

制御理論のpromナード

～古典からポストモダンまで～

Aleph 2016 / Group Epsilon Meeting #2

2016 年 3 月 13 日

石川 徹也

Web 公開用資料

自己紹介

氏名: 石川徹也(いしかわてつや)

属性: 28 歳 ♂

出身: 芝高等学校

東京工業大学(制御システム工学科)

東京工業大学大学院(情報環境学専攻)

専門: 制御理論, 力学系, 信号処理, 画像処理

職業: 研究員(横河電機)

趣味: スポーツや数理系を中心に色々と

- ランニング
- Python, Haskell, D lang, Julia lang, ...
- ニコリのパズルをコンピュータに解かせる



本日の流れ

発表タイトル

制御理論のpromナード ～古典からポストモダンまで～

- ① イン트로ダクション
- ② 具体例を用いて制御理論をご紹介
- ③ 制御理論入門
- ④ 制御理論史概要

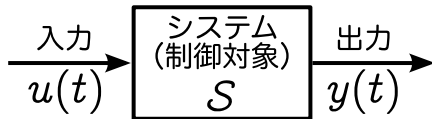
制御とは？

■ 「対象」を思い通りに操ること

- 「対象」を望ましい状態に推移させること

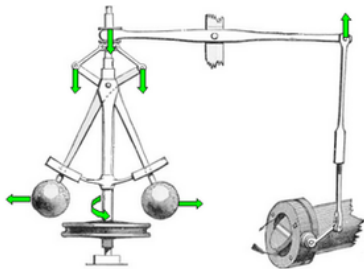
■ 「対象」とは

- 時間的に変化する何か
- 入力 $u(t)$ と出力 $y(t)$ を持つ
- システムとしてプロパーである(ただの汎関数ではない)



抽象化された制御対象

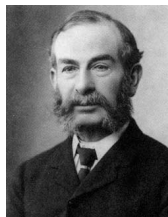
逸話: ガバナーと Maxwell と Routh



世界初の制御器 “Governor”



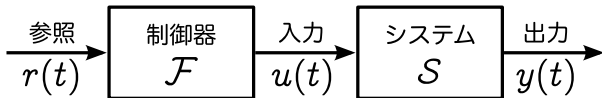
James Clerk Maxwell (1831-1879)



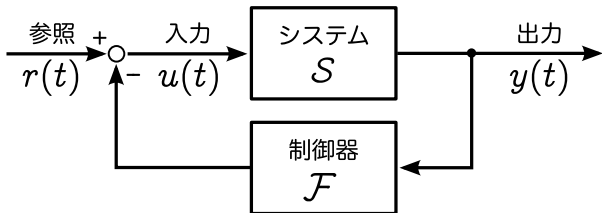
Edward Routh (1831-1907)

制御の常套手段 (1)

■ フィードフォワード制御 (Feedforward Control)



■ フィードバック制御 (Feedback Control)



制御の常套手段 (2)

- 制御手順
1. 制御仕様を策定する
 2. 対象をモデル化する
 3. 制御できるかどうかを確認する
 4. オブザーバを設計する
 5. コントローラを設計する
 6. 制御仕様が達成されていることを確認する

Part 1

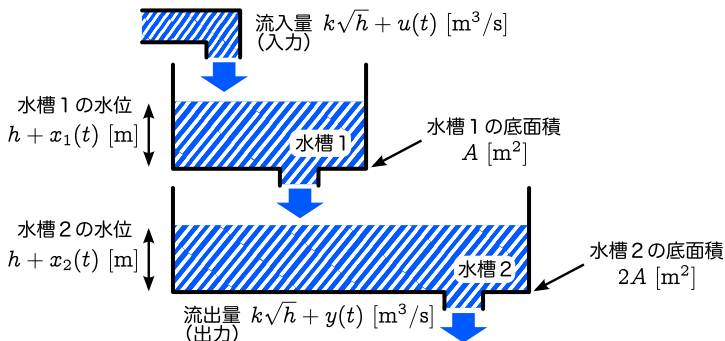
能書きばかり垂れていても仕方がないので
とりあえず制御してみよう!

例：2 水槽系と制御目標

制御対象： 2 水槽系

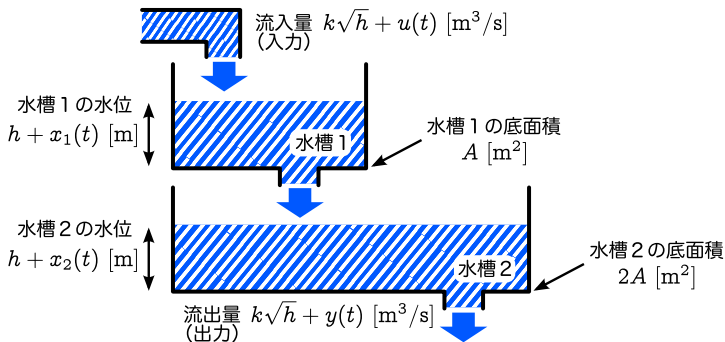
制御目標： 各水槽の水位を一定値 h [m] に保つこと

評価指標： 各水槽の水位が一定値に収束するまでの時間



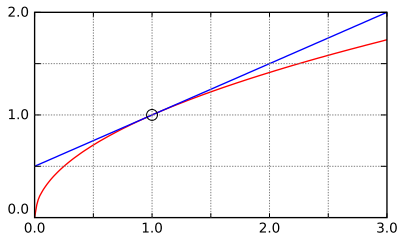
例：2水槽系の数理モデル (1)

$$S : \begin{cases} A\dot{x}_1(t) = k\sqrt{h} + u(t) - k\sqrt{h + x_1(t)}, & x_1(0) = h + 0.03, \\ 2A\dot{x}_2(t) = k\sqrt{h + x_1(t)} - k\sqrt{h + x_2(t)}, & x_2(0) = h - 0.03, \\ y(t) = k\sqrt{h + x_2(t)} - k\sqrt{h}, \end{cases}$$



例：2水槽系の数理モデル (2)

- システムを記述する方程式に非線形項が入っているのは厄介
- 目標値 $x_1(t) = x_2(t) = h$ の近傍でしか制御しないと割り切る
- 非線形項を直線で置き換えたシステムは非常に扱い易い



$y = \sqrt{x}$ を $x = 1$ で線形化した例

曲線 $y = \sqrt{x}$ を $x = h$ の周囲で線形化すると次の直線になる：

$$y = \frac{x + h}{2\sqrt{h}}$$

- 非線形システムの制御が不可能ということではない
- しかし非線形システムでは出来ることはシステムの安定化などに限られている

例：2水槽系の数理モデル (3)

システム

$$S : \begin{cases} A\dot{x}_1(t) = k\sqrt{h} + u(t) - k\sqrt{h + x_1(t)}, & x_1(0) = h + 0.03, \\ 2A\dot{x}_2(t) = k\sqrt{h + x_1(t)} - k\sqrt{h + x_2(t)}, & x_2(0) = h - 0.03, \\ y(t) = k\sqrt{h + x_2(t)} - k\sqrt{h}, \end{cases}$$

を $x_1(t) = x_2(t) = h$ で線形化すると

$$\hat{S} : \begin{cases} A\dot{x}_1(t) = -\frac{kx_1(t)}{2\sqrt{h}} + u(t), & x_1(0) = h + 0.03, \\ 2A\dot{x}_2(t) = \frac{kx_1(t)}{2\sqrt{h}} - \frac{kx_2(t)}{2\sqrt{h}}, & x_2(0) = h - 0.03, \\ y(t) = \frac{kx_2(t)}{2\sqrt{h}}, \end{cases}$$

例：2水槽系の数理モデル (4)

システム $\hat{S}$ を行列にまとめると

$$\hat{S}: \begin{cases} \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \frac{k}{4A\sqrt{h}} \begin{pmatrix} -2 & 0 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \frac{1}{A} \begin{pmatrix} 1 \\ 0 \end{pmatrix} u(t), \\ y(t) = \frac{k}{2\sqrt{h}} \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}, \end{cases}$$

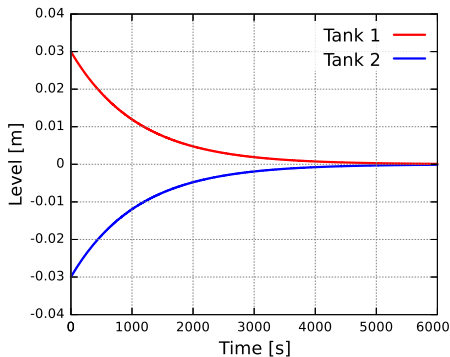
となる. 係数行列 A, B, C を適当に定めれば

$$\hat{S}: \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0, \\ y(t) = Cx(t), \end{cases}$$

と表せる. ベクトル $x(t) \triangleq (x_1(t), x_2(t))^T$ を **状態** と呼ぶ.

例：2水槽系の過渡応答

- 放っておいても目標値に収束する
- だが収束までの時間が長い(約 6000 秒 = 1 時間 40 分)
- 収束時間を半減させることは可能か？

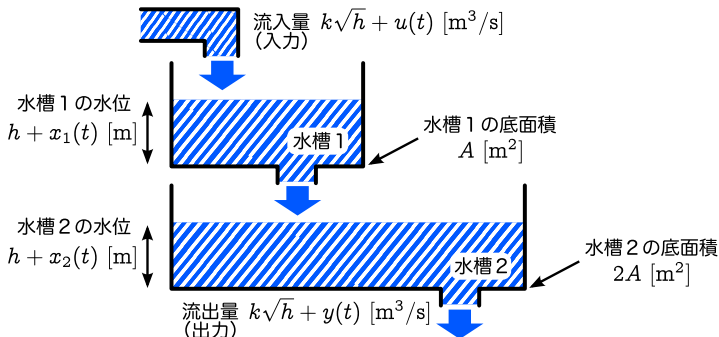


2 水槽系の過渡応答 (開ループ系の応答)

例：2水槽系を制御するための手順

問題設定

- 状態 $x(t)$ は未知
- 既知なのは出力 $y(t)$ と係数行列 A, B, C
- 制御目標は $x(t) \rightarrow 0$



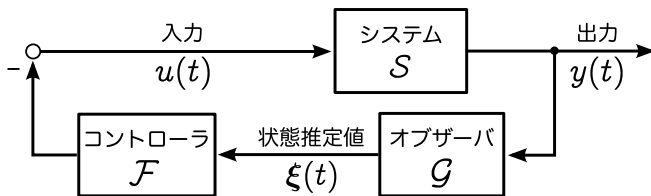
例：2水槽系を制御するための手順

問題設定

- 状態 $x(t)$ は未知
- 既知なのは出力 $y(t)$ と係数行列 A, B, C
- 制御目標は $x(t) \rightarrow 0$

制御手順

1. 出力 $y(t)$ から状態 $x(t)$ を推定 (**オブザーバ** 設計)
2. 状態の推定値を用いて制御 (**コントローラ** 設計)



コントローラ・オブザーバ併合系

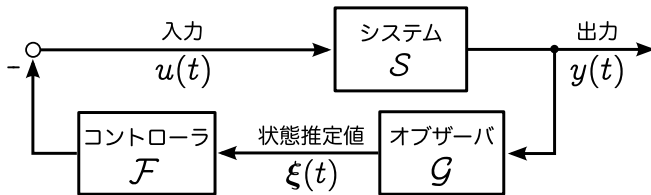
例：2水槽系のオブザーバ設計 (1)

推定戦略 システム

$$\hat{S}: \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0, \\ y(t) = Cx(t), \end{cases}$$

に次のオブザーバを付加しよう:

$$\dot{\xi}(t) = A\xi(t) + Bu(t) - G(C\xi(t) - y(t)), \quad \xi(0) = \xi_0,$$

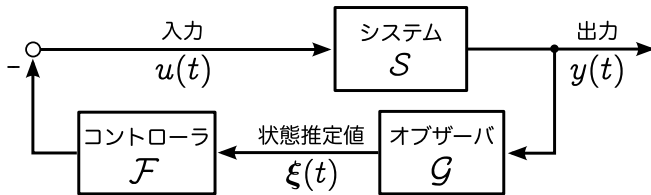


コントローラ・オブザーバ併合系

例：2水槽系のオブザーバ設計 (2)

- 特徴
- 状態 $x(t)$ に収束する変数 $\xi(t)$ を導入
 - 出力 $y(t)$ と比較しながら $\xi(t)$ を調節
 - オブザーバ行列 G が $\xi(t)$ の収束性を決定する
 - 変数 $\xi(t)$ を状態 $x(t)$ に収束させられないことも

$$\dot{\xi}(t) = A\xi(t) + Bu(t) - G(C\xi(t) - y(t)), \quad \xi(0) = \xi_0,$$



コントローラ・オブザーバ併合系

例：2水槽系のオブザーバ設計 (3)

- オブザーバ併合系は

$$\hat{\mathcal{S}}_o : \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0, \\ \dot{\xi}(t) = A\xi(t) + Bu(t) - G(C\xi(t) - y(t)), & \xi(0) = \xi_0, \\ y(t) = Cx(t), \end{cases}$$

となる

- ここで状態の真値と推定値の偏差を $e(t) \triangleq \xi(t) - x(t)$ とおくと、 $e(t)$ の従うダイナミクスは次のようになる：

$$\dot{e}(t) = (A - GC)e(t).$$

- すなわち、推定値の収束性は $A - GC$ の固有値に支配される

例：2水槽系のオブザーバ設計 (4)

- 行列 $A - GC$ の固有値が安定(実部が負)になるようオブザーバ行列 G を設計しなければならない
- ここでオブザーバ行列 G を

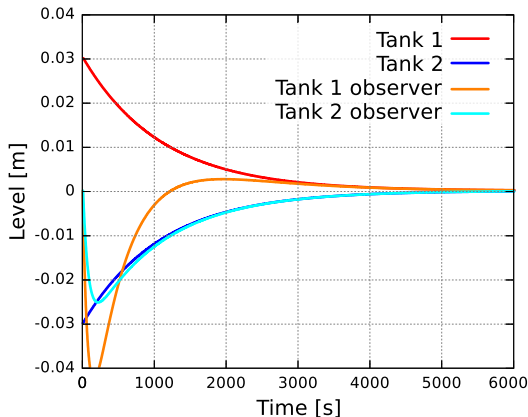
$$G = \frac{1}{2A} \begin{pmatrix} 40 \\ 20 \end{pmatrix},$$

とおけば, $A - GC$ の固有値は $-\frac{4.821}{4A\sqrt{h}}, -\frac{16.18}{4A\sqrt{h}}$ となるので, 変数 $\xi(t)$ は $x(t)$ に収束する

- ただし安定性はあくまで必要条件であるから, 最後はシステムの応答を見ながらオブザーバ行列 G を設計するのが良い

例：2水槽系のオブザーバの過渡応答

- 状態の推定値 $\xi(t)$ は状態 $x(t)$ に収束している



2 水槽系とオブザーバの過渡応答

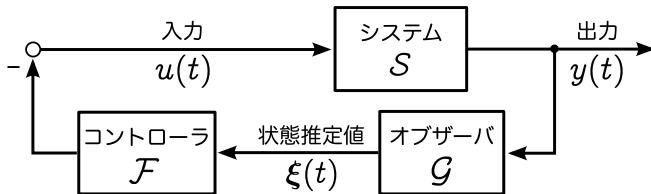
例：2水槽系のコントローラ設計 (1)

制御戦略 システム

$$\hat{S}_o : \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0, \\ \dot{\xi}(t) = A\xi(t) + Bu(t) - G(C\xi(t) - y(t)), & \xi(0) = \xi_0, \\ y(t) = Cx(t), \end{cases}$$

に次の制御器を構成する(状態フィードバック)

$$u(t) = -F\xi(t).$$



コントローラ・オブザーバ併合系

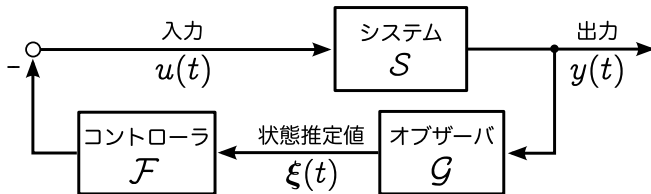
例：2水槽系のコントローラ設計 (2)

- 状態フィードバックを施したシステムは次のようになる：

$$\hat{\mathcal{S}}_{oc} : \begin{cases} \dot{x}(t) = Ax(t) - BF\xi(t), & x(0) = x_0, \\ \dot{\xi}(t) = A\xi(t) - BF\xi(t) - G(C\xi(t) - y(t)), & \xi(0) = \xi_0, \\ y(t) = Cx(t) \end{cases}$$

- オブザーバが十分に収束した後のダイナミクスは

$$\dot{x}(t) = (A - BF)x(t)$$



コントローラ・オブザーバ併合系

例：2水槽系のコントローラ設計 (3)

- 行列 $A - BF$ の固有値が安定(実部が負)になるようフィードバック行列 F を設計しなければならない
- ここでフィードバック行列 F を

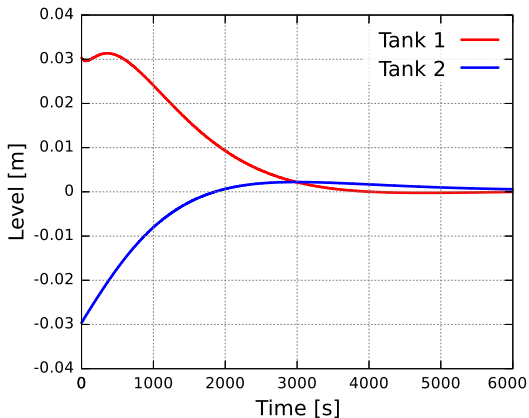
$$F = \frac{1}{2\sqrt{h}} \begin{pmatrix} 0.4 & 1.0 \end{pmatrix},$$

とおけば, $A - BF$ の固有値は $-\frac{1.900}{4A\sqrt{h}} \pm i\frac{1.091}{4A\sqrt{h}}$ となるので, 変数 $x(t)$ は 0 に収束する

- ただし安定性はあくまで必要条件であるから, 最後はシステムの応答を見ながらオブザーバ行列 G を設計するのが良い

例：2水槽系の過渡応答(制御後)

- 約 3000 秒で収束している



2 水槽系の過渡応答(制御後)

Part 2

少しだけ一般的なお話

制御理論入門

可観測性:そもそも状態推定できるのか?

- オブザーバの収束性は自由に決定できるのか?
- 行列 $A - GC$ の極は自由に設定できるのか?

$$\begin{aligned}\hat{\mathcal{S}} : \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0, \\ y(t) = Cx(t), \end{cases} \\ \dot{\xi}(t) = A\xi(t) + Bu(t) - G(C\xi(t) - y(t)), \\ \dot{e}(t) = (A - GC)e(t),\end{aligned}$$

定義:可観測性

システム $\hat{\mathcal{S}}$ において, 有限時刻 t_1 までの出力 $y(t)$, $0 \leq t \leq t_1$ を観測するとにより, 初期状態 x_0 を一意に決定できるとき, システム $\hat{\mathcal{S}}$ は初期値 x_0 について可観測であると言う. さらに, 任意の初期値 x_0 について可観測であるとき, $\hat{\mathcal{S}}$ は完全可観測であると言う.

可観測性:オブザーバの極配置可能性

定理:可観測性とオブザーバの極配置

システムが完全可観測ならば, 行列 G を適切に設定することで $A - GC$ の極を任意に設定することができる.

証明:行列 A が相異なる対角成分を持つ 2 次対角行列の場合のみを扱う

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} u(t),$$
$$y(t) = \begin{pmatrix} c_1 & c_2 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}$$

■ 可観測 $\iff c_1 \neq 0, c_2 \neq 0$

■ $\det(\lambda I - A + GC) = \lambda^2 - \lambda(a_1 a_2 - c_1 g_2 - c_2 g_1) + a_1 a_2 - a_1(c_1 g_2 + c_2 g_1) + (a_1 - a_2)g_1 c_2$

可制御性とコントローラーの極配置問題

■ 実はオブザーバの場合とほぼ同じ

定義: 可制御性

システム $\hat{S}$ において, 有限時刻 t_1 までに状態を $x(t_1) = 0$ に移す入力 $u(t)$, $0 \leq t \leq t_1$ が存在するとき, システム $\hat{S}$ は初期値 x_0 について可制御であると言う. さらに, 任意の初期値 x_0 について可制御であるとき, $\hat{S}$ は完全可制御であると言う.

定理: 可制御性とコントローラーの極配置

システムが完全可制御ならば, 行列 F を適切に設定することで $A - BF$ の極を任意に設定することができる.

最適制御: コスト関数を最小にする制御 (1)

目的意識 極(固有値)のシステムティックな定め方は?

問題設定 状態がすべて直接観測可能な n 次元システム

$$\mathcal{S}: \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0, \\ y(t) = Cx(t), \end{cases}$$

を考える. ただし $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$ とする.

アプローチ 評価関数 $J(u)$ を最小にする入力を求める:

$$\min_u J(u) = \int_0^\infty \left(x^T(t) Q x(t) + u^T(t) R u(t) \right) dt.$$

ただし $Q \in \mathbb{R}^{n \times n}$, $R \in \mathbb{R}^{m \times m}$ は実正定対称行列に限定

最適制御: コスト関数を最小にする制御 (2)

これを関数空間上の最適化問題として定式化すると

$$\begin{aligned} \min_{\substack{\mathbf{u} \in L^2(\mathbb{R}^m; \mathbb{R}) \\ \mathbf{x} \in L^2(\mathbb{R}^n; \mathbb{R})}} \quad & J(\mathbf{u}) = \langle \mathbf{x}, \mathbf{Q}\mathbf{x} \rangle + \langle \mathbf{u}, \mathbf{R}\mathbf{u} \rangle, \\ \text{s.t.} \quad & \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \quad \mathbf{x}(0) = \mathbf{x}_0, \end{aligned}$$

となる. ただし $L^2(\mathbb{R}^n; \mathbb{R})$ は

$$\langle \mathbf{x}, \mathbf{y} \rangle \triangleq \int_0^\infty \mathbf{x}^T(t) \mathbf{y}(t) dt,$$

を内積とした Hilbert 空間である. 以下, $P : L(\mathbb{R}^m; \mathbb{R}) \rightarrow L(\mathbb{R}^n; \mathbb{R})$ に対して P^* を P の共役作用素とする.

最適制御: コスト関数を最小にする制御 (3)

Linear Quadratic Regulator

最適制御問題

$$\begin{aligned} \min_u \quad & J(u) = \langle x, Qx \rangle + \langle u, Ru \rangle, \\ \text{s.t.} \quad & \dot{x}(t) = Ax(t) + Bu(t), \end{aligned}$$

の解は $u(t) = -R^{-1}B^T P x(t)$ で与えられる。ただし行列 $P \in \mathbb{R}^{n \times n}$ は次の Riccati 方程式をみたす:

$$-PBR^{-1}B^T P + PA + A^T P + Q = O.$$

(注 1) 拘束条件によって x は u によって一意に定まるため、ここでは変数 x を従属変数として取り扱っています。

(注 2) 以下の証明は登壇者のオリジナルです。間違いあればご指摘ください。

最適制御: コスト関数を最小にする制御 (4)

証明の方針

1. 制約条件より

$$x(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau,$$

となる. これを $x(t) \triangleq v(t) + (\Lambda u)(t)$ とおく.

2. 方程式

$$(\Lambda^*Q\Lambda + R)u + \Lambda^*Qv = 0, \tag{\#}$$

は唯一の解 $u = \tilde{u}$ を持ち, $\tilde{u}$ は $J(u)$ を最小化する

3. 制御則 $u = -R^{-1}B^TPx(t)$ は方程式 (＃) をみたす

最適制御: コスト関数を最小にする制御 (5)

証明

1. 自明.
2. まず解の唯一性を示す. そのためには作用素 $\Lambda^*Q\Lambda + R$ が可逆であれば十分. 行列 Q, R は実正定対象行列であるから

$$\begin{aligned}(\Lambda^*Q\Lambda + R)p = 0 &\implies \langle p, (\Lambda^*Q\Lambda + R)p \rangle = 0 \\&\implies \|\sqrt{Q}\Lambda p\|^2 + \|\sqrt{R}p\|^2 = 0 \\&\implies p = 0,\end{aligned}$$

が成り立つ. よって $\Lambda^*Q\Lambda + R$ は可逆であり, 方程式 (♯) は次の唯一解 $\tilde{u}$ を持つ:

$$\tilde{u} = -(\Lambda^*Q\Lambda + R)^{-1}\Lambda^*Qv.$$

最適制御: コスト関数を最小にする制御 (6)

証明の続き

2. 次に唯一解 $\tilde{u}$ が $J(u)$ を最小化することを示す. 1 で定めたように $x = v + \Lambda u$ であるから

$$\begin{aligned} J(\tilde{u}) &= \langle x, Qx \rangle + \langle u, Ru \rangle \\ &= \langle v + \Lambda u, Q(v + \Lambda u) \rangle + \langle u, Ru \rangle, \end{aligned}$$

が成り立つ. ここで $u = \tilde{u} + h$ とおけば

$$J(u) = J(\tilde{u}) + \left\| \sqrt{Q}\Lambda h \right\|^2 + \left\| \sqrt{R}h \right\|^2,$$

となる. よって $h = 0$ のとき $J(u)$ は最小値をとる.

最適制御: コスト関数を最小にする制御 (7)

証明の続き

3. 単純な計算により, 共役作用素 Λ^* は

$$(\Lambda^* q)(t) = B^T e^{-A^T t} \int_t^\infty e^{-A\tau} q(\tau) d\tau,$$

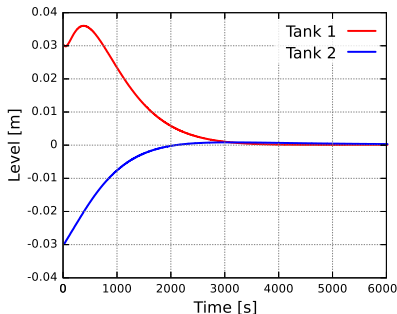
で与えられる. また P が

$$-PBR^{-1}B^TP + PA + A^TP + Q = O,$$

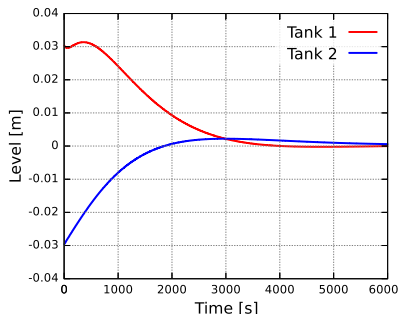
をみたすことに注意すれば, 制御則 $u(t) = -R^{-1}B^TPx(t)$ が
方程式 (‡) をみたすことは直ちに確認できる.

例：2水槽系の過渡応答(最適制御)

- $Q = I_2, R = 1$ という単純なパラメータ設定でもそれなりに満足いく性能が出せる(試行錯誤が少なく済む)



2 水槽系の過渡応答(最適制御)



2 水槽系の過渡応答(極配置)

Part 3

制御理論の歴史概観

古典制御と現代制御, そしてポストモダン

■ 古典制御

- 1860 年代よりベル研を中心に発展
- 周波数領域ベース
- ループ整形や補償を中心とした制御系設計

■ 現代制御(**これまで説明した理論**)

- 1960 年代に R. Kalman がほぼ一人で確立
- 時間領域(状態空間)ベース
- 最適制御によるシステマティックな制御系設計

■ ポストモダン制御

- 1980 年代に J. Doyle らが確立
- 周波数領域と時間領域を横断的に利用
- 古典制御と現代制御の融合

古典制御の概要:線形システムの特徴付け

■ 線形システム

$$\mathcal{S}: \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0, \\ y(t) = Cx(t), \end{cases}$$

を考える

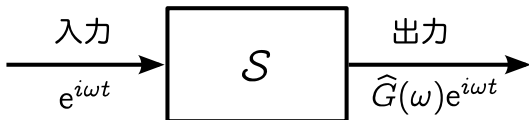
古典制御の概要:線形システムの特徴付け

■ 線形システム

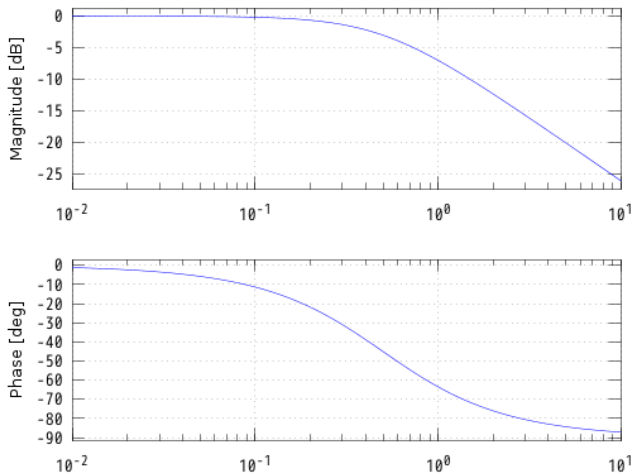
$$\mathcal{S}: \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0, \\ y(t) = Cx(t), \end{cases}$$

を考える

- 入力と出力の関係は線形
- 角周波数 ω の信号 $e^{i\omega t}$ を入力すると同じ周波数の信号を出力する
- ただし振幅と位相は変化する
- その変化をひとつの複素数 $\hat{G}(\omega)$ で表現する
- この関数 $\hat{G}(\omega)$ を周波数伝達関数と呼ぶ



古典制御の概要: 周波数伝達関数と Bode 線図



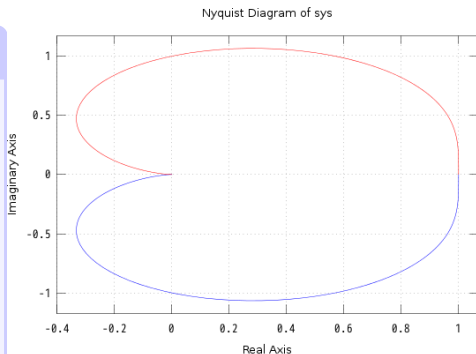
ローパスフィルタの Bode 線図(横軸は角周波数 [rad])

古典制御の概要: Nyquist の安定判別法

Nyquist 線図 周波数伝達関数 $\hat{G}(\omega)$ を複素平面上にプロットしたもの。
Nyquist 線図を見るだけで安定性が判別できる。

Nyquist の安定判別法

システムの閉ループ系が安定であるための必要十分条件は, 開ループ系の周波数伝達の Nyquist 線図が $-1 + i0$ を不安定極の数だけ反時計方向に回ることである。



Nyquist 線図の例

理論と現実のギャップ

- 1970 年までに現代制御理論はほとんど確立された
 - 現代制御はほぼ R. Kalman が一人で作り上げた
 - 1971 年の IEEE TAC には現代制御に関する膨大な量の特集号
- 一方、産業界では古典制御が主流
 - ちょうど計算機の黎明期だったため、産業界の興味はマイクロコンピュータを用いた制御に集中
 - 大学の現代制御と現場の古典制御、という 2 極化構造
- 現代制御の応用研究が 70~80 年代になされるも多くが不発に
 - 論文に "Oops..." などと書く著者まで現れるほど衝撃的 (J. Doyle & G. Stain, "Blind Spot of LQG Control," IEEE TAC, 1979.)

新しい制御理論の必要性

■ 長年の試みで分かったこと

- モデルは現実の粗い近似にすぎない
- 現代制御はモデル誤差の影響を受けやすい
- とは言え、モデル誤差の問題は古典制御でも同じこと

■ 新しい制御理論が求められる

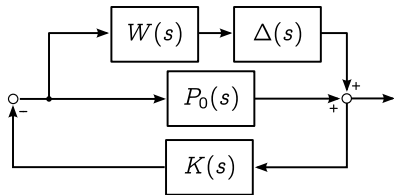
- 「モデルに基づく合理的な制御」から「不確かなモデルに基づく合理的な制御」へ

■ ロバスト制御の誕生

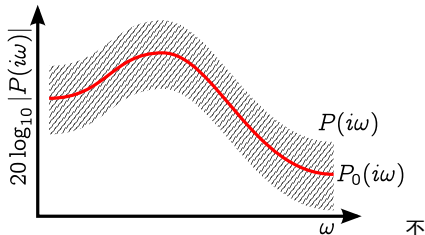
- モデルの不確かさを 周波数空間で記述
- 不確かなモデルすべてを安定化する制御器を設計

ロバスト制御の概要 (1)

- 加法的な不確かさを持つ系 $P(s) = P_0(s) + W(s)\Delta(s)$ を考える
- ただし $\|\Delta(s)\|_{H^\infty} \triangleq \sup_{\omega \in \mathbb{R}} \Delta(i\omega) \leq \gamma$
- 任意の $\Delta(s)$ に対して系を安定化できるコントローラを設計できるか？



不確かな系のブロック線図

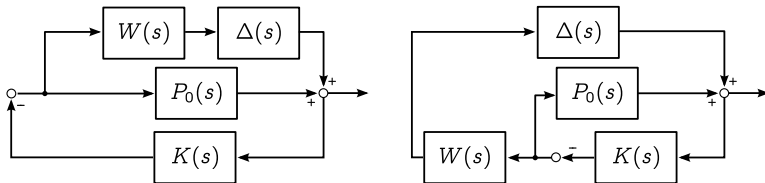
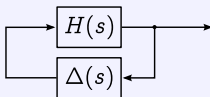


確かな系の Bode 線図

ロバスト制御の概要 (2)

小ゲイン定理

下図に示すシステムにおいて, $\|\Delta\|_{H^\infty} \triangleq \sup_{\omega \in \mathbb{R}} |\Delta(i\omega)| \leq \gamma$ をみたす任意の Δ に対し, システムが安定となるための必要十分条件は $\|H\|_{H^\infty} < \gamma^{-1}$ である.



ブロック線図の書き換え

ロバスト制御の概要 (3)

- よって次の伝達関数の H^∞ ノルム

$$\|H\|_{H^\infty} = \left\| \frac{WK}{1 + KP_0} \right\|_{H^\infty},$$

を解析することで安定性が判別できる

- この先の詳細な解析は、実は時間領域で行う
- 周波数領域と時間領域は次の関係で結ばれている:

$$\|F\|_{H^\infty} = \sup_{u \in L^2} \frac{\|Fu\|_{L^2}}{\|u\|_{L^2}}.$$

- ロバスト制御は、周波数領域と時間領域の 2つを横断する理論

現代制御の重要性

■ なぜ今日, 現代制御を教えたのか?

- 教えやすいから, などという怠慢な理由では決してない!
- 教えやすい(理解しやすい)ことは否定しませんが ...

現代制御の重要性

■ なぜ今日、現代制御を教えたのか？

- 教えやすいから、などという怠慢な理由では決していない！
- 教えやすい(理解しやすい)ことは否定しませんが ...

■ 多くの新しい制御が現代制御から生まれている

- 非線形制御(力学系, A. Lyapunov)
- 非線形な系の厳密な線形化(微分幾何学, A. Ishidori)
- 適応制御(機械学習, 数値最適化)
- モデル予測制御
- ハイブリッド制御
- 確率制御
- デジタル制御
- ネットワークド制御
- 量子制御
- Etc...

おわりに:制御理論の文化

■ 異分野交流に重点を置く

- 「制御」だけでは役に立たない
- 制御する「対象」の知識がなければ技術革新は望めない
- 現代社会では, 制御に限らずどの工学分野でも同じ

おわりに:制御理論の文化

■ 異分野交流に重点を置く

- 「制御」だけでは役に立たない
- 制御する「対象」の知識がなければ技術革新は望めない
- 現代社会では, 制御に限らずどの工学分野でも同じ

■ そのために必要なもの

- 自分の領域に閉じ込まれないこと(学生のうちはほどほどで良い)
- 自分の専門分野に精通すること

おわりに: 制御理論の文化

■ 異分野交流に重点を置く

- 「制御」だけでは役に立たない
- 制御する「対象」の知識がなければ技術革新は望めない
- 現代社会では, 制御に限らずどの工学分野でも同じ

■ そのために必要なもの

- 自分の領域に閉じ込まれないこと (学生のうちはほどほどで良い)
- 自分の専門分野に精通すること

皆様のご活躍を期待しています!