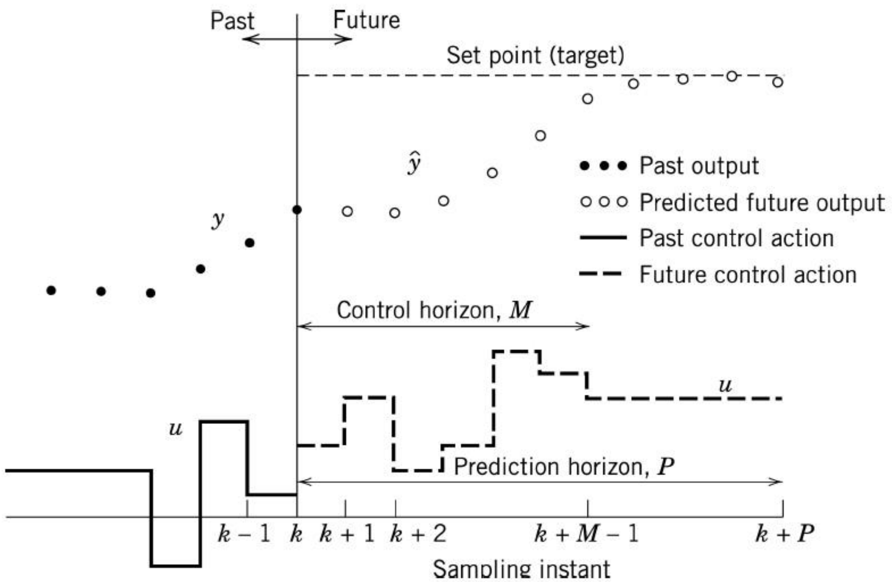
Modern Control Theory

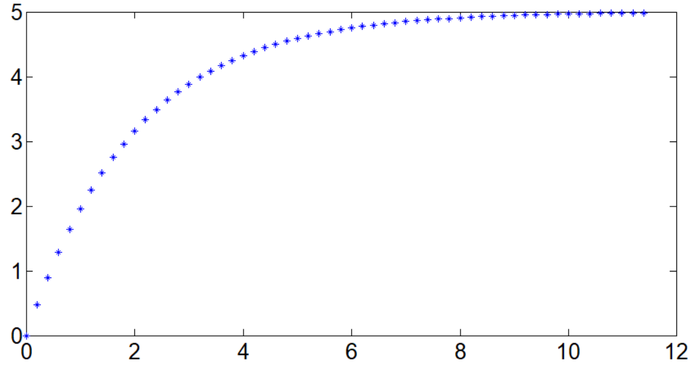
Model Predict Control - MPC – Lecture 4

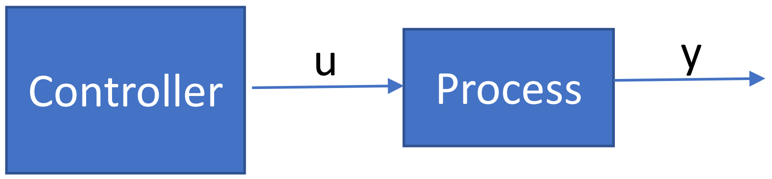
Dynamic Prediction Models

Why have multiple control moves?

- Example 5/(2s+1)

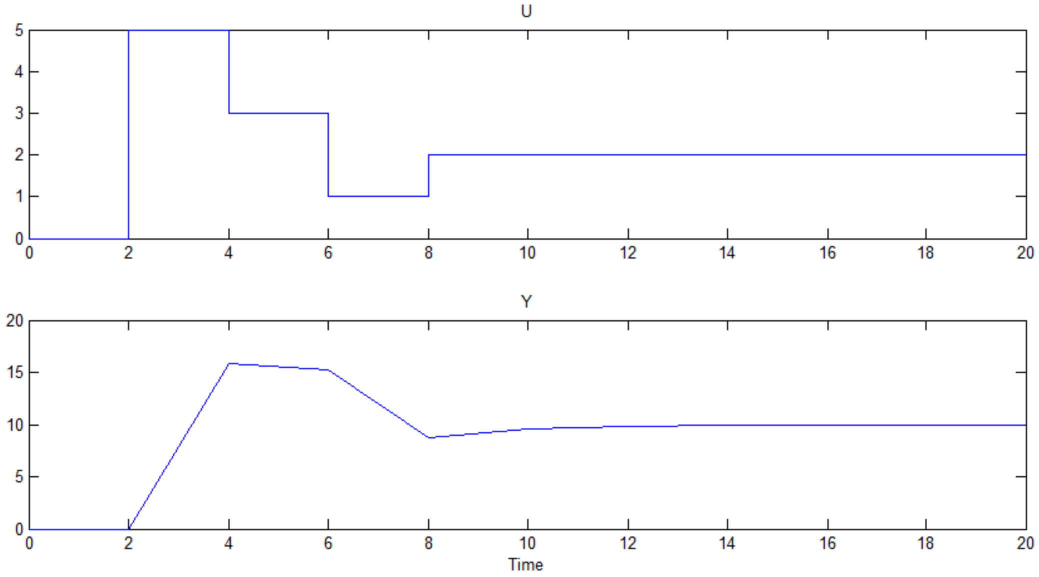


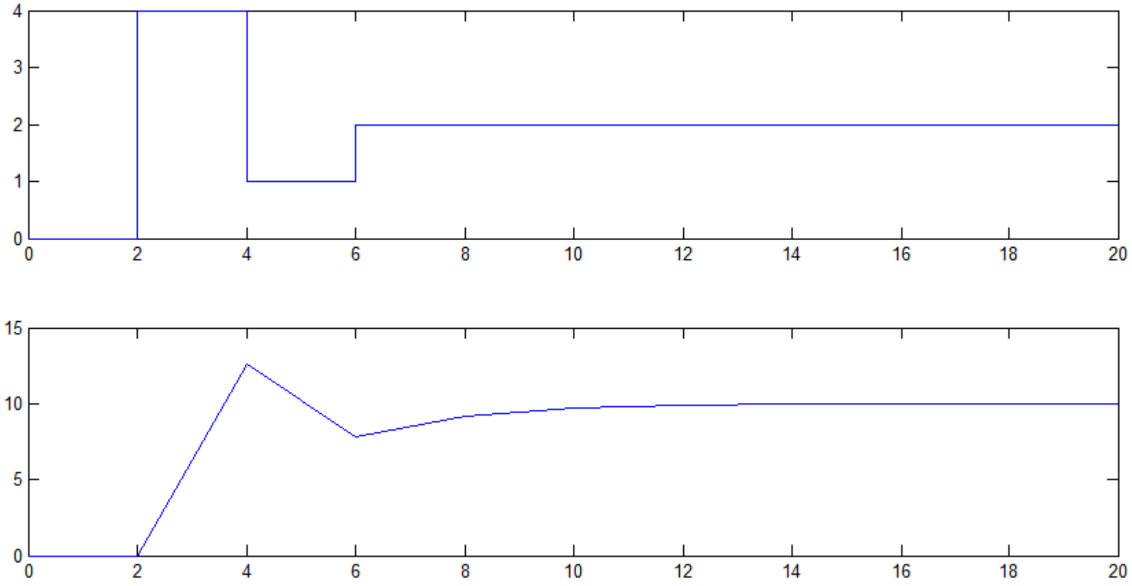




y → 10, what should be u?

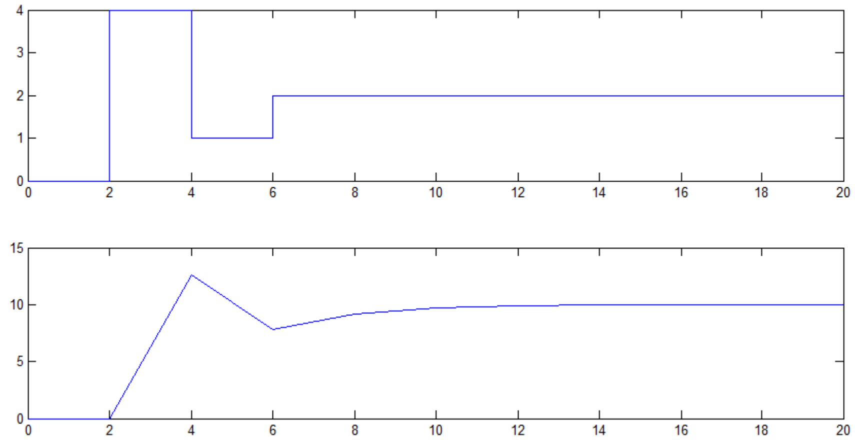
u = 2

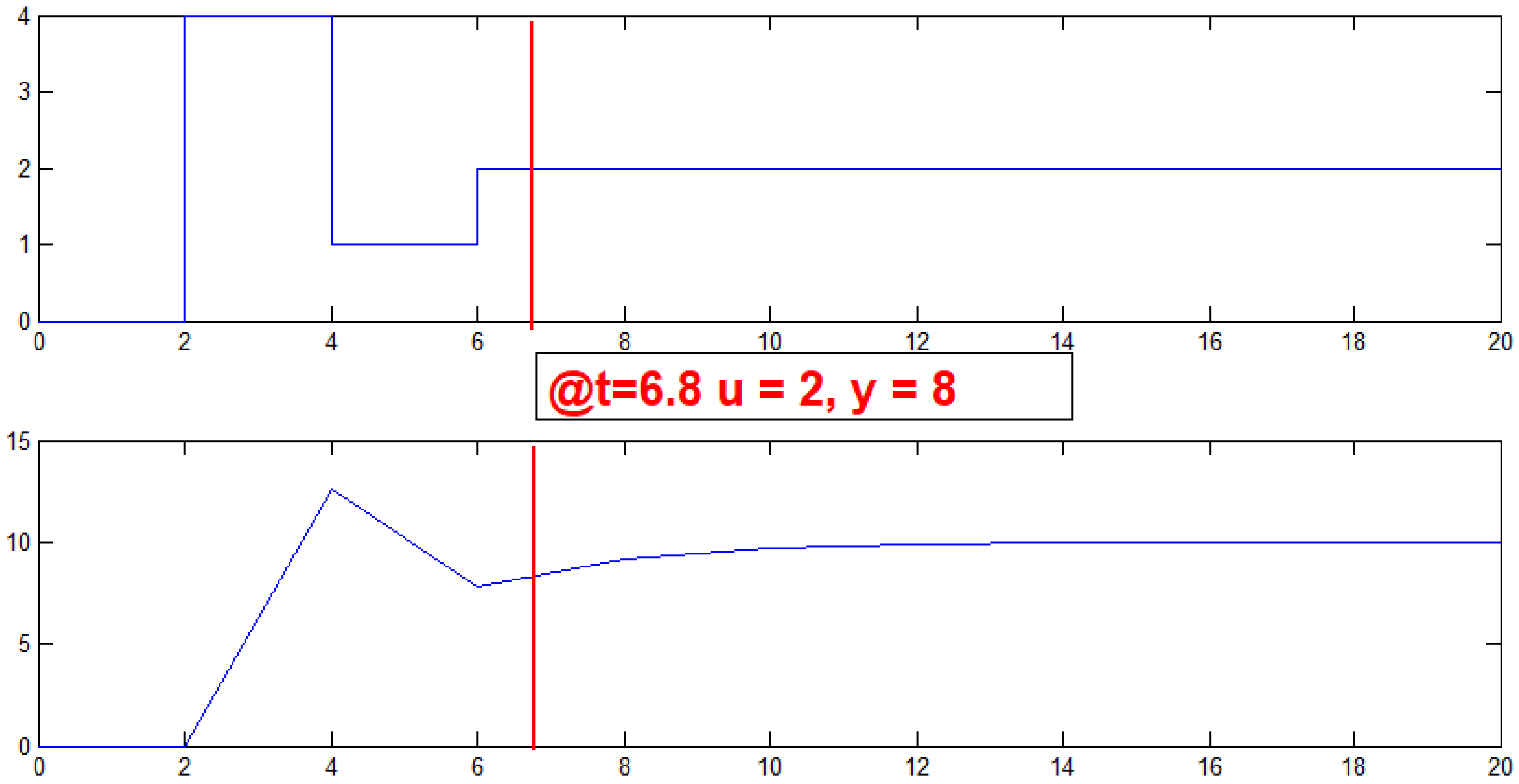


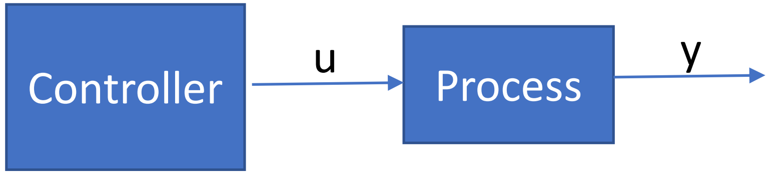


Does controller need to make control move all the time?

- Example 5/(2s+1)







y → 10

What should controller do?

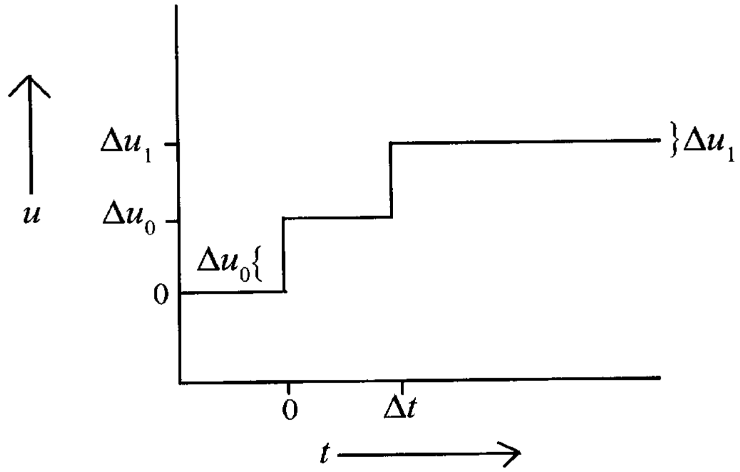
NO ACTION REQUIRED as it will reach to its Target

Free response and forced response

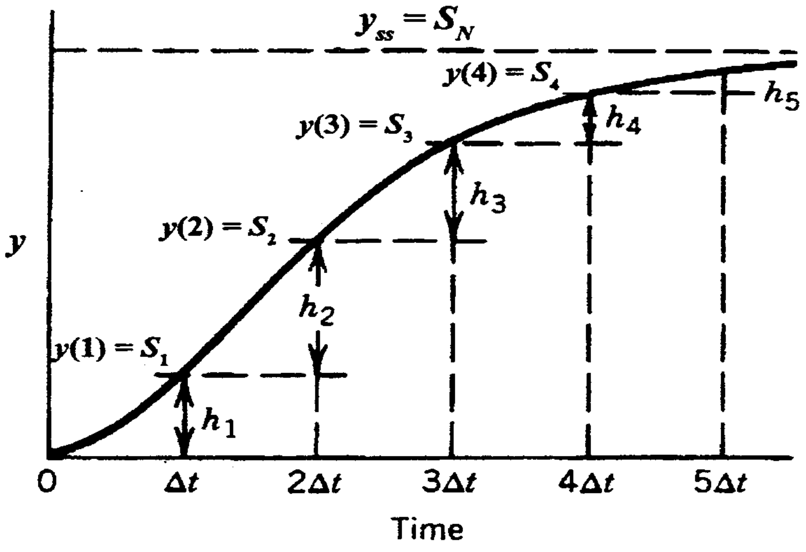
- Free response –
- Check if the system with current input can reach to the set-point or
- Find out how close it can take it to the set-point or objective

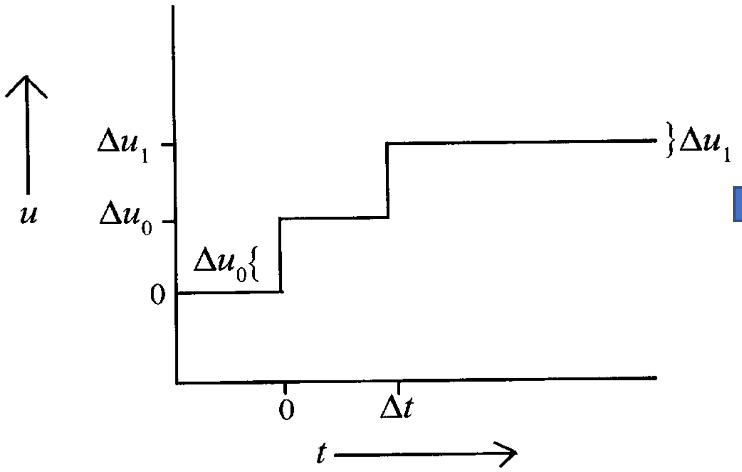
- Forced response –
- Don’t make any control action if the system can reach to the set-point or
- Compute control actions that can take it to the set-point or objective over and above the free response.

Multi move as breaking of multiple staggered step inputs



Step response predictions (1/2)





From the Principle of Superposition for linear systems:

$$y_1 = y_0 + S_1 \Delta u_0$$

$$y_2 = y_0 + S_2 \Delta u_0 + S_1 \Delta u_1$$

$$y_3 = y_0 + S_3 \Delta u_0 + S_2 \Delta u_1$$

. .

. .

$$y_N = y_0 + S_N \Delta u_0 + S_{N-1} \Delta u_1$$

The step response model of a stable SISO process

$$y(k+1) = \sum_{i=1}^{N} h_i \, u(k-i+1) = \sum_{i=1}^{N} \left(S_i - S_{i-1}\right) u(k-i+1)$$

$$\Rightarrow \quad y(k+1) = \sum_{i=1}^{N-1} S_i \, \Delta u(k-i+1) \; + \; S_N \, u(k-N+1)$$

S_i = the i-th step response coefficient

N = an integer (the **model horizon**)

$\Delta u(k) = u(k) - u(k-1)$

Step response prediction (2/2)

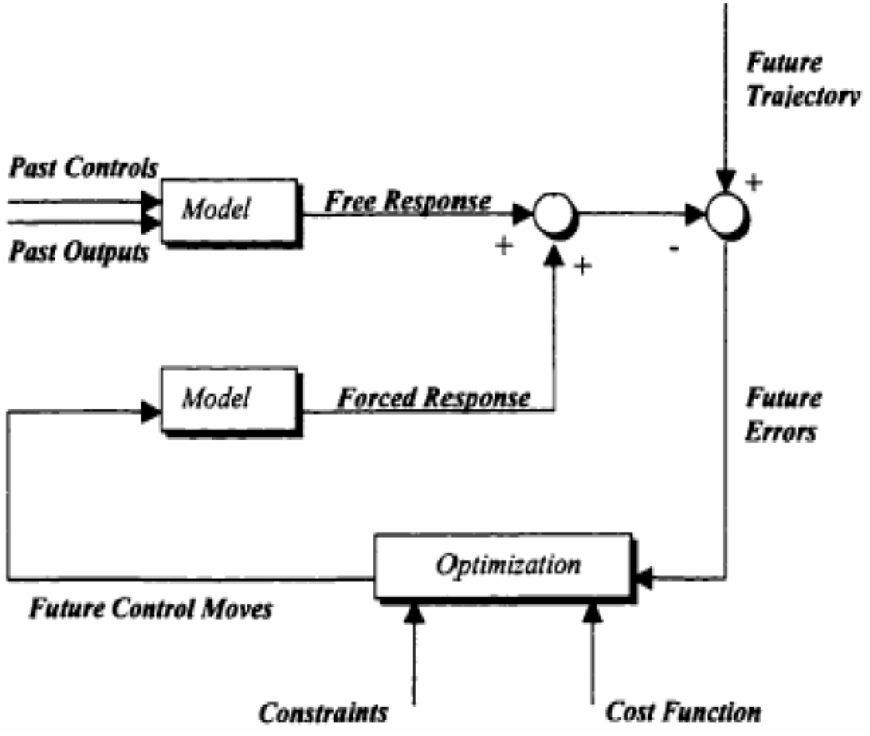
j[th] step ahead prediction

$$\hat{y}(k+j) = \underbrace{\sum_{i=1}^{j} S_i \Delta u(k+j-i)}_{\text{Effects of current and future control actions}} + \underbrace{\sum_{i=j+1}^{N-1} S_i \Delta u(k+j-i) + S_N u(k+j-N)}_{\text{Effects of past control actions}}$$

What happens when u(k-i) = u(k-1) for all i > 0

Predicted FREE response

$$\hat{y}^o(k+j) = \sum_{i=j+1}^{N-1} S_i \Delta u(k+j-i) + S_N u(k+j-N)$$



Example – simple predictive receding horizon controller

- Find a predictive control law with single receding horizon move, such that $\hat{y}(k+J) = y_{sp}$

j[th] step ahead prediction

$$\hat{y}(k+j) = \underbrace{\sum_{i=1}^{j} S_i \Delta u(k+j-i)}_{\text{Forced response}} + \underbrace{\hat{y}^o(k+j)}_{\text{Free response}}$$

Given

j = J, $\hat{y}(k+J) = y_{sp}$

$\Delta u(k + i) = 0$ for $i > 0$, i.e. $\Delta u(k + 1) = \Delta u(k + 2) .... = \Delta u(k + J\text{-}1) = 0$

⬇

$$\hat{y}(k+J) = S_J \Delta u(k) + \hat{y}^o(k+J)$$

⬇

$$y_{sp} = S_J \Delta u(k) + \hat{y}^o(k+J)$$

⬇

$$\Delta u(k) = \frac{y_{sp} - \hat{y}^o(k+J)}{S_J}$$

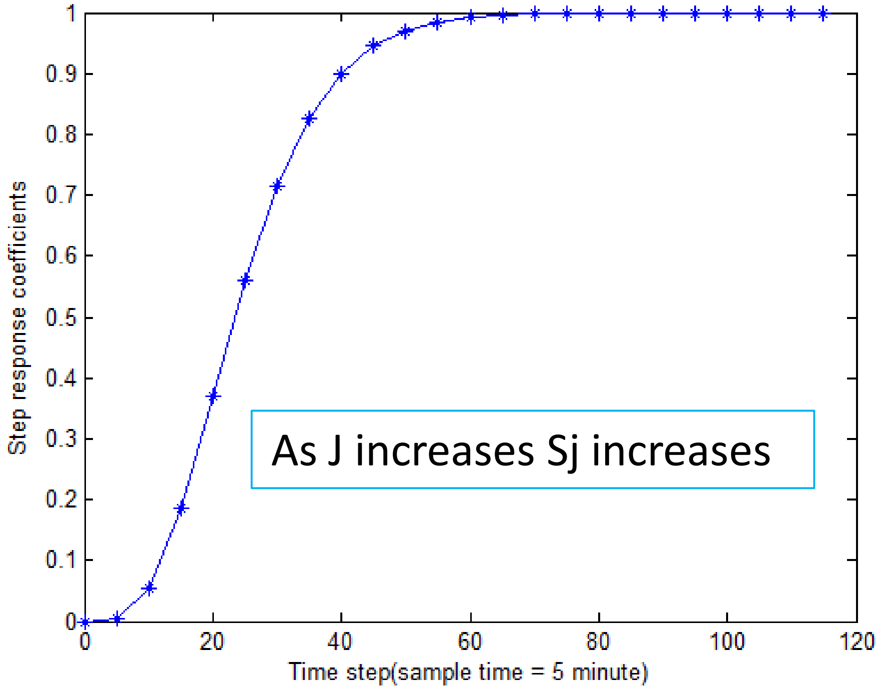
SISO system with prediction-based control law using simple step response coefficients

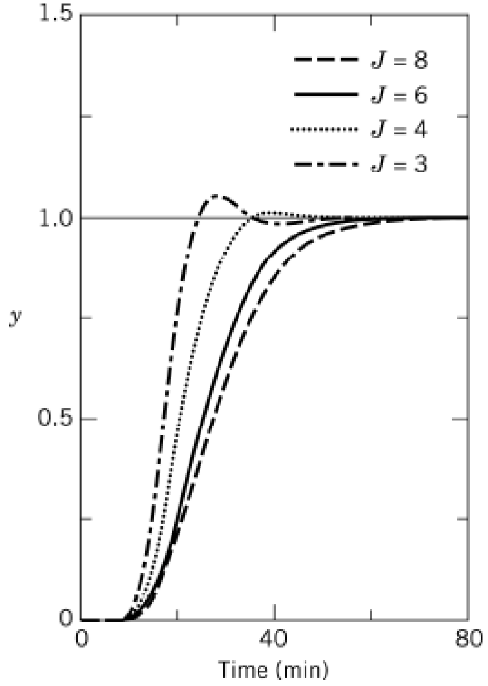
What are the pitfalls in this?

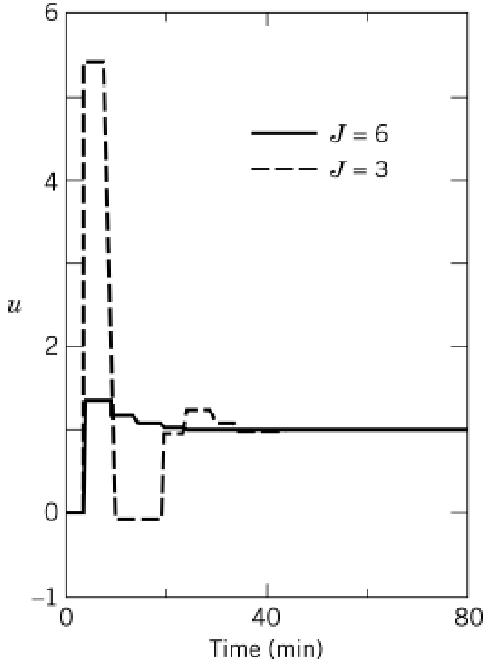
Example – simple predictive receding horizon controller

$$\frac{Y(s)}{U(s)} = \frac{1}{(5s + 1)^5}$$

J = 3, 4, 6, 8
Δt = 5 mins







$$\Delta u(k) = \frac{y_{sp} - \hat{y}^o(k + J)}{S_J}$$

The move required tend to be smaller as J increases

As J decreases, y becomes aggressive

Multi input Multi output prediction models - step response (1/2)

$$\hat{y}(k+1) = \underbrace{S_1\Delta u(k)}_{\text{Effect of current}} + \underbrace{\sum_{i=2}^{N-1} S_i\Delta u(k-i+1) + S_N\, u(k-N+1)}$$

$$\hat{y}(k+2) = \underbrace{S_1\Delta u(k+1)}_{\substack{\text{Effect of future}\\ \text{control action}}} + \underbrace{S_2\Delta u(k)}_{\substack{\text{Effect of current}\\ \text{control action}}} + \underbrace{\sum_{i=3}^{N-1} S_i\Delta u(k-i+2) + S_N\, u(k-N+2)}_{\text{Effect of past control actions}}$$

$$\hat{y}(k+j) = \underbrace{\sum_{i=1}^{j} S_i\Delta u(k+j-i)}_{\text{Forced response}} + \underbrace{\hat{y}^o(k+j)}_{\text{Free response}}$$

$$\begin{Bmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \hat{y}(k+j) \end{Bmatrix} = \begin{Bmatrix} S_1 & 0 & \cdots & 0 \\ S_2 & S_1 & 0 & \vdots \\ \vdots & \vdots & \ddots & 0 \\ S_j & S_{j-1} & & S_1 \end{Bmatrix} \begin{Bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \\ \Delta u(k+j-1) \end{Bmatrix} + \begin{Bmatrix} \hat{y}^o(k+1) \\ \hat{y}^o(k+2) \\ \hat{y}^o(k+j) \end{Bmatrix}$$

$$\hat{\mathbf{Y}}(k+1) = S\Delta\mathbf{U}(k) + \hat{\mathbf{Y}}^{\mathbf{o}}(k+1)$$

Multi input Multi output prediction models - step response (2/2)

Principle of superposition: Example 2 inputs and 2 outputs

$$\hat{y}_1(k+1) = \sum_{i=1}^{N-1} S_{11,i}\Delta u_1(k-i+1) + S_{11,N}\, u_1(k-N+1) + \sum_{i=1}^{N-1} S_{12,i}\Delta u_2(k-i+1) + S_{12,N}\, u_2(k-N+1)$$

$$\hat{y}_2(k+1) = \sum_{i=1}^{N-1} S_{21,i}\Delta u_1(k-i+1) + S_{21,N}\, u_1(k-N+1) + \sum_{i=1}^{N-1} S_{22,i}\Delta u_2(k-i+1) + S_{22,N}\, u_2(k-N+1)$$

P step ahead predictions for M future control moves

'r' inputs $\dot{\mathbf{u}} = [u_1 \quad u_2 \quad \cdots \quad u_r]^T$

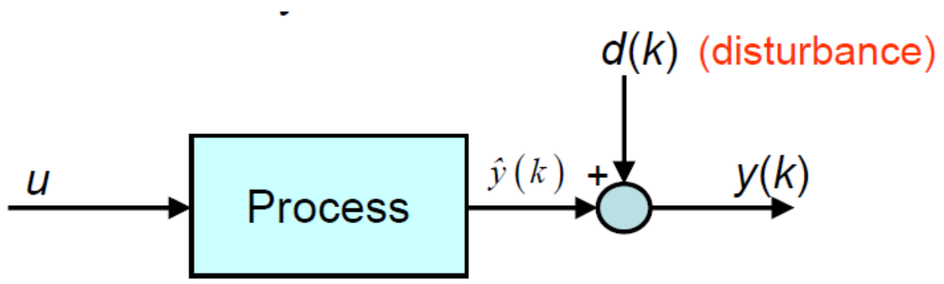
'm' outputs $\mathbf{y} = [y_1 \quad y_2 \quad \cdots \quad y_m]^T$

$$\hat{Y}(k+1) \triangleq \begin{bmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \vdots \\ \hat{y}(k+P) \end{bmatrix}_{mP\times 1} \quad \hat{Y}^o(k+1) \triangleq \begin{bmatrix} \hat{\mathbf{y}}^o(k+1) \\ \hat{\mathbf{y}}^o(k+2) \\ \vdots \\ \hat{\mathbf{y}}^o(k+P) \end{bmatrix}_{mP\times 1} \quad \Delta U(k) \triangleq \begin{bmatrix} \Delta\mathbf{u}(k) \\ \Delta\mathbf{u}(k+1) \\ \vdots \\ \Delta\mathbf{u}(k+M\text{-}1) \end{bmatrix}_{rM\times 1}$$

$$\boxed{\hat{Y}(k+1) = \underbrace{S\Delta U(k)}_{\text{Forced response}} + \underbrace{\hat{Y}^o(k+1)}_{\text{Free response}}}$$

$$S \triangleq \begin{bmatrix} S_1 & 0 & \cdots & 0 \\ S_2 & S_1 & 0 & \vdots \\ \vdots & \vdots & \ddots & 0 \\ S_M & S_{M-1} & \cdots & S_1 \\ S_{M+1} & S_M & \cdots & S_2 \\ \vdots & \vdots & \ddots & \vdots \\ S_P & S_{P-1} & \cdots & S_{P-M+1} \end{bmatrix} \quad S_i \triangleq \begin{bmatrix} S_{11,i} & S_{12,i} & \cdots & S_{1r,i} \\ S_{21,i} & \cdots & \cdots & S_{2r,i} \\ \vdots & \vdots & \vdots & \vdots \\ S_{m1,i} & \cdots & \cdots & S_{mr,i} \end{bmatrix}$$

Correcting for model prediction errors – output feedback



When can predictions drift away actual?

- Inaccurate model
- Unmeasured disturbances

How do we correct the model predictions?

Output feedback based on the latest measurement

$$\tilde{y}(k+j) \triangleq \hat{y}(k+j) + \underbrace{\left[y(k) - \hat{y}(k) \right]}_{\text{Bias term or correction term or Estimated disturbance}}$$

MIMO Model prediction with bias correction

'r' inputs $\mathbf{u} = [u_1 \quad u_2 \quad \cdots \quad u_r]^T$

'm' outputs $\mathbf{y} = [y_1 \quad y_2 \quad \cdots \quad y_m]^T$

$$\tilde{Y}(k+1) = S\Delta U(k) + \hat{Y}^o(k+1) + \mathbf{\Phi}\left[\mathbf{y}(k) - \hat{\mathbf{y}}(k) \right]$$

$\mathbf{\Phi}$ is matrix of '1' with dimension mP xm

Nomenclature

- CVs, MVs, and DVs