Lecture 34, Apr 2, 2026

Review – Part 2

Classical Control Design

- Important concepts: controllability/pole placement, observability/observers, separation principle; PBH test, deadbeat control, nonadaptive regulator design
- Controllability definition: every state is open-loop reachable from $x(0) = 0$
- Observability definition: assuming $u(k) = 0$, if we see two identical sequences $\{ y(0), y(1), \dots \}, \{ y'(0), y'(1), \dots \}$ where $y(k) = y'(k), \forall k$, then it must be that $x(0) = x'(0)$; i.e. the initial conditions must be uniquely recoverable from the outputs
- Example: Consider the system $x(k+1) = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k)$
  - Find all equilibria when $u(k) = 0$
    * We solve $A\bar{x} = \bar{x} \implies \begin{cases} \bar{x}_1 = \bar{x}_1 + \bar{x}_2 \\ \bar{x}_2 = 0 \end{cases}$
    * Therefore all equilibria have the form $\bar{x} = \begin{bmatrix} \bar{x}_1 \\ 0 \end{bmatrix}$ where $\bar{x}_1$ is arbitrary, i.e. the equilibria are a line along the x_1 axis
  - Design a controller $u(k) = f(x(k))$ to make all solutions converge to the equilibrium $\begin{bmatrix} \bar{x}_1 \\ 0 \end{bmatrix}$ in two steps
    * Define the error state $z(k) = \begin{bmatrix} x_1(k) - \bar{x}_1 \\ x_2(k) \end{bmatrix}$, so that $z(k) \to 0 \implies x(k) \to \bar{x}$
    * Then $z(k+1) = \begin{bmatrix} x_1(k+1) - \bar{x}_1 \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} x_1(k) + x_2(k) - \bar{x}_1 \\ u(k) \end{bmatrix} = \begin{bmatrix} z_1(k) + z_2(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} z(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k)$
    * Check that the error states are controllable
    * Now we define a deadbeat controller to bring $z(k)$ to 0 in 2 steps; to get the characteristic equation to λ^2, we get $K = \begin{bmatrix} -1 & -1 \end{bmatrix}$
    * Therefore $u(k) = Kz(k) = -(x_1(k) - \bar{x}_1) - x_2(k)$
  - Note that if we had taken the approach of (nonadaptive) regulator design, we would've gotten the exact same result, with $z(k)$ also being the same
- Example: Design an exosystem to model $r(k) = k + 1$
  - We would start with a second order difference equation and match the values at different values of k
  - $\xi(k+1) = \begin{bmatrix} 0 & 1 \\ -1 & 2 \end{bmatrix} \xi(k), \xi(0) = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, r(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} \xi(k)$

Dynamic Programming and Reinforcement Learning

- How did we get from the classical DP approach to RL with Q-learning with policy iteration?

Adaptive Regulation

- We will be working with stable plants only; what simplifications does this allow us to make?
- What is the key theoretical step/key lemma that gets us to our design?
- Example: describe all uses of observers in the entire course
  1. Output-based pole placement (i.e. stabilizing a system based on output measurement)
  2. Regulator design: $\hat{x}(k), \hat{\zeta}(k)$ (observers for the state and exosystem state)
  3. Adaptive regulation: $\hat{z}_s(k)$ for stabilization, $\hat{z}_d(k)$ for the disturbance observer, and $\hat{w}_f(k)$ for the filtered regressor (exosystem state)