深蓝学院
shenlanxueyuan.com

Motion Planning for Mobile Robots
第四章作业讲评

主讲人 谢晓佳

第一题

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.

State: $\mathbf{s}_k = [p_k, v_k, a_k]^T$ **Input:** $u_k = j_k$

System model: $\dot{\mathbf{s}} = f_s(\mathbf{s}, u) = [v, a, j]^T, \quad \mathbf{s}(0) = [p_0, v_0, a_0], \quad t \in [0, \mathrm{T}]$

Objective function: $J = h(\mathbf{s}(\mathrm{T})) + \int_0^{\mathrm{T}} g(\mathbf{s}(t), u(t))\, dt, \quad g(\mathbf{s}(t), u(t)) = \frac{1}{\mathrm{T}} j(t)^2$

第一题

根据题意，这里只给定最终时刻的位置约束 p_f，因此定义 $h(\mathbf{s}(\mathrm{T}))=0$。最终的代价函数为：

$$J=h(\mathbf{s}(\mathrm{T}))+\int_0^{\mathrm{T}} g(\mathbf{s}(t),u(t))\,dt=\int_0^{\mathrm{T}}\frac{1}{\mathrm{T}}j(t)^2dt \tag{1}$$

定义协态 $\boldsymbol{\lambda}=[\lambda_1,\lambda_2,\lambda_3]^T$，则哈密尔顿函数为：

$$H(\mathbf{s},u,\boldsymbol{\lambda})=g(\mathbf{s},u)+\boldsymbol{\lambda}^T f(\mathbf{s},u)=\frac{1}{\mathrm{T}}j^2+\lambda_1 v+\lambda_2 a+\lambda_3 j \tag{2}$$

应用极小值原理，有：

$$\dot{\boldsymbol{\lambda}}=-\nabla\mathbf{s}H(\mathbf{s}(*,u)*,\boldsymbol{\lambda})=[0,-\lambda_1,-\lambda_2]^T \tag{3}$$

- 边界条件：

$$\boxed{\boldsymbol{\lambda}_{2,3}(\mathrm{T})=\nabla_{\mathbf{s}_{2,3}}h(\mathbf{s}^*(\mathrm{T}))=[0,0]^T} \tag{4}$$

其中 $\boldsymbol{\lambda}_{2,3}$ 表示 $\boldsymbol{\lambda}$ 的第2和3个变量。联立公式 (3) 和 (4)，可以求得 $\boldsymbol{\lambda}(t)$ 的表达式：

$$\boldsymbol{\lambda}(t)=\frac{1}{\mathrm{T}}\begin{bmatrix}-2\alpha\\ 2\alpha(t-\mathrm{T})\\ -\alpha(t-\mathrm{T})^2\end{bmatrix} \tag{5}$$

For (partially)-free final state problem:

$$given \quad s_i(T),\ i\in I$$

We have boundary condition for other costate:

$$\boxed{\lambda_j(T)=\frac{\partial h(s^*(T))}{\partial s_j},\quad for\ j\neq i}$$

第一题

其中，v_f^*, a_f^* 表示最优的末状态。令 $\frac{\partial H(\mathbf{s}^*(t), j(t), \boldsymbol{\lambda}(t))}{\partial j(t)} = 0$，求解最优控制：

$$\begin{aligned} u^*(t) = j^*(t) &= \arg\min_{j(t)} H(\mathbf{s}^*(t), j(t), \boldsymbol{\lambda}(t)) = -\frac{\lambda_3 \mathrm{T}}{2} \\ &= \frac{1}{2}\alpha(t-\mathrm{T})^2 \end{aligned} \tag{6}$$

根据最优控制和初始状态，则最优状态为：

$$\mathbf{s}^*(t) = \begin{bmatrix} \frac{\alpha}{120}(t-\mathrm{T})^5 + \frac{(a_0+\frac{\alpha}{6}\mathrm{T}^3)}{2}t^2 + (v_0 - \frac{\alpha}{24}\mathrm{T}^4)t + (p_0 + \frac{\alpha}{120}\mathrm{T}^5) \\ \frac{\alpha}{24}(t-\mathrm{T})^4 + (a_0 + \frac{\alpha}{6}\mathrm{T}^3)t + (v_0 - \frac{\alpha}{24}\mathrm{T}^4) \\ \frac{\alpha}{6}(t-\mathrm{T})^3 + (a_0 + \frac{\alpha}{6}\mathrm{T}^3) \end{bmatrix} \tag{7}$$

- 由末状态的位置约束，有：

$$\boxed{p_f = \frac{1}{2}(a_0\mathrm{T}^2 + \frac{\alpha}{6}\mathrm{T}^5) + (v_0\mathrm{T} - \frac{\alpha}{24}\mathrm{T}^5) + (p_0 + \frac{\alpha}{120}\mathrm{T}^5)} \tag{8}$$

则可以求解 α：

$$\alpha = \frac{20\Delta p}{\mathrm{T}^5}, \quad \Delta p = p_f - p_0 - \frac{1}{2}a_0\mathrm{T}^2 - v_0\mathrm{T} \tag{9}$$

For (partially)-free final state problem:

$$\boxed{given \quad s_i(T),\ i \in I}$$

We have boundary condition for other costate:

$$\lambda_j(T) = \frac{\partial h(s^*(T))}{\partial s_j}, \quad for\ j \neq i$$

最终J化简为： $J = \int_0^{\mathrm{T}} \frac{1}{\mathrm{T}} j^*(t)^2 dt = \frac{20\Delta P^2}{\mathrm{T}^6}$

第一题

The costate is solved as:

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

α,β,γ Is solved as:

$$\begin{bmatrix}\frac{1}{120}T^5 & \frac{1}{24}T^4 & \frac{1}{6}T^3\\ \cancel{\frac{1}{24}T^4} & \cancel{\frac{1}{6}T^3} & \cancel{\frac{1}{2}T^2}\\ \cancel{\frac{1}{6}T^3} & \cancel{\frac{1}{2}T^2} & \cancel{T}\end{bmatrix}\begin{bmatrix}\alpha\\ \beta\\ \gamma\end{bmatrix}=\begin{bmatrix}\Delta p\\ \cancel{\Delta v}\\ \cancel{\Delta a}\end{bmatrix}$$

$$\begin{cases}\begin{bmatrix}2\alpha T+2\beta\\ -\alpha T^2-2\beta T-2\gamma\end{bmatrix}=\begin{bmatrix}0\\0\end{bmatrix}\\ \frac{\alpha}{120}T^5+\frac{\beta}{24}T^4+\frac{\gamma}{6}T^3=\Delta p\end{cases}$$

$$\begin{bmatrix}\alpha\\ \beta\\ \gamma\end{bmatrix}=\frac{1}{T^5}\begin{bmatrix}20\\ -20T\\ 10T^2\end{bmatrix}\Delta p$$

For (partially)-free final state problem:

$$\boxed{given\quad s_i(T),\ i\in I}$$

We have boundary condition for other costate:

$$\boxed{\lambda_j(T)=\frac{\partial h(s^*(T))}{\partial s_j},\quad for\ j\neq i}$$

第二题

深蓝学院 shenlanxueyuan.com

Build an ego-graph of the linear modeled robot.
Select the best trajectory closest to the planning target.

$$J=\int_{\tau}^{\mathrm{T}}\left(1+a_x^2+a_y^2+a_z^2\right)dt$$

$$J(T)=T+12(\delta_{px}^2+\delta_{py}^2+\delta_{pz}^2)T^{-3}-12(\delta_{px}\delta_{vx1}+\delta_{py}\delta_{vx1}+\delta_{pz}\delta_{vx1})T^{-2}+4(\delta_{vx2}+\delta_{vy2}+\delta_{vz2})T^{-1}$$

其中，δ_i 定义如下：

$$\begin{cases}\delta_{pi}=p_{Ti}-p_{\tau i}\\ \delta_{vi1}=v_{Ti}+v_{\tau i}\\ \delta_{vi2}=v_{Ti}^2+v_{Ti}v_{\tau i}+v_{\tau i}^2\end{cases}$$

要求取目标函数 J 的极小值，仅需要对其求导，令其导数为 0 即可求得最优的时间 T。

$$\frac{\partial J(T)}{\partial T}=1-36(\delta_{px}^2+\delta_{py}^2+\delta_{pz}^2)T^{-4}+24(\delta_{px}\delta_{vx1}+\delta_{py}\delta_{vx1}+\delta_{pz}\delta_{vx1})T^{-3}-4(\delta_{vx2}+\delta_{vy2}+\delta_{vz2})T^{-2}$$

第二题

Roots of Polynomials

- 解析解(degree <= 4)
- 多项式伴随矩阵求根（Eigen手写/PolynomialSolver）

$$\frac{\partial J(T)}{\partial T} = 1 - 36(\delta_{px}^2 + \delta_{py}^2 + \delta_{pz}^2)T^{-4} + 24(\delta_{px}\delta_{vx1} + \delta_{py}\delta_{vx1} + \delta_{pz}\delta_{vx1})T^{-3} - 4(\delta_{vx2} + \delta_{vy2} + \delta_{vz2})T^{-2}$$

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

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

```
PolynomialSolver<double, Eigen::Dynamic> solver;
VectorXd coeff(5);
coeff[0] = -36.0f * param1;
coeff[1] =  24.0f * param2;
coeff[2] =  -4.0f * param3;
coeff[3] =   0.0f;
coeff[4] =   1.0f;
solver.compute(coeff);
const Eigen::PolynomialSolver<double, Eigen::Dynamic>::RootsType& root = solver.roots();
```

正根
J(T)最小

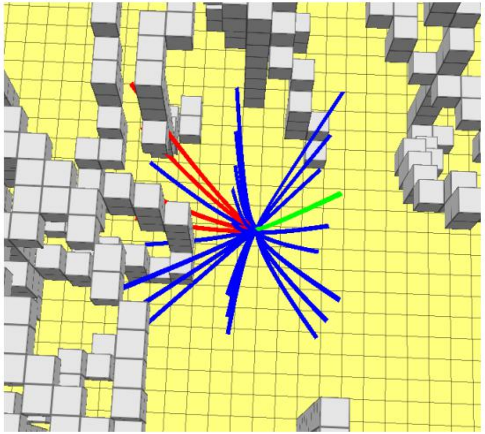
$$J(T) = T + 12(\delta_{px}^2 + \delta_{py}^2 + \delta_{pz}^2)T^{-3} - 12(\delta_{px}\delta_{vx1} + \delta_{py}\delta_{vx1} + \delta_{pz}\delta_{vx1})T^{-2} + 4(\delta_{vx2} + \delta_{vy2} + \delta_{vz2})T^{-1}$$

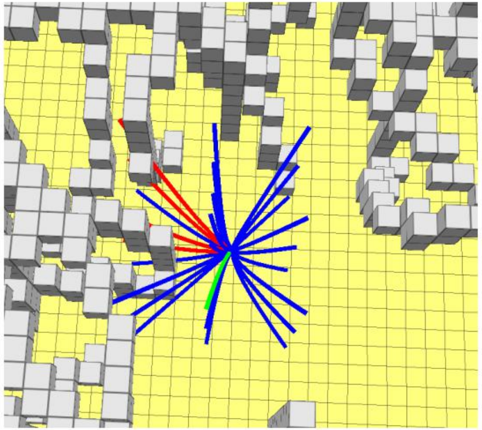
https://pdfs.semanticscholar.org/2b69/8d3602fe6ce5ec2b29f70614aa15e36fd397.pdf
http://web.mit.edu/18.06/www/Spring17/Eigenvalue-Polynomials.pdf

第二题

Roots of Polynomials

- Everyone learns the quadratic formula to find roots of a quadratic (degree-2) polynomial.
- There is a (horrible) cubic formula to find the roots of any cubic (degree-3) polynomial.
- There is a (terrifying) quartic formula to find the roots of any quartic (degree-4) polynomial.
- There is **no formula** (in terms of a *finite number* of $\pm, \times, \div, \sqrt[n]{}$) for the roots of an **arbitrary quintic** polynomial or **any degree** $\geq$ **5**. This is the Abel–Ruffini theorem, proved in the 19th century.





感谢各位聆听！

Thanks for Listening