamma V^*_p(y)). \tag{41}$$

定理 3.1　设 F^* 为最优静态策略的集合, 如果对于任意有限的 Markov 决策过程, 存在常数 $p: 1 \leqslant p < \infty$, 使得对所有的 $q > p$, 那么必定有 F_q 包含于 F^*. 也就是说静态贪婪策略是最优静态策略. 其中 F_q 为关于 $V^*_q(x)$ 的静态贪婪策略的集合.

证明　可直接由推论 2.2 和性质 3.1 得出此结论.

4　结论与未来的工作

在离散问题寻求最优控制策略的过程中, 基于 DP 的算法使用的是 max 算子. max 算子只考虑了在某一状态下, 执行最佳动作的结果, 而忽略了所有可能带来灾难的其他动作. 本文介绍了一族递归逼近算法, 这组算法通过把基于 DP 算法中的 max 算子, 替换为 p 阶广义平均(即一个非线性加权 l_p 范数). 这样的 DP 算法收敛于某一个解, 这样的解比通常的 DP 解更具有鲁棒性. 我们证明了, 对于每个指数 $p \geqslant 1$, 相应的递归算法收敛于一个唯一的不动点. 而且当 p 递增时, 能更好的一致逼近, 并且在极限 $p \to \infty$ 意义下, 它收敛于 DP 解. 同时我们还讨论了相关算法策略的最优性问题. 由于本文的问题结合了广义平均与风险敏感度的感念, 使得 MDP 中的一些基本的结论在这里并不成立. 特别是把风险敏感度广义平均作为性能评价指标, 是否满足基本的 Bellman 最优方程, 我们没有进一步讨论. 我们猜想当风险敏感度系数满足某种特定的条件时, 这种性能指标将符合 Bellman 最优方程. 另外, 如果把问题扩展到连续的状态空间甚至连续的动作空间, 将会有什么样的结论, 这也是值得讨论的课题. 另外, 我们没有对算法给出具体的实现代码, 也没有给出仿真实验. 我们希望将在未来的应用研究中来解决这些问题.

[参　考　文　献]

[1] Sutton R S. Learning to predict by the method of temporal difference[J]. Machine Learning, 1988, **3**(1): 9-44.

[2] Sutton R S. Open the oretical questions in reinforcement learning[A]. In: Proc of Euro COLT' 99 (Computational Learning Theory)[C]. Cambridge, MA: MIT Press, 1999, 11-17.

[3] Sutton R S, Barto A G. Reinforcement Learning: An Introduction[M]. Massachusetts: MIT Press, 1998, 20-300.

[4] Watkins C J C H, Dayan P. Q-learning[J]. Machine Learning, 1992, **8**(13): 279-292.

[5] Watkins C J C H. Learning from delayed rewards[D]. England: University of Cambridge, 1989.

[6] Bertsekas D P, Tsitsiklis J N. Parallel and Distributed Computation: Numerical Methods[M]. Englewood Cliffs, New Jersey: Prentice-Hall, 1989, 10-109.

[7] YIN Chang-ming, CHEN Huan-wen, XIE Li-juan. A Relative Value Iteration Q-learning Algorithm and its Convergence Based-on Finite Samples[J]. Journal of Computer Research and Development, 2002, **39**(9): 1064-1070.

[8] YIN Chang-ming, CHEN Huan-wen, XIE Li-juan. Optimality cost relative value iteration Q-learning algorithm based on finite samples[J]. Journal of Computer Engineering and Applications, 2002, **38**(11): 65-67.

[9] Wiering M, Schmidhuber J. Speeding up Q-learning[A]. In: Proc of the 10th European Conf on Machine Learning[C]. Germany: Springer-Verlag, 1998, 352-363.

[10] Singh S. Soft dynamic programming algorithms: convergence proofs[A]. In: Proceedings of Workshop on Computational Learning and Natural Learning (CLNL)[C]. Massachusetts: Town of Provinceton. University of Massachuetts, 1993.

[11] Cavazos-Cadena R, Montes-de-Oca R. The value iteration algorithm in risk-sensitive average Markov decision chains with finite state[J]. Mathematics of Operations Research, 2003, **28**(4): 752-776.

[12] Peng J, Williams R. Incremental multi-step Q-learning[J]. Machine Learning, 1996, **22**(4): 283-290.

[13] Singh S. Reinforcement learning algorithm for average-payoff Markovian decision processes[A]. Procedins of the 12th National Conference on Artificial Intelligence[C]. Taho city: Ca Morgan Kaufmann, 1994, **1**: 700-705.

Risk-Sensitive Reinforcement Learning Algorithms With Generalized Average Criterion

YIN Chang-ming[1,2], WHANG Han-xing[2], ZHAO Fei[2]

(1. College of Computer and Communicational Engineering, Changsha University of Science and Technology, Changsha 410076, P. R. China;
2. College of Sciences, Shanghai University, Shanghai 200444, P. R. China)

Abstract: A new algorithm which immolates optimality of control policies potentially to obtain the robusticity of solutions is proposed. The robusticity of solutions may become a very important property for a learning system due to when there exists non-matching between theory models and practical physical system, or the practical system is not static, or availability of a control action will change along with variety of time. The main contribution is that a set of approximation algorithms and its convergence results will be given. Applying generalized average operator instead of the general optimal operator max (or min) a class of important learning algorithm, dynamic programming algorithm were studied, and their convergence from theoretic point of view was discussed. The purpose is to improve robusticity of reinforcement learning algorithms theoretically.

Key words: reinforcement learning; risk-sensitive; generalized average; algorithm; convergence