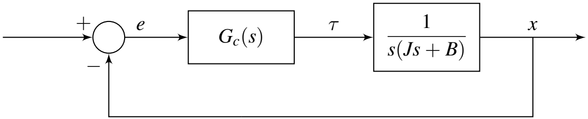
2.153 Adaptive Control
Lecture 7
Adaptive PID Control

Anuradha Annaswamy

aanna@mit.edu

- **Pset #1** out: Thu 19-Feb, **due: Fri 27-Feb**
- **Pset #2** out: Wed 25-Feb, **due: Fri 6-Mar**
- **Pset #3** out: Wed 4-Mar, **due: Fri 13-Mar**
- **Pset #4** out: Wed 11-Mar, **due: Fri 20-Mar**
- **Midterm (take home)** out: Mon 30-Mar, **due: Fri 3-Apr**

Adaptive Control of a Second-order Plant



Plant: $J\ddot{x} + B\dot{x} = \tau \qquad J > 0$

PI Control:
$$G_c(s) = k_p + \frac{k_i}{s}$$
$$\tau = k_p e(t) + k_i \int e(\tau) d\tau$$

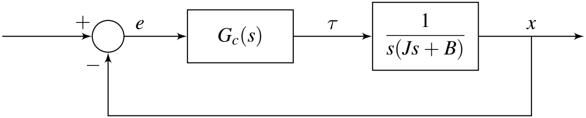
Adaptive PI Control:
$$\tau = k_p(t)e(t) + k_i(t) \int e(\tau) d\tau$$

PID Control:
$$G_c(s) = k_p + k_d s + \frac{k_i}{s}$$
$$\tau = k_p e(t) + k_i \int e(\tau) d\tau + k_d \frac{de}{dt}$$

Adaptive PID Control:
$$\tau = k_p(t)e(t) + k_i(t) \int e(\tau) d\tau + k_d(t)\dot{e}(t)$$

J and B are unknown. Adjust $k_p(t)$, $k_i(t)$ and $k_d(t)$ so that the closed-loop system is stable and $\lim_{t\to\infty} e(t) = 0$.

Adaptive Control of a Second-order Plant



Plant: $$J\ddot{x} + B\dot{x} = \tau \qquad J > 0$$

PI Control: $$G_c(s) = k_p + \frac{k_i}{s}$$

$$\tau = k_p e(t) + k_i \int e(\tau) d\tau$$

Adaptive PI Control: $$\tau = k_p(t) e(t) + k_i(t) \int e(\tau) d\tau$$

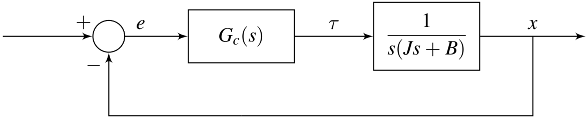
PID Control: $$G_c(s) = k_p + k_d s + \frac{k_i}{s}$$

$$\tau = k_p e(t) + k_i \int e(\tau) d\tau + k_d \frac{de}{dt}$$

Adaptive PID Control: $$\tau = k_p(t) e(t) + k_i(t) \int e(\tau) d\tau + k_d(t) \dot{e}(t)$$

J and B are unknown. Adjust $k_p(t)$, $k_i(t)$ and $k_d(t)$ so that the closed-loop system is stable and $\lim_{t\to\infty} e(t) = 0$.

Adaptive Control of a Second-order Plant



Plant: $$J\ddot{x} + B\dot{x} = \tau \qquad J > 0$$

PI Control: $$G_c(s) = k_p + \frac{k_i}{s}$$

$$\tau = k_p e(t) + k_i \int e(\tau)d\tau$$

Adaptive PI Control: $$\tau = k_p(t)e(t) + k_i(t) \int e(\tau)d\tau$$

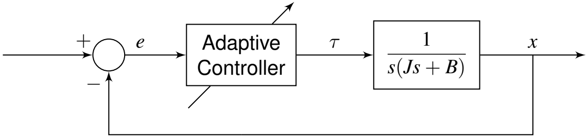
PID Control: $$G_c(s) = k_p + k_d s + \tfrac{k_i}{s}$$

$$\tau = k_p e(t) + k_i \int e(\tau)d\tau + k_d \tfrac{de}{dt}$$

Adaptive PID Control: $$\tau = k_p(t)e(t) + k_i(t) \int e(\tau)d\tau + k_d(t)\dot{e}(t)$$

J and B are unknown. Adjust $k_p(t)$, $k_i(t)$ and $k_d(t)$ so that the closed-loop system is stable and $\lim_{t\to\infty} e(t) = 0$.

Adaptive Control of a Second-order Plant



Plant: $$J\ddot{x} + B\dot{x} = \tau \qquad J > 0$$

PI Control: $$G_c(s) = k_p + \frac{k_i}{s}$$

$$\tau = k_p e(t) + k_i \int e(\tau) d\tau$$

Adaptive PI Control: $$\tau = k_p(t) e(t) + k_i(t) \int e(\tau) d\tau$$

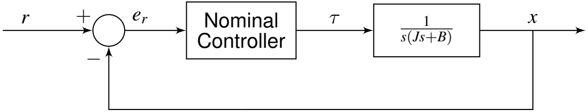
PID Control: $$G_c(s) = k_p + k_d s + \frac{k_i}{s}$$

$$\tau = k_p e(t) + k_i \int e(\tau) d\tau + k_d \frac{de}{dt}$$

Adaptive PID Control: $$\tau = k_p(t) e(t) + k_i(t) \int e(\tau) d\tau + k_d(t) \dot{e}(t)$$

J and B are unknown. Adjust $k_p(t)$, $k_i(t)$ and $k_d(t)$ so that the closed-loop system is stable and $\lim_{t\to\infty} e(t) = 0$

PID -Control: Algebraic Part



$$G_c(s) = k_p + \frac{k_i}{s} + k_d s$$ Parameterize $k_d = K,\ k_p = 2\lambda K > 0,\ \ k_i = \lambda^2 K > 0$

Closed-loop transfer function:

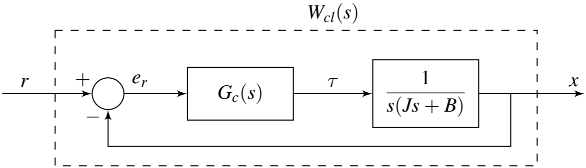
$$\frac{K(s+\lambda)^2}{s^2(Js+B) + K(s+\lambda)^2} = \frac{K(s+\lambda)^2}{Js^3 + s^2(B+K) + 2K\lambda^2 s + K\lambda}$$

Stable if

$$0 < K < \frac{J\lambda}{2} - B.$$

Design the controller so that $x \to x_d$

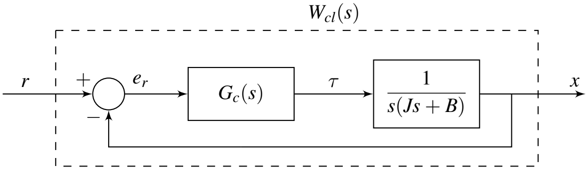
PID Control - Algebraic Part: Tracking



$$
\begin{aligned}
W_{cl}(s) &= \frac{G_c(s)}{s(Js+B)+G_c(s)} \\
W_{cl}^{-1}(s) &= 1 + s(Js+B)G_{cl}^{-1}(s)
\end{aligned}
$$

$$
\begin{aligned}
r &= W_{cl}^{-1}(s)[x_d] \\
&= x_d + \left(s(Js+B)G_{cl}^{-1}(s)\right)[x_d] \\
&= x_d + (s(Js+B))[\omega_d] = x_d + B\dot{\omega}_d + J\ddot{\omega}_d
\end{aligned}
$$

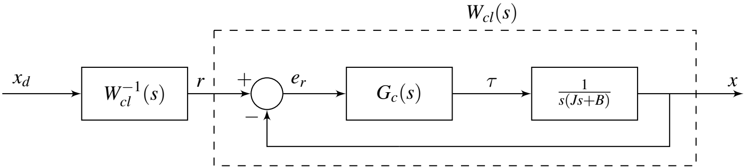
PID Control - Algebraic Part: Tracking



$$\begin{aligned} W_{cl}(s) &= \frac{G_c(s)}{s(Js+B)+G_c(s)} \\ W_{cl}^{-1}(s) &= 1 + s(Js+B)G_{cl}^{-1}(s) \end{aligned}$$

$$\begin{aligned} r &= W_{cl}^{-1}(s)[x_d] \\ &= x_d + \left(s(Js+B)G_{cl}^{-1}(s)\right)[x_d] \\ &= x_d + (s(Js+B))[\omega_d] = x_d + B\dot{\omega}_d + J\ddot{\omega}_d \end{aligned}$$

PID Control - Algebraic Part: Tracking

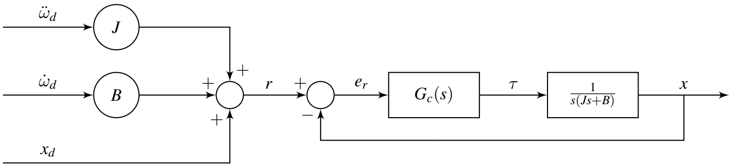


$$
\begin{aligned}
W_{cl}(s) &= \frac{G_c(s)}{s(Js+B)+G_c(s)} \\
W_{cl}^{-1}(s) &= 1 + s(Js+B)G_{cl}^{-1}(s)
\end{aligned}
$$

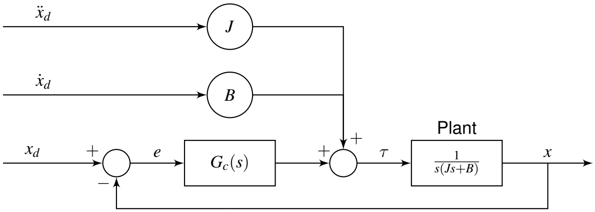
$$
\begin{aligned}
r &= W_{cl}^{-1}(s)[x_d] \\
&= x_d + \left(s(Js+B)G_{cl}^{-1}(s)\right)[x_d] \\
&= x_d + (s(Js+B))[\omega_d] = x_d + B\dot{\omega}_d + J\ddot{\omega}_d
\end{aligned}
$$

PID Control - Algebraic Part: Tracking

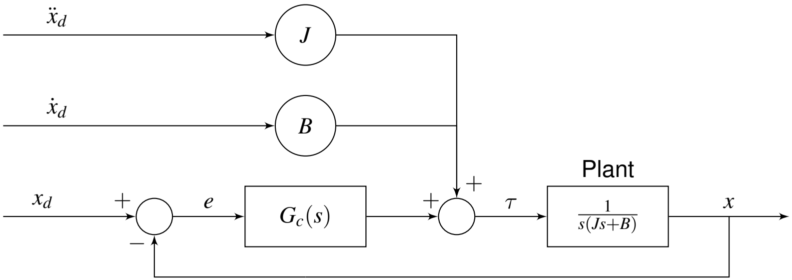
Using $r = J\ddot{\omega}_d + B\dot{\omega}_d + x_d$ the block diagram can be represented as



which can then be simplified to



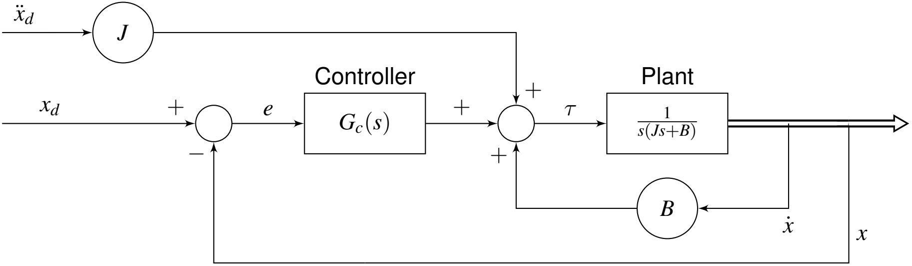
PID Control - Algebraic Part: Tracking - Revised Design



- Move B from feedforward - to feedback

$$G_c(s) = \frac{(K + J\lambda)s + K\lambda}{s}$$

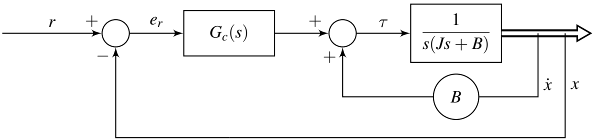
PID Control - Algebraic Part: Tracking - Revised Design



- Move B from feedforward - to feedback

$$G_c(s) = \frac{(K + J\lambda)s + K\lambda}{s}$$

PID Control - Algebraic Part: Tracking - Revised Design

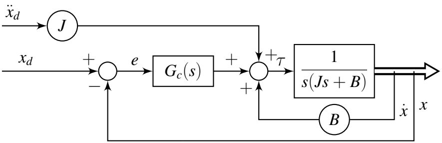


Reparameterize to accommodate J:

$$
\begin{aligned}
G_c(s) &= \frac{(K+2\lambda J)s^2 + (2\lambda K + \lambda^2 J)s + \lambda^2 K}{s} \\
W_{cl}(s) &= \frac{(K+2\lambda J)s^2 + (2\lambda K + \lambda^2 J)s + \lambda^2 K}{Js^3 + (K+2\lambda J)s^2 + (2\lambda K + \lambda^2 J)s + \lambda^2 K}
\end{aligned}
$$

- Always stable, for any J and B.

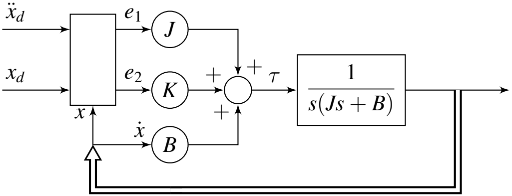
PID Control - Algebraic Part: Tracking - Complete Design



$$G_c(s) = \frac{(K+2\lambda J)s^2 + (2\lambda K + \lambda^2 J)s + \lambda^2 K}{s}, W_{cl}(s) = \frac{G_c(s)}{Js^2 + G_c(s)}$$

$$\begin{aligned}
r &= W_{cl}^{-1}(s)[x_d] \\
&= x_d + \left((Js^2)G_{cl}^{-1}(s)\right)[x_d] \\
&= x_d + J\ddot{\omega}_d \\
\tau &= J\ddot{x}_d + B\dot{x} + G_c(s)[e] \\
&= J\left(\ddot{x}_d + 2\lambda\dot{e} + \lambda^2 e\right) + B\dot{x} + K\left(\dot{e} + 2\lambda e + \lambda^2 \int e(\tau)d\tau\right) = Je_1(t) + B\dot{x} + Ke_2(t)
\end{aligned}$$

PID Control - Algebraic Part: Tracking - Complete Design



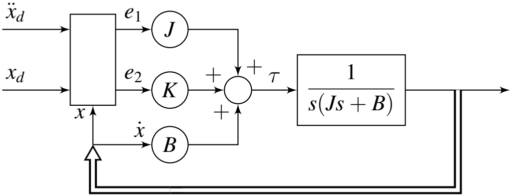
$$
\begin{aligned}
\tau &= Je_1(t) + B\dot{x} + Ke_2(t) \\
e_1 &= \left(\ddot{x}_d + 2\lambda\dot{e} + \lambda^2 e\right), \qquad e_2 = \left(\dot{e} + 2\lambda e + \lambda^2 \int e(\tau)d\tau\right)
\end{aligned}
$$

$$
\phi = \begin{bmatrix} e_1 & \dot{x} & e_2 \end{bmatrix}^\top, \quad \theta^* = \begin{bmatrix} J & B & K \end{bmatrix}^\top
$$

Adaptive PID control:

$$
\tau = \widehat{J}(t)e_1 + \widehat{B}(t)\dot{x} + Ke_2
$$

PID Control - Algebraic Part: Tracking - Complete Design



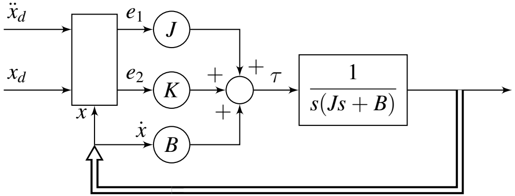
$$\begin{aligned}\tau &= Je_1(t) + B\dot{x} + Ke_2(t)\\ e_1 &= \left(\ddot{x}_d + 2\lambda\dot{e} + \lambda^2 e\right), \qquad e_2 = \left(\dot{e} + 2\lambda e + \lambda^2 \int e(\tau)d\tau\right)\end{aligned}$$

$$\phi = \begin{bmatrix} e_1 & \dot{x} & e_2 \end{bmatrix}^\top, \quad \theta^* = \begin{bmatrix} J & B & K \end{bmatrix}^\top$$

Adaptive PID control:

$$\tau = \widehat{J}(t)e_1 + \widehat{B}(t)\dot{x} + Ke_2$$

PID Control - Algebraic Part: Tracking - Complete Design



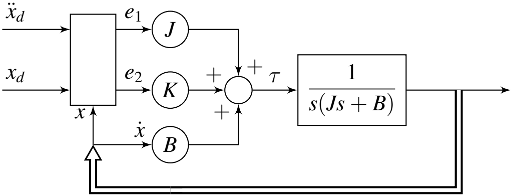
$$
\begin{aligned}
\tau &= Je_1(t) + B\dot{x} + Ke_2(t) \\
e_1 &= \left(\ddot{x}_d + 2\lambda\dot{e} + \lambda^2 e\right), \qquad e_2 = \left(\dot{e} + 2\lambda e + \lambda^2 \int e(\tau)d\tau\right)
\end{aligned}
$$

$$
\phi = \begin{bmatrix} e_1 & \dot{x} & e_2 \end{bmatrix}^\top, \quad \theta^* = \begin{bmatrix} J & B & K \end{bmatrix}^\top
$$

Adaptive PID control:

$$
\tau = \widehat{J}(t)e_1 + \widehat{B}(t)\dot{x} + Ke_2
$$

PID Control - Algebraic Part: Tracking - Complete Design



$$\tau = Je_1(t) + B\dot{x} + Ke_2(t)$$

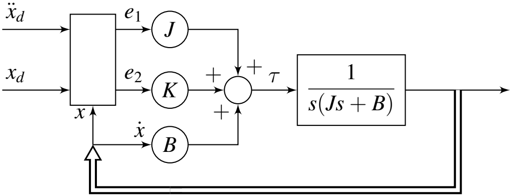
$$e_1 = \left(\ddot{x}_d + 2\lambda\dot{e} + \lambda^2 e\right), \qquad e_2 = \left(\dot{e} + 2\lambda e + \lambda^2 \int e(\tau)d\tau\right)$$

$$\phi = \left[\begin{array}{ccc} e_1 & \dot{x} & e_2 \end{array}\right]^\top, \quad \theta^* = \left[\begin{array}{ccc} J & B & K \end{array}\right]^\top$$

Adaptive PID control:

$$\tau = \widehat{J}(t)e_1 + \widehat{B}(t)\dot{x} + Ke_2$$

PID Control - Algebraic Part: Tracking - Complete Design



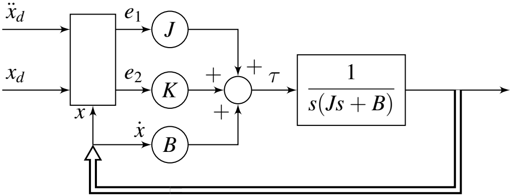
$$
\begin{aligned}
\tau &= Je_1(t) + B\dot{x} + Ke_2(t) \\
e_1 &= \left(\ddot{x}_d + 2\lambda\dot{e} + \lambda^2 e\right), \qquad e_2 = \left(\dot{e} + 2\lambda e + \lambda^2 \int e(\tau)d\tau\right)
\end{aligned}
$$

$$
\phi = \begin{bmatrix} e_1 & \dot{x} & e_2 \end{bmatrix}^\top, \quad \theta^* = \begin{bmatrix} J & B & K \end{bmatrix}^\top
$$

Adaptive PID control:

$$
\tau = \widehat{J}(t)e_1 + \widehat{B}(t)\dot{x} + Ke_2
$$

PID Control - Algebraic Part: Tracking - Complete Design



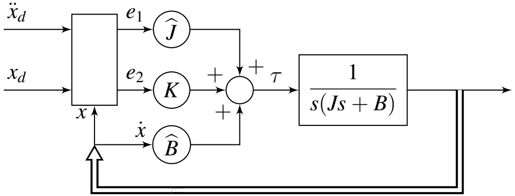
$$\begin{aligned} \tau &= Je_1(t) + B\dot{x} + Ke_2(t) \\ e_1 &= \left(\ddot{x}_d + 2\lambda\dot{e} + \lambda^2 e\right), \qquad e_2 = \left(\dot{e} + 2\lambda e + \lambda^2 \int e(\tau)d\tau\right) \end{aligned}$$

$$\phi = \begin{bmatrix} e_1 & \dot{x} & e_2 \end{bmatrix}^\top, \quad \theta^* = \begin{bmatrix} J & B & K \end{bmatrix}^\top$$

Adaptive PID control:

$$\tau = \widehat{J}(t)e_1 + \widehat{B}(t)\dot{x} + Ke_2$$

PID Control - Algebraic Part: Tracking - Complete Design



$$\begin{aligned}\tau &= Je_1(t) + B\dot{x} + Ke_2(t) \\ e_1 &= \left(\ddot{x}_d + 2\lambda\dot{e} + \lambda^2 e\right), \qquad e_2 = \left(\dot{e} + 2\lambda e + \lambda^2 \int e(\tau)d\tau\right)\end{aligned}$$

$$\phi = \begin{bmatrix} e_1 & \dot{x} & e_2 \end{bmatrix}^\top, \quad \theta^* = \begin{bmatrix} J & B & K \end{bmatrix}^\top$$

Adaptive PID control:

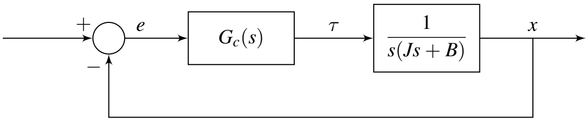
$$\tau = \widehat{J}(t)e_1 + \widehat{B}(t)\dot{x} + Ke_2$$

Adaptive PID Control ($\dot{x}$ measurable)

$$
\begin{aligned}
\tau &= \widehat{J}(t)e_1 + \widehat{B}(t)\dot{x} + Ke_2 \\
\text{Plant+controller: } \ddot{x} &= \frac{1}{J}\left(-B\dot{x} + \tau\right) \\
&= \frac{1}{J}\left(-B\dot{x} + \widehat{J}(t)e_1 + \widehat{B}(t)\dot{x} + Ke_2\right) \\
e_2 &= \left(\dot{e} + 2\lambda e + \lambda^2 \int e(\tau)d\tau\right) \\
\dot{e}_2 &= -\frac{K}{J}e_2 + \frac{1}{J}\left(-\widetilde{J}e_1 - \widetilde{B}\dot{x}\right) \text{ – Error Model 3}
\end{aligned}
$$

Globally stable; $\lim_{t\to\infty} e_2(t) = \lim_{t\to\infty} e(t) = 0$.

Adaptive Phase Lead Compensators



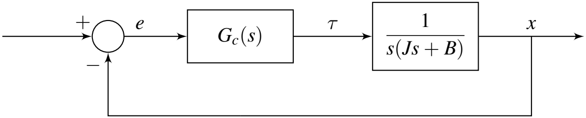
Plant: $\quad J\ddot{x} + B\dot{x} = \tau \qquad J > 0$

Phase-lead: $\quad G_c(s) = k\dfrac{s+z_0}{s+p_0}, \qquad z_0 < p_0$

$$\tau = G_c(s)e$$

J and B are unknown. Deternine τ so that $\lim_{t\to\infty} e(t) = 0$.

Adaptive Phase Lead Compensators



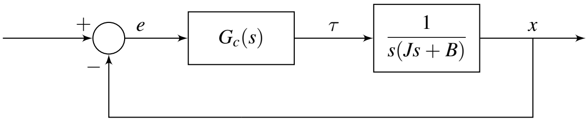
$$\text{Plant:} \qquad J\ddot{x} + B\dot{x} = \tau \qquad J > 0$$

$$\text{Phase-lead:} \quad G_c(s) = k\frac{s+z_0}{s+p_0}, \qquad z_0 < p_0$$

$$\tau = G_c(s)e$$

J and B are unknown. Deternine τ so that $\lim_{t\to\infty} e(t) = 0$.

Adaptive Phase Lead Compensators



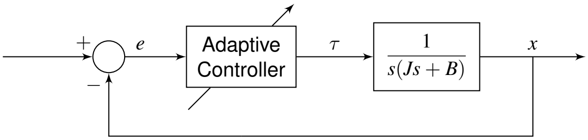
$$\text{Plant:} \qquad J\ddot{x} + B\dot{x} = \tau \qquad J > 0$$

$$\text{Phase-lead:} \quad G_c(s) = k\frac{s+z_0}{s+p_0}, \qquad z_0 < p_0$$

$$\tau = G_c(s)e$$

J and B are unknown. Deternine τ so that $\lim_{t\to\infty} e(t) = 0$.

Adaptive Phase Lead Compensators



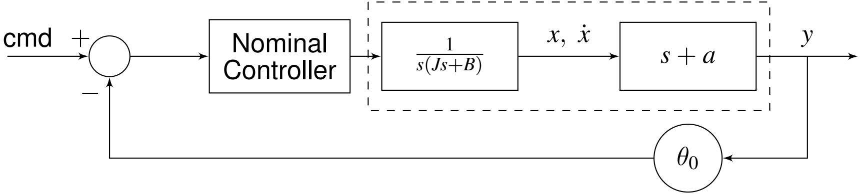
Plant: $J\ddot{x} + B\dot{x} = \tau \qquad J > 0$

Phase-lead: $G_c(s) = k\dfrac{s+z_0}{s+p_0}, \qquad z_0 < p_0$

$$\tau = G_c(s)e$$

J and B are unknown. Deternine τ so that $\lim_{t\to\infty} e(t) = 0$.

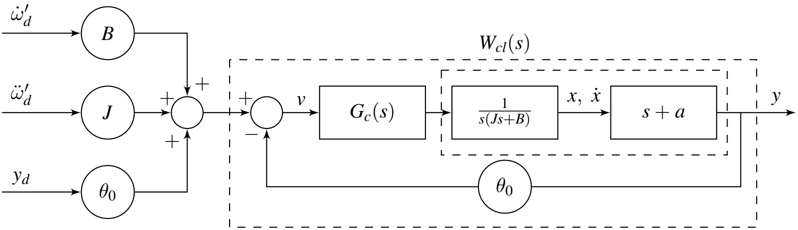
Phase Lead Compensators - Algebraic Part



$$G_c(s) = k\frac{s+z_0}{s+p_0}, \qquad z_0 < p_0$$

- Always stable for any $J, B, z_0, p_0 > 0$ with $z_0 < p_0$.
- Assume x and $\dot{x}$ measurable

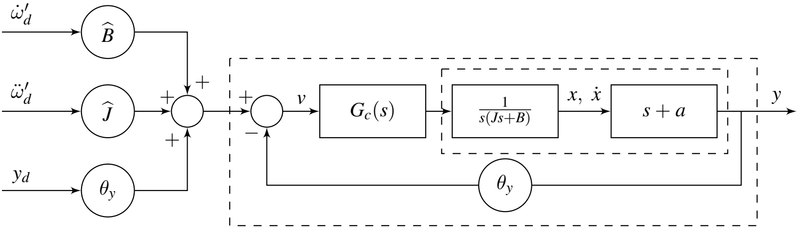
Phase Lead Compensators - Synthetic output y



$$\nu \quad = \quad \theta_0(y_d - y) + B\dot{\omega}'_d + J\ddot{\omega}'_d$$

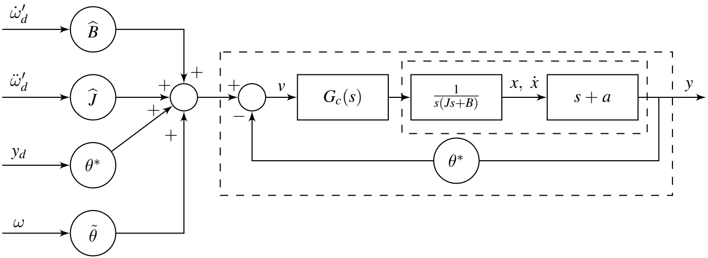
- Stable for all parameters of $G_c(s)$
- $\theta_0 = \theta^*$ - value for which $W_{cl}(s)$ has a desired phase margin

Adaptive Phase Lead Compensators - Synthetic output y



$$
\begin{aligned}
\nu &= \theta_y(t)(y_d - y) + \widehat{B}\dot{\omega}'_d + \widehat{J}\ddot{\omega}'_d \\
&= \widetilde{\theta}_y(t)(y_d - y) + \widetilde{B}(t)\dot{\omega}'_d + \widetilde{J}(t)\ddot{\omega}'_d + \theta^*(y_d - y) + B\dot{\omega}'_d + J\ddot{\omega}'_d
\end{aligned}
$$

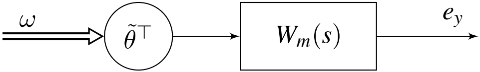
Adaptive Phase Lead Compensators - Synthetic output y



$$\nu = \widetilde{\theta}^T \omega + \theta^* e_y + B\dot{\omega}_d' + J\ddot{\omega}_d'$$

$$\widetilde{\theta} = \begin{bmatrix} \widetilde{\theta}_y \\ \widetilde{B} \\ \widetilde{J} \end{bmatrix}, \quad \omega = \begin{bmatrix} e_y \\ \dot{\omega}_d' \\ \ddot{\omega}_d' \end{bmatrix}$$

Underlying Error Model



$$
\begin{aligned}
W_m(s) &= \frac{\frac{k_c}{J}\left(s+z_c\right)\left(s+a\right)}{s\left(s+p_c\right)\left(s+\frac{B}{J}\right)+\theta^*\frac{k_c}{J}\left(s+z_c\right)\left(s+a\right)} \\
\widetilde{\theta} &= \begin{bmatrix} \widetilde{\theta}_y \\ \widetilde{B} \\ \widetilde{J} \end{bmatrix}, \quad \omega = \begin{bmatrix} e_y \\ \dot{\omega}'_d \\ \ddot{\omega}'_d \end{bmatrix}
\end{aligned}
$$