

制御理論講話 (その 5) —モデル予測制御—

A Lecture on Control Theory (Part5) — Model Predictive Control—

渡利 久規, 山崎 敬則, 古瀧 雅和*, 黒須 茂**

Hisaki WATARI, Takanori YAMAZAKI, Masakazu KOTAKI*, Shigeru KUROSU**

1. はじめに

1970 年代後半から, ヨーロッパやアメリカにおいてモデル予測制御が多くのプラントに適用され, 実績をあげてきた。わが国においても, ようやくそれらの成果を背景にモデル予測制御の現場への適用が急速に増えている。とくにプロセス産業における CIM(Computer Integrated Manufacturing)が模索されている今日, 制御の基本アルゴリズムのひとつとして受け止められている。

モデル予測制御が適用されたプロセスを見ると, つぎのような背景があった。

- 1) プラントに大きなむだ時間があったり, 応答がおそく制御動作の決定をオペレータの判断に頼ってきた。さらに入出力についての制約条件が重要で, これらを考慮した操作の決定にもオペレータの判断が必要とされた。
- 2) プラント入出力間の相互干渉を非干渉化することに関心が集まり, 非干渉化を積極的に利用する多変数制御に懐疑的であった。
- 3) 安定で効率のよいオペレーションを基礎にして最適化制御への要望が高まった。

本稿では, プロセス制御の分野において優れた実績をもち, 確固たる地位を築いているモデル予測制御の基本的な考え方を紹介し, 内部モデル制御との関連について述べる。

今年は異常に涼しかった夏も終わって, 台風が来るわ, 地震は来るわ, 政界でもプロ野球でも権力争いで日本中大騒ぎしている内に, 肌寒さを感じる秋

になってしまった。相も変らぬ極楽とんぼのわが長屋の住人八つあん, 熊さんは, ご隠居の振舞うお酒ですっかり酩酊してしまった。それでも, ご隠居の壊れた蓄音機のような講話は延々と続くのであった。

2. モデル予測制御の基本的考え方¹⁾

熊さん「ご隠居。内部モデル制御についてはいくらかわかってきましたので, つぎに現場でそれなりに実績をあげている制御方式を話してくれませんか?」
ご隠居「それでは, モデル予測制御について話すことにしよう。二人とも寝るなよ。その昔, 寺子屋の真似ごとをしていたとき, 生徒の中で居眠りをして連中に限って, 「教え方が悪い」とか「バカにした」とか匿名の紙飛行機を飛ばすのには往生したよ」
熊さん「よして下さいよ。わしら, これでも江戸っ子ですぜ。北関東の片田舎の餓鬼どもと同じにされちゃあ, 立つ瀬がありませんぜ。なあ, 八つあん?」
八つあん「うんだ。うんだ。ご隠居。徳利が空いてますがな」

ご隠居「大体, おまえたちはどこの国の言葉を喋っているのかね。まあ, よい。座れ。」

モデル予測制御の基本的考え方を述べると, “プラントの制御量, 操作量のモデルを使って, 現時刻より未来の制御量の挙動を予測し, その予測される挙動ができるだけ希望とする挙動になるように操作量を定める。この決定の手続きをサンプリング時刻ごとに繰り返す” ということになる

熊さん「プラントが予測していた動きと異なる動きをした場合, オペレータは目標値に近づくように操作量を決め直してプラントの動きを眺める。まさに熟練オペレータのやっていることをそっくり真似し

*) 平成 15 年度機械工学科在学中

**) 研究所 “Crotech”

ているわけですね」

ご隠居「まさに、その通りじゃ。この手続きの中には、操作量に対してプラントがどのように動くかということの知識(モデル)に基づく予測がある。つぎに、“プラントの状態の観測”，“制御量の動きの予測”，“操作量の決定”の手続きが繰り返し行われている。この繰り返しこそ、モデル予測制御の原点なのじゃ」

3. 基本概念

ご隠居「それでは予測制御アルゴリズムを用語の説明も兼ねて紹介することにしよう。Fig.1を見てもらいたい。現時刻からある区間(制御ホライズン(control horizon)という)にわたり、入力を変化させたとき、ステップ先にわたる区間 $[t+1, t+P]$ で(予測ホライズン(prediction horizon)という)出力がどのような挙動をとるかを、プラントの入出力間の動的モデルを使って予測する。その予測値が目標値にできるだけ近づくように、入力 $(u(t), u(t+1), \dots, u(t+M-1))$ を、入力や出力に課せられた制約を満たす範囲の中で求める。計算された M ステップの入力のうち、最初の1ステップである $u(t)$ だけを実際にプロセスに加える。つぎのサンプル時刻では、新たに得られた出力の測定値 $y(t+1)$ を加味して出力の予測をやり直す。その際、制御ならびに予測ホライズンを1ステップずらし、時刻 $t+1$ から、 P ステップ先にわたる予測値が目標値にできるだけ近づくように、時刻 $t+1$ から M ステップにわたる入力を決めるという最適化問題を再び繰り返す。このようにサンプル時刻ごとに各ホライズンをずらし、入力決定のために最適化問題を解くこと(これを **moving horizon** 法、または **receding horizon** 法という)がモデル予測制御アルゴリズムの基本的考え方である。」

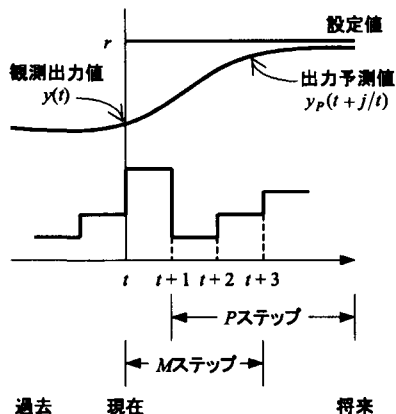


Fig.1 モデル予測制御

熊さん「内部モデル制御では、連続時間系だったのが、どうしてモデル予測制御では、離散時間系となり、1ステップごとに入力を求めるのですか?」

ご隠居「やはり、有限次元の最適化問題に結びつけるには、連続時間系よりも離散時間系でないとなまずい。だから、サンプリング周期が長くて、計算時間が問題とならないプロセス制御にモデル予測制御が適しているともいえるがね」

4. 制御アルゴリズム

ご隠居「モデル予測制御アルゴリズムを実行するには、

1. 出力の挙動を計算するためのモデルの構築
2. 外乱・モデル誤差を考慮した出力の予測
3. 操作量の決定

をいかに行うかを定めなければならない。以下、具体的にアルゴリズムを説明することにしよう」

4.1 出力の挙動を計算するためのモデル

(1) ステップ応答モデル

ご隠居「Fig.2に示すような大きさ1のステップ状入力に対する応答をステップ応答と呼ぶことは熊さんも知っているだろう。ステップ応答の各サンプル時刻における値を $\{a_i\}$ と表わすことにする。ここで $t=s$ は、出力が定常になったとみなせる時間である。すなわち

$$a_i = a_s, i \geq s \quad (1)$$

とする。なお L ステップのむだ時間が存在する場合には、

$$a_i = 0, i = 1, 2, \dots, L \quad (2)$$

となる。

プロセスが線形であると仮定すると、時刻 t で大きさ $\Delta u(t)$ のステップ状入力を加える場合、この入力に対する時刻 $t+j$ での出力の大きさは

$$y_M(t+j) = a_j \Delta u(t) = a_j \{u(t) - u(t-1)\} \quad (3)$$

で与えられる。時刻 $t-1$ で $\Delta u(t-1)$ の入力が変化したことに対しては、 $t+j$ での出力の大きさは

$$y_M(t+j) = a_{j+1} \Delta u(t-1) = a_{j+1} \{u(t-1) - u(t-2)\} \quad (4)$$

で与えられる。

この考え方を拡張して、Fig.3のように入力がサンプル時刻ごとにステップ状に変化して、連続的にプラントに入ってきたときの出力の動きを考えよう。

まず、Fig.3(a)のような入力を、Fig.3(b)のように、いくつものステップ入力に分解して考える。このとき、時刻 $t-k$ での入力のステップ変化は、 $\Delta u(t-k) = u(t-k) - u(t-k-1)$ となる。こうして分解

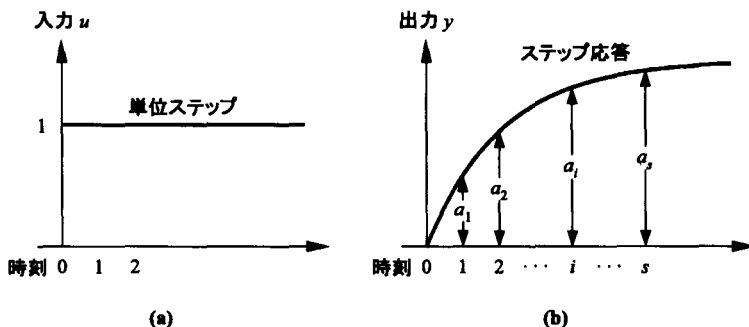


Fig.2 ステップ応答(離散系)

した入力 $\Delta u(t+j)$, $\Delta u(t+j-1)$ から $\Delta u(-\infty)$ までの入力をプロセスに加えたことによって、時刻 $t+j$ で得られる出力の値は、重ね合わせの原理を用いて、

$$y_M(t+j) = \sum_{k=1}^{\infty} a_k \Delta u(t+j-k) \quad (5)$$

$$= \sum_{k=1}^{s-1} a_k \Delta u(t+j-k) + a_s u(t+j-s)$$

のように表わすことができる」

熊さん「 $\Delta u(t)$ として操作量の変化分として求める理由は何ですか？」

ご隠居「直接、操作量 $u(t)$ を与える方式を位置形といい、1 サンプル周期分の操作量の変化分 $\Delta u(t)$ をコンピュータの出力として操作部に与える方式を速度形とよんでいる。これは故障のさいに、いきなり操作量が極限まで変化するような事態をさけるためなのだ」

八つあんな「ちょっと、待って下さい。(5)式の第2項目はどこから出てきたんですか？」

ご隠居「 $i \geq s$ となる時刻で、出力の値は、 $a_i = a_s$ とみなし、

$$\sum_{k=s}^{\infty} a_k \Delta u(t+j-k)$$

$$= a_s \sum_{k=s}^{\infty} \Delta u(t+j-k)$$

$$= a_s \{ \Delta u(t+j-s) + \Delta u(t+j-(s+1)) + \dots \}$$

$$= a_s u(t+j-s) \quad (6)$$

がなりたつ。これは入力のステップ変化に対して、応答が最終的には a_s に漸近することを前提としているのだ」

八つあんな「ということは、モデル予測制御は安定なプラントしか適用できないんですね？」

ご隠居「1980年代に提案された当初はそうであったが、その後、改良が加えられて積分系、不安定系にも適用できるようになっている」。

現時刻を t とし、いままでにプラントに加えられた入力と、いまから加えられるはずの未来の入力に分けて、(5)式を見直すと、

$$y_M(t+j) = \sum_{i=0}^{M-1} a_{j-i} \Delta u(t+i) + \sum_{i=1}^{s-1} a_{j+i} \Delta u(t+i)$$

$$\underbrace{\Delta u(t) \cdots \Delta u(t+M-1)}_{M \text{ 個の未来の入力}} + \underbrace{\Delta u(t-1) \cdots \Delta u(t-s+1)}_{\text{過去の入力}}$$

$$+ a_s u(t-s) \quad (7)$$

となる。さらに、現時刻 t の出力が

$$y_M(t) = \sum_{i=1}^{\infty} a_i \Delta u(t-i)$$

$$= \sum_{i=1}^{s-1} a_i \Delta u(t-i) + a_s u(t-s) \quad (8)$$

と表わせることを用いて、時刻 $t+j$ で得られる出力の値は、(8)式を(7)式に代入して

$$y_M(t+j) = y_M(t) + \sum_{i=0}^{M-1} a_{j-i} \Delta u(t+i)$$

$$+ \sum_{i=1}^{s-1} (a_{j+i} - a_i) \Delta u(t-i) \quad (9)$$

(9)式を用いて、 L ステップまでむだ時間が存在する場合に、 $y_M(t+j)$ の値を $j=L$ から $L+P-1$ まで表現すると、

$$\begin{bmatrix} y_M(t+L) \\ y_M(t+L+1) \\ \vdots \\ y_M(t+L+P-1) \end{bmatrix} = \begin{bmatrix} y_M(t) \\ y_M(t) \\ \vdots \\ y_M(t) \end{bmatrix}$$

$$+ \begin{bmatrix} a_L & a_{L-1} & \cdots & a_{L-M+1} \\ a_{L+1} & a_L & \cdots & a_{L-M+2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{L+P-1} & \cdots & \cdots & a_{L-M+P} \end{bmatrix} \begin{bmatrix} \Delta u(t) \\ \Delta u(t+1) \\ \vdots \\ \Delta u(t+M-1) \end{bmatrix}$$

$$+ \begin{bmatrix} a_{L+1}-a_1 & a_{L+2}-a_2 & \cdots & a_{L+s-1}-a_{s-1} \\ a_{L+2}-a_1 & a_{L+3}-a_2 & \cdots & a_{L+s}-a_{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{L+P}-a_1 & \cdots & \cdots & a_{L+P+s-2}-a_{s-1} \end{bmatrix} \begin{bmatrix} \Delta u(t-1) \\ \Delta u(t-2) \\ \vdots \\ \Delta u(t-s+1) \end{bmatrix} \quad (10)$$

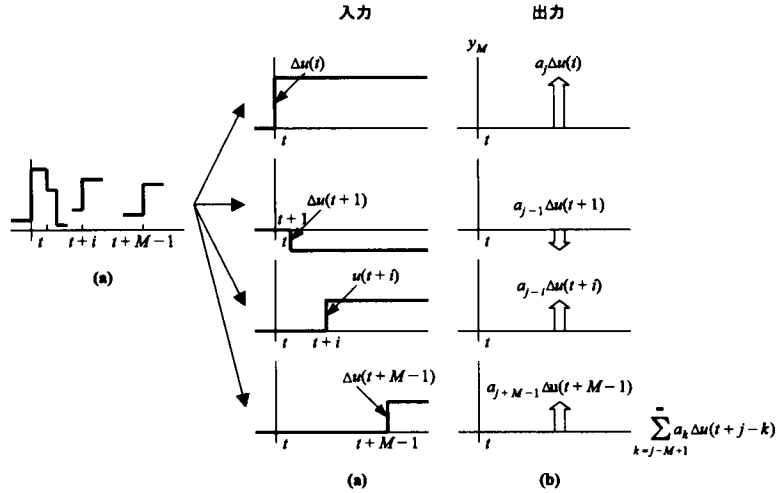


Fig.3 ステップ応答の重ね合わせ

ただし、ここで $i < 0$ に対しては、 $a_i = 0$ とする。

さらに、(10)式をベクトル・行列で表現すると

$$y_M = y_{M0} + A_F \Delta u_n + A_0 \Delta u_0 \quad (11)$$

となる。ここで

y_{M0} : 時刻 t のモデルの出力

Δu_n : 時刻 t 以降のモデルの出力

Δu_0 : 過去の入力

ということになる」

熊さん「だいたい、ごちゃごちゃしてきましたが、紙と鉛筆で理解できそうです。これを使って、出力 y_M を予測するんですね。ステップ応答が求められないようなプラントではどうするんです？」

(2) パラメトリックモデル

ご隠居「プラントのインパルス応答やステップ応答を用いて記述されるモデルをノンパラメトリックモデルとよんでいる。それに対して、現時刻 t の出力が入力の過去の値の関数として決まるようなモデルをパラメトリックモデル⁸⁾という。たとえば、

$$y_M(t) = a_1 y_M(t-1) + a_2 y_M(t-2) + \dots + a_n y_M(t-n) + b_1 u(t-1) + b_2 u(t-2) + \dots + b_m u(t-m) \quad (12)$$

(12)式を用いて、現時刻より j ステップ未来の時刻 $t+j$ の出力を表わすと、

$$y_M(t+j) = a_1 y_M(t+j-1) + a_2 y_M(t+j-2) + \dots + a_n y_M(t+j-n) + b_1 u(t+j-1) + b_2 u(t+j-2) + \dots + b_m u(t+j-m) \quad (13)$$

となり、 $j \geq 2$ の時には、 $y_M(t+j)$ は未来の出力値 $y_M(t+j-1), \dots, y_M(t+j)$ の関数となる。これは今から決める入力 $u(t), \dots, u(t+j-2)$ の関数でもあり、

入力を決める手続きが複雑になる。 $y_M(t+j)$ の値を、過去の出力値と入力の関数として表わすことを考えよう。

(13)式を用いて時刻 $t+1$ の出力を求めてみよう。

$$y_M(t+1) = a_1 y_M(t) + a_2 y_M(t-1) + \dots + a_n y_M(t-n+1) + b_1 u(t) + b_2 u(t-1) + \dots + b_m u(t-m+1) \quad (14)$$

(14)式より(12)式を引くと、

$$\Delta y_M(t+1) = y_M(t+1) - y_M(t) = a_1 \Delta y_M(t) + a_2 \Delta y_M(t-1) + \dots + a_n \Delta y_M(t-n+1) + b_1 \Delta u(t) + b_2 \Delta u(t-1) + \dots + b_m \Delta u(t-m+1) \quad (15)$$

ここに、

$$\Delta y_M(k) = y_M(k) - y_M(k-1)$$

$$\Delta u(k) = u(k) - u(k-1)$$

である。

(15)式を用いて、時刻 $t+2$ の出力を求めると

$$\Delta y_M(t+2) = a_1 \Delta y_M(t+1) + a_2 \Delta y_M(t) + \dots + a_n \Delta y_M(t-n+2) + b_1 \Delta u(t+1) + b_2 \Delta u(t) + \dots + b_m \Delta u(t-m+2) \quad (16)$$

(16)式の右辺に $-y_M(t) + y_M(t)$ を加えると、

$$\Delta y_M(t+2) = y_M(t+2) - y_M(t+1) + y_M(t+1) - y_M(t) = a_1 \Delta y_M(t+1) + a_2 \Delta y_M(t) + \dots + a_n \Delta y_M(t-n+2) + b_1 \Delta u(t+1) + b_2 \Delta u(t) + \dots + b_m \Delta u(t-m+2) \quad (17)$$

となる。(15)式より

$$\begin{aligned} \Delta y_M(t+1) &= y_M(t+1) - y_M(t) \\ &= a_1 \Delta y_M(t) + a_2 \Delta y_M(t-1) + \dots + a_n \Delta y_M(t-n+1) \\ &\quad + b_1 \Delta u(t) + b_2 \Delta u(t-1) + \dots + b_m \Delta u(t-m+1) \end{aligned} \quad (18)$$

を導いて、(17)式の $\Delta y_M(t+1)$ の項に代入すると

注) ARX(Auto Regressive Exogenous model)(自己帰帰+外生入力型)モデルともいう。

$$\begin{aligned}
y_M(t+2) = & y_M(t) + \{(1+a_1)a_1+a_2\} \Delta y_M(t) + \dots \\
& + \{(1+a_1)a_{n-1}+a_n\} \Delta y_M(t-n+2) \\
& + (1+a_1)a_n \Delta y_M(t-n+1) \\
& + b_1 \Delta u(t+1) + \{b_2+(1+a_1)b_1\} \Delta u(t) + \dots \\
& + \{b_m+(1+a_1)b_{m-1}\} \Delta u(t-m+2) \\
& + (1+a_1)b_m \Delta u(t-m+1)
\end{aligned} \quad (19)$$

同様な手続きを踏むことにより、 $y_M(t+j)$ を現在の出力の測定値と過去の入出力値と、いまから決める入力の間関数として

$$\begin{aligned}
y_M(t+j) = & y_M(t) \\
& + q_{j,1} \Delta y_M(t) + q_{j,2} \Delta y_M(t-1) + \dots \\
& + q_{j,n} \Delta y_M(t-n+1) \\
& + g_{j,j-1} \Delta u(t+j-1) + g_{j,j-2} \Delta u(t+j-2) \\
& + g_{j,j-m-1} \Delta u(t+j-m+1)
\end{aligned} \quad (20)$$

と表わすことができる」

熊さん「ご隠居。もっとシステムティック行方方法はないですかね?」

ご隠居「(12)式を遅延演算子 z^{-1} を用いて表わすと、 $y_M(t) = \{a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}\} y_M(t)$

$$+ \{b_1 z^{-1} + b_2 z^{-2} + \dots + b_m z^{-m}\} u(t) \quad (12)$$

となり、パルス伝達関数を用いると

$$\begin{aligned}
y_M(t) = & \frac{B(z^{-1})}{A(z^{-1})} u(t) \\
= & \frac{b_1 z^{-1} + b_2 z^{-2} + \dots + b_m z^{-m}}{1 - a_1 z^{-1} - a_2 z^{-2} - \dots - a_n z^{-n}} u(t)
\end{aligned} \quad (21)$$

となる。

さて、 z^{-1} の多項式 $A(z^{-1})$ に対して、Diophantine 方程式

$$E(z^{-1})(1-z^{-1})A(z^{-1}) + z^{-j}F(z^{-1}) = 1 \quad (22)$$

を満たす $E(z^{-1})$ 、 $F(z^{-1})$ が存在したとする。(21)式を用いて、時刻 $t+j$ の出力を表わすと、

$$A(z^{-1})y_M(t+j) = B(z^{-1})u(t+j)$$

となり、両辺に $(1-z^{-1})E(z^{-1})$ をかけると、

$$(1-z^{-1})E(z^{-1})A(z^{-1})y_M(t+j) =$$

$$E(z^{-1})B(z^{-1})(1-z^{-1})u(t+j)$$

(22)式より、左辺は

$$(1-z^{-j}F(z^{-1}))y_M(t+j) =$$

$$E(z^{-1})B(z^{-1})(1-z^{-1})u(t+j)$$

となる。 $z^{-j}y_M(t+j) = y_M(t)$ 、 $(1-z^{-1})u(t+j) = \Delta u(t+j)$ であるから

$$y_M(t+j) = F(z^{-1})y_M(t) + E(z^{-1})B(z^{-1})\Delta u(t+j) \quad (23)$$

と表わせる。

一方、(22)式において、 $z^{-1} = 1$ のときを考えると、

$$F(1) = 1$$

を満たさなければならない。つまり、 $(1-F(z^{-1}))$ は $(1-z^{-1})$ で割り切れなければならない。すなわち、

$$F(z^{-1}) = (1-z^{-1})Q(z^{-1}) + 1 \quad (24)$$

がなりたつ。ただし、 $Q(z^{-1})$ は $F(z^{-1}) - 1$ を $(1-z^{-1})$ で割った商である。(24)式を(23)式に代入すると、

$$y_M(t+j) = y_M(t) + Q(z^{-1})\Delta y_M(t) + E(z^{-1})B(z^{-1})\Delta u(t+j) \quad (25)$$

となり、ここで

$$\left. \begin{aligned}
Q_j(z^{-1}) &= q_{j,1} + q_{j,2}z^{-1} + q_{j,3}z^{-2} + \dots \\
E_j(z^{-1})B(z^{-1}) &= g_{j,j-1}z^{-1} + g_{j,j-2}z^{-2} + \dots
\end{aligned} \right\} \quad (26)$$

のような多項式が存在したとすると、(25)式は(20)式と同じ形式で書けるのじゃよ」

熊さん「Diophantine方程式を満たす $E(z^{-1})$ 、 $F(z^{-1})$ を求めることなんか、できるんですかい?」

ご隠居「 $A(z^{-1})$ の係数さえ与えれば $E(z^{-1})$ 、 $F(z^{-1})$ を求めるプログラムが文献1)にでているから、心配は無用じゃよ」

八つあん「さて、(20)式からどう処理するんですかい?」

ご隠居「(20)式を用いて、 $y_M(t+j)$ の出力を $j = L$ から $L+P+1$ まで表現すると、

$$\begin{aligned}
& \begin{bmatrix} y_M(t+L) \\ y_M(t+L+1) \\ \vdots \\ y_M(t+L+P-1) \end{bmatrix} = \begin{bmatrix} y_M(t) \\ y_M(t) \\ \vdots \\ y_M(t) \end{bmatrix} \\
& + \begin{bmatrix} g_{L,L} & g_{L,L-1} & \dots & 0 \\ g_{L+1,L+1} & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ g_{L+P-1,L+P-1} & \dots & \dots & g_{L+P-1,L+P-m} \end{bmatrix} \begin{bmatrix} \Delta u(t) \\ \Delta u(t+1) \\ \vdots \\ \Delta u(t+M-1) \end{bmatrix} \\
& + \begin{bmatrix} g_{L,L+1} & g_{L,L+2} & \dots & g_{L,L+m-1} \\ g_{L+1,L+2} & g_{L+1,L+3} & \dots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ g_{L+P-1,L+P} & \dots & \dots & g_{L+P-1,L+P+m-1} \end{bmatrix} \begin{bmatrix} \Delta u(t-1) \\ \Delta u(t-2) \\ \vdots \\ \Delta u(t-m+1) \end{bmatrix} \\
& + \begin{bmatrix} q_{L,1} & q_{L,2} & \dots & q_{L,n} \\ q_{L+1,1} & \dots & \dots & q_{L+1,n} \\ \vdots & \vdots & \vdots & \vdots \\ q_{L+P-1,1} & \dots & \dots & q_{L+P-1,n} \end{bmatrix} \begin{bmatrix} \Delta y_M(t) \\ \Delta y_M(t-1) \\ \vdots \\ \Delta y_M(t-n+1) \end{bmatrix}
\end{aligned} \quad (27)$$

(27)式をベクトル・行列で表現すると

$$y_M = y_{M0} + G_F \Delta u_n + G_0 \Delta u_0 + Q \Delta y_{Mold} \quad (28)$$

となり、時刻 $t+L$ から P ステップにわたる未来の出力の値

$$y_M(t+L), y_M(t+L-1), \dots, y_M(t+L+P-1),$$

を計算することができるわけじゃよ」

熊さん「ご隠居. この辺でまとめてくれないか?」
 ご隠居「出力の挙動を予測するためにモデルが必要であることを述べ, その中で代表的な2つのモデル

ステップ応答モデル:

$$y_M = y_{M0} + A_F \Delta u_n + A_0 \Delta u_0 \quad (29)$$

パラメトリックモデル:

$$y_M = y_{M0} + G_F \Delta u_n + G_0 \Delta u_0 + Q \Delta y_{Mold}$$

時刻 $t+L$ から P ステップにわたる未来の出力値 y_M を時刻 t から M ステップにわたる入力値 Δu_n から求める形式になっている」

4.2 出力予測式

熊さん「モデルがわかったところで, どのように出力を予測するんですか?」

ご隠居「もしモデルがプラントを完全に表現していれば, プラントの実際の出力値 $y(t+j)$ は, モデルで計算した値 $y_M(t+j)$ に一致するはずである. しかし, モデルがプラントを完全に表現できることは現実にはあり得ない. プラントには測定できない外乱が入ったり, モデル誤差により, 現実のプラントの出力値とモデルの出力値とに食い違いが生じるのは明らかである. このようなモデルとプラントのずれや, プラントに加わる外乱の影響を考慮して, 時刻 $t+j$ の出力の予測値を, つぎのように与える.

$$y_p(t+j) = y_M(t+j) + \{y(t) - y_M(t)\} \quad (30)$$

$(j = L, \dots, L+P-1)$

この予測式において, モデルとプラントのずれを考慮して, モデルの出力を補正している項が $y(t) - y_M(t)$ である. これはモデルの時刻 t の値 $y_M(t)$ と実際の出力の測定値 $y(t)$ との差を現時刻 t での外乱の値とみなし, それと同じ大きさの外乱が予測期間中プラントに入り続けると想定し, 予測値を与えていると解釈できる (Fig.4)

熊さん「(30)式で t 時刻のずれ $\{y(t) - y_M(t)\}$ を測定不能な外乱 $d(t)$ とみなしているわけですね?」

ご隠居「その通り. また, 予測式を

$$y_p(t+j) = y(t) + y_M(t+j) - y_M(t) \quad (31)$$

のように変形して眺めると, 測定値 $y(t)$ を始点として, モデルを使って現時刻からの出力の変化量 $y_M(t+j) - y_M(t)$ を計算して出力を予測している

とも解釈できる. (Fig.5)

出力・目標値一致希望区間 $[L, L+P-1]$ での出力の予測値を (31) 式を使って表現すると

$$\begin{bmatrix} y_p(t+L) \\ y_p(t+L+1) \\ \vdots \\ y_p(t+L+P-1) \end{bmatrix} = \begin{bmatrix} y(t) \\ y(t) \\ \vdots \\ y(t) \end{bmatrix} + \begin{bmatrix} y_M(t+L) \\ y_M(t+L+1) \\ \vdots \\ y_M(t+L+P-1) \end{bmatrix} - \begin{bmatrix} y_M(t) \\ y_M(t) \\ \vdots \\ y_M(t) \end{bmatrix} \quad (32)$$

これをベクトルで表現すると

$$y_P = y + y_M - y_{M0} \quad (33)$$

のようになるのじゃ」

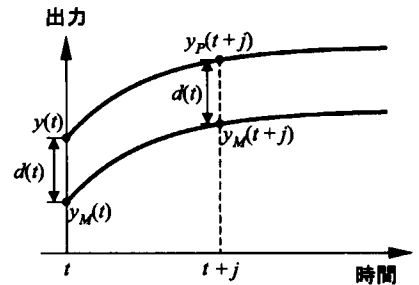


Fig.4 外乱の予測(その1)

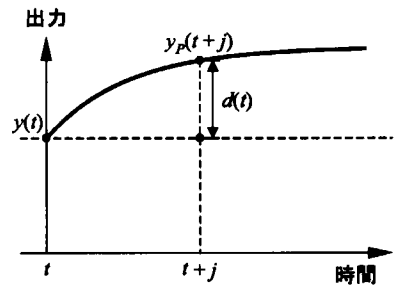


Fig.5 外乱の予測(その2)

4.3 参照軌道

熊さん「これで, 予測できるわけですから, あとは操作量を決めればいいんですね」

ご隠居「ここで予測される動きができるだけ希望とする動きになるように操作量を定めるわけだが, 時刻 $t+j$ において出力の予測値を一致させようとする値を $y_R(t+j)$ とする. 設定値 r を直接, その値とすることも可能である. しかし, 出力を一気に設定値 r にもっていこうとするのではなく, ある滑らかな軌道に沿って最終的に設定値 r に達するように制御しようとする. この軌道を参照軌道 (reference trajectory) とよぶ.

参照軌道のつくり方として種々提案されている¹⁾. その中から代表的な二つについて述べよう.

・タイプ1

$$y_R(t+j) = \alpha^{j-L+1}y(t) + (1-\alpha^{j-L+1})r(t+j) \quad (34)$$

・タイプ2

$$y_R(t+j) = \alpha^{j-L+1}y_R^*(t+L-1) + (1-\alpha^{j-L+1})r(t+j) \quad (35)$$

タイプ1の参照軌道は、時刻 $t+j$ の設定値 $r(t+j)$ と現時刻 t の測定値 $y(t)$ を $\alpha^{j-L+1}:1-\alpha^{j-L+1}$ に内分したところに軌道を設定している。これはFig.6に示すように、出力を直接、最終設定値 r にもっていかうとするのではなく、現時刻 r の測定値から1次おくれのステップ応答軌道のような理想的な動きとして制御系に与え、この軌道に沿って最終設定値 r に出力が辿りつくように制御することを目的に導入されている。

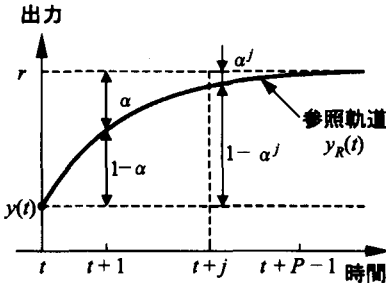


Fig.6 参照軌道(タイプ1)

タイプ2の参照軌道は、むだ時間や非最小位相系など、いまから決める入力に呼応して出力がすぐに設定値の方向に動かない対象に使われる。たとえば、むだ時間が T_d の系では、 $L = T_d + 1$ とすることにより、むだ時間の間は出力の目標となる軌道は設定しないことになる。具体的には、いまから決める操作量 $u(t+i)$ (ただし $i = 0, 1, \dots, L+P-2$)をすべて操作量 $u(t)$ に保ち続けると仮定したうえで、時刻 $t+L-1$ での出力の予測値 $y_p^*(t+L-1)$ を計算し、その値とそれ以降の時刻の設定値 $r(t+j)$ とを $\alpha^{j-L+1}:1-\alpha^{j-L+1}$ に内分して軌道をつくっている。具体的に、(35)式の $y_p^*(t+L-1)$ の項は、ステップ応答モデル式(9)式を使うならば、つぎのように計算できる。(Fig.7)

$$y_p^*(t+L-1) = y(t) + a_{L-1}\Delta u(t) + \sum_{i=1}^{L-1} (a_{L-1+i} - a_i)\Delta u(t-i) \quad (36)$$

操作量の決定の仕方を簡潔に表現するために、参照軌道として与えられる目標値もベクトルで表現しておく。

・タイプ1

$$y_R = \tilde{\alpha}fy + (I - \tilde{\alpha})r \quad (37)$$

ここに

$$y_R = [y_R(t+L), \dots, y_R(t+L+P-1)]^t$$

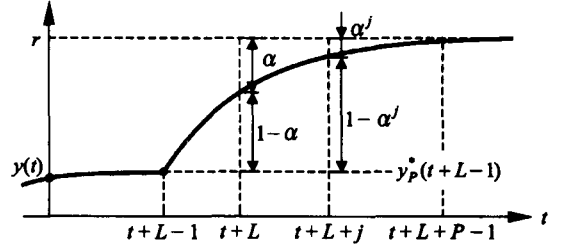


Fig.7 参照軌道(タイプ2)

$$\tilde{\alpha} = \text{diag}(\alpha, \alpha^2, \dots, \alpha^P)$$

$$r = [r(t+L), \dots, r(t+L+P-1)]^t$$

$$f = [1, 1, \dots, 1]^t$$

である。具体的に書けば

$$\begin{bmatrix} y_R(t+L) \\ y_R(t+L+1) \\ \vdots \\ y_R(t+L+P-1) \end{bmatrix} = \begin{bmatrix} \alpha & 0 & \dots & 0 \\ 0 & \alpha^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \alpha^P \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} y(t) + \begin{bmatrix} 1-\alpha & 0 & \dots & 0 \\ 0 & 1-\alpha^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1-\alpha^P \end{bmatrix} \begin{bmatrix} r(t+L) \\ r(t+L+1) \\ \vdots \\ r(t+L+P-1) \end{bmatrix} \quad (37)$$

となる。

・タイプ2

$$y_R = \tilde{\alpha}f(y(t)a_{L-1}\Delta u(t) + a_0^t\Delta u_0) + (I - \tilde{\alpha})r \quad (38)$$

ここに、

$$a_0^t = [a_L - a_1, a_{L+1} - a_2, \dots, a_{L+s-2} - a_{s-1}]$$

である。具体的に書けば

$$\begin{bmatrix} y_R(t+L) \\ y_R(t+L+1) \\ \vdots \\ y_R(t+L+P-1) \end{bmatrix} = \begin{bmatrix} \alpha & 0 & \dots & 0 \\ 0 & \alpha^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \alpha^P \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} (y(t) + a_{L-1}\Delta u(t) + [a_L - a_1, a_{L+1} - a_2, \dots, a_{L+s-2} - a_{s-1}] \begin{bmatrix} \Delta u(t-1) \\ \Delta u(t-2) \\ \vdots \\ \Delta u(t-s+1) \end{bmatrix}) + \begin{bmatrix} 1-\alpha & 0 & \dots & 0 \\ 0 & 1-\alpha^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1-\alpha^P \end{bmatrix} \begin{bmatrix} r(t+L) \\ r(t+L+1) \\ \vdots \\ r(t+L+P-1) \end{bmatrix} \quad (38)$$

となる」

熊さん「 α にどんな意味があるのですか?」
ご隠居「 α をゼロに近く設定することは、目標値 y_R をより設定値 r に近いところにおくことにより、速

やかに設定値にもっていかうとしている。αを1より少しずつ小さくしていくことによって、安定な範囲でより速応性の高い制御系が得られることを意味しておるのじゃよ]

4.4 操作量の決定

ご隠居「入力ならびに出力になんら制約がない場合は、モデル予測制御のアルゴリズムは、時刻 $t+L$ から P ステップにわたる出力の予測値がその区間 $[t+L, \dots, t+L+P-1]$ においてできるだけ近づくように、現時刻 t 以降 M ステップの操作量を定める。すなわち、新たな測定値が得られるサンプル時刻ごとに、つぎの最適化問題を解いている。

いま、時刻 t での測定値 $y(t)$ が得られたとしよう。その時点で定式化される最適化問題はつぎのようになる。

評価関数

$$(y_R - y_P)^t \Lambda (y_R - y_P) + \Delta u_n^t \Psi \Delta u_n \quad (39)$$

予測式

$$y_P = y + y_M - y_{M0} \quad (40)$$

ここで、 Λ , Ψ は重み係数、 ϕ が入った対角化行列である。この解は、最小二乗法より次式のように与えられる。

・ステップ応答モデルを使った場合

$$\Delta u_n = (A_F^t \Lambda A_F + \Psi)^{-1} A_F^t \Lambda (y_R - y - A_0 \Delta u_0) \quad (41)$$

・パラメトリックモデルを使った場合

$$\Delta u_n = (G_F^t \Lambda G_F + \Psi)^{-1} G_F^t \Lambda (y_R - y - G_0 \Delta u_0) - Q_0 \Delta y_{M0} \quad (42)$$

計算された Δu_n のうち、現時刻の操作量 $\Delta u(t)$ だけを実際のプロセスに加え、つぎのサンプル時刻をまつ。つぎのサンプル時刻では、新たに出力の測定値 $y(t+1)$ を得て、そのサンプル時刻 $t+1$ を現時刻 t として上述の最適化問題を再び定式化し直して解く。この手続きをサンプル時刻ごとに繰り返す]

熊さん「(39)式の第1項は予測値と参照軌道の近さを表わす指標ですが、第2項はなんですかい？」

ご隠居「操作量の変化に対してペナルティー(罰則)を加えているのじゃよ。Ψを大きくすれば、速応性を犠牲にして安定性を向上させることになる」

熊さん「それにしても、各サンプル時刻ごとに(41)式なり、(42)式で操作量を定めるのはしんどいですな？」

ご隠居「その心配はない。入出力の重み係数を固定してしまえば、その逆行列の計算は1回ぼっきりで、毎回計算する必要はないのじゃよ」

八つあん「最小二乗法というのは例のご隠居が書いた文献3)にてている公式ですね?知ってますよ」

ご隠居「そうだ。最小二乗法は Gauss が考えたらしいが、制御のみならずいろいろなところに使われるのじゃ」

熊さん「モデルにはステップ応答モデル、パラメトリックモデルの二つを教えてくださいましたが、状態方程式でも非線形モデルでも利用できるんですかい？」

ご隠居「入力の関数として出力が計算できるものであればどのような形式でもよい。ただし、非線形モデルでは最小二乗法を利用して操作量を定めることはできない」

八つあん「参照軌道と予測値を一致させようとする区間 $[L, L+P-1]$ を現時刻 t より遠くにとることは、制御入力を時間をかけて設定値に一致させることに対応し、操作量の動きもゆっくりになりますね？」

ご隠居「その通り。ホライズンの始まり L を遠くにとることはゆっくりとした応答を期待していることになるんだ」

熊さん「ホライズンの幅 P と M の決め方に何かルールはないんですかい？」

ご隠居「通常 $P \geq M$ に設定され、その差が安定性に影響を与えることになる。PをMに対して大きくとればとるほど、ゆっくりとした応答を期待していることになる」

八つあん「むだ時間について説明がなかったように思いますが？」

ご隠居「むだ時間が T_d であるとき、少なくとも $[L, L+P-1]$ の区間が T_d を含むか、 $L \geq T_d$ を満たすように決めなければならない」

5. 他の制御系との構造比較

5.1 PID 制御との構造比較

ご隠居「まず、 $P = M = L = 1$ とする。これを一段予測のモデル予測制御とよんでいる。操作量は時刻 $t+1$ の目標値と出力の予測値が一致するように決められる。つまり、

モデルによる出力の予測値：

$$\left. \begin{aligned} y_P(t+1) &= y_M(t+1) - y_M(t) + y(t) \\ \text{参照軌道(タイプ1)による目標値:} \\ y_P(t+1) &= \alpha y(t) + (1-\alpha)r(t+1) \\ \text{予測値が目標値に一致:} \\ y_P(t+1) &= y_R(t+1) \end{aligned} \right\} \quad (43)$$

プラントの入出力関係は

$$y(t) = z^{-1} \frac{N(z)}{D(z)} u(t) \quad (44)$$

で与えられ、プラントのモデルは(8)式より

$$y_M(t) = \sum_{i=1}^L a_i \Delta u(t-i) + a_0 u(t-s)$$

$a_{i+1} - a_i = h_i$ を用いて表わすと,

$$y_M(t) = z^{-1} \tilde{N}(z) u(t) \quad (45)$$

ここに $\tilde{N}(z) = h_1 + h_2 z^{-1} + \dots + h_n z^{-n+1}$, と表わせる.
これをインパルス応答モデルという.

(43)~(45)式を用いて, $u(t)$ を求める.

$$y_M(t+1) - y_M(t) + y(t) = \alpha y(t) + (1-\alpha)r(t+1) \quad (46)$$

を变形すると,

$$y_M(t+1) - y_M(t) + (1+\alpha)y(t) = (1-\alpha)zr(t+1)$$

となり, $y_M(t+1) - y_M(t) = (z-1)y_M(t)$ であるから

$$(1-z^{-1})\tilde{N}(z)u(t) + (1-\alpha)z^{-1}\frac{N(z)}{D(z)}u(t) = (1-\alpha)zr(t)$$

となる. よって,

$$u(t) = \frac{(1-\alpha)z}{(1-z^{-1})\tilde{N}(z) + (1-\alpha)z^{-1}\frac{N(z)}{D(z)}} r(t) \quad (47)$$

と書ける. 偏差

$$e(t) = r(t) - y(t)$$

を導入すると, (46)式を变形して

$$y_M(t+1) - y_M(t) = -(1-\alpha)y(t) + (1-\alpha)r(t+1)$$

右辺に $(1-\alpha)r(t)$ を加えて引くと

$$y_M(t+1) - y_M(t) = (1-\alpha)\{r(t+1) - r(t)\} + (1-\alpha)\{r(t) - y(t)\}$$

となり,

$$(1-z^{-1})y_M(t+1) = (1-\alpha)z\Delta r(t) + (1-\alpha)e(t)$$

ここに, $\Delta r(t) = r(t) - r(t-1)$ である.

(45)式を用いると,

$$\dot{u}(t) = \frac{(1-\alpha)}{(1-z^{-1})\tilde{N}(z)} e(t) + \frac{(1-\alpha)z}{(1-z^{-1})\tilde{N}(z)} \Delta r(t) \quad (48)$$

と表わせる. $1/(1-z^{-1})$ が積分器に相当することから, モデル予測制御では, 積分器を1つもっていることがわかる.

もし, モデルが1次おくれ系(時定数 T , ゲイン定数 K , サンプル周期 ΔT)

$$z^{-1}\tilde{N}(z) = \frac{qz^{-1}}{1-pz^{-1}} \quad (49)$$

ここに, $p = e^{-(\Delta T/T)}$, $q = K(1 - e^{-(\Delta T/T)})$ であったとすると, $e(t)$ から $u(t)$ への伝達関数 $G_C(z)$ は (48)式より

$$G_C(z) = \frac{(1-\alpha)}{(1-z^{-1})\frac{q}{1-pz^{-1}}} = \frac{p(1-\alpha)}{q} \left\{ 1 + \frac{(1-p)}{p(1-z^{-1})} \right\}$$

$$= K_p \left\{ 1 + \frac{1}{T_I(1-z^{-1})} \right\} \quad (50)$$

ここに,

$$K_p = \frac{p(1-\alpha)}{q} \left. \vphantom{K_p} \right\} \quad (51)$$

$$T_I = \frac{p}{1-p} \left. \vphantom{T_I} \right\}$$

となり, PI コントローラの構造になっている. これはモデル予測制御の観点から PI 制御系のパラメータの調整法が提案できることを示唆している」

5.2 モデルフィードバック

ご隠居「つぎに, 一段予測のモデル予測制御の制御構造を見ることにする. (46)式より

$$y_M(t+1) - y_M(t) + y(t) = \alpha y(t) + (1-\alpha)r(t+1) \quad (46)$$

(45)式より

$$y_M(t) = z^{-1}\tilde{N}(z)u(t) \quad (45)$$

(45)式を(46)式に代入し, まとめると,

$$\tilde{N}(z)u(t) = \alpha y(t) + (1-\alpha)r(t+1) + y_M(t) - y(t)$$

となり, 両辺に $-\alpha y_M(t) = -\alpha z^{-1}\tilde{N}(z)u(t)$ を加えると

$$(1-\alpha z^{-1})\tilde{N}(z)u(t) = (1-\alpha)\{zr(t) - y(t) + y_M(t)\}$$

となり, よって

$$u(t) = \frac{(1-\alpha)z^{-1}}{(1-\alpha z^{-1})z^{-1}\tilde{N}(z)} \{zr(t) - y(t) + y_M(t)\} = G_C(z) \{zr(t) - y(t) + z^{-1}\tilde{N}(z)u(t)\} \quad (52)$$

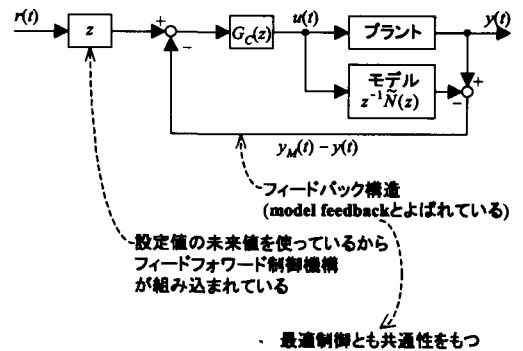


Fig.8 Model Feedback

ここに,

$$G_C(s) = \frac{(1-\alpha)z^{-1}}{(1-\alpha z^{-1})z^{-1}\tilde{N}(z)} \quad (53)$$

である. (52)式を忠実にブロック線図に画くと Fig.8 のようになる. これは前稿の内部モデル制御構造になっている」

5.3 内部モデル制御

ご隠居「おくれのないプラントに対しては $L = M = P = 1$ として, タイプ1の参照軌道をとった場合のモデル予測制御の構造が, Fig.8で与えられることを学んだ. また, おくれ系 ($z^{-(T_d+1)}\tilde{N}(z)$) に関して, $L = T_d+1, M = P = 1$ とし, タイプ2の参照軌道を使うと, 同じようなブロック線図に到達することを示そう.

$L = T_d+1, P = M = 1$ の場合, 操作量 $u(t)$ は時刻 $t+T_d+1$ の目標値と出力の予測値が一致するように決められる. つまり,

$$\begin{aligned} y_P(t+T_d+1) &= y_M(t+T_d+1) - y_M(t) + y(t) \\ &= \alpha y_P^*(t+T_d) + (1-\alpha)r(t+T_d+1) \\ &= y_R(t+T_d+1) \end{aligned} \quad (54)$$

プラントのモデルが

$$y_M(t) = z^{-(T_d+1)}\tilde{N}(z)u(t) \quad (55)$$

で与えられることから, $z^{(T_d+1)}$ を作用させると,

$$y_M(t+T_d+1) = \tilde{N}(z)u(t) \quad (56)$$

となる. (56)式を(54)式に代入し,

$$\begin{aligned} \tilde{N}(z)u(t) &= \alpha y_P^*(t+T_d) + (1-\alpha)r(t+T_d+1) \\ &\quad + y_M(t) - y(t) \end{aligned} \quad (57)$$

となる. 参照軌道の $y_P^*(t+T_d)$ は(36)式より

$$y_P^*(t+T_d) = y_M(t+T_d) + y(t) - y_M(t) \quad (58)$$

である. (58)式を(57)式に代入すると,

$$\begin{aligned} \tilde{N}(z)u(t) &= \alpha y_P^*(t+T_d) + (1-\alpha)r(t+T_d+1) \\ &\quad + (1-\alpha)y_M(t) - (1-\alpha)y(t) \end{aligned} \quad (59)$$

ここに $y_M(t+T_d) = z^{-1}\tilde{N}(z)u(t)$ であるから, (59)式は

$$\begin{aligned} (1-\alpha z^{-1})\tilde{N}(z)u(t) &= (1-\alpha)\{r(t+T_d+1) \\ &\quad + y_M(t) - y(t)\} \end{aligned} \quad (60)$$

と書ける. したがって,

$$u(t) = G_C(z)\{z^{T_d+1}r(t) + z^{-(T_d+1)}\tilde{N}(z)u(t) - y(t)\}$$

と書ける. ただし,

$$G_C(z) = \frac{(1-\alpha)z^{-1}}{(1-\alpha z^{-1})z^{-1}\tilde{N}(z)} \quad (61)$$

である. (60)式を忠実にブロック線図に画くと,

Fig.9 のようになる. これは前稿でやった内部モデル制御そのものの構造になっている」

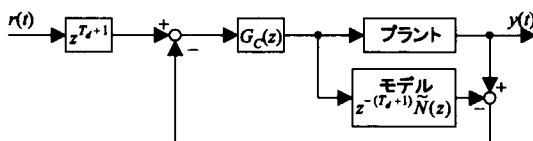


Fig.9 IMC

熊さん「なんだか, モデル予測制御というのは奥が

深そうですね?」

ご隠居「あたり前だよ. だから1980年代から今にかけてこれだけの評価を得ているのだよ. 状態フィードバックや最適予見制御との関連は文献5)に詳しいから, 勉強してくれたまえ」

八つあん「ご隠居. ひとつ胸にひっかかる問題があるんですがね. モデル予測制御では, 目標値への追従性だけを問題にして, 外乱を抑制するということはまったく考えていませんね. これはプロセス制御の本来のあり方として邪道ではありませんか?」

ご隠居「確かに, 明確にわかっている目標値に対しては定式化しているが, 未知の外乱に対してはまったく対処していないのは事実である. しかし, 外乱をカルマンフィルタで予測したり, センサのあとに緩和フィルタを挿入して外乱に対処している方法もあるが...」

参考文献

- 1) 大嶋正裕: プロセスと制御システム, コロナ社, 152/172 (2003)
- 2) 橋本伊織, 長谷部伸治, 加納学: プロセス制御工学, 朝倉書店, 132/149 (2002)
- 3) 伊澤, 上村, 黒須ほか: 仕事に役立つ微分・積分, パワー社, 194/208 (2001)
- 4) 大嶋正裕, 橋本伊織: モデル予測制御(基礎コース), モデル予測制御チュートリアル資料 (1990)
- 5) 大嶋正裕, 橋本伊織: モデル予測制御(アドバンスドコース), 現代制御理論と応用講座 (1990)

「受理年月日 2003年9月30日」