

有限时间二次型最优调节器的数值解

叶彪明

(南京师范大学电气与工程学院, 210042, 南京)

[摘要] 通过对有限时间二次型最优调节器设计中矩阵 Riccati 微分方程的离散化, 将微分方程化为代数方程, 并将离散化后的代数方程通过变换使之成为矩阵 Riccati 代数方程的形式, 利用 MATLAB 控制系统工具箱中计算无限时间二次型最优调节器的 $\text{lqr}()$ 函数, 编程求解各离散时刻的矩阵 Riccati 代数方程, 从而得到矩阵 Riccati 微分方程的数值解以及二次型最优调节器最优控制的数值解.

[关键词] 线性二次型, Riccati 方程, MATLAB, 数值解

[中图分类号] O231, [文献标识码] A, [文章编号] 1672- 1292- (2003) 01- 0030- 04

0 引言

Riccati 方程的求解对于二次型控制问题来说是至关重要的, 人们研究和发展了关于它们的许多解法^[1~3]. 矩阵 Riccati 代数方程的求解有哈密顿矩阵的不变子空间解法、牛顿迭代法、广义特征子空间法、矩阵符号函数法等. 在 MATLAB 的控制系统工具箱中, 有相应的 $\text{lqr}()$ 函数, 可以对无限时间二次型最优调节器中的 Riccati 代数方程及最优控制进行数值求解^[4], 而对于矩阵 Riccati 微分方程, MATLAB 的控制系统工具箱中还缺乏相应的求解函数. 本文首先对矩阵 Riccati 微分方程进行离散化, 将矩阵 Riccati 微分方程化为代数方程, 并对此代数方程进行适当变换, 使之化为无限时间二次型最优调节器中的矩阵 Riccati 代数方程的形式, 利用已有的 $\text{lqr}()$ 函数, 设计出求解矩阵 Riccati 微分方程的数值解程序.

1 问题的提法

考虑线性可控系统

$$\dot{x}(t) = A(t)x(t) + B(t)u(t), \quad x(t_0) = x_0 \quad (1)$$

性能指标

$$J = \frac{1}{2}x^T(t_f)P_f x(t_f) + \frac{1}{2}\int_{t_0}^{t_f} [x^T(t)Q(t)x(t) + u^T(t)R(t)u(t)]dt \quad (2)$$

其中 $x(t) \in R^n$, $u(t) \in R^r$, 控制变量 $u(t)$ 不受约束, t_f 有限, $x(t_f)$ 自由. 对于 $t \in [t_0, t_f]$, 矩阵 $A(t)$ 、 $B(t)$ 、 $Q(t)$ 和 $R(t)$ 均连续、有界. $P_f \geq 0$, $P_f = P_f^T$; $Q \geq 0$, $Q = Q^T$; $R > 0$, $R = R^T$.

根据有约束泛函取极值的条件可知, 性能指标泛函取极小值

$$J^* = \frac{1}{2}x^T(t_0)P(t_0)x(t_0), \quad \forall x(t_0) \neq 0$$

的充要条件是

$$u^*(t) = -K(t)x(t) \quad (3)$$

$$K(t) = R^{-1}(t)B^T(t)P(t) \quad (4)$$

式(4)中 $P(t)$ 为如下矩阵 Riccati 微分方程的解

收稿日期: 2003- 02- 11.

基金项目: 江苏省教育厅重点课题(02KJA470001).

作者简介: 叶彪明, 1968- , 南京师范大学电气与工程学院讲师, 主要从事控制理论及应用的教学与研究.

$$\dot{P}(t) = -P(t)A(t) - A^T(t)P(t) + P(t)B(t)R^{-1}(t)B^T(t)P(t) - Q(t) \quad (5)$$

(5) 式中 $P(t_f) = P_f$, 从式(3)、(4)、(5)可以看出, 设计 t_f 有限时的最优调节器问题就变成求解矩阵 Riccati 微分方程的 $P(t)$ 问题, 矩阵 Riccati 微分方程包括 n^2 个一阶非线性时变微分方程, 可以证明, 只要矩阵 $P(t)$ 是对称矩阵, 只需求解 $n(n+1)/2$ 个一阶非线性微分方程, 由于变量之间互相耦合, 通常不能直接求得 $P(t)$ 的解析解, 需要用计算机进行离线计算, 得到 Riccati 微分方程的数值解。

2 有限时间二次型最优调节器的数值解

在稳态的情况下, 终止时间假定为 $t_f \rightarrow \infty$, 矩阵 Riccati 微分方程的解矩阵 $P(t)$ 将趋于常数矩阵, 使得 $\dot{P}(t) = 0$. 在这种情况下, 矩阵 Riccati 微分方程将简化为矩阵 Riccati 代数方程

$$P(t)A(t) + A^T(t)P(t) - P(t)B(t)R^{-1}(t)B^T(t)P(t) + Q(t) = 0 \quad (6)$$

在 MATLAB 的控制系统工具箱中提供了 `lqr()` 函数, 用来依照给定加权矩阵设计 $t_f \rightarrow \infty$ 的二次型最优控制器, 该函数的调用格式为

$$[K, P] = \text{lqr}(A, B, Q, R)$$

其中 (A, B) 为给定的对象状态方程模型, (Q, R) 分别为加权矩阵. 返回的向量 K 为最优控制的状态反馈向量, P 为矩阵 Riccati 代数方程的解, 该函数中使用了基于 Schur 算法的代数方程求解函数 `care()`, 具有较高的计算精度. 为用该函数对矩阵 Riccati 微分方程进行近似计算, 首先对矩阵 Riccati 微分方程进行离散化, 将 Riccati 微分方程转换成近似的 Riccati 代数方程, 令

$$\dot{P}(t) \approx \frac{P(t + \Delta t) - P(t)}{\Delta t} \quad (7)$$

式中 Δt 是采样周期, 可选择 $\Delta t = (t_f - t_0)/n$, 由式(5)和式(7)得:

$$P(t + \Delta t) = P(t) + \Delta t[-P(t)A(t) - A^T(t)P(t) + P(t)B(t)R^{-1}(t)B^T(t)P(t) - Q(t)] \quad (8)$$

为将式(8)化为矩阵 Riccati 代数方程(6)的形式, 进行如下变换

$$\begin{aligned} \frac{P(t + \Delta t) - P(t)}{\Delta t} &= \frac{P(t)}{\Delta t} + [-P(t)A(t)] - A^T(t)P(t) + P(t)B(t)R^{-1}(t)B^T(t)P(t) - Q(t) \\ &\quad - P(t)\left[A(t) - \frac{1}{2\Delta t}I\right] - \left[A(t) - \frac{1}{2\Delta t}I\right]^T P(t) + P(t)B(t)R^{-1}(t)B^T(t)P(t) \\ &\quad - \left[Q(t) + \frac{P(t + \Delta t) - P(t)}{\Delta t}\right] = 0 \\ &\quad - P(t)A_1(t) - A_1^T(t)P(t) + P(t)B(t)R^{-1}(t)B^T(t)P(t) - Q_1(t) = 0 \end{aligned} \quad (9)$$

$$\text{式中} \quad A_1(t) = A(t) - \frac{1}{2\Delta t}I, \quad Q_1(t) = Q(t) + \frac{P(t + \Delta t) - P(t)}{\Delta t}$$

显然, 这一变换并不改变系统的可控性. 通过上面的变换, 矩阵 Riccati 的微分方程转化为近似的 Riccati 代数方程, 对矩阵 Riccati 微分方程在每一个离散时刻的求解, 就可以通过函数 `lqr()` 求得, 已知 $P(t_f) = P_f$, 以此为初始条件, 从终端时刻的 $P(t_f)$ 出发, 以 $-\Delta t$ 为单位逆时针方向逐次求出各离散时刻的 $P(t)$ 值. 用 MATLAB 编写的函数文件为:

```
function [K1, P1] = lqr1(A, B, Q, R, Pf, t0, tf, n);
```

```
dt = (tf - t0) / n;
```

```
m = size(A);
```

```
m1 = size(B);
```

```
A1 = A - (1/(2*dt))*eye(m(1));
```

```
P = Pf;
```

```
for i = 1:n
```

```

Q1= Q+ P/ dt;
[K, P, E]= lqr(A1, B, Q1, R);
for il= 1:m(1)
    for jl= 1:m(1)
        P1(i, il, jl) = P(il, jl);
    end
    for j2= 1:m1(2)
        K1(i, j2, il) = K(j2, il);
    end
end
end
end

```

该函数的调用格式为

$$[K1, P1] = \text{lqr1}(A, B, Q, R, Pf, t0, tf, n)$$

式中, n 可以根据计算精度的要求选择合适的大小. 返回的向量 $K1$ 为各离散时刻最优控制的状态反馈向量, $P1$ 为离散时刻矩阵 Riccati 代数方程的解, 它们都是三维的.

3 仿真实验

例1 考虑下面给出的对象模型

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u,$$

选择的加权矩阵为

$$Pf = 0, Q = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, R = 1.$$

根据式(5), Riccati 微分方程及终端边界条件为:

$$\begin{aligned} p_{\dot{x}1} &= -1 + p_{12}^2 \\ p_{\dot{x}2} &= -p_{11} + p_{12}p_{22} \\ p_{\dot{x}2} &= -2p_{12} + p_{22}^2 \\ p_{11}(t_f) &= p_{12}(t_f) = p_{22}(t_f) = 0 \end{aligned}$$

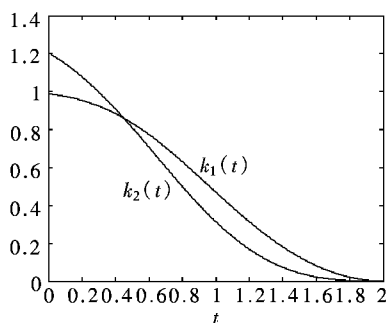
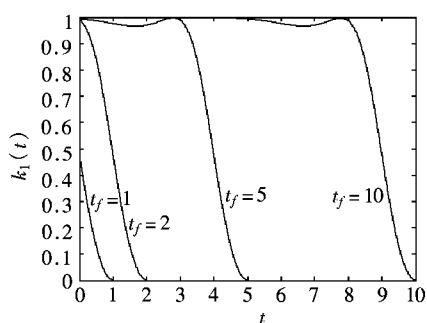
可以看出, 即使是二阶系统, 这组非线性微分方程的计算也很困难, 采用作者编写的数值计算程序进行计算, 可以得到从 $t = t_0 = 0$ 到 $t = t_f$ 的 $K(t)$ 和 $P(t)$ 值. 对于本系统 $K(t) = [k_1(t) \quad k_2(t)]$, 根据 $t_f = 2$ 时的计算结果绘制的 $k_1(t)$ 和 $k_2(t)$ 随时间变化的曲线如图1所示. 选择几组不同的 t_f , 计算 $k_1(t)$ 并将其绘制成随时间变化的曲线如图2所示.

由于 Riccati 微分方程是一组非线性微分方程, 一般只有一维问题才有解析解, 虽然本文提供了任意维 Riccati 微分方程的数值解, 但无解析解比较, 下面只能给出一个一维数例以与解析解比较.

例2 一阶系统状态方程 $\dot{x} = -0.5x + u$, 选择的加权矩阵为 $Pf = 10, Q = 2, R = 1$. 求得 $K(t)$ 的解析解为:

$$K(t) = \frac{1 + 1.5e^{3t-3}}{1 - 0.75e^{3t-3}}$$

任取 $t = 0.8$ 和 $t = 0.3$, 通过解析解计算得到的 $K(t)$ 为 $K(0.8) = 3.098648$ 和 $K(0.3) = 1.303391$, 而通过数值计算得到的 $K(0.8) = 3.098814$ 和 $K(0.3) = 1.303425$, 这说明本文的数值计算方法是有效的.

图1 $t_f = 2$ 时的 $k_1(t)$ 和 $k_2(t)$ 变化曲线图2 不同 t_f 时的 $k_1(t)$ 变化曲线

4 结论

从上面的举例可以看出, 利用函数 $\text{lqr1}()$ 可以方便地求解不同维数, 不同 t_f 时的矩阵 Riccati 微分方程, 从而得到二次型最优调节器最优控制的数值解, 在 n 合理的条件下, 其求解的精度由 MATLAB 中的 $\text{lqr}()$ 函数保证。

[参考文献]

- [1] 钟万勰, 钟翔翔. 线性二次最优控制的精细积分法[J]. 自动化学报, 2001, 27(2): 166~ 173.
- [2] 钟万勰. LQ 控制区段混合能矩阵的微分方程及其应用[J]. 自动化学报, 1992, 18(3): 325~ 331.
- [3] 姜长生, 吴庆宪, 孙隆和, 等. 系统理论与鲁棒控制[M]. 北京: 航空工业出版社, 1998.
- [4] 薛定宇. 反馈控制系统设计与分析——MATLAB 语言应用[M]. 北京: 清华大学出版社, 2000.

The Numerical Solution to Linear Quadratic Optimal Regulator within Finite Time

Ye Biaoming

(College of Electrical and Electronic Engineering, Nanjing Normal University, 210042, Nanjing, PRC)

Abstract: With the Riccati differential equation of linear quadratic optimal regulator within finite time discretized, the Riccati differential equation is changed into Riccati algebraic equation. By using the function of $\text{lqr}()$ in MATLAB control system toolbox to solve Riccati algebraic equation, the numerical solution to both Riccati differential equation and the linear quadratic optimal control can be obtained.

Key words: linear quadratic, Riccati equation, MATLAB, numerical solution

[责任编辑: 刘健]