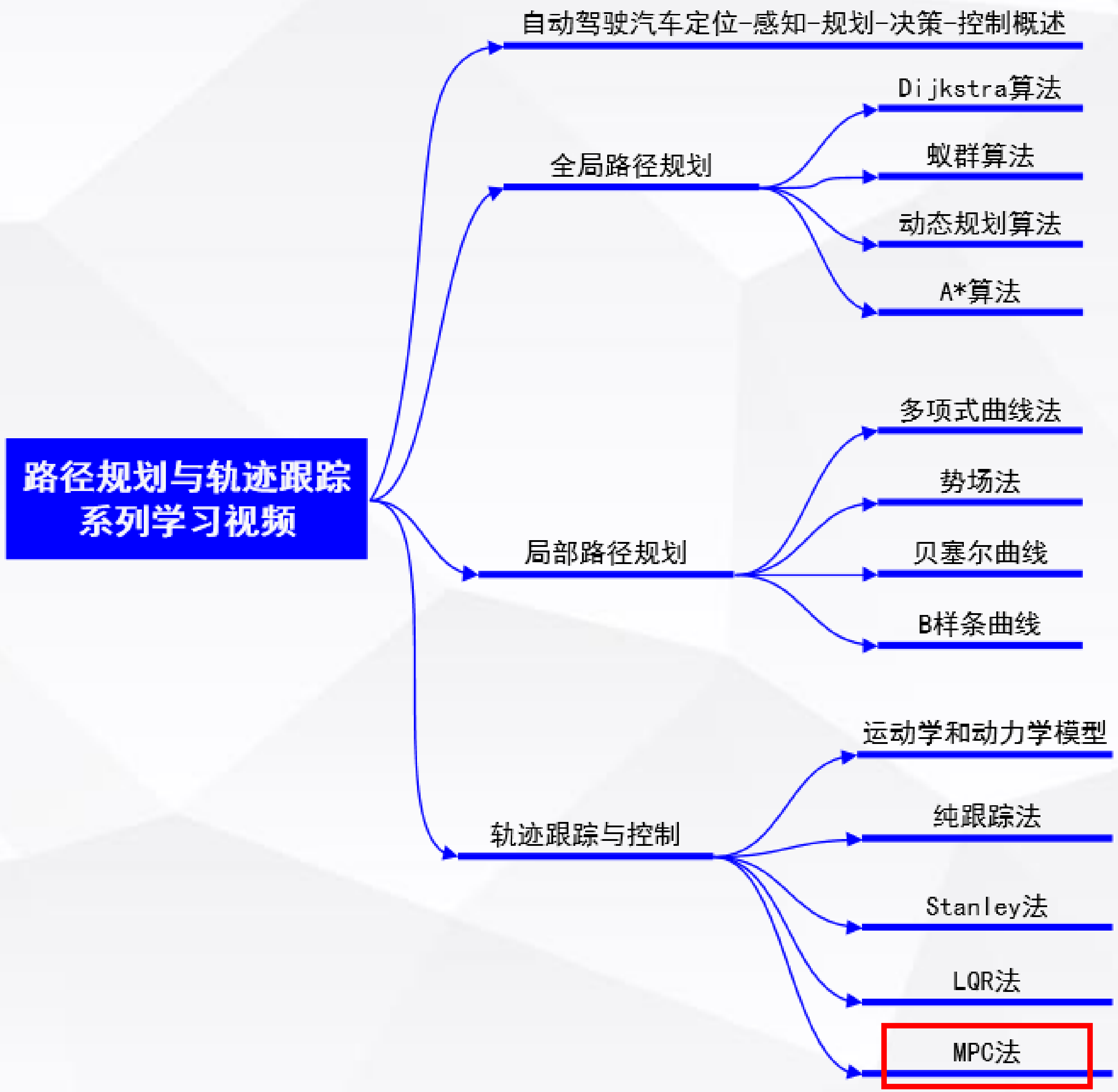
智能汽车路径规划与轨迹跟踪系列算法精讲及Matlab程序实现

第13讲　模型预测控制(MPC)法

创作者：Ally

时间：2021/4/3

1 学习课程大纲目录



路径规划与轨迹跟踪系列学习视频
自动驾驶汽车定位-感知-规划-决策-控制概述
全局路径规划
Dijkstra算法
蚁群算法
动态规划算法
A*算法
局部路径规划
多项式曲线法
势场法
贝塞尔曲线
B样条曲线
轨迹跟踪与控制
运动学和动力学模型
纯跟踪法
Stanley法
LQR法
MPC法

初识模型预测控制

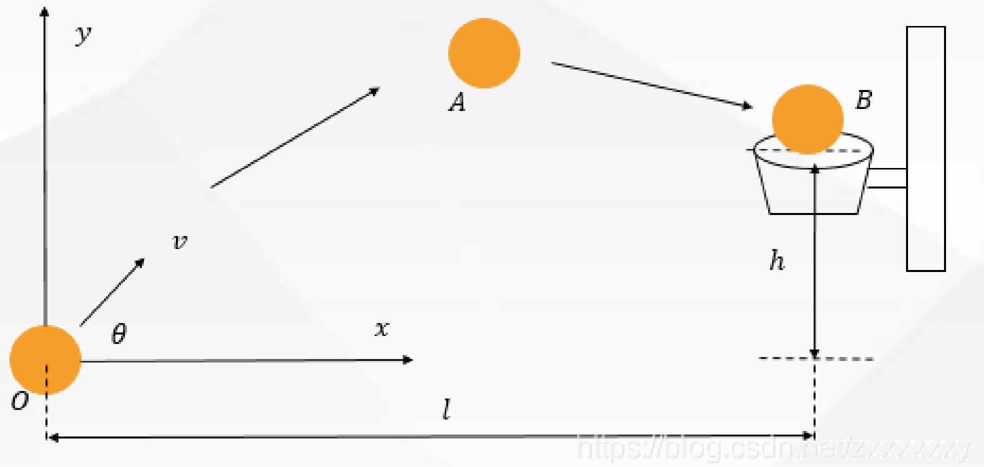
- 根据高中物理知识，将篮球视为质点，在不考虑空气阻力作用下，设篮球斜抛的角度为θ，斜抛初始速度为v，初始高度为y0（右图为0），则篮球的斜抛运动由竖直方向的上抛运动和水平方向的匀速运动构成：
- （1）竖直方向：

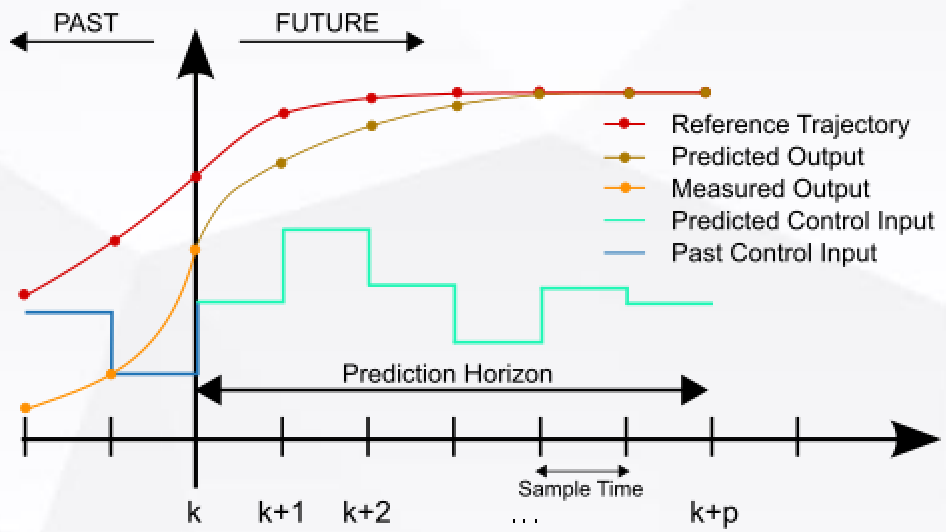
$$y(t)=y_0+v\sin\theta\cdot t-\frac{1}{2}gt^2 \qquad (1)$$

- （2）水平方向：

$$x(t)=v\cos\theta\cdot t \qquad (2)$$

- 那么对于离初始点水平距离为l、竖直距离h的球框位置，若想在初始投球点能够顺利投中球框位置，可以联立上面两式求得一组解集（v, θ ）。
- 那么这里的式（1）和式（2）就是"模型（Model）"，篮球能否按照预期投进球框就是"预测（Prediction）"，通过调整初始点投球时的角度和速度就是"控制（Control）"。
- 当然，若期望预测控制量比较精确，我们会需要更细致地考虑不同位置的空气阻力等因素对篮球投射过程的影响，而不只是简单地假设没有空气阻力存在以及球在空气中的水平速度保持不变。





模型（Model）

➢ 设车辆的状态量偏差和控制量偏差如式（3）所示：

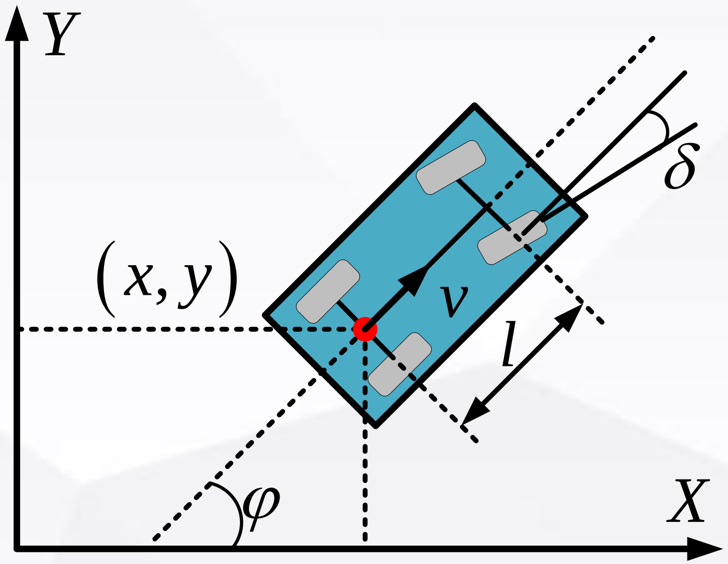
$$\tilde{\boldsymbol{x}}=\begin{bmatrix}\dot{x}-\dot{x}_r\\ \dot{y}-\dot{y}_r\\ \dot{\varphi}-\dot{\varphi}_r\end{bmatrix},\tilde{\boldsymbol{u}}=\begin{bmatrix}v-v_r\\ \delta-\delta_r\end{bmatrix} \tag{3}$$

➢ 基于第9讲的运动学模型的离散状态空间方程如下，

$$\tilde{\mathbf{x}}(k+1)=\begin{bmatrix}1 & 0 & -Tv_r\sin\varphi_r\\ 0 & 1 & Tv_r\cos\varphi_r\\ 0 & 0 & 1\end{bmatrix}\tilde{\mathbf{x}}(k)+\begin{bmatrix}T\cos\varphi_r & 0\\ T\sin\varphi_r & 0\\ T\dfrac{\tan\varphi_r}{l} & T\dfrac{v_r}{l\cos^2\delta_r}\end{bmatrix}\tilde{\mathbf{u}}(k)=\boldsymbol{a}\tilde{\mathbf{x}}(k)+\boldsymbol{b}\tilde{\mathbf{u}}(k) \tag{4}$$

➢ 定义输出方程：

$$\boldsymbol{y}(k)=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}\tilde{\mathbf{x}}(k)=\boldsymbol{c}\tilde{\mathbf{x}}(k) \tag{5}$$



4 轨迹跟踪控制算法——模型预测控制(MPC)法

模型（Model）

- 构建新的状态向量：

$$\xi(k)=\begin{bmatrix}\tilde{\boldsymbol{x}}(k)\\ \tilde{\boldsymbol{u}}(k-1)\end{bmatrix} \tag{6}$$

- 那么新的状态空间表达式：

$$\begin{aligned}
\boxed{\xi(k+1)=}\begin{bmatrix}\tilde{\boldsymbol{x}}(k+1)\\ \tilde{\boldsymbol{u}}(k)\end{bmatrix}&=\begin{bmatrix}\boldsymbol{a}\tilde{\boldsymbol{x}}(k)+\boldsymbol{b}\tilde{\boldsymbol{u}}(k)\\ \tilde{\boldsymbol{u}}(k)\end{bmatrix}=\begin{bmatrix}\boldsymbol{a}\tilde{\boldsymbol{x}}(k)+\boldsymbol{b}\tilde{\boldsymbol{u}}(k-1)+\boldsymbol{b}\tilde{\boldsymbol{u}}(k)-\boldsymbol{b}\tilde{\boldsymbol{u}}(k-1)\\ \tilde{\boldsymbol{u}}(k-1)+\tilde{\boldsymbol{u}}(k)-\tilde{\boldsymbol{u}}(k-1)\end{bmatrix}=\begin{bmatrix}\boldsymbol{a}\tilde{\boldsymbol{x}}(k)+\boldsymbol{b}\tilde{\boldsymbol{u}}(k-1)\\ \tilde{\boldsymbol{u}}(k-1)\end{bmatrix}+\begin{bmatrix}\boldsymbol{b}\tilde{\boldsymbol{u}}(k)-\boldsymbol{b}\tilde{\boldsymbol{u}}(k-1)\\ \tilde{\boldsymbol{u}}(k)-\tilde{\boldsymbol{u}}(k-1)\end{bmatrix}\\
&=\begin{bmatrix}\begin{bmatrix}\boldsymbol{a} & \boldsymbol{b}\end{bmatrix}\begin{bmatrix}\tilde{\boldsymbol{x}}(k)\\ \tilde{\boldsymbol{u}}(k-1)\end{bmatrix}\\ \begin{bmatrix}\boldsymbol{0} & \boldsymbol{I}_{N_u}\end{bmatrix}\begin{bmatrix}\tilde{\boldsymbol{x}}(k)\\ \tilde{\boldsymbol{u}}(k-1)\end{bmatrix}\end{bmatrix}+\begin{bmatrix}\boldsymbol{b}\\ \boldsymbol{I}_{N_u}\end{bmatrix}\left(\tilde{\boldsymbol{u}}(k)-\tilde{\boldsymbol{u}}(k-1)\right)=\begin{bmatrix}\boldsymbol{a} & \boldsymbol{b}\\ \boldsymbol{0} & \boldsymbol{I}_{N_u}\end{bmatrix}\begin{bmatrix}\tilde{\boldsymbol{x}}(k)\\ \tilde{\boldsymbol{u}}(k-1)\end{bmatrix}+\begin{bmatrix}\boldsymbol{b}\\ \boldsymbol{I}_{N_u}\end{bmatrix}\left(\tilde{\boldsymbol{u}}(k)-\tilde{\boldsymbol{u}}(k-1)\right)=\begin{bmatrix}\boldsymbol{a} & \boldsymbol{b}\\ \boldsymbol{0} & \boldsymbol{I}_{N_u}\end{bmatrix}\xi(k)+\begin{bmatrix}\boldsymbol{b}\\ \boldsymbol{I}_{N_u}\end{bmatrix}\Delta\tilde{\boldsymbol{u}}(k)\\
&=\boxed{\boldsymbol{A}\xi(k)+\boldsymbol{B}\Delta\tilde{\boldsymbol{u}}(k)}
\end{aligned} \tag{7}$$

- 那么输出方程为：

$$\eta(k)=\begin{bmatrix}\boldsymbol{I}_{N_x} & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\tilde{\boldsymbol{x}}(k)\\ \tilde{\boldsymbol{u}}(k-1)\end{bmatrix}=\boldsymbol{C}\xi(k) \tag{8}$$

Ally

预测（Predict）

- 设预测时域未Np，控制时域为Nc（ Np ≥Nc），对式（7）进行多步骤推导：

$$
\begin{aligned}
&\xi(k+1)=A\xi(k)+B\Delta\tilde{u}(k)\\
&\xi(k+2)=A\xi(k+1)+B\Delta\tilde{u}(k+1)=A^2\xi(k)+AB\Delta\tilde{u}(k)+B\Delta\tilde{u}(k+1)\\
&\xi(k+3)=A\xi(k+2)+B\Delta\tilde{u}(k+2)=A^3\xi(k)+A^2B\Delta\tilde{u}(k)+AB\Delta\tilde{u}(k+1)+B\Delta\tilde{u}(k+2)\\
&\vdots\\
&\xi(k+N_C)=A^{N_C}\xi(k)+A^{N_C-1}B\Delta\tilde{u}(k)+A^{N_C-2}B\Delta\tilde{u}(k+1)+\cdots+A^0B\Delta\tilde{u}(k+N_C-1)\\
&\vdots\\
&\xi(k+N_P)=A^{N_P}\xi(k)+\boxed{A^{N_P-1}B\Delta\tilde{u}(k)+A^{N_P-2}B\Delta\tilde{u}(k+1)+\cdots+A^0B\Delta\tilde{u}(k+N_P-1)}
\end{aligned}
\tag{9}
$$

- 规律：A的指数与u的控制步之和为Np+k-1

- 那么同样对式（8）进行多步骤推导，输出方程为：

$$
\begin{aligned}
&\eta(k+1)=C\xi(k+1)=CA\xi(k)+CB\Delta\tilde{u}(k)\\
&\eta(k+2)=CA^2\xi(k)+CAB\Delta\tilde{u}(k)+CB\Delta\tilde{u}(k+1)\\
&\eta(k+3)=CA^3\xi(k)+CA^2B\Delta\tilde{u}(k)+CAB\Delta\tilde{u}(k+1)+CB\Delta\tilde{u}(k+2)\\
&\vdots\\
&\eta(k+N_C)=CA^{N_C}\xi(k)+CA^{N_C-1}B\Delta\tilde{u}(k)+CA^{N_C-2}B\Delta\tilde{u}(k+1)+\cdots+CA^0B\Delta\tilde{u}(k+N_C-1)\\
&\vdots\\
&\eta(k+N_P)=CA^{N_P}\xi(k)+CA^{N_P-1}B\Delta\tilde{u}(k)+CA^{N_P-2}B\Delta\tilde{u}(k+1)+\cdots+CA^0B\Delta\tilde{u}(k+N_P-1)
\end{aligned}
\tag{10}
$$

- 规律：红框可以单独组成关于CA形式的大矩阵；蓝框可以单独组成CAB形式的大矩阵

Ally

预测（ Predict ）

➢ 对于输出方程（10），令

$$Y=\begin{bmatrix}\eta(k+1)\\ \eta(k+2)\\ \cdots\\ \eta(k+N_C)\\ \cdots\\ \eta(k+N_P)\end{bmatrix},\Psi=\begin{bmatrix}CA\\ CA^2\\ \cdots\\ CA^{N_C}\\ \cdots\\ CA^{N_P}\end{bmatrix},\Theta=\begin{bmatrix}CB & 0 & 0 & \cdots & 0\\ CAB & CB & 0 & \cdots & 0\\ \cdots & \cdots & \cdots & \ddots & \cdots\\ CA^{N_C-1}B & CA^{N_C-2}B & CA^{N_C-3}B & \cdots & CA^{0}B\\ \cdots & \cdots & \cdots & \ddots & \cdots\\ CA^{N_P-1}B & CA^{N_P-2}B & CA^{N_P-3}B & \cdots & CA^{N_P-N_C}B\end{bmatrix},\Delta U=\begin{bmatrix}\Delta\tilde{u}(k)\\ \Delta\tilde{u}(k+1)\\ \Delta\tilde{u}(k+2)\\ \cdots\\ \Delta\tilde{u}(k+N_C-1)\end{bmatrix} \quad (11)$$

➢ 那么输出方程可以改写为：

$$Y=\Psi\xi(k)+\Theta\Delta U \quad (12)$$

➢ 因此，若已知当前时刻的状态量，和控制时域Nc内的控制增量，也就可以预测未来预测时域Np的系统输出量。

➢ 规由于预测时域大于控制时域，当预测的时间步大于了Nc时，输出方程的表达就要受到Nc的限制，故最后一项的A指数不为0

目标函数设计

- 定义系统输出量的参考值为：

$$Y_r = \left[\eta_r(k+1) \quad \eta_r(k+2) \quad \cdots \quad \eta_r(k+N_C) \quad \cdots \quad \eta_r(k+N_P)\right]^{\mathrm{T}} \\ = [\mathbf{0} \quad \mathbf{0} \quad \cdots \quad \mathbf{0} \quad \cdots \quad \mathbf{0}]^{\mathrm{T}} \tag{13}$$

- 设 $E = \Psi\xi(k)$， $Q_Q = I_{N_P} \boxed{\otimes} Q, R_R = I_{N_P} \otimes R$， 参考第12讲，定义优化目标函数为：

$$J = \tilde{Y}^{\mathrm{T}} Q_Q \tilde{Y} + \Delta U^{\mathrm{T}} R_R \Delta U = (Y - Y_r)^{\mathrm{T}} Q_Q (Y - Y_r) + \Delta U^{\mathrm{T}} R_R \Delta U \\ = \left[\left(\Psi x(k) + \Theta \Delta U\right) - Y_r\right]^{\mathrm{T}} Q_Q \left[\left(\Psi x(k) + \Theta \Delta U\right) - Y_r\right] + \Delta U^{\mathrm{T}} R_R \Delta U \\ = \Delta U^{\mathrm{T}}\left(\Theta^{\mathrm{T}} Q_Q \Theta + R_R\right)\Delta U + 2E^{\mathrm{T}} Q_Q \Theta \Delta U \boxed{+ E^{\mathrm{T}} Q_Q E - Y_r Q_Q \Theta \Delta U + Y_r^{\mathrm{T}} Q_Q Y - 2Y_r^{\mathrm{T}} Q_Q E} \tag{14}$$

- 令 $H = \Theta^{\mathrm{T}} Q_Q \Theta + R_R, g = E^{\mathrm{T}} Q_Q \Theta$ ，则式（14），可以改写为：

$$\min_{\Delta U} J = 2\left(\frac{1}{2}\Delta U^{\mathrm{T}} H \Delta U + g^{\mathrm{T}} \Delta U\right) \Leftrightarrow \min_{\Delta U} J = \frac{1}{2}\Delta U^{\mathrm{T}} H \Delta U + g^{\mathrm{T}} \Delta U \tag{15}$$

- Yr代表参考输出量，由于状态量是误差形式，故参考输出量为0，

- 克罗内克积是两个任意大小的矩阵间的运算如果A是一个m×n的矩阵，而B是一个p×q的矩阵，克罗内克积则是一个mp×nq的分块矩阵

- 红色矩形框为常数，在求解目标函数时可以舍去

Ally

控制（Control ）

- 对于控制量和控制增量，有如下递推式：

$$\begin{aligned}
&\tilde{\boldsymbol{u}}(k)=\tilde{\boldsymbol{u}}(k-1)+\Delta\tilde{\boldsymbol{u}}(k)\\
&\tilde{\boldsymbol{u}}(k+1)=\tilde{\boldsymbol{u}}(k)+\Delta\tilde{\boldsymbol{u}}(k+1)=\tilde{\boldsymbol{u}}(k-1)+\Delta\tilde{\boldsymbol{u}}(k)+\Delta\tilde{\boldsymbol{u}}(k+1)\\
&\cdots\\
&\tilde{\boldsymbol{u}}(k+N_C-1)=\tilde{\boldsymbol{u}}(k+N_C-2)+\Delta\tilde{\boldsymbol{u}}(k+N_C-1)=\tilde{\boldsymbol{u}}(k-1)+\Delta\tilde{\boldsymbol{u}}(k)+\Delta\tilde{\boldsymbol{u}}(k+1)+\cdots\Delta\tilde{\boldsymbol{u}}(k+N_C-1)
\end{aligned} \tag{16}$$

- 上式可以改写为：

$$\boldsymbol{U}=\begin{bmatrix}\tilde{\boldsymbol{u}}(k)\\ \tilde{\boldsymbol{u}}(k+1)\\ \tilde{\boldsymbol{u}}(k+2)\\ \cdots\\ \tilde{\boldsymbol{u}}(k+N_C-1)\end{bmatrix}=\begin{bmatrix}\tilde{\boldsymbol{u}}(k-1)\\ \tilde{\boldsymbol{u}}(k-1)\\ \tilde{\boldsymbol{u}}(k-1)\\ \cdots\\ \tilde{\boldsymbol{u}}(k-1)\end{bmatrix}+\begin{bmatrix}\boldsymbol{I}_2 & 0 & 0 & \cdots & 0\\ \boldsymbol{I}_2 & \boldsymbol{I}_2 & 0 & \cdots & 0\\ \boldsymbol{I}_2 & \boldsymbol{I}_2 & \boldsymbol{I}_2 & \cdots & 0\\ \cdots & \cdots & \cdots & \ddots & 0\\ \boldsymbol{I}_2 & \boldsymbol{I}_2 & \boldsymbol{I}_2 & \cdots & \boldsymbol{I}_2\end{bmatrix}\begin{bmatrix}\Delta\tilde{\boldsymbol{u}}(k)\\ \Delta\tilde{\boldsymbol{u}}(k+1)\\ \Delta\tilde{\boldsymbol{u}}(k+2)\\ \cdots\\ \Delta\tilde{\boldsymbol{u}}(k+N_C-1)\end{bmatrix}=\boldsymbol{U}_t+\boldsymbol{A}_I\Delta\boldsymbol{U} \tag{17}$$

- 综上参照式（5）和式（6），有：

$$\boldsymbol{U}_{\min}=\begin{bmatrix}\tilde{\boldsymbol{u}}_{\min}\\ \tilde{\boldsymbol{u}}_{\min}\\ \tilde{\boldsymbol{u}}_{\min}\\ \cdots\\ \tilde{\boldsymbol{u}}_{\min}\end{bmatrix}\le\begin{bmatrix}\tilde{\boldsymbol{u}}(k)\\ \tilde{\boldsymbol{u}}(k+1)\\ \tilde{\boldsymbol{u}}(k+2)\\ \cdots\\ \tilde{\boldsymbol{u}}(k+N_C-1)\end{bmatrix}\le\begin{bmatrix}\tilde{\boldsymbol{u}}_{\max}\\ \tilde{\boldsymbol{u}}_{\max}\\ \tilde{\boldsymbol{u}}_{\max}\\ \cdots\\ \tilde{\boldsymbol{u}}_{\max}\end{bmatrix}=\boldsymbol{U}_{\max}
\;\Rightarrow\;
\boldsymbol{U}_{\min}\le\boldsymbol{U}_t+\boldsymbol{A}_I\Delta\boldsymbol{U}\le\boldsymbol{U}_{\max}
\;\Downarrow\;
\begin{cases}\boldsymbol{A}_I\Delta\boldsymbol{U}\le\boldsymbol{U}_{\max}-\boldsymbol{U}\\ -\boldsymbol{A}_I\Delta\boldsymbol{U}\le-\boldsymbol{U}_{\min}+\boldsymbol{U}_t\end{cases} \tag{18}$$

总结

- 综上，模型预测控制问题转为了一个标准二次型规划（ Quadratic Programming ，QP）问题

$$
\begin{aligned}
&\min_{\Delta \boldsymbol{U}} J = \frac{1}{2}\Delta \boldsymbol{U}^{\mathrm{T}} \boldsymbol{H} \Delta \boldsymbol{U} + \boldsymbol{g}^{\mathrm{T}} \Delta \boldsymbol{U} \\
&\text{s.t.} \begin{cases} \boldsymbol{A}_I \Delta \boldsymbol{U}_t \le \boldsymbol{U}_{\max} - \boldsymbol{U} \\ \boldsymbol{A}_I \Delta \boldsymbol{U}_t \le -\boldsymbol{U}_{\min} + \boldsymbol{U}_t \\ \Delta \boldsymbol{U}_{\min} \le \Delta \boldsymbol{U} \le \Delta \boldsymbol{U}_{\max} \end{cases}
\end{aligned} \tag{19}
$$