DOI: 10.7641/CTA.2016.41195

未知饱和控制系统有穷域最优控制

崔小红[1,2], 罗艳红[1], 张化光[1†], 祖培福[2]

(1. 东北大学 信息科学与工程学院, 辽宁 沈阳 110819; 2. 牡丹江师范学院 数学科学学院, 黑龙江 牡丹江 157011)

摘要: 针对带有饱和执行器且局部未知的非线性连续系统的有穷域最优控制问题, 设计了一种基于自适应动态规划(ADP)的在线积分增强学习算法, 并给出算法的收敛性证明. 首先, 引入非二次型函数处理控制饱和问题. 其次, 设计一种由常量权重和时变激活函数构成的单一网络, 来逼近未知连续的值函数, 与传统双网络相比减少了计算量. 同时, 综合考虑神经网络产生的残差和终端误差, 应用最小二乘法更新神经网络权重, 并且给出基于神经网络的迭代值函数收敛到最优值的收敛性证明. 最后, 通过两个仿真例子验证了算法的有效性.

关键词: 有穷域; 最优控制; 神经网络; 自适应动态规划

中图分类号: TP273 **文献标识码**: A

Finite-horizon optimal control for unknown systems with saturating control inputs

CUI Xiao-hong[1,2], LUO Yan-hong[1], ZHANG Hua-guang[1†], ZU Pei-fu[2]

(1.School of Information Science and Engineering, Northeastern University, Shenyang Liaoning 110819, China;
2. Institute of Mathematical Sciences, Mudanjiang Normal College, Mudanjiang Heilongjiang 157011, China)

Abstract: An adaptive dynamic programming (ADP)-based online integral reinforcement learning algorithm is designed for finite-horizon optimal control of nonlinear continuous-time systems with saturating control inputs and partially unknown dynamics. Moreover, the convergence of the algorithm is proved. Firstly, the control constraints are handled through non-quadratic function. Secondly, a single neural network (NN) with constant weights and time-dependent activation functions is designed in order to approximate the unknown and continuous value function. Compared with the traditional dual neural networks, the burden of computation by the single NN is lessened. Meanwhile, the NN weights are updated by the least square method with considering both the residual error and terminal error. Furthermore, the convergence of iterative value function on the base of NN is proved. Lastly, two simulation examples show the effectiveness of the proposed algorithm.

Key words: finite-horizon; optimal control; neural network; adaptive dynamic programming

1 引言(Introduction)

20世纪50年代, 为解决最优控制问题, Bellman设计了动态规划方法, 其核心思想是最优化原理, 即无论初始状态和初始决策如何, 余下的策略相对于目前产生的状态而言也必将是最优策略, 因此可以将多阶段决策问题转化一系列单阶段决策问题, 然后逐个加以解决. 但是多数实际现象呈现出非线性特点, 这时求解最优控制问题就转化为求解Hamilton-Jacobi-Bellman(HJB)方程, 但是此方程具有较强的内在非线性, 故很难求解. 而传统的动态规划是按照时间进行反向搜索的, 不仅容易产生“维数灾”问题而且是离线计算的. 于是, Werbos等人[1–3]对此进行改进, 提出了自适应动态规划(adaptive dynamic programming, ADP), 其方法融合了传统的动态规划思想和神经网络(neural network, NN)思想, 设计了按照时间正向求解最优控制问题的一种增强学习方法, 目前已成为解决复杂非线性系统最优控制问题的行之有效的方法之一. Beard等人在文献[4–5]设计Galerkin谱方法逼近HJB方程, 通过逐次逼近法反复地改进控制策略, 最终逼近最优控制. Abu-Khalaf等人在文献[6]提出了策

收稿日期: 2014−12−26; 录用日期: 2016−01−29.

†通信作者. E-mail: hgzhang@ieee.org.

本文责任编委: 胡跃明.

牡丹江市科学技术计划项目(G2015k1991), 牡丹江师范学院一般项目(YB201605), 国家自然科学基金项目(61104010), 中国博士后自然科学基金项目(2012M510825, 2014T70260), 中央高校基本科研基金项目(N140404004)资助.

Supported by Science and Technology Project of Mudanjiang (G2015k1991), General Project of Mudanjiang Normal college (YB201605), National Natural Science Foundation of China (61104010), China Postdoctoral Science Foundation (2012M510825, 2014T70260) and Fundamental Research Funds for Central Universities (N140404004).

略迭代算法, 并给出算法的收敛性证明. 不过前面的方法都是采取离线计算形式. 为此, Vrabie等人[7]结合最小二乘法和积分增强学习思想针对系统未知的情况给出了在线的学习算法. 文献[8–10]在此基础上提出了求解最优控制的可以实时更新权重的在线学习算法. 然而这些文献都是求解无穷域最优控制问题的.

有穷域最优控制近些年受到广泛关注, 但由于HJB方程呈现出时变特点使得有穷域最优控制问题难于处理. 文献[11–14]针对非线性离散系统的有穷域最优控制问题, 给出一种基于迭代思想的自适应动态规划算法, 并把此方法应用到执行器饱和系统和时滞系统的最优控制中. 文献[15–16]针对终端时间固定的非线性系统的有穷域最优控制问题, 采用具有时变权重结构的神经网络进行近似, 通过迭代算法得到时变HJB方程的解. 文献[17]针对含有饱和执行器的非线性离散系统, 采用二次启发式规划(DHP)结构, 利用值迭代算法, 按照时间正向求解得到有穷域最优控制. 但是前面提到的文献都是采取离线计算形式, 计算量大. 虽然文献[18]针对非线性系统, 设计了实时更新神经网路权重的在线学习算法, 但文中要求终端状态的惩罚函数为常函数, 极大限制了它的应用.

本文在上述研究的基础上, 针对带有饱和执行器且模型信息局部未知的非线性系统, 提出一种改进的ADP迭代算法. 本文的贡献如下: 1) 改进的算法按照时间正向更新神经网络权重, 是一种在线的学习算法; 2) 考虑的系统是局部信息未知且含有饱和控制的非线性系统; 3) 借助牛顿迭代算法, 数学上严格证明了在线ADP迭代算法的收敛性.

2 问题描述(Problem description)

考虑一类含有饱和执行器的非线性连续系统

$$\dot{x} = f(x) + g(x)u, \tag{1}$$

其中: $x(t) \in \mathbb{R}^n$是系统的状态且是可测的; $u \in \mathbb{R}^p$是控制输入并且假设是有界的, 即$\|u\| \leqslant \lambda$, 这里$\lambda > 0$. $f(x) \in \mathbb{R}^n,\ g(x) \in \mathbb{R}^{n\times p}$.

假设 1 $f(0) = 0$, 且$f(x), g(x)$在包含原点的紧集$\Omega \subset \mathbb{R}^n$上是Lipschitz连续的, 且$f(x)$未知, 并假设系统(1)在$\Omega$上是可镇定的.

定义如下的有穷域性能指标:

$$J(x_0, u) = \psi(x(t_{\rm f}), t_{\rm f}) + \int_{t_0}^{t_{\rm f}} r(x, u){\rm d}t, \tag{2}$$

其中: $r(x,u) = Q(x)+M(u), Q(x), M(u)$是正定的函数, $\psi(x(t_{\rm f}), t_{\rm f})$是终端状态的惩罚函数.

定义 1[15] 如果一个控制策略μ在集合$\bar{\Omega} = (\Omega \times [t_0, t_{\rm f}])$是连续的, $\mu(0, t_0) = 0$, 且μ在集合Ω上可以镇定系统(1), 并且能使性能指标(2)是有限的, 则称μ为容许控制.

定义相对于容许控制μ的价值函数

$$V(x(t), t, \mu) = \psi(x(t_{\rm f}), t_{\rm f}) + \int_t^{t_{\rm f}} r(x, \mu){\rm d}\tau. \tag{3}$$

控制目标: 选择容许控制μ, 使得上面的价值函数是最小的.

最优的$V^*(x(0), 0)$如下:

$$V^*(x(0), 0) = \min_{\mu}\{\psi(x(t_{\rm f}), t_{\rm f}) + \int_0^{t_{\rm f}} r(x, \mu){\rm d}\tau\}. \tag{4}$$

对于任意的时间$t > T$和任意的时间间隔T, 价值函数(3)的离散形式可以写成

$$V(x(t), t) = \int_t^{t+T} r(x, \mu){\rm d}\tau + V(x(t+T), t+T), \tag{5}$$

其中$t + T \leqslant t_{\rm f}$.

对于任意的容许控制μ, 如果相应的性能指标(3)具有连续的一阶偏导数, 那么它的极小形式称作Lyapunov方程[2, 18].

根据式(5)并利用全微分的定义可以得到

$$\lim_{T\to 0}\int_t^{t+T} \frac{r(x, \mu){\rm d}\tau}{T} = -\lim_{T\to 0}(\frac{V(x(t+T), t+T) - V(x(t), t)}{T}), \tag{6}$$

即

$$r(x, \mu) = -\lim_{T\to 0}(\frac{\partial V}{\partial x}^{\rm T}\frac{\Delta x}{T} + \frac{\partial V}{\partial t} + \frac{{\rm o}(T)}{T}), \tag{7}$$

则有Lyapunov方程

$$r(x, \mu) + \frac{\partial V}{\partial x}^{\rm T}(f(x) + g(x)\mu) + \frac{\partial V}{\partial t} = 0. \tag{8}$$

定义哈密尔顿函数

$$H(x, t, \frac{\partial V}{\partial x}, u) = r(x, u) + \frac{\partial V}{\partial x}^{\rm T}(f(x) + g(x)u) + \frac{\partial V}{\partial t}. \tag{9}$$

考虑执行器饱和的情况, 引入非二次型的函数[6]

$$M(u) = 2\int_0^u \lambda \tanh^{-{\rm T}}(v/\lambda)R{\rm d}v, \tag{10}$$

其中R为正定矩阵, 为方便讨论进一步假设R为对角型矩阵.

应用分部积分公式, 可得

$$M(u) = 2\lambda(\tanh^{-1}(v/\lambda))^{\rm T}Rv|_0^u - 2\int_0^u \lambda({\rm d}(\tanh^{-1}(v/\lambda))^{\rm T}R)v =$$
$$2\lambda(\tanh^{-1}(u/\lambda))^{\rm T}Ru - 2\lambda\int_0^u(\frac{{\rm d}v/\lambda}{\underline{1} - \tanh^2(\tanh^{-1}(v/\lambda))})^{\rm T}Rv =$$
$$2\lambda(\tanh^{-1}(u/\lambda))^{\rm T}Ru - 2\lambda\int_0^u(\frac{{\rm d}v/\lambda}{\underline{1} - (v/\lambda)^2})^{\rm T}Rv \overset{u=-\lambda\tanh(D)}{=} \lambda\frac{\partial V}{\partial x}^{\rm T} g\tanh(D) + \lambda^2\bar{R}\ln(\underline{1} - (u/\lambda)^2), \tag{11}$$

其中: $D=\dfrac{1}{2\lambda}R^{-1}g^{\mathrm{T}}\dfrac{\partial V}{\partial x}$, 且$\underline{1}$是元素都为1的列向量, $\bar{R}$是由R的对角元素构成的行向量.

将非二次型函数$M(u)$代入哈密尔顿函数, 进一步应用稳态条件, 即$\dfrac{\partial H}{\partial u}=0$, 得到满足饱和条件的最优控制

$$\mu^*=-\lambda\tanh(\frac{1}{2\lambda}R^{-1}g^{\mathrm{T}}\frac{\partial V^*}{\partial x}),\tag{12}$$

其中$\dfrac{\partial V^*}{\partial x}$满足下面的时变HJB方程:

$$(\frac{\partial V^*}{\partial x})^{\mathrm{T}}(f(x)-g\lambda\tanh(\frac{1}{2\lambda}R^{-1}g^{\mathrm{T}}\frac{\partial V^*}{\partial x}))+$$
$$Q(x)+M(\mu^*)+\frac{\partial V^*}{\partial t}=0.\tag{13}$$

为找到问题的最优控制, 只需求解HJB方程获得$V^*(x,t)$, 然后代入式(12), 则可以得到最优控制. 但求解HJB方程是相当困难的, 甚至是不可能的. 况且求解HJB方程时需要系统全部的信息已知, 即要求函数$f(x),g(x)$已知, 而在实际问题中很难知道. 因此, 本文在下一章节中给出一种在线的学习算法, 近似求得HJB方程的解.

3　基于神经网络的有穷域最优控制设计 (Finite-horizon optimal control based on neural network)

在有穷域的最优控制问题中, 值函数是时变的, 即$V(x,t)$显含时间t. 因此, 本文设计如下形式的BP神经网络, 其输入为系统状态和剩余时间, 输出为价值函数. 为使问题简化, 这里采用的神经网络关于未知权重是线性的, 结构如下:

$$V=\sum_{i=1}^{\infty}C_i\varphi_i(x(t),t_{\mathrm{f}}-t)=$$
$$C_{\mathrm{M}}^{\mathrm{T}}\phi(x,t_{\mathrm{f}}-t)+\varepsilon_{\mathrm{V}}.\tag{14}$$

这里: $C_{\mathrm{M}}=[C_1\ C_2\ \cdots\ C_m]^{\mathrm{T}}$表示理想的神经网络权重向量, $\phi(x,t_{\mathrm{f}}-t)=[\varphi_1\ \cdots\ \varphi_m]^{\mathrm{T}}\in\mathbb{R}^m$是激活函数向量, ε_{V}是神经网络重建误差, m为隐含层神经元的个数.

本文采用如下函数对V进行估计:

$$\hat{V}(x,t)=W^{\mathrm{T}}\phi(x(t),t_{\mathrm{f}}-t),\tag{15}$$

其中$W=[W_1\ W_2\ \cdots\ W_m]^{\mathrm{T}}$是理想的神经网络权重向量的估计值.

因此, 此时对应的近似控制为

$$\mu=-\lambda\tanh(\frac{1}{2\lambda}R^{-1}g^{\mathrm{T}}\frac{\partial\hat{V}}{\partial x}).$$

由于V用$\hat{V}$替代后将产生残差项, 包括误差

$$\zeta(x(t),t)=$$
$$\int_t^{t+T}r(x,\mu)\mathrm{d}\tau+W^{\mathrm{T}}(\phi(x(t+T),t_{\mathrm{f}}-(t+T))-$$
$$\phi(x(t),t_{\mathrm{f}}-t))\tag{16}$$

和终端误差

$$\varsigma=W^{\mathrm{T}}\phi(x(t_{\mathrm{f}}),0)-\psi(x(t_{\mathrm{f}}),t_{\mathrm{f}}).\tag{17}$$

本文将采用增维方式处理上述两种形式的误差, 形式如下:

$$\delta=[\zeta,\varsigma].\tag{18}$$

同时定义几个增维后的参数

$$\bar{\psi}=[\phi(x(t+T),t_{\mathrm{f}}-(t+T))-\phi(x(t),t_{\mathrm{f}}-t),$$
$$\phi(x(t_{\mathrm{f}}),0)],\tag{19}$$
$$\bar{V}=[\int_t^{t+T}r(x,\mu)\mathrm{d}\tau,-\psi(x(t_{\mathrm{f}}),t_{\mathrm{f}})],\tag{20}$$

则$\delta=W^{\mathrm{T}}\bar{\psi}+\bar{V}$.

应用最小二乘法更新神经网络权重[6], 这里引入内积符号$<f,g>=\int_\Omega f^{\mathrm{T}}g\mathrm{d}x$, 可以得到神经网络权重更新公式如下:

$$W=-\langle\bar{\psi}^{\mathrm{T}},\bar{\psi}^{\mathrm{T}}\rangle_{\bar{\Omega}}^{-1}\langle\bar{\psi}^{\mathrm{T}},\bar{V}\rangle_{\bar{\Omega}}.\tag{21}$$

注 1　选取线性无关的激活函数向量能够保证$\bar{\psi}$线性无关, 从而$\langle\bar{\psi}^{\mathrm{T}},\bar{\psi}^{\mathrm{T}}\rangle_{\bar{\Omega}}$是可逆的.

注 2　神经网络权重公式(21)表面上只有一个解, 而实际上是通过迭代进行的, 每步迭代都会产生一个权重, 在下节中将给出迭代算法.

注 3　本节仅设计一个神经网络去逼近未知的值函数V, 进而给出近似的最优控制, 与传统的执行网–评价网双网络相比, 单网络更大程度上减少了计算量.

4　收敛性证明(Convergence proof)

前面提到神经网络权重更新本质上是以迭代方式进行的, 本节将明确给出ADP迭代算法, 同时给出两方面的收敛性证明: 1) 提出的ADP迭代算法是收敛的; 2) 基于神经网络的迭代值函数能够收敛到最优.

4.1　ADP算法的收敛性证明(Convergence of ADP algorithm)

针对有穷域最优控制问题, 本节首先设计了一种改进的在线ADP迭代算法, 此算法不需要系统全部的信息已知且考虑了系统的终端条件.

选择初始的容许控制$\mu^{(0)}(x(t),t)$[3]按照下面步骤进行迭代:

$$\begin{cases}V^{\mu^{(j)}}(x(t),t)=\int_t^{t+T}r(x,\mu^{(j)})\mathrm{d}\tau+\\ \qquad\qquad V^{\mu^{(j)}}(x(t+T),t+T),\\ V^{\mu^{(j)}}(x(t_{\mathrm{f}}),t_{\mathrm{f}})=\psi(x(t_{\mathrm{f}}),t_{\mathrm{f}}),\end{cases}\tag{22}$$

$$\mu^{(j+1)}(x,t)=-\lambda\tanh(\frac{1}{2\lambda}R^{-1}g^{\mathrm{T}}\frac{\partial V^{\mu^{(j)}}}{\partial x}). \quad (23)$$

注 4 本文设计的ADP迭代算法是一种在线学习算法, 因此持续性激励条件(PE)必不可少, 这里通过每0.1s重置初始状态来实现PE条件[20].

接下来进一步给出改进的ADP迭代算法的收敛性证明. 首先证明本文所提算法数学上等价于牛顿迭代算法, 进而说明此算法收敛.

考虑Banach空间$\Gamma\subset\{V(x,t):\bar{\Omega}\to\mathbb{R}\}$.

定义映射

$$\Upsilon=\begin{cases}r(x,\mu)+\dfrac{\partial V}{\partial t}+\dfrac{\partial V}{\partial x}^{\mathrm{T}}(f(x)+g(x)\mu),\ t<t_{\mathrm{f}},\\ V-\psi(x(t),t), \qquad\qquad t=t_{\mathrm{f}}.\end{cases} \quad (24)$$

在牛顿迭代算法中, 要应用Frechet导数$\Upsilon'(V)$, 但是此导数不易求得, 本文借助Gateaux导数去求解Frechet导数.

定义 2[19] 令$\Upsilon:U(V)\subseteq X\to Y$是一给定的映射, 其中$X$和$Y$是Banach空间, 这里$U(V)$表示$V$的邻域. 映射$\Upsilon$在$V$处是Gateaux可微的当且仅当存在一线性算子$L:X\to Y$使得

$$\Upsilon(V+sM)-\Upsilon(V)=sL(M)+\mathrm{o}(s),\ s\to 0.$$

对于满足$\|M\|_{\bar{\Omega}}=1$且$M\in U(V)$的M, 和所有零附近的s, 这里$\lim\limits_{s\to 0}\mathrm{o}(s)/s=0$, $\mathrm{o}(s)$为s的高阶无穷小量, 则称L为V处的Gateaux导数, 而在V处, Gateaux微分为

$$L(M)=\lim_{s\to 0}\frac{\Upsilon(V+sM)-\Upsilon(V)}{s}.$$

为了证明改进的ADP算法的收敛性, 下面给出两个引理.

引理 1[19] 如果Gateaux导数Υ'在V的邻域存在, 且Gateaux导数Υ'在V处是连续的, 则$L=\Upsilon'(V)$也是V处的Frechet导数.

引理 2[19] Υ如前面定义, 则在V处的Frechet微分为

$$\Upsilon'(V)M=L(M)=\begin{cases}(\dfrac{\partial M}{\partial x})^{\mathrm{T}}(f-g\lambda\tanh D)+\dfrac{\partial M}{\partial t},\ t<t_{\mathrm{f}},\\ M, \qquad\qquad\qquad\qquad t=t_{\mathrm{f}}.\end{cases} \quad (25)$$

证

$$\Upsilon(V+sM)-\Upsilon(V)=$$
$$\begin{cases}Q(x)+(\dfrac{\partial(V+sM)}{\partial x})^{\mathrm{T}}f(x)+\dfrac{\partial(V+sM)}{\partial t}+\lambda^2\bar{R}\ln(\underline{1}-\tanh^2(\dfrac{R^{-1}g^{\mathrm{T}}}{2\lambda}\dfrac{\partial(V+sM)}{\partial x}))-\\ [Q(x)+(\dfrac{\partial V}{\partial x})^{\mathrm{T}}f(x)+\dfrac{\partial V}{\partial t}+\lambda^2\bar{R}\ln(\underline{1}-\tanh^2(\dfrac{R^{-1}g^{\mathrm{T}}}{2\lambda}\dfrac{\partial V}{\partial x}))],\ t<t_{\mathrm{f}},\\ V+sM-\psi(x(t),t)-(V-\psi(x(t),t)), \qquad\qquad t=t_{\mathrm{f}}.\end{cases} \quad (26)$$

上式经过合并处理, 得到Gateaux微分为

$$L(M)=\lim_{s\to 0}\frac{\Upsilon(V+sM)-\Upsilon(V)}{s}=\begin{cases}(\dfrac{\partial M}{\partial x})^{\mathrm{T}}(f-g\lambda\tanh D)+\dfrac{\partial M}{\partial t},\ t<t_{\mathrm{f}},\\ M, \qquad\qquad\qquad\qquad t=t_{\mathrm{f}}.\end{cases} \quad (27)$$

下面证明$L=\Upsilon'(V)$的连续性.

$\forall M_0\in\Gamma$, 应用式(27), 可以得到

$$\|L(M)-L(M_0)\|_{\bar{\Omega}}=\begin{cases}\|(\dfrac{\partial(M-M_0)}{\partial x})^{\mathrm{T}}[f-g\lambda\tanh D]+\dfrac{\partial(M-M_0)}{\partial t}\|_{\bar{\Omega}},\ t<t_{\mathrm{f}}\\ \|M-M_0\|_{\bar{\Omega}}, \qquad\qquad\qquad\qquad t=t_{\mathrm{f}}\end{cases}\leqslant$$
$$\begin{cases}(\|f\|_{\bar{\Omega}}+\|g\lambda\tanh D\|_{\bar{\Omega}})\|\dfrac{\partial(M-M_0)}{\partial x}\|_{\bar{\Omega}}+\|\dfrac{\partial(M-M_0)}{\partial t}\|_{\bar{\Omega}},\ t<t_{\mathrm{f}}\\ \|M-M_0\|_{\bar{\Omega}}, \qquad\qquad\qquad\qquad t=t_{\mathrm{f}}\end{cases}\leqslant$$
$$\begin{cases}(\|f\|_{\bar{\Omega}}+\|g\lambda\tanh D\|_{\bar{\Omega}})\alpha_1\|M-M_0\|_{\bar{\Omega}}+\alpha_2\|M-M_0\|_{\bar{\Omega}},\ t<t_{\mathrm{f}}\\ \|M-M_0\|_{\bar{\Omega}}, \qquad\qquad\qquad\qquad t=t_{\mathrm{f}}\end{cases}\leqslant\beta(M-M_0),\ \forall t\leqslant t_{\mathrm{f}}, \quad (28)$$

其中:

$$\beta=\max\{(\|f\|_{\bar{\Omega}}+\|g\lambda\tanh D\|_{\bar{\Omega}})\alpha_1+\alpha_2,1\},$$
$$\alpha_1>0,\ \alpha_2>0.$$

根据连续的定义, 易证$L=\Upsilon'(V)$是连续函数.

应用引理1能够得到$L(M)$同样为Frechet微分.

证毕.

接下来, 本文通过定理1阐述ADP算法在数学上等价于牛顿迭代算法, 从而说明ADP算法的收敛性.

定理 1　由式(22)−(23)构成的在线的ADP算法数学上等价于牛顿迭代算法, 即

$$V^{(j+1)}=V^{(j)}-\frac{\Upsilon(V^{(j)})}{\Upsilon'(V^{(j)})},\ j=1,2,\cdots. \tag{29}$$

证　通过引理2和前面Υ的定义, 得到

$$\Upsilon'(V^{(j)})V^{(j)}-\Upsilon(V^{(j)})=$$
$$\begin{cases}-Q(x)-[\lambda(\dfrac{\partial V^{(j)}}{\partial x})^{\mathrm{T}}g\tanh(D_j)+\\ \lambda^2\bar{R}\ln(\underline{1}-(u^{(j+1)}/\lambda)^2)],\ t<t_{\mathrm{f}},\\ V^{(j)},\qquad\qquad\qquad t=t_{\mathrm{f}},\end{cases}=$$
$$\begin{cases}-r(x,u^{(j+1)}(x,t)),\ t<t_{\mathrm{f}},\\ \psi(x(t),t),\qquad\qquad t=t_{\mathrm{f}}.\end{cases} \tag{30}$$

其中$D_j=\dfrac{1}{2\lambda}R^{-1}g^{\mathrm{T}}\dfrac{\partial V^{(j)}}{\partial x}$. $\Upsilon'(V^{(j)})V^{(j+1)}$可写成

$$\Upsilon'(V^{(j)})V^{(j+1)}=$$
$$\begin{cases}(\dfrac{\partial V^{(j+1)}}{\partial x})^{\mathrm{T}}(f-g\lambda\tanh D_j)+\dfrac{\partial V^{(j+1)}}{\partial t},\ t<t_{\mathrm{f}}\\ V^{(j+1)},\qquad\qquad\qquad\qquad t=t_{\mathrm{f}}\end{cases}=$$
$$\begin{cases}-r(x,u^{(j+1)}(x,t)),\ t<t_{\mathrm{f}},\\ \psi(x(t),t),\qquad\qquad t=t_{\mathrm{f}}.\end{cases} \tag{31}$$

因此

$$\Upsilon'(V^{(j)})V^{(j+1)}=\Upsilon'(V^{(j)})V^{(j)}-\Upsilon(V^{(j)}),\ \forall t\leqslant t_{\mathrm{f}}.$$

证毕.

进一步根据牛顿迭代算法, 由初始的容许控制出发可以推导出值函数$V^{(j)}$能够收敛到最优的值函数$V^*(j\to\infty)$.

4.2　近似值函数$\hat{V}^{(j)}$的收敛性(Convergence of $\hat{V}^{(j)}$)

定理 2　若下面的条件成立:

1) 系统动态和相应的性能指标能够保证Lyapunov方程具有连续可导的正定解;

2) 存在线性无关且完备的神经网络激活函数, 可以一致近似V;

3) 神经网路激活函数的梯度是线性无关的且是完备的, 则有下面结论成立$\hat{V}^{(j)}\to V^{(j)}(m\to\infty)$.

证　令$\bar{\varphi}_i=\varphi_i(x(t+T),t_{\mathrm{f}}-(t+T))-\varphi_i(x(t),t_{\mathrm{f}}-t)$, 由假设4[6]可知, $\{\bar{\varphi}_i\}_1^\infty$是完备的, 则$\delta^{\mu^{(j)}}\to 0(m\to\infty)$.

又$\delta^{\mu^{(j)}}=(W-C_{\mathrm{M}})^{\mathrm{T}}\bar{\psi}-\sum\limits_{k=m+1}^{\infty}C_k\bar{\varphi}_k$, 其中: $C_{\mathrm{M}}=[C_1\ \cdots\ C_m]$, C_k是$\{C_k\}_1^\infty$的第k个元素, $\bar{\psi}=[\bar{\varphi}_1\ \cdots\ \bar{\varphi}_m]$. 因此有

$$(W-C_{\mathrm{M}})^{\mathrm{T}}\bar{\psi}=\delta^{\mu^{(j)}}+\sum_{k=m+1}^{\infty}C_k\bar{\varphi}_k=\delta^{\mu^{(j)}}+\epsilon. \tag{32}$$

又根据魏尔斯特拉斯逼近定理[6], $\epsilon\to 0(m\to\infty)$, 于是有$(W-C_{\mathrm{M}})^{\mathrm{T}}\bar{\psi}\to 0(m\to\infty)$.

根据引理5[7]的必要性, 可证明$W\to C_{\mathrm{M}}(m\to\infty)$. 又神经网络激活函数是线性无关的和完备性的, 再一次根据此引理的充分性得到$(W-C_{\mathrm{M}})^{\mathrm{T}}\times\phi\to 0(m\to\infty)$, 即$\hat{V}^{(j)}\to V^{(j)}(m\to\infty)$.

定理 3　满足定理2的假设条件, 则有

$$\sup_{x\in\Omega}|\hat{V}^{(j)}-V^*|\to 0. \tag{33}$$

证　通过定理1和定理2易得定理3结论.

注 5　定理2的假设条件2存在线性无关且完备的激活函数, 使之可以一致近似理想的值$V^{(j)}$, 应用魏尔斯特拉斯定理, 这个条件是可以实现的. 但在实际选取神经网络激活函数时, 其个数是有限的. 因此, $\hat{V}^{(j)}$与$V^{(j)}$之间是存在逼近误差的, 只有当神经元个数选取的充分大时, $\hat{V}^{(j)}$才能很好的趋近$V^{(j)}$. 所以, 在实际操作中, 会根据系统本身的特点选择足够多的神经元尽量减小其与真实函数的误差.

5　仿真(Simulation)

本文所提算法适用于一般形式的惩罚函数$M(u)$, 当然也包括二次型形式的惩罚函数, 这一结果将通过仿真算例1来验证. 在仿真例子2中, 由于含有饱和控制, 引入关于控制u的非二次型的惩罚函数$M(u)$, 按照本文算法进行迭代. 两个仿真例子从多个角度验证了本文算法的有效性.

例 1　考虑一阶线性系统

$$\dot{x}=-\frac{x}{2}+v, \tag{34}$$

选取性能指标为$J=5x(t_{\mathrm{f}})^2+\int_0^{t_{\mathrm{f}}}(x^2+\dfrac{v^2}{2})\mathrm{d}t$. 根据最优控制理论, 通过求解Riccati方程, 能够求得最优控制为

$$v^*=-x(1+1.5\exp(-3\tau))/(1-0.75\exp(-3\tau)), \tag{35}$$

其中: $\tau = t_{\rm f} - t,\ t_{\rm f} = 1.$

通过观察, 发现式(35)中含有指数项. 因此, 选取如下形式的神经网络激活函数:

$$[x\ \ \tau\ \ x^2(1+1.5\exp(-3\tau))/(1-0.75\exp(-3\tau))].$$

选取神经网络初始权重为$W_0 = [0.1\ \ 0.5\ \ 1]^{\rm T}$和时间间隔$T = 0.01$. 根据本文算法能够计算出神经网络权重收敛到$W^* = [0\ \ 0\ \ 0.5]^{\rm T}$, 权重收敛情况见图1. 进一步计算最优控制, 结果同式(35)一致, 说明本文算法是有效的. 同以往的依赖于模型的算法相比[15,17], 本文算法优势在于不需要系统模型知识完全已知. 图2描绘了在最优控制下状态发展趋势, 可以发现随着时间向前推进, 系统状态响应是向零趋近的.

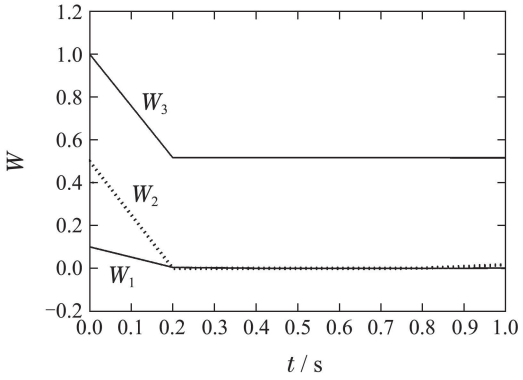

图 1 神经网络权重收敛图

Fig. 1 History of NN weights

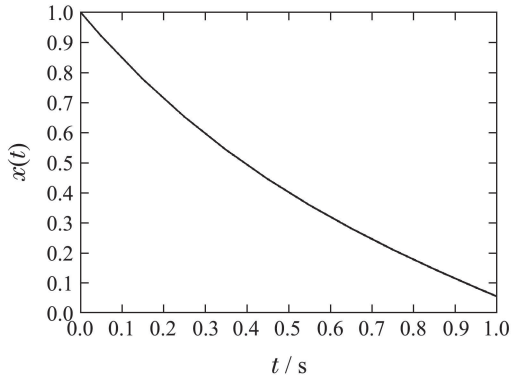

图 2 状态曲线图

Fig. 2 Curve of state

例 2 考虑如下的非线性仿射系统:

$$\dot{x} = f(x) + g(x)u,\ x \in \mathbb{R}^2, \tag{36}$$

这里:

$$f(x) = \begin{bmatrix} x_2 - 2x_1 \\ -x_2 - \dfrac{1}{2}x_1 + \dfrac{1}{4}x_2((\cos(2x_1)+2)^2 - \\ (\sin(4x_1^2)+2)^2) \end{bmatrix}, \tag{37}$$

$$g(x) = \begin{bmatrix} 0 \\ \cos(2x_1)+2) \end{bmatrix}. \tag{38}$$

选取$\psi = 10x(t_{\rm f})^{\rm T}x(t_{\rm f}),\ Q = 2I,\ R = 2I,\ t_{\rm f} = 2$. 系统的饱和界$\lambda = 1$. 同时选取神经网络激活函数为

$$\phi = [x_1\ \ x_2\ \ x_1^2\exp(-(t_{\rm f}-t))\ \ x_2^2\exp(-(t_{\rm f}-t))].$$

选取神经网络初始权重为$W_0 = [1\ \ 0.1\ \ 8\ \ 7]^{\rm T}$和时间间隔$T = 0.01$. 图3描述了神经网络权重收敛情况, 可以发现经过8步迭代, 每步迭代时间0.25 s, 神经网络权重收敛到常数值$W = [-0.0018\ \ -0.0203\ \ 9.9382\ \ 10.0100]^{\rm T}$, 则对应的最优控制为

$$u^* = -\tanh(5.005(\cos(2x_1)+2)x_2\exp(-(t_{\rm f}-t))).$$

图4 描绘了最优控制下的状态曲线图, 系统的状态很好的向零趋近. 图5详细绘制了初始的容许控制(不考虑饱和限制)和最优的控制(考虑饱和限制)的图形. 可以发现, 不考虑饱和限制的初始容许控制很快超出了饱和界$\lambda = 1$, 最终无法形成满足饱和条件的最优控制. 对比发现, 考虑饱和限制, 利用文中的算法, 每步迭代得到的控制和最终的最优控制都不会超过饱和界, 说明本文设计的算法很好的处理了饱和约束的问题.

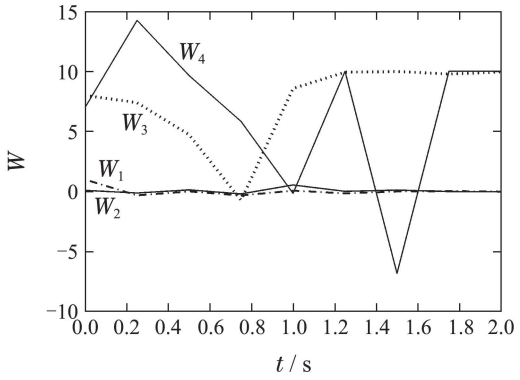

图 3 神经网络权重收敛图

Fig. 3 History of NN weights

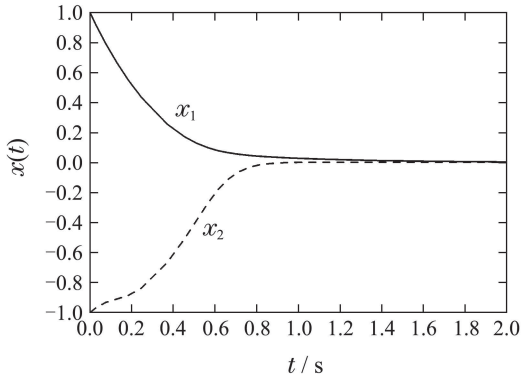

图 4 状态曲线图

Fig. 4 Curve of states

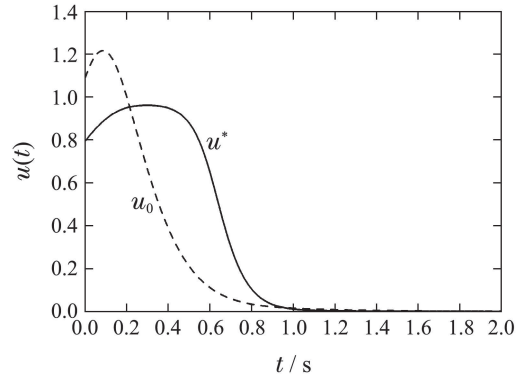

图 5 最优控制(实线)与初始控制(虚线)曲线对比图

Fig. 5 Comparison between optimal control(solid) and initial control(dashed)

6 结论(Conclusions)

本文针对模型局部未知且含有饱和执行器的非线性系统, 基于自适应动态规划方法, 提出一种改进的ADP在线学习算法. 借助人工神经网络对值函数进行近似, 同时考虑两方面的误差并使用最小二乘法更新神经网络权重. 文中给出了迭代算法和近似值函数的收敛性证明. 最后, 仿真结果表明所提算法是有效的. 在今后的研究工作中, 笔者将致力于研究神经网络激活函数的设计、初始容许控制的选取、非线性时延系统的有穷域最优控制的求解等问题.

参考文献(References):

[1] WERBOS P J. *Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches* [M]. New York: Van Nostrand Reinhold, 1992: 23 – 38.

[2] LEWIS F L, SYRMOS V L. *Optimal Control* [M]. New York: Wiley, 1995: 213 – 260.

[3] LIU D, WEI Q. Policy iteration adaptive dynamic programming algorithm for discrete-time nonlinear systems [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2014, 25(3): 621 – 634.

[4] BEARD R W. *Improving the closed-loop performance of nonlinear systems* [D]. Troy: Rensselaer Polytechnic Institute, 1995.

[5] BEARD R W, SARIDIS G N, WEN J T. Approximate solutions to the time-invariant Hamilton-Jacobi-Bellman equation [J]. *Journal of Optimization Theory and Applications*, 1998, 96(3): 589 – 626.

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

[7] VRABIE D, LEWIS F. Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems [J]. *Neural Networks*, 2009, 22(3): 237 – 246.

[8] ZHANG H G, QIN C B, LUO Y H, Neural-network-based constrained optimal control scheme for discrete-time switched nonlinear system using dual heuristic programming [J]. *IEEE Transactions on Automation Science and Engineering*, 2014, 11(3): 839 – 849.

[9] ZHANG H G, QIN C B, JIANG B, et al. Online adaptive policy learning algorithm for H_∞ state feedback control of unknown affine nonlinear discrete-time systems [J]. *IEEE Transactions on Cybernetics*, 2014, 44(12): 2706 – 2718.

[10] ZHANG H G, ZHANG J L, YANG G H, et al, Leader-based optimal coordination control for the consensus problem of multi-agent differential games via fuzzy adaptive dynamic programming [J]. *IEEE Transactions on Fuzzy Systems*, 2015, 23(1): 152 – 163.

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

[12] SONG R Z, ZHANG H G. 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] LIN X F, HUANG Y J, CAO N Y. Optimal control scheme for nonlinear systems with saturating actuator using ε-iterative adaptive dynamic programming [C] //*UKACC International Conference on Control*. Cardiff: IEEE, 2012: 3 – 5.

[14] LIN X F, CAO N Y, LIN Y Z. Optimal control for a class of nonlinear systems with state delay based on adaptive dynamic programming withε Error bound [C] //*IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning*. Singapore: IEEE, 2013: 170 – 175.

[15] CHENG T, LEWIS F L, ABU-KHALAF M. A neural network solution for fixed-final time optimal control of nonlinear systems [J]. *Automatica*, 2007, 43(3): 482 – 490.

[16] HEYDARI A, BALAKRISHNAN S N. Fixed-final-time optimal tracking control of input-affine nonlinear systems [J]. *Neurocomputing*, 2014, 129(4): 528 – 539.

[17] HEYDARI A, BALAKRISHNAN S N. Finite-horizon control-constrained nonlinear optimal control using single network adaptive critics [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2013, 24(1): 145 – 157.

[18] XU H, ZHAO Q M, DIERKS T, et al. Neural network-based finite-horizon approximately optimal control of uncertain affine nonlinear continuous-time systems [C] //*American Control Conference (ACC)*. Portland: IEEE, 2014: 1243 – 1248.

[19] WU H, LUO B. Neural network based online simultaneous policy update algorithm for solving the HJI equation in nonlinear H_∞ control [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2012, 23(12): 1884 – 1895.

[20] LUO B, WU H N. Data-based approximation policy iteration for affine nonlinear continuous-time optimal control design [J]. *Automatica*, 2014, 50: 3281 – 3290.

作者简介:

崔小红 (1982–) 女, 博士研究生, 讲师, 主要研究方向为自适应动态规划、最优控制, E-mail: xiaohong19821206@126.com;

罗艳红 (1981–) 女, 副教授, 主要研究方向为近似最优控制和神经网络控制, E-mail: neuluo@gmail.com;

张化光 (1959–) 男, 教授, 长江学者特聘教授, 博士生导师, 主要研究方向为神经网络控制和模糊控制, E-mail: hgzhang@ieee.org;

祖培福 (1981–) 男, 副教授, 主要研究方向为近似最优控制和神经网络控制, E-mail: zpf007007@163.com.