ディジタル制御 第 **3** 回

離散時間システムの表現 (2)
離散時間状態空間表現

準備

- $A(z), B(z)$ を z の多項式とし, $\rho(A(z)), \rho(B(z))$ をそれらの次数とする.
- 伝達関数 $\dfrac{B(z)}{A(z)}$ がプロパー であるとは$\rho(A(z)) \geq \rho(B(z))$ であることを言い, 厳密にプロパーであるとは $\rho(A(z)) > \rho(B(z))$ となること言う.

- 因果的なシステムの伝達関数はプロパーである．今回の講義では因果的なシステムのみを検討の対象とするので, 今後出て来る伝達関数はすべてプロパーである.

- $A(z) = z^N + \sum_{i=1}^{N} a_i z^{N-i}$, $B(z) = \sum_{i=0}^{N} \beta_i z^{N-i}$ とする． $Y(z) = \dfrac{B(z)}{A(z)} U(z)$ というシステムが与えられているとき,

$$\frac{B(z)}{A(z)} = \frac{\beta_0 z^N + \beta_1 z^{N-1} + \cdots \beta_N}{z^N + a_1 z^{N-1} + \cdots a_N}$$

$$= \beta_0 + \frac{(\beta_1 - a_1\beta_0) z^{N-1} + \cdots (\beta_N - a_N \beta_0)}{z^N + a_1 z^{N-1} + \cdots a_N}$$

.

- 以上の計算により, プロパーな伝達関数は, 定数と厳密にプロパーな伝達関数の和で書き表される. これからしばらく, この処理がすでに済んでいることを前提とし, 厳密にプロパーな伝達関数を中心に議論を進める.

- N 次の伝達関数は N 階の差分方程式に対応するが, 高階の差分方程式よりも 1 階の連立差分方程式の方が数学的な取り扱いが容易. そこで, この伝達関数から, 対応する 1 階の連立差分方程式を導くことを考える. このような操作を, 伝達関数の実現と呼ぶ.

- z 変換で初期値由来の繁雑な項が現れることを抑制するため，当面，計算の仮定で出てくる信号の初期値はすべて零と仮定する．この条件のもとで，z は時間軸に関する進み演算子に対応する．

伝達関数の実現

- 分母が N 次, 分子が $N-1$ 次以下の厳密にプロパーな (パルス) 伝達関数を考える:

$$G(z) = \frac{B(z)}{A(z)} = \frac{b_1 z^{N-1} + \cdots + b_N}{z^N + a_1 z^{N-1} + \cdots + a_N}.$$

入力を u, 出力を y とし, $U(z) = \mathcal{Z}[u]$, $Y(z) = \mathcal{Z}[y]$ とする.

- 天下り的であるが, $l = 1, \ldots, N$ に対し,

$$X_l(z) = \frac{1}{A(z)} z^{l-1} U(z)$$

と定義すると,

$$Y(z) = b_1 X_N(z) + b_2 X_{N-1}(z) + \cdots + b_N X_1(z)$$

である.

- $l = 1, \ldots, N-1$ に対し次式が成り立つ.

$$zX_l(z) = X_{l+1}(z)$$

- $l = N$ については,

$$
\begin{aligned}
zX_N(z) &= \frac{z^N}{A(z)} U(z) \\
&= \left(1 - \frac{a_1 z^{N-1} + \cdots + a_N}{A(z)} z^N\right) U(z) \\
&= U(z) - a_1 X_N(z) - \cdots - a_N X_1(z).
\end{aligned}
$$

記号の準備

$\boldsymbol{I}_{N-1}$: $N-1$ 次の単位行列

$\boldsymbol{0}_{N-1}$: $N-1$ 次の零ベクトル

$\boldsymbol{B}_c$: N 次で N 番目の単位ベクトル

$(=\boldsymbol{e}_N = (0,\ldots,0,1)^T)$

$\boldsymbol{C}_c = (b_N,\ldots,b_1)$(行ベクトル)

$$\boldsymbol{A}_c = \left(\begin{array}{c|c} \boldsymbol{0}_{N-1} & \boldsymbol{I}_{N-1} \\ \hline -a_N & \cdots \quad -a_1 \end{array}\right)$$ (行列の標記に丸括弧を用いる)

$\boldsymbol{X}(z) = (X_1(z),\ldots,X_N(z))^T$

- 以上の準備のもとで先程の式を纏めると以下のようになる.

$$
\begin{aligned}
z\boldsymbol{X}(z) &= \boldsymbol{A}_c\boldsymbol{X}(z) + \boldsymbol{B}_c U(z) \\
Y(z) &= \boldsymbol{C}_c\boldsymbol{X}(z).
\end{aligned}
$$

- 対応する差分方程式は, $\boldsymbol{x} = \mathcal{Z}^{-1}[\boldsymbol{X}(z)]$ とすると $\cdots$

$$\begin{aligned} \boldsymbol{x}(n+1) &= \boldsymbol{A}_c \boldsymbol{x}(n) + \boldsymbol{B}_c u(n) \\ y(n) &= \boldsymbol{C}_c \boldsymbol{x}(n). \end{aligned}$$

- このような表現を状態空間表現と呼び, 伝達関数から対応する状態空間表現を求めることを, 伝達関数の実現と呼ぶ.

- 厳密にプロパーでない伝達関数の実現は:

$$\boldsymbol{x}(n+1) = \boldsymbol{A}_c\boldsymbol{x}(n) + \boldsymbol{B}_c u(n)$$
$$y(n) = \boldsymbol{C}_c\boldsymbol{x}(n) + Du(n).$$

$Y(z) = G_0(z)U(z)$ で, 伝達関数 $G_0(z)$ がプロパーだが厳密にプロパーではないという状況を考える. 冒頭で述べた手順によって, $G_0(z) = D + G(z)$ かつ $G(z)$ が厳密にプロパーであるようにできる. $G(z)$ の実現は上記の手順で求めることができるが, $Y(z)$ については D に関する項を加える必要があり, 結果的に以下のようになる:

$Y(z) = \boldsymbol{C}_c\boldsymbol{X}(z) + DU(z)$

- 実現には無限個のバリエーションがあり, その中で, 先に挙げたものを可制御正準形と呼ぶ.
- 他に可観測正準形と呼ばれるもの (次ページ) もある.
- 可制御正準形は制御システムの安定化, 可観測正準形は状態観測器の構成にあたって重要 (後の講義で述べる).
- 再び $G(z)$ は厳密にプロパーとする.

記号の準備 (先程の表に追加)

$\boldsymbol{a} = (a_1, \ldots, a_N)^T$

$\boldsymbol{B}_o = (b_1, \ldots, b_N)^T$

$\boldsymbol{C}_o = (1, 0, \ldots, 0)$:

(N 次の 1 番目の単位ベクトル)

$$\boldsymbol{A}_o = \left(\begin{array}{c|c} & \boldsymbol{I}_{N-1} \\ \text{-}\boldsymbol{a} & \\ \hline & \boldsymbol{0}_{N-1}^T \end{array} \right)$$

$\boldsymbol{X}(z) = (X_1(z), \ldots, X_N(z))^T$

- 以上の準備のもとで, $G(z)$ に以下が対応する (これを逆 z 変換したものが可観測正準形).

$$\begin{aligned} z\boldsymbol{X}(z) &= \boldsymbol{A}_o\boldsymbol{X}(z) + \boldsymbol{B}_o U(z) \\ Y(z) &= \boldsymbol{C}_o\boldsymbol{X}(z) \end{aligned} \qquad (\bigstar)$$

証明: 記法の簡単のために, $a_0 = 1$ とおく. 帰納法により, $1 \le m \le N-1$ に対し,

$$\left(\sum_{l=0}^{m} a_l z^{m-l}\right) X_1(z) = X_{m+1}(z) + \left(\sum_{l=1}^{m} b_l z^{m-l}\right) U(z) \qquad (♪)$$

となることを示す.

- $m = 1$ のとき: これは (★) の第 1 式である.
- $m \le k$ に対して (♪) が正しいと仮定する. $m = k$ とし, 両辺に z を掛けてから (★) の第 $k+1$ 式を使うと,

$$\begin{aligned}\left(\sum_{l=0}^{k} a_l z^{k+1-l}\right) X_1(z) &= (-a_{k+1}X_1(z) + X_{k+2}(z) + b_{k+1}U(z)) \\ &\qquad + \left(\sum_{l=1}^{k} b_l z^{k+1-l}\right) U(z)\end{aligned}$$

  これを整理すると $k+1$ の場合の (♪) が得られる.

(♪) において $m = N-1$ として両辺に z を乗じてから (★) の第 N 式を使い, $a_0 = 1$ を思い出すと,

$$\left(z^N + a_1 z^{N-1} + \cdots a_N\right) X_1(z) = (b_1 z^{N-1} + \cdots + b_N) U(z)$$

である. $Y(z) = X_1(z)$ であったから,

$$Y(z) = G(z)U(z)$$

となることがわかる.

- 両辺を逆 z 変換すると, この伝達関数の実現は以下のようになる.

$$\boldsymbol{x}(n+1) = \boldsymbol{A}_o \boldsymbol{x}(n) + \boldsymbol{B}_o u(n),$$
$$y(n) = \boldsymbol{C}_o \boldsymbol{x}(n).$$

- 伝達関数が厳密にプロパーでない場合には

$$\boldsymbol{x}(n+1) = \boldsymbol{A}_o \boldsymbol{x}(n) + \boldsymbol{B}_o u(n),$$
$$y(n) = \boldsymbol{C}_o \boldsymbol{x}(n) + Du(n).$$

- 可制御正準形, 可観測正準形ともに, $\mathbf{0}_{N-1}$, $\boldsymbol{I}_{N-1}$ および $(a_1, \ldots, a_N)$, $(b_1, \ldots, b_N)$ の並べ方にバリエーションがある．教科書の可観測正準形は講義で述べたものと並べ方が異なる．後に述べる状態空間表現の自由度の関係で, 可制御正準形および可観測正準形にも何通りかの定義があり, 文献によって記述が異なる．さらに, 名称の付け方にもバリエーションがある．

- 連続時間伝達関数の実現 (状態空間における微分方程式への変換) の方法は, 離散時間と同様．上記で z を s に置き換え, s が微分演算子に対応することを思い出せば, 直ちに遂行できる．

状態空間表現

- $\boldsymbol{A} \in \mathbb{R}^{N\times N}$, $\boldsymbol{B} \in \mathbb{R}^{N\times M}$, $\boldsymbol{C} \in \mathbb{R}^{P\times N}$, $\boldsymbol{D} \in \mathbb{R}^{P\times P}$ とする $(N, M, P \in \mathbb{N})$.
- $\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{y}$ は因果的な信号で, $\boldsymbol{x}(n) \in \mathbb{R}^{N}$, $\boldsymbol{u}(n) \in \mathbb{R}^{M}$, $\boldsymbol{y}(n) \in \mathbb{R}^{P}$ とする.

- 離散時間システムの状態空間表現とは, 次の形の差分方程式である.

$$\boldsymbol{x}(n+1) = \boldsymbol{A}\boldsymbol{x}(n) + \boldsymbol{B}\boldsymbol{u}(n),$$
$$\boldsymbol{y}(n) = \boldsymbol{C}\boldsymbol{x}(n) + \boldsymbol{D}\boldsymbol{u}(n).$$

  上記第 1 式を状態方程式, 第 2 式を出力方程式, $\boldsymbol{x}$ を状態変数と呼ぶ.

- 可制御正準形および可観測正準形は, いずれも SISO システムの状態空間表現の例である.

- 状態空間表現を z 変換すると,

$$\begin{aligned} z\boldsymbol{X}(z) &= \boldsymbol{A}\boldsymbol{X}(z) + \boldsymbol{B}\boldsymbol{U}(z), \\ \boldsymbol{Y}(z) &= \boldsymbol{C}\boldsymbol{X}(z) + \boldsymbol{D}\boldsymbol{U}(n) \end{aligned}$$

となる. したがって,

$$\boldsymbol{Y}(z) = \left(\boldsymbol{C}(z\boldsymbol{I}_N - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D}\right)\boldsymbol{U}(z)$$

である. $\boldsymbol{C}(z\boldsymbol{I}_N - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D}$ を伝達関数行列と呼ぶ.

- 伝達関数行列は, 伝達関数を要素として持つ行列である.
- SISO システムでは,

$$Y(z) = \left(\boldsymbol{C}(z\boldsymbol{I}_N - \boldsymbol{A})^{-1}\boldsymbol{B} + D\right)U(z)$$

であり, これは伝達関数である.

- ある伝達関数 (行列) に対応する状態空間表現の無数に存在し (可制御正準形と可観測正準形はその (応用上重要な) 一種), そのバリエーションは, N 次の正則行列のバリエーションと一致する.

- $\boldsymbol{T}$ を正則行列とする. 以下に見るように, $\boldsymbol{T}$ をひとつ取るごとに, ある伝達関数 (行列) に対応する状態空間表現がひとつ定まる.

- 以下の状態空間表現を考える.

$$\boldsymbol{x}(n+1) = \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{T}\boldsymbol{x}(n) + \boldsymbol{T}^{-1}\boldsymbol{B}\boldsymbol{u}(n),$$
$$\boldsymbol{y}(n) = \boldsymbol{C}\boldsymbol{T}\boldsymbol{x}(n) + \boldsymbol{D}\boldsymbol{u}(n).$$

- 次ページで見るように, これに対応する伝達関数行列は $(\boldsymbol{C}(z\boldsymbol{I}_N - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D})\boldsymbol{U}(z)$ となり, 先に述べた伝達関数行列と一致する.
- 状態空間表現を z 変換してから計算すると…

$\boldsymbol{I}_N = \boldsymbol{T}^{-1}\boldsymbol{I}_N\boldsymbol{T}$ を使うと, 先の状態空間表現に対応する伝達関数行列は

$$
\begin{aligned}
&\boldsymbol{C}\boldsymbol{T}\left(z\boldsymbol{I}_N - \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{T}\right)^{-1}\boldsymbol{T}^{-1}\boldsymbol{B} + \boldsymbol{D} \\
&= \boldsymbol{C}\boldsymbol{T}\left(z\boldsymbol{T}^{-1}\boldsymbol{I}_N\boldsymbol{T} - \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{T}\right)^{-1}\boldsymbol{T}^{-1}\boldsymbol{B} + \boldsymbol{D} \\
&= \boldsymbol{C}\boldsymbol{T}\left(\boldsymbol{T}^{-1}\left(z\boldsymbol{I}_N - \boldsymbol{A}\right)\boldsymbol{T}\right)^{-1}\boldsymbol{T}^{-1}\boldsymbol{B} + \boldsymbol{D} \\
&= \boldsymbol{C}\boldsymbol{T}\boldsymbol{T}^{-1}\left(z\boldsymbol{I}_N - \boldsymbol{A}\right)^{-1}\boldsymbol{T}\boldsymbol{T}^{-1}\boldsymbol{B} + \boldsymbol{D} \\
&= \boldsymbol{C}\left(z\boldsymbol{I}_N - \boldsymbol{A}\right)^{-1}\boldsymbol{B} + \boldsymbol{D}
\end{aligned}
$$

となる. これは先ほどの伝達関数行列と同一である.

状態方程式の解

離散時間システムでは, 状態方程式から逐次代入により直ちに解が求められる. 初期値を $\boldsymbol{x}(0)$ とする.

$$
\begin{aligned}
\boldsymbol{x}(1) &= \boldsymbol{A}\boldsymbol{x}(0) + \boldsymbol{B}\boldsymbol{u}(0) \\
\boldsymbol{x}(2) &= \boldsymbol{A}\boldsymbol{x}(1) + \boldsymbol{B}\boldsymbol{u}(1) \\
&= \boldsymbol{A}\left(\boldsymbol{A}\boldsymbol{x}(0) + \boldsymbol{B}\boldsymbol{u}(0)\right) + \boldsymbol{B}\boldsymbol{u}(1) \\
&= \boldsymbol{A}^2\boldsymbol{x}(0) + \boldsymbol{A}\boldsymbol{B}\boldsymbol{u}(0) + \boldsymbol{B}\boldsymbol{u}(1) \\
&\dots
\end{aligned}
$$

以下帰納的に $\cdots$

$$\boldsymbol{x}(n) = \boldsymbol{A}^n \boldsymbol{x}(0) + \sum_{l=0}^{n-1} \boldsymbol{A}^{n-1-l} \boldsymbol{B}\boldsymbol{u}(l)$$

証明:　上式は $n = 1$ で正しい. $m \geq 1$ とし, $n \leq m$ で上式が正しいものと仮定する. $n = m + 1$ でも上式が成り立つことを確認する.

$$\begin{aligned}
\boldsymbol{x}(n+1) &= \boldsymbol{A}\boldsymbol{x}(n) + \boldsymbol{B}\boldsymbol{u}(n) = \boldsymbol{A}\left(\boldsymbol{A}^n \boldsymbol{x}(0) + \sum_{l=0}^{n-1} \boldsymbol{A}^{n-1-l} \boldsymbol{B}\boldsymbol{u}(l)\right) + \boldsymbol{B}\boldsymbol{u}(n) \\
&= \boldsymbol{A}^{n+1} \boldsymbol{x}(0) + \sum_{l=0}^{n-1} \boldsymbol{A}^{n+1-1-l} \boldsymbol{B}\boldsymbol{u}(l) + \boldsymbol{B}\boldsymbol{u}(n) \\
&= \boldsymbol{A}^{n+1} \boldsymbol{x}(0) + \sum_{l=0}^{n+1-1} \boldsymbol{A}^{n+1-1-l} \boldsymbol{B}\boldsymbol{u}(l)
\end{aligned}$$

可制御性と可観測性

- 大雑把に言うと, 可制御性は状態変数を目的とする位置に移動させる入力の系列が存在するか否かに関係した性質で, 可観測性は有限の出力信号の系列から状態変数を復元できるか否かに関連した性質. これらは, システムの性質を調べるにあたって重要.

- 歴史的に言うと, 可制御性と可観測性は, まず連続時間線形システムに関して調べられ, 続いて離散時間システムや非線形システム, 無限次元システムなどに拡張された.
- 拡張にあたって, 古い定義に不便な点が見付かったなどの理由により, 様々な手直しが施されたのだが, 今日でも古い定義を用いた教科書もある (この講義の教科書の定義は古い; 著者の専門は通信工学で制御ではないので $\cdots$).
- この講義では, 太田 (編): 現代制御, コロナ社, 2014 にしたがい, 現代的な定義を述べる.

- 再び, 状態空間表現されたシステムを考える.

$$
\begin{aligned}
\boldsymbol{x}(n+1) &= \boldsymbol{A}\boldsymbol{x}(n) + \boldsymbol{B}\boldsymbol{u}(n), \\
\boldsymbol{y}(n) &= \boldsymbol{C}\boldsymbol{x}(n) + \boldsymbol{D}\boldsymbol{u}(n).
\end{aligned}
\tag{$\bigstar$}
$$

- $(\bigstar)$ が可制御であるとは, 任意の $\boldsymbol{x}_f$ と任意の $\boldsymbol{x}(0)$ に対し, ある $k \geq 0$ とある入力の系列 $(\boldsymbol{u}(0), \ldots, \boldsymbol{u}(k-1))$ が存在し, 対応する $(\bigstar)$ の解が $\boldsymbol{x}(k) = \boldsymbol{x}_f$ となることをいう.
- $(\bigstar)$ が可観測であるとは, 任意の $\boldsymbol{x}(0)$ と任意の入力の系列 $(\boldsymbol{u}(n))_{n\in\mathbb{N}}$ に対し, ある $k \geq 0$ が存在し, $(\boldsymbol{u}(0), \ldots, \boldsymbol{u}(k))$ と $(\boldsymbol{y}(0), \ldots, \boldsymbol{y}(k))$ から $\boldsymbol{x}(0)$ が一意的に定まることをいう.

- 状態方程式の解を行列を使って書き直すと, 次式が得られる:

$$\boldsymbol{x}(k) = \boldsymbol{A}^k \boldsymbol{x}(0) + \begin{pmatrix} \boldsymbol{B} & \boldsymbol{AB} & \cdots & \boldsymbol{A}^{k-1}\boldsymbol{B} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}(k-1) \\ \vdots \\ \boldsymbol{u}(0) \end{pmatrix} \quad (♪)$$

- 行列 $\begin{pmatrix} \boldsymbol{B} & \boldsymbol{AB} & \cdots & \boldsymbol{A}^{k-1}\boldsymbol{B} \end{pmatrix}$ がフルランクであれば, (♪) を $\boldsymbol{x}_f$ に関する方程式と解釈して $(\boldsymbol{u}(0), \ldots, \boldsymbol{u}(k-1))$ について解くことができるから, (★) は可制御である. 一方, この行列がフルランクでない場合には, (♪) は必ずしも $\boldsymbol{x}_f$ について解けないから, (★) は可制御でない.

- Cayley-Hamilton の定理より, $\boldsymbol{A}^N$ は $\boldsymbol{I}, \boldsymbol{A}, \ldots, \boldsymbol{A}^{N-1}$ の線形結合で書けるから, $k = N$ の場合を考えれば十分で, k をそれより大きくしても無意味.

- 以上より，(★) が可制御であるための必要十分条件は，$\boldsymbol{M}_c = \begin{pmatrix} \boldsymbol{B} & \boldsymbol{AB} & \cdots & \boldsymbol{A}^{N-1}\boldsymbol{B} \end{pmatrix}$ としたとき，$\operatorname{rank} \boldsymbol{M}_c = N$ となることであるということがわかる．
- 行列 $\boldsymbol{M}_c$ を 可制御性行列，可制御行列 などと呼ぶ．
- システムが可制御であることを，$\operatorname{rank} \boldsymbol{M}_c = N$ であることと定義する流儀もある．

- 一方, 状態方程式の解と出力方程式を組み合わせて次式を得る.

$$\begin{pmatrix} \boldsymbol{y}(0) \\ \boldsymbol{y}(1) \\ \vdots \\ \boldsymbol{y}(k) \end{pmatrix} = \begin{pmatrix} \boldsymbol{C} \\ \boldsymbol{C}\boldsymbol{A} \\ \vdots \\ \boldsymbol{C}\boldsymbol{A}^{k-1} \end{pmatrix} \boldsymbol{x}(0) + \begin{pmatrix} \boldsymbol{D} & & \\ & \ddots & \\ * & & \boldsymbol{D} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}(0) \\ \boldsymbol{u}(1) \\ \vdots \\ \boldsymbol{u}(k) \end{pmatrix} \quad (♪)$$

  右辺第に項の行列はブロック下三角行列である. 対角要素の下の部分も計算できるが, 重要でないので略した.

- 可制御性に関する議論と同様の理由で, システムが可観測であるための必要十分条件は, この k に対して $\begin{pmatrix} \boldsymbol{C}^T & \boldsymbol{A}^T\boldsymbol{C}^T & \cdots & (\boldsymbol{A}^T)^{k-1}\boldsymbol{C}^T \end{pmatrix}^T$ がフルランクになることであるが, 再び Cayley-Hamilton の定理より, $k = N$ の場合のみを考えればよい.

- よって，(★) が可観測であるための必要十分条件は，$\boldsymbol{M}_o = \begin{pmatrix} \boldsymbol{C} \\ \boldsymbol{C}\boldsymbol{A} \\ \vdots \\ \boldsymbol{C}\boldsymbol{A}^{N-1} \end{pmatrix}$ としたとき，$\operatorname{rank} \boldsymbol{M}_o = N$ となることである．
- 行列 $\boldsymbol{M}_o$ を 可観測性行列, 可観測行列 などと呼ぶ．

- システムが可観測であることを, $\operatorname{rank} \boldsymbol{M}_o = N$ であることと定義する流儀もある.

連続時間と離散時間の状態空間表現

- 連続時間状態方程式

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}$$

を離散化することを考える．$\boldsymbol{x} \in \mathbb{R}^N$, $\boldsymbol{u} \in \mathbb{R}^M$ で, $\boldsymbol{A}$, $\boldsymbol{B}$ は適合する次元を持つものとする．サンプリング周期は T で, $[nT, (n+1)T)$ において入力は一定値であると仮定する．

$\boldsymbol{x}((n+1)T) = \exp[\boldsymbol{A}T]\boldsymbol{x}(nT) + \int_0^T \exp[\boldsymbol{A}(T-s)]\boldsymbol{B}\boldsymbol{u}(s)ds$ である. $t \in [nT, (n+1)T)$ において $\boldsymbol{u}(t) = \boldsymbol{u}(n)$(一定) と仮定すると,

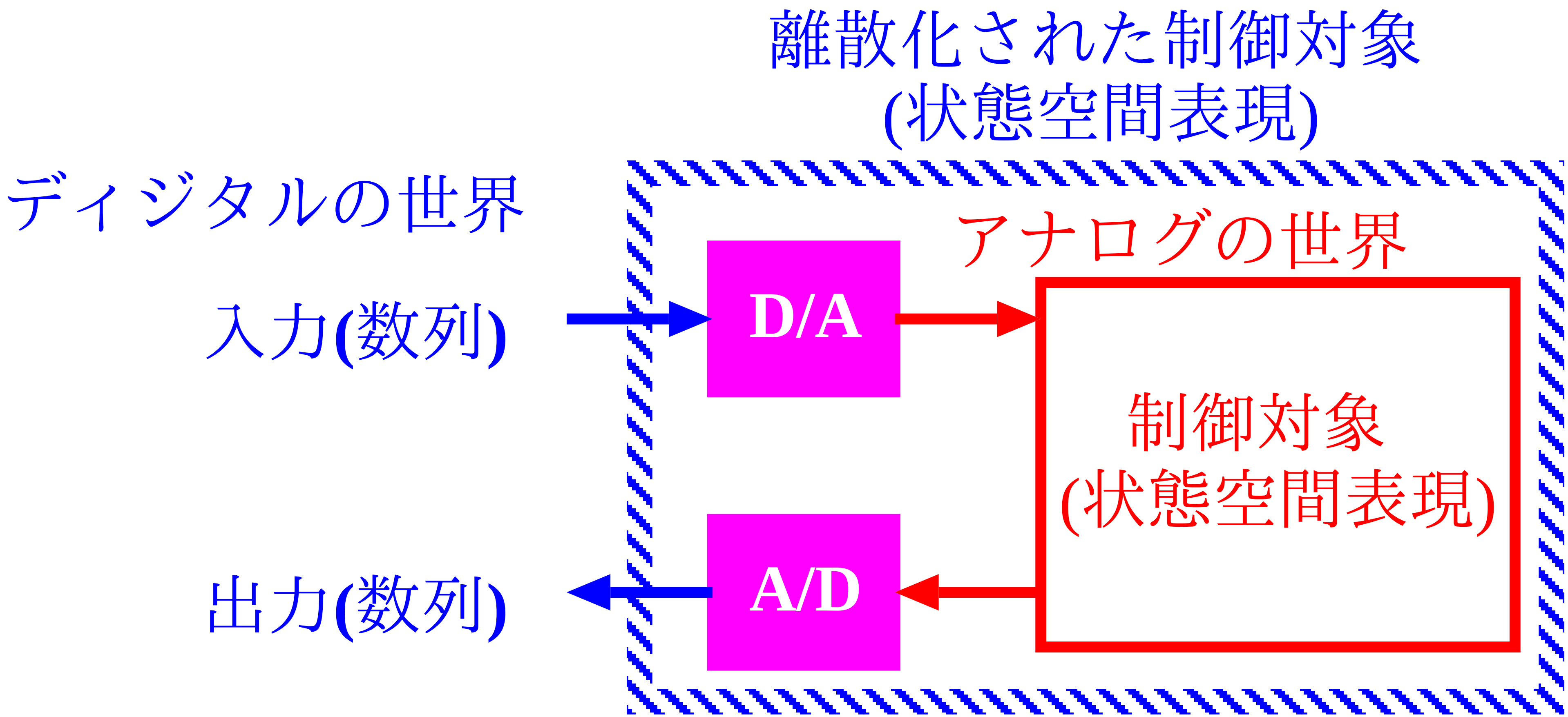
$$\begin{aligned}\boldsymbol{x}((n+1)T) &= \exp[\boldsymbol{A}T]\boldsymbol{x}(nT) \\ &\quad + \exp[\boldsymbol{A}T]\left(\int_0^T \exp[-\boldsymbol{A}s]ds\right)\boldsymbol{B}\boldsymbol{u}(n)ds.\end{aligned}$$

($\boldsymbol{x}(nT)$ などを $\boldsymbol{x}(n)$ などと略記している).

- $\boldsymbol{A}_D = \exp[\boldsymbol{A}T]$, $\boldsymbol{B}_D = \left(\int_0^T \exp[\boldsymbol{A}s]ds\right)\boldsymbol{B}$ とおけば, 上式より,

$$\boldsymbol{x}(n+1) = \boldsymbol{A}_D\boldsymbol{x}(n) + \boldsymbol{B}_D\boldsymbol{u}(n).$$

これが, 連続時間状態空間表現に対応する, これは離散化に伴う誤差を含まない離散時間の状態空間表現が構築できる.


離散化された制御対象
(状態空間表現)
ディジタルの世界
アナログの世界
入力(数列)
出力(数列)
D/A
A/D
制御対象
(状態空間表現)

- より簡単には, オイラー法による近似を用いるという手法がある.

$$\boldsymbol{x}(n+1) = (\boldsymbol{I}_N + T\boldsymbol{A})\,\boldsymbol{x}(n) + T\boldsymbol{B}\boldsymbol{u}(n).$$

ただし, $\boldsymbol{x}(nT)$ などを $\boldsymbol{x}(n)$ などと略記している. こちらはあくまで近似であり, 無視できない誤差を含む.

連続時間伝達関数の離散化

- 連続時間伝達関数 $G(s)$ に対応する 離散時間伝達関数を求めるには, $G(s)$ の実現 (状態空間表現) を経由するのが簡単.
- $G(s)$ の実現が以下の通りなら $\cdots$

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u,$$
$$y = \boldsymbol{C}\boldsymbol{x} + Du$$

- $t \in [nT, (n+1)T)$ において $\boldsymbol{u}(t) = \boldsymbol{u}(n)$ (一定) と仮定し, $\boldsymbol{A}_D = \exp[\boldsymbol{A}T]$, $\boldsymbol{B}_D = \left(\int_0^T \exp[\boldsymbol{A}s]ds\right)\boldsymbol{B}$ とおけば, 出力方程式は離散化で不変だから,

$$\boldsymbol{x}(n+1) = \boldsymbol{A}_D\boldsymbol{x}(n) + \boldsymbol{B}_D\boldsymbol{u}(n)$$

$$y(n) = \boldsymbol{C}\boldsymbol{x}(n) + Du(n).$$

- したがって，

$$Y(z) = \left(\boldsymbol{C}(z\boldsymbol{I} - \boldsymbol{A}_D)^{-1}\boldsymbol{B}_D + D\right) U(z).$$

- 以上で述べたように, 連続時間の伝達関数を一旦状態空間表現し, それを離散化することにより, 数学的に正当かつ離散化誤差を含まない形で離散時間伝達関数を求めることができるのだが ⋯

- 連続時間伝達関数から直接的に離散時間伝達関数を求める方法も (一応) ある.

- 文献に述べられた連続時間伝達関数の離散化の手順は数学的に正当性が保証されない部分を含むため，この講義では概略のみ述べ，詳細は略す．
- 量子化の影響は無視する．また，サンプリング周期を T とし，これを略さずに離散時間信号 x を $(x(nT))_{n\in\mathbb{N}}$ などと書く．

- 以下の議論では, 連続時間の Dirac のデルタ関数 (δ_c と書く) を用いる. 以下これを単にデルタ関数と呼ぶ. δ_c は, 関数 f がある性質を満たせば(この講義では深入りしない),

$$\int_{-\infty}^{\infty} f(t)\delta_c(t)dt = f(0)$$

となる.

- $\delta_c(t-nT)$ は デルタ関数を 時間軸に関して nT シフトしたものであり, よって

$$\int_{-\infty}^{\infty} f(t)\delta_c(t-nT)dt = f(nT)$$

である.

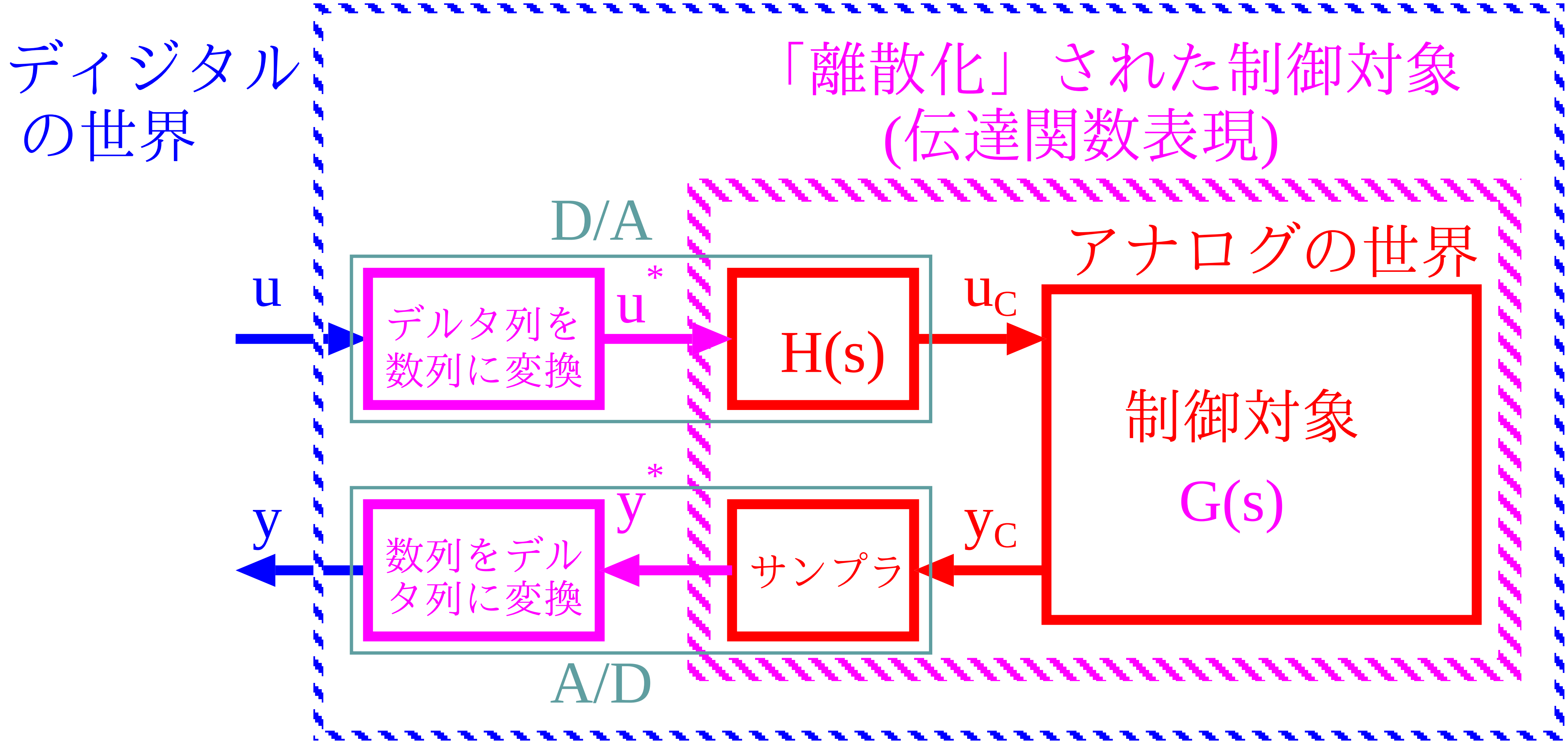
- 信号 $x = (x(nT))_{n\in\mathbb{N}}$ に $x^* = \displaystyle\sum_{n\in\mathbb{Z}} x(nT)\delta_c(t-nT)$ を対応させる．これを信号 x に対応するインパルス列と呼ぶ
- 連続時間信号 $x_c(t)$ の 標本化を, $x_c(t)$ に $m(t) = \sum_{n\in\mathbb{Z}} \delta(t-nT)$ を乗ずる演算と見倣す．この演算をおこなう素子を, 理想サンプラ と呼ぶ.

- D/A 変換を，数列 $x = (x(nT))$ を インパルス列に変換し，次に x^* を伝達関数 $H(s) = \dfrac{1 - e^{-sT}}{s}$ に通して x_C を得る処理と見做す．
  $H(s)$ のインパルス応答を h とすると，

$$h(t) = \begin{cases} 1, & 0 \leq t < T, \\ 0, & \text{それ以外} \end{cases}$$

  である．

- A/D 変換を, 連続時間信号 $x_C(t)$ をインパルス列 $x^* = \sum_{n\in\mathbb{N}} x(nT)\delta_C(t-nT)$ に変換し, 続いてそれを数列 $x = (x(nT))_{n\in\mathbb{N}}$ に変換する処理と解釈する.
- 次のページに, アナログとディジタルの世界のあいだに「中間表現」を含む図を示す. この「中間表現」は, コンピュータの処理を仮想的に分解したもので, 物理的な実体はない.


ディジタル
の世界
「離散化」された制御対象
(伝達関数表現)
D/A
アナログの世界
u
デルタ列を
数列に変換
u*
H(s)
uc
制御対象
G(s)
y*
yc
y
数列をデル
タ列に変換
サンプラ
A/D

- ラプラス変換作用素を $\mathcal{L}[\cdot]$, 逆ラプラス変換作用素を $\mathcal{L}^{-1}[\cdot]$, z 変換作用素を $\mathcal{Z}[\cdot]$ と書く.
- 積分と無限和の順序は交換できるものとする.
- 離散化の対象となる伝達関数が $G(s)$ から $G(s)H(s) = G(s)\frac{1-e^{-sT}}{s}$ に変わることに注意する. 記法の簡単のため, これを $G_H(s)$ と書く. $g_H(t) = \mathcal{L}^{-1}[G_H(s)]$ とする.

- u^* から y^* への (連続時間) 伝達関数を $G_H^*(s)$ とする. $G_H^*(s)$ は, u^* を連続時間単位インパルスとし, 対応する y^* をLaplace変換することで求められる. 具体的に計算すると,

$$
\begin{aligned}
G_H^*(s) &= \int_0^\infty g_H(t) \sum_{n\in\mathbb{N}} \delta_C(t-nT) e^{-st} dt \\
&= \sum_{n\in\mathbb{N}} g_H(nT) e^{-snT}
\end{aligned}
$$

- u から y へのパルス伝達関数を $G_D(z)$ とする. $G_D(z)$ は, u を離散時間単位インパルスとし, 対応する y を z 変換することで求められる. $y(n)$ と $y_C(nT)$ を対応させたことを思い出せば, $G_D(z) = \sum_{n=0}^{\infty} g_H(nT) z^{-n}$ である.
- 以上によって, $G_H^*(s) = G_D(z)\big|_{z=e^{sT}}$ である.

- 文献では $G_H(s)$ から $G_H^*(s)$ を求める方法も述べられているが, この部分の数学的な正当性は必ずしも保証されないため, この講義では 古田, ディジタルコントロール, コロナ社, 1989 から結果のみ引用する (ただし $\omega_k = 2\pi k/T$):

$$G_H^*(s) = \frac{1}{T}\sum_{k\in\mathbb{Z}} G_H(s + j\omega_k) + \frac{1}{2}g_H(0)$$

- デルタ関数に機械的に Fourier 級数展開の公式を適用し，積分と無限和の順序交換も機械的におこなうと，

$$G_H^*(s) = \frac{1}{T}\sum_{k\in\mathbb{Z}} G_H(s + j\omega_k)$$

となり，$\frac{1}{2}g_H(0)$ の項が消えるが，具体的な関数に上式を適用してみると，正しい伝達関数が計算されていないことが確認できる．