文章编号: 1001-0920 (2015) 03-0495-05 DOI: 10.13195/j.kzyjc.2014.0102

基于评价网络近似误差的自适应动态规划优化控制

林小峰, 丁 强

(广西大学 电气工程学院，南宁 530004)

摘 要: 为了求解有限时域最优控制问题, 自适应动态规划 (ADP) 算法要求受控系统能一步控制到零. 针对不能一步控制到零的非线性系统, 提出一种改进的 ADP 算法, 其初始代价函数由任意的有限时间容许序列构造. 推导了算法的迭代过程并证明了算法的收敛性. 当考虑评价网络的近似误差并满足假设条件时, 迭代代价函数将收敛到最优代价函数的有界邻域. 仿真例子验证了所提出方法的有效性.

关键词: 自适应动态规划；优化控制；人工神经网络；近似误差

中图分类号: TP18 **文献标志码:** A

Adaptive dynamic programming optimal control based on approximation error of critic network

LIN Xiao-feng, *DING Qiang*

(School of Electrical Engineering，Guangxi University，Nanning 530004，China. Correspondent：DING Qiang, E-mail：819476292@qq.com)

Abstract: In order to solve finite horizon optimal control problems, the adaptive dynamic programming(ADP) algorithm demands the system can reach zero in one step of control. For the nonlinear systems which cannot be controlled to zero in one step, an improved ADP algorithm is presented, and the initial cost is constructed by arbitrary finite horizon admissible sequence. After giving the iterative process, the convergence analysis of the improved algorithm is conducted. If the approximation error of the critic network is considered and several assumptions are satisfied, the iterative cost function will converge to a finite neighborhood of the optimal cost function. A simulation example is provided to verify the effectiveness of the presented approach.

Keywords: adaptive dynamic programming；optimal control；artificial neural network；approximation error

0 引 言

动态规划是处理最优控制的有效方法, 但在实际求解非线性系统最优控制问题时, 它的反向搜索特点以及“维数灾”问题[1]极大地限制了其应用. 由 Werbos[2-4]提出的自适应动态规划 (ADP) 本质上基于强化学习原理, 将动态规划与人工神经网络有机结合在一起, 是解决复杂非线性系统最优控制的重要理论和方法. ADP 采用非线性函数拟合方法逼近动态规划的性能指标, 在求解非线性 Hamilton-Jacobi-Bellman (HJB) 方程[5-6]的同时避免了“维数灾”难题. 作为一种有效的智能控制方法, 近年来 ADP 及其相关研究受到了人们的关注, 取得了一些进展[7-10]. 文献 [7] 严格证明了迭代 ADP 算法的收敛性, 为迭代 ADP 算法求解离散系统最优控制问题提供了理论依据. 文献 [9] 提出了一种在线执行-评价算法, 为求解连续时间系统最优控制问题提供了新的思路. 文献 [10] 为了实现在线学习和优化, 提出了一种新的 ADP 结构, 与传统 ADP 结构不同, 该结构增加了一个参考网络用来自适应地建立内部强化信号.

为了求解离散非线性系统的有限时域最优控制问题, 文献 [11] 提出了一种迭代 ADP 算法, 目前, 它已成功地用于解决跟踪控制[12]、执行器饱和[13]、状态时滞[14]等问题. 文献 [12] 处理跟踪问题的策略是将其转化为最优控制问题进行求解. 文献 [13] 针对执行器饱和约束, 引入新的性能指标, 继而推导出带饱和约束的非线性 HJB 方程, 并采用迭代 ADP 算法求解.

收稿日期: 2014-01-17；**修回日期:** 2014-06-27.

基金项目: 国家自然科学基金重点项目(61034002)；国家自然科学基金项目(61364007).

作者简介: 林小峰(1955−), 男, 教授, 从事智能优化控制等研究；丁强(1988−), 男, 硕士生, 从事智能优化控制的研究.

文献[11]指出该算法要求初始状态必须能一步控制到平衡态, 因此它适用于能一步控制到零的非线性系统. 然而对于非仿射的非线性系统, 该算法条件往往不能成立, 这极大地限制了它的应用.

本文在上述研究的基础上提出一种改进迭代ADP算法. 该改进算法与原算法的不同之处主要有两点: 第一, 改进算法对于任意有限时间可控初态都适用, 而原算法要求对初态存在一步容许控制序列; 第二, 改进算法的初始化代价函数由任意的有限时间容许控制序列构造, 而原算法初始化代价为零. 由于人工神经网络具有自适应、非线性、较强的输入输出映射等特点, ADP算法的实现大多采用人工神经网络. 目前, 关于ADP算法的研究并未考虑人工神经网络的近似误差, 但这种误差是不可避免的, 因此, 考虑人工神经网络近似误差对ADP算法收敛性的影响具有现实意义. 本文将在执行网络精确逼近的前提下, 对评价网络近似误差进行分析.

1 问题描述

本文研究的离散时间非线性系统为

$$x_{k+1}=F(x_k,u_k),\ k=0,1,2,\cdots. \tag{1}$$

其中: $x_k\in R^n$ 表示状态, $u_k\in R^m$ 表示控制向量. 令初始状态是 x_0, 系统函数 $F(x_k,u_k)$ 对于 $\forall x_k,u_k$ 都连续且满足 $F(0,0)=0$. 因此, $x=0$ 是系统(1)在控制 $u=0$ 下的平衡态. 关于状态 x_0 和有限控制序列 $\underline{u}_0^{N-1}=(u_0,u_1,\cdots,u_{N-1})$ 的代价函数定义为

$$J(x_0,\underline{u}_0^{N-1})=\sum_{i=0}^{N-1}\gamma^iU(x_i,u_i). \tag{2}$$

其中: $U(x,u)\geqslant 0$ 是效用函数, 通常可取为二次型; γ 是折扣因子, $0<\gamma\leqslant 1$.

有限控制序列 $\underline{u}_0^{N-1}$ 的长度 $|\underline{u}_0^{N-1}|$ 定义为序列中控制量的个数, 即 $|\underline{u}_0^{N-1}|=N$. 系统(1)在 $\underline{u}_0^{N-1}$ 作用下从 x_0 开始的轨迹记为 $\underline{x}_0^N=(x_0,x_1,\cdots,x_N)$, 将轨迹的终态记为 $x^{(F)}(x_0,\underline{u}_0^{N-1})$, 即 $x^{(F)}(x_0,\underline{u}_0^{N-1})=x_N$. 相应地, 从 k 时刻开始的长度为 i 的控制序列记为 $\underline{u}_k^{k+i-1}=(u_k,u_{k+1},\cdots,u_{k+i-1})$, 对应的终态是 $x^{(F)}(x_k,\underline{u}_k^{k+i-1})=x_{k+i}$.

定义控制序列 $\underline{u}_k^{k+i-1}$ 关于状态 $x_k\in R^n$ 是有限时间容许的, 如果 $x^{(F)}(x_k,\underline{u}_k^{k+i-1})=0$ 且 $J(x_k,\underline{u}_k^{k+i-1})$ 取有限值. 将存在有限时间容许序列的 x_k 定义为有限时间可控状态. 令

$$\boldsymbol{A}_{x_k}^{(i)}=\{u_k^{k+i-1}:x^{(F)}(x_k,\underline{u}_k^{k+i-1})=0,|\underline{u}_k^{k+i-1}|=i\}$$

是所有长度为 i 的关于 x_k 的有限时间容许控制序列集合, $\boldsymbol{A}_{x_k}=\{u_k:x^{(F)}(x_k,u_k)=0\}$是所有关于 x_k 的有限时间容许控制序列集合, 显然 $\boldsymbol{A}_{x_k}=\bigcup\limits_{1\leqslant i<\infty}\boldsymbol{A}_{x_k}^{(i)}$.

定义最优代价函数为

$$J^*(x_k)=\inf_{u_k}\{J(x_k,u_k):u_k\in\boldsymbol{A}_{x_k}\}. \tag{3}$$

根据贝尔曼最优性原理, $J^*(x_k)$ 满足离散时间HJB (DTHJB) 方程, 即

$$J^*(x_k)=\min_{u_k}\{U(x_k,u_k)+\gamma J^*(F(x_k,u_k))\}, \tag{4}$$

相应的最优控制律为

$$u^*(x_k)=\arg\min_{u_k}\{U(x_k,u_k)+\gamma J^*(F(x_k,u_k))\}. \tag{5}$$

当需要处理一般的非线性最优控制问题时, 上述的最优代价 $J^*(x_k)$ 以及最优控制律 $u^*(x_k)$ 通常是难以精确求解的. 本文将采用改进迭代ADP算法来处理DTHJB方程.

2 改进迭代ADP算法

设 x_k 为有限时间可控状态, 令 $\underline{v}_k^{N-1}=(v_k,v_{k+1},\cdots,v_{N-1})$ 为关于 x_k 的任意的有限时间容许序列, N 是终点时刻. 定义初始代价函数 $V_0(x_k)=\phi(x_k)=J(x_k,\underline{v}_k^{N-1})$, 当 $i=1,2,\cdots$ 时, 有

$$v_i(x_k)=\arg\min_{u_k}\{U(x_k,u_k)+\gamma V_{i-1}(F(x_k,u_k))\}, \tag{6}$$

$$V_i(x_k)=\min_{u_k}\{U(x_k,u_k)+\gamma V_{i-1}(F(x_k,u_k))\}=U(x_k,v_i(x_k))+\gamma V_{i-1}(F(x_k,v_i(x_k))). \tag{7}$$

其中: i 是迭代步, k 是时间步. 式(6)和(7)构成了改进迭代ADP算法.

通过上述的循环迭代过程, 代价函数和控制律被不断更新. 在下一节中, 将证明迭代代价 $V_i\to J^*$ 以及迭代控制律 $v_i\to u^*$.

3 改进迭代ADP算法性质

引理1 迭代代价函数序列 $V_i(x_k)$ 满足

$$V_i(x_k)=\min_{\underline{u}_k^{k+i-1}}\Big\{\sum_{j=0}^{i-1}\gamma^jU(x_{k+j},u_{k+j})\Big\}+\gamma^i\phi(x_{k+i})=\sum_{j=0}^{i-1}\gamma^jU(x_{k+j},v_{i-j}(x_{k+j}))+\gamma^i\phi(x_{k+i}).$$

证明 由 $V_i(\cdot)$ 的定义可知

$$V_i(x_k)=\min_{u_k}\{U(x_k,u_k)+\gamma V_{i-1}(x_{k+1})\}=\min_{u_k}\{U(x_k,u_k)+\min_{u_{k+1}}\{\gamma U(x_{k+1},u_{k+1})+\cdots+\min_{u_{k+i-1}}\{\gamma^{i-1}U(x_{k+i-1},u_{k+i-1})+\gamma^i\phi(x_{k+i})\}\cdots\}\}.$$

由最优性准则可得

$$V_{i+1}(x_k)=\min_{\underline{u}_k^{k+i-1}}\{U(x_k,u_k)+\gamma U(x_{k+1},u_{k+1})+\cdots+\gamma^{i-1}U(x_{k+i-1},u_{k+i-1})+\gamma^i\phi(x_{k+i})\}=\min_{\underline{u}_k^{k+i-1}}\left\{\sum_{j=0}^{i-1}\gamma^jU(x_{k+j},u_{k+j})\right\}+\gamma^i\phi(x_{k+i}).\tag{8}$$

另外, 按照式 (7), $V_i(x_k)$ 可以等价地记为

$$V_i(x_k)=\sum_{j=0}^{i-1}\gamma^jU(x_{k+j},v_{i-j}(x_{k+j}))+\gamma^i\phi(x_{k+i}),\tag{9}$$

由此引理得证. □

引理 2　$V_i(x_k)$ 为算法 (6) 和 (7) 定义的迭代代价. 设 $\mu_k=(\mu_k,\mu_{k+1},\cdots)$ 为任意的有限时间容许控制序列, 并定义另一代价 $\{G_i\}$ 为 $G_0(x_k)=\phi(x_k)$, $G_i(x_k)=U(x_k,\mu_k)+\gamma G_{i-1}(x_{k+1})$, 则 $V_i(x_k)\leqslant G_i(x_k)$.

证明　比较 V_i 和 G_i 的定义, V_i 是式 (7) 右边关于控制 u_k 求最小值的结果, 而 G_i 中的控制是任选的, 显然结论成立. □

定理 1　设 x_k 是有限时间可控状态, 则由迭代 ADP 算法 (6) 和 (7) 产生的代价函数序列 $V_i(x_k)$ 是单调非增的, 即 $V_{i+1}(x_k)\leqslant V_i(x_k)$, $i\geqslant 0$.

证明　采用数学归纳法证明. 当 $i=0$ 时, 有

$$V_0(x_k)=U(x_k,v_k)+\gamma\phi(x_{k+1}),$$
$$G_1(x_k)=U(x_k,\mu_k)+\gamma\phi(x_{k+1}).$$

因为 μ_k 是任意选取的, 若令 $\mu_k=v_k$, 则有 $V_0(x_k)=G_1(x_k)$. 又由引理 2 可得 $G_1(x_k)\geqslant V_1(x_k)$, 这说明 $V_0(x_k)\geqslant V_1(x_k)$.

假设定理对于 $i=p-1,p\geqslant 1$ 成立, 则由引理 1 可知

$$V_p(x_k)=\sum_{j=0}^{p-1}\gamma^jU(x_{k+j},v_{p-j}(x_{k+j}))+\gamma^p\phi(x_{k+p}),$$

其中 $\phi(x_{k+p})=J(x_{k+p},\underline{v}_{k+p}^{N-1})$. 若令

$$\mu_k=(v_p(x_k),v_{p-1}(x_{k+1}),\cdots,v_1(x_{k+p-1}),\underline{v}_{k+p}^{N-1}),$$

则有

$$G_{p+1}(x_k)=U(x_k,v_p(x_k))+\gamma U(x_{k+1},v_{p-1}(x_{k+1}))+\cdots+\gamma^{p-1}U(x_{k+p-1},v_1(x_{k+p-1}))+\gamma^pU(x_{k+p},v_{k+p})+\gamma^{p+1}G_0(x_{k+p+1})=V_p(x_k).$$

又由引理 2 知 $G_{p+1}(x_k)\geqslant V_{p+1}(x_k)$, 所以定理对于 $i=p$ 成立. □

由定理 1 可知, 代价序列 $V_i(x_k)$ 是单调非增有下界的, 这说明其极限是存在的.

定理 2　设 x_k 是有限时间可控状态, $V_i(x_k)$ 的定义见算法 (6) 和 (7), 记 $V_i(x_k)$ 的极限为 $V_\infty(x_k)=\lim\limits_{i\to\infty}V_i(x_k)$, 则有 $V_\infty(x_k)=J^*(x_k)$.

证明　由 $J^*(x_k)$ 的定义可知 $J^*(x_k)\leqslant V_i(x_k)$, 令 $i\to\infty$, 有

$$J^*(x_k)\leqslant V_\infty(x_k).\tag{10}$$

另一方面, 对于任取的小正数 δ, 根据下确界的定义, 存在某个 $\eta_q\in\boldsymbol{A}_{x_k}$ 使得 $G_q\leqslant J^*(x_k)+\delta$. 由引理 2 和定理 1 可知

$$V_\infty(x_k)\leqslant V_q(x_k)\leqslant G_q(x_k),$$

因此

$$V_\infty(x_k)\leqslant V_l(x_k)\leqslant J^*(x_k)+\delta.$$

因为 δ 是任取的, 所以

$$V_\infty(x_k)\leqslant J^*(x_k).\tag{11}$$

根据式 (10) 和 (11), 即可得到

$$\lim_{i\to\infty}V_i(x_k)=J^*(x_k).$$ □

推论 1　迭代控制律将收敛于最优控制律, 即

$$u^*(x_k)=\lim_{i\to\infty}v_i(x_k).$$

4　基于评价网络近似误差的收敛性分析

为了执行 ADP 算法, 必须使用函数近似结构来得到 $V_i(x_k)$ 和 $v_i(x_k)$, 通常人们引入神经网络来逼近 $V_i(x_k)$ 和 $v_i(x_k)$ 时并没有考虑神经网络的近似误差. 本文在文献 [15-16] 的误差限分析方法基础上, 考虑评价网络近似误差对算法收敛性的影响.

取小正数 $\varepsilon<1$, 令 $\underline{\varepsilon}=1-\varepsilon$, $\bar{\varepsilon}=1+\varepsilon$. 初始代价近似值 $\tilde{V}_0(x_k)$ 满足

$$\underline{\varepsilon}V_0(x_k)\leqslant\tilde{V}_0(x_k)\leqslant\bar{\varepsilon}V_0(x_k).$$

当 $i\geqslant 1$ 时, 记 $V_i(x_k)$ 和 $v_i(x_k)$ 的神经网络近似值分别为 $\tilde{V}_i(x_k)$ 和 $\tilde{v}_i(x_k)$. 当 $i=1,2,\cdots$ 时, 算法迭代更新如下:

$$\tilde{v}_i(x_k)=\arg\min_{u_k}\{U(x_k,u_k)+\gamma\tilde{V}_{i-1}(x_{k+1})\},\tag{12}$$

$$\min_{u_k}\left\{\underline{\varepsilon}U(x_k,u_k)+\gamma\tilde{V}_{i-1}(x_{k+1})\right\}\leqslant\tilde{V}_i(x_k)\leqslant\min_{u_k}\left\{\bar{\varepsilon}U(x_k,u_k)+\gamma\tilde{V}_{i-1}(x_{k+1})\right\}.\tag{13}$$

记 $\tilde{v}_i^{\varepsilon}(x_k)=\arg\min\limits_{u_k}\{\underline{\varepsilon}U(x_k,u_k)+\gamma\tilde{V}_{i-1}(x_{k+1})\}$.

注 1　在上面的迭代式中, 评价网络的近似值存在误差, 并且每一步的误差都是有界的. 因为神经网络强大的非线性逼近能力, 能够使得每一步的逼近误

差都足够小, 所以该假设是合理的. 还应该注意的是, 本文假设执行网络是精确的, 执行网络近似值与理论值之间存在的误差是由评价网络误差造成的. 下面将证明存在误差的迭代序列不仅收敛, 而且能收敛到最优值有界邻域.

定理 3 V_i 和 v_i 的定义见式 (6) 和 (7), 其神经网络近似值 $\tilde{V}_i$ 和 $\tilde{v}_i$ 按照式 (12) 和 (13) 更新, 若初始近似值满足 $\underline{\varepsilon}V_0(x_k) \leqslant \tilde{V}_0(x_k) \leqslant \bar{\varepsilon}V_0(x_k)$, 则不等式 $\underline{\varepsilon}V_i(x_k) \leqslant \tilde{V}_i(x_k) \leqslant \bar{\varepsilon}V_i(x_k)$ 对于任意 x_k 和 i 都成立. 在此基础上, 有 $\underline{\varepsilon}J^*(x_k) \leqslant \tilde{V}_\infty(x_k) \leqslant \bar{\varepsilon}J^*(x_k)$.

证明 首先根据数学归纳法证明不等式左边 $\varepsilon V_i(x_k) \leqslant \tilde{V}_i(x_k)$. 当 $i=0$ 时结论显然成立. 假设 $i=p-1, p\geqslant 1$ 时结论成立, 即 $\underline{\varepsilon}V_{p-1}(x_k) \leqslant \tilde{V}_{p-1}(x_k)$. 由式 (13) 可知

$$\begin{aligned}&\tilde{V}_p(x_k) \geqslant\\ &\underline{\varepsilon}U(x_k, \tilde{v}_p^{\underline{\varepsilon}}(x_k)) + \gamma\tilde{V}_{p-1}(x_{k+1}) \geqslant\\ &\underline{\varepsilon}\{U(x_k, \tilde{v}_p^{\underline{\varepsilon}}(x_k)) + \gamma V_{p-1}(x_{k+1})\}.\end{aligned}$$

考虑到

$$\begin{aligned}&V_p(x_k) =\\ &\min_{u_k}\{U(x_k, u_k) + \gamma V_{p-1}(x_{k+1})\} \leqslant\\ &U(x_k, \tilde{v}_p^{\underline{\varepsilon}}(x_k)) + \gamma V_{p-1}(x_{k+1}),\end{aligned}$$

所以 $\underline{\varepsilon}V_p(x_k) \leqslant \tilde{V}_p(x_k)$, 结论对于 $i=p$ 成立. 不等式左边得证.

同样使用数学归纳法证明不等式右边. 当 $i=0$ 时结论显然成立. 若 $i=q-1, q\geqslant 1$ 时结论成立, 即 $\tilde{V}_{q-1}(x_k) \leqslant \bar{\varepsilon}V_{q-1}(x_k)$. 由式 (13) 可知

$$\begin{aligned}&\bar{\varepsilon}V_q(x_k) =\\ &\bar{\varepsilon}\{U(x_k, v_q(x_k)) + \gamma V_{q-1}(x_{k+1})\} \geqslant\\ &\bar{\varepsilon}U(x_k, v_q(x_k)) + \gamma\tilde{V}_{q-1}(x_{k+1}) \geqslant\\ &\min_{u_k}\{\bar{\varepsilon}U(x_k, u_k) + \gamma\tilde{V}_{q-1}(x_{k+1})\} \geqslant\\ &\tilde{V}_q(x_k).\end{aligned}$$

结论对于 $i=q$ 成立, 故不等式右边得证. 令 $i\to\infty$, 由定理 2 即可得到 $\underline{\varepsilon}J^*(x_k) \leqslant \tilde{V}_\infty(x_k) \leqslant \bar{\varepsilon}J^*(x_k)$. □

注 2 定理 3 表明, 在满足一定条件下由评价网络能够得到近似最优代价函数, 使用神经网络实现改进迭代 ADP 算法是可行的.

5 仿真研究

考虑如下离散时间仿射非线性系统:

$$x_{k+1} = F(x_k, u_k) = x_k + \sin(0.1x_k^2 + 0.5u_k),$$

其中 $x_k, u_k \in R$.

代价函数的定义为

$$J(x_0, \underline{u}_0^{N-1}) = \sum_{i=0}^{N-1}\gamma^i(x_i^{\mathrm{T}}Qx_i + u_i^{\mathrm{T}}Ru_i),$$

此处取 $Q=R=1, \gamma=0.9$. 令初始状态为 $x_0=-1.2$.

注意到不存在实数 u_k 使 $-1.2+\sin(0.1\times 1.2^2 + 0.5u_k)=0$, 因此不能使用已有的 ADP 算法求解该系统. 使用改进算法对其进行求解. 取初始有限时间容许序列

$$\underline{v}_0^1 = [2\arcsin(0.4) - 0.288, 2\arcsin(0.8) - 0.128],$$

$V_0(x_0) = 4.985\,3$. 选用 3 层的 BP 网络作为执行网络和评价网络, 其结构均为 1-8-1. 给定小正数 $\varepsilon = 0.000\,01$ 作为算法的截止条件, 为了达到该精度, 对执行网络和评价网络训练 100 个迭代步, 每一个迭代步包括 1 000 个训练步, 评价、执行网络的学习率分别为 0.5、0.2, 迭代代价函数的收敛过程见图 1. 由图 1 可知, 迭代代价函数只需 5 个迭代步就能收敛, 这与较大的训练步取值有关. 将优化控制律应用到系统 10 个时间步, 得到的优化状态轨迹见图 2, 相应的优化控制轨迹见图 3. 由图 2 和图 3 可知, 对于初态 −1.2, 根据改进算法求出的近似最优控制序列长度为 3.

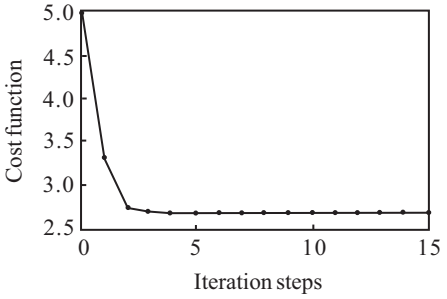

图 1 迭代代价函数收敛过程

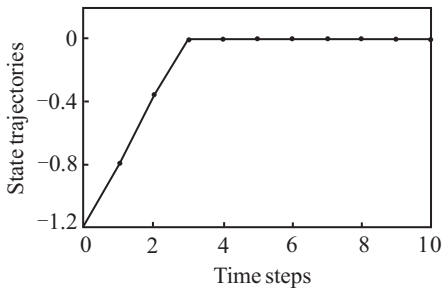

图 2 优化状态轨迹

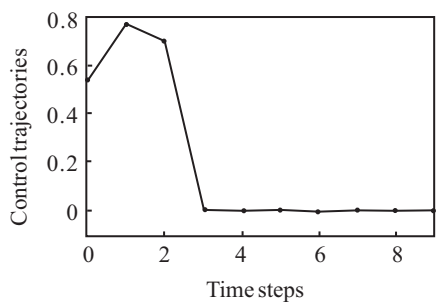

图 3 相应的优化控制轨迹

6 结　论

本文针对初始状态不能一步控制到平衡态的非线性系统, 提出了一种改进迭代ADP算法. 在推导出算法迭代过程之后, 严格证明了算法的收敛性, 即迭代代价和迭代控制律分别收敛到相应的最优值. 在实际设计控制器时, 即使考虑评价网络的近似误差, 迭代代价序列也能收敛到最优代价函数的有界邻域. 仿真结果表明所提出的算法是有效的.

参考文献(References)

[1] Lewis F L, Syrmos V L. Optimal control[M]. New York: Wiley, 1995: 71-79.

[2] Werbos P J. Handbook of intelligent control: Neural, fuzzy, and adaptive approaches[M]. New York: Van Nostrand Reinhold, 1992: 23-38.

[3] Werbos P J. ADP: The key direction for future research in intelligent control and understanding brain intelligent[J]. IEEE Trans on Systems, Man, and Cybernetics, Part B: Cybernetics, 2008, 38(4): 898-900.

[4] Werbos P J. Intelligent in the Brain: A theory of how it works and how to build it[J]. Neural Networks, 2009, 22(3): 200-212.

[5] Murad Abu-Khalaf, 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.

[6] Wang Fei-yue, Zhang Hua-guang, Liu De-rong. Adaptive dynamic programming: An introduction[J]. IEEE Computational Intelligence Magazine, 2009, 4(2): 39-47.

[7] Asma Al-Tamimi, Lewis F L, Murad Abu-Khalaf. Discrete-time nonlinear HJB solution using approximate dynamic programming: Convergence proof[J]. IEEE Trans on Systems, Man, and Cybernetics, Part B: Cybernetics, 2008, 38(4): 943-949.

[8] Dierks T, Thumati B T, Sarangapani J. Optimal control of unknown affine nonlinear discrete-time systems using offline-trained neural networks with proof of convergence[J]. Neural Networks, 2009, 22(5/6): 851-860.

[9] Vamvoudakis K G, Lewis F L. Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem[J]. Automatica, 2010, 46(5): 878-888.

[10] He Haibo, Ni Zhen, Fu Jian. A three-network architecture for on-line learning and optimization based on adaptive dynamic programming[J]. Neurocomputing, 2012, 78(1): 3-13.

[11] Wang Fei-yue, Jin Ning, Liu De-rong, ea al. Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε-error bound[J]. IEEE Trans on Neural Networks, 2011, 22(1): 24-36.

[12] Song Ruizhuo, Zhang Huaguang. N-step optimal time-invariant trajectory tracking control for a class of nonlinear systems[C]. IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning. Paris: IEEE, 2011: 184-189.

[13] Xiaofeng Lin, Yuanjun Huang, Nuyun Cao, et al. Optimal control scheme for nonlinear systems with saturating actuator using ε — Iterative adaptive dynamic programming[C]. UKACC Int Conf on Control. Cardiff: IEEE, 2012: 3-5.

[14] Xiaofeng Lin, Nuyun Cao, Yuzhang Lin. Optimal control for a class of nonlinear systems with state delay based on adaptive dynamic programming with ε — Error bound[C]. IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning. Singapore: IEEE, 2013: 170-175.

[15] Rantzer A. Relaxed dynamic programming in switching systems[J]. IEE Proc of Control Theory and Application, 2006, 153(5): 567-574.

[16] Li H, Liu D. Optimal control for discrete-time affine non-linear systems using general value iteration[J]. IET Control Theory and Applications, 2012, 6(18): 2725-2736.

(责任编辑: 李君玲)