doi: 10.3969/j.issn.1005-3026.2014.04.002

带饱和执行器的非线性离散时滞系统的最优控制

王 涛, 罗艳红
(东北大学 信息科学与工程学院, 辽宁 沈阳 110819)

摘 要: 主要针对带有饱和执行器的时滞非线性离散时间系统更加一般的形式,通过启发式动态规划(HDP)算法求解无限时间最优控制策略问题,并在值函数中引入折扣因子. 首先通过迭代HDP算法给出值函数序列和相应的控制序列,并给出了收敛性证明,即值函数序列收敛到值函数的最优值,以及控制序列收敛到最优控制;其次为了实现HDP算法,引入3个神经网络:模型网络、评判网络、控制作用网络. 模型网络用来近似系统模型,评判网络用来近似值函数,控制作用网络用来近似控制;最后通过一个仿真例子说明上述方法的可行性.

关 键 词: 近似动态规划;启发式动态规划;值函数;神经网络;最优控制
中图分类号: TP 273.1 **文献标志码**: A **文章编号**: 1005-3026(2014)04-0461-05

Optimal Control for Nonlinear Discrete-Time Time Delay Systems with Saturating Actuators

WANG Tao, *LUO Yan-hong*
(School of Information Science & Engineering, Northeastern University, Shenyang 110819, China. Corresponding author: WANG Tao, E-mail: wtnuhai@163.com)

Abstract: For the more general form of nonlinear discrete-time time delays systems with saturating actuators, an infinite-time optimal control scheme was developed by heuristic dynamic programming (HDP) algorithm. In the proposed scheme, the discount factor was added in the value function. Firstly, value function series and control series were given through iterative HDP algorithm, and the convergence analysis was presented to prove that value function series and control series reach the optimal value simultaneously. Secondly, three neural networks (NN) which are model NN, critic NN, action NN were introduced to carry out the HDP algorithm. Model NN was used to approximate system model, critic NN to approximate value function, action NN to approximate control policy. Lastly, the validity of HDP algorithm was illustrated by one simulation example.

Key words: approximate dynamic programming; heuristic dynamic programming; value function; neural networks; optimal control

非线性系统的最优控制问题一直是控制领域的研究热点. 如果系统是线性的且值函数关于状态和控制是二次型的,那么最优控制是状态的线性反馈,控制增益矩阵就可以通过求解Riccati方程得到;如果系统是非线性的或性能指标关于状态和控制是非二次型的,那么最优控制需要求解Hamilton-Jacobi-Bellman(HJB)方程[1]. 但是HJB方程固有的非线性特性,往往很难得到其解析解,为了获得HJB方程的近似解,近似动态规划(ADP)方法得到了广泛的关注.

Murray[2]采用ADP算法给出未知的非线性连续时间系统的最优值函数. Tamimi[3]利用ADP算法给出非线性离散时间系统的HJB方程的解,即最优值函数,并给出了收敛性的证明. Werbos[4]将ADP方法分为启发式动态规划(HDP)、二次启发式动态规划(DHP)、执行依赖启发式动态规

收稿日期: 2013-01-14
基金项目: 国家自然科学基金资助项目(50977008,61034005); 国家基础研究计划项目(2009CB320601); 辽宁省自然科学基金资助项目(201202201).
作者简介: 王 涛(1979-),男,辽宁丹东人,东北大学博士研究生,沈阳师范大学讲师.

划(ADHDP)及执行依赖二次启发式动态规划(ADDHP).

在实际问题中,控制系统往往存在状态时滞、控制饱和等现象,这些现象可能导致控制系统不稳定,所以受到很多研究者的关注. Sussmann 等[5]和 Saberi 等[6]给出了带有饱和执行器的线性系统最优控制方法. Luo 等[7]利用贪婪迭代 HDP 算法给出了带有饱和执行器的非线性离散时间系统的近似最优控制. Wei 等[8]通过迭代 ADP 算法给出带有时滞的非线性离散时间系统的最优控制. Song 等[9]利用 HDP 算法解决了带有饱和执行器的非线性离散时间系统的最优控制.

本文在文献[7,9]的基础上,针对具有更加一般形式的带有饱和执行器的非线性离散时滞系统,讨论了它的最优控制问题,并且在值函数中引入了折扣因子. 利用迭代 HDP 算法给出了值函数序列和控制序列,并证明其收敛性,最后采用 BP 神经网络实现迭代 HDP 算法,仿真结果验证了所提算法的有效性.

1 问题陈述

考虑带有饱和执行器的非线性离散时滞系统的一般形式,其状态方程为

$$\boldsymbol{x}(k+1)=\boldsymbol{F}(\boldsymbol{x}(k),\boldsymbol{x}(k-\tau),\boldsymbol{u}(\boldsymbol{x}(k))),k\geqslant 0. \tag{1}$$

式中:$\boldsymbol{x}(k),\boldsymbol{x}(k-\tau)\in\mathbf{R}^n$ 分别为系统的状态变量和时滞状态变量;状态时滞 τ 为正整数.

初始状态:$\boldsymbol{x}(s)=\boldsymbol{\theta}(s),s=-\tau,-\tau+1,\cdots,0,\boldsymbol{u}(\boldsymbol{x}(k))\in\mathbf{R}^m$ 为系统的控制变量,为书写方便,记 $\boldsymbol{x}(k)$ 为 $\boldsymbol{x}_k,\boldsymbol{x}(k-\tau)$ 为 $\boldsymbol{x}_{k-\tau},\boldsymbol{u}(\boldsymbol{x}(k))$ 为 $\boldsymbol{u}_k$.

模型假定:

1)系统(1)完全能控,即存在控制 $\boldsymbol{u}_k$ 使得 $\boldsymbol{x}_{k+1}=\boldsymbol{F}(\boldsymbol{x}_k,\boldsymbol{x}_{k-\tau},\boldsymbol{u}_k)\to 0,k\to\infty$;

2)$\boldsymbol{F}(\cdot)$ 是利普希茨(Lipschitz)连续,且 $\boldsymbol{F}(0,0)=0$;

3)$\boldsymbol{u}_k\in\boldsymbol{\Omega}_u,\boldsymbol{\Omega}_u=\{\boldsymbol{u}_k=(\boldsymbol{u}_k(1),\boldsymbol{u}_k(2),\cdots,\boldsymbol{u}_k(m))^{\mathrm{T}}\mid|\boldsymbol{u}_k(i)|\leqslant\bar{\boldsymbol{u}}(i),i=1,2,\cdots,m\}$,$\bar{\boldsymbol{u}}(i)$ 表示第 i 个控制 $\boldsymbol{u}_k(i)$ 的饱和上界.

任何一个控制过程都必须有一个度量其好坏的准则——值函数(性能指标),本文的问题是如何确定最优控制 $\boldsymbol{u}_k$,使下列值函数达到最小值:

$$\boldsymbol{J}(\boldsymbol{x}_k,\underline{\boldsymbol{u}}_k^{\infty})=\sum_{i=k}^{\infty}\gamma^{i-k}\boldsymbol{U}(\boldsymbol{X}_i,\boldsymbol{u}_i). \tag{2}$$

式中:$\underline{\boldsymbol{u}}_k^{\infty}=(\boldsymbol{u}_k,\boldsymbol{u}_{k+1},\cdots)$ 为控制序列;$\boldsymbol{X}_i=(\boldsymbol{x}_i^{\mathrm{T}},\boldsymbol{x}_{i-\tau}^{\mathrm{T}})^{\mathrm{T}}\in\mathbf{R}^{2n}$;$0<\gamma\leqslant 1$ 为折扣因子;$\boldsymbol{U}(\boldsymbol{X}_i,\boldsymbol{u}_i)=\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{X}_i+\boldsymbol{W}(\boldsymbol{u}_i)$,$\boldsymbol{W}(\boldsymbol{u}_i)=2\int_0^{\boldsymbol{u}_i}\boldsymbol{\phi}^{-\mathrm{T}}(\bar{\boldsymbol{U}}^{-1}s)\bar{\boldsymbol{U}}\boldsymbol{R}\mathrm{d}s$,$\boldsymbol{R}>0,\boldsymbol{Q}\in\mathbf{R}^{2n\times 2n}\geqslant 0,\bar{\boldsymbol{U}}=\mathrm{diag}(\bar{\boldsymbol{u}}(1),\bar{\boldsymbol{u}}(2),\cdots,\bar{\boldsymbol{u}}(m))$,$\boldsymbol{\phi}^{-1}(\boldsymbol{u}_i)=(\boldsymbol{\varphi}^{-1}(\boldsymbol{u}_i(1)),\boldsymbol{\varphi}^{-1}(\boldsymbol{u}_i(2)),\cdots,\boldsymbol{\varphi}^{-1}(\boldsymbol{u}_i(m)))^{\mathrm{T}}$,这里要求 $\boldsymbol{\varphi}(\cdot)$ 是连续可积且具有任意阶导数的单调递增的奇函数,通常取 $|\boldsymbol{\varphi}(\cdot)|\leqslant 1$. 这样的函数 $\boldsymbol{\varphi}(x)$ 很常见,例如双曲正切函数 $\tanh(\boldsymbol{x})=\dfrac{\mathrm{e}^{x}-\mathrm{e}^{-x}}{\mathrm{e}^{x}+\mathrm{e}^{-x}}$.

注 1 $\boldsymbol{\varphi}(\boldsymbol{x})$ 为单调递增奇函数,且 $\boldsymbol{R}>0$,$\bar{\boldsymbol{U}}\geqslant 0$,由定积分性质易知,$\boldsymbol{W}(\boldsymbol{u}_i)>0$. 而 $\boldsymbol{Q}\geqslant 0$,所以 $\boldsymbol{U}(\boldsymbol{X}_i,\boldsymbol{u}_i)>0$,且 $\boldsymbol{U}(0,0)=0$.

下面的任务是如何选择控制序列 $\underline{\boldsymbol{u}}_k^{\infty}$ 使式(2)最小,同时使系统(1)稳定,为此引出容许控制的定义.

定义 1[3] 如果控制序列 $\underline{\boldsymbol{u}}_k^{\infty}$ 满足以下条件:在紧集 $\boldsymbol{\Omega}\subset\mathbf{R}^n$ 上连续;$\boldsymbol{u}_i(0)=0,i=k,k+1,\cdots$;$\underline{\boldsymbol{u}}_k^{\infty}$ 使系统(1)稳定;$\boldsymbol{J}(\boldsymbol{x}_k,\underline{\boldsymbol{u}}_k^{\infty})$ 是有限的. 那么称 $\underline{\boldsymbol{u}}_k^{\infty}$ 为容许控制序列.

式(2)最小值[10]为

$$\boldsymbol{J}^*(\boldsymbol{x}_k)=\inf_{\underline{\boldsymbol{u}}_k^{\infty}}\boldsymbol{J}(\boldsymbol{x}_k,\underline{\boldsymbol{u}}_k^{\infty})=\inf_{\underline{\boldsymbol{u}}_k^{\infty}}\Big(\sum_{i=k}^{\infty}\gamma^{i-k}\boldsymbol{U}(\boldsymbol{X}_i,\boldsymbol{u}_i)\Big). \tag{3}$$

由式(2)知,

$$\begin{aligned}\boldsymbol{J}(\boldsymbol{x}_k,\underline{\boldsymbol{u}}_k^{\infty})&=\sum_{i-k}^{\infty}\gamma^{i-k}\boldsymbol{U}(\boldsymbol{X}_i,\boldsymbol{u}_i)=\\&\boldsymbol{X}_k^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{X}_k+\boldsymbol{W}(\boldsymbol{u}_k)+\gamma\sum_{i=k+1}^{\infty}\gamma^{i-k-1}\boldsymbol{U}(\boldsymbol{X}_i,\boldsymbol{u}_i)=\\&\boldsymbol{X}_k^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{X}_k+\boldsymbol{W}(\boldsymbol{u}_k)+\gamma\boldsymbol{J}(\boldsymbol{x}_{k+1},\underline{\boldsymbol{u}}_{k+1}^{\infty}).\end{aligned}$$

根据 Bellman 最优性原理,最优值函数与最优控制分别为

$$\boldsymbol{V}^*(\boldsymbol{x}_k)=\min_{\boldsymbol{u}_k}\{\boldsymbol{X}_k^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{X}_k+\boldsymbol{W}(\boldsymbol{u}_k)+\gamma\boldsymbol{V}^*(\boldsymbol{x}_{k+1})\}=\min_{\boldsymbol{u}_k}\left\{\begin{matrix}\boldsymbol{X}_k^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{X}_k+\boldsymbol{W}(\boldsymbol{u}_k)+\\\gamma\boldsymbol{V}^*(\boldsymbol{F}(\boldsymbol{x}_k,\boldsymbol{x}_{k-\tau},\boldsymbol{u}_k))\end{matrix}\right\}, \tag{4}$$

$$\boldsymbol{u}^*(\boldsymbol{x}_k)=\arg\min_{\boldsymbol{u}_k}\{\boldsymbol{X}_k^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{X}_k+\boldsymbol{W}(\boldsymbol{u}_k)+\gamma\boldsymbol{V}^*(\boldsymbol{x}_{k+1})\}=\arg\min_{\boldsymbol{u}_k}\left\{\begin{matrix}\boldsymbol{X}_k^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{X}_k+\boldsymbol{W}(\boldsymbol{u}_k)+\\\gamma\boldsymbol{V}^*(\boldsymbol{F}(\boldsymbol{x}_k,\boldsymbol{x}_{k-\tau},\boldsymbol{u}_k))\end{matrix}\right\}. \tag{5}$$

2 迭代 HDP 算法的推导

在迭代 HDP 算法中,值函数序列和控制序列通过迭代进行求解,迭代指标 i 从 0 变化到 ∞. 首

先从值函数 $V_0(x_k)=0$ 开始,那么初始控制

$$u_0(x_k)=\arg\min_{u_k}\{X_k^{\mathrm{T}}QX_k+W(u_k)+\gamma V_0(x_{k+1})\}=\arg\min_{u_k}\{X_k^{\mathrm{T}}QX_k+W(u_k)\}. \tag{6}$$

对应的值函数更新为

$$V_1(x_k)=\min_{u_k}\{X_k^{\mathrm{T}}QX_k+W(u_k)\}=X_k{}^{\mathrm{T}}QX_k+W(u_0(x_k)). \tag{7}$$

对于 $i=1,2,\cdots$,迭代 HDP 算法按下列方式进行迭代:

$$u_i(x_k)=\arg\min_{u_k}\{X_k^{\mathrm{T}}QX_k+W(u_k)+\gamma V_i(x_{k+1})\}=\arg\min_{u_k}\left\{\begin{matrix}X_k^{\mathrm{T}}QX_k+W(u_k)+\\ \gamma V_i(F(x_k,x_{k-\tau},u_k))\end{matrix}\right\}, \tag{8}$$

$$V_{i+1}(x_k)=\min_{u_k}\{X_k^{\mathrm{T}}QX_k+W(u_k)+\gamma V_i(x_{k+1})\}=X_k^{\mathrm{T}}QX_k+W(u_i(x_k))+\gamma V_i(F(x_k,x_{k-\tau},u_i(x_k))). \tag{9}$$

3　迭代 HDP 算法收敛性的证明

设值函数和控制策略按照式(8)、式(9)迭代,下面证明当 $i\to\infty$ 时,值函数序列收敛到最优值,即 $V_i(x_k)\to V^*(x_k)$,控制序列收敛到最优控制,即 $u_i(x_k)\to u^*(x_k)$.

引理 1[11]　设 $\mu_i(x_k)$ 是任意控制策略,定义值函数

$$\Lambda_{i+1}(x_k)=X_k^{\mathrm{T}}QX_k+W(\mu_i(x_k))+\gamma V_i(F(x_k,x_{k-\tau},\mu_i(x_k))). \tag{10}$$

$V_{i+1}(x_k)$ 定义见式(9). 如果 $V_0(\cdot)=\Lambda_0(\cdot)=0$,那么 $V_{i+1}(x_k)\leqslant\Lambda_{i+1}(x_k)$,$i=0,1,2,\cdots$.

引理 2　设系统(1)完全可控,$V_{i+1}(x_k)$ 定义见式(9),则一定存在上界 $Y(x_k)$,使得 $0\leqslant V_{i+1}(x_k)\leqslant Y(x_k)$.

证明　设 $\eta(x_k)$ 是任一容许控制,定义值函数

$$Z_{i+1}(x_k)=X_k^{\mathrm{T}}QX_k+W(\eta(x_k))+\gamma Z_i(F(x_k,x_{k-\tau},\eta(x_k))), \tag{11}$$

这里要求 $Z_0(\cdot)=V_0(\cdot)=0$.

$$Z_{i+1}(x_k)-Z_i(x_k)=\gamma(Z_i(x_{k+1})-Z_{i-1}(x_{k+1}))=\gamma^2(Z_{i-1}(x_{k+2})-Z_{i-2}(x_{k+2}))=\cdots=\gamma^i(Z_1(x_{k+i})-Z_0(x_{k+i}))=\gamma^iZ_1(x_{k+i}),$$

故

$$Z_{i+1}(x_k)=\gamma^iZ_1(x_{k+i})+Z_i(x_k)=\gamma^iZ_1(x_{k+i})+\gamma^{i-1}Z_1(x_{k+i-1})+Z_{i-1}(x_k)=\cdots=\gamma^iZ_1(x_{k+i})+\gamma^{i-1}Z_1(x_{k+i-1})+\gamma^{i-2}Z_1(x_{k+i-2})+\cdots+Z_1(x_k).$$

由此可得

$$Z_{i+1}(x_k)=\sum_{j=0}^{i}\gamma^jZ_1(x_{k+j})=\sum_{j=0}^{i}\gamma^j(X_{k+j}^{\mathrm{T}}QX_{k+j}+W(\eta(x_{k+j})))\leqslant\sum_{j=0}^{\infty}\gamma^j(X_{k+j}^{\mathrm{T}}QX_{k+j}+W(\eta(x_{k+j}))).$$

因为 $\eta(x_k)$ 是容许控制,所以 $Z_{i+1}(x_k)\leqslant\sum_{j=0}^{\infty}\gamma^j(X_{k+j}^{\mathrm{T}}QX_{k+j}+W(\eta(x_{k+j})))\leqslant Y(x_k)$. 由引理 1和引理 2 可得如下定理.

定理 1　设 $u_i(x_k)$ 和 $V_{i+1}(x_k)$ 定义如式(8)、式(9)所示,如果 $V_0(\cdot)=0$,那么 $V_{i+1}(x_k)\geqslant V_i(x_k)$,即 $\{V_i(x_k)\}$ 是单调递增序列.

证明　设 $\mu_i(x_k)$ 是任意控制策略,定义值函数如式(10)所示. 由引理 1 知,$V_i(x_k)\leqslant\Lambda_i(x_k)$. 下面预证 $\Lambda_i(x_k)\leqslant V_{i+1}(x_k)$. 由于 $\mu_i(x_k)$ 的任意性,不妨设 $\mu_i(x_k)=u_{i+1}(x_k)$,则

$$\Lambda_{i+1}(x_k)=X_k^{\mathrm{T}}QX_k+W(u_{i+1}(x_k))+\gamma V_i(F(x_k,x_{k-\tau},u_{i+1}(x_k))),$$

而

$$V_{i+1}(x_k)=X_k^{\mathrm{T}}QX_k+W(u_i(x_k))+\gamma V_i(F(x_k,x_{k-\tau},u_i(x_k))).$$

用数学归纳法证明 $\Lambda_i(x_k)\leqslant V_{i+1}(x_k)$:当 $i=0$时,$V_1(x_k)-\Lambda_0(x_k)=X_k^{\mathrm{T}}QX_k+W(u_0(x_k))\geqslant 0$,即 $\Lambda_0(x_k)\leqslant V_1(x_k)$.

假设当 $i=l-1$ 时,$\Lambda_{l-1}(x_k)\leqslant V_l(x_k)$,$\forall x_k\in\mathbf{R}^n$,则当 $i=l$ 时,$V_{l+1}(x_k)-\Lambda_l(x_k)=X_k^{\mathrm{T}}QX_k+W(u_l(x_k))+\gamma V_l(F(x_k,x_{k-\tau},u_l(x_k)))-(X_k^{\mathrm{T}}QX_k+W(u_l(x_k))+\gamma V_{l-1}(F(x_k,x_{k-\tau},u_l(x_k))))=\gamma(V_l(x_{k+1})-\Lambda_{l-1}(x_{k+1}))\geqslant 0$. 所以 $V_{l+1}(x_k)\geqslant\Lambda_l(x_k)$,即 $\Lambda_i(x_k)\leqslant V_{i+1}(x_k)$. 综上所述,$V_i(x_k)\leqslant\Lambda_i(x_k)\leqslant V_{i+1}(x_k)$,即 $\{V_i(x_k)\}$ 是单调递增序列.

由引理 2 及定理 1 知,$\{V_i(x_k)\}$ 是单调递增有界序列,故序列 $\{V_i(x_k)\}$ 收敛,记 $\lim_{i\to\infty}V_i(x_k)=V_\infty(x_k)$. 下面定理给出 $V_\infty(x_k)=V^*(x_k)$,即值函数序列 $\{V_i(x_k)\}$ 收敛到最优值函数 $V^*(x_k)$.

定理 2　设 $u_i(x_k)$ 和 $V_{i+1}(x_k)$ 定义如式(8)、式(9)所示,则 $\lim_{i\to\infty}V_i(x_k)=V^*(x_k)$.

证明　由定义 $V^*(x_k)=\inf_{\underline{u}_k^\infty}J(x_k,\underline{u}_k^\infty)$ 知,$V^*(x_k)\leqslant V_i(x_k)$,$i=1,2,\cdots$,当 $i\to\infty$ 时,$V^*(x_k)\leqslant V_\infty(x_k)$. 下面往证 $V_\infty(x_k)\leqslant V^*(x_k)$.

由 $V^*(x_k)$ 的定义知,对任意给定的 $\varepsilon>0$,总存在容许控制序列 $\underline{\zeta}_k^\infty(x_k)$,使得

$$V_i(x_k)\leqslant J(x_k,\underline{\zeta}_k^\infty(x_k))\leqslant V^*(x_k)+\varepsilon,\ i=1,2,\cdots$$

当 $i\to\infty$ 时,$V_\infty(x_k)\leqslant V^*(x_k)+\varepsilon$.

由于 ε 的任意性,所以 $\boldsymbol{V}_\infty(\boldsymbol{x}_k) \leqslant \boldsymbol{V}^*(\boldsymbol{x}_k)$.

综上所述,$\lim\limits_{i\to\infty}\boldsymbol{V}_i(\boldsymbol{x}_k) = \boldsymbol{V}^*(\boldsymbol{x}_k)$.

注 2 当 $i\to\infty$ 时,$\boldsymbol{V}_i(\boldsymbol{x}_k)\to\boldsymbol{V}^*(\boldsymbol{x}_k)$,由式(8)知,当 $i\to\infty$ 时,$\boldsymbol{u}_i(\boldsymbol{x}_k)\to\boldsymbol{u}^*(\boldsymbol{x}_k)$.

4 HDP 算法神经网络的实现

对于非线性系统,即使值函数是二次的,对应的最优控制也可能不是线性的,因此用神经网络来近似值函数序列 $\boldsymbol{V}_i(\boldsymbol{x}_k)$ 和控制序列 $\boldsymbol{u}_i(\boldsymbol{x}_k)$.

设隐含层神经元的个数为 l,输入层与隐含层的权重矩阵为 $\boldsymbol{\nu}$,隐含层与输出层的权重矩阵为 $\boldsymbol{\omega}$,则神经网络的输出 $\boldsymbol{y} = \boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\sigma}(\boldsymbol{\nu}^{\mathrm{T}}\boldsymbol{x})$. 其中 $\boldsymbol{\sigma}(\cdot)$ 为激活函数,常取 tansig(·),logsig(·);$\boldsymbol{\sigma}(\boldsymbol{\nu}^{\mathrm{T}}\boldsymbol{x}) \in \mathbf{R}^l$.

为了实现迭代 HDP 算法(8)和(9),本文采用 3 个前馈神经网络近似系统模型、值函数和控制策略,它们分别是模型网络、评判网络、控制网络,结构见图 1.

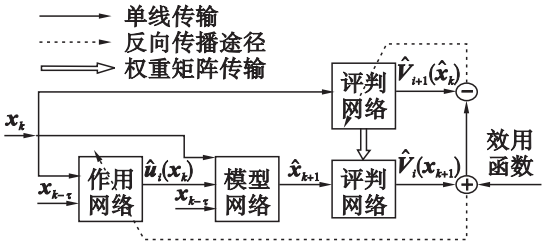

图 1 启发式动态规划结构图

Fig. 1 Structure diagram of HDP algorithm

在运行迭代 HDP 算法之前,首先对模型网络进行训练.

4.1 模型网络

状态估计值:$\hat{\boldsymbol{x}}_{k+1} = \boldsymbol{\omega}_{\mathrm{m}}^{\mathrm{T}}\boldsymbol{\sigma}(\boldsymbol{\nu}_{\mathrm{m}}^{\mathrm{T}}\boldsymbol{I}_{\mathrm{m}}(k))$,其中 $\boldsymbol{I}_{\mathrm{m}}(k) = (\boldsymbol{x}_k^{\mathrm{T}}, \boldsymbol{x}_{k-\tau}^{\mathrm{T}}, \hat{\boldsymbol{u}}_i^{\mathrm{T}}(\boldsymbol{x}_k))^{\mathrm{T}}$.

状态真值:$\boldsymbol{x}_{k+1} = \boldsymbol{F}(\boldsymbol{x}_k, \boldsymbol{x}_{k-\tau}, \hat{\boldsymbol{u}}_i(\boldsymbol{x}_k))$.

模型网络误差:$\boldsymbol{e}_{\mathrm{m}}(k) = \hat{\boldsymbol{x}}_{k+1} - \boldsymbol{x}_{k+1}$.

令 $\boldsymbol{E}_{\mathrm{m}}(k) = \frac{1}{2}\boldsymbol{e}_{\mathrm{m}}(k)^{\mathrm{T}}\boldsymbol{e}_{\mathrm{m}}(k)$,根据梯度下降法则,模型网络权重矩阵更新为

$$\boldsymbol{\omega}_{\mathrm{m}}(k+1) = \boldsymbol{\omega}_{\mathrm{m}}(k) - \alpha_{\mathrm{m}}\left(\frac{\partial \boldsymbol{E}_{\mathrm{m}}(k)}{\partial \boldsymbol{\omega}_{\mathrm{m}}(k)}\right),$$

$$\boldsymbol{\nu}_{\mathrm{m}}(k+1) = \boldsymbol{\nu}_{\mathrm{m}}(k) - \alpha_{\mathrm{m}}\left(\frac{\partial \boldsymbol{E}_{\mathrm{m}}(k)}{\partial \boldsymbol{\nu}_{\mathrm{m}}(k)}\right).$$

式中,α_{m} 表示模型网络的学习率.

模型网络训练后,权重矩阵保持不变.

4.2 评判网络

值函数估计值:$\boldsymbol{V}_i(\boldsymbol{x}_k) = \boldsymbol{\omega}_{\mathrm{c}}^{\mathrm{T}}\boldsymbol{\sigma}(\boldsymbol{\nu}_{\mathrm{c}}^{\mathrm{T}}\boldsymbol{x}_k)$.

值函数真值:$\boldsymbol{V}_i(\boldsymbol{x}_k) = \boldsymbol{X}_k^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{X}_k + \boldsymbol{W}(\hat{\boldsymbol{u}}_{i-1}(k)) + \gamma\hat{\boldsymbol{V}}_{i-1}(\hat{\boldsymbol{x}}_{k+1})$.

评判网络误差:$\boldsymbol{e}_{\mathrm{c}}(k) = \hat{\boldsymbol{V}}_i(\boldsymbol{x}_k) - \boldsymbol{V}_i(\boldsymbol{x}_k)$.

令 $\boldsymbol{E}_{\mathrm{c}}(k) = \frac{1}{2}\boldsymbol{e}_{\mathrm{c}}(k)^{\mathrm{T}}\boldsymbol{e}_{\mathrm{c}}(k)$,评判网络的目标是使 $\boldsymbol{E}_{\mathrm{c}}(k)$ 最小化,根据梯度下降法则,评判网络权重矩阵更新为

$$\boldsymbol{\omega}_{\mathrm{c}}(k+1) = \boldsymbol{\omega}_{\mathrm{c}}(k) - \alpha_{\mathrm{c}}\left(\frac{\partial \boldsymbol{E}_{\mathrm{c}}(k)}{\partial \boldsymbol{\omega}_{\mathrm{c}}(k)}\right),$$

$$\boldsymbol{\nu}_{\mathrm{c}}(k+1) = \boldsymbol{\nu}_{\mathrm{c}}(k) - \alpha_{\mathrm{c}}\left(\frac{\partial \boldsymbol{E}_{\mathrm{c}}(k)}{\partial \boldsymbol{\nu}_{\mathrm{c}}(k)}\right).$$

式中,α_{c} 表示评判网络的学习率.

4.3 控制作用网络

控制作用估计值:$\hat{\boldsymbol{u}}_i(\boldsymbol{x}_k) = \boldsymbol{\omega}_{\mathrm{a}}^{\mathrm{T}}\boldsymbol{\sigma}(\boldsymbol{\nu}_{\mathrm{a}}^{\mathrm{T}}\boldsymbol{I}_{\mathrm{a}}(k))$,式中,$\boldsymbol{I}_{\mathrm{a}}(k) = (\boldsymbol{x}_k^{\mathrm{T}}, \boldsymbol{x}_{k-\tau}^{\mathrm{T}})^{\mathrm{T}}$.

控制作用真值:$\boldsymbol{u}_i(\boldsymbol{x}_k) = \arg\min\limits_{\boldsymbol{u}_k}\{\boldsymbol{X}_k^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{X}_k + \boldsymbol{W}(\boldsymbol{u}_k) + \gamma\hat{\boldsymbol{V}}_i(\hat{\boldsymbol{x}}_{k+1})\}$.

控制作用网络误差:$\boldsymbol{e}_{\mathrm{a}}(k) = \hat{\boldsymbol{u}}_i(\boldsymbol{x}_k) - \boldsymbol{u}_i(\boldsymbol{x}_k)$.

令 $\boldsymbol{E}_{\mathrm{a}}(k) = \frac{1}{2}\boldsymbol{e}_{\mathrm{a}}(k)^{\mathrm{T}}\boldsymbol{e}_{\mathrm{a}}(k)$,控制作用网络的目标是使 $\boldsymbol{E}_{\mathrm{a}}(k)$ 最小化,根据梯度下降法则,控制作用网络权重矩阵更新为

$$\boldsymbol{\omega}_{\mathrm{a}}(k+1) = \boldsymbol{\omega}_{\mathrm{a}}(k) - \alpha_{\mathrm{a}}\left(\frac{\partial \boldsymbol{E}_{\mathrm{a}}(k)}{\partial \boldsymbol{\omega}_{\mathrm{a}}(k)}\right),$$

$$\boldsymbol{\nu}_{\mathrm{a}}(k+1) = \boldsymbol{\nu}_{\mathrm{a}}(k) - \alpha_{\mathrm{a}}\left(\frac{\partial \boldsymbol{E}_{\mathrm{a}}(k)}{\partial \boldsymbol{\nu}_{\mathrm{a}}(k)}\right).$$

式中,α_{a} 表示控制作用网络的学习率.

5 系统仿真

本文所讨论的非线性离散时滞系统是在文献[4]中加入时滞而得到,这里考虑 $\tau = 1$,

$$\boldsymbol{x}(k+1) = \begin{bmatrix} -0.8\boldsymbol{x}_2(k) - \boldsymbol{x}_1(k-1)\boldsymbol{u}_k \\ \sin(0.8\boldsymbol{x}_1(k) - \boldsymbol{x}_2(k)) + 1.8\boldsymbol{x}_2(k) + \\ 0.5\boldsymbol{x}_2(k-1) - \boldsymbol{x}_2(k)\boldsymbol{u}_k^2 \end{bmatrix}.$$

初始状态 $\boldsymbol{x}_k = [1, 1]^{\mathrm{T}}$,时滞状态 $\boldsymbol{x}_{k-1} = [0.5, 0.5]^{\mathrm{T}}$,$\boldsymbol{Q} = 0.1\boldsymbol{I}_4$,$\boldsymbol{R} = 0.1\boldsymbol{I}_2$,$\overline{\boldsymbol{U}} = [0.5]$.

选择三层前馈神经网络,模型网络、评判网络、控制网络的结构分别为 5-8-2,2-8-1,4-8-1,3 个网络初始权重矩阵中的元素来自[-0.5,0.5]中的随机数,学习率 $\alpha_{\mathrm{m}} = \alpha_{\mathrm{c}} = \alpha_{\mathrm{a}} = 0.1$,折扣因子 $\gamma = 1$. 首先训练模型网络,训练后固定模型网络的权重矩阵,然后训练控制网络和评判网络 200 个循环,并且每个网络各训练 1000 个循环,系统的状态曲线、值函数曲线、控制作用曲线如图 2 ~ 图 4 所示. 仿真结果证明了迭代 HDP 算法的收敛性.

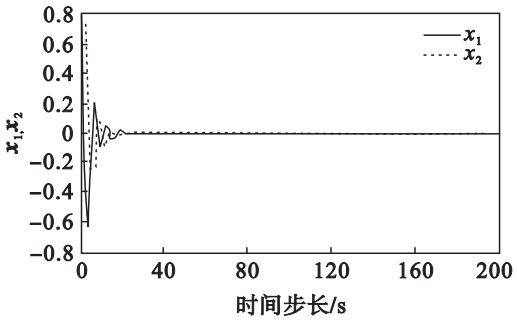

图 2　系统状态 x_1 和 x_2 曲线

Fig. 2　Trajectories of state variables (x_1, x_2)

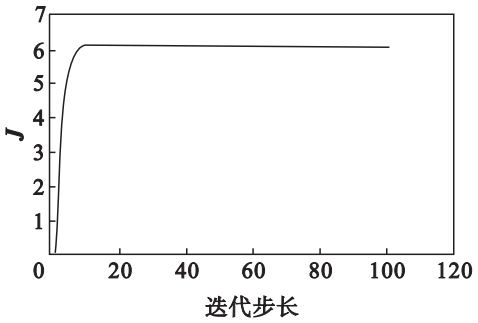

图 3　值函数 J 的曲线

Fig. 3　Trajectory of value function J

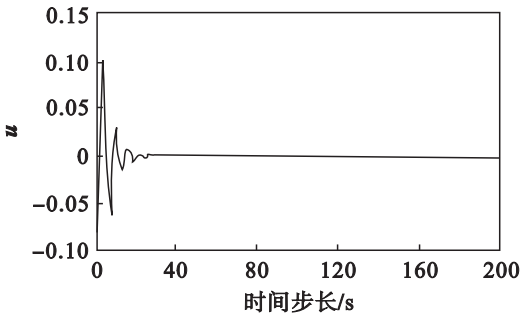

图 4　控制输入 u 曲线

Fig. 4　Trajectory of control u

6　结　　论

本文通过启发式动态规划算法给出了具有一般形式的带有饱和执行器的非线性离散时滞系统的最优控制策略，并且引入 3 个神经网络来近似系统模型、值函数、控制. 最后给出了一个仿真例子，说明了迭代算法的有效性.

参考文献：

[1] Lewis F L, Syrmos V L. Optiaml control [M]. 2nd ed. Hoboken: Wiley, 1995.

[2] Murray J J, Cox C J, Lendaris G G, *et al*. Adaptive dynamic programming [J]. *IEEE Transactions on Systems, Man and Cybernetics: C*, 2002, 32(1): 140 – 153.

[3] 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: B*, 2008, 38(4): 943 – 949.

[4] Werbos P J. Approximate dynamic programming for real-time control and neural modeling [M]. New York: Van Nostrand Reinhold, 1992.

[5] Sussmann H J, Sontag E D, Yang Y D. A general result on the stabilization of linear systems using bounded controls [J]. *IEEE Transactions on Automatic Control*, 1994, 39(12): 2411 – 2425.

[6] Saberi A, Lin Z L, Teel A R. Control of linear systems with saturating actuators [J]. *IEEE Transaction on Automatic Control*, 1996, 41(3): 368 – 378.

[7] Luo Y H, Zhang H G. Approximate optimal control for a class of nonlinear discrete-time systems with saturating actuators [J]. *Progress in Natural Science*, 2008, 18: 1023 – 1029.

[8] Wei Q L, Zhang H G, Liu D R, *et al*. An optimal 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] Song R Z, Zhang H G, Luo Y H, *et al*. Optimal control laws for time-delays systems with saturating actuators based on heuristic dynamic programming [J]. *Neurocomputing*, 2010, 73: 3020 – 3027.

[10] Wang F Y, Jing N, Liu D R, *et al*. Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε-error bound [J]. *IEEE Transaction on Neural Networks*, 2011, 22(1): 24 – 36.

[11] Liu D R, Wang D, Zhao D B, *et al*. Neural-network-based optimal control for a class of unknown discrete-time nonlinear systems using globalized dual heuristic programming [J]. *IEEE Transactions on Automation Science and Engineering*, 2012, 19(13): 628 – 634.

（上接第 460 页）

[7] 杜昭平，张庆灵，刘丽丽. 时变时延广义网络控制系统的稳定性分析 [J]. 东北大学学报：自然科学版，2011，32(8)：1065 – 1067.
（Du Zhao-ping, Zhang Qing-ling, Liu Li-li. Stability analysis for singular networked control systems with time-varying delays [J]. *Journal of Northeastern University: Natural Science*, 2011, 32(8): 1065 – 1067.）

[8] 邱占芝，张庆灵. 一类基于观测器的网络控制系统鲁棒控制器设计 [J]. 控制与决策，2007，22(10)：1165 – 1169.
（Qiu Zhan-zhi, Zhang Qing-ling. Robust controller design for a class of networked control systems based on state observer [J]. *Control and Decision*, 2007, 22(10): 1165 – 1169.）

[9] 邱占芝，张庆灵，杨春雨. 网络控制系统分析与控制 [M]. 北京：科学出版社，2009.
（Qiu Zhan-zhi, Zhang Qing-ling, Yang Chun-yu. Analysis and control of networked control systems [M]. Beijing: Science Press, 2009.）

[10] 俞立. 鲁棒控制线性矩阵不等式处理办法 [M]. 北京：清华大学出版社，2002.
（Yu Li. Robust control-linear matrix inequalities method [M]. Beijing: Tsinghua Unversity Press, 2002.）