Coincidence Point Control

Goal: Find the set of manipulated inputs that force the output to be equal to the setpoint in P time steps

- Three different horizon-based solutions
  - Single control move
  - Min sum of squares of control action
  - Min sum of squares of control increments
- Incorporation into feedback formulation
  - Open-loop model state predictions, with output updates/ corrections based on output measurements

B. Wayne Bequette

- Predict output P steps into the future, by adjusting P control moves, assuming x_k is known
- The first and second steps are

$$x_{k+1} = \Phi x_k + \Gamma u_k$$

$$y_{k+1} = C x_{k+1} = C\Phi x_k + C\Gamma u_k$$

$$x_{k+2} = \Phi x_{k+1} + \Gamma u_{k+1} = \Phi\left[\Phi \mathrm{x}_\mathrm{k} + \Gamma \mathrm{u}_\mathrm{k}\right] + \Gamma \mathrm{u}_{\mathrm{k+1}}$$

$$= \Phi^2 x_k + \Phi\Gamma u_k + \Gamma u_{k+1}$$

$$y_{k+2} = C x_{k+2} = C\Phi^2 x_k + C\Phi\Gamma u_k + C\Gamma u_{k+1}$$

- Continuing for P steps, we find

$$y_{k+P} = C\Phi^P x_k + \sum_{i=1}^{P} C\Phi^{P-i}\Gamma u_{k+i-1}$$

$$y_{k+P} = C\Phi^P x_k + \sum_{i=1}^{P} C\Phi^{P-i}\Gamma u_{k+i-1}$$

- **Write this in matrix-vector form, with a vector of the manipulated inputs**

$$\begin{bmatrix} u_k \\ \vdots \\ u_{k+P-1} \end{bmatrix}$$

- This can be written in matrix-vector form as

$$y_{k+P} = C\Phi^{p} x_k + \begin{bmatrix} C\Phi^{p-1}\Gamma & \cdots & C\Gamma \end{bmatrix} \begin{bmatrix} u_k \\ \vdots \\ u_{k+P-1} \end{bmatrix}$$

- Rearranging

$$\begin{bmatrix} C\Phi^{p-1}\Gamma & \cdots & C\Gamma \end{bmatrix} \begin{bmatrix} u_k \\ \vdots \\ u_{k+P-1} \end{bmatrix} = y_{k+P} - C\Phi^{p} x_k$$

- And setting the output = setpoint at step P

$$\begin{bmatrix} C\Phi^{p-1}\Gamma & \cdots & C\Gamma \end{bmatrix} \begin{bmatrix} u_k \\ \vdots \\ u_{k+P-1} \end{bmatrix} = \underbrace{r_{k+P} - C\Phi^{p} x_k}$$

$$Ax = b$$

- Case 1: Assume all manipulated inputs are equal

$$y_{k+P} = C\Phi^{P} x_k + \sum_{i=1}^{P} C\Phi^{P-i}\Gamma u_{k+i-1}$$

And, since $u_{k+P-1} = u_{k+P-2} = \ldots = u_{k+1} = u_k$

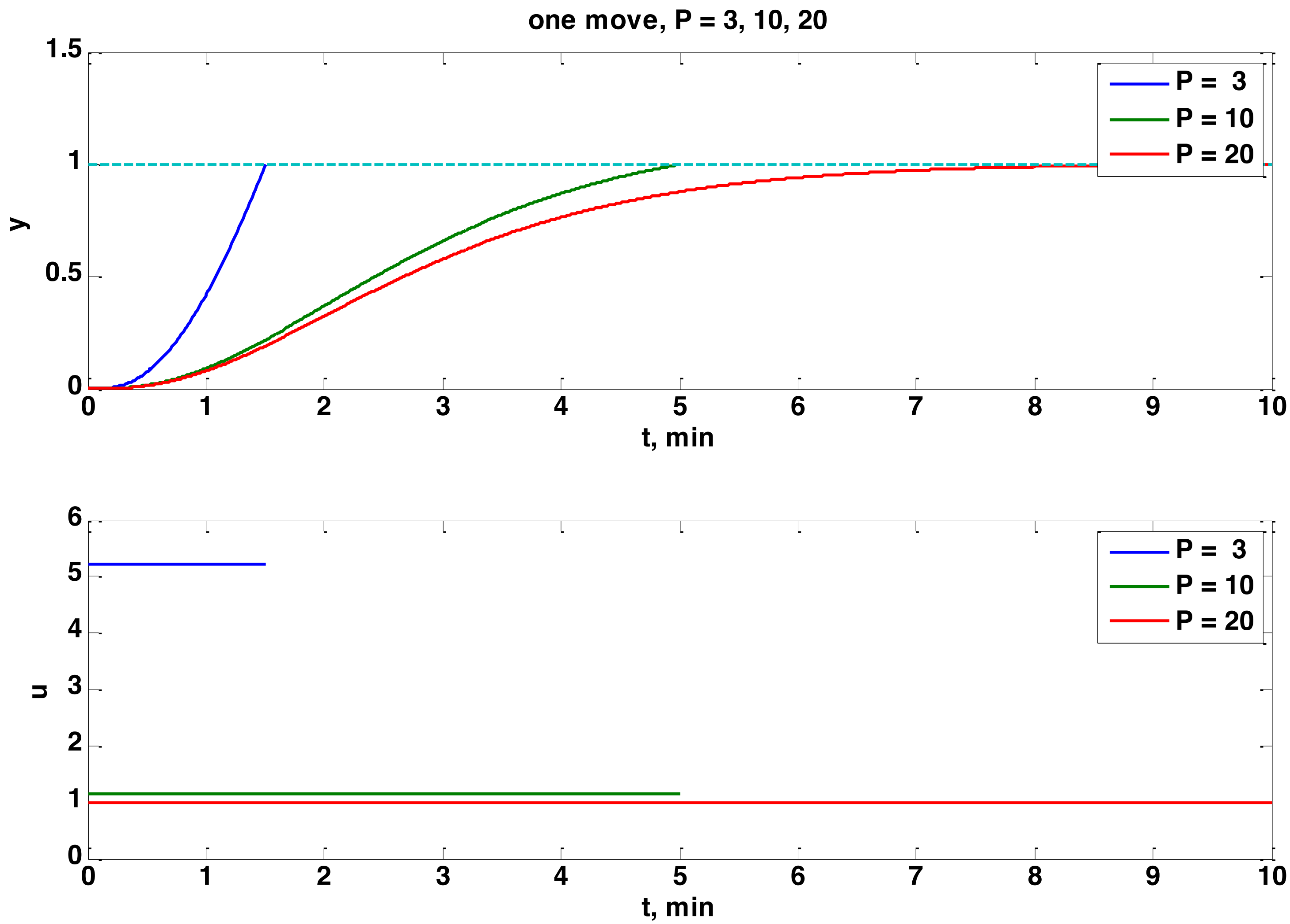
$$r_{k+P} = y_{k+P} = C\Phi^{P} x_k + \left(\sum_{i=1}^{P} C\Phi^{P-i}\Gamma\right) u_k$$

Solving for the input

$$u_k = \frac{r_{k+P} - C\Phi^{P} x_k}{\sum_{i=1}^{P} C\Phi^{P-i}\Gamma}$$

At large P:
setpoint / process gain

Three-tank Example, P = 3, 10 & 20 (sample time = 0.5 minutes)



- Case 2: Minimize sum-of-squares of inputs

$$\min_{u_{k+i-1}} \sum_{i=1}^{P} u_{k+i-1}^2 = \min_{u} u^T u$$

s.t. $$\underbrace{\begin{bmatrix} C\Phi^{p-1}\Gamma & \cdots & C\Gamma \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} u_k \\ \vdots \\ u_{k+P-1} \end{bmatrix}}_{x} = \underbrace{r_{k+P} - C\Phi^p x_k}_{b}$$

Form of $$Ax = b$$

$$\min_x x^T x$$

$$s.t.\, Ax = b$$

Solution $$x = A^T (AA^T)^{-1} b$$

Next Case (Δu)

- Many model predictive control strategies are based on using the changes in control action, so the following slides derive output predictions as a function of the control changes (Δu)

- Case 3: Minimize sum-of-squares of input changes (Δu)

Formulate in terms of Δu

$$\mathrm{x}_{\mathrm{k+1}} = \Phi x_k + \Gamma u_k = \Phi x_k + \Gamma\left(u_{k-1} + \Delta u_k\right) = \Phi x_k + \Gamma u_{k-1} + \Gamma \Delta u_k$$

$$y_{k+1} = C x_{k+1} = C\Phi x_k + C\Gamma u_{k-1} + C\Gamma\Delta u_k$$

$$\begin{aligned} x_{k+2} &= \Phi x_{k+1} + \Gamma u_{k+1} \\ &= \Phi^2 x_k + \Phi\Gamma u_k + \Gamma u_{k+1} = \Phi^2 x_k + \Phi\Gamma\left(u_{k-1} + \Delta u_k\right) + \Gamma\left(u_{k-1} + \Delta u_k + \Delta u_{k+1}\right) \\ &= \Phi^2 x_k + \left(\Phi\Gamma + \Gamma\right) u_{k-1} + \left(\Phi\Gamma + \Gamma\right)\Delta u_k + \Gamma u_{k+1} \end{aligned}$$

$$y_{k+2} = C\Phi x_{k+2} = C\Phi^2 \mathrm{x}_{\mathrm{k}} + \left(\mathrm{C}\Phi\Gamma + \mathrm{C}\Gamma\right)\mathrm{u}_{\mathrm{k-1}} + \left(\mathrm{C}\Phi\Gamma + \mathrm{C}\Gamma\right)\Delta\mathrm{u}_{\mathrm{k}} + C\Gamma\Delta u_{k+1}$$

$$x_{k+P} = \Phi^P x_k + \left(\sum_{i=1}^{P}\Phi^{i-1}\Gamma\right)u_{k-1} + \left(\sum_{i=1}^{P}\Phi^{i-1}\Gamma\right)\Delta u_k + \left(\sum_{i=1}^{P-1}\Phi^{i-1}\Gamma\right)\Delta u_{k-1} + \cdots + \Gamma\Delta u_{k+P-1}$$

$$y_{k+P} = C\Phi^P x_k + \left(\sum_{i=1}^{P}C\Phi^{i-1}\Gamma\right)u_{k-1} + \left(\sum_{i=1}^{P}C\Phi^{i-1}\Gamma\right)\Delta u_k + \left(\sum_{i=1}^{P-1}C\Phi^{i-1}\Gamma\right)\Delta u_{k-1} + \cdots + C\Gamma\Delta u_{k+P-1}$$

The summation terms are step response coefficients

$$S_1 = C\Gamma$$

$$S_2 = C\Phi\Gamma + C\Gamma$$

$$\vdots$$

$$S_{P-1} = \left(\sum_{i=1}^{P-1} C\Phi^{i-1}\Gamma \right)$$

$$S_P = \left(\sum_{i=1}^{P} C\Phi^{i-1}\Gamma \right)$$

The output predictions can be written

$$y_{k+P} = C\Phi^P x_k + S_p u_{k-1} + S_P \Delta u_k + S_{P-1}\Delta u_{k+1} + \cdots + S_1 \Delta u_{k+P-1}$$

Think of "free" (if no new input changes are made) and "forced" responses (effect of input changes)

$$y_{k+P} = \underbrace{C\Phi^P x_k + S_p u_{k-1}}_{\text{free response}} + \underbrace{S_P \Delta u_k + S_{P-1}\Delta u_{k+1} + \cdots + S_1 \Delta u_{k+P-1}}_{\text{forced response}}$$

Assume that our goal is to force the output at step k+P to be equal to the setpoint at step k+P, that is, $y_{k+P} = r_{k+P}$. Then we are solving for the set of control moves that satisfy the following equation

$$S_P \Delta u_k + S_{P-1} \Delta u_{k+1} + \cdots + S_1 \Delta u_{k+P-1} = r_{k+P} - C\Phi^P x_k - S_p u_{k-1}$$

$$\begin{bmatrix} S_P & S_{P-1} & \cdots & S_1 \end{bmatrix} \begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \vdots \\ \Delta u_{k+P-1} \end{bmatrix} = r_{k+P} - C\Phi^P x_k - S_p u_{k-1}$$

For a single input, single output system, we note the following dimensions

$$\underbrace{\begin{bmatrix} S_P & S_{P-1} & \cdots & S_1 \end{bmatrix}}_{1xP} \underbrace{\begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \vdots \\ \Delta u_{k+P-1} \end{bmatrix}}_{Px1} = \underbrace{r_{k+P} - C\Phi^P x_k - S_p u_{k-1}}_{1x1}$$

So this is an over-determined problem, requiring the notion of a “generalized” inverse. Writing this expression in the following form

$$S_f \Delta u_f = E$$

$$\underbrace{\begin{bmatrix} S_P & S_{P-1} & \cdots & S_1 \end{bmatrix}}_{1xP} \underbrace{\begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \vdots \\ \Delta u_{k+P-1} \end{bmatrix}}_{Px1} = \underbrace{r_{k+P} - C\Phi^P x_k - S_p u_{k-1}}_{1x1}$$

"Unforced Error" (error if no Manipulated input changes are made

$$S_f \Delta u_f = E$$

$$\min \left\| \Delta u_f \right\|_2$$

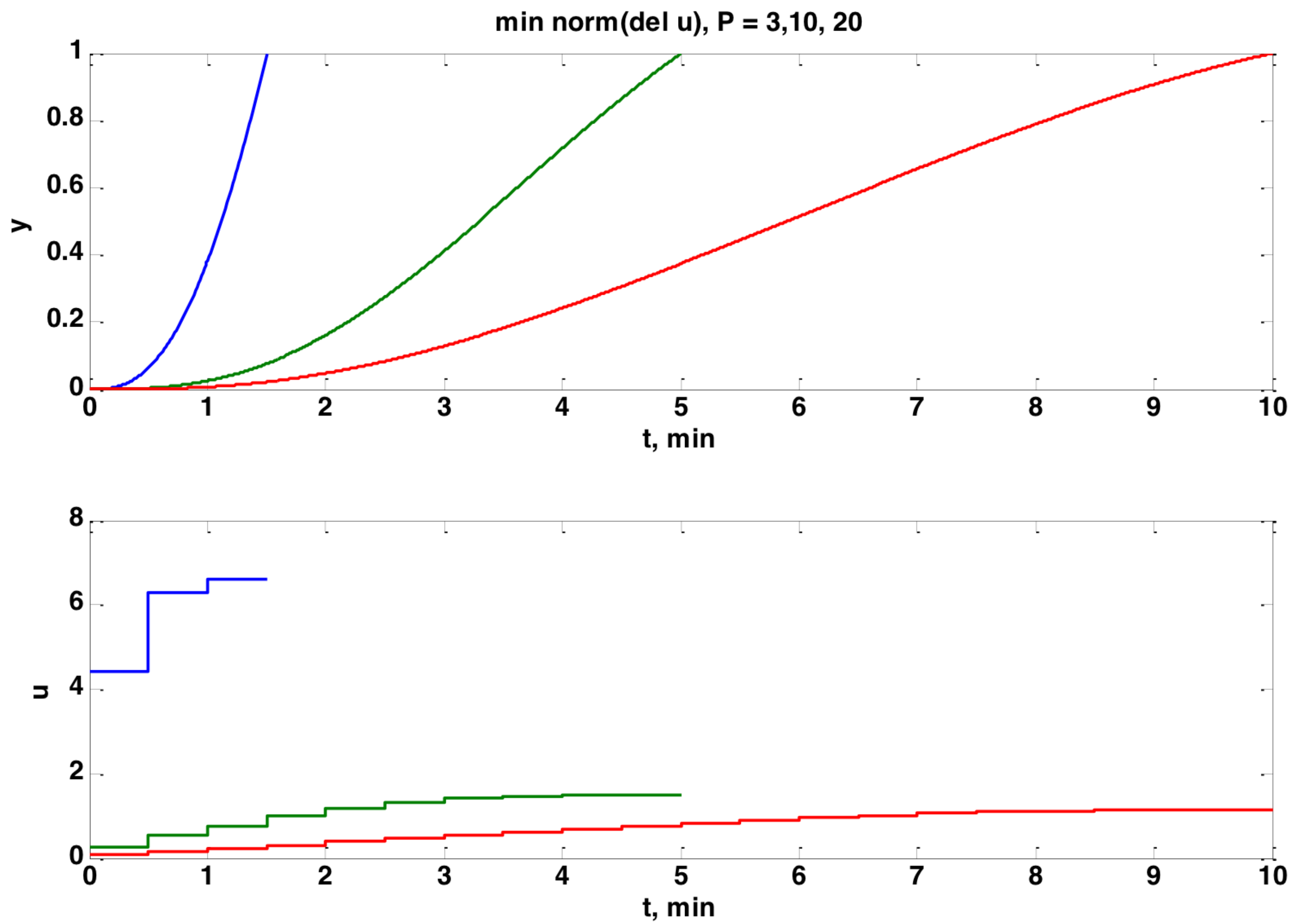
$$s.t. \quad S_f \Delta u_f = E$$

$$\left\| \Delta u_f \right\|_2 = \sum_{i=k}^{i=k+P-1} \Delta u_i^2$$

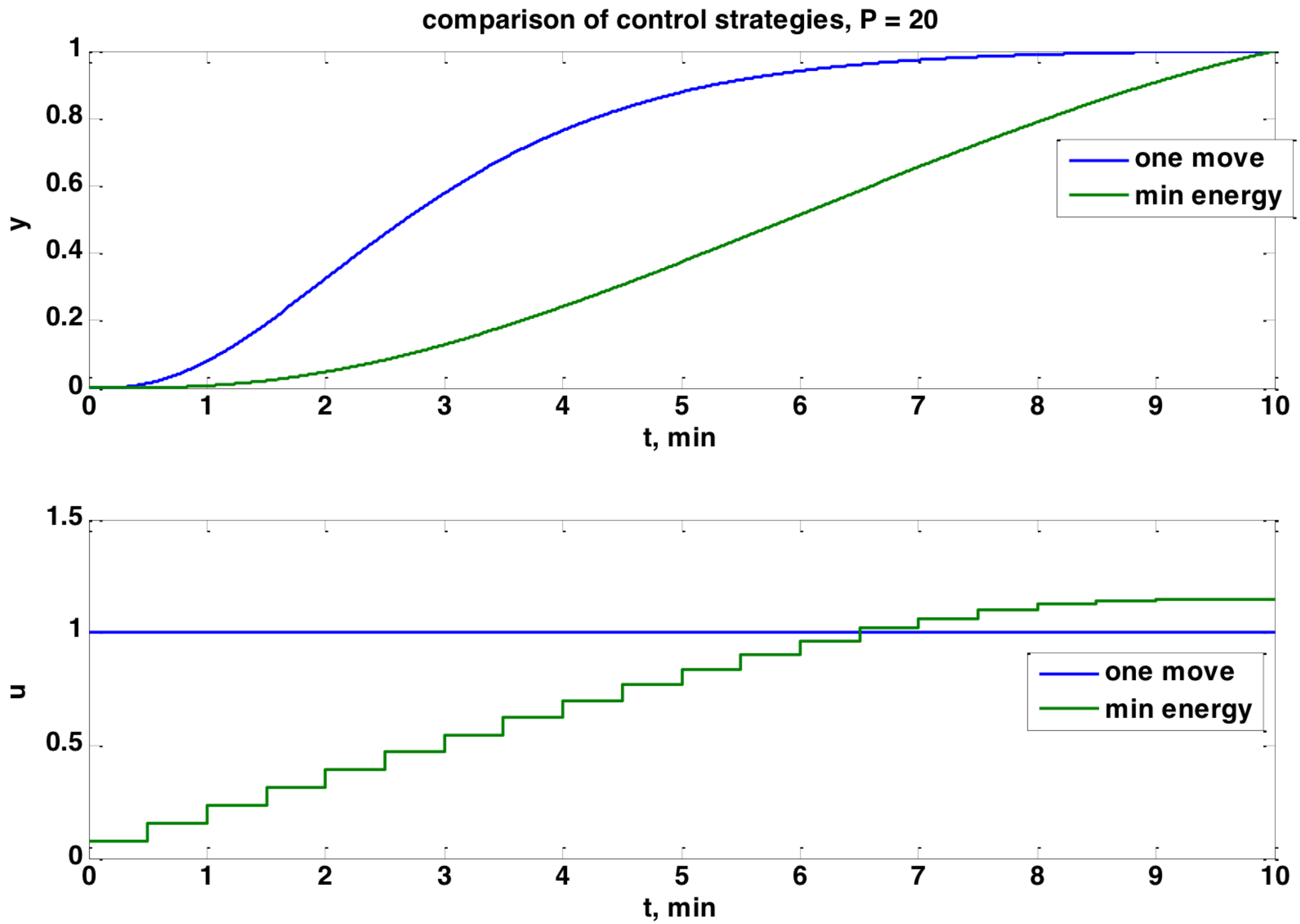
Analytical solution

$$\Delta u_f = S_f^T \left(S_f S_f^T \right)^{-1} E$$

Three-tank Example, P = 3, 10 & 20
(sample time = 0.5 minutes)



Three-tank Example, P = 3, 10 & 20
Comparison of one move vs. minimum effort



Option: Control Horizon less than Prediction Horizon (M<P)

$$\underbrace{\begin{bmatrix} S_P & S_{P-1} & \cdots & S_{P-M+1} \end{bmatrix}}_{1xM} \underbrace{\begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \vdots \\ \Delta u_{k+M-1} \end{bmatrix}}_{Mx1} = \underbrace{r_{k+P} - C\Phi^P x_k - S_p u_{k-1}}_{1x1}$$

"Unforced Error" (error if no Manipulated input changes are made

$$S_f \Delta u_f = E$$

$$\min \left\| \Delta u_f \right\|_2$$

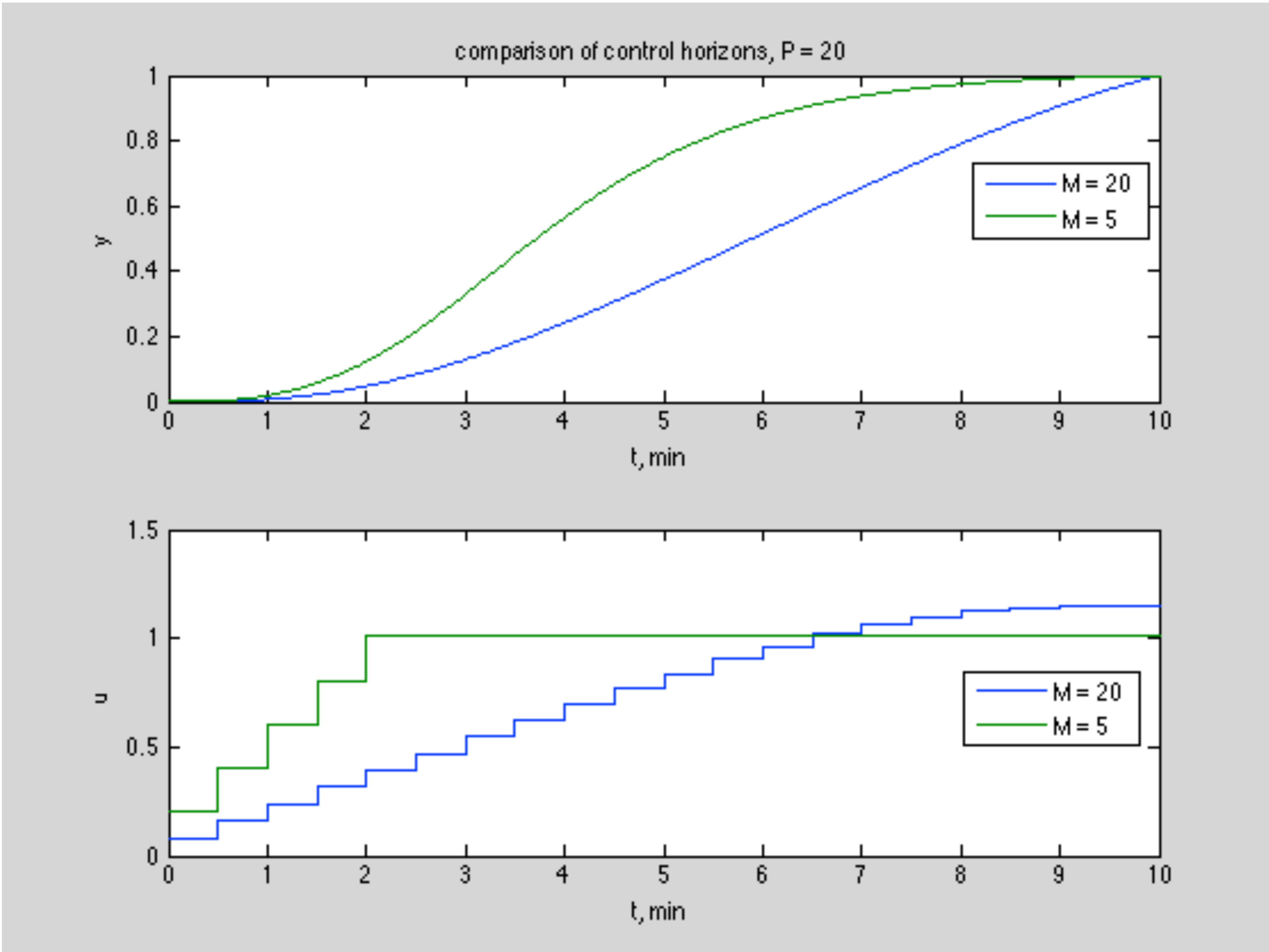
$$s.t. \quad S_f \Delta u_f = E$$

$$\left\| \Delta u_f \right\|_2 = \sum_{i=k}^{i=k+P-1} \Delta u_i^2$$

Analytical solution

$$\Delta u_f = S_f^T \left(S_f S_f^T \right)^{-1} E$$

Three-tank Example, P = 20, M = 20 or 5



Pre-Summary

- Coincidence point (achieve a setpoint P steps into the future)
- Single move (case 1)
- Minimum energy/effort (for Δu)(case 3)
- Did not show simulation results for minimizing the 2-norm of u (case 2)
- Thus far we have solved "open-loop" problems, and assumed a perfect model.
- **Extension to closed-loop is shown on the next slides**

Model predictions and updates based on measured output

$\hat{x}_1 = \Phi \hat{x}_0 + \Gamma u_0$ Start with initial condition assumption

$\hat{x}_k = \Phi \hat{x}_{k-1} + \Gamma u_{k-1}$ Update model state at each time step, using previous input

$\hat{y}_k = C \hat{x}_k$ Model output based on model state

y_k Plant output measurement

$\hat{d}_k = y_k - \hat{y}_k$ Plant-model mismatch (additive disturbance)

$\hat{d}_{k+P} = \hat{d}_{k+P-1} = \cdots = \hat{d}_k$ Future plant-model mismatch assumed constant

$$\hat{y}^c_{k+P} = \underbrace{C\Phi^P \hat{x}_k + S_p u_{k-1} + \hat{d}_k}_{\text{free response}} + \underbrace{S_P \Delta u_k + S_{P-1} \Delta u_{k+1} + \cdots + S_1 \Delta u_{k+P-1}}_{\text{forced response}}$$

Corrected output prediction

$$\hat{y}_{k+P|k} = \underbrace{C\Phi^P \hat{x}_k + S_p u_{k-1} + \hat{d}_k}_{\text{free response}} + \underbrace{S_P \Delta u_k + S_{P-1} \Delta u_{k+1} + \cdots + S_1 \Delta u_{k+P-1}}_{\text{forced response}}$$

Newer notation

Output prediction to step k+P, based on a measurement at step k

Calculation with model update based on measurement

$$\underbrace{\begin{bmatrix} S_P & S_{P-1} & \cdots & S_{P-M+1} \end{bmatrix}}_{1xM} \underbrace{\begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \vdots \\ \Delta u_{k+M-1} \end{bmatrix}}_{Mx1} = \underbrace{r_{k+P} - C\Phi^P \hat{x}_k - S_p u_{k-1} - \hat{d}_k}_{1x1}$$

Effect of plant-model mismatch

"Unforced Error" (error if no Manipulated input changes are made

$$S_f \Delta u_f = E$$

$$\min \left\| \Delta u_f \right\|_2$$

$$s.t. \quad S_f \Delta u_f = E$$

$$\left\| \Delta u_f \right\|_2 = \sum_{i=k}^{i=k+P-1} \Delta u_i^2$$

Analytical solution

$$\Delta u_f = S_f^T \left(S_f S_f^T \right)^{-1} E$$

Implement the first element of Δu_f $\quad \Delta u_k$

Which is applied to the plant as $\quad u_k = u_{k-1} + \Delta u_k$

Then, new optimization performed at the next time step, based on the new measurement

Three-tank Example: open-loop (applying all moves) vs. closed-loop (applying first move, then resolving at each time step)