Indian Institute of Technology, Bombay — Chemical Engineering

CL686: Advanced Process Control — **Assignment 4, 2025**

**Date Due**: 30 March 2025, 11.59 PM (Note that time is not 12 PM midnight since that is confusing to set.)

**Mode of Submission**: Moodle submission

**Late Submission Policy**: Submissions late by 0-24 hrs will incur a 25% penalty, and submissions late by 24-48 hrs will incur a 50% penalty. Submissions later than 48 hrs will not be considered.

---

**Topic: State Estimation**

**AIM:** To implement pole placement based observer and Kalman Filter for Quadruple Tank System.

## Task at Hand

1. Implement Open Loop State Estimator (OLSE).
2. Implement Pole Placement based Observer (PPO).
3. Implement Kalman Filter using a steady-state solution of the Riccati equation.
4. Implement Kalman Filter by solving the Riccati equations forward in time.
5. Compare the performance of all of the above estimators in terms of Sum of squared error (SSE) value.
   **NOTE:** All estimators must be implemented considering nonlinear ODEs of Quadruple Tank System as the plant. The estimators will be based on linear-model though. Mat file containing the (linearized) matrices for the continuous-time model have been separately uploaded (file Continuous_time_linear_perturbation_model_without_disturbance.mat).

## Quadruple Tank System Description

Consider a nonlinear system governed by the following set of coupled nonlinear ODEs,

$$\frac{dh_1}{dt} = -\frac{a_1}{A_1}\sqrt{2gh_1} + \frac{a_3}{A_1}\sqrt{2gh_3} + \frac{\gamma_1 k_1}{A_1}v_1 \tag{1}$$

$$\frac{dh_2}{dt} = -\frac{a_2}{A_2}\sqrt{2gh_2} + \frac{a_4}{A_2}\sqrt{2gh_4} + \frac{\gamma_2 k_2}{A_2}v_2 \tag{2}$$

$$\frac{dh_3}{dt} = -\frac{a_3}{A_3}\sqrt{2gh_3} + \frac{(1-\gamma_2)k_2}{A_3}v_2 + \frac{\gamma_3}{A_3}F_d \tag{3}$$

$$\frac{dh_4}{dt} = -\frac{a_4}{A_4}\sqrt{2gh_4} + \frac{(1-\gamma_1)k_1}{A_4}v_1 + \frac{(1-\gamma_3)}{A_4}F_d \tag{4}$$

The initial steady-state operating conditions of the system are listed in the table below:

| **Variable** | **Optimum operating point** |
|---|---|
| $A_1, A_3$ $[cm^2]$ | 28 |
| $A_2, A_4$ $[cm^2]$ | 32 |
| $a_1, a_3$ $[cm^2]$ | 0.071 |
| $a_2, a_4$ $[cm^2]$ | 0.057 |
| $g$ $[cm/s^2]$ | 981 |
| $k_1, k_2$ $[cm^3/Vs]$ | 3.33, 3.35 |
| $\gamma_1, \gamma_2, \gamma_3$ | 0.7, 0.6, 0.4 |

- **System State:** $h_1, h_2, h_3, h_4$ are the liquid level in the quadruple tank system.
- **Process inputs:** $v_1$ and $v_2$ are the voltage applied to the pump.
- **Disturbance inputs:** $F_d$ acts as the disturbance input to the process. For this assignment, it will not be considered (i.e. its value will be kept at 0). [However, enterprising students can investigate its effect as an additional not-to-be-submitted exercise]

## Condition for Simulation:

- **Steady state inputs:** $\mathbf{U}_s = [3 \quad 3]^T$ ; $D_s = 0$
- **Equilibrium/Steady State Operating Point:**
  $\mathbf{X}_s = [12.263 \quad 12.7831 \quad 1.6339 \quad 1.409]^T$
- **Sampling interval**, $T = 4$ sec
- **Number of samples**, $N_s = 200$

### Nonlinear plant simulation: True state and measurement generation

- Initial state $\mathbf{X}(0) = \mathbf{X}_s$
- Input profile generated using the following expression:

$$u_1(k) = 2 \times sin\ (0.025\ k) + 0.15 \times cos\ (0.02\ k)$$
$$u_2(k) = 2 \times sin\ (0.02\ k) - 0.1 \times cos\ (0.025\ k)$$

  with $\mathbf{u}(k) = [u_1(k)\ u_2(k)]^T$ and $\mathbf{u}(k)$ is deviation inputs at $k^{th}$ instant.
- True state is generated by solving a nonlinear dynamical equation for above generated input profile using an ode-integrator such as explicit Euler or Runge-Kutta method. In Matlab, you can use inbuilt integrators such as ode45.

### Open Loop State Estimator (OLSE)

- In this implementation, refer to the class notes or lecture slides.

### Pole Placement based Observer (PPO)

- **Desired observer pole:** $\mathbf{p} = [0.7503 \quad 0.7655 \quad 0.6711 \quad 0.7004]^T$
- **Estimator gain matrix (L):** Estimator gain matrix ($\mathbf{L}$) can be obtained using MATLAB inbuilt command *place* as follows
  $\mathbf{L}^T = place(\mathbf{\Phi}^T, \mathbf{C}^T, \mathbf{p})$. Note that the place command will give you $\mathbf{L}^T$ and not $\mathbf{L}$. This function places the desired closed-loop poles $\mathbf{p}$ by computing an estimator gain matrix $\mathbf{L}$. Here $\mathbf{\Phi}$ and $\mathbf{C}$ are discrete-time states and measurement matrices.
  **NOTE:** Please refer to lecture notes for the detailed algorithm and MATLAB help for the above function.
- Initial state $\widehat{\mathbf{X}}(0) = \mathbf{X}_s + [1 \quad 1 \quad 1 \quad 1]^T$.

### Kalman Filter (KF)

1. Covariance matrix of measurement noise is $\mathbf{R} = 0.025^2 \times \mathbf{I}$
2. Covariance matrix of state noise is $\mathbf{Q} = 0.05^2 \times \mathbf{I}$
3. Initial state $\widehat{\mathbf{X}}(0|0) = \mathbf{X}_s + [1 \quad 1 \quad 1 \quad 1]^T$.
4. Initial covariance $\mathbf{P}(0|0) = 100 \times \mathbf{Q}$

**A: Steady State Estimator Gain**
In this implementation, use $\mathbf{L}_\infty$ as the estimator gain where $\mathbf{L}_\infty$ is the steady state solution of the Riccati equation.

- Solution for Algebraic Riccati Equation (ARE) can be found using MATLAB inbuilt function *kalman* as follows
  $[KEST, \mathbf{L}, \mathbf{P}, \mathbf{M}, \mathbf{Z}] = kalman(sys, \mathbf{Q}, \mathbf{R})$
  **NOTE:** Please refer to lecture notes for the detailed algorithm and MATLAB help for above function.

**Additional information:** The $sys$ is a state-space model representation, can be generated using the following sample MATLAB code:

$$\Gamma_w = eye(n_{st})$$
$$\Gamma_{u_y} = zeros(n_{op}, n_{ip})$$
$$\Gamma_{w_y} = zeros(n_{op}, n_{st})$$
$$sys = ss(\boldsymbol{\phi},\ [\Gamma\ \Gamma_w],\ \mathbf{C},\ [\Gamma_{u_y}\ \Gamma_{w_y}])$$

where, $n_{st}$, $n_{ip}$, and $n_{op}$ are the number of states, known inputs ($u$ variables), and outputs. $\boldsymbol{\phi}, \Gamma, \mathbf{C}$ are as before.

**B: Time Varying Estimator Gain Matrix**
In this implementation, solve for $\mathbf{P}(k|k-1), \mathbf{L}(k), \mathbf{P}(k|k)$, forward in time as discussed in class and slides.

### Performance index calculation

- **Sum Squared Error**
  $SSE_i = \sum_{k=1}^{N_s}(\widehat{X}_i(k) - X_i(k))^2$ , for $i = 1, 2, 3, 4$. Where $\widehat{X}_i(k) = \widehat{X}_i(k|k)$ for KF and $\widehat{X}_i(k) = \widehat{X}_i(k|k-1)$ for PPO. Also $X_i(k)$ is the true value of $i^{th}$ state.
  Calculate the SSE value for OLSE, PPO, KF-Steady state, and KF-Time Varying implementation.

### Plots to generate

1. Plot of the estimated state obtained using OLSE, PPO, KF-Steady State, and KF-Time Varying along with true value for each state in the same plot using 'hold on'. You can have one figure per state. Don't forget to insert a legend.
2. Plot the manipulated inputs profile for both inputs.
3. Plot of predicted states for KF-Time varying, KF-Steady state and PPO estimator for all four states.

## MATLAB Code To Submit

- Submit all files in a zipped folder. The submitted folder should have the following MATLAB files:
  - File for open loop state estimator implementation.
  - File for Pole placement based observer implementation.

  - File for KF-steady state estimator implementation.
  - File for KF-time-varying estimator implementation.
  - Function file of system dynamics.
  - MAT file for continuous-time linear state space model. (Use the MAT file provided in Assignment 4).
  - File, which on execution, will generate the various plots and return the value of performance indices.

  Keep the file name as $Assn\_04\_ROLLNO.m$ where ROLLNO is your roll number. [**Get in touch with TA for any issues.**]

- Make sure that the plots are labelled properly with axes names, titles, and legends.

---

Learning is fun. Best of Luck!