第四章作业思路分享

主讲人 丁成科

内容

- 1. 最终状态v, a 是free情况的OBVP
- 2. ros 部分lattice+OBVP问题

第一题

Homework 1

- For the OBVP problem stated in slides p.25-p.29, please get the optimal solution (control, state, and time) for partially free final state case.
- Suppose the position is fixed, velocity and acceleration are free here.

1. 公式推导

深蓝学院
shenlanxueyuan.com

Define the Hamiltonian function:

$$H(s,u,\lambda) = \frac{1}{T} j^2 + \lambda^T f_s(s,u)$$

$$= \frac{1}{T} j^2 + \lambda_1 v + \lambda_2 a + \lambda_3 j$$

- 由Pontryagin's minimum principle:

$$\dot{\lambda}(t) = -\nabla_s H(s^*(t), u^*(t), \lambda(t))$$

$$\lambda(t) = \frac{1}{T}\begin{bmatrix} -2\alpha \\ 2\alpha t + 2\beta \\ -\alpha t^2 - 2\beta t - 2\gamma \end{bmatrix}$$

- 使用边界条件:

$$\lambda(T) = -\nabla h(s^*(T))$$

- 由于末状态v,a自由，因此h的表达式与T时刻的v和a并不相关

- 因此:

$$\lambda_2(T) = 0, \lambda_3(T) = 0$$

- 因此 表达式可以化简为:

$$\begin{cases} \beta = -\alpha T \\ \gamma = \dfrac{\alpha}{2} T^2 \end{cases} \qquad \lambda(t) = \frac{1}{T}\begin{bmatrix} -2\alpha \\ 2\alpha(t-T) \\ -\alpha t^2 + 2\alpha T t - \alpha T^2 \end{bmatrix}$$

1. 公式推导

The optimal input is solved as:

$$u^*(t) = j^*(t) = \arg\min_{j(t)} H(s^*(t), j(t), \lambda(t))$$

- 可以计得

$$\mathrm{u}^*(\mathrm{t}) = -\frac{\lambda_3 T}{2} = -\frac{T}{2}\frac{1}{T}(-\alpha t^2 + 2\alpha T t - \alpha \mathrm{T}^2) = \frac{1}{2}(\alpha t^2 - 2\alpha T t + \alpha \mathrm{T}^2)$$

- 积分可得

$$s^*(t) = \begin{bmatrix} \frac{\alpha}{120}t^5 - \frac{\alpha T}{24}t^4 + \frac{\alpha T^2}{12}t^3 + \frac{a_0}{2}t^2 + v_0 t + p_0 \\ \frac{\alpha}{24}t^4 - \frac{\alpha T}{6}t^3 + \frac{\alpha T^2}{4}t^2 + a_0 t + v_0 \\ \frac{\alpha}{6}t^3 - \frac{\alpha T}{2}t^2 + \frac{1}{2}\alpha T^2 t + a_0 \end{bmatrix}$$

1. 公式推导

深蓝学院 shenlanxueyuan.com

- 在知道s*(t)后，由于末状态的位置是已知的(pf)，代入求α

$$\frac{\alpha}{120}T^5-\frac{\alpha T^5}{24}+\frac{\alpha T^5}{12}+\frac{a_0T^2}{2}+v_0T+p_0 = p_f$$

- 解得

$$\alpha=\frac{20\Delta p}{T^5}$$ 其中 $$\Delta p=p_f-p_0-v_0T-\frac{1}{2}a_0T^2$$

- 把α代入代入先前的u*(t)表达式，再积分求出J

$$u^*(t) = -\frac{\lambda_3 T}{2} = -\frac{T}{2}\frac{1}{T}(-\alpha t^2+2\alpha Tt-\alpha T^2) = \frac{1}{2}(\alpha t^2-2\alpha Tt+\alpha T^2)$$

$$J=\frac{1}{T}\int_0^T(u^*(t))^2dt = \frac{20(\Delta p)^2}{T^6}$$ 其中 $$\Delta p=p_f-p_0-v_0T-\frac{1}{2}a_0T^2$$

1. 公式推导

- 可以看出J是只关于T的函数
- 为了令J最小，对T求导，让求导表达式=0，

$$(p_f - p_0 - v_0 T - \frac{1}{2} a_0 T^2)(a_0 T^2 + 4v_0 \mathrm{T} - 6p_f + 6p_0) = 0$$

可得

$$T^* = \frac{-v_0 \pm \sqrt{v_0^2 + 2a_0(p_f - p_0)}}{a_0} \text{或} T^* = \frac{-2v_0 \pm \sqrt{4v_0^2 + 6a_0(p_f - p_0)}}{a_0}$$

实际使用时，T不能为虚数和负数

比较剩下的T(正数)其cost的大小，令cost最小的T即为T*

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7299672

2(1) forward integration

深蓝学院
shenlanxueyuan.com

- 前向积分的公式是我们中学学到的恒加速度的运动公式:

$$v_1 = v_0 + a_0 t$$
$$p_1 = p_0 + v_0 t + 0.5 a_0 t^2$$

```
pos(0) = pos(0) + vel(0)*delta_time + 0.5*acc_input(0)*delta_time*delta_time;
pos(1) = pos(1) + vel(1)*delta_time + 0.5*acc_input(1)*delta_time*delta_time;
pos(2) = pos(2) + vel(2)*delta_time + 0.5*acc_input(2)*delta_time*delta_time;

vel(0) = vel(0) + acc_input(0)*delta_time;
vel(1) = vel(1) + acc_input(1)*delta_time;
vel(2) = vel(2) + acc_input(2)*delta_time;
```

2(2) find T*

- 本题的cost function

$$J = T + \left(\frac{1}{3}{\alpha_1}^2T^3 + \alpha_1\beta_1T^2 + {\beta_1}^2T\right) + \left(\frac{1}{3}{\alpha_2}^2T^3 + \alpha_2\beta_2T^2 + {\beta_2}^2T\right) + \left(\frac{1}{3}{\alpha_3}^2T^3 + \alpha_3\beta_3T^2 + {\beta_3}^2T\right)$$

- 其中α1 α2 α3 β1 β2 β3如下所示:

$$\begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix} = \begin{pmatrix} -\frac{12}{T^3} & 0 & 0 & \frac{6}{T^2} & 0 & 0 \\ 0 & -\frac{12}{T^3} & 0 & 0 & \frac{6}{T^2} & 0 \\ 0 & 0 & -\frac{12}{T^3} & 0 & 0 & \frac{6}{T^2} \\ \frac{6}{T^2} & 0 & 0 & -\frac{2}{T} & 0 & 0 \\ 0 & \frac{6}{T^2} & 0 & 0 & -\frac{2}{T} & 0 \\ 0 & 0 & \frac{6}{T^2} & 0 & 0 & -\frac{2}{T} \end{pmatrix} \begin{pmatrix} \Delta p_x \\ \Delta p_y \\ \Delta p_z \\ \Delta v_x \\ \Delta v_y \\ \Delta v_z \end{pmatrix}$$

2(2) find T*

深蓝学院 shenlanxueyuan.com

- 把α1 α2 α3 β1 β2 β3代入J后，J是只与T有关的函数
- 为了求得最优T，J对T求导，使得到的表达式=0(做法和第一题基本一致)
- 求导后让分子为0求极值。其中分子为关于T的四次多项式

$$\frac{d(J)}{\mathrm{d(T)}} = \frac{\mathrm{T}^4 - 4(\Delta_{\mathrm{vvx}} + \Delta_{\mathrm{vvy}} + \Delta_{\mathrm{vvz}})\mathrm{T}^2 + 24(\Delta_{\mathrm{x}}\Delta_{\mathrm{vx}} + \Delta_{\mathrm{y}}\Delta_{\mathrm{vy}} + \Delta_{\mathrm{z}}\Delta_{\mathrm{vz}})\mathrm{T} - 36({\Delta_{\mathrm{x}}}^2 + {\Delta_{\mathrm{y}}}^2 + {\Delta_{\mathrm{z}}}^2)}{\mathrm{T}^4}$$

$$\begin{cases} \Delta_{\mathrm{vvi}} = {V_{\mathrm{Ti}}}^2 + {V_{0\mathrm{i}}}^2 + V_{\mathrm{Ti}}V_{0\mathrm{i}} \\ \Delta_{\mathrm{vi}} = V_{\mathrm{Ti}} + V_{0\mathrm{i}} \\ \Delta_{\mathrm{i}} = \mathrm{P_{Ti}} - \mathrm{P_{0i}} \end{cases}$$

- 把得到四个根T进行筛选，不要虚数和负数
- 把剩下的T代入cost表达式中，比较得出最小J值，其对应的T即为最优解。

2(2) 多项式求根的方法

- 1. 求根公式法
- 一元四次多项式的求解问题存在求根公式(费拉里法求根)
- 注意根的取舍，舍去负数和虚根
- 但解的结构十分复杂，不太建议使用

2（2） 多项式求根的方法

- 2. 伴随矩阵求特征值。定义多项式伴随矩阵C(p)

In linear algebra, the Frobenius **companion matrix** of the monic polynomial

$$p(t) = c_0 + c_1 t + \cdots + c_{n-1} t^{n-1} + t^n \ ,$$

is the square matrix defined as

$$C(p) = \begin{bmatrix} 0 & 0 & \dots & 0 & -c_0 \\ 1 & 0 & \dots & 0 & -c_1 \\ 0 & 1 & \dots & 0 & -c_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -c_{n-1} \end{bmatrix}.$$

- 对矩阵C(p)求特征值，把正根代入J表达式中比较，从而得出最优T
- Eigen库可以去定义矩阵和计算特征值

2（2） 多项式求根的方法

- 3. Eigen PolynomialSolver
- 使用Eigen内置的多项式求解器
- 十分简单，调用Eigen API即可

```
Eigen::PolynomialSolver<double, Eigen::Dynamic>::RootsType & r = solver.roots();
```

- 使用example见下连结，

http://www.ce.unipr.it/people/medici/eigen-poly.html

2(2) 多项式求根的方法

- 4. Root-Finder
- 连结如下所示:

  https://github.com/ZJU-FAST-Lab/Root-Finder

- 包含了多种多项式求解的方法
- 使用简单，只需要包含其hpp头文件和Eigen库
- 使用时直接调用RootFinder::solvePolynomial即可，返回根的std::set
- 详情可查看README (包括使用方法和安装方法)

深蓝学院
shenlanxueyuan.com

感谢各位聆听！

Thanks for Listening