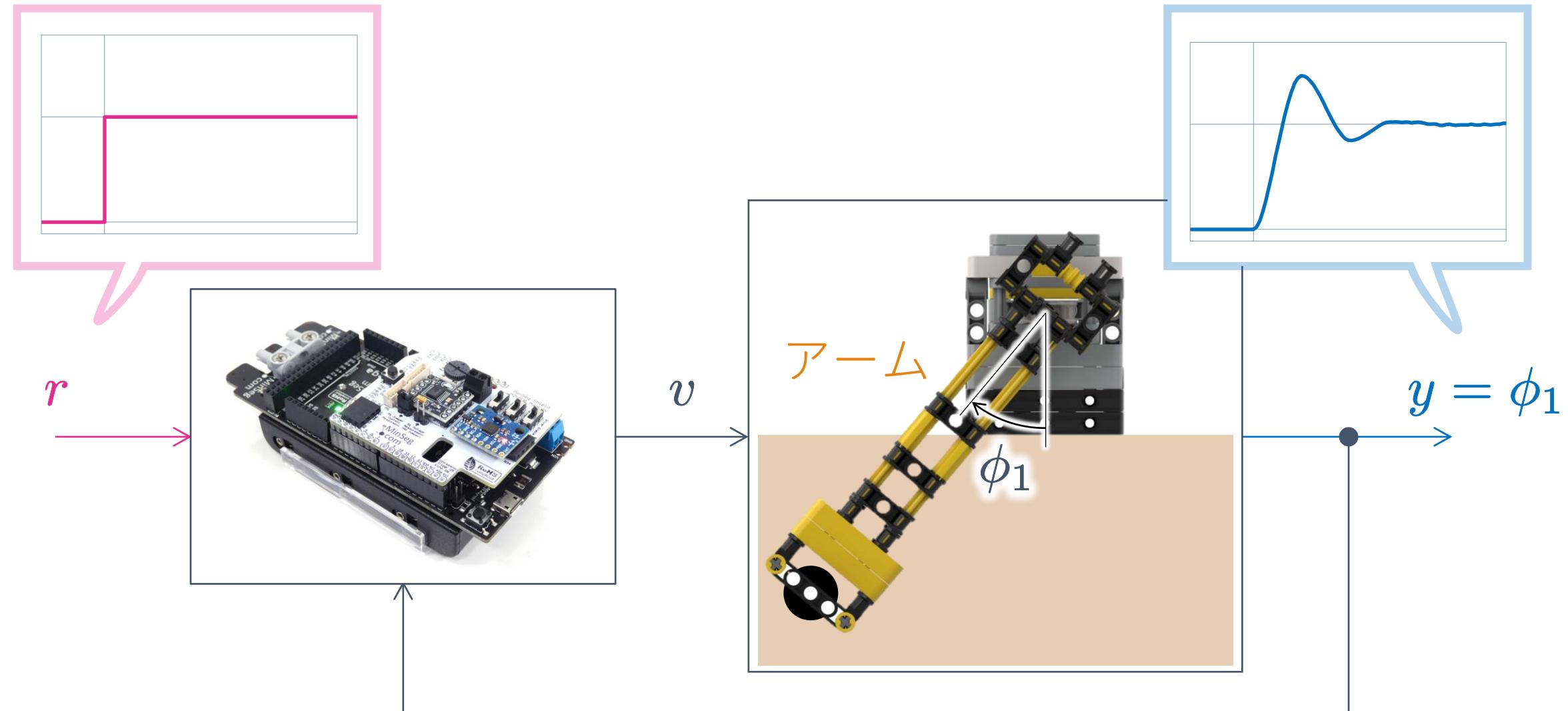


- はじめに
- LEGO 倒立振子の製作
- **PID 制御の学習**
- 現代制御の学習
- アドバンスト制御の学習
- まとめ

PID 制御の学習 (鉛直面を回転するアームの角度制御)



PID 制御の学習 (鉛直面を回転するアームの角度制御)

- ▶ 各動作の効果
- ▶ FRIT による設計
- ▶ モデルに基づく設計

- モデリング
- 2次遅れ系の特性に基づくパラメータ同定
- 最小2乗法に基づくパラメータ同定
- モデルマッチング法による設計
- 非線形補償
- モデル誤差抑制補償 (MEC)

学習できる
内容の一例

PID 制御の学習 (鉛直面を回転するアームの角度制御)

▶ 各動作の効果

▶ FRIT による設計

▶ モデルに基づく設計

- 非線形モデル
- 2次遅れ系の特性に基づくパラメータ同定
- 最小2乗法に基づくパラメータ同定
- モデルマッチング法による設計
- 非線形補償
- モデル誤差抑制補償 (MEC)



P 動作

I 動作

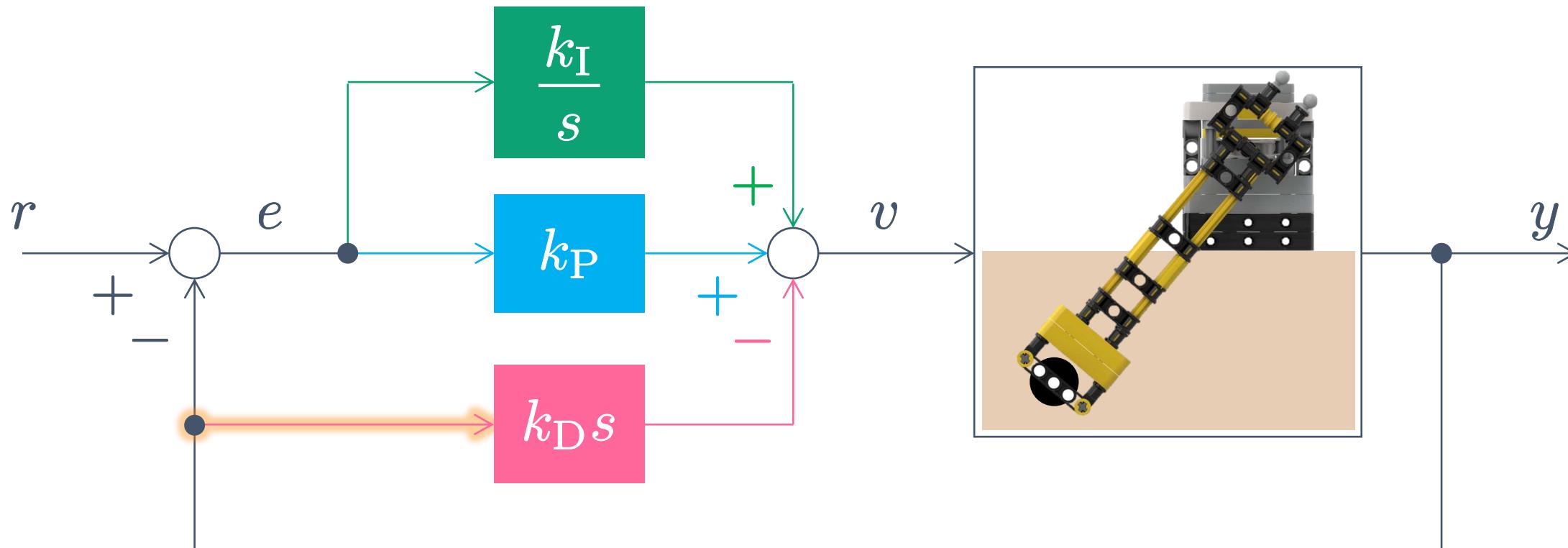
D 動作

► 各動作の効果

PI-D (微分先行型 PID) コントローラ

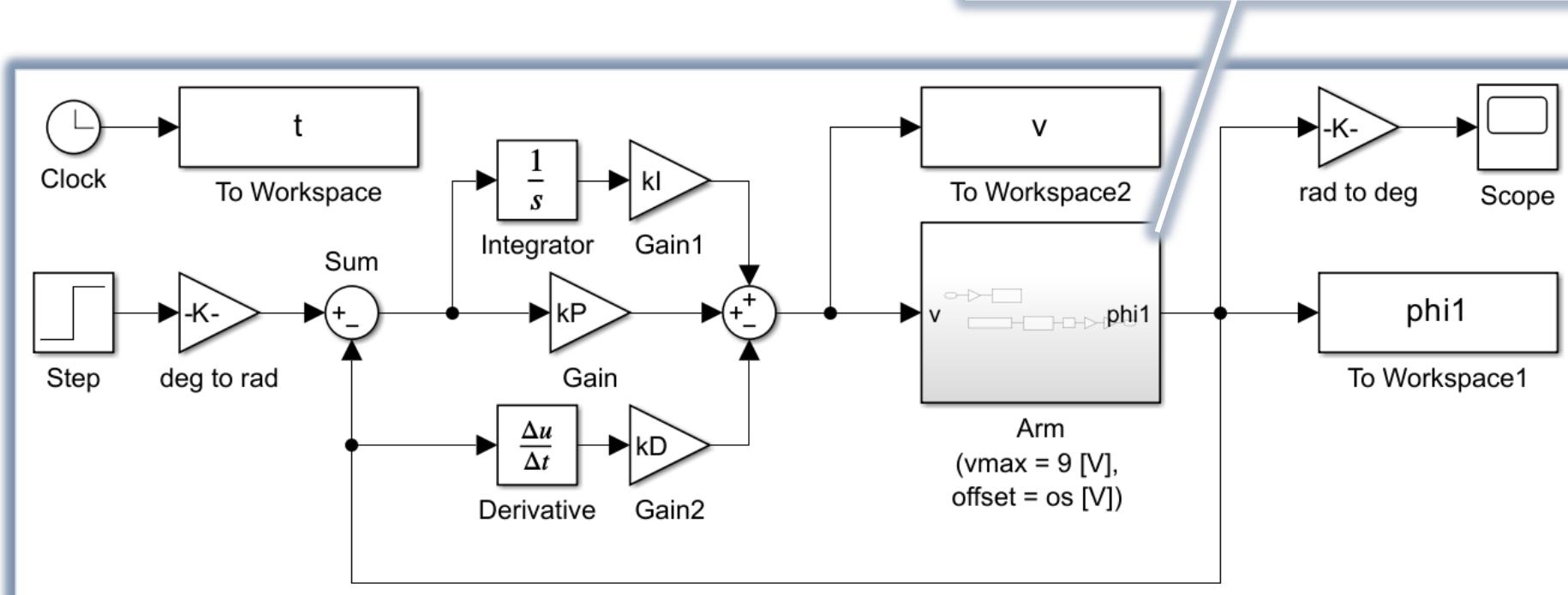
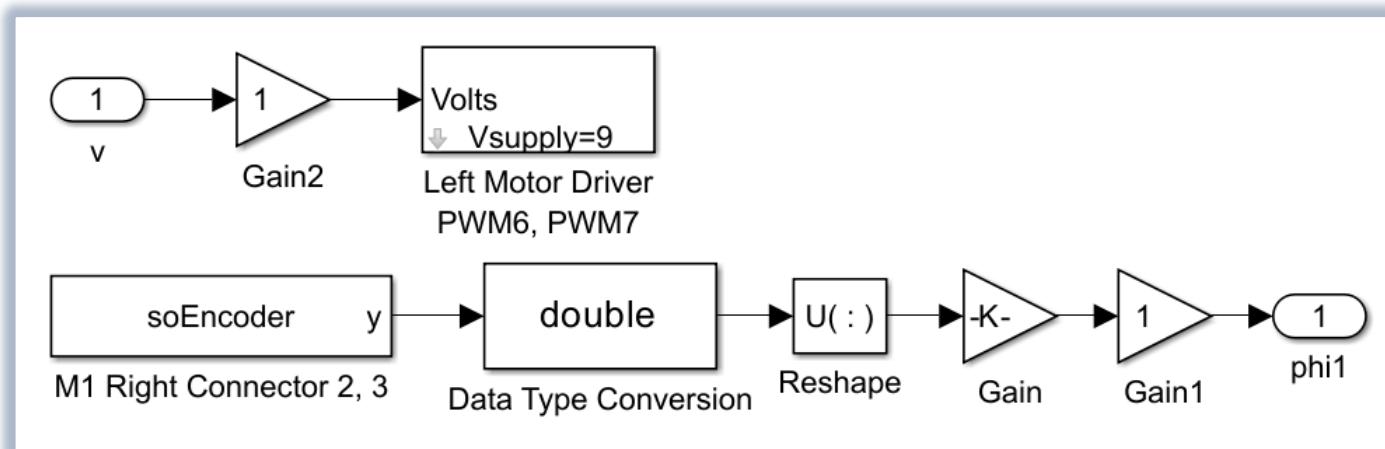
$$v(t) = k_P e(t) + k_I \int_0^t e(t) dt - k_D \frac{dy(t)}{dt}$$

目標角度が急激に
変化しても制御入力が
過大にならない



Simulink モデルでの実装

- サンプリング周期：5 ms
- ode1 (オイラー法) で離散化
- 最大電圧：9 V
- エクスターナルモードで実行

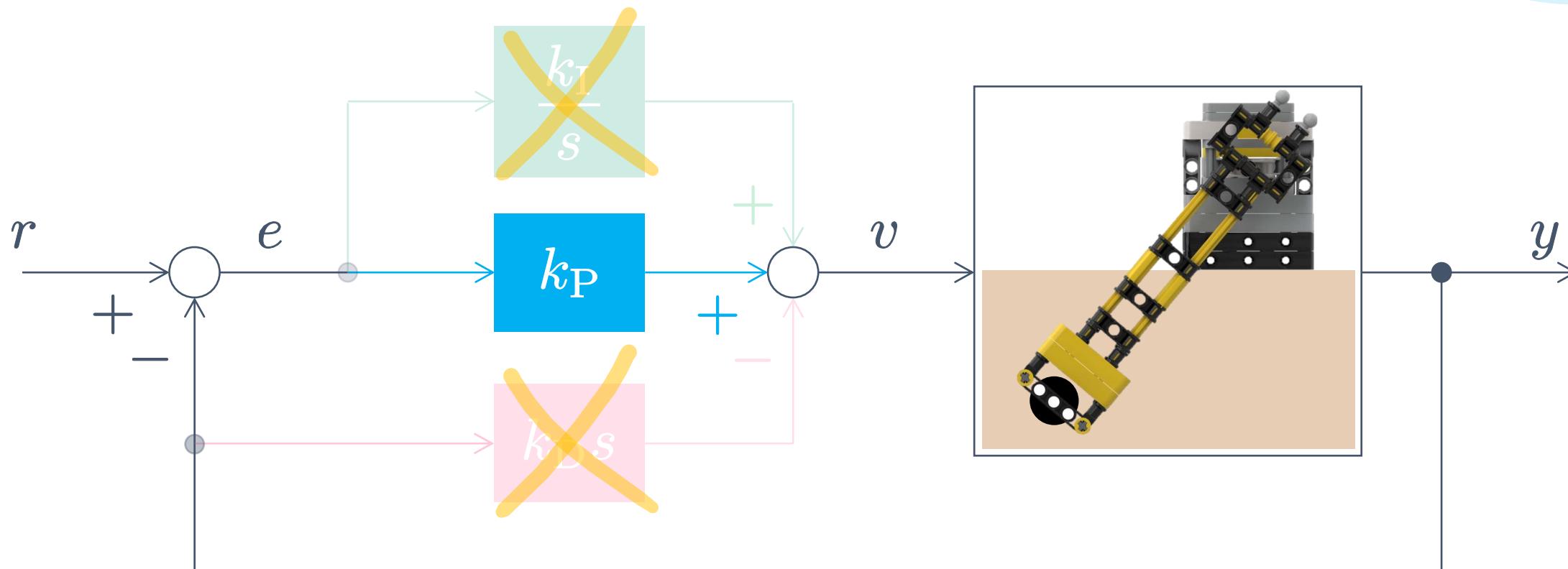


比例 (P) 動作の効果

P コントローラ

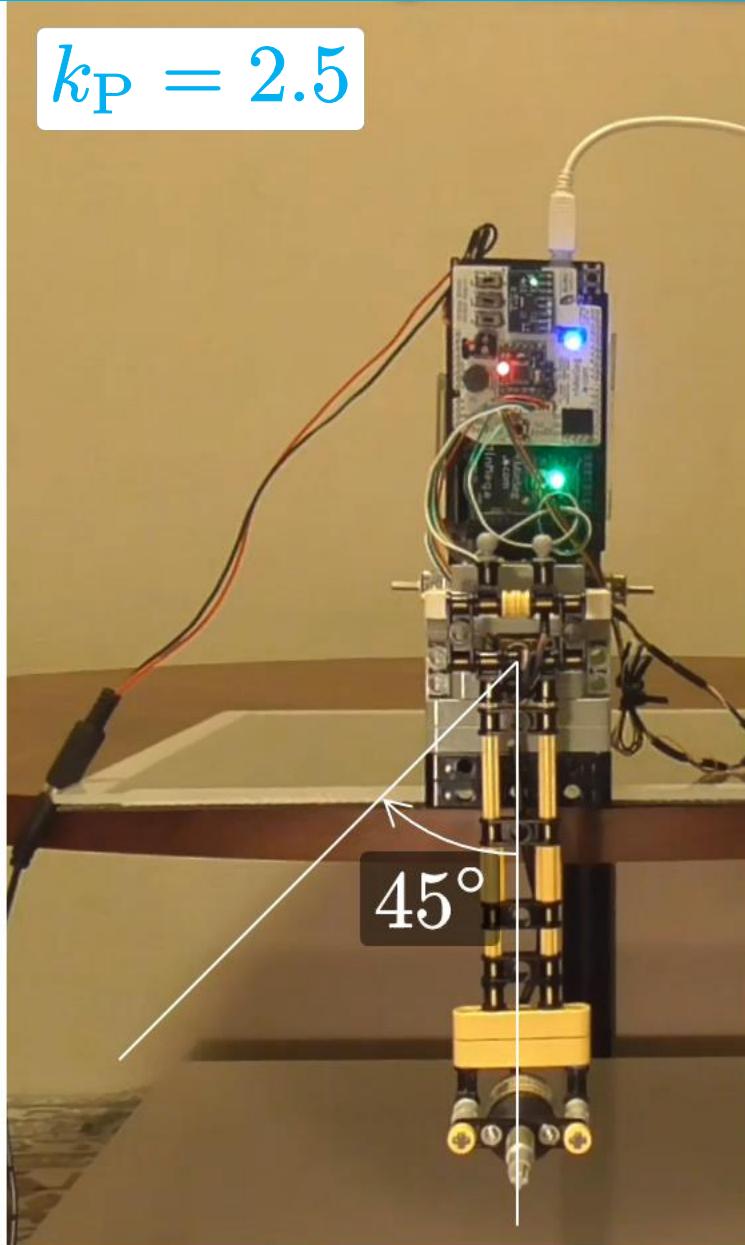
$$v(t) = k_P e(t) + k_I \int_0^t e(t) dt - k_D \frac{dy(t)}{dt}$$

最も単純な
制御手法！

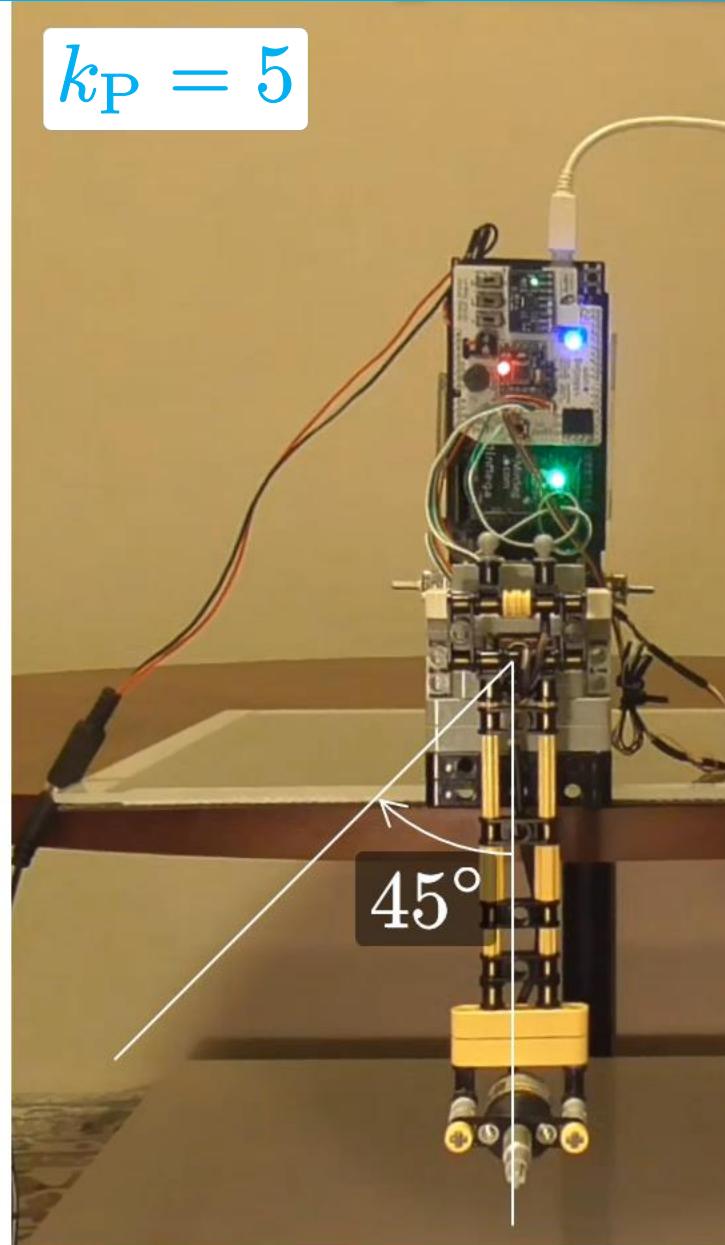


比例 (P) 動作の効果

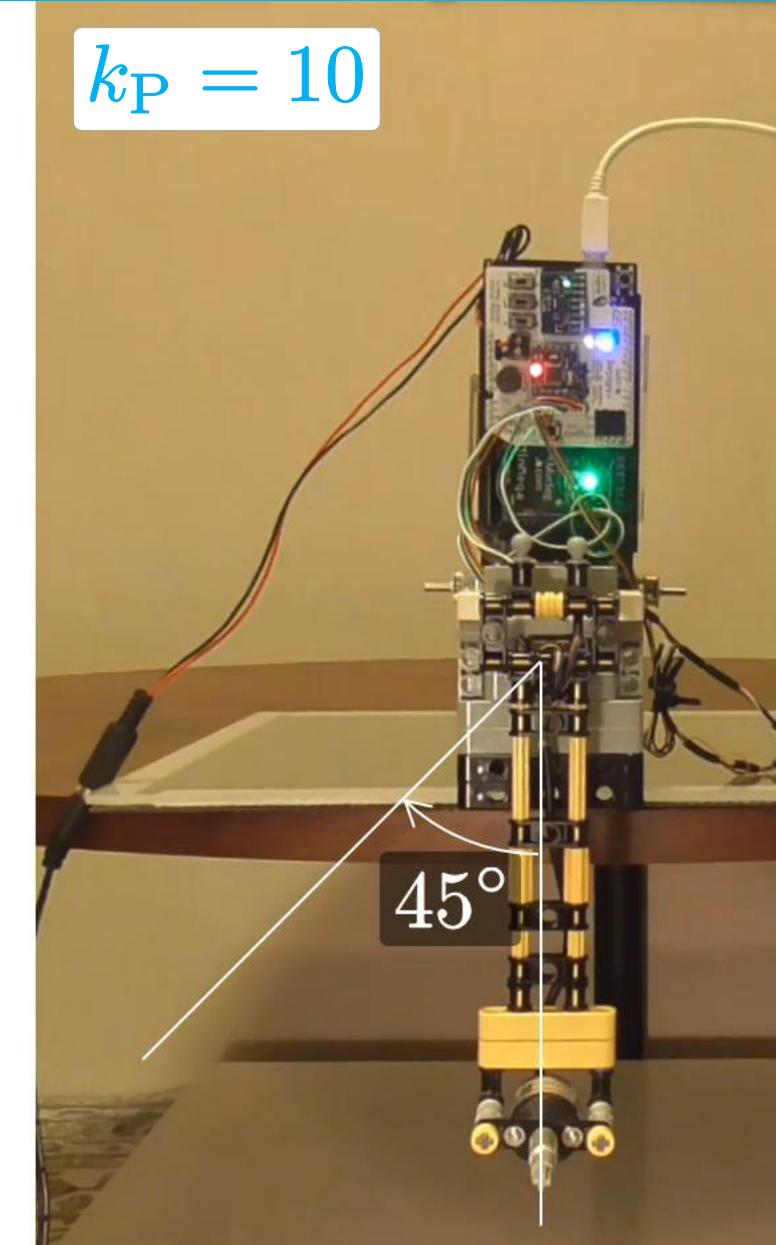
$k_P = 2.5$



$k_P = 5$



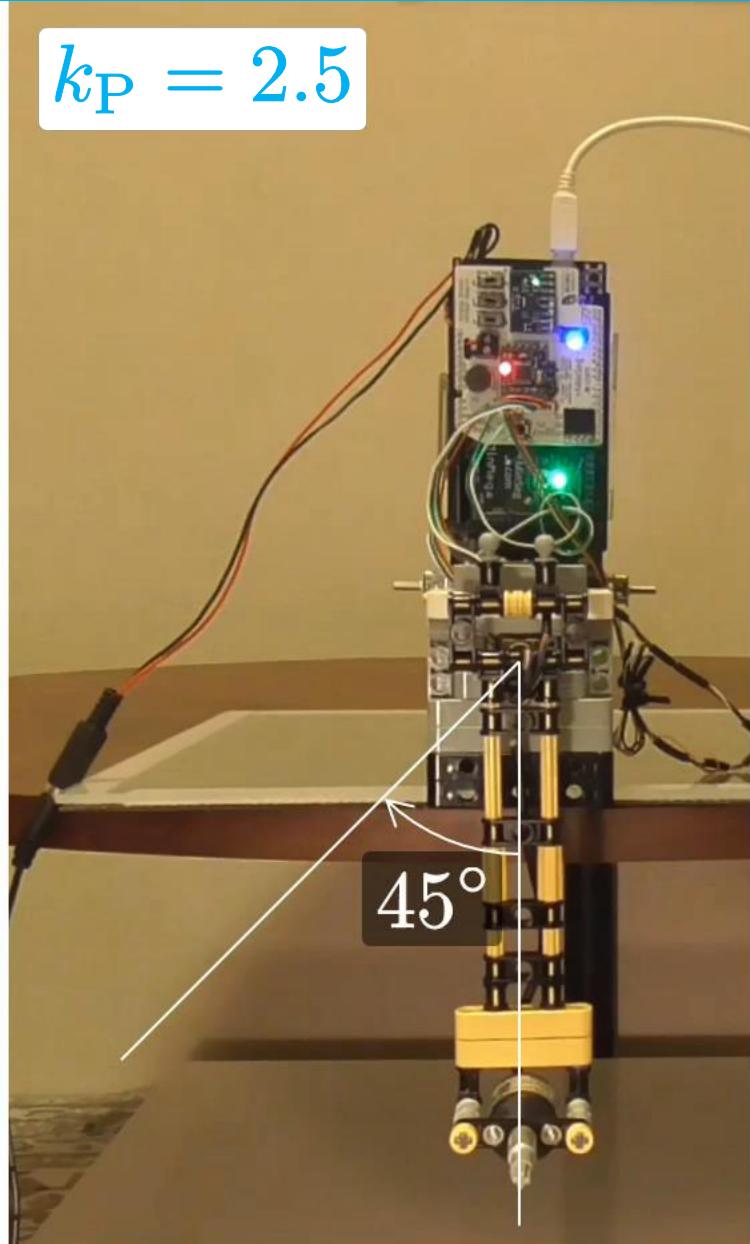
$k_P = 10$



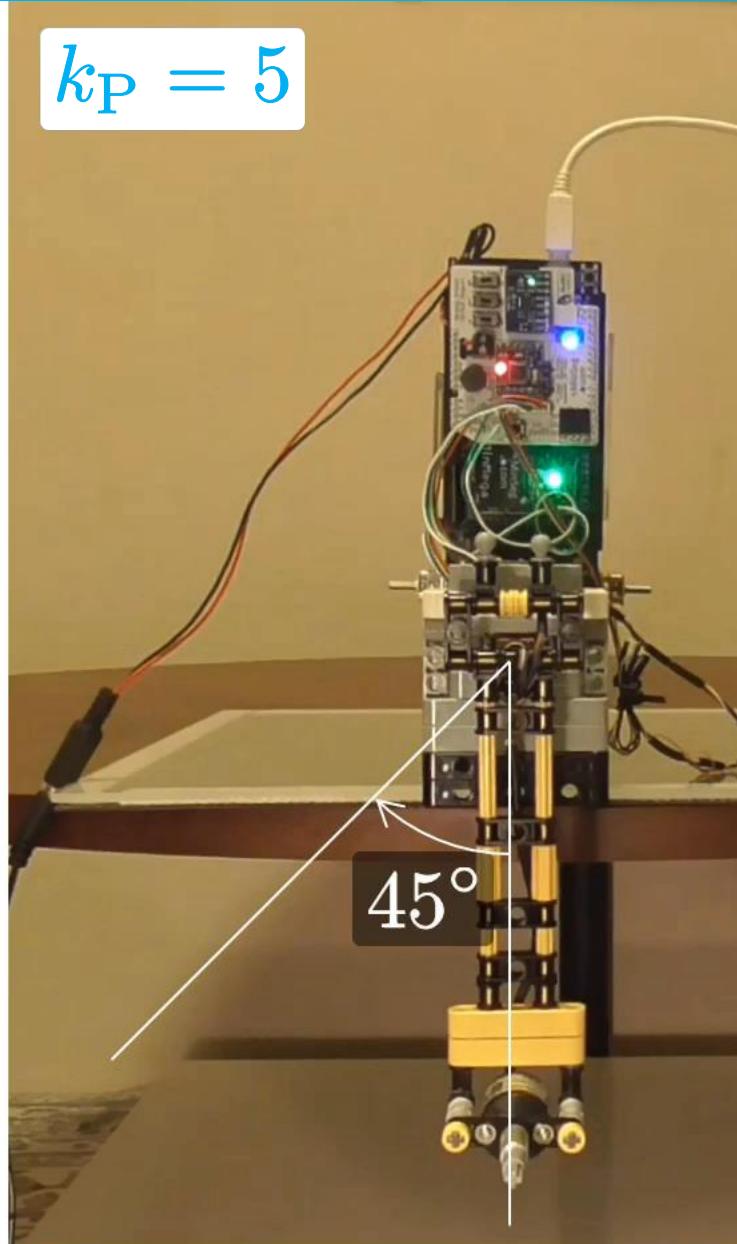
比例(P)動作の効果

0.25倍速

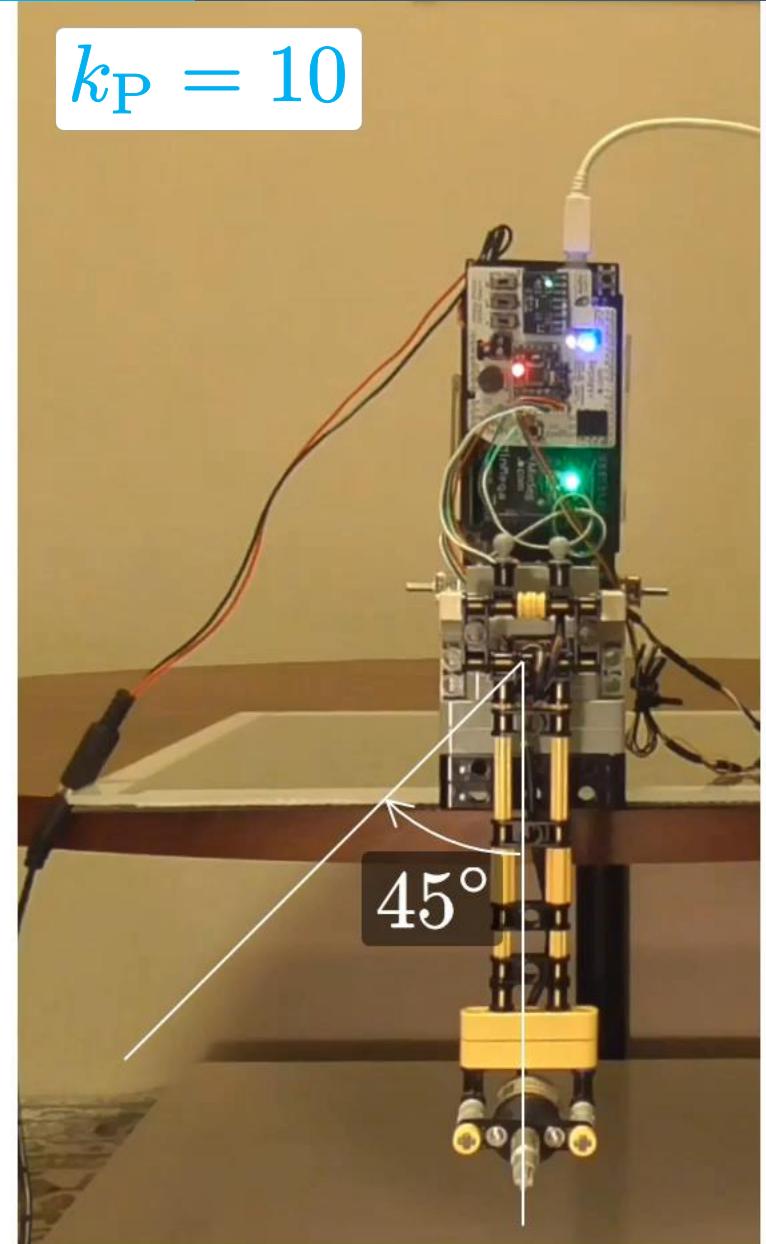
$k_P = 2.5$



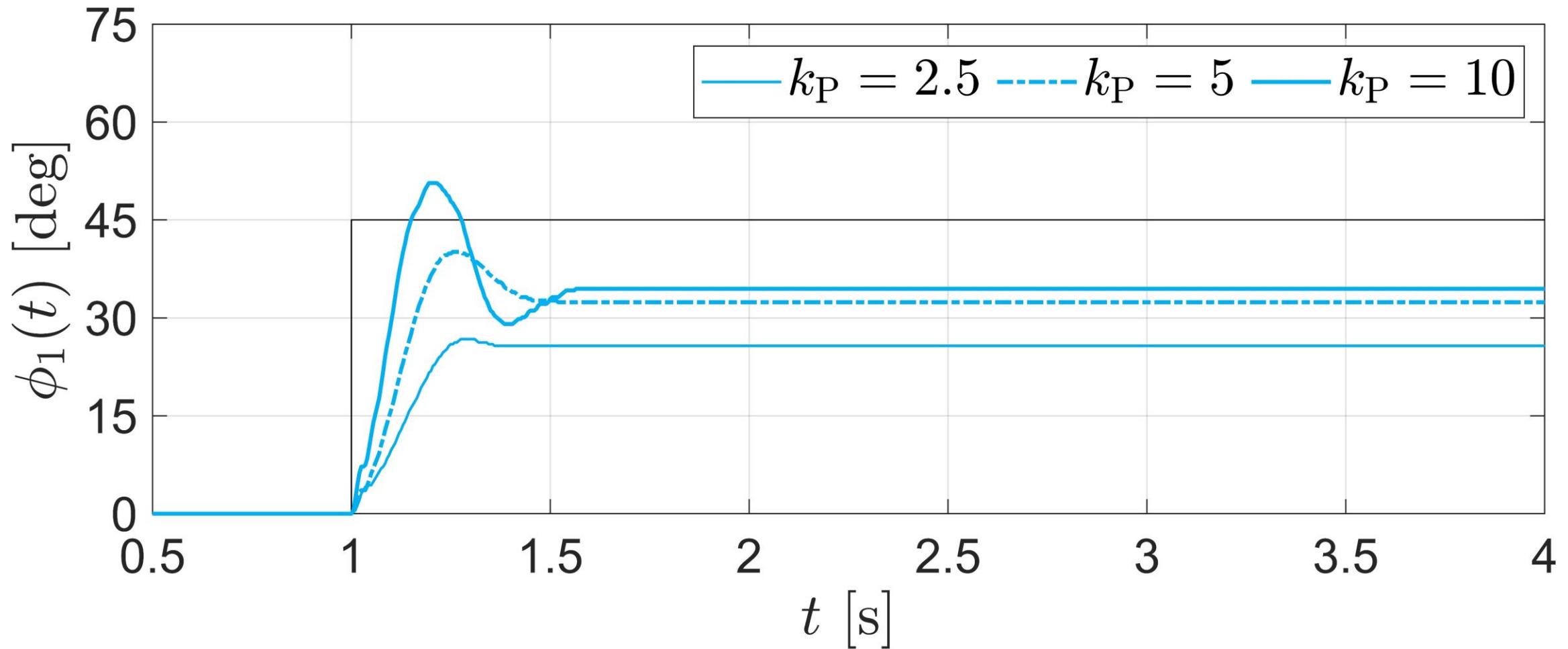
$k_P = 5$



$k_P = 10$



比例 (P) 動作の効果

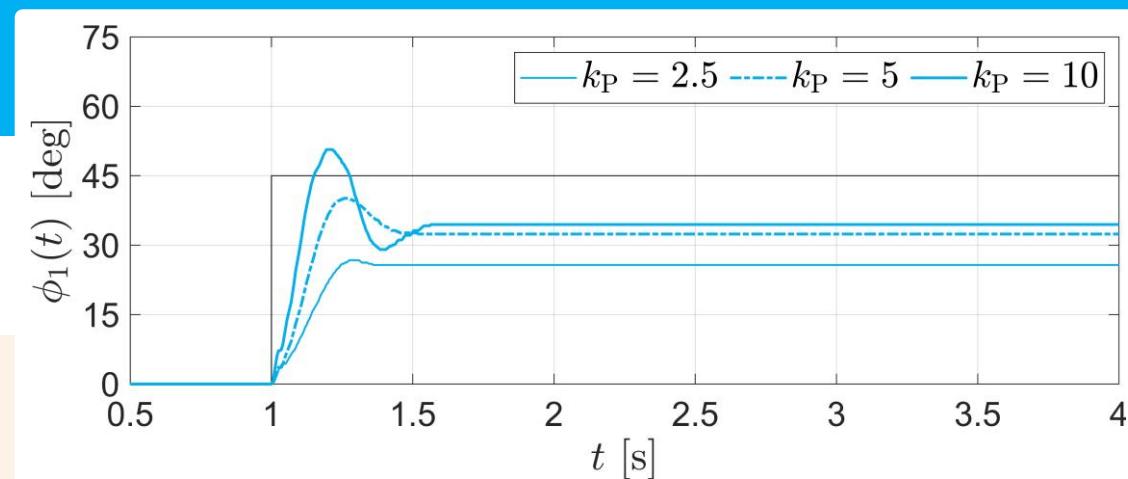


比例(P)動作の効果

比例ゲイン k_P を大きくすると …



- 反応が速くなる！
- 定常偏差が小さくなる！



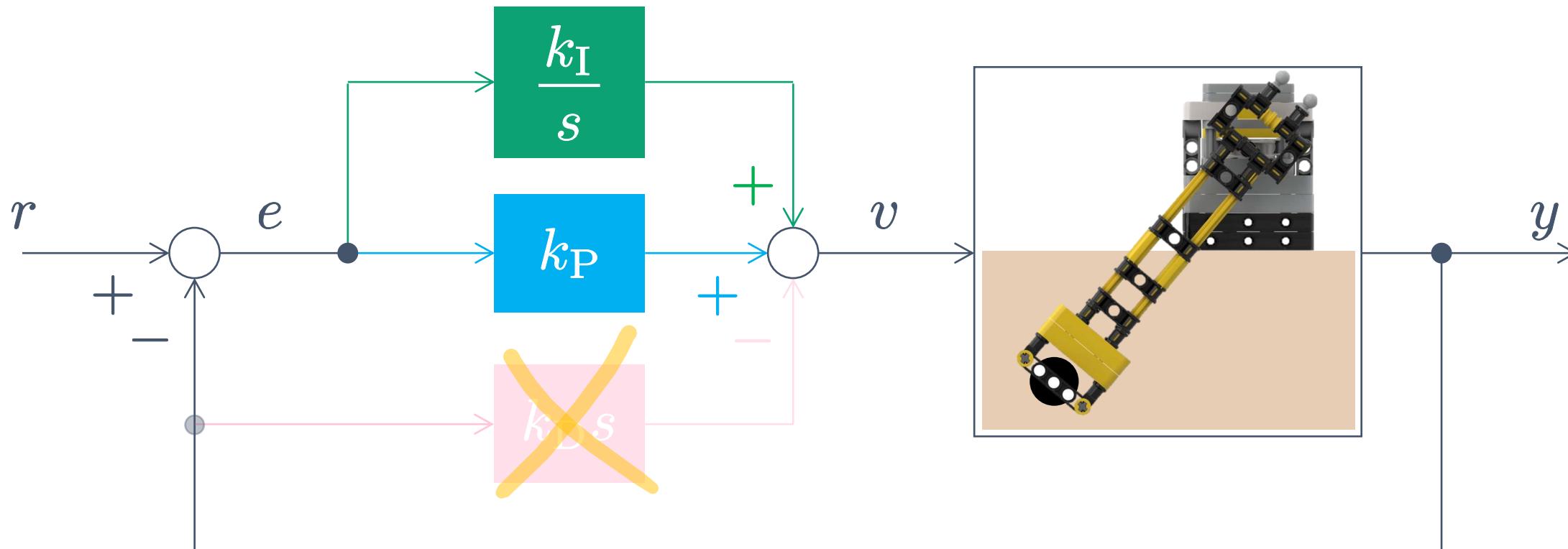
- 定常偏差が残る！
(重力や静止摩擦・動摩擦の影響)
- 振動的になる！
(オーバーシュートが大きくなる！)

積分 (I) 動作の効果

PI コントローラ

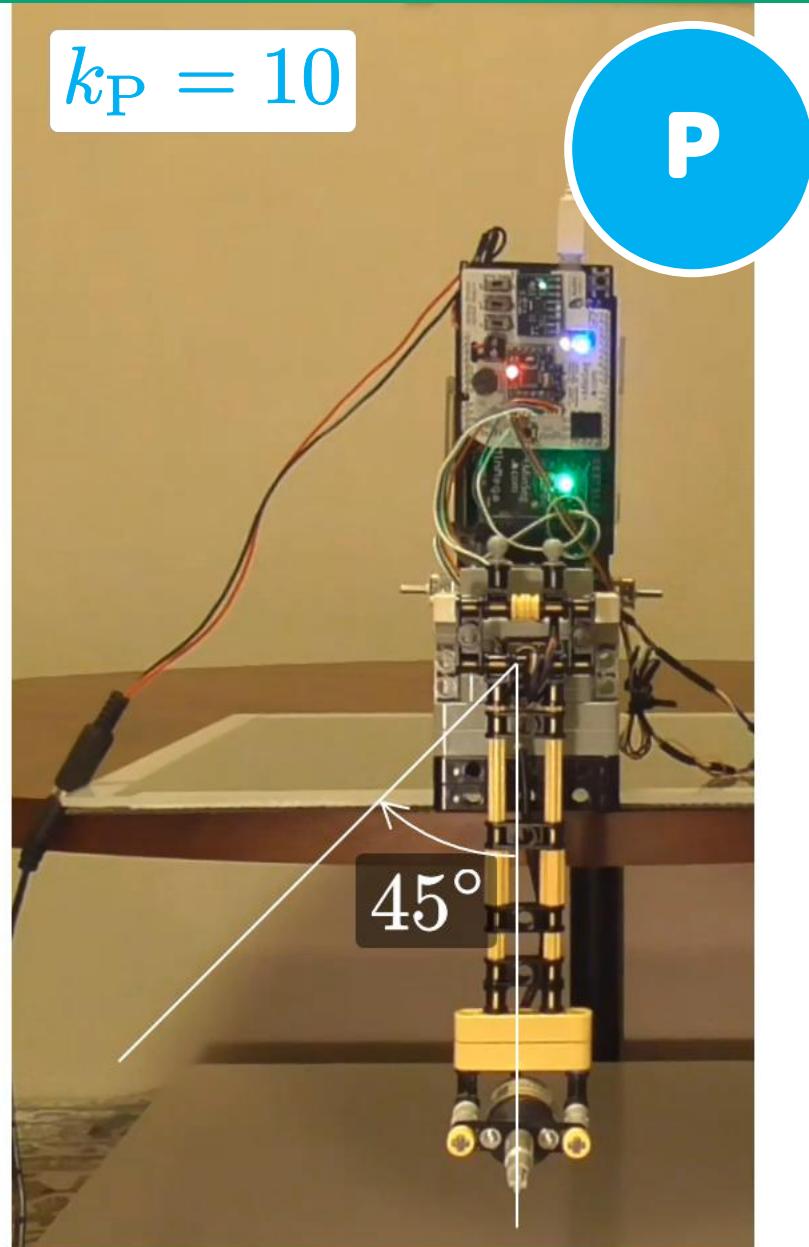
$$v(t) = k_P e(t) + k_I \int_0^t e(t) dt - k_D \cancel{\frac{dy(t)}{dt}}$$

定常特性
の改善！

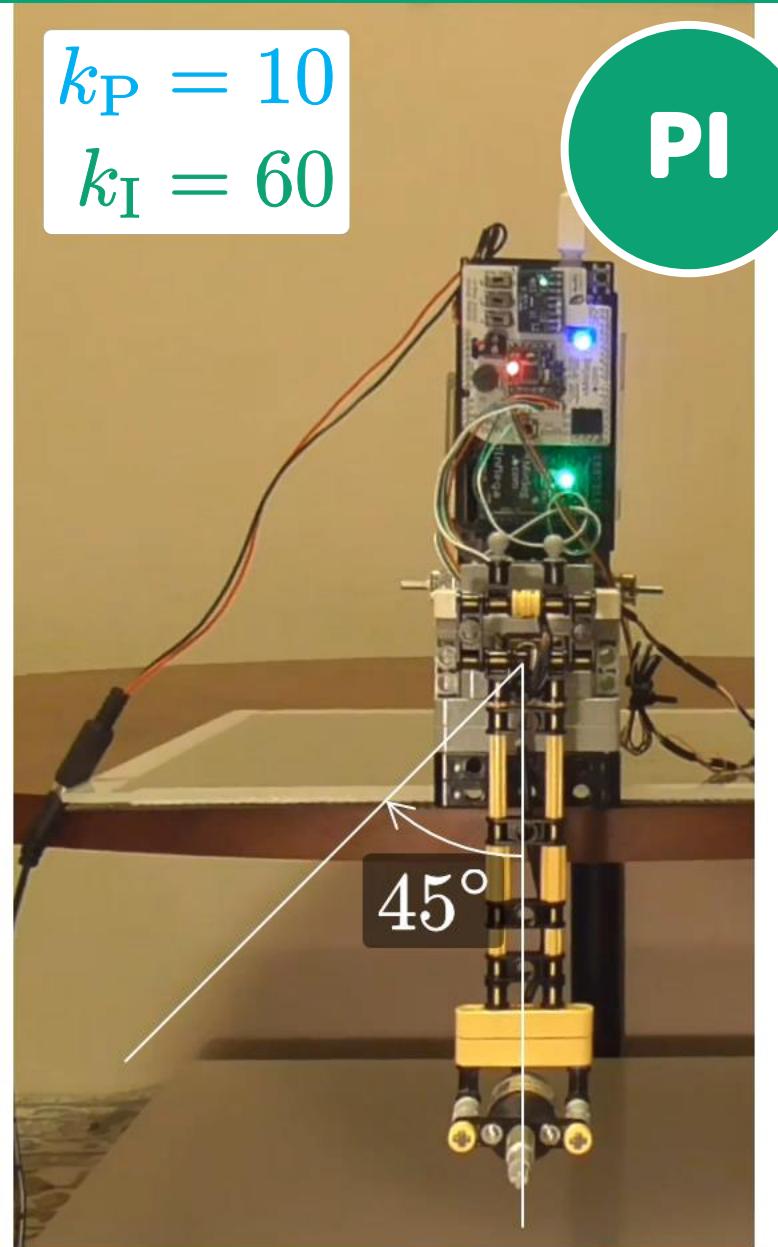


積分 (I) 動作の効果

$k_P = 10$



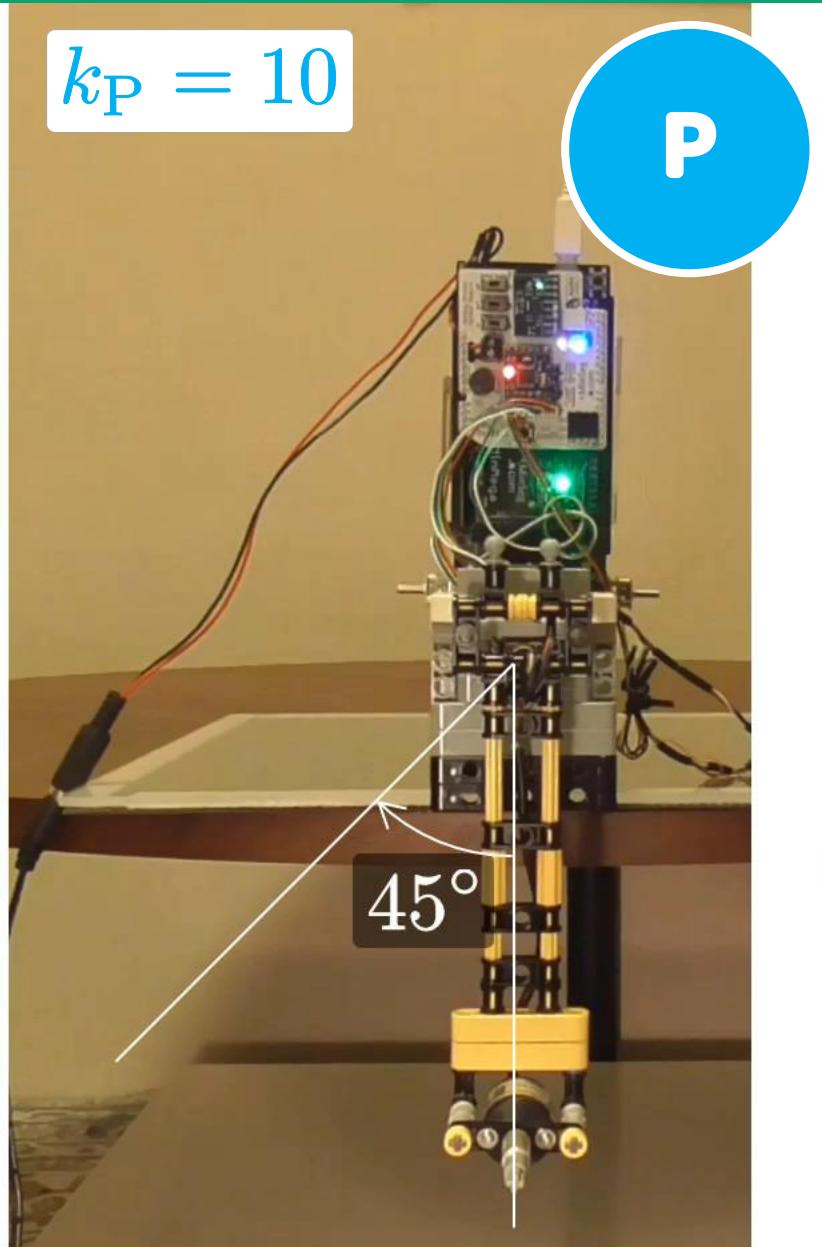
$k_P = 10$
 $k_I = 60$



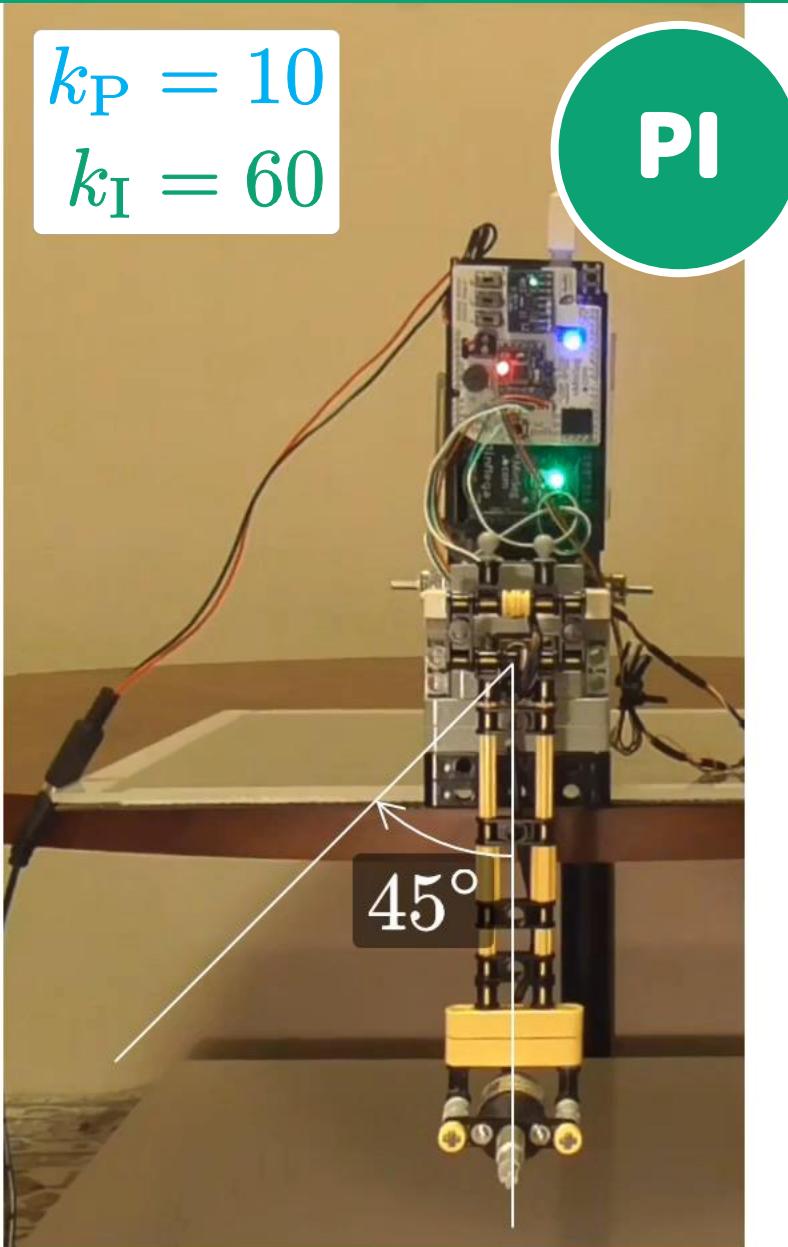
積分 (I) 動作の効果

0.25 倍速

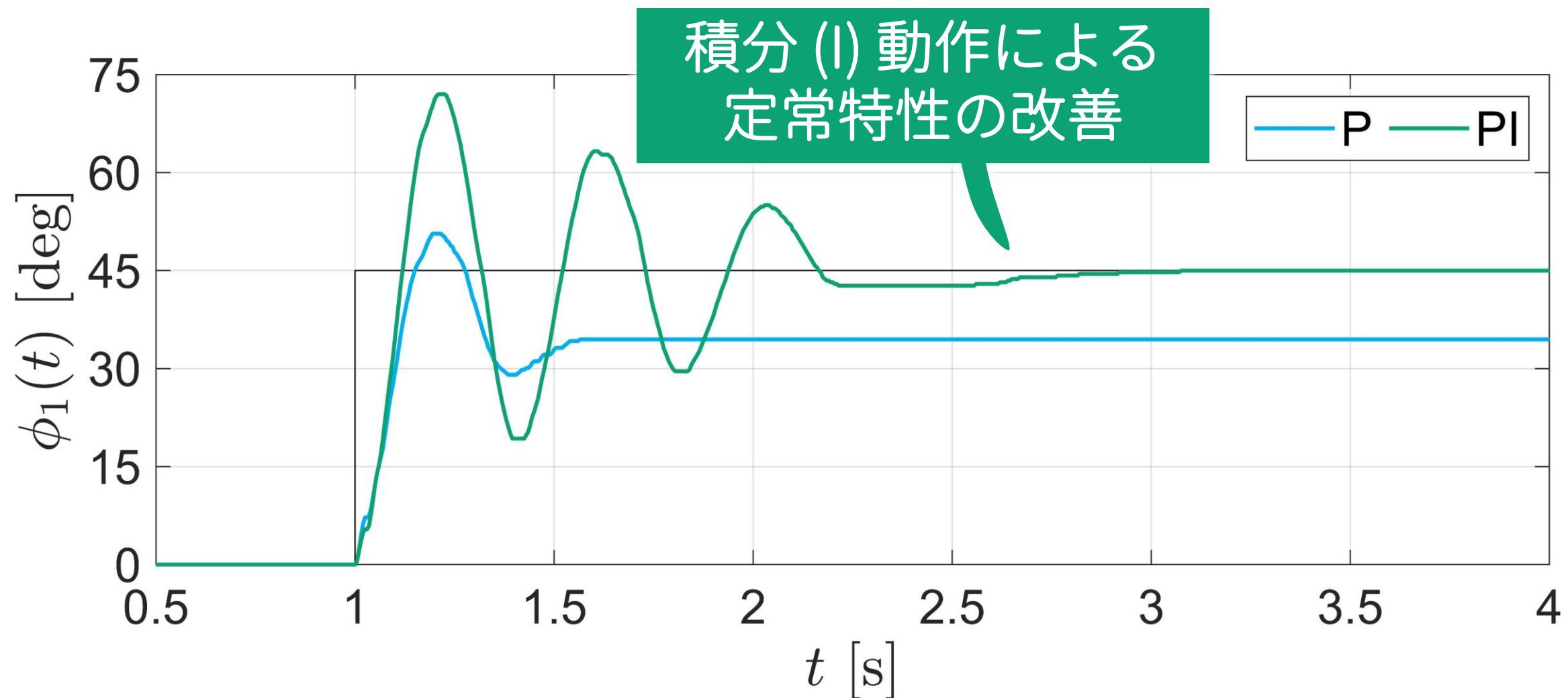
$$k_P = 10$$



$$\begin{aligned} k_P &= 10 \\ k_I &= 60 \end{aligned}$$



積分 (I) 動作の効果



積分(I) 動作の効果

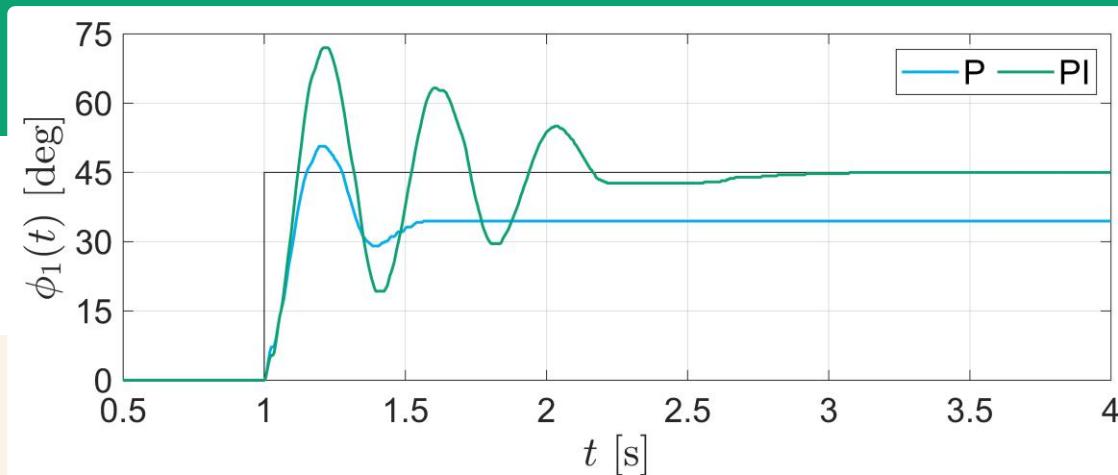
積分動作を加えることにより …



- 定常偏差を 0 にする働きがある！



- 振動的になる！
→ 大きくしそぎると 不安定 になる！



ラウス・フルビツの安定判別法

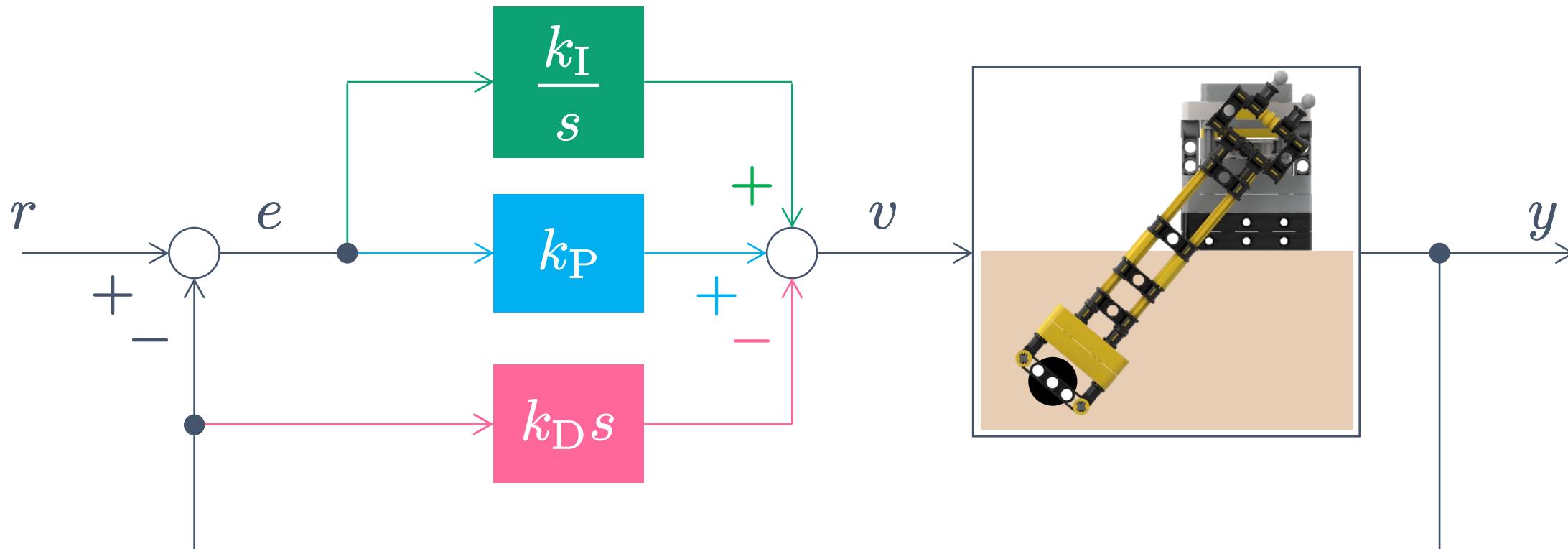
$$\left\{ \begin{array}{l} P(s) = \frac{1}{\alpha_1 s^2 + \beta_1 s + \gamma_1} \\ C(s) = \frac{k_P s + k_I}{s} \end{array} \right. \longrightarrow 0 < k_I < \frac{\beta_1(\gamma_1 + k_P)}{\alpha_1}$$

微分 (D) 動作の効果

PI-D (微分先行型 PID) コントローラ

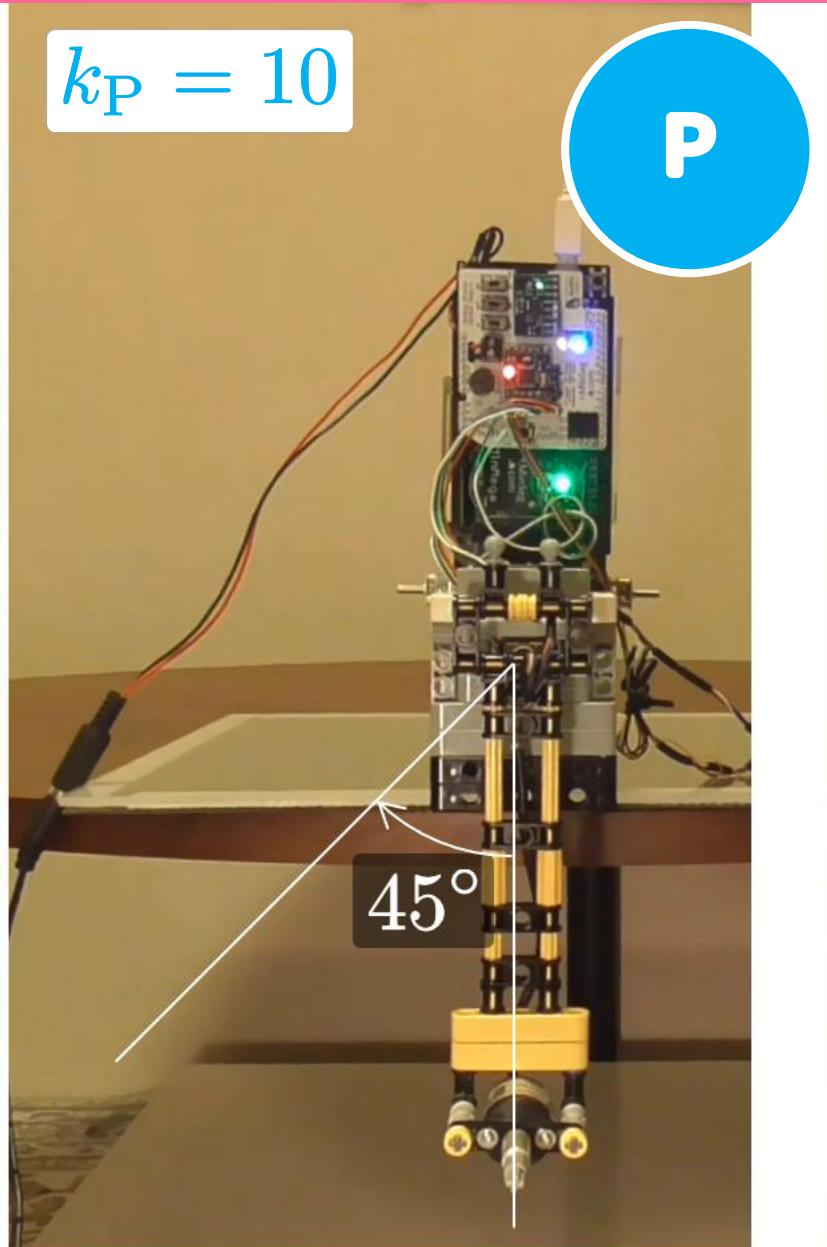
$$v(t) = k_P e(t) + k_I \int_0^t e(t) dt - k_D \frac{dy(t)}{dt}$$

周期振動
の抑制！



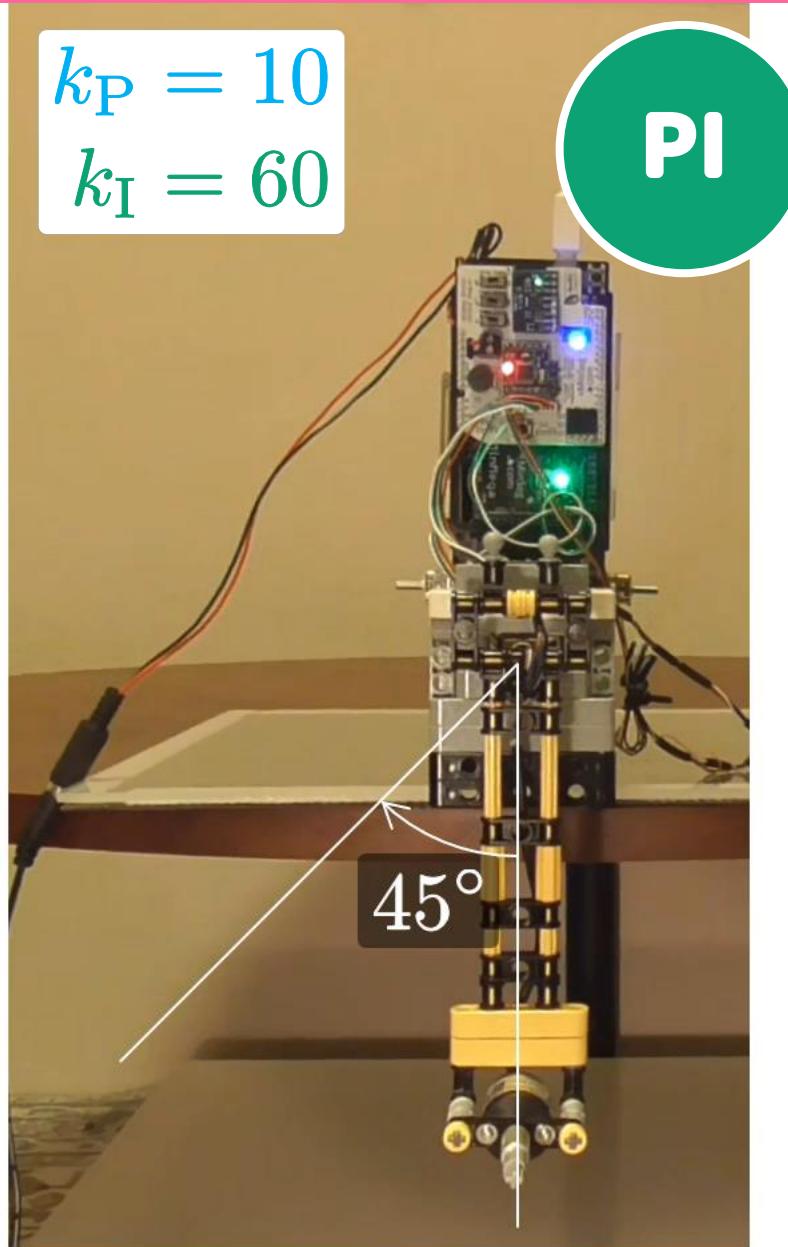
微分 (D) 動作の効果

$k_P = 10$



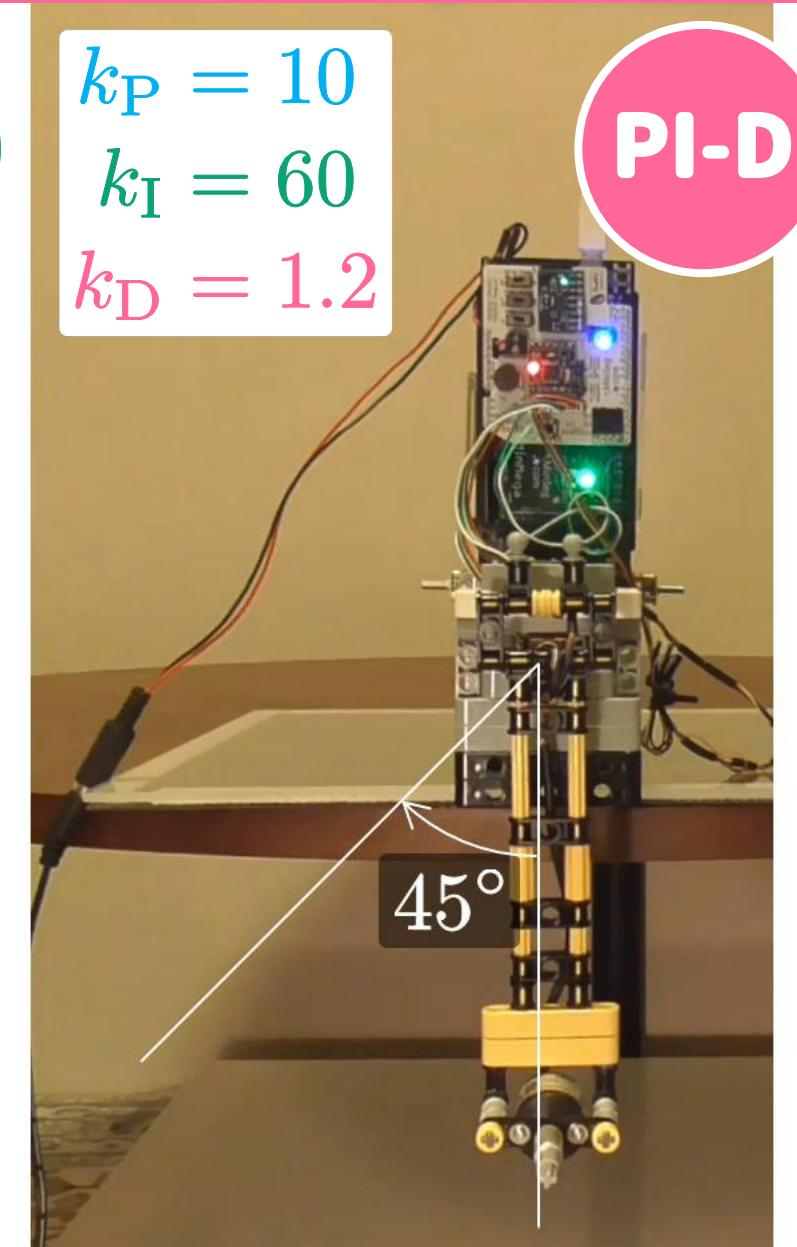
P

$k_P = 10$
 $k_I = 60$



PI

$k_P = 10$
 $k_I = 60$
 $k_D = 1.2$

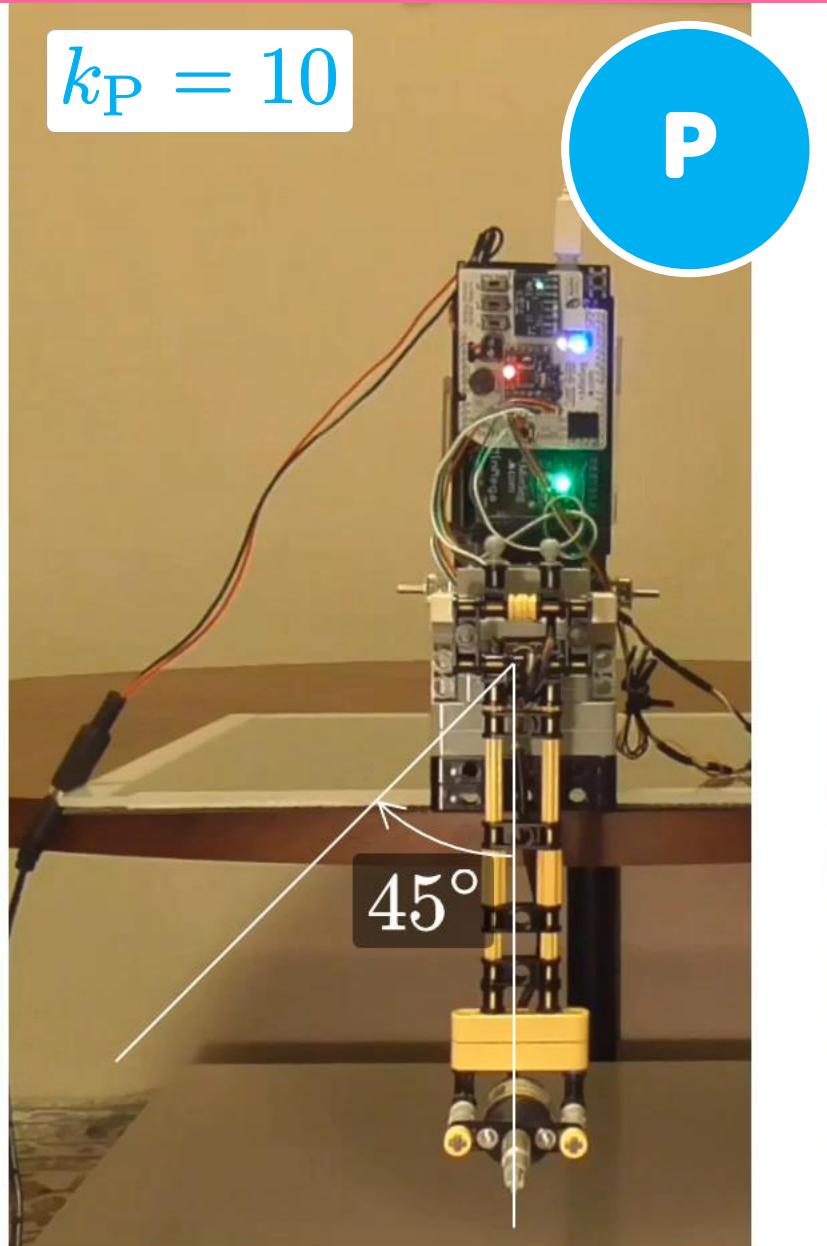


PI-D

微分 (D) 動作の効果

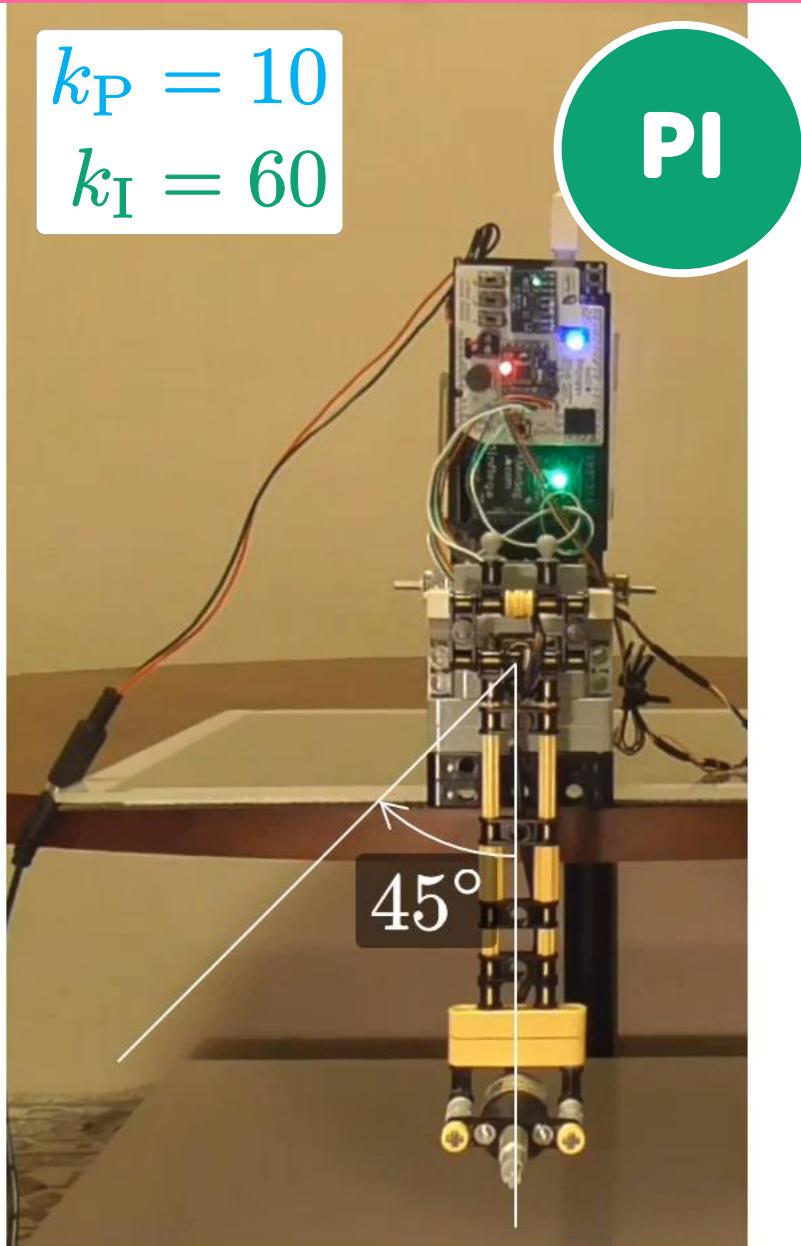
0.25 倍速

$k_P = 10$



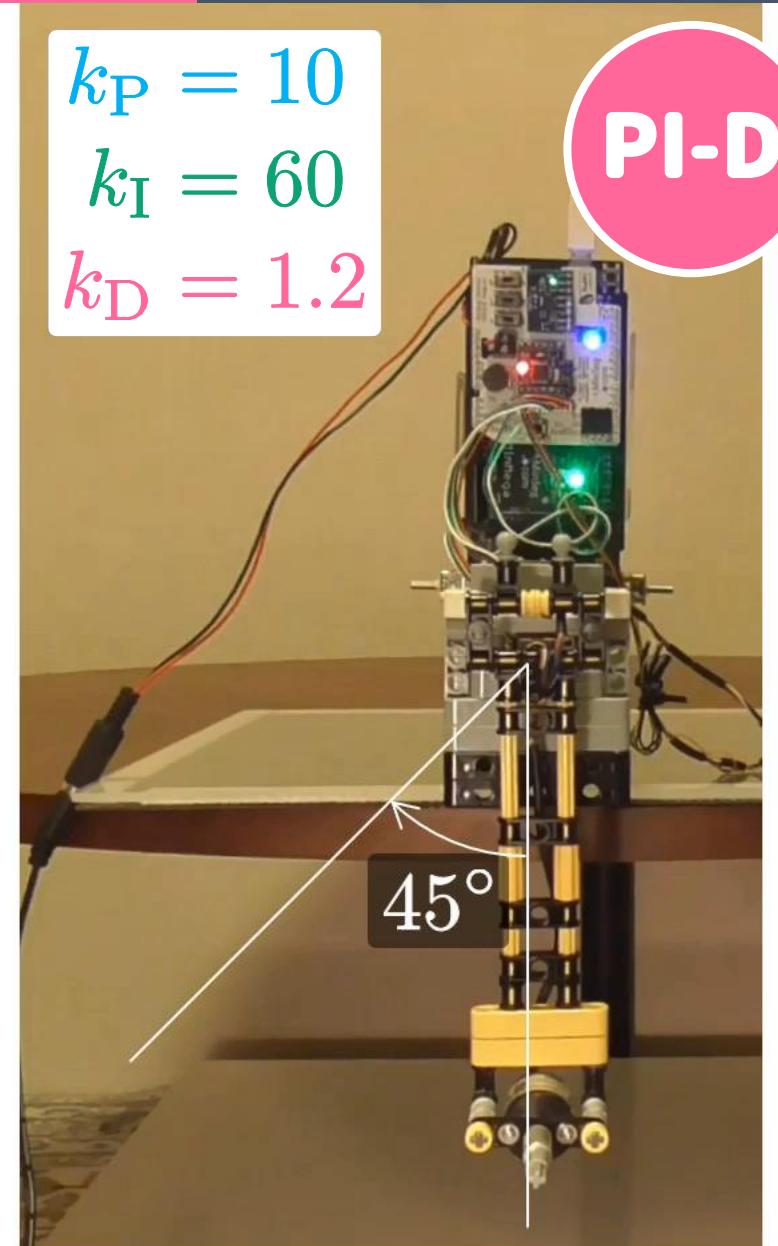
P

$k_P = 10$
 $k_I = 60$



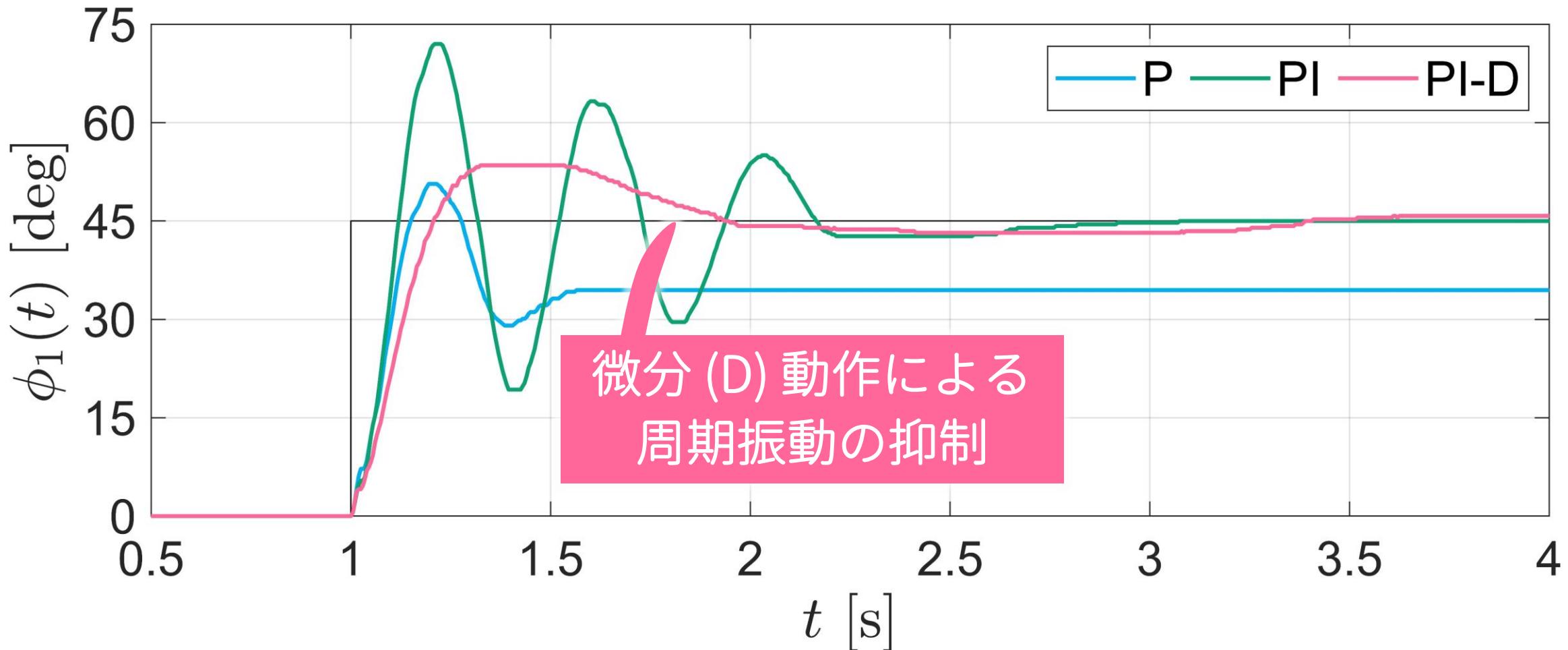
PI

$k_P = 10$
 $k_I = 60$
 $k_D = 1.2$

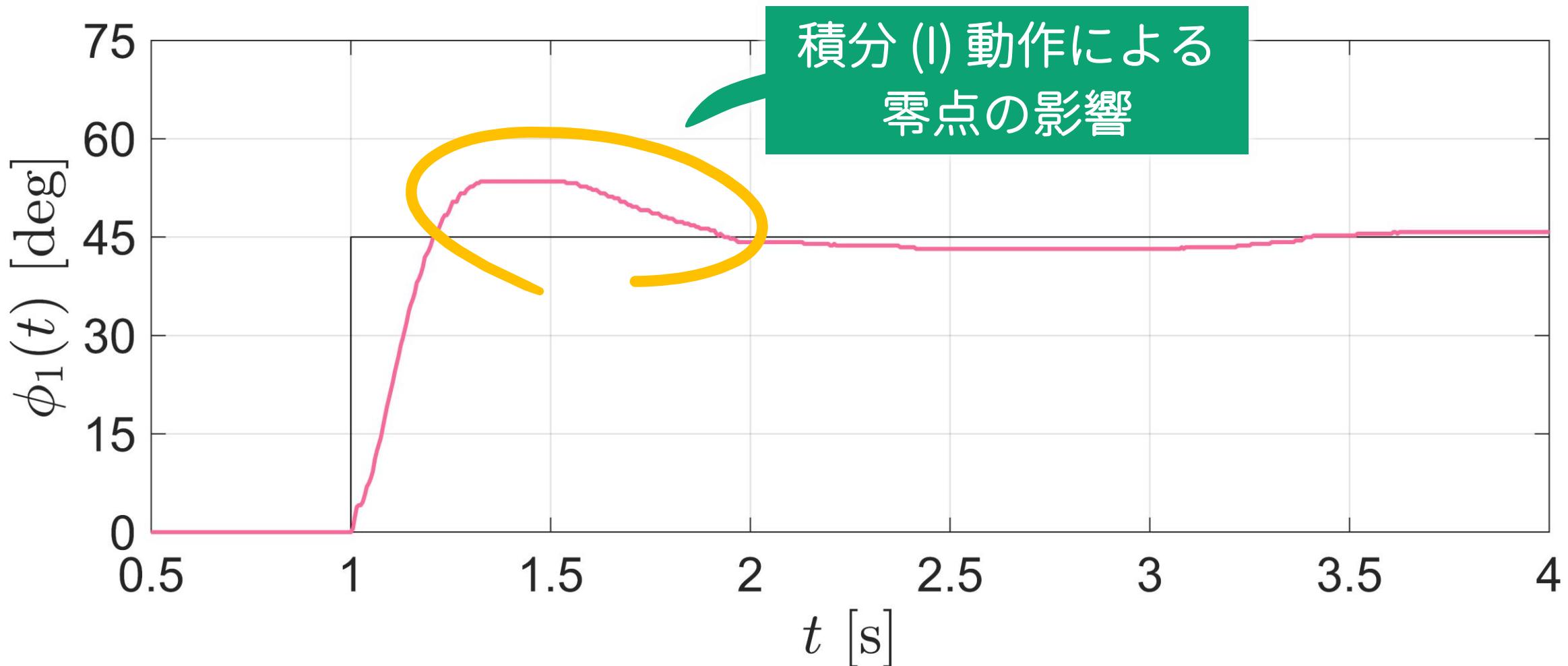


PI-D

微分 (D) 動作の効果



微分 (D) 動作の効果



微分 (D) 動作の効果

微分動作を加えることにより …



- **周期的な振動を抑制**
することが可能！



- 零点に起因するオーバーシュートが残る！

→ 制御対象が 2 次遅れ系なので
極零相殺によりオーバーシュートを 0
とすることも可能！

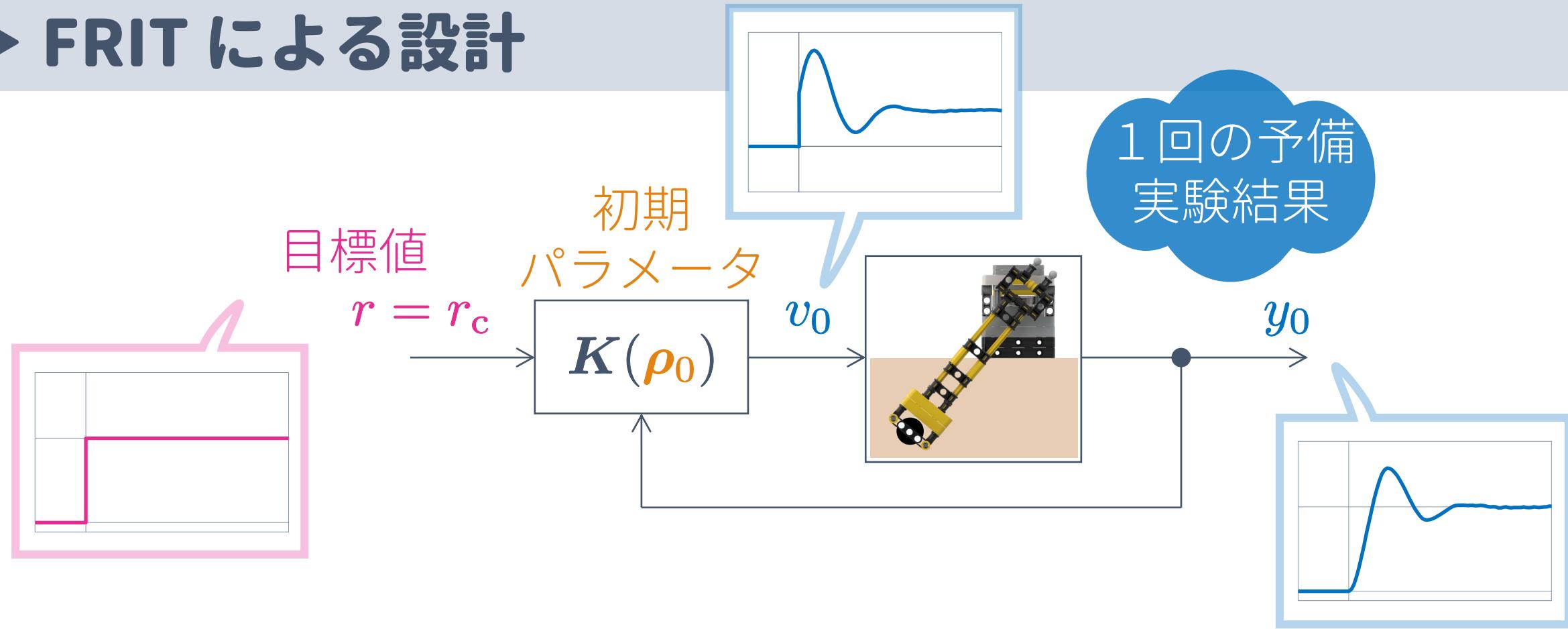
PID 制御の学習 (鉛直面を回転するアームの角度制御)

- ▶ 各動作の効果
- ▶ FRIT による設計
- ▶ モデルに基づく設計
 - 非線形モデル
 - 2次遅れ系の特性に基づくパラメータ同定
 - 最小2乗法に基づくパラメータ同定
 - モデルマッチング法による設計
 - 非線形補償
 - モデル誤差抑制補償 (MEC)



データ駆動型制御
自動チューニング

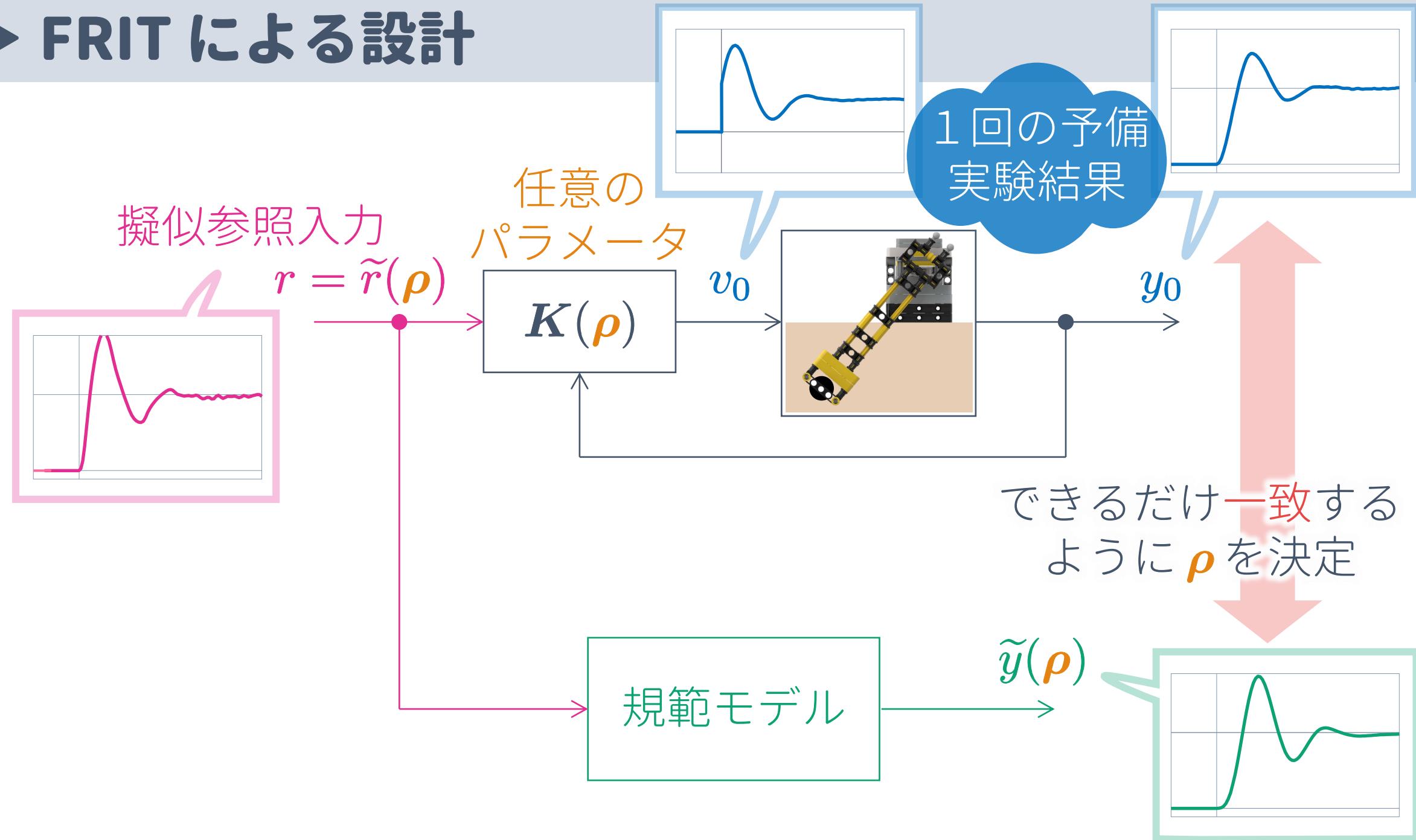
▶ FRIT による設計



擬似参照入力

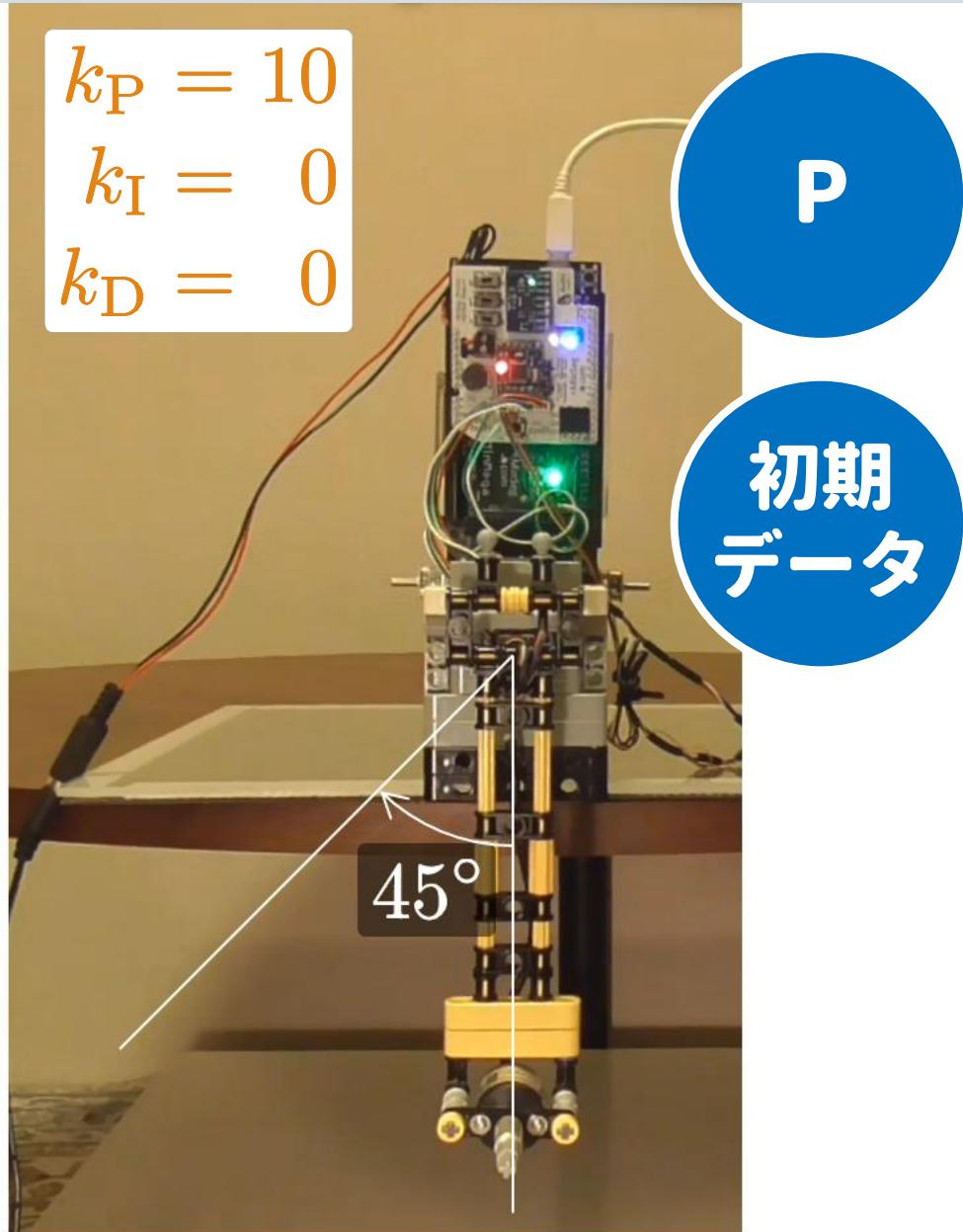
$$\tilde{r}(\rho) = \frac{1}{k_P s + k_I} \left\{ s v_0 + (k_D s^2 + k_P s + k_I) y_0 \right\}, \quad \rho = \begin{bmatrix} k_P \\ k_I \\ k_D \end{bmatrix}$$

► FRIT による設計



► FRIT による設計

$$\begin{aligned} k_P &= 10 \\ k_I &= 0 \\ k_D &= 0 \end{aligned}$$

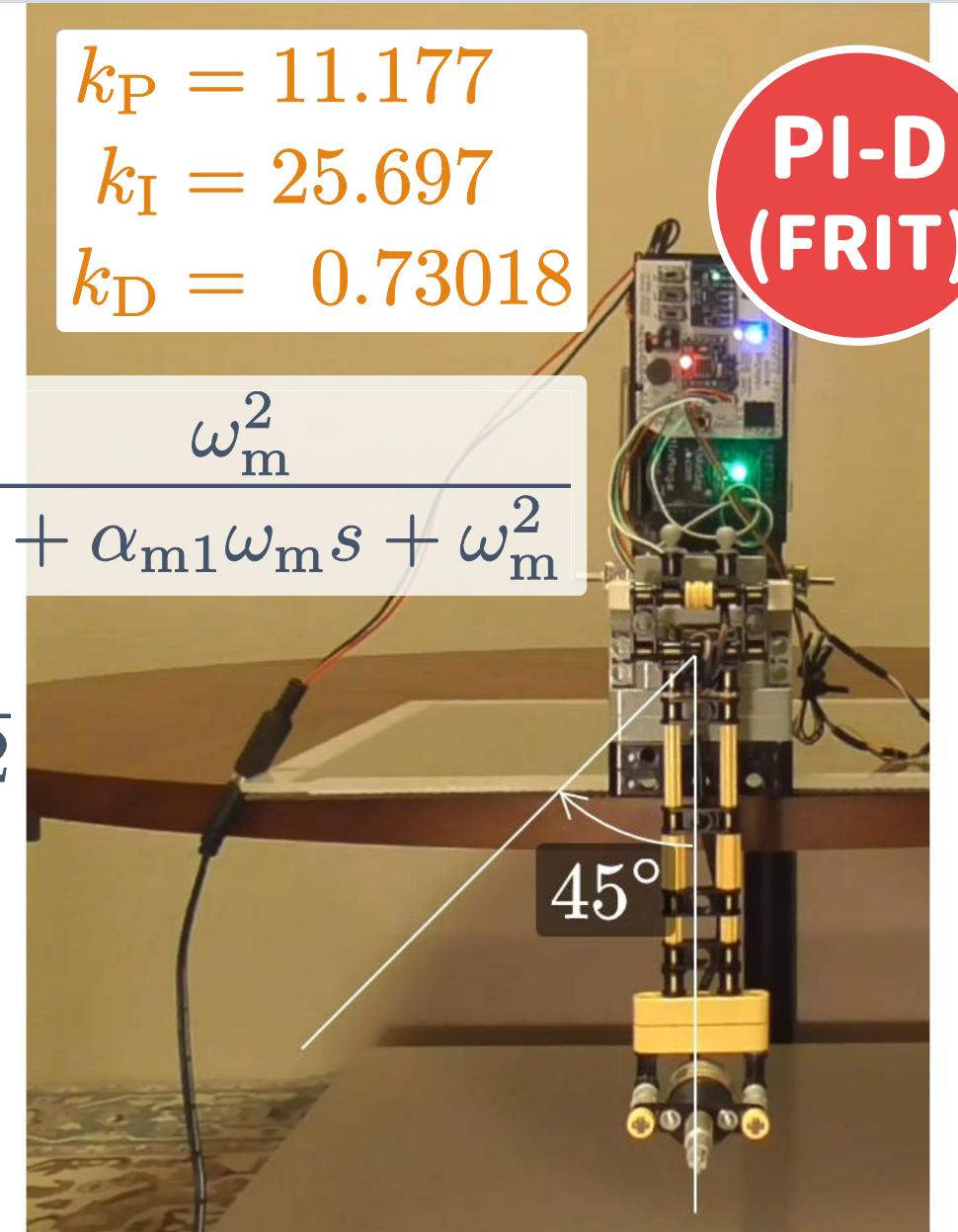


P

初期
データ

$$\begin{aligned} k_P &= 11.177 \\ k_I &= 25.697 \\ k_D &= 0.73018 \end{aligned}$$

$$G_{m2}(s) = \frac{\omega_m^2}{s^2 + \alpha_{m1}\omega_m s + \omega_m^2}$$
$$\left\{ \begin{array}{l} \omega_m = 16 \\ \alpha_{m1} = \sqrt{2} \end{array} \right.$$



PI-D
(FRIT)

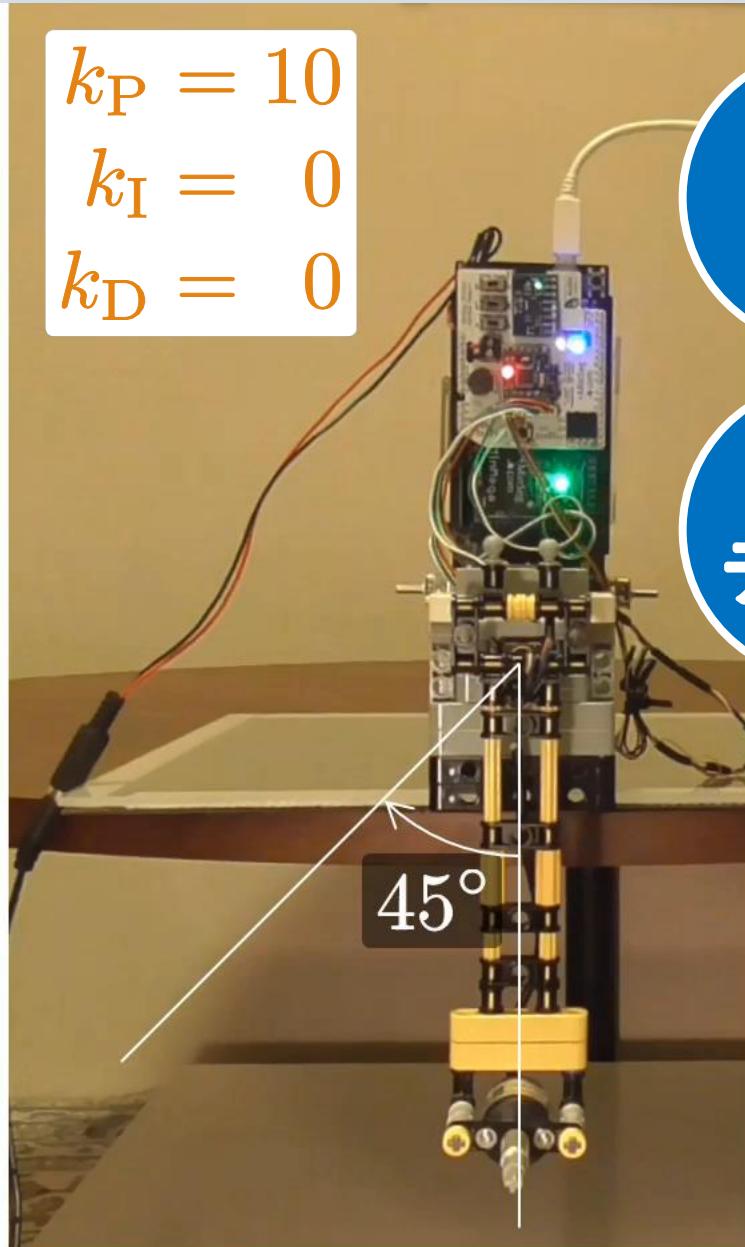
► FRIT による設計

0.25 倍速

$$\begin{aligned}k_P &= 10 \\k_I &= 0 \\k_D &= 0\end{aligned}$$

P

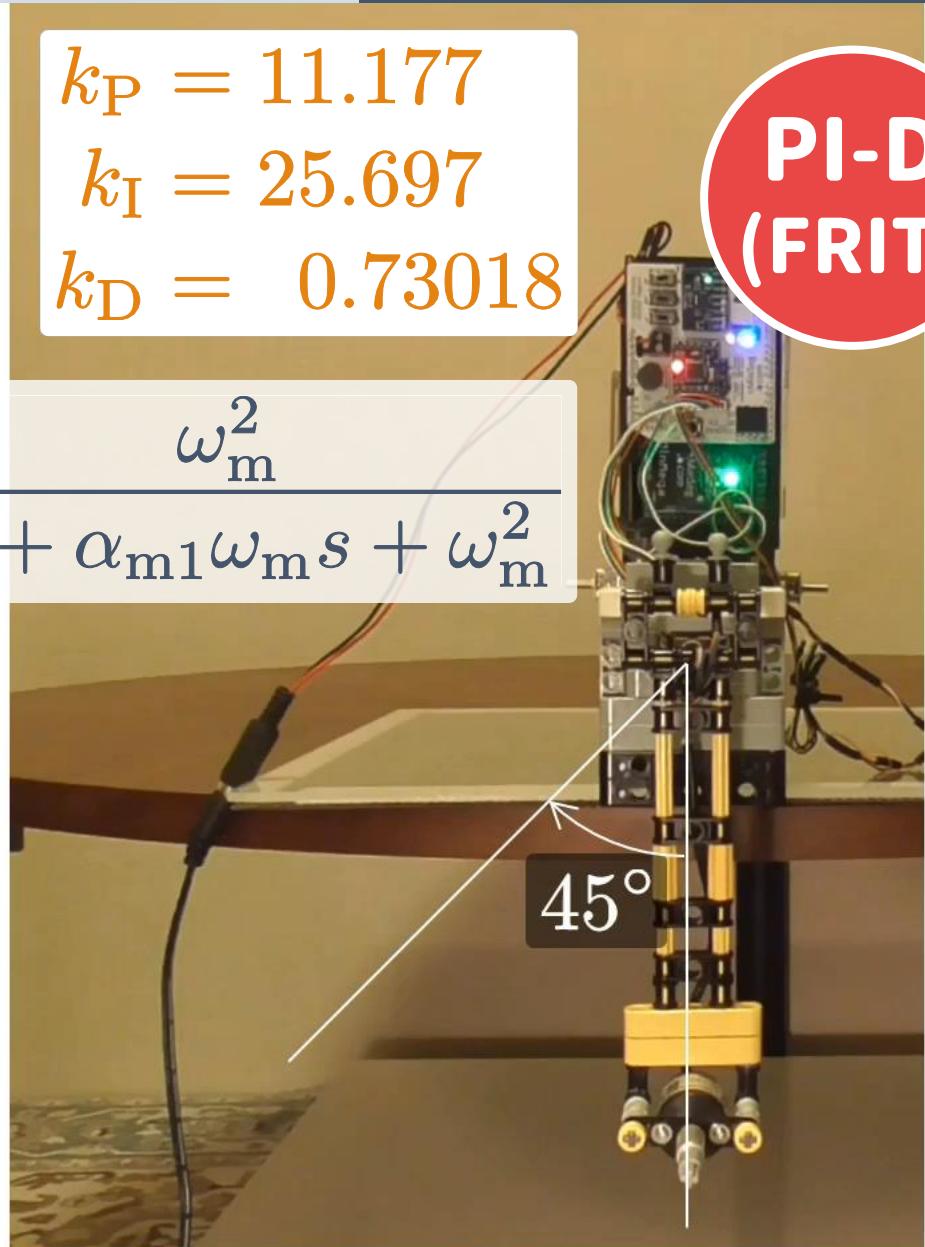
初期
データ



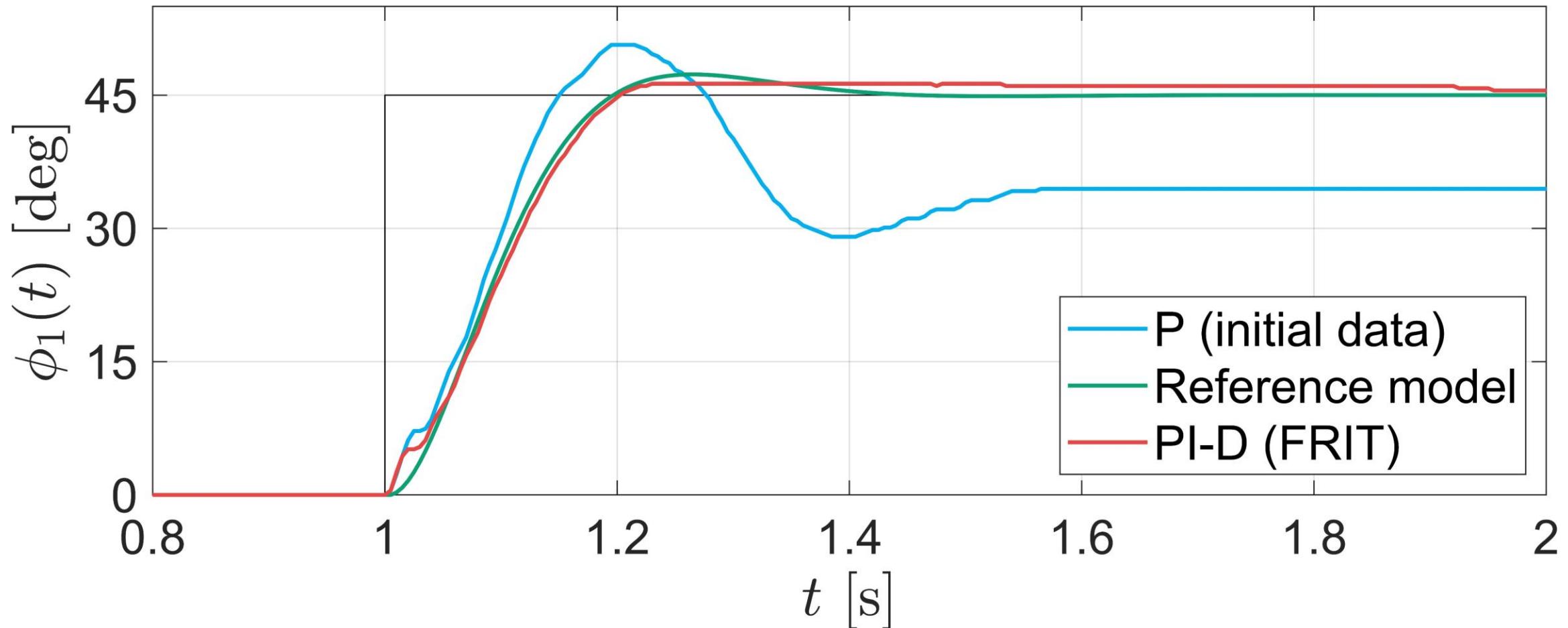
$$\begin{aligned}k_P &= 11.177 \\k_I &= 25.697 \\k_D &= 0.73018\end{aligned}$$

PI-D
(FRIT)

$$G_{m2}(s) = \frac{\omega_m^2}{s^2 + \alpha_{m1}\omega_m s + \omega_m^2}$$
$$\left\{ \begin{array}{l} \omega_m = 16 \\ \alpha_{m1} = \sqrt{2} \end{array} \right.$$



► FRIT による設計



PID 制御の学習 (鉛直面を回転するアームの角度制御)

- ▶ 各動作の効果
- ▶ FRIT による設計
- ▶ モデルに基づく設計
 - **非線形モデル**
 - 2次遅れ系の特性に基づくパラメータ同定
 - 最小2乗法に基づくパラメータ同定
 - モデルマッチング法による設計
 - 非線形補償
 - モデル誤差抑制補償 (MEC)

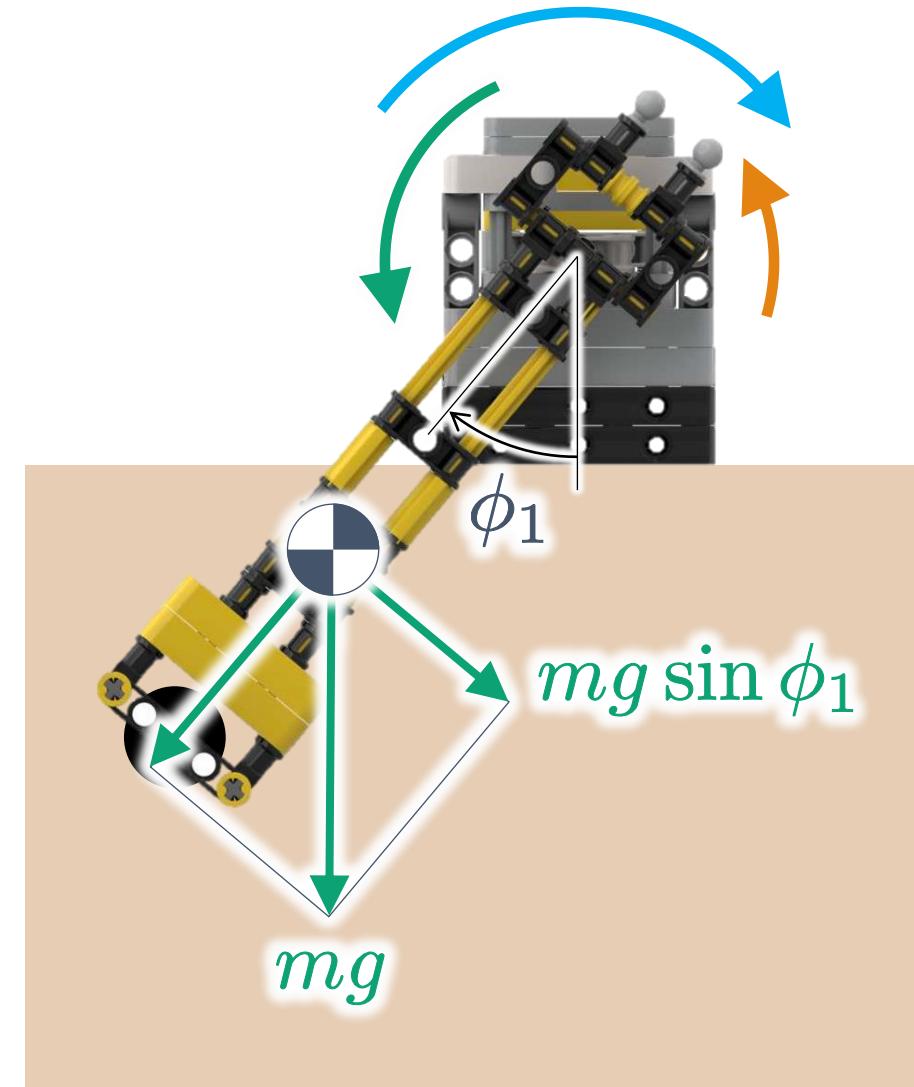
● 非線形モデル

非線形モデル

$$\alpha_1 \ddot{\phi}_1 = -\beta_1 \dot{\phi}_1 - \gamma_1 \sin \phi_1 + v$$

慣性項 摩擦項 重力項 入力項

$$\mathbf{p}_1 = \begin{bmatrix} \alpha_1 \\ \beta_1 \\ \gamma_1 \end{bmatrix} : \text{未知パラメータ}$$



PID 制御の学習 (鉛直面を回転するアームの角度制御)

- ▶ 各動作の効果
- ▶ FRIT による設計
- ▶ モデルに基づく設計

- 非線形モデル

- **2次遅れ系の特性に基づくパラメータ同定**

- 最小2乗法に基づくパラメータ同定

- モデルマッチング法による設計

- 非線形補償

- モデル誤差抑制補償 (MEC)



2次遅れ

● 2次遅れ系の特性に基づくパラメータ同定

非線形モデル

$$\alpha_1 \ddot{\phi}_1 = -\beta_1 \dot{\phi}_1 - \gamma_1 \underline{\sin \phi_1} + v$$

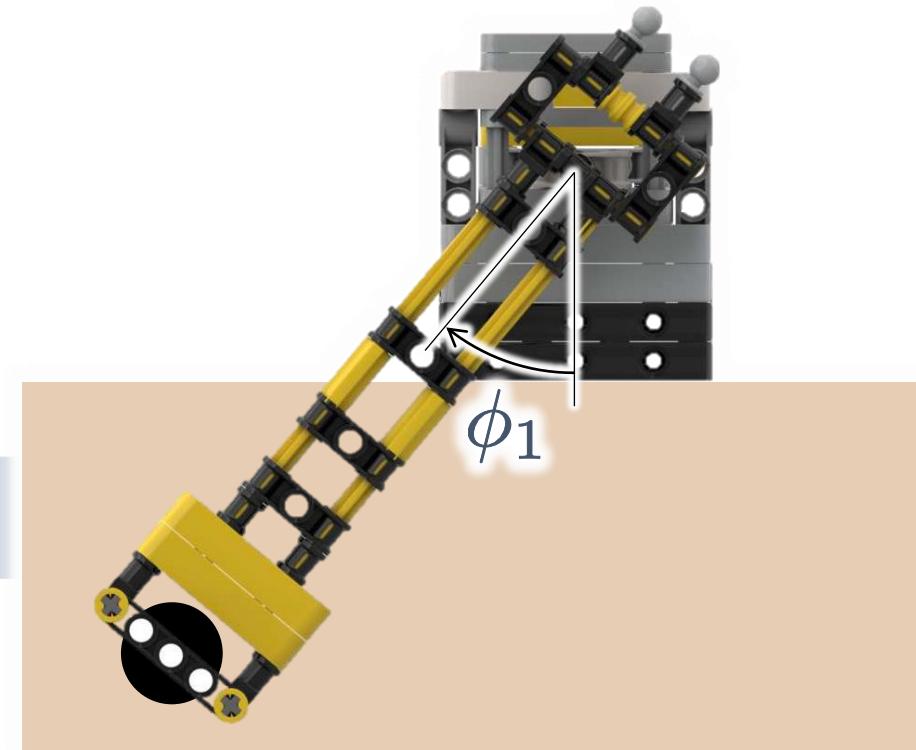


$$\sin \phi_1 \simeq \phi_1 \quad (\phi_1 \simeq 0)$$

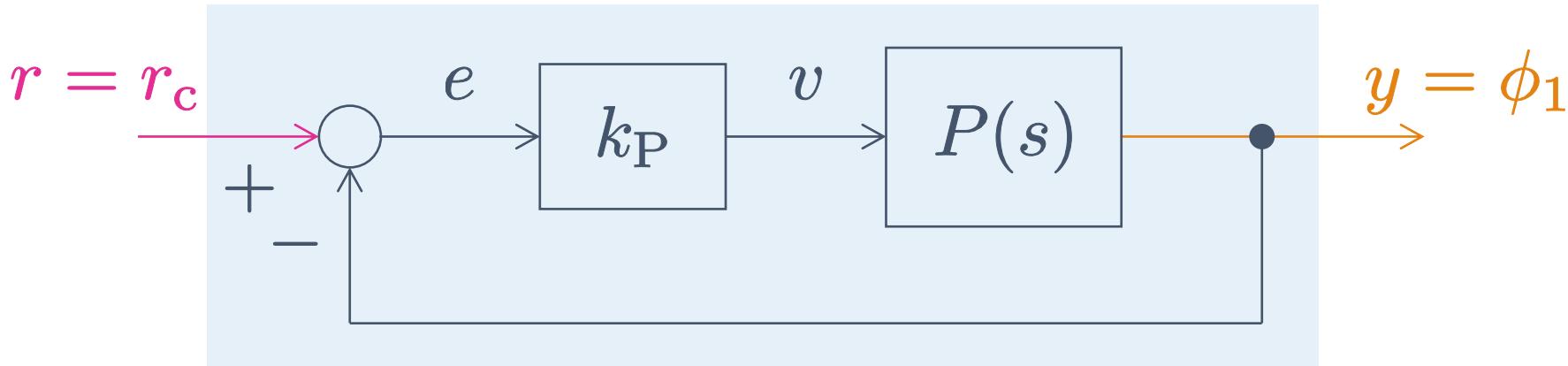
線形化モデル(真下近傍)

$$\alpha_1 \ddot{\phi}_1 = -\beta_1 \dot{\phi}_1 - \gamma_1 \underline{\phi_1} + v$$

$$\phi_1(s) = P(s)v(s), \quad P(s) = \frac{1}{\alpha_1 s^2 + \beta_1 s + \gamma_1}$$



● 2次遅れ系の特性に基づくパラメータ同定



2次遅れ系

$$y(s) = G_{yr}(s)r(s)$$

$$\begin{aligned}G_{yr}(s) &= \frac{k_P}{\alpha_1 s^2 + \beta_1 s + \gamma_1 + k_P} \\&= \frac{K_1 \omega_{n1}^2}{s^2 + 2\zeta_1 \omega_{n1} s + \omega_{n1}^2}\end{aligned}$$

● 2次遅れ系の特性に基づくパラメータ同定

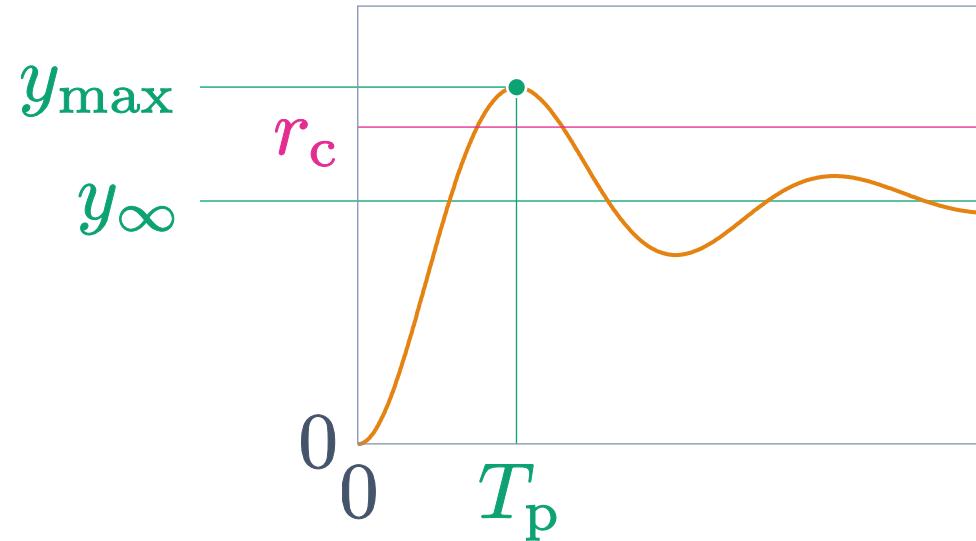
$$r = r_c$$

$$\frac{K_1 \omega_{n1}^2}{s^2 + 2\zeta_1 \omega_{n1} s + \omega_{n1}^2}$$

$$y = \phi_1$$

パラメータ同定

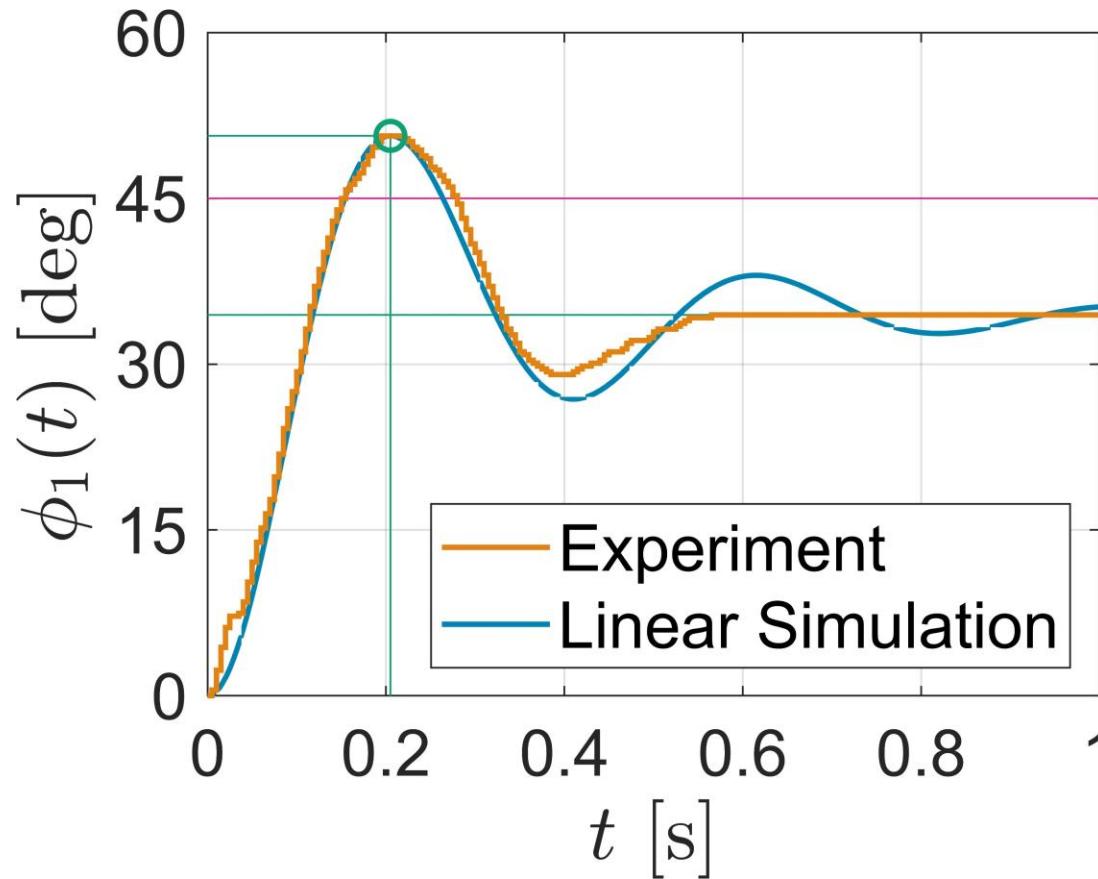
$$\begin{cases} y_\infty \\ T_p \\ y_{\max} \end{cases} \rightarrow \begin{cases} K_1 \\ \omega_{n1} \\ \zeta_1 \end{cases} \rightarrow \begin{cases} \alpha_1 \\ \beta_1 \\ \gamma_1 \end{cases}$$



オーバーシュート
を生じる

● 2次遅れ系の特性に基づくパラメータ同定

2次遅れ系の特性に基づく方法



PID 制御の学習 (鉛直面を回転するアームの角度制御)

- ▶ 各動作の効果
- ▶ FRIT による設計
- ▶ モデルに基づく設計

- 非線形モデル
- 2次遅れ系の特性に基づくパラメータ同定
- **最小 2 乗法に基づくパラメータ同定**
- モデルマッチング法による設計
- 非線形補償
- モデル誤差抑制補償 (MEC)



**最小 2 乗法
数値微分
フィルタ処理**

● 最小2乗法に基づくパラメータ同定

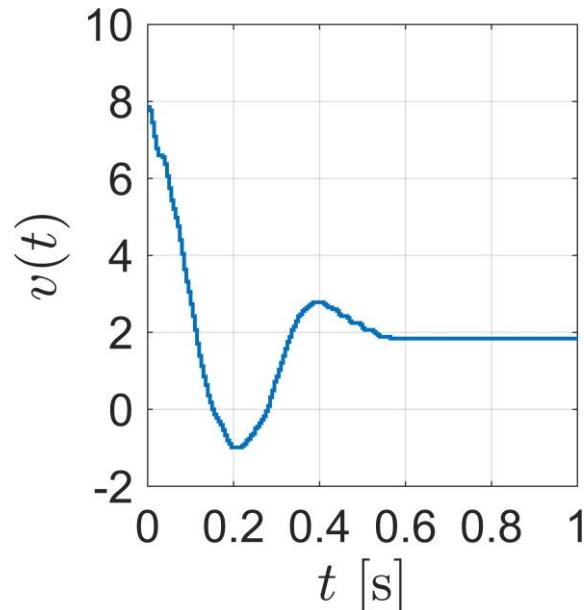
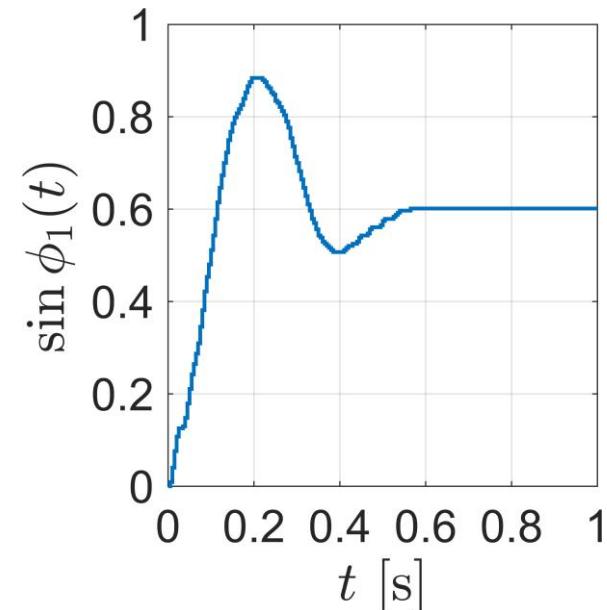
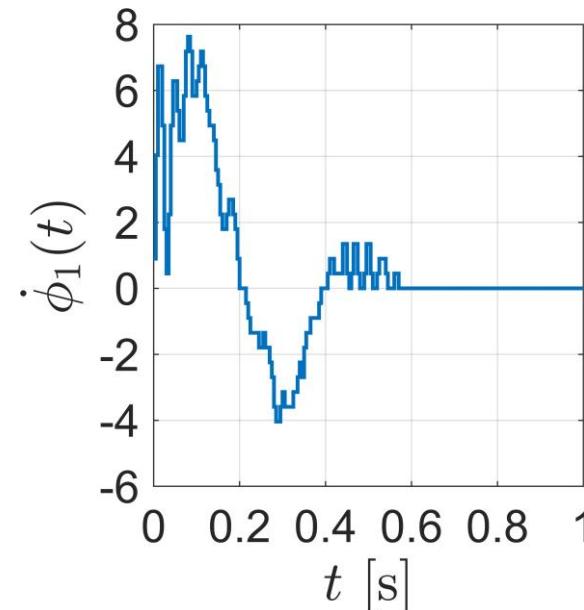
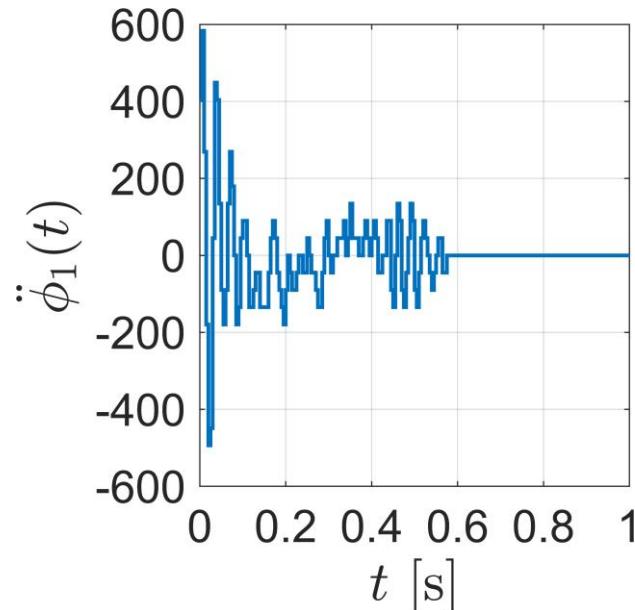
非線形モデル

$$\alpha_1 \ddot{\phi}_1 = -\beta_1 \dot{\phi}_1 - \gamma_1 \sin \phi_1 + v$$

量子化誤差
に起因する
高周波信号



$$\begin{bmatrix} \ddot{\phi}_1(t) & \dot{\phi}_1(t) & \sin \phi_1(t) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \beta_1 \\ \gamma_1 \end{bmatrix} = v(t)$$



● 最小2乗法に基づくパラメータ同定

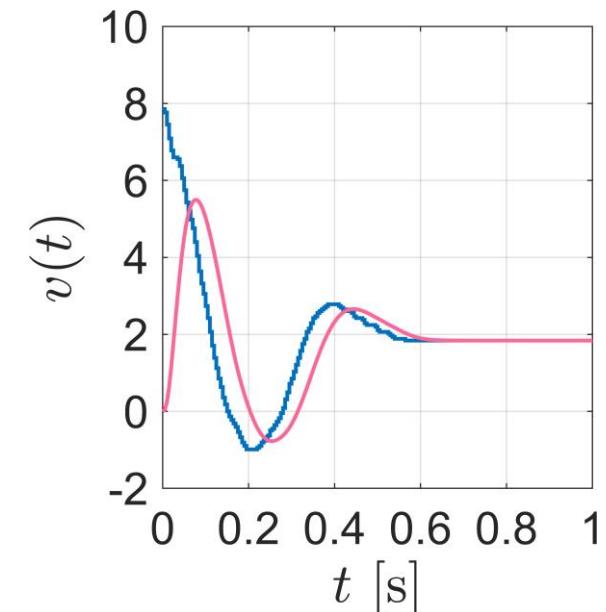
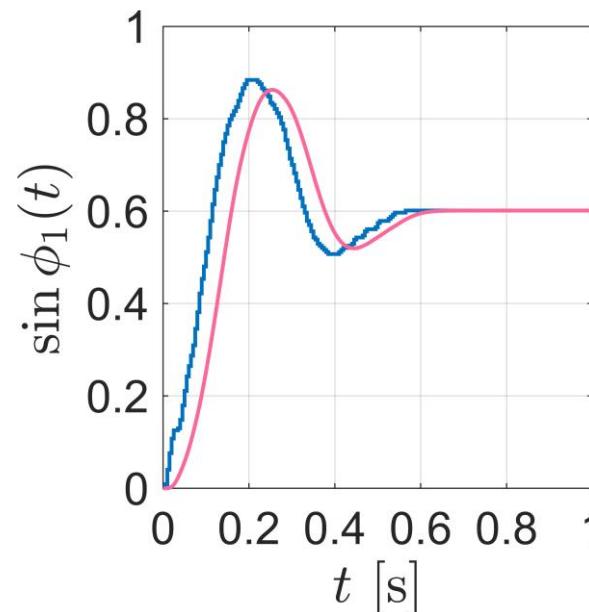
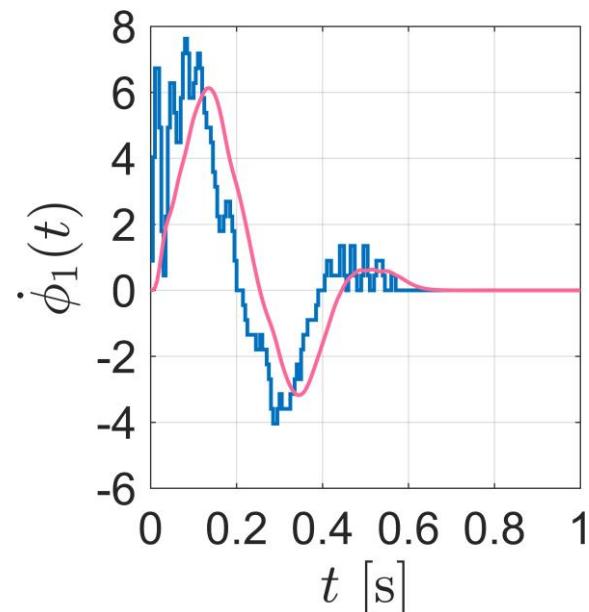
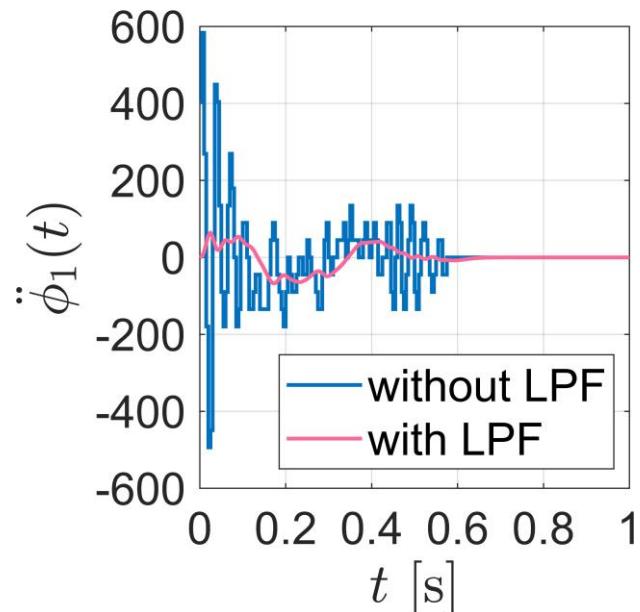
非線形モデル

$$\alpha_1 \ddot{\phi}_1 = -\beta_1 \dot{\phi}_1 - \gamma_1 \sin \phi_1 + v$$

ローパス
フィルタ

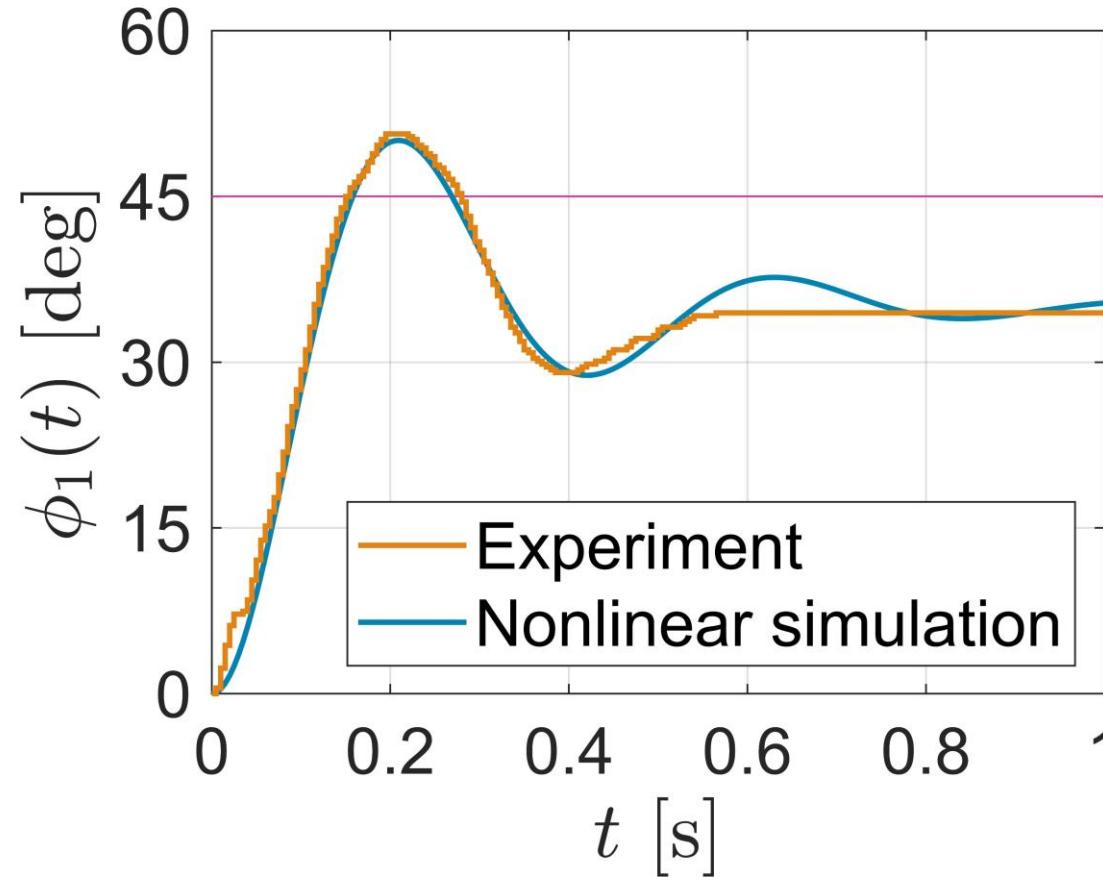
$$G_{f1}(s) = \frac{1}{(1 + T_{f1}s)^3}$$

$$G_{f1}(s) \begin{bmatrix} \ddot{\phi}_1(t) & \dot{\phi}_1(t) & \sin \phi_1(t) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \beta_1 \\ \gamma_1 \end{bmatrix} = G_{f1}(s)v(t)$$

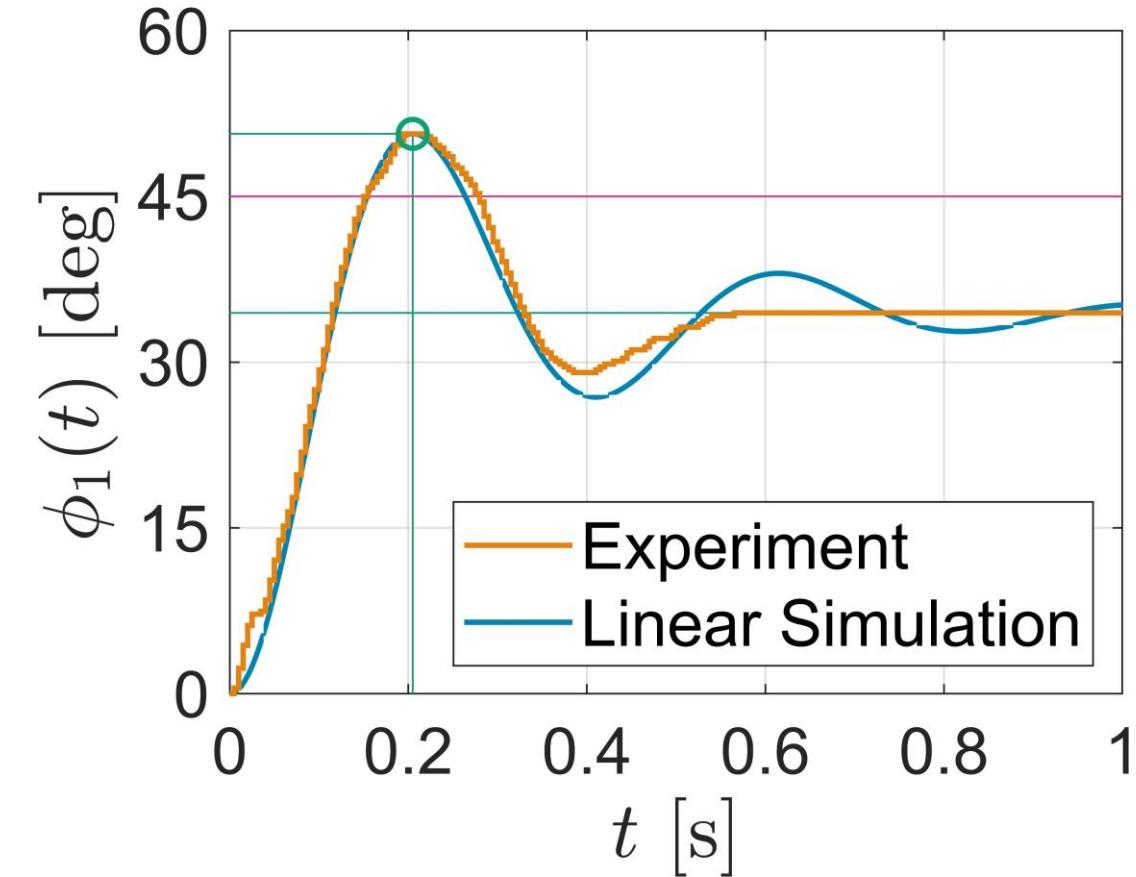


● 最小 2 乗法に基づくパラメータ同定

最小 2 乗法



2 次遅れ系の特性に基づく方法



PID 制御の学習 (鉛直面を回転するアームの角度制御)

- ▶ 各動作の効果
- ▶ FRIT による設計
- ▶ モデルに基づく設計

- 非線形モデル
- 2次遅れ系の特性に基づくパラメータ同定
- 最小2乗法に基づくパラメータ同定
- **モデルマッチング法による設計**
- 非線形補償
- モデル誤差抑制補償 (MEC)

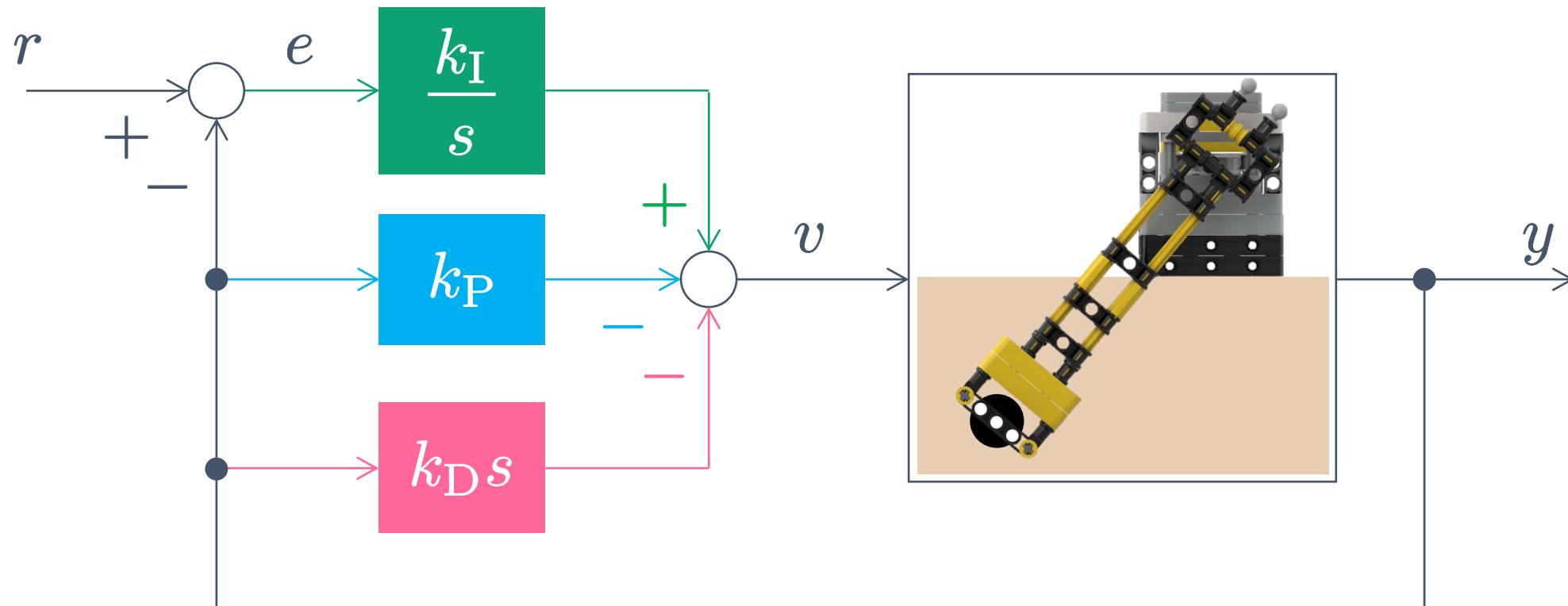


モデル
マッチング

● モデルマッチング法による設計

I-PD (比例・微分先行型 PID) コントローラ

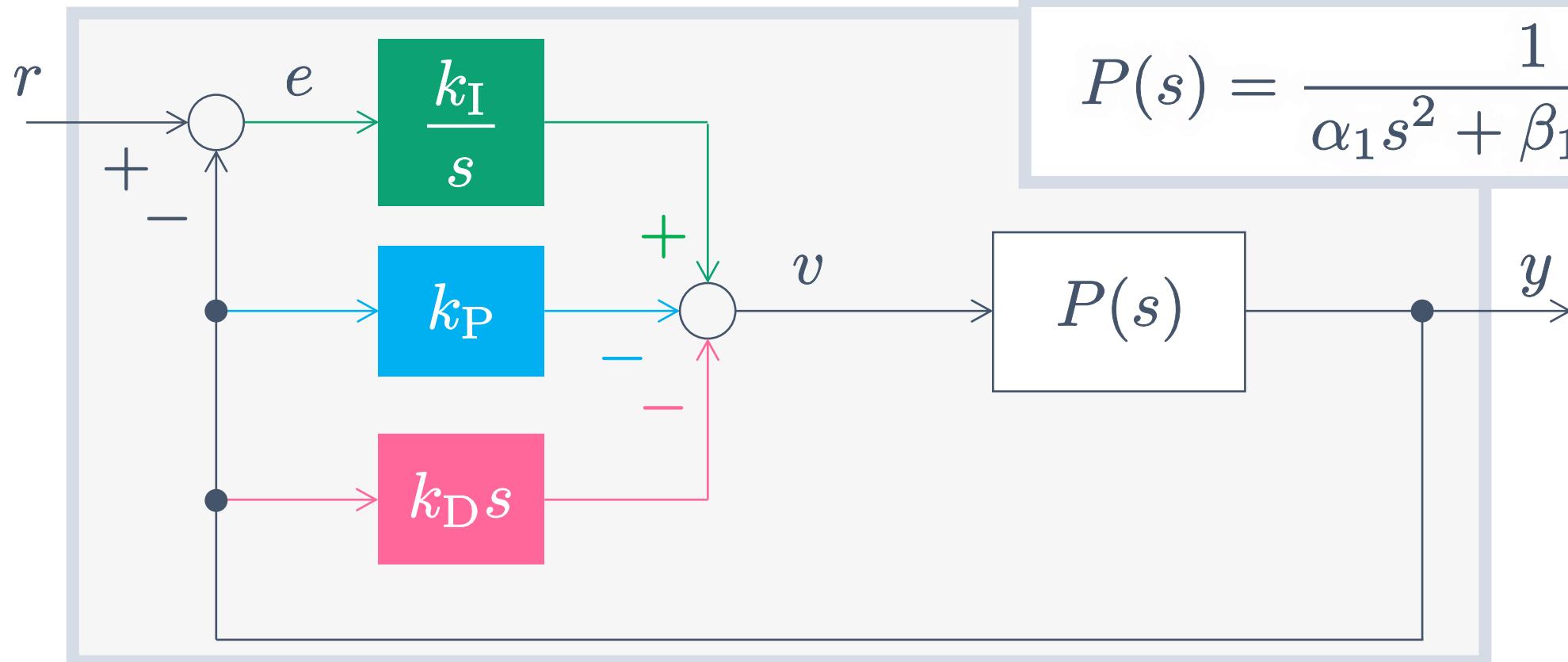
$$v(t) = -k_P e(t) + k_I \int_0^t e(t) dt - k_D \frac{dy(t)}{dt}$$



● モデルマッチング法による設計

$$G_{yr}(s) = \frac{k_I}{\alpha_1 s^3 + (\beta_1 + k_D)s^2 + (\gamma_1 + k_P)s + k_I}$$

線形化モデル (真下近傍)



● モデルマッチング法による設計

$$G_{yr}(s) = \frac{k_I}{\alpha_1 s^3 + (\beta_1 + k_D)s^2 + (\gamma_1 + k_P)s + k_I}$$

一致するように k_P , k_I , k_D を決定



3次の規範モデル(バターワース標準形)

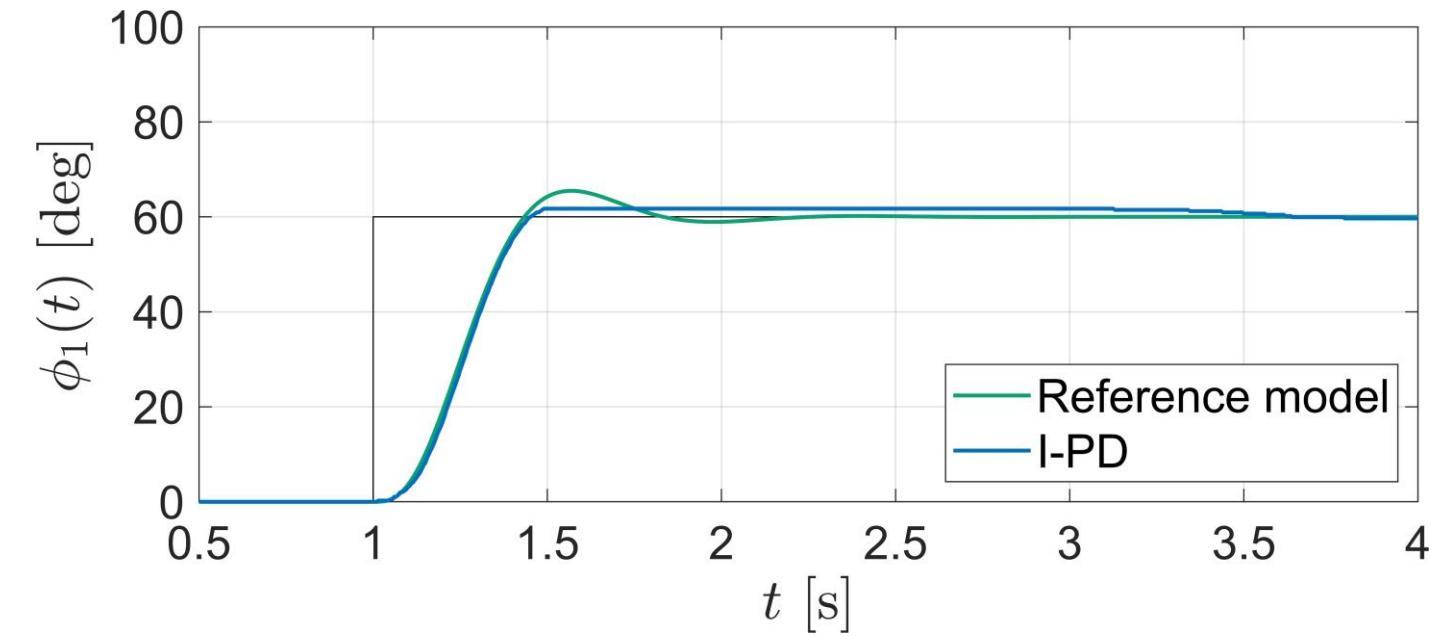
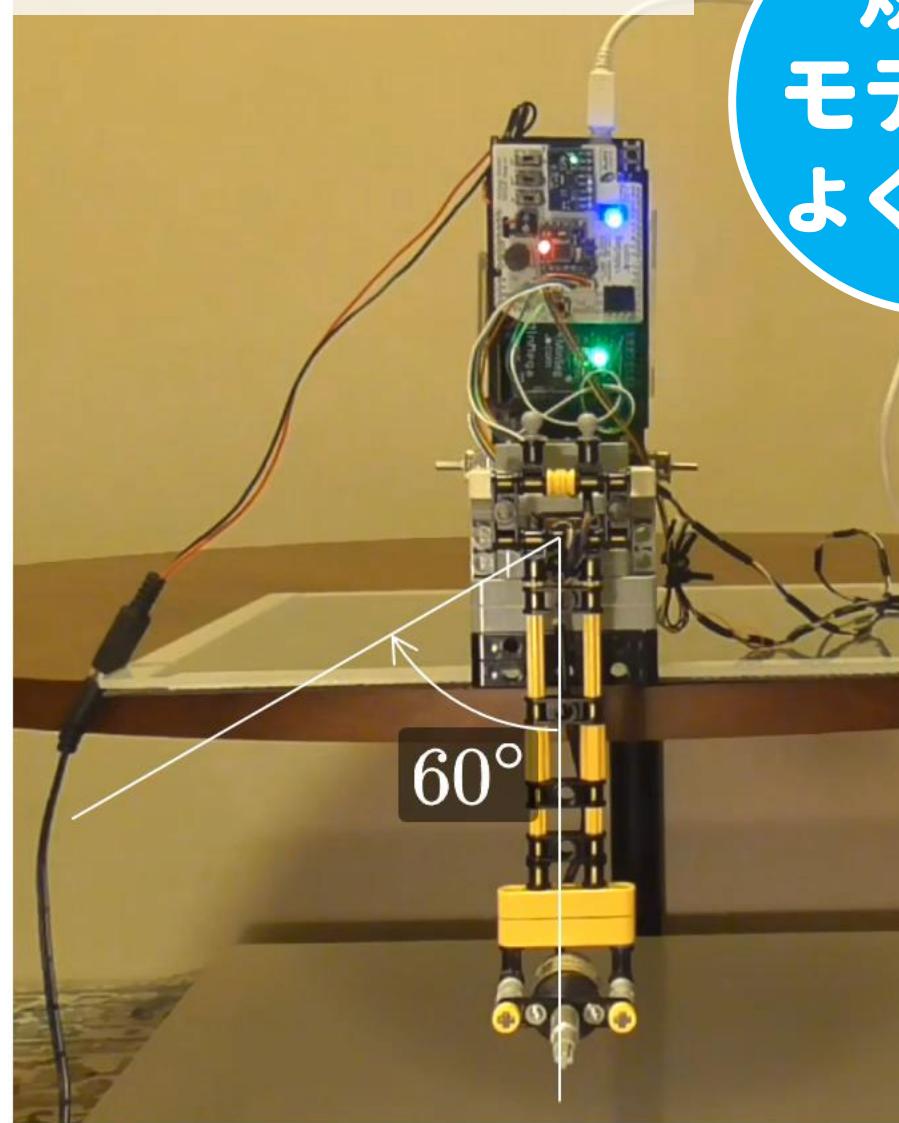
$$M_3(s) = \frac{\omega_m^3}{s^3 + \alpha_{m2}\omega_m s^2 + \alpha_{m1}\omega_m^2 s + \omega_m^3}$$

$$\begin{cases} \omega_m = 8.5 \\ \alpha_{m1} = 2 \\ \alpha_{m2} = 2 \end{cases}$$

● モデルマッチング法による設計

線形化誤差が小さい

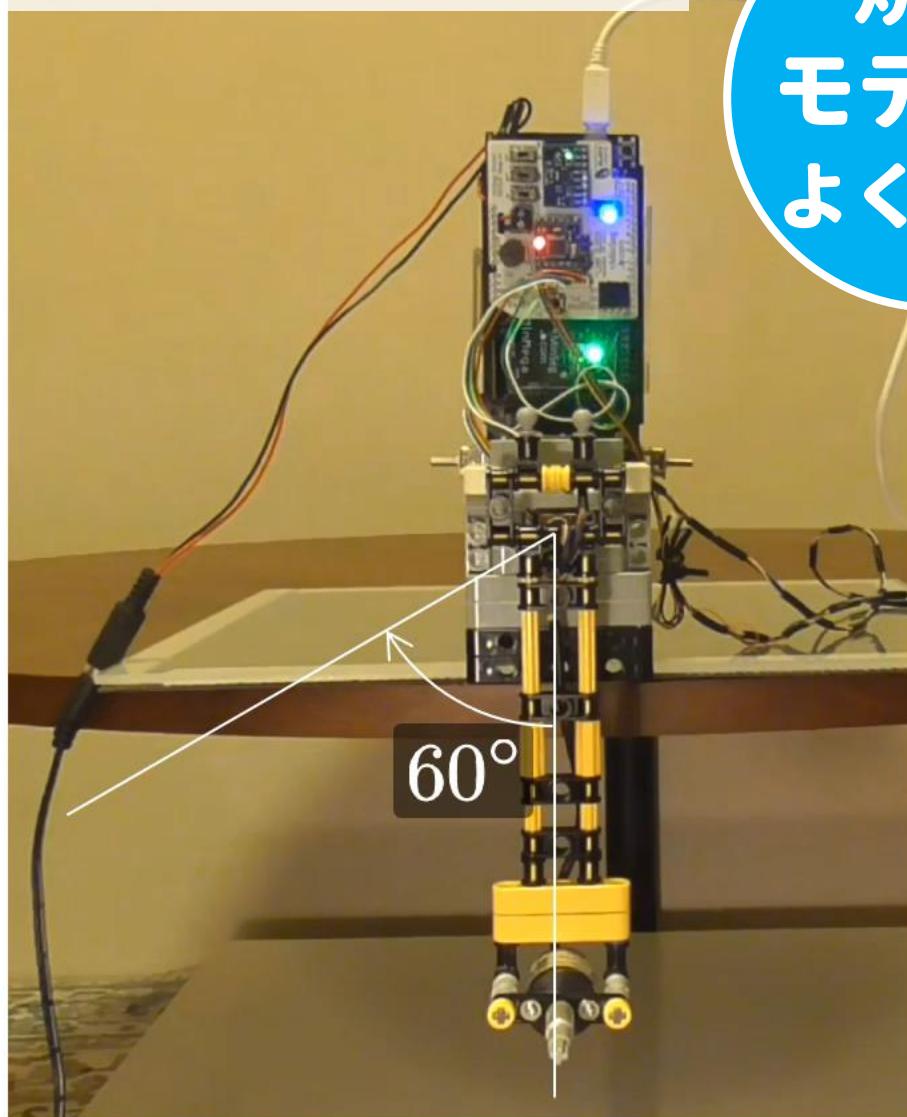
規範
モデルに
よく一致



● モデルマッチング法による設計

線形化誤差が小さい

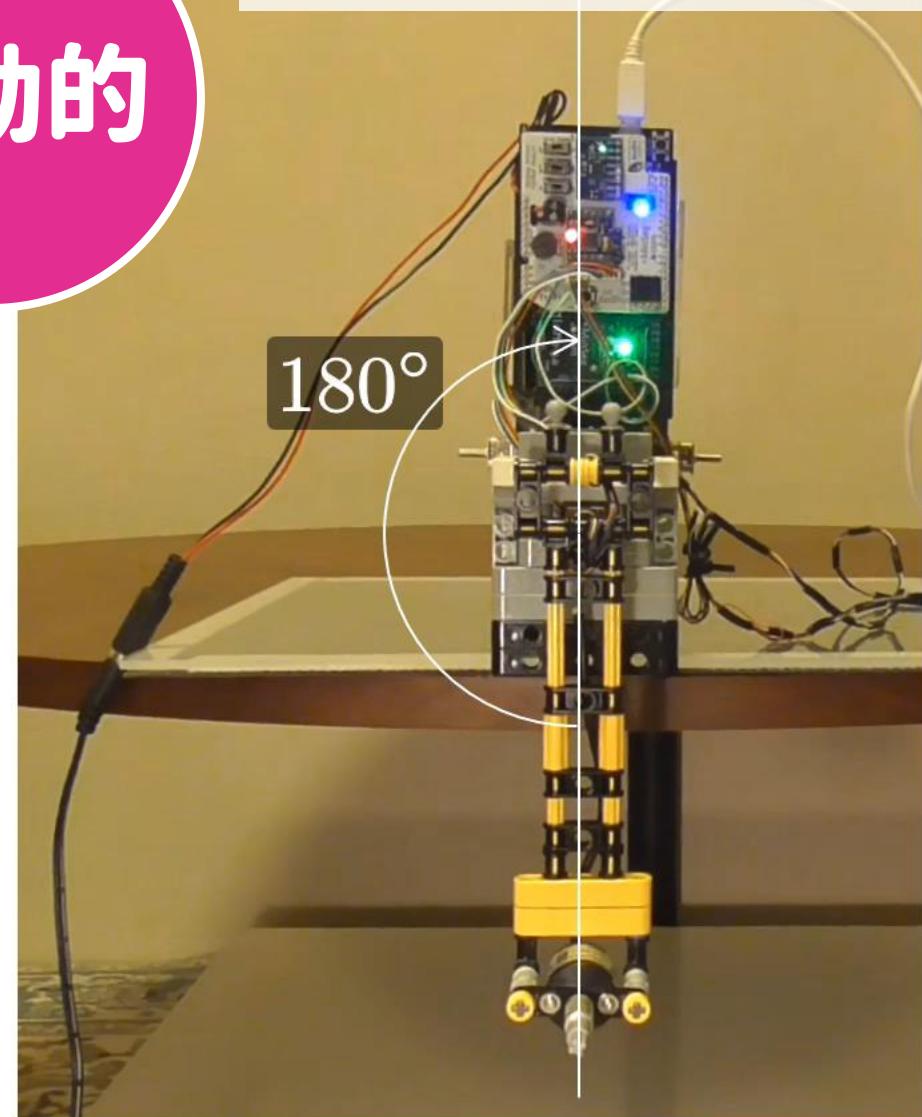
規範
モデルに
よく一致



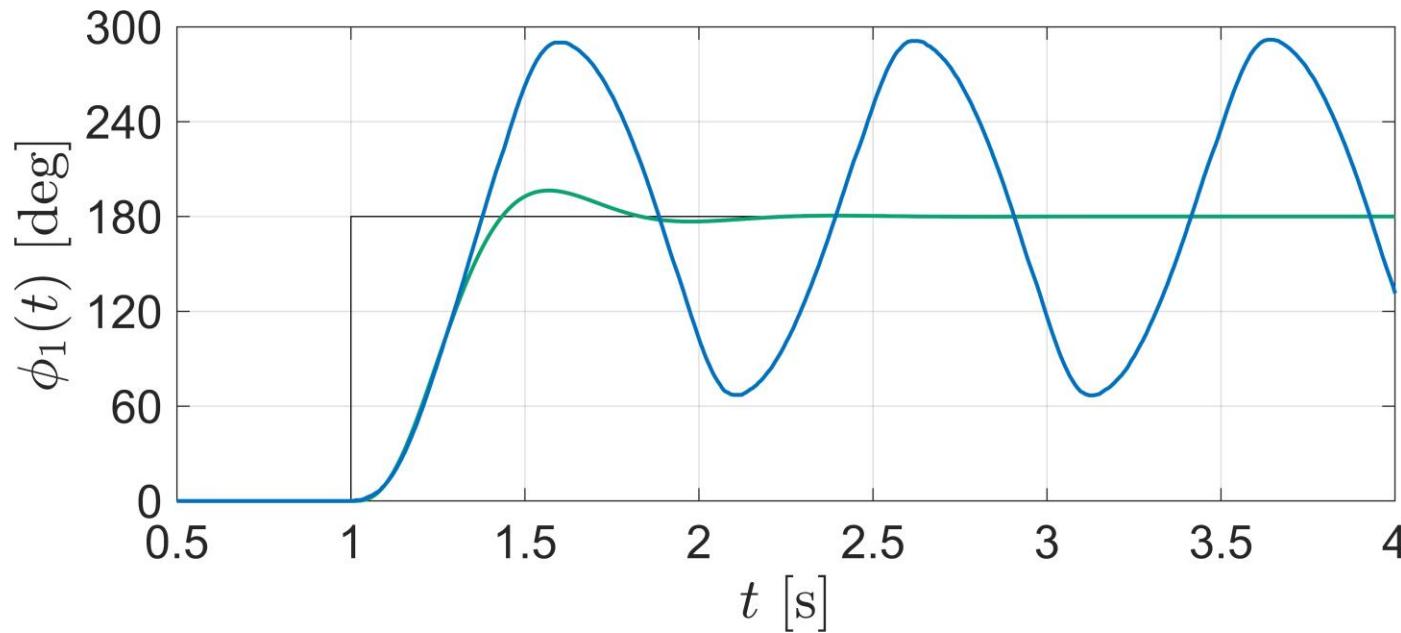
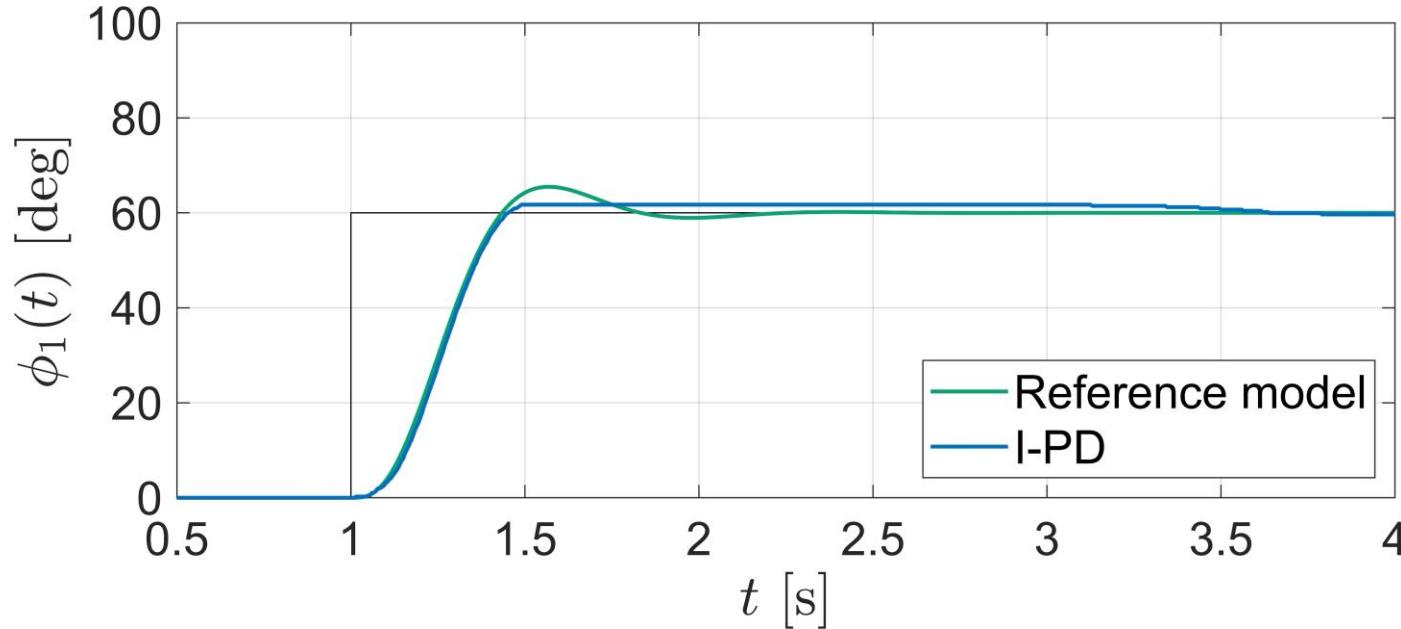
振動的

線形化誤差が大きい

180°



● モデルマッチング法による設計



線形化誤差が小さい

規範
モデルに
よく一致

線形化誤差が大きい

振動的

PID 制御の学習 (鉛直面を回転するアームの角度制御)

- ▶ 各動作の効果
- ▶ FRIT による設計
- ▶ モデルに基づく設計

- 非線形モデル
- 2次遅れ系の特性に基づくパラメータ同定
- 最小2乗法に基づくパラメータ同定
- モデルマッチング法による設計
- **非線形補償**
- モデル誤差抑制補償 (MEC)



**非線形補償
厳密な線形化**

● 非線形補償

非線形モデル

$$\alpha_1 \ddot{\phi}_1 = -\beta_1 \dot{\phi}_1 - \gamma_1 \underline{\sin \phi_1} + v$$

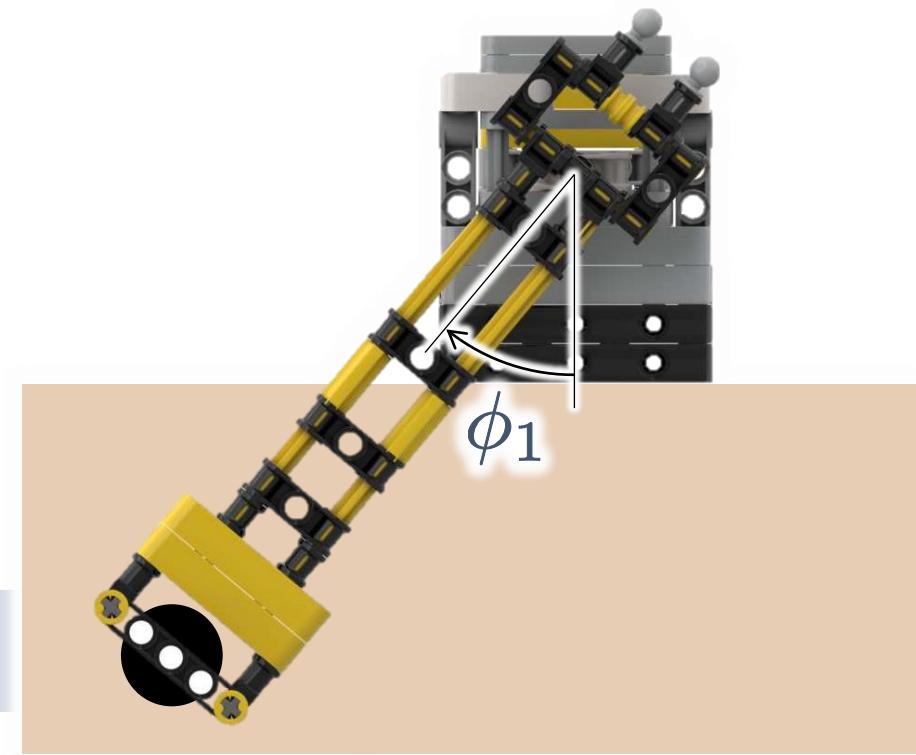
非線形補償 (入力変換)

$$v = u + \gamma_1 (\underline{\sin \phi_1} - \phi_1)$$

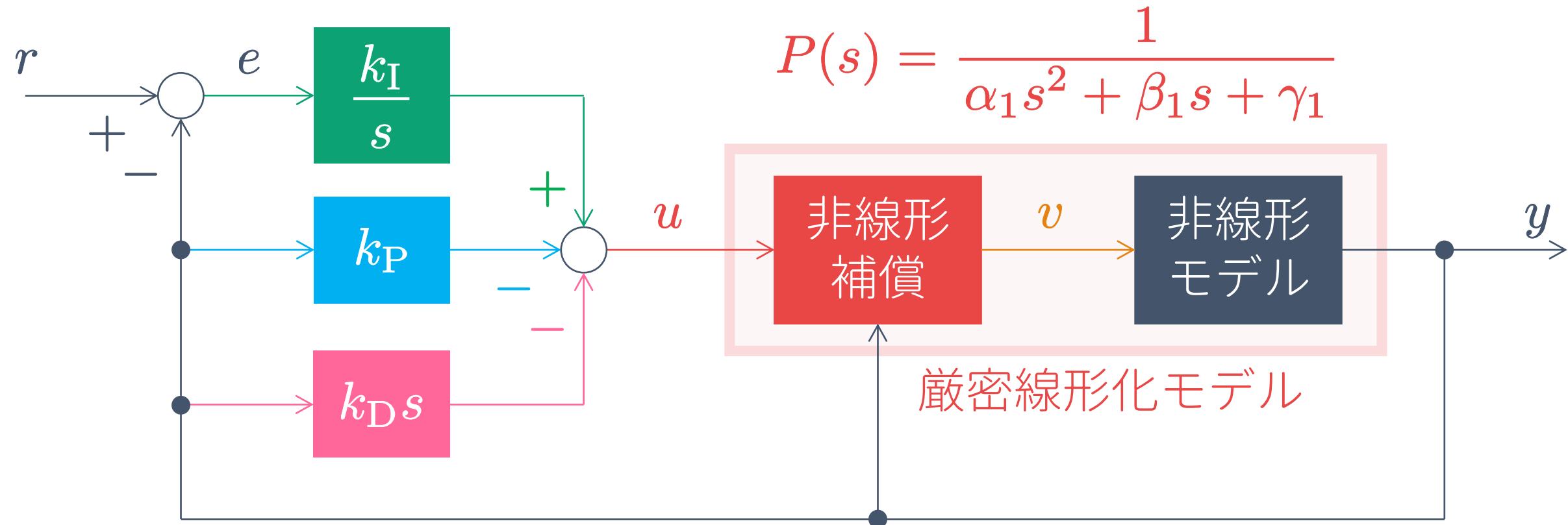
近似によらない厳密線形化モデル

$$\alpha_1 \ddot{\phi}_1 = -\beta_1 \dot{\phi}_1 - \gamma_1 \phi_1 + u$$

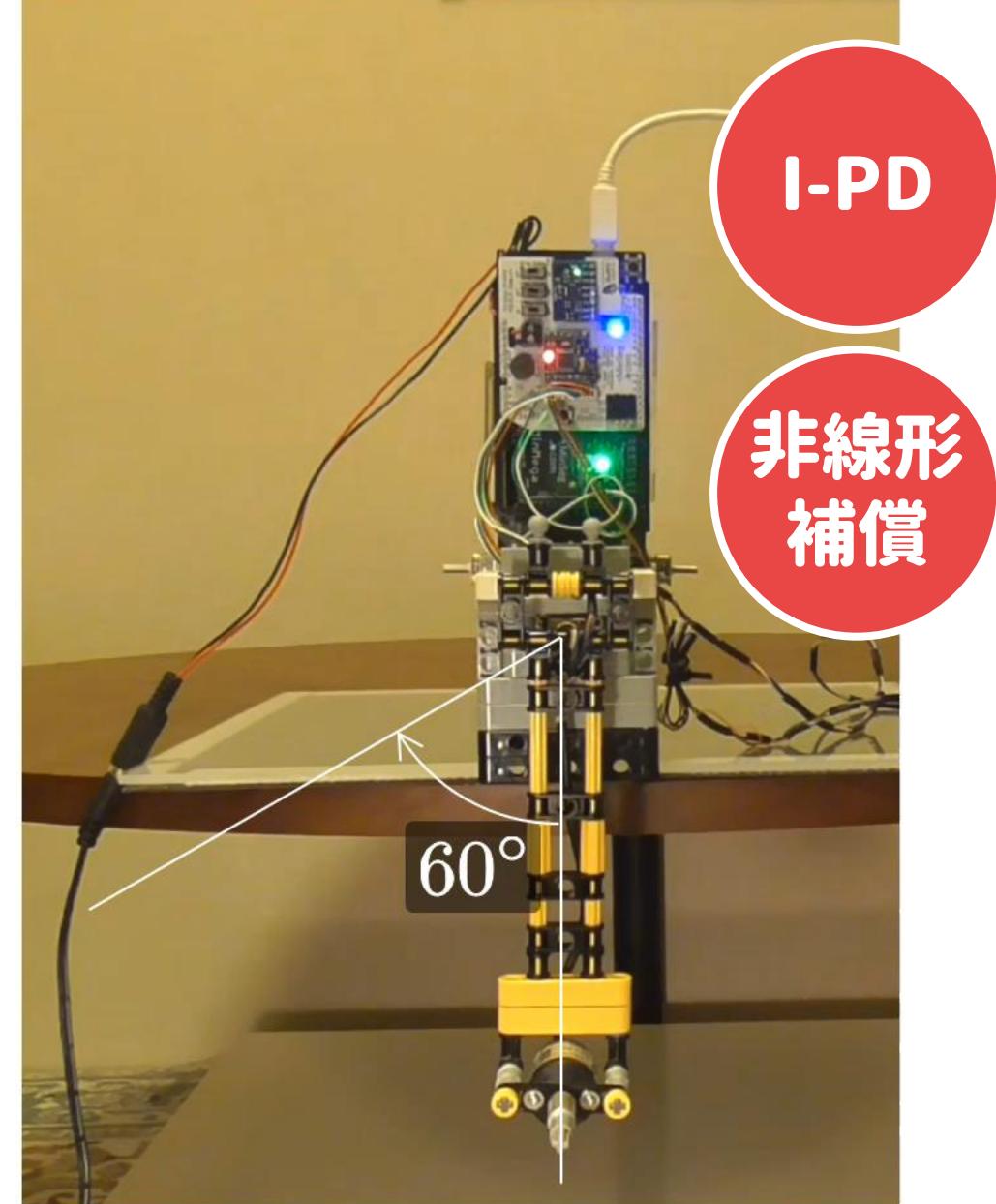
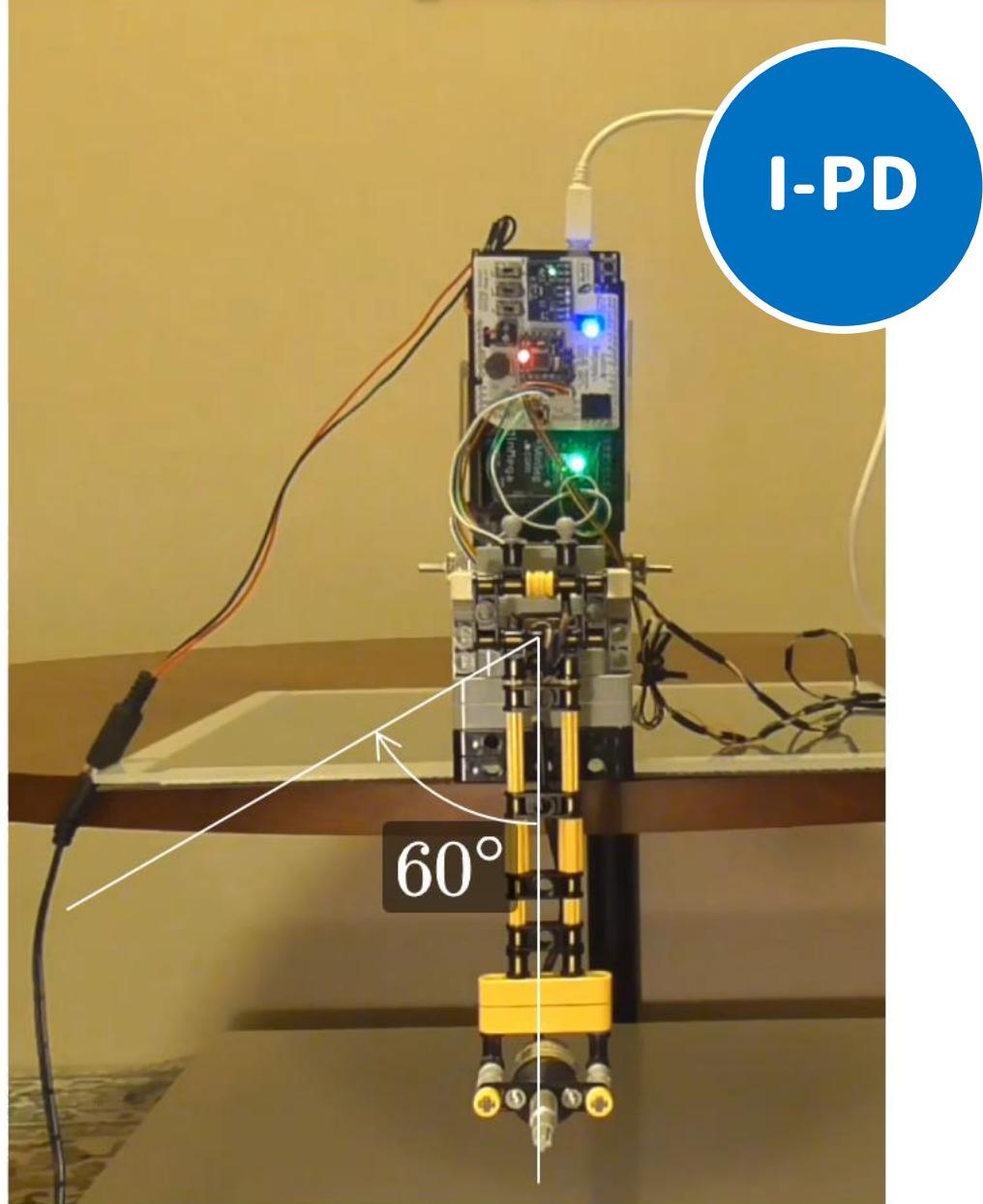
$$\phi_1(s) = P(s)u(s), \quad P(s) = \frac{1}{\alpha_1 s^2 + \beta_1 s + \gamma_1}$$



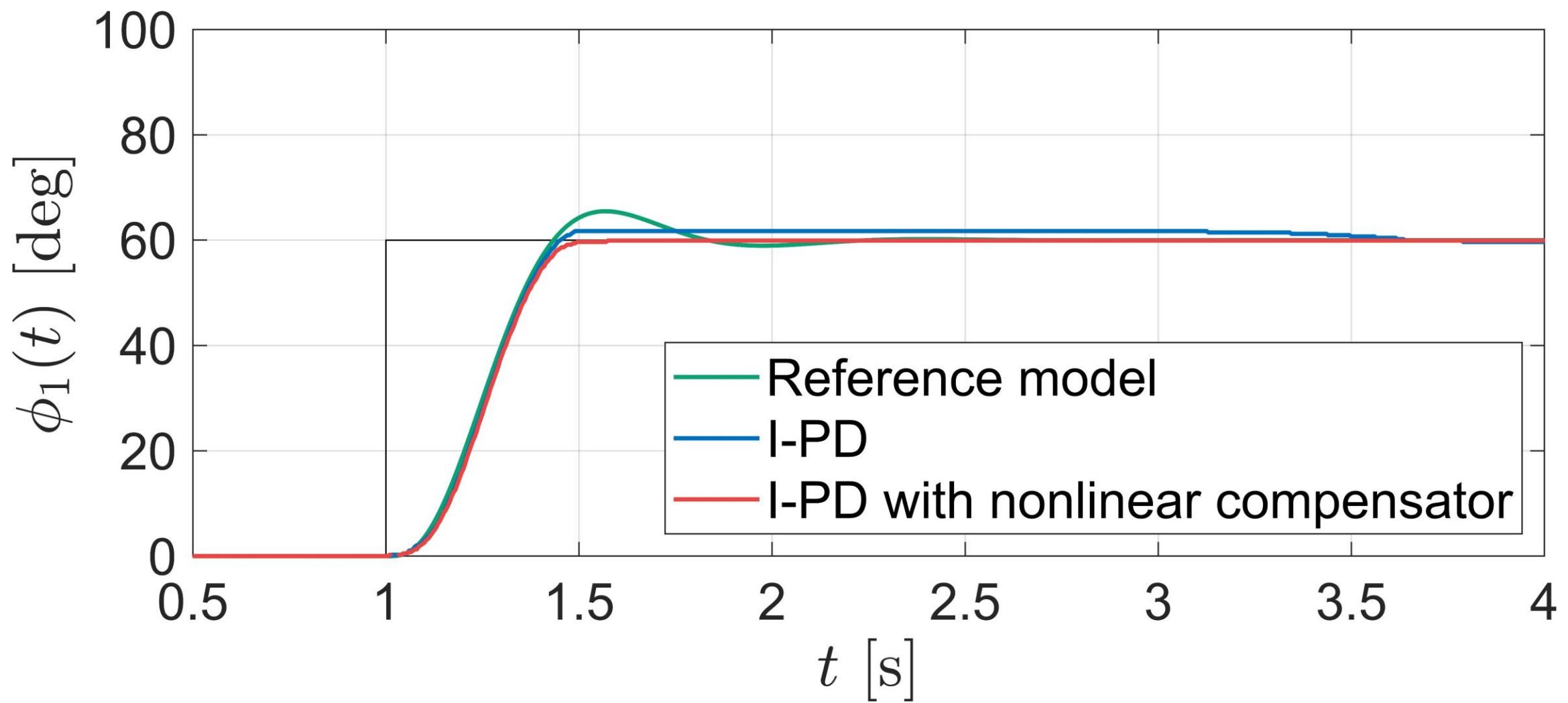
● 非線形補償



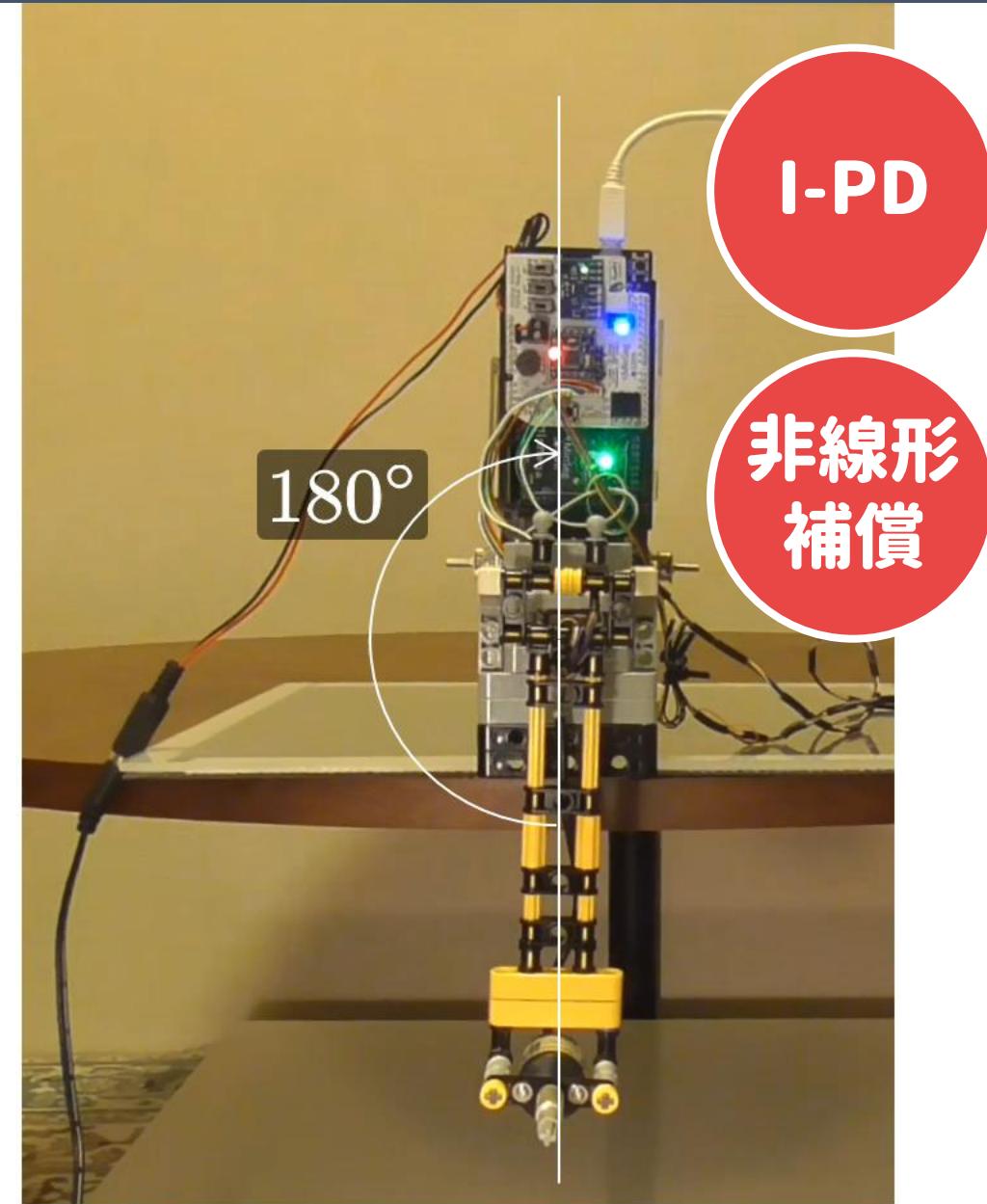
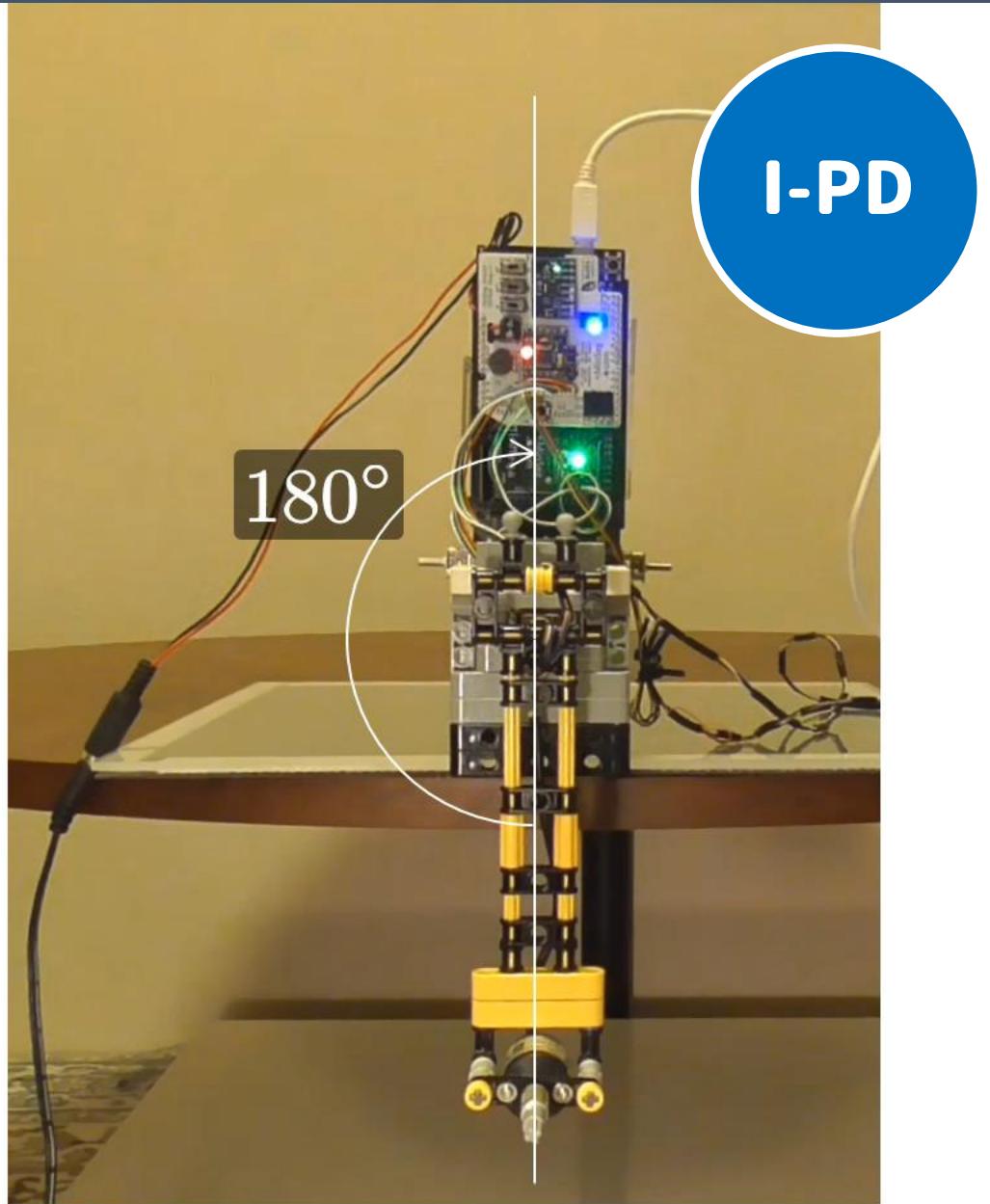
● 非線形補償



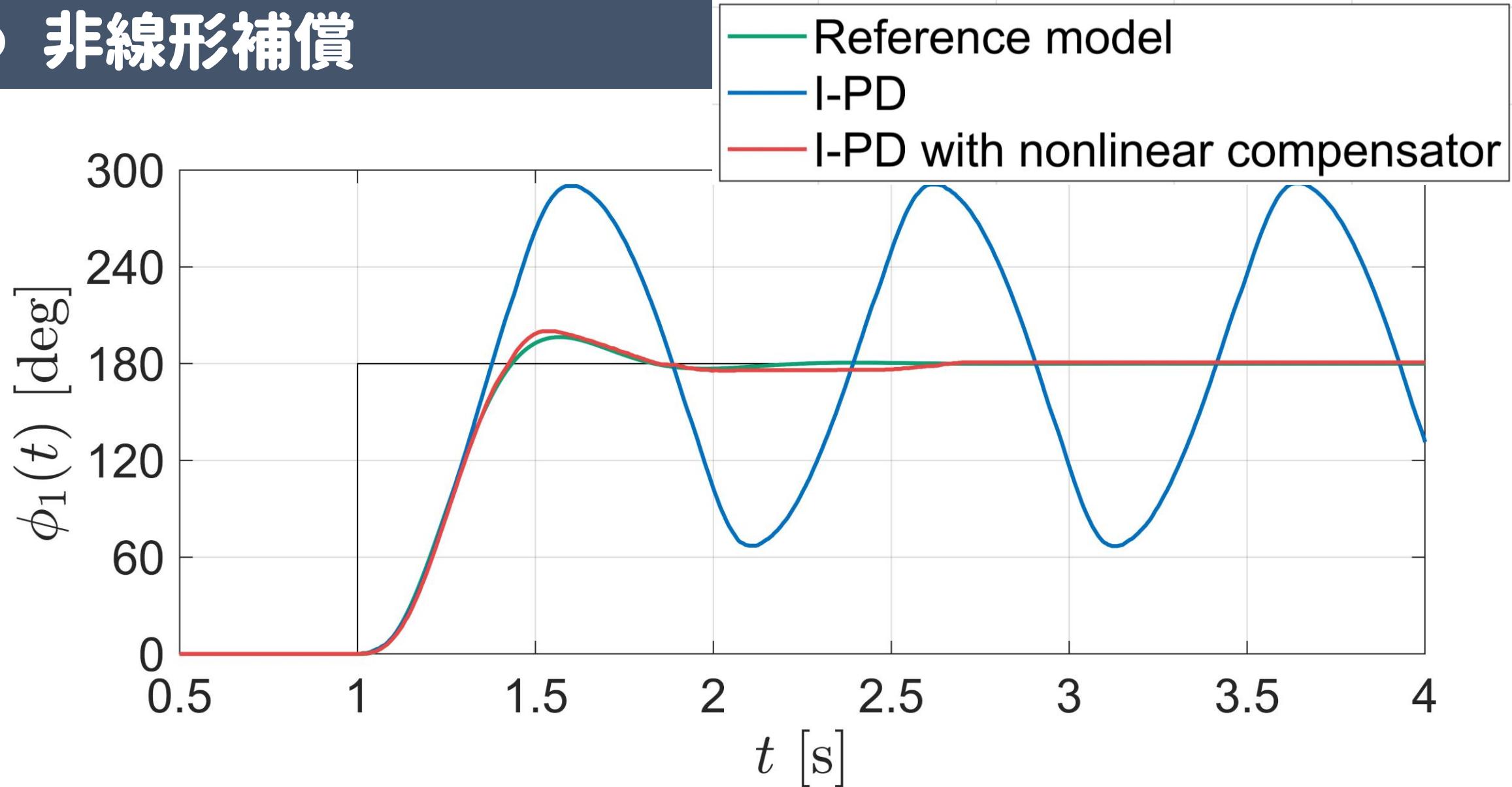
● 非線形補償



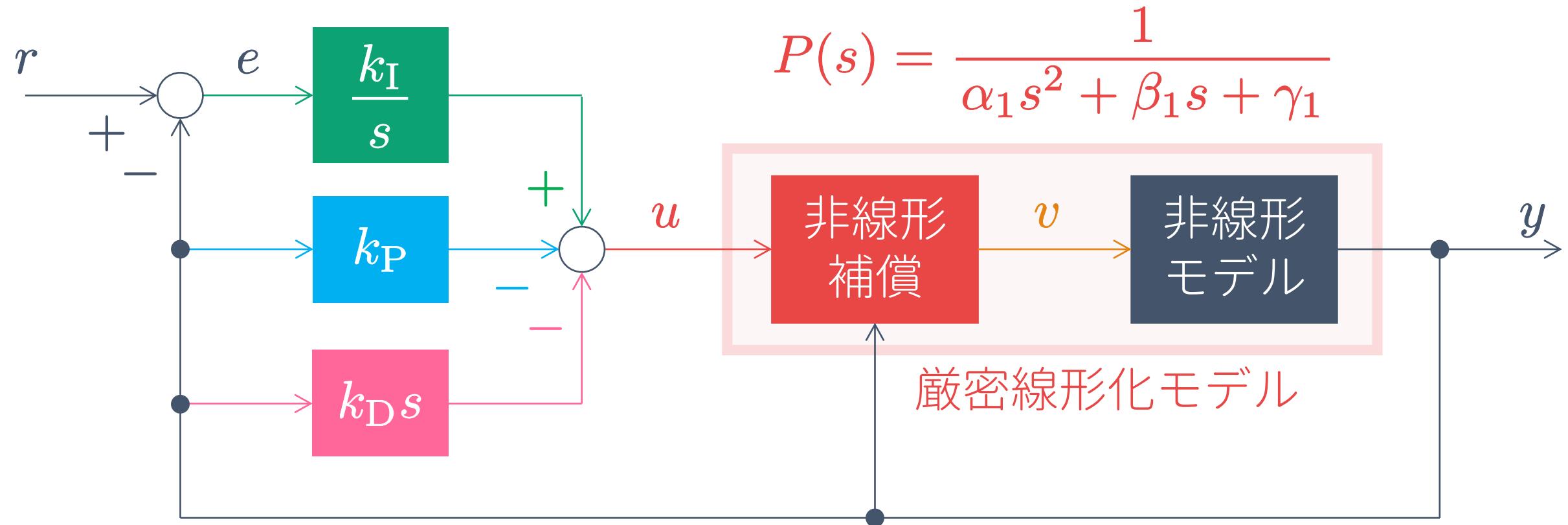
● 非線形補償



● 非線形補償



● 非線形補償



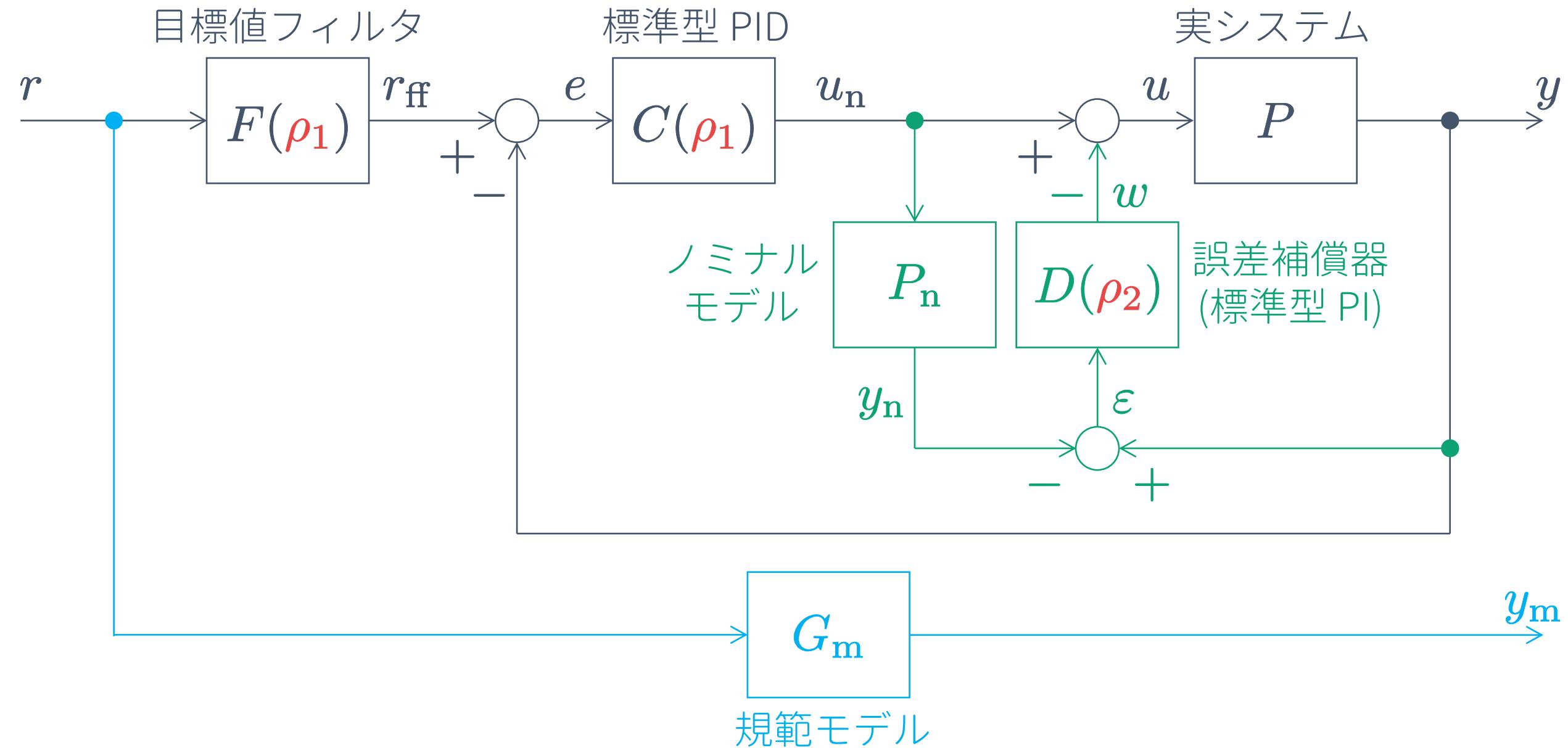
PID 制御の学習 (鉛直面を回転するアームの角度制御)

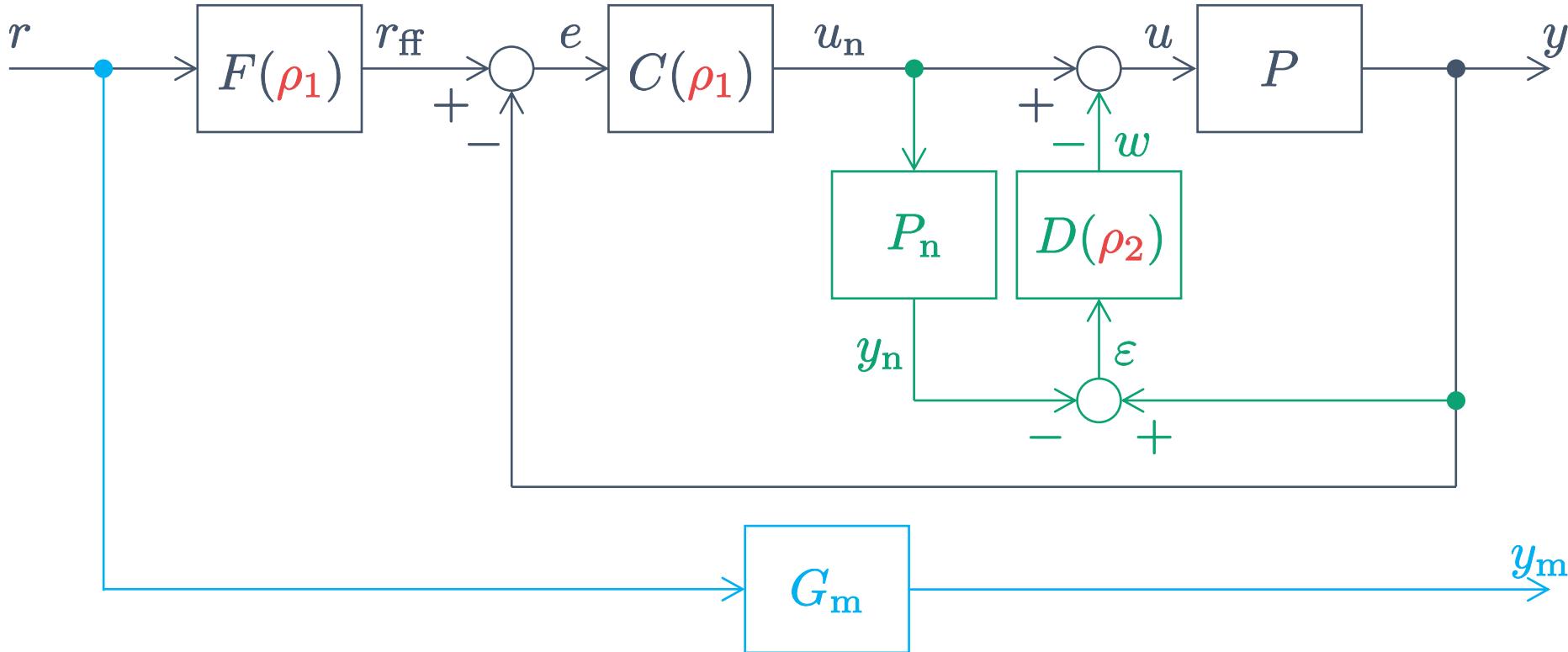
- ▶ 各動作の効果
- ▶ FRIT による設計
- ▶ モデルに基づく設計
 - 非線形モデル
 - 2次遅れ系の特性に基づくパラメータ同定
 - 最小2乗法に基づくパラメータ同定
 - モデルマッチング法による設計
 - 非線形補償
 - モデル誤差抑制補償 (MEC)



モデル化誤差

● FRIT による I-PD + MEC のパラメータ調整

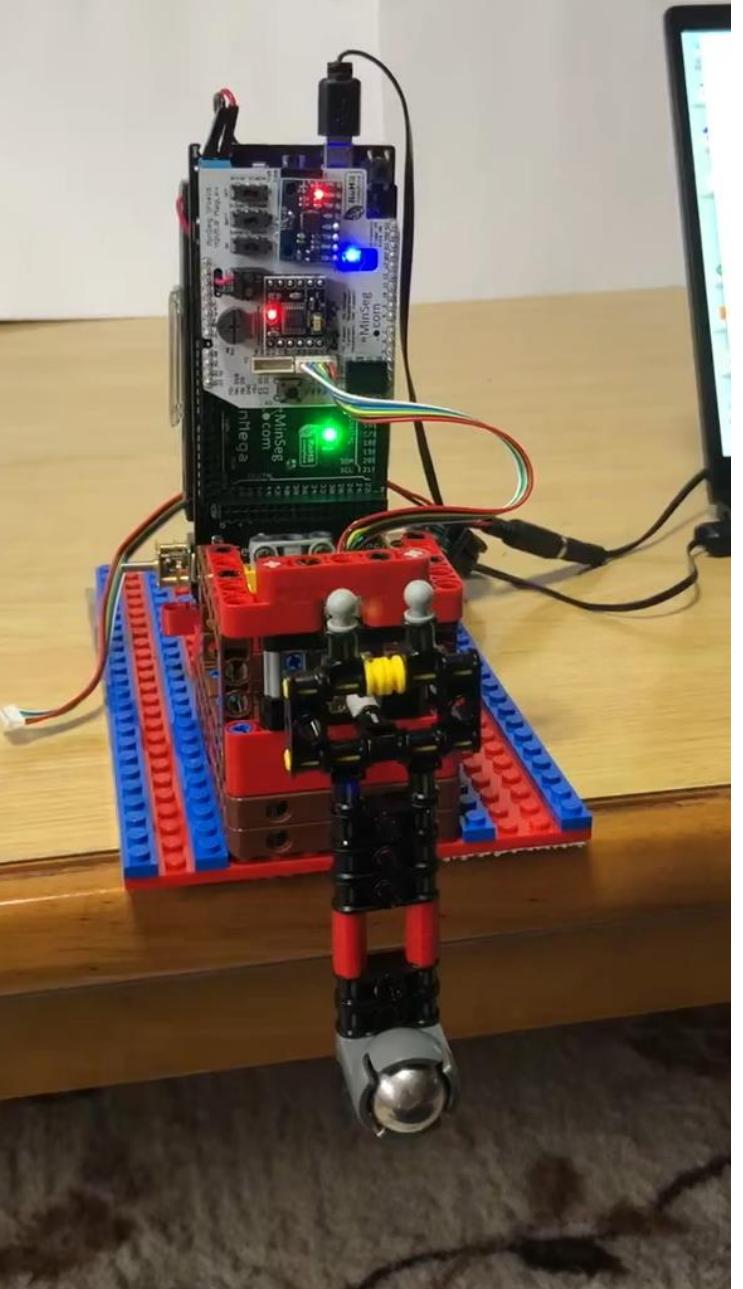




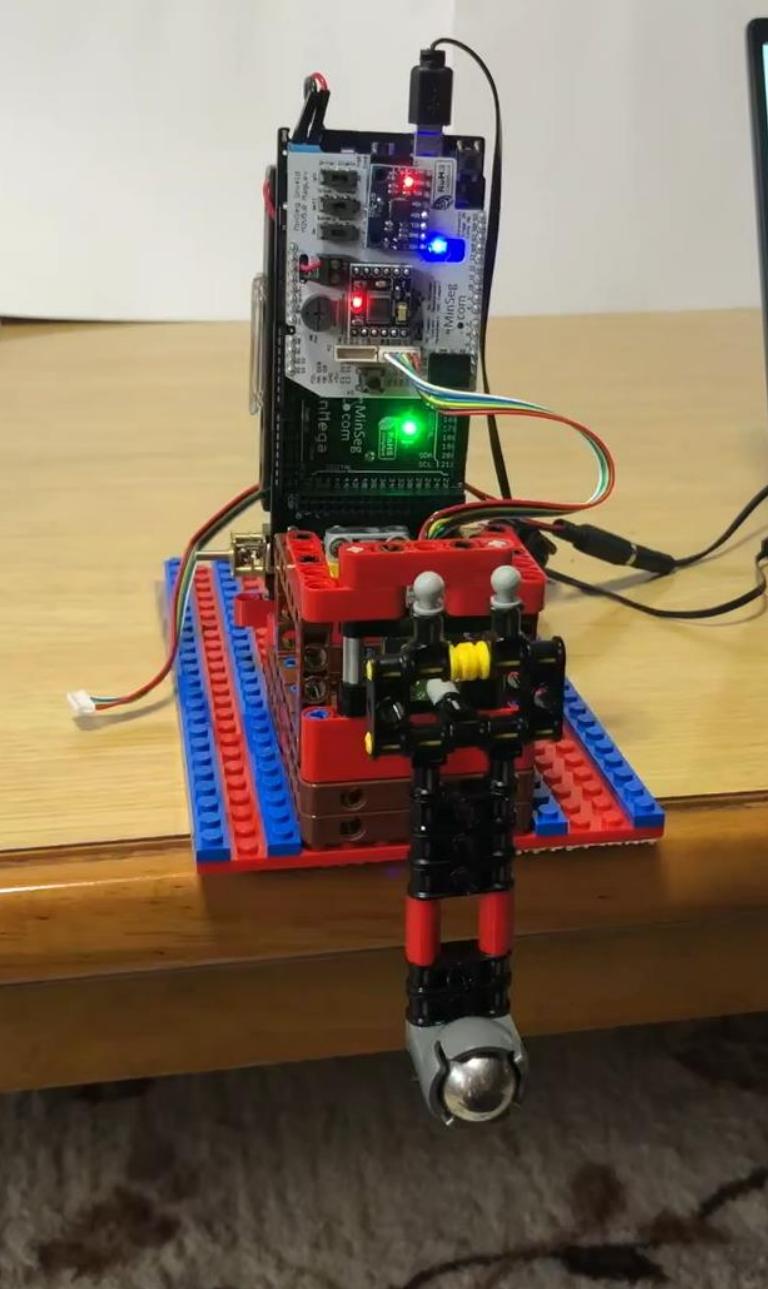
$$C = \frac{k_{D1}s^2 + k_{P1}s + k_{I1}}{s} : \text{標準型 PID} \quad D = \frac{k_{P2}s + k_{I2}}{s} : \text{誤差補償器}$$

$$F = \frac{k_{I1}}{k_{D1}s^2 + k_{P1}s + k_{I1}} : \text{目標値フィルタ}$$

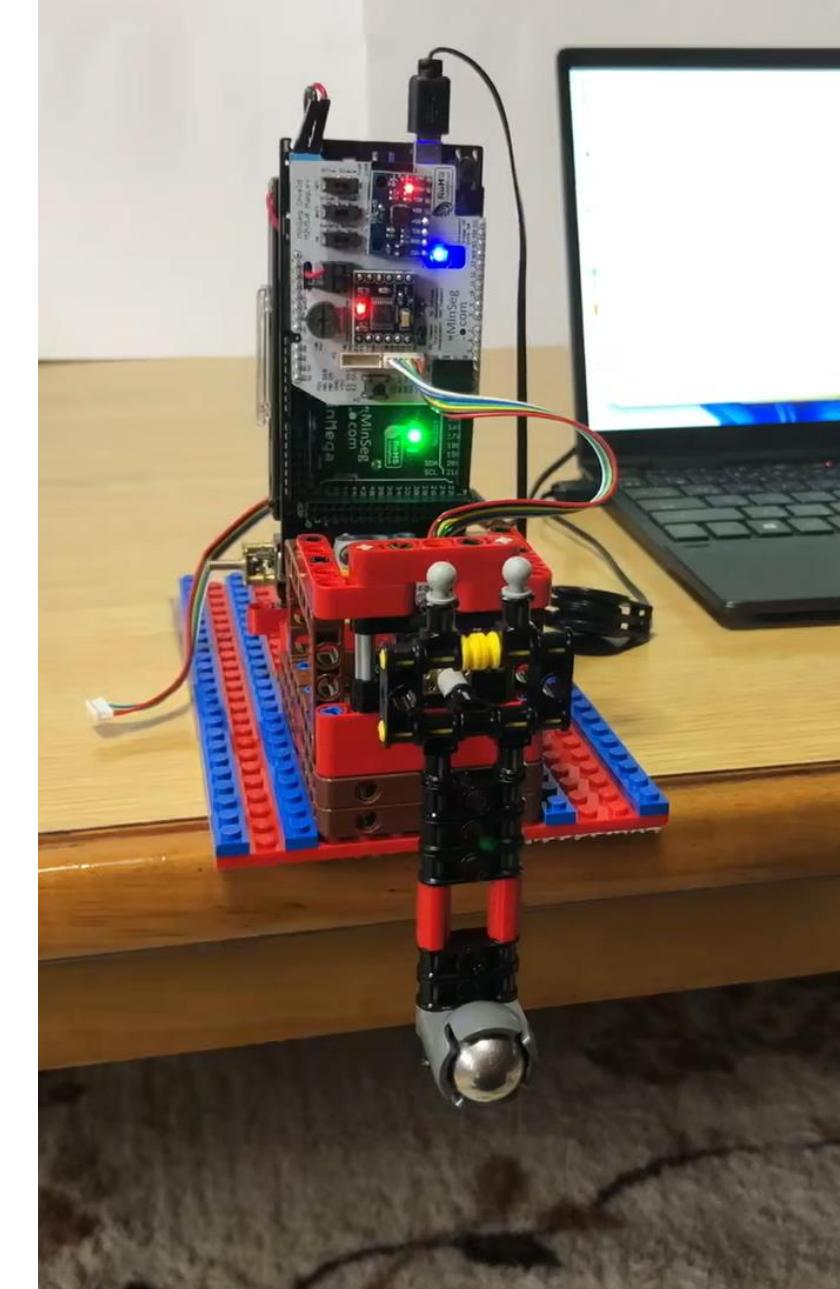
I-PD (モデルマッチング)

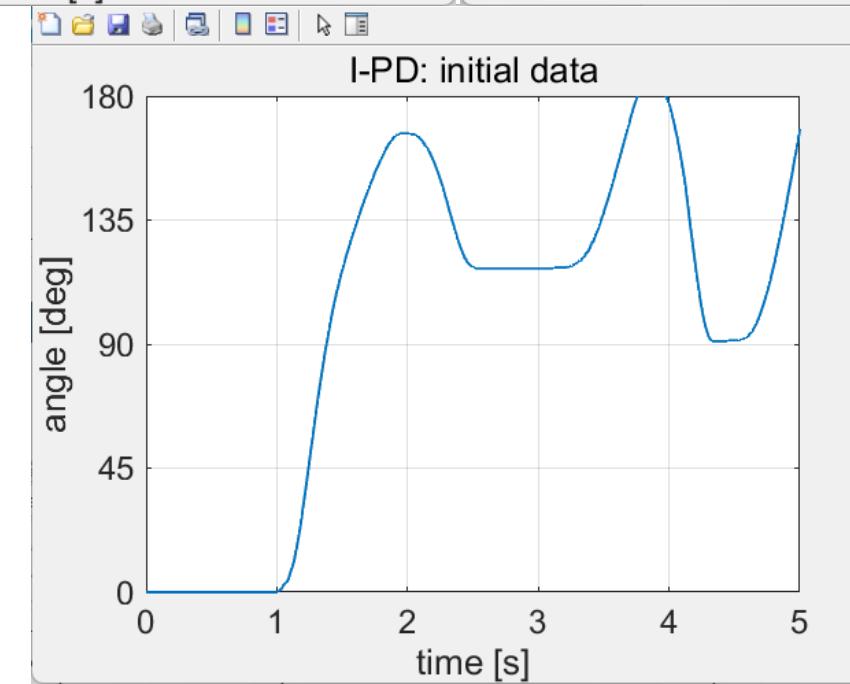
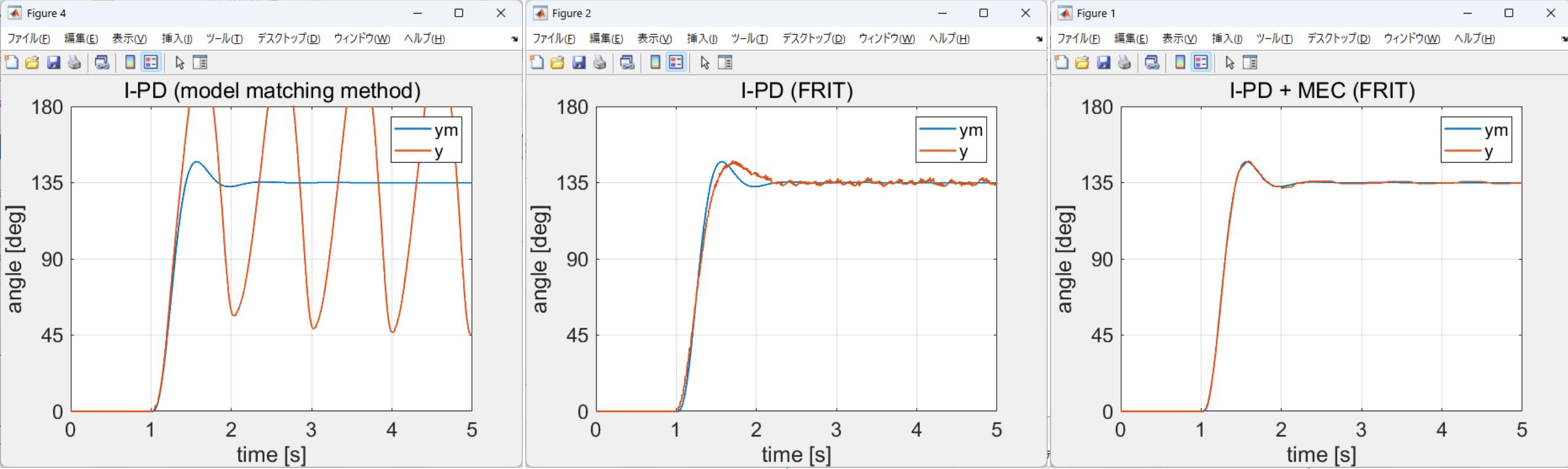


I-PD (FRIT)



I-PD + MEC (FRIT)





このスライドは終了です。