文章编号: 1001-0920(2008)12-1359-04

基于强化学习的 JLQ 模型的直接自适应最优控制

徐琰恺, 陈 曦
(清华大学 a. 自动化系, b. 智能与网络化系统研究中心, 北京 100084)

摘 要: 研究离散时间跳变线性二次(JLQ)模型的直接自适应最优控制问题. 将强化学习的理论和方法应用于 JLQ 模型,设计基于 Q 函数的策略迭代算法,以优化系统性能. 在系统参数以及模态跳变概率未知的情况下,Q 函数对应的参数矩阵,可通过观察给定策略下系统行为,应用递归最小二乘算法在线估计. 基于此参数矩阵,可构造出新的策略使得系统性能更优. 该算法可收敛到最优策略.

关键词: Markov 跳变线性系统; 策略迭代; Q 函数; 直接自适应最优控制
中图分类号: TP13 **文献标识码**: A

Reinforcement learning-based direct adaptive optimal control of JLQ model

XU Yan-kai, CHEN Xi
(a. Department of Automation, b. Center for Intelligent and Networked Systems, Tsinghua University, Beijing 100084, China. Correspondent: XU Yan-kai, E-mail: xuyankai99@mails.thu.edu.cn)

Abstract: The discrete-time direct adaptive optimal control problem of jump linear quadratic (JLQ) model is investigated. Reinforcement learning theory and approaches are applied to JLQ model and Q function-based policy iteration algorithm is designed to optimize system performance. When the system parameters and jump probabilities of modes are unknown, the parameter matrix with respcet to Q function is online estimated by observing system behavior under a given control law with recursive least square algorithm. Moreover, based on this matrix, a new policy which can improve system performanc is constructed. The algorithm can converge to the optimal policy.

Key words: Markov jump linear system; Policy iteration; Q function; Direct adaptive optimal control

1 引 言

切换系统是近年来系统理论研究的热点之一[1],有着很强的工程应用背景. 其中 Markov 跳变线性系统(MJLS)是切换系统中研究较成熟的一类,它在柔性制造系统、电力系统、经济系统、容错系统设计、库存控制等具有模态跳变的实际系统中得到了广泛的应用[2,3]. 目前,国内外关于 MJLS 的研究成果很丰富[4-11].

本文考虑离散时间跳变线性二次(JLQ)模型的直接自适应最优控制问题. 在系统参数未知的情况下,传统的自适应最优控制方法是先进行参数辨识,然后进行最优控制. 与传统的方法不同,本文直接通过对系统行为的观察和学习得到最优的反馈控制律. 这种自适应控制方案称为直接自适应最优控制.

在近似动态规划(ADP)领域,将策略迭代和强化学习的理论与方法应用于控制系统,可得到直接自适应控制的算法[12-14]. 针对 JLQ 模型, Costa 等[15]研究了在模态跳变概率未知的情况下,基于 TD(λ)算法的自适应最优控制. 本文则基于 JLQ 模型,在系统参数以及模态跳变概率未知的情况下,通过观察给定策略下系统的行为(包括系统模态、状态、控制量以及代价),利用强化学习的方法进行直接自适应最优控制.

考虑离散无穷时间折扣代价准则模型,首先定义 JLQ 模型的 Q 函数,基于该函数设计策略迭代算法,以求得最优控制律. 在一个镇定的控制律下,Q 函数对应的参数矩阵可通过观察系统行为来估计,完成策略迭代中的策略评价. 利用估计得到的 Q 函

收稿日期: 2007-10-19; **修回日期**: 2008-02-28.
基金项目: 国家自然科学基金项目(60574064, 60736027).
作者简介: 徐琰恺(1981—),男,江苏常州人,博士生,从事混沌系统、Markov 决策过程等研究; 陈曦(1965—),女,成都人,副研究员,从事随机最优控制、传感器网络等研究.

数实施策略改进,以得到更好的反馈控制律. 在评价、改进的过程中,不需辨识系统参数以及模态跳变概率. 在系统可控及输入信号持续激励的条件下,可证明该方法收敛到最优的反馈控制律.

2 问题描述

Markov 跳变系统是一类具有离散与连续两种动态的混杂系统:一种称为模态,由有限离散状态的 Markov 链描述;另一种称为状态,由每一模态下的状态空间方程描述. 考虑如下离散时间线性系统:

$$x_{t+1} = A(\theta_t)x_t + B(\theta_t)u_t. \tag{1}$$

其中:t 为离散时刻,$x_t \in R^n$ 为状态,$\theta_t \in S = \{1, \cdots, S\}$ 为模态,$u_t \in R^m$ 为控制变量,$A(\theta_t)$ 和 $B(\theta_t)$ 为依赖于模态的合适维数的矩阵. 模态 θ_t 依 Markov 链变化,其转移概率矩阵为 $P = \{p_{ij}\}_{i,j\in S}$. 假设该 Markov 链是遍历的,稳态概率为 $\pi = [\pi_1, \cdots, \pi_S]$. 待优化的性能指标为

$$V(\theta_0, x_0) = \lim_{T\to\infty} \mathrm{E}\Big\{\sum_{t}^{T-1}[\gamma^t f^u(\theta_t, x_t)] \mid \theta_0, x_0\Big\}. \tag{2}$$

其中:代价函数 $f^u(\theta_t, x_t) = x_t^{\mathrm{T}} M(\theta_t)x_t + u_t^{\mathrm{T}} N(\theta_t)u_t$,$0 \le \gamma < 1$ 是折扣因子. 为表述方便,当 $\theta_t = i$ 时,矩阵 $A(\theta_t)$,$B(\theta_t)$,$M(\theta_t)$ 和 $N(\theta_t)$ 可简记作 A_i,B_i,M_i 和 N_i. 假设矩阵 M_i 和 N_i 是半正定矩阵,并假设系统(1) 是可镇定的,则对于稳定系统,性能指标(2) 存在.

任意给定一个镇定的反馈律 $u(i,x) = -L_i x$. 为简单起见,称 $L_i (i \in S)$ 为策略. 该策略下的系统性能也称作值函数,为如下二次型[5]:

$$V(i,x) = x^{\mathrm{T}} K_i x, \tag{3}$$

其中对称正定阵 $K_i (i \in S)$ 称作策略 L_i 下的代价矩阵. 定义 $F_i = \sum_j p_{ij} K_j$. 对于最优策略,有:

引理 1[5] JLQ 模型的最优反馈控制律为

$$u^*(i,x) = -L_i^* x,$$
$$L_i^* = \gamma[N_i + \gamma B_i^{\mathrm{T}} F_i^* B_i]^{-1} B_i^{\mathrm{T}} F_i^* A_i.$$

其中:$F_i^* = \sum_j p_{ij} K_j^*$,$K_i^*$ 为对称正定矩阵,是代数耦合的 Riccati 方程的解,即

$$K_i^* = M_i + \gamma A_i^{\mathrm{T}} F_i^* A_i - \gamma A_i^{\mathrm{T}} F_i^* B_i L_i^*.$$

此时系统最优性能为 $V^*(i,x) = x^{\mathrm{T}} K_i^* x$.

3 Q 函数与策略迭代

对于策略 L_i,定义 Q 函数为

$$Q(i,x,u) = f^u(i,x) + \gamma \mathrm{E}\{V(\theta_1, x_1) \mid \theta_0 = i, x_0 = x, u_0 = u\}. \tag{4}$$

其中:$x_1 = A_i x + B_i u$,即在 $t = 0$ 以及给定 0 时刻模态、状态及控制量时,由式(1) 得到;θ_1 为随机变量,由转移概率矩阵 P 决定. Q 函数也称作行动值函数,表示的是当前状态为 x 采取控制量 u,在此之后系统都采用策略 L_i 情况下的总折扣代价. Q 函数是对所有 $(i,x) \in S \times R^n$ 和所有 $u \in R^m$ 定义的. 对比值函数 $V(i,x)$ 的定义,可知

$$V(i,x) = Q(i,x,-L_i x). \tag{5}$$

结合式(4) 和(5),Q 函数的定义等价于

$$Q(i,x,u) = f^u(i,x) + \gamma \mathrm{E}\{Q(\theta_1, x_1, -L_{\theta_1} x_1) \mid \theta_0 = i, x_0 = x, u_0 = u\}. \tag{6}$$

将式(3) 代入(4),Q 函数可改写为

$$Q(i,x,u) = [x^{\mathrm{T}}, u^{\mathrm{T}}]\begin{bmatrix} H_{i(11)} & H_{i(12)} \\ H_{i(21)} & H_{i(22)} \end{bmatrix}\begin{bmatrix} x \\ u \end{bmatrix} = [x^{\mathrm{T}}, u^{\mathrm{T}}] H_i \begin{bmatrix} x \\ u \end{bmatrix}. \tag{7}$$

其中:$H_i (i \in S)$ 为对称正定矩阵,是 Q 函数的参数矩阵;而

$$H_{i(11)} = M_i + A_i^{\mathrm{T}} F_i A_i,\ H_{i(12)} = \gamma A_i^{\mathrm{T}} F_i B_i,$$
$$H_{i(21)} = \gamma B_i^{\mathrm{T}} F_i A_i,\ H_{i(22)} = N_i + \gamma B_i^{\mathrm{T}} F_i B_i. \tag{8}$$

应用强化学习[16] 理论,可构造基于 Q 函数的策略迭代方法,以求解最优策略 L_i^*. 策略迭代分为两个步骤:策略评价和策略改进. 对于某个给定策略,计算或估计该策略对应的值函数或 Q 函数(即相应的代价矩阵 K_i 或参数矩阵 $H_i, i \in S$),称作策略评价;基于值函数或 Q 函数,计算得到新的更优的策略,称作策略改进. 这两步反复进行,从而实现策略优化.

3.1 策略评价 ——估计 Q 函数

给定策略 L_i,由式(7) 可知,如果能通过观察系统行为,估计得到 H_i 矩阵,那么 Q 函数容易求得. 因此,实际上是期望通过观察和估计得到参数矩阵 $H_i, i \in S$.

对于任意 n 维向量 y,定义 $n(n+1)/2$ 维向量 $\bar{y}$ 表示 y 的元素的所有二次基函数

$$\bar{y} = [y_1^2, y_1 y_2, \cdots, y_1 y_n, y_2^2, \cdots, y_2 y_n, \cdots, y_n^2]^{\mathrm{T}}.$$

对于任意 $n \times n$ 对称矩阵 C,第 i 行第 j 列元素为 c_{ij},定义 $n(n+1)/2$ 维向量

$$\tilde{C} = [c_{11}, 2c_{12}, 2c_{13}, \cdots, 2c_{1n}, c_{22}, 2c_{23}, \cdots, 2c_{2n}, \cdots, c_{nn}]^{\mathrm{T}}.$$

反之,如果已知 $\tilde{C}$,也容易得到对称矩阵 C. 那么,对于任意二次型的表达式,有 $y^{\mathrm{T}} C y = \bar{y}^{\mathrm{T}} \tilde{C}$. 记行向量 $\alpha_{xu} = \begin{bmatrix} x \\ u \end{bmatrix}^{\mathrm{T}}$,则 Q 函数可改写为

$$Q(i,x,u) = \alpha_{xu} \tilde{H}_i,\ \forall i \in S. \tag{9}$$

定义$(n+m)(n+m+1)S/2$维向量$\Theta=[\widetilde{H}_1^{\mathrm{T}},\cdots,\widetilde{H}_S^{\mathrm{T}}]^{\mathrm{T}}$,向量$\Theta$中包含所有矩阵$H_i(i\in S)$的参数值.如果得到$\Theta$,那么也就得到了所有的$Q$函数.用向量$\Theta$表示$Q$函数,式(9)可写为

$$Q(i,x,u)=\beta(i,x,u)\Theta.$$

其中行向量$\beta(i,x,u)=[0,\cdots,\alpha_{xu},\cdots,0]$可分成$S$块,每块有$(n+m)(n+m+1)/2$个元素,第$i$块为$\alpha_{xu}$,其余均为零.

进一步,式(6)可改写为

$$\begin{aligned}&f^{u_t}(\theta_t,x_t)=\\&Q(\theta_t,x_t,u_t)-\gamma\mathrm{E}\{Q(\theta_{t+1},x_{t+1},-L\theta_{t+1}x_{t+1})\}=\\&\mathrm{E}\{\beta(\theta_t,x_t,u_t)-\gamma\beta(\theta_{t+1},x_{t+1},-L\theta_{t+1}x_{t+1})\}\Theta.\end{aligned}\tag{10}$$

对于每一个时刻t,式(10)都成立.行向量$\beta(i,x,u)$仅与系统状态和控制量有关,$f^u(i,x)$也可以观察到.因此,对所有时刻$t=0$到$T-1$列出方程(10),组成如下数据集:

$$Y=\begin{bmatrix}f^{u_0}(\theta_0,x_0)\\\cdots\\f^{u_{T-1}}(\theta_{T-1},x_{T-1})\end{bmatrix}=$$

$$\mathrm{E}\begin{bmatrix}\beta(\theta_0,x_0,u_0)-\gamma\beta(\theta_1,x_1,-L\theta_1x_1)\\\cdots\\\beta(\theta_{T-1},x_{T-1},u_{T-1})-\gamma\beta(\theta_T,x_T,-L\theta_Tx_T)\end{bmatrix}\Theta=$$

$$\mathrm{E}\{X\Theta\}.$$

对于$Y=\mathrm{E}\{X\Theta\}$这样的表示形式,应用最小二乘法求解,可得到Θ的估计值$\hat{\Theta}=(X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}Y$.当$T$足够大时,信息足够多,估计值$\hat{\Theta}$便足以接近真实值.那么由$\hat{\Theta}$便可得到参数矩阵$H_i,i\in S$.

上述算法在数据集全部得到之后才进行估计.事实上可在系统运行的同时,边观察边估计.简记

$$f_t=f^{u_t}(\theta_t,x_t),$$

$$\beta_t=\beta(\theta_t,x_t,u_t)-\gamma\beta(\theta_{t+1},x_{t+1},-L\theta_{t+1}x_{t+1}).$$

设计递归最小二乘(RLS)算法如下:

$$\begin{aligned}&G(0)=G_0,\\&e(l)=f_t-\beta_t\hat{\Theta}(l-1),\\&\hat{\Theta}(l)=\hat{\Theta}(l-1)+\frac{G(l-1)\beta_t^{\mathrm{T}}e(l)}{1+\beta_tG(l-1)\beta_t^{\mathrm{T}}},\\&G(l)=G(l-1)-\frac{G(l-1)\beta_t^{\mathrm{T}}\beta_tG(l-1)}{1+\beta_tG(l-1)\beta_t^{\mathrm{T}}}.\end{aligned}\tag{11}$$

其中:$G_0=\delta I$,δ为足够大的正常数,I为单位阵;$\hat{\Theta}(l)$为参数向量Θ的第l个估计值;$\hat{\Theta}(0)$是使Q函数参数矩阵H_i正定的给定初值.

Goodwin等[17]证明了如果存在$\varepsilon_1\le\varepsilon_2$,以及某个正数$T_0$,使得对于所有$T\ge T_0$,$t\ge T$,$\beta_t$满足下面的持续激励条件:

$$\varepsilon_1 I\le\frac{1}{T}\sum_{\tau}^{T}\beta_{t-\tau}^{\mathrm{T}}\beta_{t-\tau}\le\varepsilon_2 I,\tag{12}$$

那么RLS算法(11)渐近收敛到真实值Θ.对于所考虑的确定性系统(1),系统状态会很快达到原点,持续激励条件无法满足.所以本文采用带有噪声的输入信号$u_t=-L\theta_tx_t+\xi_t$,以满足条件(12),其中ξ_t为方差有限的白噪声.在该输入信号作用下,所得到的数据序列显然满足时间序列分析的平稳性条件,因此上述RLS算法是适用的.

3.2　策略改进　——基于Q函数的改进公式

用k表示策略迭代步数,上标'(k)'表示策略迭代第k步的相关量.第k步的策略为$L_i^{(k)}$,相应的代价矩阵为$K_i^{(k)}$,Q函数的参数矩阵为$H_i^{(k)}$,$i\in S$.

基于Q函数的策略改进公式为

$$L_i^{(k+1)}x=\arg\min_{u\in R^m}\{Q(i,x,u)\}.\tag{13}$$

上式对u求导数,使其为零,可解出

$$L_i^{(k+1)}=\gamma[N_i+\gamma B_i^{\mathrm{T}}F_i^{(k)}B_i]^{-1}B_i^{\mathrm{T}}F_i^{(k)}A_i.$$

结合式(8),容易得到第$k+1$步策略

$$L_i^{(k+1)}=[H_{i(22)}^{(k)}]^{-1}H_{i(21)}^{(k)}.\tag{14}$$

第$k+1$步的策略$L_i^{(k+1)}$也一定是镇定的,因为在该策略下的性能要优于$L_i^{(k)}$.基于这个策略,可估计得到新的Q函数,因而策略迭代过程可以继续.如果系统参数都已知,Zhang等[18]证明了运用策略迭代求解JLQ问题可收敛到最优解.

需要指出的是,基于Q函数的策略评价算法(11)和策略改进公式(14)中都没有显式地出现系统参数和模态转移概率矩阵,它们被隐含在参数矩阵H_i中.因此,基于Q函数的策略迭代方法可实现直接自适应最优控制,而不需要辨识系统参数.

3.3　直接自适应策略迭代

如上所述,应用RLS算法(11)估计Q函数的参数矩阵以进行策略评价;利用式(14)实施策略改进以得到更优的策略.下面给出直接自适应策略迭代算法:

算法1　直接自适应策略迭代算法

Step1:给定初始策略$L_i^{(0)}$,$k=0$,$t=0$;给定时间长度T,小正数$\varepsilon>0$;给定初始参数$\hat{\Theta}(0)$;选定噪声分布$\xi_t\sim N(0,\sigma^2)$.

Step2:$l=0$,给定RLS算法初值δ.

Step2.1:在时刻t,选取$u_t=-L_{\theta_t}^{(k)}x_t+\xi_t$作为输入信号,以得到新的状态$x_{t+1}$.记录一步代价$f_t$,应用算法(11)更新$\hat{\Theta}(l)$;$t=t+1$,$l=l+1$;重复该

步骤直至 $l = T - 1$.

Step2.2: 令 $\Theta^{(k)} = \hat{\Theta}(T-1)$. 由 $\Theta^{(k)}$ 得到 $H_i^{(k)}$, 应用式(14) 进行策略改进,得到新的策略 $L_i^{(k+1)}$.

Step3: 如果 $\max_{i\in S} | L_i^{(k)} - L_i^{(k+1)} | < \varepsilon$,则算法结束;否则,初始化参数 $\hat{\Theta}(0) = \Theta^{(k)}$, $k = k+1$,重复 Step2.

定理 1 如果 JLQ 系统(1) 是可控的,初始策略 $L_i^{(0)}$ 是镇定的,向量 β 满足持续激励条件(12),那么存在时间长度 $T < \infty$,使得策略迭代产生的策略序列 $L_i^{(k)}$ 收敛到最优策略 L_i^*.

针对线性二次(LQ) 系统,文献[12] 给出了收敛性的证明. 对于 JLQ 系统,证明过程类似,此略.

如果系统参数慢变,那么算法 1 持续运行,它具有自适应的能力. 利用算法 1 可设计直接自适应最优控制器,随着参数的变化调整最优控制策略.

4 数值算例

两模态一维的系统, $S = \{1,2\}$. $A_1 = 1, B_1 = 1, M_1 = 1, N_1 = 1, A_2 = 1, B_2 = -1, M_2 = 2, N_2 = 2$. 模态间的转移概率矩阵为

$$P = \begin{bmatrix} 0.3 & 0.7 \\ 0.6 & 0.4 \end{bmatrix}, \quad \gamma = 0.9.$$

如果系统参数都确切已知,则通过理论计算可得到最优控制律为 $L_1^* = 0.70, L_2^* = -0.50$. 设计控制器时这些参数是未知的,系统行为通过仿真得到. 应用算法 1,选取 $T = 10\,000$, $\varepsilon = 0.01$,控制变量叠加噪声为正态分布, $\xi_t \sim N(0, 0.09)$. 初始参数为 $\hat{\Theta}(0) = [1,0,1,1,0,1]^{\mathrm{T}}$, 初始策略为 $L_1^{(0)} = 0.2, L_2^{(0)} = -1.4$. RLS 算法初值取 $\delta = 10$. 算法 1 经过 6 个策略迭代步骤结束,此时得到的策略为 $L_1 = 0.69, L_2 = -0.50$. 此时 Q 函数对应的参数矩阵为

$$H_1 = \begin{bmatrix} 3.43 & 2.16 \\ 2.16 & 3.13 \end{bmatrix}, \quad H_2 = \begin{bmatrix} 3.69 & -2.03 \\ -2.03 & 4.12 \end{bmatrix}.$$

应用策略迭代,线性反馈律不断改进的过程如图 1 所示.

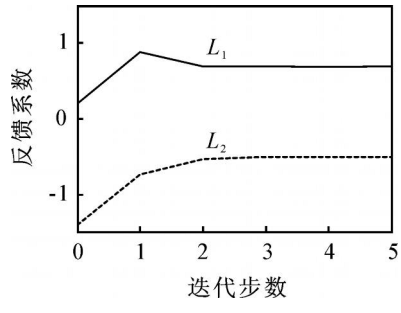

图 1 策略改进

5 结 论

本文考虑了 JLQ 模型的最优控制问题. 应用基于 Q 函数的策略迭代方法,可导出直接自适应最优控制算法. 对比传统的最优控制方法,该算法在系统参数未知的情况下,不需要辨识系统参数,而是直接估计 Q 函数对应的参数矩阵,并由该矩阵构造新的策略;同时,该算法不需要求解耦合的 Riccati 方程,避免了复杂的非线性方程组的求解问题;而且算法可以在线运行,边观察系统行为边进行优化. 以上优点使得本文提出的方法尤其适用于实际工程系统的控制和优化.

参考文献(References)

[1] 程代展, 郭宇骞. 切换系统进展 [J]. 控制理论与应用, 2005, 22(6): 954-960.
(Cheng D Z, Guo Y Q. Advances on switched systems [J]. Control Theory &Applications, 2005, 22(6): 954-960.)

[2] Abou-Kandil H, Smet O D, Freiling G, et al. Flow control in a failure-prone multi-machine manufacturing system [C]. Proc of INRIA/IEEE Symposium on Emerging Technologies and Factory Automation. Paris, 1995, 2: 575-583.

[3] Boukas E K, Shi P, Andijani A. Robust inventory-production control problem with stochastic demand[J]. Optimal Control Application and Methods, 1999, 20(11): 1-20.

[4] Ji Y, Chizeck H J. Controllability, stabilizability and continuous-time Markovian jump linear quadratic control [J]. IEEE Trans on Automatic Control, 1990, 35(7): 777-788.

[5] Costa O, Fragoso M D, Maroues R P. Discrete-time Markov jump linear systems [M]. London: Springer-Verlag, 2005.

[6] Xue F, Guo L. Necessary and sufficient conditions for adaptive stabilizability of jump linear systems [J]. Communications in Information and Systems, 2001, 1(2): 205-224.

[7] 张利军, 李春文, 程代展. 参数不确定马尔可夫跳变系统的鲁棒自适应控制[J]. 控制与决策, 2005, 20(9): 1030-1033.
(Zhang L J, Li C W, Cheng D Z. Robust adaptive control of Markov jump systems with parameter uncertainties[J]. Control and Decision, 2005, 20(9): 1030-1033.)

[8] 陈娇蓉, 刘飞. 采用输出反馈的执行器饱和跳变系统 H_∞控制[J]. 西安交通大学学报, 2007, 41(8): 934-938.
(Chen J R, Liu F. H_∞ control of jump systems with saturated actuator using output feedback[J]. J of Xi'an Jiaotong University, 2007, 41(8): 934-938.)

(下转第 1372 页)

变的振动控制.对于慢变子系统,基于鲁棒滑模微分估计器设计二阶滑模控制,使系统状态跟踪期望的轨迹,保留了滑模的鲁棒性和易于实现的特点,有效地去除了抖振和信号噪声.对于快变子系统,采用动态补偿器抑制输入信号的高频分量,设计最优控制规律,使柔性模态快速趋于稳定值.仿真结果表明,本文提出的混合控制方法,在保证刚性轨迹精确跟踪期望值的同时,弹性振动得到了有效抑制.

参考文献(References)

[1] Levant A. Universal single-input-single-output (SISO) sliding-mode controllers with finite-time convergence [J]. IEEE Trans on Automatic Control, 2001, 46(9): 1447-1451.

[2] Levant A. Quasi-continuous high-order sliding-mode controllers[C]. Proc of the 42nd IEEE Conf on Decision and Control Maui. Hawaii, 2003: 4605-4610.

[3] Levant A. Higher-order sliding modes, differentiation and output-feedback control[J]. Int J of Control, 2003, 76(9/10): 924-941.

[4] Xu J X, Lee T H, Pan Y J. On the sliding mode control for DC servo mechanisms in the presence of unmodeled dynamics [J]. Mechatronics, 2003, 13(7): 755-770.

[5] Bartolini G, Pisano A, Punta E, et al. A survey of applications of second-order sliding mode control to mechanical systems[J]. Int J of Control, 2003, 76(9): 875-892.

[6] Bruno Siciliano, Wayne J Book. A singular perturbation approach to control of lightweight flexible manipulators [J]. The Int J of Robotics Research, 1988, 7(4): 79-90.

[7] Feng Y, Bao S, Yu X. Inverse dynamics terminal sliding mode control of two-link flexible manipulators [C]. Proc of the 3rd Int DCDIS Conf on Engineering Applications and Computational Algorithms. Guecph, Ontario, 2003: 52-57.

[8] Hashtrudi-Zaad K, Khorasani K. Control of nonminimum phase singularly perturbed systems with applications to flexible link manipulators[J]. Int J of Control, 1996, 63(4): 679-701.

[9] Sanz A, Etxebarria V. Composite robust control of a laboratory flexible manipulator [C]. Decision and Control, 2005 European Control Conf. CDC-ECC '05. 44th IEEE Conf on Plazade Espana Seville, 2005: 3614-3619.

[10] Anderson B D O, Moore J B, Mingori D L. Ralations between frequency-dependent control and state weighting in LQG problems[J]. Proc of IEEE Conf on Decision and Control, 1983, 22: 612-617.

(上接第 1362 页)

[9] 刘飞, 张曦煌. L_2 增益约束下跳变系统鲁棒控制[J]. 控制理论与应用, 2006, 23(13): 1030-1037.
(Liu F, Zhang X H. Robust control for jump systems with L_2 gain constraints [J]. Control Theory & Applications, 2006, 23(13): 1030-1037.)

[10] 刘飞, 苏宏业, 褚健. 含参数不确定性的马尔可夫跳变过程鲁棒正实控制[J]. 自动化学报, 2003, 29(5): 761-766.
(Liu F, Su H Y, Chu J. Robust positive real control of Markov jump systems with parametric uncertainties [J]. Acta Automatica Sinica, 2003, 29(5): 761-766.)

[11] 徐琰恺, 陈曦. 模态跳变概率可控的 Markov 跳变线性系统的优化[J]. 控制与决策, 2008, 23(3): 246-250.
(Xu Y K, Chen X. Optimization of Markov jump linear system with controlled jump probabilities of modes[J]. Control and Decision, 2008, 23(3): 246-250.)

[12] Bradtke S J, Ydstie B E, Barto A G. Adaptive linear quadratic control using policy iteration [C]. Proc of American Control Conf. Maryland, 1994: 3475-3479.

[13] Hagen S, Krose B. Linear quadratic regulation using reinforcement learning[C]. Proc of Belgian-dutch Conf on Machine Learning. Wageningen, 1998: 39-46.

[14] Al-Tamimi A, Lewis F L, Abu-Khalaf M. Model-free Q-Learning designs for linear discrete-time zero-sum games with application to H-infinity control [J]. Automatica, 2007, 43(3): 473-481.

[15] Costa O, Aya J. Monte Carlo TD(λ) methods for the optimal control of discrete-time Markovian jump linear systems[J]. Automatica, 2002, 38(2): 217-225.

[16] Sutton R S, Barto A G. Reinforcement learning: An introduction [M]. Cambridge: The MIT Press, 1998.

[17] Goodwin G C, Sin K S. Adaptive filtering prediction and control[M]. New Jersey: Prentice-Hall, 1984.

[18] Zhang K J, Xu Y K, Chen X, et al. Policy iteration based feedback control[J]. Automatica, 2008, 44(4): 1055-1061.