

単体近似を用いた非線形制御システムの解析

神戸女学院大学 潮 俊光 (Toshimitsu Ushio)

1. まえがき

制御システムには、飽和要素、不感帯要素、負性抵抗など様々な非線形素子が利用されている。これらの非線形特性によってシステムの平衡点が複数個存在したり、様々な分岐現象、振動現象が発生したりする。このような現象を解析するために様々な手法が提案されている。

本報告では、非線形制御システムの平衡点を単体近似法を用いて求める手法について述べる。このとき、線形制御部を可観測標準形に変換することにより、システムの出力の次元に等しい空間を単体分割することによって、全状態空間を単体分割しなくても解が求まることを示す。まず2章では、単体分割を用いて、いくつかの変数に関しては線形となっているような方程式に対して、その近似解を効率的に解くアルゴリズムについて述べる。このアルゴリズムはHsuとZhu⁽¹⁾の結果の拡張になっている。このアルゴリズムを非線形制御システムに応用した結果を3章で述べる。

2. 単体分割

非線形方程式

$$G(x) = Ax_c + g(x_s) = 0 \quad (1)$$

の解を単体近似を用いて求めることを考える. 但し, $x = [x_c^T, x_s^T]^T$, $x \in \mathbb{R}^N$, $x_c \in \mathbb{R}^n$, $x_s \in \mathbb{R}^m$, $N = n + m$ である. このとき, $\mathbb{R}^N$ を単体近似する必要はなく, x_c に関して式(1)は線形であることを利用すると, $\mathbb{R}^m$ だけを単体近似することにより計算量を減らすことができる.

$z_j \in \mathbb{R}^m (j=0, 1, \dots, m)$ に対して, $z_j - z_0 (j=1, 2, \dots, m)$ が線形独立であるとき, この $m+1$ 個のベクトルの凸包を単体といい, $\sigma(z_0, z_1, \dots, z_m)$ と書くことにする. $\mathbb{R}^N$ における $m+1$ 個の点 $x_j = [x_{c_j}^T, z_j^T]^T (j=0, 1, \dots, m)$ に対して, 混合単体を

$$\Delta(x_0, x_1, \dots, x_m) = \mathbb{R}^n \times \sigma(z_0, z_1, \dots, z_m)$$

と定義する^[2]. 明らかに, x_{c_j} に関係なく混合単体は決定される.

図1に $n=1$, $m=2$ の時の混合単体の例を示す.

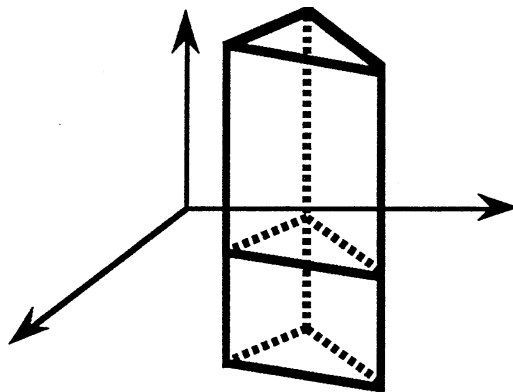


図1 混合単体の例

$\mathbb{R}^N$ の点 $x = [x_c^T, x_s^T]^T$ に対して混合単体 $\Delta(x_0, x_1, \dots, x_m)$ に関する混合重心座標 $[x_c^T, t^T]^T \in \mathbb{R}^n \times \mathbb{R}^{m+1}$ を次式で定義する.

$$x_s = \sum_{i=0}^m t_i z_i = [z_0, z_1, \dots, z_m] t \quad (2)$$

$$\sum_{i=0}^m t_i = 1$$

1次元(混合)ファセット $\mathbb{R}^n \times \{z_j\}$ ($j=0, 1, \dots, m$) 上で式(1)と等しくなるようなアフィン写像 F を考える.

$$F(x) = F(x_c, t) = A x_c + \sum_{i=0}^m t_i g_i$$

但し, $g_i = g(z_i)$ である. さらに, 式(2)より,

$$\begin{bmatrix} 1 \\ F(x_c, t) \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 0 \\ g_0 & g_1 & \dots & A \end{bmatrix} \begin{bmatrix} t \\ x_c \end{bmatrix} \quad (3)$$

となる. 従って, $F(x_c, t) = 0$ の解は,

$$\begin{bmatrix} 1 & 1 & \dots & 0 \\ g_0 & g_1 & \dots & A \end{bmatrix} \begin{bmatrix} t \\ x_c \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (4)$$

を解くことによって求められ, その解 $[x_c^T, t^T]^T$ が混合単体 $\Delta(x_0, x_1, \dots, x_m)$ の内部に存在するための必要十分条件は $t_j \geq 0$ ($j=0, 1, \dots, m$) である.

混合単体 Δ に対して, $d_j(\Delta)$ と $d(\Delta)$ を次のように定義する.

$$d_j(\Delta) = (-1)^j \det[g_0, \dots, \overline{g_j}, \dots, A_1, \dots, A_n] \quad j=0, \dots, m$$

$$(-1)^j \det[g_0, \dots, A_1, \dots, \overline{A_{j-m}}, \dots, A_n] \quad j=m+1, \dots, N$$

$$d(\Delta) = \sum_{j=0}^m d_j(\Delta)$$

但し, $A_j (j=1, 2, \dots, n)$ は A の第 j 行目の行ベクトル, $\overline{}$ はその行を除くことを表す. 解が非退化, すなわち $d(\Delta) \neq 0$ ならば, 式(4)の解は次式のようにになる.

$$t_j = (-1)^j d_j(\Delta) / d(\Delta) \quad j=0, 1, \dots, m$$

$$x_{c_j} = (-1)^{m+j} d_{m+j}(\Delta) / d(\Delta) \quad j=1, 2, \dots, n$$

このことより, 式(4)の解が Δ の内部にあるための必要十分条件は $d_j(\Delta) (j=0, 1, \dots, m)$ が同符号か 0 となることである.

以上のことより, 式(1)の近似解を求めるアルゴリズムは次のようになる.

[アルゴリズム]

- (I) 解を探す領域を混合単体で分割する. Σ をその混合単体分割とおく.
- (II) Σ が空集合ならば終了する. そうでなければ $\Delta \in \Sigma$ に対して以下の計算を行なう.
 - (i) $d_j(\Delta)$ を求める. その中で異符号のものがあれば(III)に飛ぶ.
 - (ii) $d(\Delta)$ を計算し, もし 0 ならばある x_j を微少に動かして, (i)に戻る.
 - (iii) $d_j(\Delta)$ より t , x_c を求め, さらに式(2)より x_s を求める.
- (III) $\Sigma = \Sigma - \{\Delta\}$ として(II)に戻る.

3. 非線形制御システムの動作点解析

ここでは、簡単のために図2に示す1入力1出力の非線形フィードバック制御システムを考える。

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A} \mathbf{x} + \mathbf{b} u \\ y &= \mathbf{c} \mathbf{x} \\ u &= r - f(y)\end{aligned}\tag{5}$$

但し、 $\mathbf{x} \in \mathbb{R}^n$, $u \in \mathbb{R}$, $y \in \mathbb{R}$ である。式(5)の平衡点は、

$$\mathbf{A} \mathbf{x} + \mathbf{b} (r - f(\mathbf{c} \mathbf{x})) = 0\tag{6}$$

を満足する。以下、 $(\mathbf{c}, \mathbf{A})$ は可観測対であると仮定すると、 $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$ は次のように書ける^[3]。

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & \cdots & 0 & -a_1 \\ 1 & 0 & \cdots & 0 & -a_2 \\ 0 & 1 & \cdots & 0 & -a_3 \\ & & \ddots & \vdots & \\ & & & \ddots & \\ & & & & 1 & -a_n \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix} \quad \mathbf{c} = [0 \quad \cdots \quad 1]$$

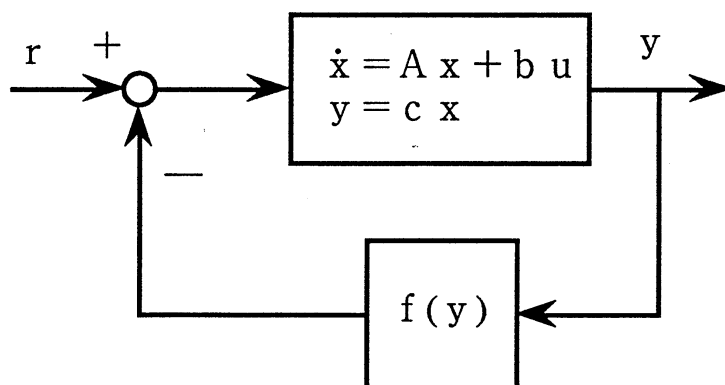


図2 非線形フィードバック制御システム

従って、式(6)は次のようになる。

$$\begin{bmatrix} 0 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ & & \ddots & \vdots \\ 0 & & & \ddots \\ & & & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ \vdots \\ x_{n-1} \end{bmatrix} + \begin{bmatrix} -a_1 x_n + b_1(r - f(x_n)) \\ -a_2 x_n + b_2(r - f(x_n)) \\ -a_3 x_n + b_3(r - f(x_n)) \\ \vdots \\ \vdots \\ -a_n x_n + b_n(r - f(x_n)) \end{bmatrix} = 0$$

$x_c = [x_1, x_2, \dots, x_{n-1}]^T$, $x_s = [x_n]$ とおくと、2章のアルゴリズムが適用できる。一般に、多入力多出力システムにおいても線形制御部を可観測標準形に変換して、 m が出力の次元に等しい混合単体分割を行なえばよい。

[例題]

式(4)において $n = 3$ の場合を考える。 $a_1 = -1$, $a_2 = 4$, $a_3 = 2$, $b_1 = 5$, $b_2 = 3$, $b_3 = 4$, $r = 0.5$ とおく。そして非線形フィードバック f を

$$f(y) = \tanh(y)$$

で表される飽和要素とする。ここでは、1次元空間の単体分割なので、区間に分割されることになる。各区間の長さはすべて同じ h とする。 h を変化させたときの近似解を表1に示す。この例題には3個の平衡点が存在している。特に解1は $h = 10^{-2}$ ですでに良い近似になっている。

表1 例題における平衡点の近似解

	h	x 1	x 2	x 3
解 1	10^{-2}	-34.4999859284	-20.9999914346	-7.4999969409
	10^{-4}	-34.4999859284	-20.9999914346	-7.4999969409
	10^{-6}	-34.4999859284	-20.9999914346	-7.4999969409
	10^{-8}	-34.4999859284	-20.9999914346	-7.4999969409
解 2	10^{-2}	3.6289178884	2.2089065408	0.7888951931
	10^{-4}	3.6288719149	2.2088785569	0.7888951988
	10^{-6}	3.6288719090	2.2088785533	0.7888951976
	10^{-8}	3.6288719090	2.2088785533	0.7888951976
解 3	10^{-2}	11.1404281096	6.7811301537	2.4218321977
	10^{-4}	11.1404404973	6.7811376940	2.4218348907
	10^{-6}	11.1404404992	6.7811376951	2.4218348911
	10^{-8}	11.1404404992	6.7811376951	2.4218348911

参考文献

- [1] Hsu, C. S. & Zhu, W. H., "A Simplicial Mapping Method for Locating the Zeros of a Function", Quart. Appl. Math., Vol.XLII, No.1, pp.41-59 (1984).
- [2] 潮俊光, "混合単体分割による非線形方程式の近似解法", 電子情報通信学会技術報告書, NLP88-12, pp.19-24, (1988).
- [3] Kailath, T., *Linear Systems*, Prentice-Hall, Englewood Cliffs, N.J., (1980).