MPC-QP Controller for CL686 Expermental Project

1 Control Relevant Perturbation Model and Kalman Predictor Design

Consider the discrete time linear model obtained through linearization of the mechanistic model in the neighborhood of $(\mathcal{X}_s, \mathcal{U}_s, \mathcal{D}_s)$

$$\mathbf{x}(k+1) = \mathbf{\Phi x}(k) + \mathbf{\Gamma}_u \mathbf{u}(k) + \mathbf{\Gamma}_d \mathbf{d}(k) \tag{1}$$

$$\mathbf{y}(k) = \mathbf{Cx}(k) + \mathbf{D} \begin{bmatrix} \mathbf{u}(k) \\ \mathbf{d}(k) \end{bmatrix} + \mathbf{v}(k) \tag{2}$$

$$\mathbf{x}(k) = \mathcal{X}(k) - \mathcal{X}_s, \mathbf{u}(k) = \mathcal{U}(k) - \mathcal{U}_s, \tag{3}$$

$$\mathbf{d} = \mathcal{D}(k) - \mathcal{D}_s, \mathbf{y}(k) = \mathcal{Y}(k) - \mathbf{C}\mathcal{X}_s \tag{4}$$

where $\mathbf{d}(k)$ and $\mathbf{v}(k)$ are a zero mean Gaussian white noise sequences with covariance matrices $\mathbf{Q}_d$ and $\mathbf{R}$,respectively. Here, $\mathbf{D}$ is a **null matrix** of dimension $r \times (m+d)$. Also, define covariance matrix

$$\mathbf{N} = E\left[\mathbf{d}(k)\mathbf{v}(k)^T\right] = [\mathbf{0}]_{d \times r}$$

which is a null matrix since disturbance $\mathbf{d}(k)$ and measurement noise $\mathbf{v}(k)$ are uncorrelated.

This model can be used to develop Kalman predictor of the form

$$\mathbf{y}(k) = \mathcal{Y}(k) - \mathcal{Y}_s \tag{5}$$

$$\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{C}\widehat{\mathbf{x}}(k) \tag{6}$$

$$\widehat{\mathbf{x}}(k+1) = \mathbf{\Phi}\widehat{\mathbf{x}}(k) + \mathbf{\Gamma}_u \mathbf{u}(k) + \mathbf{L}_p \mathbf{e}(k) \tag{7}$$

where the steady state Kalman gain is obtained by solving the algebraic Riccati equation (ARE)

$$\mathbf{L}_p = \left[\mathbf{\Phi P}_\infty \mathbf{C}^T + \mathbf{N}\right]\left[\mathbf{CP}_\infty \mathbf{C}^T + \mathbf{R}\right]^{-1} \tag{8}$$

$$\mathbf{P}_\infty = \mathbf{\Phi P}_\infty \mathbf{\Phi}^T + \mathbf{\Gamma}_d \mathbf{Q}_d \mathbf{\Gamma}_d^T - \mathbf{L}_p \left[\mathbf{CP}_\infty \mathbf{C}^T + \mathbf{R}\right] \mathbf{L}_p^T \tag{9}$$

Solution of the ARE can be found using Matlab Control System Toolbox function *kalman* as follows

1. Step 1: Create a state space object using ss command
   dmod = ss($\mathbf{\Phi}$, $\begin{bmatrix} \mathbf{\Gamma}_u & \mathbf{\Gamma}_d \end{bmatrix}$, $\mathbf{C}$, $\mathbf{D}$, T *)*
   where T represents sampling interval.

2. Call Matlab function *kalman*
   [KEST, $\mathbf{L}_p$,$\mathbf{P}_\infty$*] = kalman(dmod,* $\mathbf{Q}_d$, $\mathbf{R}$,$\mathbf{N}$*) ;*

Use the Kalman predictor for MPC implementation. For the purpose of innovation bias based MPC implementation, the filtered innovation sequence is computed as follows

$$\mathbf{e}_f(k) = \mathbf{\Phi}_e \ \mathbf{e}_f(k-1) + [\mathbf{I} - \mathbf{\Phi}_e]\mathbf{e}(k) \tag{10}$$

$$\mathbf{\Phi}_e = \alpha \mathbf{I}_{r \times r} \tag{11}$$

$$0 \leq \alpha < 1 \;\; \text{is tuning parameter}$$

The innovation signal, $\mathbf{e}(k)$, is computed using eq. (5). Further, the target steady state $\mathbf{x}_s(k)$ and target input $\mathbf{u}_s(k)$ are computed as follows

$$\mathbf{u}_s(k) = \mathbf{K}_u^{-1}\left[\mathbf{r}(k) - \mathbf{K}_e\,\mathbf{e}_f(k)\right] \tag{12}$$

$$\mathbf{x}_s(k) = (\mathbf{I} - \mathbf{\Phi})^{-1}\;\left[\mathbf{\Gamma}_u\mathbf{u}_s(k) + \mathbf{L}_p\mathbf{e}_f(k)\right] \tag{13}$$

$$\mathbf{K}_u = \mathbf{C}\left(\mathbf{I} - \mathbf{\Phi}\right)^{-1}\Gamma_u \quad ; \qquad \mathbf{K}_e = \mathbf{C}\left(\mathbf{I} - \mathbf{\Phi}\right)^{-1}\mathbf{L}_p + \mathbf{I} \tag{14}$$

where $\mathbf{r}(k)$ denotes desired setpoint at instant k.

2 Model Identified from Data

Alternatively, you may have innovation form of state space model of the form

$$\mathbf{x}(k+1) = \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Gamma}_u\mathbf{u}(k) + \mathbf{L}_p\mathbf{e}(k)$$

identified directly from input-output data.

$$\mathbf{y}(k) = \mathcal{Y}(k) - \mathcal{Y}_s \tag{15}$$

$$\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{C}\widehat{\mathbf{x}}(k) \tag{16}$$

$$\widehat{\mathbf{x}}(k+1) = \mathbf{\Phi}\widehat{\mathbf{x}}(k) + \mathbf{\Gamma}_u\mathbf{u}(k) + \mathbf{L}_p\mathbf{e}(k) \tag{17}$$

For the purpose of innovation bias based MPC implementation, the filtered innovation sequence is computed as follows

$$\mathbf{e}_f(k) = \mathbf{\Phi}_e\;\mathbf{e}_f(k-1) + [\mathbf{I} - \mathbf{\Phi}_e]\mathbf{e}(k) \tag{18}$$

$$\mathbf{\Phi}_e = \alpha\mathbf{I}_{r\times r} \tag{19}$$

$$0 \leq \alpha < 1 \;\; \text{is tuning parameter}$$

The innovation signal, $\mathbf{e}(k)$, is computed using eq. (5). Further, the target steady state $\mathbf{x}_s(k)$ and target input $\mathbf{u}_s(k)$ are computed as follows

$$\mathbf{u}_s(k) = \mathbf{K}_u^{-1}\left[\mathbf{r}(k) - \mathbf{K}_e\,\mathbf{e}_f(k)\right] \tag{20}$$

$$\mathbf{x}_s(k) = (\mathbf{I} - \mathbf{\Phi})^{-1}\;\left[\mathbf{\Gamma}_u\mathbf{u}_s(k) + \mathbf{L}_p\mathbf{e}_f(k)\right] \tag{21}$$

$$\mathbf{K}_u = \mathbf{C}\left(\mathbf{I} - \mathbf{\Phi}\right)^{-1}\Gamma_u \quad ; \qquad \mathbf{K}_e = \mathbf{C}\left(\mathbf{I} - \mathbf{\Phi}\right)^{-1}\mathbf{L}_p + \mathbf{I} \tag{22}$$

where $\mathbf{r}(k)$ denotes desired setpoint at instant k.

3 MPC QP Formulation

The model predictive control problem at the sampling instant k is defined as a constrained optimization problem whereby the future manipulated input moves

$$\mathcal{U}_f = \{\mathbf{u}(k+j|k) : \; j = 0, 1, ...q-1\}$$

are determined by minimizing a cost function defined over prediction horizon p. Here, q is known as the control horizon. Note that use of control horizon $q < p$ implies inclusion of the following constraints on the future manipulated inputs

$$\mathbf{u}(k+q|k) = \mathbf{u}(k+q+1|k) = ... = \mathbf{u}(k+p-1|k) = \mathbf{u}(k+q-1|k) \tag{23}$$

Let $\mathbf{w}_x$,$\mathbf{w}_u$ represent +ve definite matrices. As part the course project, you are expected to implement the following MPC algorithm

$$\underset{\mathcal{U}_f}{\arg Min} \quad J = \sum_{j=1}^{p} \boldsymbol{\varepsilon}(k+j|k)^T \mathbf{w}_x \boldsymbol{\varepsilon}(k+j|k) + \sum_{j=0}^{q-1} \delta\mathbf{u}(k+j|k)^T \mathbf{w}_u \delta\mathbf{u}(k+j|k) \tag{24}$$

$$\begin{aligned} \boldsymbol{\varepsilon}(k+j|k) &= \widehat{\mathbf{z}}(k+j) - \mathbf{x}_s(k) \text{ for } \quad j = 1, 2, ...p \\ \mathbf{x}_s(k) &: \text{ computed using (13)} \end{aligned} \tag{25}$$

$$\begin{aligned} \delta\mathbf{u}(k+j|k) &= \mathbf{u}(k+j|k) - \mathbf{u}_s(k) \quad \text{for } \ j = 1, ...q-1 \\ \mathbf{u}_s(k) &: \text{ computed using (12)} \end{aligned} \tag{26}$$

Subject to

$$\widehat{\mathbf{z}}(k+j+1) = \boldsymbol{\Phi}\widehat{\mathbf{z}}(k+j) + \boldsymbol{\Gamma}\mathbf{u}(k+j|k) + \mathbf{L}_p\mathbf{e}_f(k) \tag{27}$$

$$\text{for } \ j = 1, 2, ...p. \text{ with } \widehat{\mathbf{z}}(k) = \widehat{\mathbf{x}}(k) \tag{28}$$

$$\begin{aligned} \widehat{\mathbf{x}}(k) &: \text{ computed using eq. (7)} \\ \mathbf{e}_f(k) &: \text{ computed using eq. (10)} \end{aligned}$$

$$\begin{aligned} \mathbf{u}_L &\leq \mathbf{u}(k+j|k) \leq \mathbf{u}_H \\ j &= 0, 1, 2, ...q-1 \end{aligned} \tag{29}$$

Defining vector of future inputs and outputs, as

$$\mathbf{U}_f(k) = \begin{bmatrix} \mathbf{u}(k|k)^T & \mathbf{u}(k+1|k)^T & .... & \mathbf{u}(k+q-1|k)^T \end{bmatrix}^T \tag{30}$$

$$\widehat{\mathbf{Z}}(k) = \begin{bmatrix} \widehat{\mathbf{z}}(k+1)^T & \widehat{\mathbf{z}}(k+2)^T & ..... & \widehat{\mathbf{z}}(k+p)^T \end{bmatrix}^T \tag{31}$$

the model based prediction equations and future prediction error can be expressed as follows

$$\widehat{\mathbf{Z}}(k) = \mathbf{S}_x\widehat{\mathbf{x}}(k|k-1) + \mathbf{S}_e\widehat{\mathbf{e}}_f(k) + \mathbf{S}_u\mathbf{U}_f(k) \tag{32}$$

$$\mathbf{S}_x = \begin{bmatrix} \boldsymbol{\Phi} \\ \boldsymbol{\Phi}^2 \\ ...... \\ \boldsymbol{\Phi}^p \end{bmatrix} \quad ; \quad \mathbf{S}_e = \begin{bmatrix} \mathbf{L} \\ (\boldsymbol{\Phi}+\mathbf{I})\mathbf{L} \\ ...... \\ (\boldsymbol{\Phi}^{p-1} + \boldsymbol{\Phi}^{p-2} + ...+\mathbf{I})\mathbf{L} \end{bmatrix} \quad ; \quad \mathbf{S}_{xI} = \begin{bmatrix} \mathbf{I}_n \\ \mathbf{I}_n \\ ... \\ \mathbf{I}_n \end{bmatrix} \tag{33}$$

$$\mathbf{S}_u = \begin{bmatrix} \mathbf{\Gamma}_u & [0] & [0] & .... & [0] \\ \mathbf{\Phi\Gamma}_u & \mathbf{\Gamma}_u & [0] & .... & [0] \\ ..... & ..... & .... & ... & [0] \\ \mathbf{\Phi}^{q-1}\mathbf{\Gamma}_u & \mathbf{\Phi}^{q-2}\mathbf{\Gamma}_u & .... & ... & \mathbf{\Gamma}_u \\ \mathbf{\Phi}^{q}\mathbf{\Gamma}_u & \mathbf{\Phi}^{q-1}\mathbf{\Gamma}_u & ... & .... & (\mathbf{\Phi}+I)\mathbf{\Gamma}_u \\ ..... & ...... & .... & .... & .... \\ \mathbf{\Phi}^{p-1}\mathbf{\Gamma}_u & \mathbf{\Phi}^{p-2}\mathbf{\Gamma}_u & ... & ... & (\mathbf{\Phi}^{p-q}+....+I)\mathbf{\Gamma}_u \end{bmatrix} \; ; \; \mathbf{S}_{uI} = \begin{bmatrix} \mathbf{I}_m \\ \mathbf{I}_m \\ ... \\ \mathbf{I}_m \end{bmatrix} \tag{34}$$

$$\mathbf{W}_X = block\ diag \begin{bmatrix} \mathbf{w}_x & \mathbf{w}_x & .... & \mathbf{w}_x \end{bmatrix} \tag{35}$$

$$\mathbf{W}_U = block\ diag \begin{bmatrix} \mathbf{w}_u & \mathbf{w}_u & .... & \mathbf{w}_u \end{bmatrix} \tag{36}$$

$$J = \left[\widehat{\mathbf{Z}}(k) - \mathbf{S}_{xI}\mathbf{x}_s(k)\right]^T \mathbf{W}_X \left[\widehat{\mathbf{Z}}(k) - \mathbf{S}_{xI}\mathbf{x}_s(k)\right] + \left[\mathbf{U}_f(k) - \mathbf{S}_{uI}\mathbf{u}_s(k)\right]^T \mathbf{W}_U \left[\mathbf{U}_f(k) - \mathbf{S}_{uI}\mathbf{u}_s(k)\right]$$

Consider the term

$$\begin{aligned} \widehat{\mathbf{Z}}(k) - \mathbf{S}_{xI}\mathbf{x}_s(k) &= \mathbf{S}_x\widehat{\mathbf{x}}(k|k-1) + \mathbf{S}_e\widehat{\mathbf{e}}_f(k) + \mathbf{S}_u\mathbf{U}_f(k) - \mathbf{S}_{xI}\mathbf{x}_s(k) \\ &= \mathbf{S}_u\mathbf{U}_f(k) + \boldsymbol{\eta}(k) \\ \boldsymbol{\eta}(k) &= \mathbf{S}_x\widehat{\mathbf{x}}(k|k-1) + \mathbf{S}_e\widehat{\mathbf{e}}_f(k) - \mathbf{S}_{xI}\mathbf{x}_s(k) \end{aligned}$$

The terms that contribute to QP formulation are

$$\begin{aligned} J &= \left[\mathbf{S}_u\mathbf{U}_f(k) + \boldsymbol{\eta}(k)\right]^T \mathbf{W}_X \left[\mathbf{S}_u\mathbf{U}_f(k) + \boldsymbol{\eta}(k)\right] \\ &\quad + \mathbf{U}_f(k)^T\mathbf{W}_U\mathbf{U}_f(k) - 2\left(\mathbf{S}_{uI}\mathbf{u}_s(k)\right)^T \mathbf{W}_U\mathbf{U}_f(k) + \left(\mathbf{S}_{uI}\mathbf{u}_s(k)\right)^T \mathbf{W}_U \left(\mathbf{S}_{uI}\mathbf{u}_s(k)\right) \\ &= \mathbf{U}_f(k)^T\mathbf{S}_u^T\mathbf{W}_X\mathbf{S}_u\mathbf{U}_f(k) + 2\boldsymbol{\eta}(k)^T\mathbf{W}_X\mathbf{S}_u\mathbf{U}_f(k) + \boldsymbol{\eta}(k)^T\mathbf{W}_X\boldsymbol{\eta}(k) \\ &\quad + \mathbf{U}_f(k)^T\mathbf{W}_U\mathbf{U}_f(k) - 2\left(\mathbf{S}_{uI}\mathbf{u}_s(k)\right)^T \mathbf{W}_U\mathbf{U}_f(k) + \left(\mathbf{S}_{uI}\mathbf{u}_s(k)\right)^T \mathbf{W}_U \left(\mathbf{S}_{uI}\mathbf{u}_s(k)\right) \\ &= \mathbf{U}_f(k)^T \left(\mathbf{S}_u^T\mathbf{W}_X\mathbf{S}_u + \mathbf{W}_U\right)\mathbf{U}_f(k) + 2\left[\boldsymbol{\eta}(k)^T\mathbf{W}_X\mathbf{S}_u - \left(\mathbf{S}_{uI}\mathbf{u}_s(k)\right)^T \mathbf{W}_U\right]\mathbf{U}_f(k) + \\ &\quad \text{remaining terms not containing } \mathbf{U}_f(k) \end{aligned}$$

Thus, defining matrix H and vector F(k) as follows

$$\begin{aligned} \mathcal{H} &= 2\left(\mathbf{S}_u^T\mathbf{W}_X\mathbf{S}_u + \mathbf{W}_U\right) \\ \mathcal{F}(k) &= 2\left[\boldsymbol{\eta}(k)^T\mathbf{W}_X\mathbf{S}_u - \left(\mathbf{S}_{uI}\mathbf{u}_s(k)\right)^T \mathbf{W}_U\right]^T \\ \boldsymbol{\eta}(k) &= \mathbf{S}_x\widehat{\mathbf{x}}(k|k-1) + \mathbf{S}_e\widehat{\mathbf{e}}_f(k) - \mathbf{S}_{xI}\mathbf{x}_s(k) \end{aligned}$$

we can formulate QP problem equivalent to MPC-1 as follows

$$\begin{matrix} \arg Min \\ \mathcal{U}_f \end{matrix} \; \frac{1}{2}\mathbf{U}_f(k)^T\mathcal{H}\mathbf{U}_f(k) + \mathcal{F}(k)\mathbf{U}_f(k)$$

$$\begin{bmatrix} \mathbf{I}_{mq\times mq} \\ -\mathbf{I}_{mq\times mq} \end{bmatrix} \mathbf{U}_f(k) \leq \begin{bmatrix} \mathbf{S}_{uI}\mathbf{u}_H \\ -\mathbf{S}_{uI}\mathbf{u}_L \end{bmatrix}$$

Use Matlab function *quadprog* to implement MPC.

You are expected to implement this formulation using either linearized mechanistic model or state space model identified from data. The controller tuning parameters are

- Mechanistic model based MPC

$$\begin{aligned}
\mathbf{w}_x &= \mathbf{I}_{2\times 2} \;\; ; \; \mathbf{w}_u = \mathbf{I}_{2\times 2} \;\; ; \;\; p = 75 \;\; ; q = 5 \;\; ; \alpha = 0.9 \\
\mathbf{U}_L &= [0 \quad 0]^T; \quad \mathbf{U}_H = [100 \quad 100]^T; \\
\mathbf{u}_L &= \mathbf{U}_L - \mathbf{U}_s \quad ; \quad \mathbf{u}_H = \mathbf{U}_H - \mathbf{U}_s
\end{aligned}$$

- State space model identified from data

$$\begin{aligned}
\mathbf{w}_x &= \mathbf{C}^T\mathbf{C} \;\; ; \; \mathbf{w}_u = \mathbf{I}_{2\times 2} \;\; ; \;\; p = 75 \;\; ; q = 5 \;\; ; \alpha = 0.9 \\
\mathbf{U}_L &= [0 \quad 0]^T; \quad \mathbf{U}_H = [100 \quad 100]^T; \\
\mathbf{u}_L &= \mathbf{U}_L - \mathcal{U}_s \quad ; \quad \mathbf{u}_H = \mathbf{U}_H - \mathcal{U}_s
\end{aligned}$$

After finding $\mathbf{U}_f(k)$ using QP, extract $\mathbf{u}(k)$ from $\mathbf{U}_f(k)$, set

$$\mathcal{U}(k) = \mathbf{u}(k) + \mathcal{U}_s$$

and communicate $\mathcal{U}(k)$ to the plant.