文章编号: 1000−8152(2012)01−0104−05

带ε误差限的近似最优控制

林小峰, 黄元君, 宋春宁

(广西大学 电气工程学院, 广西 南宁 530004)

摘要: 近似动态规划方法求解非线性系统最优控制, 需要迭代无限步才能得到最优控制律. 本文提出了一种ε–近似最优控制算法, 选择ε误差限, 通过自适应迭代不断逼近哈密顿−雅可比−贝尔曼(HJB)方程的解, 应用神经网络实现在有限步迭代后得到带ε误差限的近似最优控制律. 计算机仿真结果表明了该算法的有效性.

关键词: ε误差限; 非线性系统; 近似动态规划; 最优控制

中图分类号: TP183 **文献标识码:** A

Approximate optimal control with ε-error bound

LIN Xiao-feng, HUANG Yuan-jun, SONG Chun-ning

(School of Electrical Engineering, Guangxi University, Nanning Guangxi 530004, China)

Abstract: In applying the approximate dynamic programming algorithm to determine the optimal control law for nonlinear systems, we need an infinite number of iterations to obtain the result. To deal with this problem, we propose an algorithm for obtaining an approximate optimal control law. Based on the given value of the ε-error bound, the approximation optimal control law will approach the solution of Hamilton-Jacobi-Bellman(HJB) equation through self-adaptive iteration. By applying neural networks, we can obtain the ε-approximation control law in a finite number of iterations. Simulation validates the efficacy of the above algorithm.

Key words: ε-error bound; nonlinear systems; approximate dynamic programming; optimal control

1 引言(Introduction)

近年来许多研究者用迭代方法来设计控制器[1−2], 然而用迭代方法来解决非线性系统的最优控制问题并不多. 非线性系统的分析常采用线性化处理, 但线性化必须满足苛刻的条件[3]. 尤其是非线性系统的最优控制问题, 难点在于如何求解非线性的哈密顿–雅可比–贝尔曼(Hamilton-Jacobi-Bellman equation, HJB)方程, 其解析求解一般不存在.

近似动态规划(approximate dynamic programming, ADP)通过自适应迭代不断逼近HJB方程得到近似解, 为解决非线性系统的最优控制问题提供了有效的方法[4−10]. 文献[5]中Landeliu用贪婪迭代ADP算法求解LQR(linear quadratic regulator)问题, 并证明了它的收敛, 结果表明这个迭代事实上等价于对Riccati代数方程的求解. 文献[6]提出用贪婪迭代ADP算法求解离散时间HJB方程, 证明了经过无限步迭代后能收敛得到最优控制律. 针对实际控制器的设计, 迭代通常是有限的, 文献[7]采用了ε误差限的迭代算法. 但其研究的是有限时间最优控制问题.

本文用带ε误差限迭代近似方法研究无限时间最优控制问题的有限步迭代. 首先, 在迭代ADP算法中引入ε误差限, 用$K_\varepsilon(x(k))$来截止无限步的迭代, 形成ε-迭代ADP算法; 其次, 通过仿真分析了不同ε下的控制性能; 最后, 对比分析仿真最优控制效果, 表明采用ε-迭代ADP算法得到近似控制律的有效性.

2 问题描述(Problem formulation)

2.1 问题背景(Problem background)

本文研究如下一类离散时间非线性系统:

$$x(k+1)=F(x(k),u(k)),\ k=0,1,2,\cdots, \quad (1)$$

其中: $x(k)\in\mathbb{R}^n$, $u(k)\in\Omega$, 对$\forall x(k)$,$u(k)$, 假设系统$F(x(k),u(k))$在包含原点的集Ω上利普希茨连续(Lipschitz)可控, 且有$F(0,0)$. 其中$x=0$为式(1)在$u=0$作用下的一个平衡点. 无限时间离散非线性系统最优状态反馈控制问题为找到一个最优控制律$u(k)$, 使性能指标函数式(2)最小化.

$$J(x(k),u(\cdot))=\sum_{i=k}^{\infty}U(x(i),u(i)), \quad (2)$$

其中: $U(x(i),u(i))$为效用函数, $u(x)$对应的输出控制序列$u(\cdot)=(u(k),u(k+1),\cdots),u(k)=u(x(k))$.

收稿日期: 2011−01−10; 收修改稿日期: 2011−04−18.

基金项目: 国家自然科学基金资助项目(60964002); 广西理工科学实验中心重点资助项目(LGZX201106); 广西研究生教育创新计划资助项目(105931003009).

求解上述最优控制问题, 根据贝尔曼最优性原理可得HJB方程

$$J^*(x(k),u(\cdot))=\min_{u(k)}\{U(x(k),u(k))+J^*(F(x(k+1)))\}=\min_{u(k)}\{U(x(k),u(k))+J^*(F(x(k),u(k)))\}. \quad (3)$$

满足式(3)解的最优控制律$u^*(x)$:

$$u^*(x)=\arg\min_{u(k)}\{U(x(k),u(k))+J^*(x(k+1))\}. \quad (4)$$

若从式(3)中解出最优性能指标函数$J^*(x(k),u(\cdot))$, 就可解出最优控制律$u^*(x)$.

2.2 迭代ADP算法(Iterative ADP algorithm)

为了与动态规划中的符号有所区别, 将迭代算法中的性能指标函数写成$V(x(k))=J(x(k),u(\cdot))$, 也称之为代价函数, 对应的控制律写成$\upsilon(x)$.

贪婪迭代公式.

迭代算法从初始值$V_0(\cdot)=0$开始, 然后根据式(5)计算出对应的单步控制律$\upsilon_0(x)$:

$$\upsilon_0(x(k))=\arg\min_{u(k)}\{U(x(k),u(k))+V_0(x(k+1))\}, \quad (5)$$

其中$V_0(x(k+1))=0$再根据式(6)计算代价函数

$$V_1(x(k))=U(x(k),\upsilon_0(x(k))). \quad (6)$$

对于$i=1,2,\cdots$, 迭代算法在

$$\upsilon_i(x(k))=\arg\min_{u(k)}\{U(x(k),u(k))+V_i(x(k+1))\} \quad (7)$$

和

$$V_{i+1}(x(k))=\arg\min_{u(k)}\{U(x(k),u(k))+V_i(x(k+1))\} \quad (8)$$

之间迭代, 其中$x(k+1)=F(x(k),\upsilon(x(k)))$. 在式(7)的基础上式(8)可写成

$$V_{i+1}(x(k))=U(x(k),\upsilon_i(x(k)))+V_i(x(k+1)), \quad (9)$$

其中$x(k+1)=F(x(k),\upsilon_i(x(k)))$. 这个迭代算法的思想是在式(7)和式(9)之间反复迭代更新代价函数序列$\{V_i\}$和控制律序列$\{\upsilon_i\}$.

定理 1　如式(8)中定义的代价函数序列$\{V_i\}$, 若$V_0(\cdot)=0$, 得到$\{V_i\}$是一个非下降序列, 即$V_{i+1}(x(k))\geqslant V_i(x(k))$, 当$i\to\infty$时$V_i(x(k))$收敛于最优性能指标函数$J^*(x(k))$, 即, $\lim\limits_{i\to\infty}V_i(x(k))=J^*(x(k))$, 并满足控制律序列$\lim\limits_{i\to\infty}\upsilon_i(x(k))=u^*(x(k))$.

文献[6]证明了定理1, 表明当迭代步数$i\to\infty$时, 理论上能找到最优控制律$\upsilon_i(x)=u^*(x)$. 而实际迭代步i通常都是有限的, 并不能得到等式$J^*(x(k))=V_i(x(k))$, 故无法保证能够找到的最优控制律$\upsilon_i(x)=u^*(x)$.

3 ε–迭代ADP算法(ε-iterative ADP algorithm)

为解决上述问题, 在迭代ADP算法$V_i(x(k))$和$J^*(x(k))$之间引入ε, 由定理1有$\lim\limits_{i\to\infty}V_i(x)=J^*(x)$, 总能找到一个$i$使得$|J^*(x(k))-V_i(x(k))|\leqslant\varepsilon$成立. 因此$\{i:|J^*(x(k))-V_i(x(k))|\leqslant\varepsilon\}\neq\phi$, 于是可给出迭代步数$K_\varepsilon(x(k))$的定义.

定义 1　令$x(k)\in\Gamma_\infty$, 其中: Γ_∞是容许可控状态集, $\varepsilon>0$为任意小的正数, 则$K_\varepsilon(x(k))$定义为

$$K_\varepsilon(x(k))=\min\{i:|J^*(x(k))-V_i(x(k))|\leqslant\varepsilon\}. \quad (10)$$

定义 2　令$x(k)\in\Gamma_\infty$, ε是一个正数, 定义

$$\Gamma_0^{(\varepsilon)}=\{0\},\ i=0, \quad (11)$$

$$\Gamma_i^{(\varepsilon)}=\{x(k)\in\Gamma_\infty:K_\varepsilon(x(k))\leqslant i\},\quad i=1,2,\cdots. \quad (12)$$

根据文献[7]可得:

1) $x(k)\in\Gamma_i^{(\varepsilon)}\Leftrightarrow$有且仅有 $V_i(x(k))\leqslant J^*(x(k))+\varepsilon$;
2) $\Gamma_i^{(\varepsilon)}\subseteq\Gamma_i$;
3) $\Gamma_i^{(\varepsilon)}\subseteq\Gamma_{i+1}^{(\varepsilon)}$;
4) $\bigcup\limits_i\Gamma_i^{(\varepsilon)}=\Gamma_\infty$;
5) 如果$\varepsilon>\delta>0$, 那么$\Gamma_i^{(\varepsilon)}\supseteq\Gamma_i^{(\delta)}$.

3.1 ε–最优控制(ε-optimal control)

若$x(k)\in\Gamma_\infty$为任意可控, 给定任意小的正数ε, 在控制律$\upsilon_i(x(k))$作用下代价函数满足$|J^*(x(k))-V_i(x(k))|\leqslant\varepsilon$, 称这个控制律为$\varepsilon$–近似最优控制律, 用$\mu_\varepsilon^*(x(k))$表示

$$\mu_\varepsilon^*(x(k))=\upsilon_i(x(k))=\arg\min_{u(k)}\{U(x(k),u(k))+V_{i-1}(F(x(k),u(x(k))))\}. \quad (13)$$

推论 1　根据文献[7], 式(13)得到的控制律$\mu_\varepsilon^*(x(k))$满足对任意$x'(k)\in\Gamma_i^{(\varepsilon)}$, 式(14)成立.

$$|J^*(x'(k))-V_i(x'(k))|\leqslant\varepsilon. \quad (14)$$

推论1表明, 无须通过搜寻子集$\Gamma_i^{(\varepsilon)}$内所有状态来获得最优控制律. 只要在$\Gamma_i^{(\varepsilon)}$内用一个状态点$x(k)\in\Gamma_i^{(\varepsilon)}$来获得$\varepsilon$–近似最优控制律$\mu_\varepsilon^*(x(k))$, 得到的这个控制律对任意状态$x'(k)\in\Gamma_i^{(\varepsilon)}$也是有效的. 这个特性减少了迭代ADP算法的计算量.

对于ε迭代ADP算法的最优性准则式$|J^*(x(k))-V_i(x(k))|\leqslant\varepsilon$, 性能指标函数的最优值通常是未知的. 可以用迭代最优准则$|V_i(x(k))-V_i(x(k))|\leqslant\varepsilon$进行替代.

3.2 ε–迭代算法流程(ε-iterative algorithm flowchart)

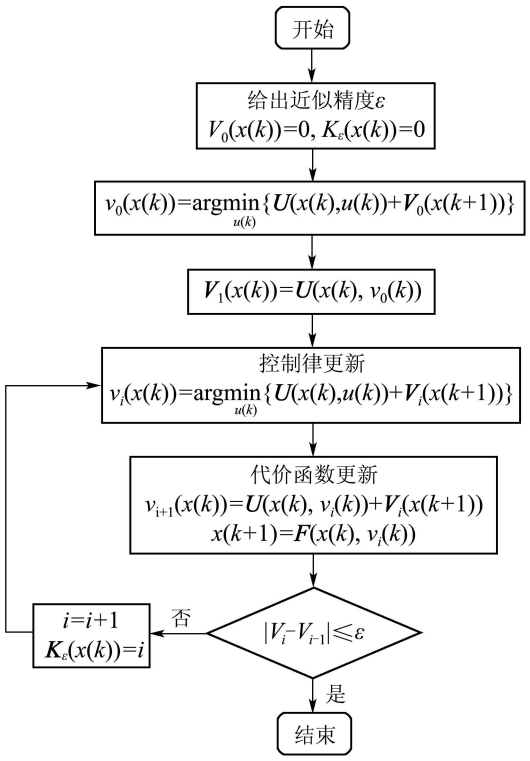

图 1 ε–迭代算法流程

Fig. 1 ε-iterative algorithm flowchart

3.3 ε–迭代ADP算法的神经网络实现(ε-iterative ADP algorithms using neural network)

用神经网络近似任意非线性函数的特性来实现迭代ADP算法[7−10]. 其框架如图2. 图2中的评价网络用来近似代价函数$V_i(x)$, 动作网络用来近似控制律$\upsilon_i(x)$. 动作网络和评价网络选2–8–1的3层BP神经网络结构. 图2中实线表示前向计算, e是神经网输出值与目标值之间的误差. 虚线表示根据误差调整神经网络的权值. 空心条形箭头表示下一时的刻状态进入评价网络. 其中效用函数(Utility)为

$$U(x(k),u(k))=x(k)^{\mathrm{T}}Qx(k)+u(k)^{\mathrm{T}}Ru(k).$$

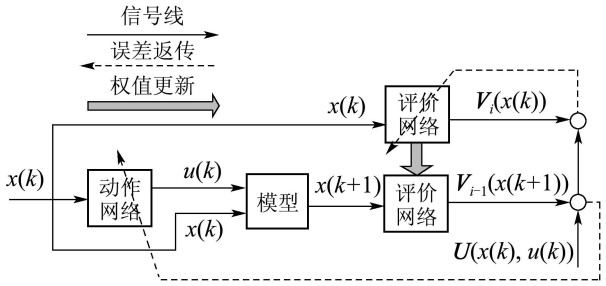

图 2 基于神经网络近似函数的迭代ADP算法结构图

Fig. 2 The structure diagram of the iterative ADP algorithm using neural-network approximate $V_i(x)$ and $\upsilon_i(x)$

4 仿真实验(Simulation)

考虑如下非线性离散时间系统:

$$x(k+1)=F(x(k),u(k)), \tag{15}$$

其中

$$F(x(k),u(k))=\begin{bmatrix}-0.2x_2(k)-0.5u(k)\\ \sin(u(k)-x_1(k))+0.5x_2(k)u(k))\end{bmatrix}.$$

无限时间二次型性能指标函数为

$$J(x(k),u(\cdot))=\sum_{i=k}^{\infty}x(i)^{\mathrm{T}}Qx(i)+u(i)^{\mathrm{T}}Ru(i), \tag{16}$$

其中Q, R为对应维数的单位矩阵.

4.1 不同ε的控制性能(Controller performance with different ε)

为分析不同ε下得到控制律的控制性能. 在配置为Windows XP操作系统, Pentium IV处理器的Lenovo个人电脑上运行MATLAB编写的程序. 选初始状态$x(k)=[0.5\ \ -0.5]^{\mathrm{T}}$, 取不同的$\varepsilon$, 执行$\varepsilon$–迭代ADP算法, 运行结果得到对应的迭代步数为$K_\varepsilon(x(k))$, 运行时间为$T$, 如下表1. 控制律$\mu_\varepsilon^*(x(k))$作用下的控制轨迹如图3, 状态轨迹如图4, 5.

表 1 取不同ε得到的各参数值

Table 1 The parameter values with different ε

ε	$K_\varepsilon(x(k))$	T/s	增时(ΔT)/s
0.1	21	1.203	–
0.01	23	1.508	0.305
0.0001	47	1.857	0.654
0.00001	51	2.215	1.012

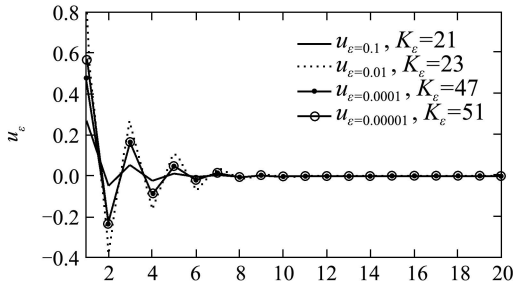

图 3 不同ε对应的控制律$\mu_\varepsilon^*(x(k))$作用的控制轨迹

Fig. 3 Control trajectory with different ε-control law$\mu_\varepsilon^*(x(k))$

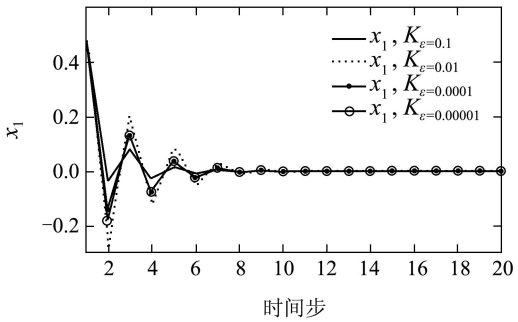

图 4 不同ε对应控制律$\mu_\varepsilon^*(x(k))$ 控制的状态轨迹x_1

Fig. 4 State x_1 trajectory with different ε-control law $\mu_\varepsilon^*(x(k))$

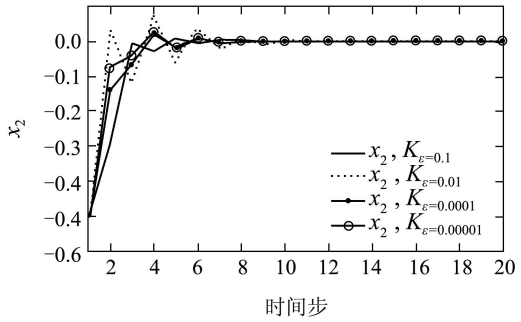

图 5 不同ε对应控制律$\mu_\varepsilon^*(x(k))$控制的状态轨迹x_2

Fig. 5 State x_2 trajectory with different ε-control law $\mu_\varepsilon^*(x(k))$

由表1可知, 随着ε减小, 迭代截止步数$K_\varepsilon(x(k))$与运行时间T都在不断增加. 从仿真结果控制轨迹(参照图3)与其对应的状态轨迹(参照图4, 5)可知, $\varepsilon = 0.1(K_\varepsilon(x(k)) = 21)$至$\varepsilon = 0.0001(K_\varepsilon(x(k)) = 47)$中控制性能逐渐改进, 因此带来的时间的增加是有意义的. 当精度$\varepsilon \leqslant 0.0001$时, $\varepsilon = 0.0001$与$\varepsilon = 0.00001$对应的控制轨迹接近重合(如图3示), 此时对性能的改善已不明显. 如果再减小ε, 只会增加较多的计算时间. 因此, 根据实际问题选择ε而截止迭代得到的控制律μ_ε^*替代无限步迭代的最优控制律$u^*(x)$, 可以提高算法的效率.

4.2 仿真对比(Comparison simulation)

为进一步说明该算法有效性, 取$\varepsilon = 0.0001$, 执行ε–迭代ADP算法, 当ε迭代满足$|J^*(x(k)) - V_i(x(k))| \leqslant \varepsilon$时, 得到$K_\varepsilon(x(k)) = 47$, 从而提前截止迭代, 相应的控制律为$\mu_\varepsilon^*(x(k))$. 图6显示了代价函数迭代收敛过程. 其结果符合定理1中$V_i(x(k))$递增且有界收敛的性质.

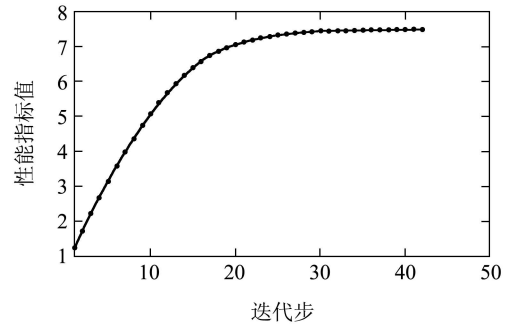

图 6 代价函数迭代收敛过程图

Fig. 6 The convergence process of cost function

取$K(x(k)) = 1000$作为近似无限迭代, 比较两者的控制轨迹(如图7)与状态轨迹(如图8, 9). 其中: u_ε, x_ε表示控制律μ_ε^*作用下的控制轨迹与状态轨迹; u, x表示$K(x(k)) = 1000$步迭代得到的最优控制律$u^*(x)$作用下的控制轨迹和状态轨迹. 结合图7−9分析可知, $\mu_\varepsilon^*(x(k))$作用下的控制轨迹及状态轨迹与无限迭代最优控制律$u^*(x)$作用下的控制轨迹及状态轨迹已非常接近.

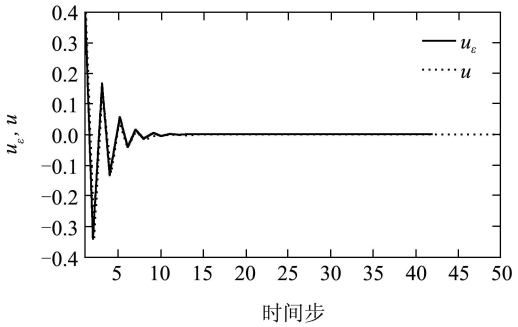

图 7 控制轨迹u_ε与u

Fig. 7 The control trajectory u_ε and u

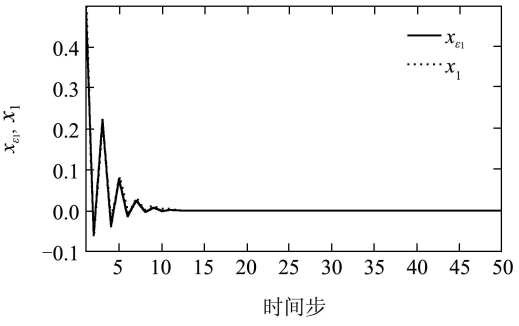

图 8 状态轨迹$x_{\varepsilon 1}$与x_1

Fig. 8 The state trajectories $x_{\varepsilon 1}$ and x_1

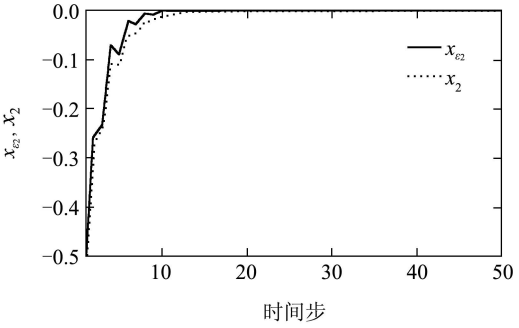

图 9 状态轨迹$x_{\varepsilon 2}$与x_2

Fig. 9 The state trajectories $x_{\varepsilon 2}$ and x_2

5 结论(Conclusion)

本文采用ε–近似最优控制方法, 使无限贪婪迭代ADP算法在有限步迭代内得到近似最优控制律. 并通过仿真实验分析了ε对控制性能的影响. ε-ADP迭代算法根据给定的ε误差限, 采用$K_\varepsilon(x(k))$截止迭代, 能够减少迭代步数, 提高算法效率; 得到的近似控制律μ_ε^*能有效的把状态控制到稳定点, 且使性能指标与最优性能指标保持在精度$|J^*(x(k)) - V_i(x(k))| \leqslant \varepsilon$内.

参考文献(References):

[1] 吴忠伟, 陈辉堂, 王月娟. 基于反馈控制的迭代学习控制器设计[J]. 控制理论与应用, 2001, 18(5): 785 – 791.
(WU Zhongwei, CHEN Huitang, WANG Yuejuan. Iterative learning controller design based on feedback[J]. *Control Theory & Applications*, 2001, 18 (5): 785 – 791.)

[2] 孙明轩, 何熊熊, 俞立. 迭代学习控制器设计: 一种有限时间死区方法[J]. 控制理论与应用, 2007, 24(3): 349 – 354.
(SUN Mingxuan, HE Xiongxiong, YU Li. Iterative learning controller designs: a finite time dead-zone approach[J]. *Control Theory & Applications*, 2007, 24(3): 349 – 354.)

[3] 胡跃明, 胡终须, 毛宗源, 等. 非线性控制系统的近似化方法[J]. 控制理论与应用, 2001, 18(2): 160 – 164.
(HU Yueming, HU Zhongxu, MAO Zongyuan, et al. Approximation methods of nonlinear control systems[J]. *Control Theory & Applications*, 2001, 18(2): 160 – 164.)

[4] MURRAY J J, COX C J, LENDARIS G G, et al. Adaptive dynamic programming[J]. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 2002, 32(2): 140 – 153.

[5] LANDELIUS T. *Reinforcement learning and distributed local model synthesis*[D]. Linkoping, Sweden: Dissertation, Linkoping University, 1997.

[6] AL-TAMIMI A, LEWIS F L, ABU-KHALAF M. Discrete-time nonlinear HJB solution using approximate dynamic programming: convergence proof[J]. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 2008, 38(4): 943 – 950.

[7] WANG F Y, JIN N, LIU D R, et al. Adaptive dynamic programming for finite-horizon optimal control of discrete-time system with epsilon-error bound[J]. *IEEE Transactions on Neural Networks*, 2011, 22(1): 24 – 36.

[8] WEI Q L, ZHANG H G, LIU D R, et al. An optmal control scheme for a class of discrete-time nonlinear systems with time delays using adaptive dynamic programming[J]. *Acta Automatica Sinica*, 2010, 36(1): 121 – 129.

[9] WERBOS P J. Approximate dynamic programming for real-time control and neural modeling[M] //WHITE D A, SOFGE D A. *In Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches*, New York: Van Nostrand Reinhold, 1992.

[10] ABU-KHALAF M, LEWIS F L. Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach[J]. *Automatica*, 2005, 41(5): 779 – 791.

作者简介:

林小峰 (1955—), 男, 教授, 研究方向为复杂系统优化控制, E-mail: gxulinxf@163.com;

黄元君 (1983—), 男, 硕士研究生, 研究方向为复杂系统优化控制, E-mail: huangyuanjungege@126.com;

宋春宁 (1969—), 男, 副教授, 研究方向为电子技术、嵌入式系统, E-mail: scn206@gxu.edu.cn.