深蓝学院
shenlanxueyuan.com

第五章作业思路提示

主讲人 陈铭楠

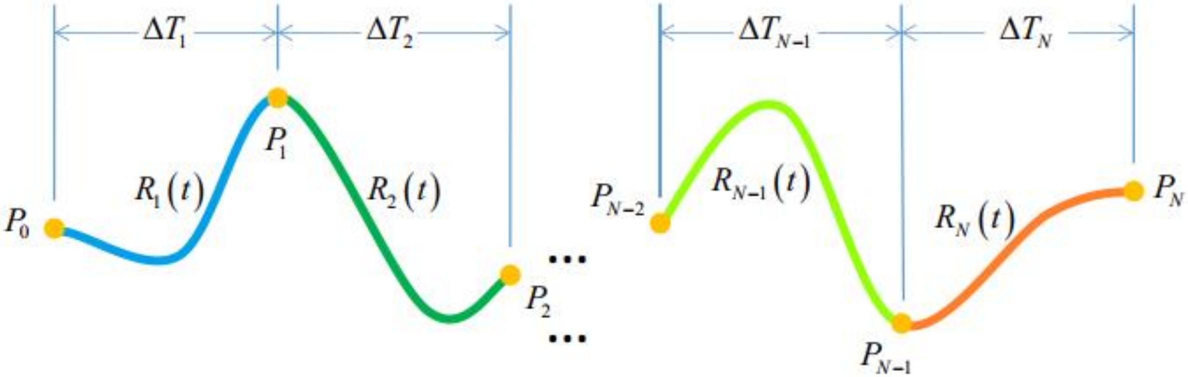
纲要

- 第一部分：作业解读
- 第二部分：思路提示

作业解读

Homework

3D Minimum Jerk Trajectory Generation



Input:

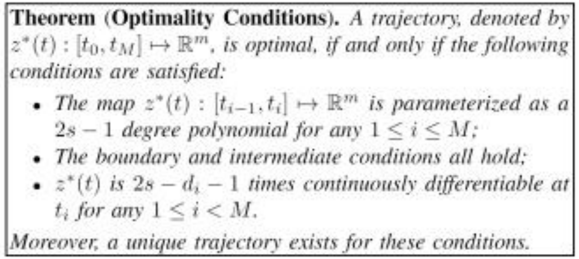
3-d pos, vel, and acc at start and terminal stamp;
(M-1) 3-d waypoint pos;
M durations for each trajectory pieces.

Output:

Coefficients of 3-d minimum jerk trajectory.

Requirement:

Use optimality conditions.



Theorem (Optimality Conditions). *A trajectory, denoted by $z^*(t) : [t_0, t_M] \mapsto \mathbb{R}^m$, is optimal, if and only if the following conditions are satisfied:*

- *The map $z^*(t) : [t_{i-1}, t_i] \mapsto \mathbb{R}^m$ is parameterized as a $2s-1$ degree polynomial for any $1 \leq i \leq M$;*
- *The boundary and intermediate conditions all hold;*
- *$z^*(t)$ is $2s - d_i - 1$ times continuously differentiable at t_i for any $1 \leq i < M$.*

Moreover, a unique trajectory exists for these conditions.

作业解读

本作业采用BIVP的方法，构建三维Minimum Jerk Trajectory Generation形式。Jerk为三阶导，故s=3，因此作业中构建的是5次（2s-1=5）多项式：

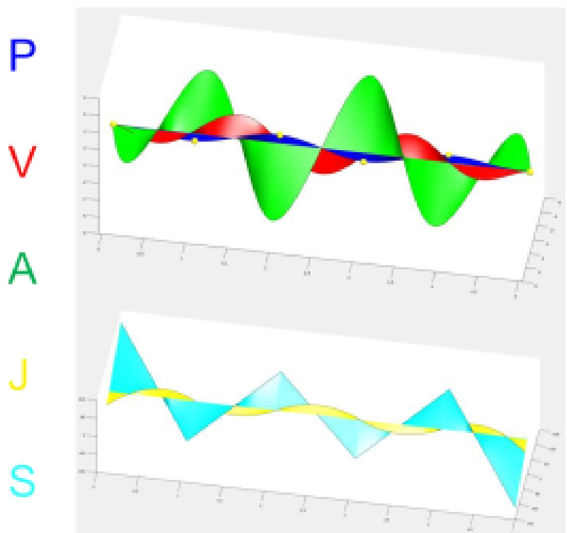
$$x(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4 + c_5 t^5$$

首先给定初末状态的p,v,a；以及中间状态，其中中间点的pvajs连续。我们构建最小 jerk 轨迹，把轨迹分成M段，每段可以用一个五阶的多项式来描述，通过最优条件理论，即每个位点左右两段轨迹的多阶导数的连续性以及位置的约束来求解轨迹的系数；

Boundary-intermediate value problem (BIVP)

$$
\begin{aligned}
\min_{z(t)} \int_{t_0}^{t_M} & v(t)^{\mathrm{T}} \mathbf{W} v(t) \mathrm{d}t, \\
s.\,t. \quad & z^{(s)}(t) = v(t),\ \forall t \in [t_0, t_M], \\
& z^{[s-1]}(t_0) = \bar{z}_o,\ z^{[s-1]}(t_M) = \bar{z}_f, \\
& z^{[d_i-1]}(t_i) = \bar{z}_i,\ 1 \le i < M, \\
& t_{i-1} < t_i,\ 1 \le i \le M.
\end{aligned}
$$

Minimum Jerk Trajectory



The solution of BIVP with **waypoint-only intermediate conditions** is

1. a spline with piece-wise 5 degree polynomials with continuous snap when s=3 (minimum jerk trajectory)
2. a spline with piece-wise 7 degree polynomials with continuous pop when s=4 (minimum snap trajectory)

$$\boxed{\mathbf{Mc} = \mathbf{b}}$$

Direct constructing the optimal solution is far easier and more efficient than implicit or explicit optimization. The result is same as one obtained through QP formulation(Mellinger ICRA 2011 and Bry IJRR 2015).

Mellinger et al., Minimum Snap Trajectory Generation and Control for Quadrotors, ICRA 2011.
Bry et al., Aggressive flight of fixed-wing and quadrotor aircraft in dense indoor environments, IJRR 2015.

作业解读

由上页课件PPT的BIVP原理可知，我们可以直接施加轨迹上关于连续的条件，然后构成关于系数的等式，线性方程组。然后直接解这个线性方程组。

BVIP其实就是根据最优性条件把问题变成Mc=b。M是关于c在t等于不同时刻的系数，c就是我们待求的多段多项式样条的系数矩阵。

因此，解这个问题的重点在于构建每一段的Mc=b方程并求解系数c，得出多项式方程。

我们只要把矩阵M和b写出来，就可以把c的系数解出来。

求出来之后我们可以任意给定终末pva和给一些waypoint来获得最优轨迹。

有些同学采用对M来求逆的方法来求解方程，也有同学直接采用Eigen库求解方程的方法求解，两种方法都可以。

本次作业完成所使用的参考文献：Geometrically Constrained Trajectory Optimization for Multicopters, (TRO, 2022)

思路提示 —基本分析

我们给定的点为三维的空间点，可以单独看成（x, y, z)三个方向，各自满足约束条件，每一段c的形式为：

以 x 方向上为例，其轨迹为：

$$c(i)=\begin{pmatrix} c_{0x} & c_{0y} & c_{0z} \\ c_{1x} & c_{1y} & c_{1z} \\ c_{2x} & c_{2y} & c_{2z} \\ c_{3x} & c_{3y} & c_{3z} \\ c_{4x} & c_{4y} & c_{4z} \\ c_{5x} & c_{5y} & c_{5z} \end{pmatrix}$$

$$\begin{aligned} x(t) &= c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4 + c_5 t^5 \\ \dot{x}(t) &= c_1 + 2c_2 t + 3c_3 t^2 + 4c_4 t^3 + 5c_5 t^4 \\ \ddot{x}(t) &= 2c_2 + 6c_3 t + 12c_4 t^2 + 20c_5 t^3 \\ x^{(3)}(t) &= 6c_3 + 24c_4 t + 60c_5 t^2 \\ x^{(4)}(t) &= 24c_4 + 120c_5 t \end{aligned}$$

我们的目的在于求解xyz三个维度的Mc=b方程，下面我们开始构建M矩阵

思路提示 —基本分析

由论文可知，M的组成为如下形式：

$$\mathbf{M}=\begin{pmatrix}\mathbf{F}_0 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{E}_1 & \mathbf{F}_1 & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{0} & \mathbf{E}_2 & \mathbf{F}_2 & \cdots & \mathbf{0}\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{F}_{M-1}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{E}_M\end{pmatrix}$$

$$\mathbf{M}\in\mathbb{R}^{2Ms\times 2Ms}$$

数字M为轨迹段数，2s（s=3）为每一维度多项式系数的个数。

矩阵b形式如下：

$$\mathbf{b}=(\mathbf{D}_0^{\mathrm{T}},\mathbf{D}_1^{\mathrm{T}},\mathbf{0}_{m\times\bar{d}_1},\ldots,\mathbf{D}_{M-1}^{\mathrm{T}},\mathbf{0}_{m\times\bar{d}_{M-1}},\mathbf{D}_M^{\mathrm{T}})^{\mathrm{T}}.$$

$$\mathbf{b}\in\mathbb{R}^{2Ms\times m}$$

$$\mathbf{D}_0,\mathbf{D}_M\in\mathbb{R}^{s\times m}\text{ and }\mathbf{D}_i\in\mathbb{R}^{d_i\times m}$$

其中s=3，m=3（表示坐标点为三维），
$d_i=1$（视频中说此值根据经验获得）
系数矩阵c的形式如下：

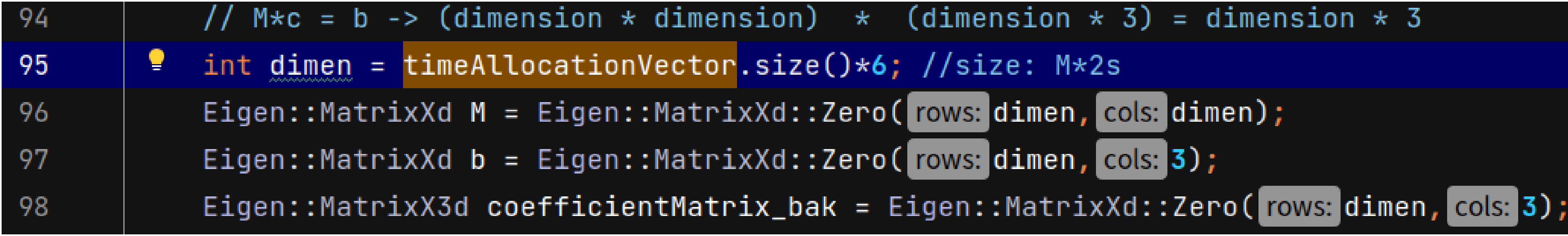
系数矩阵c的形式如下：

$$\mathbf{c}=\left(\mathbf{c}_1^{\mathrm{T}},\ldots,\mathbf{c}_M^{\mathrm{T}}\right)^{\mathrm{T}}$$

$$\mathbf{c}\in\mathbb{R}^{2Ms\times m}$$

思路提示 —基本分析

根据上面的基本分析，我们可以声明矩阵如下：



```
    // M*c = b -> (dimension * dimension)  *  (dimension * 3) = dimension * 3
    int dimen = timeAllocationVector.size()*6; //size: M*2s
    Eigen::MatrixXd M = Eigen::MatrixXd::Zero( rows: dimen, cols: dimen);
    Eigen::MatrixXd b = Eigen::MatrixXd::Zero( rows: dimen, cols: 3);
    Eigen::MatrixX3d coefficientMatrix_bak = Eigen::MatrixXd::Zero( rows: dimen, cols: 3);
```

思路提示 —构造矩阵

$$\mathbf{M} = \begin{pmatrix} \mathbf{F}_0 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{E}_1 & \mathbf{F}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_2 & \mathbf{F}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{F}_{M-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{E}_M \end{pmatrix}$$

$$\mathbf{b} = (\mathbf{D}_0^{\mathrm{T}}, \mathbf{D}_1^{\mathrm{T}}, \mathbf{0}_{m\times\bar{d}_1}, \ldots, \mathbf{D}_{M-1}^{\mathrm{T}}, \mathbf{0}_{m\times\bar{d}_{M-1}}, \mathbf{D}_M^{\mathrm{T}})^{\mathrm{T}}.$$

$$\mathbf{F}_0 = (\beta(0), \ldots, \beta^{(s-1)}(0))^{\mathrm{T}}$$ $F_0 \epsilon R^{s\times 2s}\ (s=3)$

$$\beta(x) = (1, x, \ldots, x^N)^{\mathrm{T}}$$ N=2s-1

首先，我们计算M矩阵中的F0

$$\beta(0) = (1,0,0,0,0,0)^T$$
$$\beta^1(0) = (0,1,0,0,0,0)^T$$
$$\beta^2(0) = (0,0,2,0,0,0)^T$$

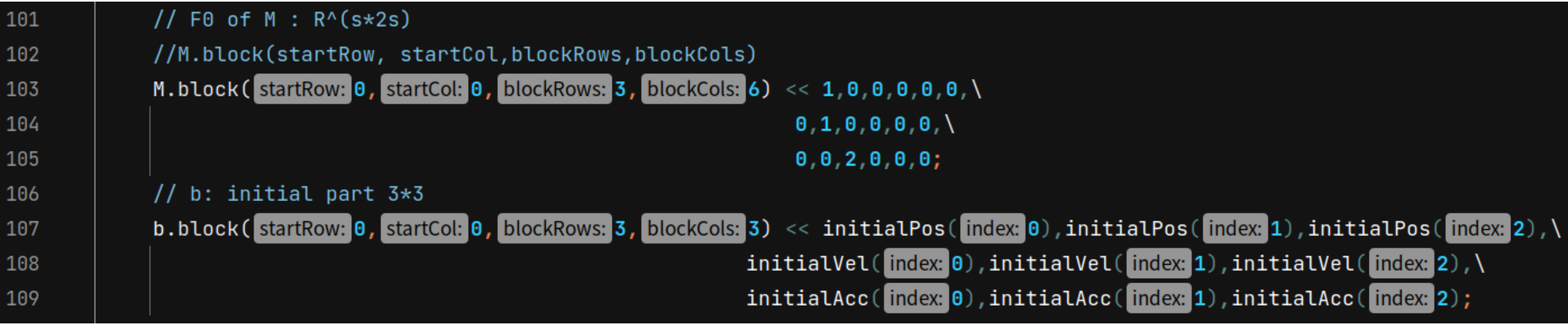
$$F_0 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \end{pmatrix}$$

D0为初始状态的pva：

$$D_0 = \begin{pmatrix} p_{x0} & p_{y0} & p_{z0} \\ v_{x0} & v_{y0} & v_{z0} \\ a_{x0} & a_{y0} & a_{z0} \end{pmatrix}$$

思路提示 —构造矩阵

我们将F_0和D_0填入M矩阵和b矩阵对应的位置，代码如下：



```
    // F0 of M : R^(s*2s)
    //M.block(startRow, startCol,blockRows,blockCols)
    M.block(startRow: 0, startCol: 0, blockRows: 3, blockCols: 6) << 1,0,0,0,0,0,\
                                                                  0,1,0,0,0,0,\
                                                                  0,0,2,0,0,0;
    // b: initial part 3*3
    b.block(startRow: 0, startCol: 0, blockRows: 3, blockCols: 3) << initialPos(index: 0),initialPos(index: 1),initialPos(index: 2),\
                                                              initialVel(index: 0),initialVel(index: 1),initialVel(index: 2),\
                                                              initialAcc(index: 0),initialAcc(index: 1),initialAcc(index: 2);
```

思路提示 —构造矩阵

接下来是构造矩阵元素Ei，Fi，Di：

由论文中可得

$$\mathbf{E}_i = \left(\beta(T_i), \ldots, \beta^{(d_i-1)}(T_i), \beta(T_i), \ldots, \beta^{(\bar{d}_i-1)}(T_i)\right)^{\mathrm{T}},$$

$$\mathbf{F}_i = \left(\mathbf{0}, -\beta(0), \ldots, -\beta^{(\bar{d}_i-1)}(0)\right)^{\mathrm{T}}$$

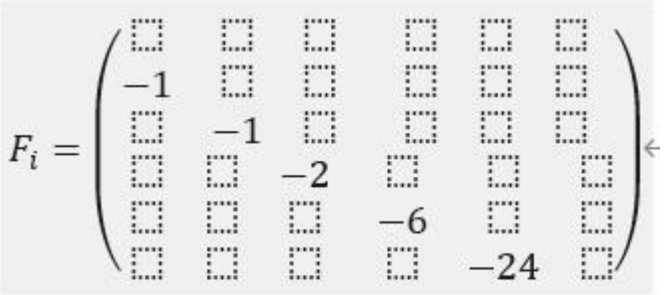
$\mathbf{E}_i, \mathbf{F}_i \in \mathbb{R}^{2s\times 2s}$; $\bar{d}_i = 2s - d_i$

(s=3,di=1)

$\mathbf{D}_i \in \mathbb{R}^{d_i\times m}$

$$D_i = \left(p_{xi}, p_{yi}, p_{zi}\right)$$

由此可得：



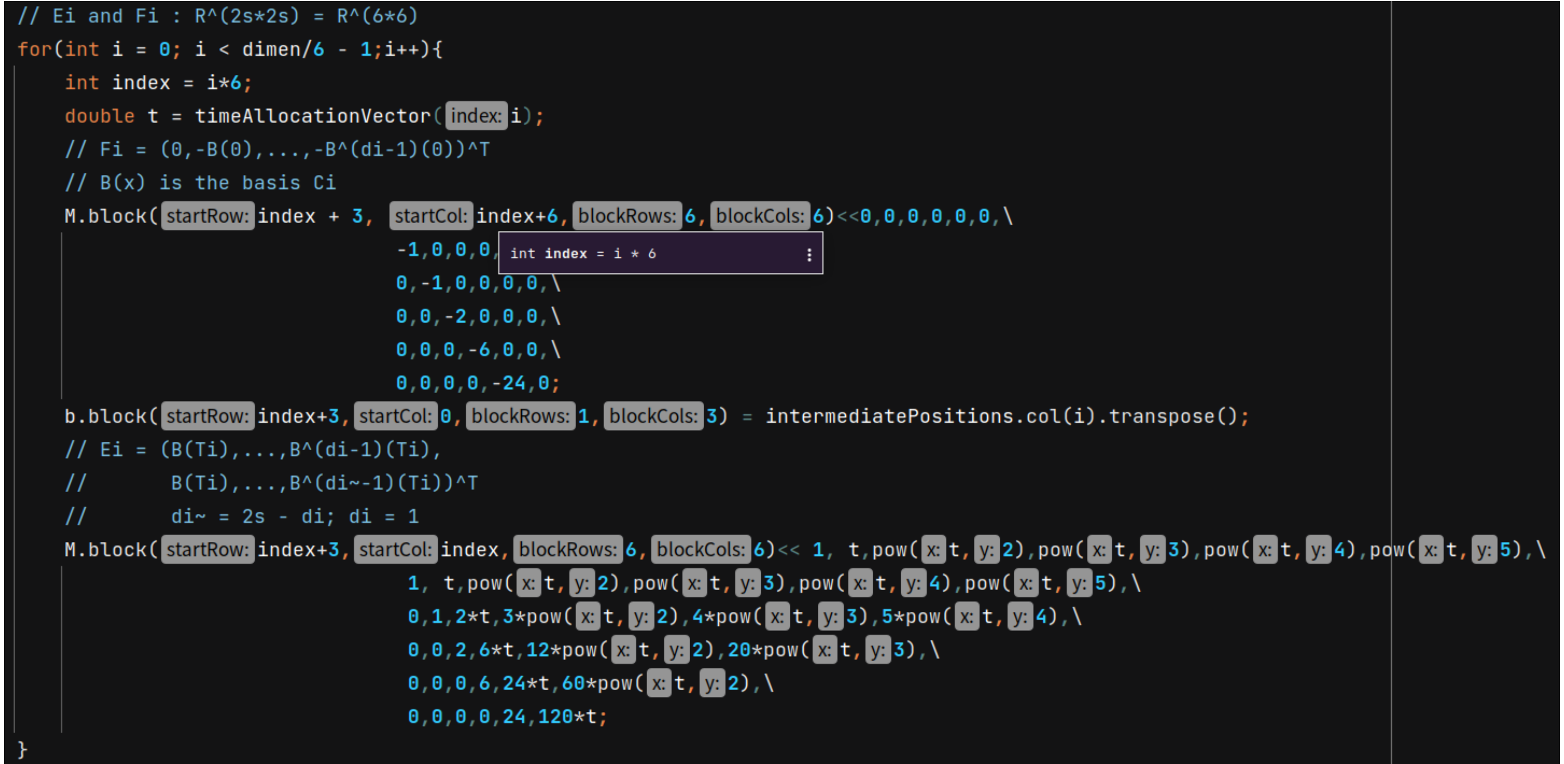
$$F_i = \begin{pmatrix} □ & □ & □ & □ & □ & □ \\ -1 & □ & □ & □ & □ & □ \\ □ & -1 & □ & □ & □ & □ \\ □ & □ & -2 & □ & □ & □ \\ □ & □ & □ & -6 & □ & □ \\ □ & □ & □ & □ & -24 & □ \end{pmatrix}$$

$$E_i = \left(\beta(T_i), \beta(T_i), \beta^1(T_i), \beta^2(T_i), \beta^3(T_i), \beta^4(T_i)\right)^T$$

$$= \begin{pmatrix} 1 & t & t^2 & t^3 & t^4 & t^5 \\ 1 & t & t^2 & t^3 & t^4 & t^5 \\ & 1 & 2t & 3t^2 & 4t^3 & 5t^4 \\ & & 2 & 6t & 12t^2 & 20t^3 \\ & & & 6 & 24t & 60t^2 \\ & & & & 24 & 120t \end{pmatrix}$$

思路提示 —构造矩阵

通过for循环的方式，填补到M矩阵和b矩阵对应的位置，代码如下：



```
// Ei and Fi : R^(2s*2s) = R^(6*6)
for(int i = 0; i < dimen/6 - 1;i++){
    int index = i*6;
    double t = timeAllocationVector( index: i);
    // Fi = (0,-B(0),...,-B^(di-1)(0))^T
    // B(x) is the basis Ci
    M.block( startRow: index + 3,  startCol: index+6, blockRows: 6, blockCols: 6)<<0,0,0,0,0,0,\
                                -1,0,0,0,
                                0,-1,0,0,0,0,\
                                0,0,-2,0,0,0,\
                                0,0,0,-6,0,0,\
                                0,0,0,0,-24,0;
    b.block( startRow: index+3, startCol: 0, blockRows: 1, blockCols: 3) = intermediatePositions.col(i).transpose();
    // Ei = (B(Ti),...,B^(di-1)(Ti),
    //       B(Ti),...,B^(di~-1)(Ti))^T
    //       di~ = 2s - di; di = 1
    M.block( startRow: index+3, startCol: index, blockRows: 6, blockCols: 6)<< 1,  t,pow( x: t, y: 2),pow( x: t, y: 3),pow( x: t, y: 4),pow( x: t, y: 5),\
                                 1,  t,pow( x: t, y: 2),pow( x: t, y: 3),pow( x: t, y: 4),pow( x: t, y: 5),\
                                 0,1,2*t,3*pow( x: t, y: 2),4*pow( x: t, y: 3),5*pow( x: t, y: 4),\
                                 0,0,2,6*t,12*pow( x: t, y: 2),20*pow( x: t, y: 3),\
                                 0,0,0,6,24*t,60*pow( x: t, y: 2),\
                                 0,0,0,0,24,120*t;
}
```

```
int index = i * 6
```

思路提示 —构造矩阵

构造Em和Dm：

由论文中可得

$$\mathbf{M} = \begin{pmatrix} \mathbf{F}_0 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{E}_1 & \mathbf{F}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_2 & \mathbf{F}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{F}_{M-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{E}_M \end{pmatrix}$$

$$\mathbf{b} = (\mathbf{D}_0^{\mathrm{T}}, \mathbf{D}_1^{\mathrm{T}}, \mathbf{0}_{m\times\bar{d}_1}, \ldots, \mathbf{D}_{M-1}^{\mathrm{T}}, \mathbf{0}_{m\times\bar{d}_{M-1}}, \mathbf{D}_M^{\mathrm{T}})^{\mathrm{T}}.$$

$$\mathbf{E}_M = (\beta(T_M), \ldots, \beta^{(s-1)}(T_M))^{\mathrm{T}}$$

$$\beta(x) = (1, x, \ldots, x^N)^{\mathrm{T}} \qquad N = 2s - 1 = 5$$

$$\mathbf{D}_M \in \mathbb{R}^{s\times m} \qquad (\text{s=3,} \quad \text{m=3})$$

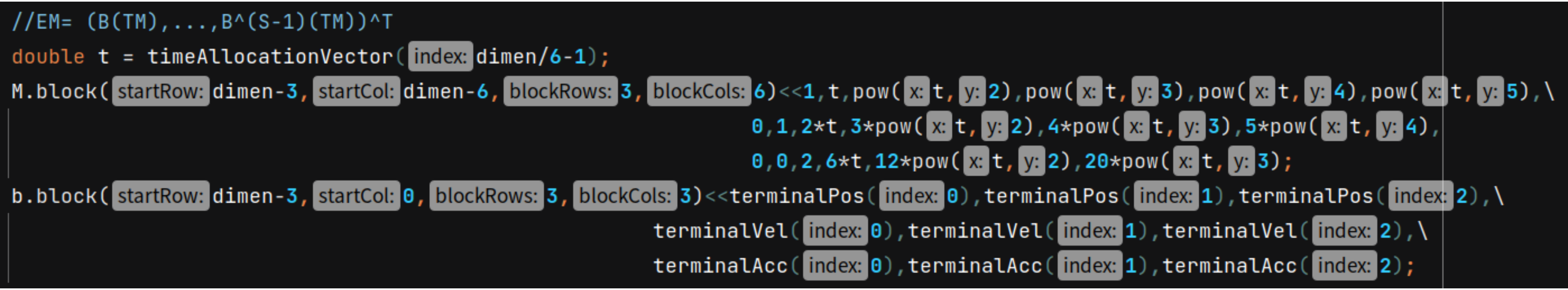
由此可得：

$$E_M = \begin{pmatrix} 1 & t & t^2 & t^3 & t^4 & t^5 \\ 0 & 1 & 2t & 3t^2 & 4t^3 & 5t^4 \\ 0 & 0 & 2 & 6t & 12t^2 & 20t^3 \end{pmatrix}$$

$$D_M = (p_{xM}, p_{yM}, p_{zM})$$

思路提示 —构造矩阵

填补到M矩阵和b矩阵对应的位置，代码如下：



```
//EM= (B(TM),...,B^(S-1)(TM))^T
double t = timeAllocationVector( index: dimen/6-1);
M.block( startRow: dimen-3, startCol: dimen-6, blockRows: 3, blockCols: 6)<<1,t,pow( x: t, y: 2),pow( x: t, y: 3),pow( x: t, y: 4),pow( x: t, y: 5),\
                                                              0,1,2*t,3*pow( x: t, y: 2),4*pow( x: t, y: 3),5*pow( x: t, y: 4),
                                                              0,0,2,6*t,12*pow( x: t, y: 2),20*pow( x: t, y: 3);
b.block( startRow: dimen-3, startCol: 0, blockRows: 3, blockCols: 3)<<terminalPos( index: 0),terminalPos( index: 1),terminalPos( index: 2),\
                                                      terminalVel( index: 0),terminalVel( index: 1),terminalVel( index: 2),\
                                                      terminalAcc( index: 0),terminalAcc( index: 1),terminalAcc( index: 2);
```

思路提示 —求解器求解

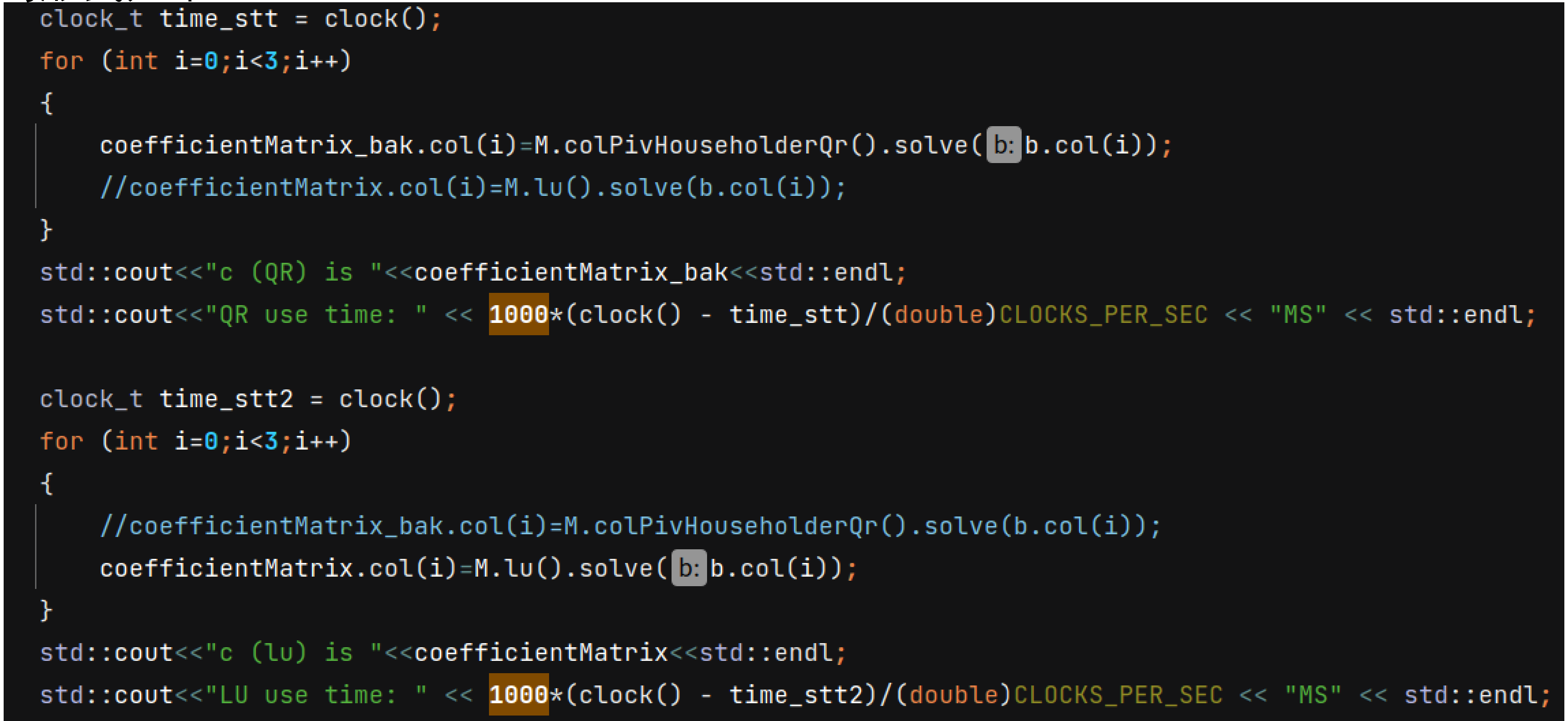
这里我分别采用QR和lu方法。注意，我们构建的M矩阵是针对单一维度而言的，因此，我们对每一维度分开求解。也就是说我们分别求解xyz维度的三个方程:

$$Mc(0) = b(0)$$
$$Mc(1) = b(1)$$
$$Mc(2) = b(2)$$

代码形式如下:

思路提示 —求解器求解

代码形式如下：

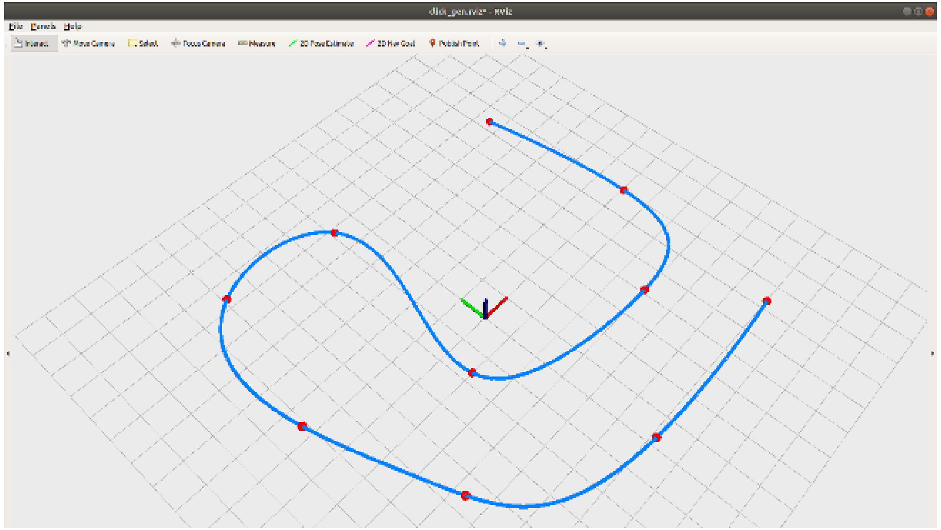


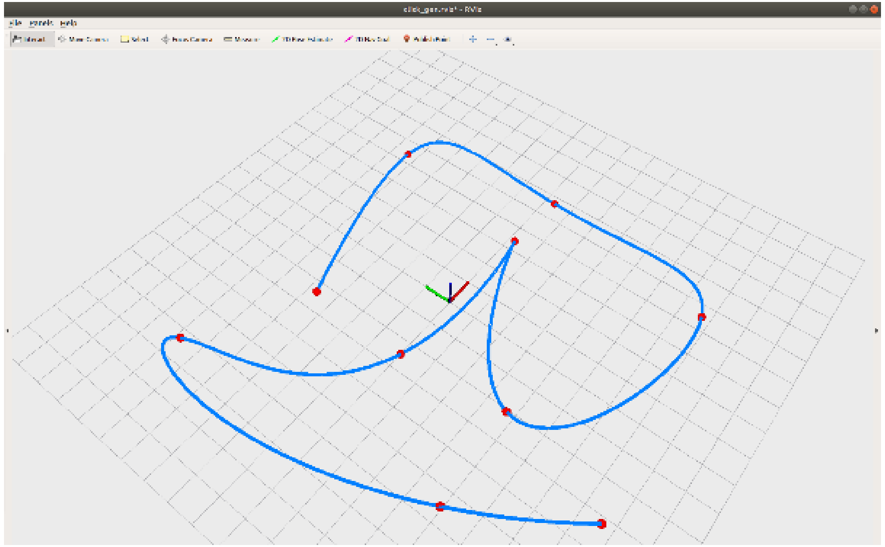
```
clock_t time_stt = clock();
for (int i=0;i<3;i++)
{
    coefficientMatrix_bak.col(i)=M.colPivHouseholderQr().solve( b: b.col(i));
    //coefficientMatrix.col(i)=M.lu().solve(b.col(i));
}
std::cout<<"c (QR) is "<<coefficientMatrix_bak<<std::endl;
std::cout<<"QR use time: " << 1000*(clock() - time_stt)/(double)CLOCKS_PER_SEC << "MS" << std::endl;

clock_t time_stt2 = clock();
for (int i=0;i<3;i++)
{
    //coefficientMatrix_bak.col(i)=M.colPivHouseholderQr().solve(b.col(i));
    coefficientMatrix.col(i)=M.lu().solve( b: b.col(i));
}
std::cout<<"c (lu) is "<<coefficientMatrix<<std::endl;
std::cout<<"LU use time: " << 1000*(clock() - time_stt2)/(double)CLOCKS_PER_SEC << "MS" << std::endl;
```

思路提示 —效果

深蓝学院
shenlanxueyuan.com

我们用“2D Nav Goal”给定waypoint的二维坐标信息，z轴的坐标可以通过config/click_gen.yaml设定





在线问答

深蓝学院
shenlanxueyuan.com

Q&A

深蓝学院
shenlanxueyuan.com

感谢各位聆听！

Thanks for Listening