Advanced Process Control
Computing Examination I (2.5 hours)

Consider linear perturbation model

$$\frac{d\mathbf{x}}{dt} = \begin{bmatrix} -\frac{1}{62} & 0 & \frac{1}{23} & 0 \\ 0 & -\frac{1}{90} & 0 & \frac{1}{30} \\ 0 & 0 & -\frac{1}{23} & 0 \\ 0 & 0 & 0 & -\frac{1}{30} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} \frac{7}{84} & 0 \\ 0 & \frac{1}{16} \\ 0 & \frac{1}{21} \\ \frac{1}{32} & 0 \end{bmatrix} \mathbf{u}(t)$$

$$\mathbf{y}(t) = \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \end{bmatrix} \mathbf{x}(t)$$

- **Step 1:** Convert the continuous time model into a discrete time model of the form

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k) \quad (1)$$
$$\mathbf{y}(k) = C\mathbf{x}(k) \quad (2)$$

where $\Phi = e^{\mathbf{A}T}$ and $\Gamma = [\Phi - \mathbf{I}]\mathbf{A}^{-1}\mathbf{B}$ with sampling interval of $T = 5$ units.

Note: MATLAB command to compute $e^{\mathbf{M}}$, where M represents a matrix, is '$expm(\mathbf{M})$'

- **Step 2: Generation of data for system identification**

Simulate the plant in open loop for k = 1,2,...500 samples using the following set of equations

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\left[\mathbf{u}(k) + \mathbf{w}(k)\right] \quad (3)$$
$$\mathbf{y}(k) = C\mathbf{x}(k) + \mathbf{v}(k) \quad (4)$$

Here, $\mathbf{w}(k)$ and $\mathbf{v}(k)$ are zero mean Gaussian white noise sequences with

$$Cov\left[\mathbf{w}(k)\right] = \begin{bmatrix} (0.06)^2 & 0 \\ 0 & (0.08)^2 \end{bmatrix} \quad \text{and} \quad Cov\left[\mathbf{v}(k)\right] = \begin{bmatrix} (0.05)^2 & 0 \\ 0 & (0.04)^2 \end{bmatrix}$$

Generate $\mathbf{u}(k)$ PRBS input signal using following sequence of MATLAB commands

uk = zeros(2,500) ;

Sk=sign(randn(2,50)) ;

jk = 0 ; uk1 = 0.5 ; uk2 = -0.6 ;

```
for k = 1 : 500
    if (rem(k,10) == 0)
   jk = jk + 1 ;
        uk1 = 0.5 * Sk(1,jk) ;   uk2 = 0.6 * Sk(2,jk) ;
   end
   uk(1,k) = uk1 ; uk(2,k) = uk2 ;
end
```

- **Step 3:** Using data generated in Step 2 for output 1, develop a 2'th order ARX model of the form

$$\begin{aligned} y(k) &= -a_1 y(k-1) - a_2 y(k-2) + b_1 u_1(k-1) + b_2 u_1(k-2) \\ &\quad + \beta_1 u_2(k-1) + \beta_2 u_2(k-2) + e(k) \end{aligned}$$

Parameters of this model can be identified using the following algorithm

$$\mathbf{M} = \begin{bmatrix} -y(2) & -y(1) & u_1(2) & u_1(1) & u_2(2) & u_2(1) \\ -y(3) & -y(2) & u_1(3) & u_1(2) & u_2(3) & u_2(2) \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ -y(N-1) & -y(N-2) & u_1(N-1) & u_1(N-2) & u_2(N-1) & u_2(N-2) \end{bmatrix}$$

$$\mathbf{Y} = \begin{bmatrix} y(3) & y(4) & \ldots & y(N-1) & y(N) \end{bmatrix}^T$$

$$\boldsymbol{\theta} = \begin{bmatrix} a_1 & a_2 & b_1 & b_2 & \beta_1 & \beta_2 \end{bmatrix}^T$$

$$\widehat{\boldsymbol{\theta}} = \left[\mathbf{M}^T\mathbf{M}\right]^{-1}\mathbf{M}^T\mathbf{Y}$$

- **Step 4: Graphical presentation of results**

Figure 1: Find predicted output as

$$\widehat{\mathbf{Y}} = \mathbf{M}\widehat{\boldsymbol{\theta}}$$

and compare vectors $\widehat{\mathbf{Y}}$ and $\mathbf{Y}$ in same figure.

Figure 2 and 3: Plot input sequences *uk(1,:)* and *uk(2,:)* using MATLAB function *'stairs'*

Figure 4: Compute model residual vector

$$\mathbf{E} = \mathbf{Y} - \widehat{\mathbf{Y}}$$

and plot it.