

# DDC シ リ ー ズ (その 1)

## DDC 演算機構のフローチャート作成のための一手法

### DDC Series (Part 1)

#### A Method of Calculating Direct Digital Control System

杉 浦 至\*

Itaru Sugiura

DDC (Direct Digital Control) System は従来のアナログ調節計のかわりに、設備費、運転費などのコストダウンをねらって、英国で開発されたのに端を発する。この System は、従来のコンベンショナルコントローラの組合せに比べて、最適制御、多層制御、多重制御などの制御系の構成がハード的にも、ソフト的にも容易である。

当社もこのシステムについて、そのソフトウェア、ハードウェア、アプリケーションウェアなどの開発に努力しているが、このシリーズでは、それらについて紹介する。

## I. ま え が き

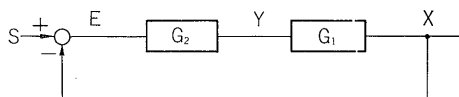
DDC (Direct Digital Control) では、対象とする制御量が、多変数関数のプロセスであるため、コントロール・ループも、単一のフィードバック・ループや、フィードフォワード・ループなどではなく、それらが何段も結合されたマルチプル・コントロールとなる。

またコントロール・アクションも、従来のアナログ調節器が行なってきた、P, PI, PD, PID などの演算をデジタル量で行なうことはもちろんとして、さらに一歩進んで、演算のアルゴリズムと演算結果をプロセスに伝える出力装置が開発されさえすればプロセスの最適制御演算までも行なうことができる。

ここでは、それらのもとになる演算の方式と、そのフローチャート化について検討する。

## II. シングル・ループの場合

コントロール・ループが一段の、プラントの制御系としては最も簡単な形 (第 1 図) を考える。



第 1 図 フィードバック系  
Fig. 1. Feed back control system

注) 記号について

- S: 設定値      E: 偏差  
Y: 操作量  $\left\{ \begin{array}{l} P: \text{位置形のときの操作量} \\ V: \text{速度形のときの操作量} \end{array} \right.$   
X: 制御量  
 $G_1$ : プロセスの伝達関数  
 $G_2$ : コントローラの伝達関数  
K: コントローラの比例定数

$T_I$ : コントローラの積分定数

$T_D$ : コントローラの微分定数

$K_P$ : プロセスの比例定数

$T_0$ : 一時遅れ時定数

L: むだ時間      s: ラプラス変換

大文字: s 領域での値      小文字: 時間領域での値  
コントローラ部分に計算機を使うと、プロセスからは一般には伝達関数が一次遅れとむだ時間をもった

$$G_1(s) = \frac{K_P}{1 + T_0 s} \cdot e^{-Ls} \quad \dots\dots\dots(1)$$

で規定された出力がアナログ量でてくる。

コンピュータは、これをあるサンプリング時間ごとにスキャンし、P, PI, PD, PID などの演算を行ない、アウトプット・デバイスに伝え設定値 S に制御量 X を追従させて、偏差を打ち消すようにコントロール・アクションを行なう。

代表的に  $G_2$  として、PID アクションをとると、(P, PI, PD の場合は、それぞれ  $T_I = \infty$ ,  $T_D = 0$ ;  $T_D = 0$ ;  $T_I = \infty$  と考えればよいから)

$$G_2(s) = K \left( 1 + \frac{1}{T_I s} + T_D s \right) \dots\dots\dots(2)$$

時間領域では、式(2)は

$$y(t) = p(t) = K \left( e + \frac{1}{T_I} \int_0^t e dt + T_D \frac{de}{dt} \right) \dots\dots(3)$$

(操作量として位置形のものとき)

$$y(t) = v(t) = \frac{dp(t)}{dt} = K \left( \frac{de}{dt} + \frac{1}{T_I} e + T_D \frac{d^2e}{dt^2} \right) \dots\dots\dots(4)$$

(速度形のものとき)

サンプリング時間は、主に AD 変換するマルチプレクサーの性能によって決まるが、この値がプロセスの時定数に比べて充分小さいと、あるサンプリング時刻、 $t = t_n$  での値は式(3), (4)をそのまま階差式表示して、

$$p_n = K \left( e_n + \frac{1}{T_I} \sum e_n \cdot \Delta t + T_D \cdot \frac{e_n - e_{n-1}}{\Delta t} \right) \dots(5)$$

$$v_n = K \left( \frac{e_n - e_{n-1}}{\Delta t} + \frac{1}{T_I} \cdot e_n + T_D \cdot \frac{e_n - 2e_{n-1} + e_{n-2}}{\Delta^2 t} \right) \dots\dots\dots(6)$$

となる. ここで  $\Delta t$  は数学的には  $t$  の微小量だが, ここではサンプリング間隔を表わす.

偏差  $e_n$  の変動が激しく, Digital Filter をつけたいときは,  $G_2$  の前に, たとえば二次遅れの  $G_3$  をつける (第 2 図).

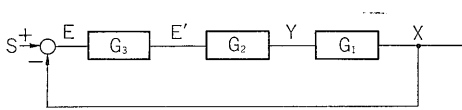
$$G_3(s) = \frac{E'(s)}{E(s)} = \frac{1}{1 + T_1 s + T_2 s^2} \dots\dots\dots(7)$$

これは,  $t$  領域では,

$$e' + T_1 \frac{de'}{dt} + T_2 \frac{d^2 e'}{dt^2} = e \dots\dots\dots(8)$$

$$e'_n + T_1 \frac{e'_n - e'_{n-1}}{\Delta t} + T_2 \frac{e'_n - 2e'_{n-1} + e'_{n-2}}{\Delta^2 t} = e_n \dots\dots\dots(9)$$

式 (9) より,  $e'_n$  は, 1 サンプリング時刻前の値  $e'_{n-1}$  と, 2 サンプリング時刻前の値  $e'_{n-2}$  およびそのときのエラー  $e_n$  より求まる.



第 2 図 フィルタをつけたフィードバック系  
Fig.2. Feed back control system with a filter

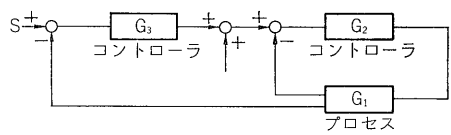
さらにエラー  $e_n$  の大きさによっては, 非線形要素をそう入して, 系の域衰度をかえて早い応答の, しかもオーバーシュートの少ないシステムを作ったりすることも可能である.

### III. 多段ループの場合

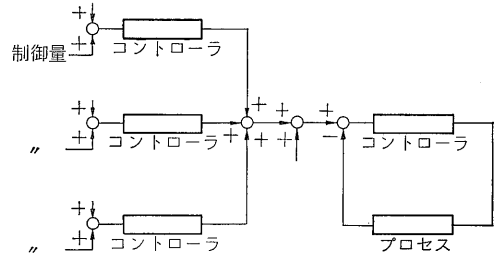
プロセスが複雑になり, ブロックダイアグラムも第 1 図のような簡単なものから, もっと進んだ, カスケードコントロール, パラレルコントロール, シリーズコントロール (第 3, 4, 5 図) などの, マルティプルループの形態をとっても, II. で述べたシングルループの延長だから第 6 図のように, ブロックダイアグラムを各セクションに分割し, それぞれのセクションで伝達関数を作り, 入力と出力の関係式を求めて行けば, ブロックダイアグラムから, 計算機のフローチャートを作ることは容易に行なえる.

すなわち

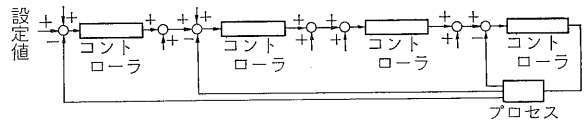
- セクション : 1 では,  $O_1 = I_1 - O_{10}$
- セクション : 2 では,  $O_2 = I_1 \cdot K \left( 1 + \frac{1}{T_I s} + T_D s \right)$
- セクション : 3 では,  $O_3 = I_3 - O_7$
- セクション : 4 では,  $O_4 = I_4 \cdot K \left( 1 + \frac{1}{T_I s} + T_D s \right)$
- セクション : 5 では,  $O_5 = I_5 + O_{u1}$



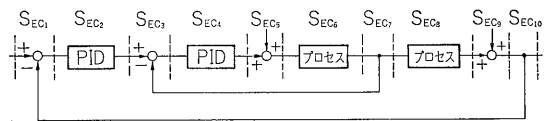
第 3 図 カスケードコントロール  
Fig.3. Cascade control system



第 4 図 パラレルコントロール  
Fig.4. Parallel control system



第 5 図 シリーズコントロール  
Fig.5. Series control system



第 6 図 カスケードコントロール  
Fig.6. Cascade control system

セクション : 6 では,  $O_6 = I_6 \cdot K \left( 1 + \frac{1}{T_I s} + T_D s \right)$

セクション : 7 では,  $O_7 = I_7$

セクション : 8 では,  $O_8$  はプロセスより, スキャンして Memory に入れておく.

セクション : 9 では,  $O_9 = I_9 + O_{u2}$

セクション : 10 では,  $O_{10} = I_{10}$

注)  $O_n$  : セクション  $n$  の出力

$I_n$  : セクション  $n$  の入力

$O_{u1}$  : セクション 5 に入る外乱

$O_{u2}$  : セクション 9 に入る外乱

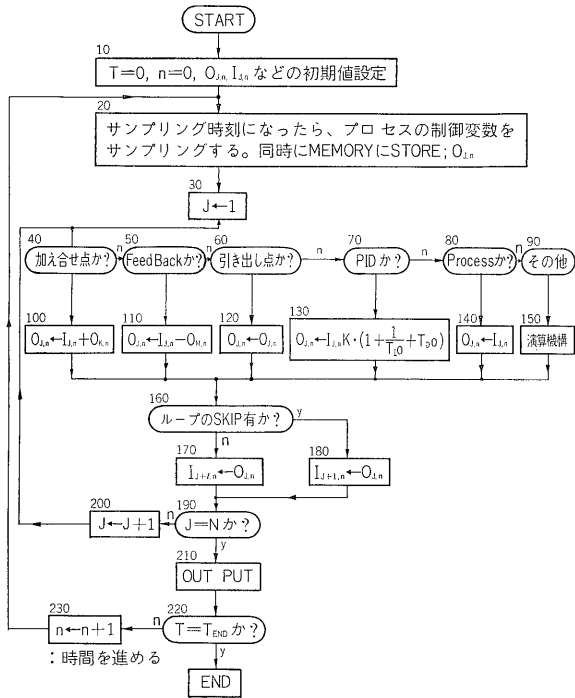
そして, あるセクションの入力は, その前のセクションの出力となっている.

これより, 第 7 図のフローチャートができる.

### IV. フローチャートの説明

ブロック : 10, 時間 = 0, 微分演算や, 初期設定値のイニシャルセットを行なう.

: 20, プロセスの出力をサンプルしてメモリーに入れておく.



第7図 演算フローチャート  
Fig. 7. A flow chart for computation

: 30, 最初のセクションを設定する。  
: 40~90, そのセクションの状態によって、演算は分岐する。  
また、たとえばセクション: 1のフィードバックは、何番目のセクションからのもどりであるかは、ブロックダイアグラム作成時で分るから、これらの値もあらかじめ、イニシャルセットで読ませておく。  
また、ブロックダイアグラムにより、非線形要素とか、フィードフォワードの要素があれば、90のその他の項につき足していく。

: 130, I. で述べた式 (5) あるいは(6)にしたがって計算する。  
: 160, ブロックダイアグラムの中のあるループは、飛ばしても差しつかえないというときはそこをスキップする。  
: 170, 180, 次段の入力を出す。  
: 190, 全セクション完了か?  
: 210, アウトプット機構。  
: 220, もう計算しなくてよいか? この値はプロセスにオンラインのときは、きわめて大きな値。  
: 230, 次の時間に進める。

V. む す び

制御系が第1図のように簡単な場合には、別にセクションに分けなくとも、設定値、偏差、制御量などの関係式はすぐわかる (付録1参. ここでは、プロセス部分もデジタルに、シミュレートした)。

しかし、ブロックダイアグラムごとに、毎回、関係式を出していたのではきわめて面倒である。

ここではブロックダイアグラムの状態を入力データとして、与えるだけで計算結果を出す一つの方式を述べた。

これは、別にDDCだけでなく、広く一般の制御系の解析にも使える。

- すなわち、フィードバック・コントロール
- フィード・フォワード・コントロール
- オンオフ・コントロール
- プログラム・コントロール
- レシオ・コントロール
- マルチプル・コントロール
- ノンリニア・コントロール

などの演算は、自由にできる。

付 録 Appendix

DDCのデジタル、シミュレーションについて

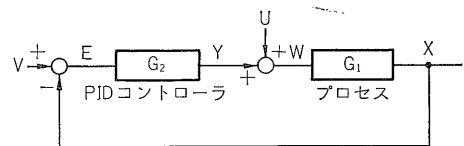
1. ま え が き

いまプロセスをむだ時間のある一次遅れの式で近似し、これをPIDコントローラで調節する系を対象にする (第1図)。

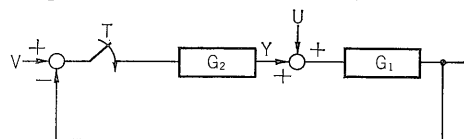
$$G_1(s) = \frac{K_P}{1 + T_I s} e^{-Ls} \dots\dots\dots(1)$$

$$G_2(s) = K \left( 1 + \frac{1}{T_I s} + T_D s \right) \dots\dots\dots(2)$$

PID を、デジタルコンピュータでおきかえると、コントロールは連続的ではなくて、時間  $T$  でサンプルするサンプリングコントロールになる (第2図)。



第1図 PID連続制御系  
Fig.1. PID continuous control system



第2図 サンプリング制御系  
Fig.2 Sampling control system

サンプル値は、高周波成分を含むから制御対象に入る前にこれをフィルタするために 0 次ホールド回路を通すが、ここでは簡単化のためそれは省く。

ここでは、第 1 図および第 2 図のような系のシミュレーションを行なう。

2. 解 析

まず、第 1 図のような、連続制御の場合は、

制御偏差は、

$$e(t) = v(t) - x(t) \dots\dots\dots(3)$$

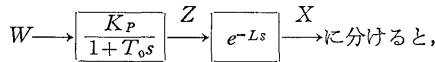
操作量は、

$$y(t) = K \cdot e(t) + \frac{K}{T_I} \int_0^t e(t) dt + K \cdot T_D \cdot \frac{de(t)}{dt} \dots\dots(4)$$

プロセス入力は、

$$w(t) = y(t) + u(t) \dots\dots\dots(5)$$

プロセスは



$$Z(t) + T_0 \frac{dZ(t)}{dt} = K_P w(t) \dots\dots\dots(6)$$

制御量は、

$$x(t) = Z(t-L) \dots\dots\dots(7)$$

式(4)は、さらに

$$y(t) = K \cdot e(t) + \frac{K}{T_I} (SUM + x(t) \cdot \Delta t) + K \cdot T_D \cdot \frac{e(t) - e(t-\Delta t)}{\Delta t} \dots\dots\dots(8)$$

式(6)も

$$\Delta Z(t) = \frac{1}{T_0} (K_P \cdot w(t) - Z(t)) \dots\dots\dots(9)$$

とし、時間のピッチ  $\Delta t$  を進めるごとに

$$Z(t) \leftarrow Z(t) + \Delta Z(t) \dots\dots\dots(10)$$

$$SUM \leftarrow SUM + e(t) \cdot \Delta t \dots\dots\dots(11)$$

とする。このフローチャートを第 3 図に示す。

次に、いま第 2 図を外乱  $U=0$  の追値制御にして考えるとブロックダイアグラムは、第 4 図ようになる。ここで

$G_2$  は  $Z$  変換したものを考えると

$$G_2 = \frac{a_0 + a_1 Z^{-1} + a_2 Z^{-2}}{1 + b_1 Z^{-1}} \dots\dots\dots(12)$$

ただし

$$a_0 = K \left( 1 + \frac{1}{T_I} + T_D \right)$$

$$a_1 = -K (1 + 2 T_D) \dots\dots\dots(13)$$

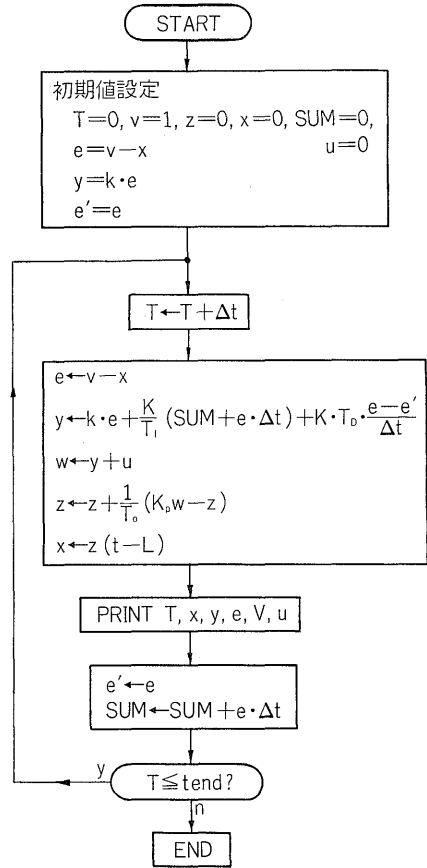
$$a_2 = T_D K$$

$$b_1 = -1$$

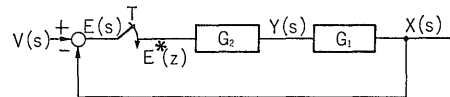
式(12)は、時間領域では、

$$y(t) = a_0 e(t) + a_1 e(t-T) + a_2 e(t-2T) - b_1 y(t-T) \dots\dots\dots(14)$$

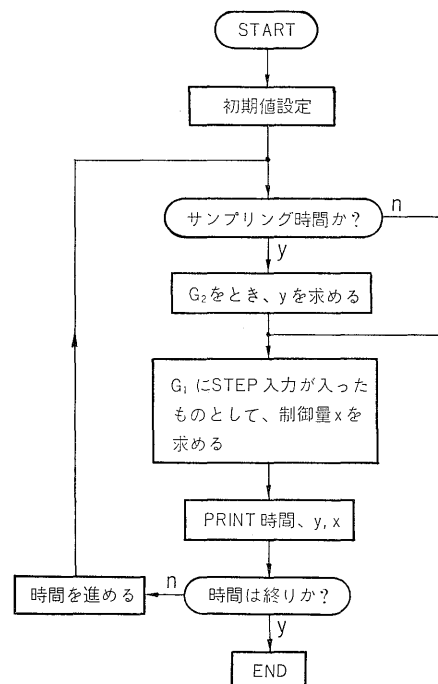
この  $y(t)$  は、あるサンプリング時間 ( $t=nT$ ) と、次のサンプリング時間 ( $t=(n+1) \cdot T$ ) の間は、コンスタントにプロセスに加わるから、 $G_1$  での応答は、この  $y(t)$  の  $G_1$  に対する、インディシャル応答として考える。 $n=n+2, n+3, \dots$  の



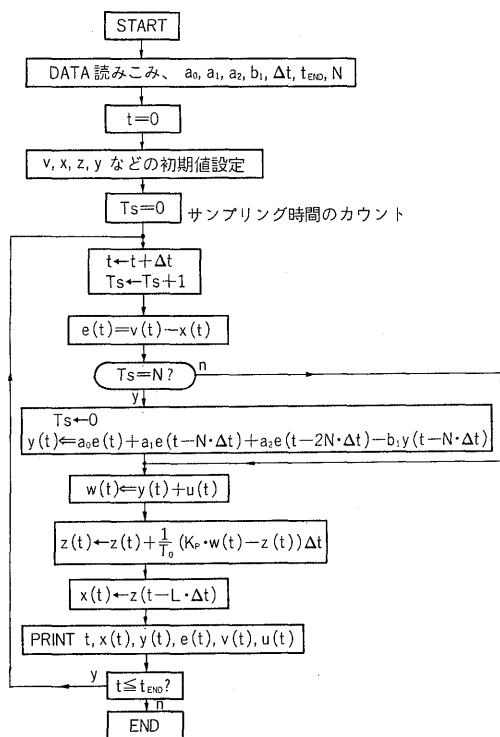
第 3 図 連続制御系演算フローチャート  
Fig.3. A flow chart for computation of continuous control system



第 4 図 サンプリングコントロール  
Fig.4. Sampling control system

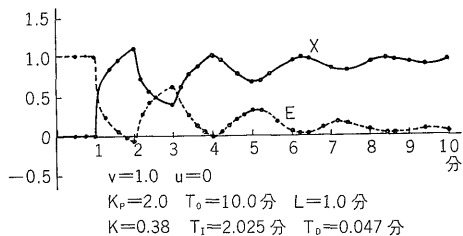


第 5 図 サンプリング制御の概略のフローチャート  
Fig.5. General flow chart for sampling control



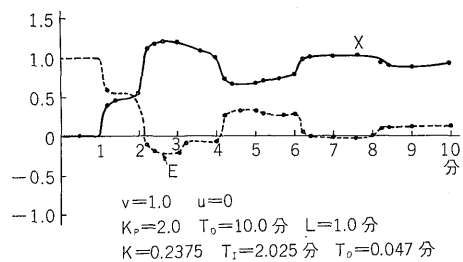
第6図 サンプルコントロール演算フローチャート

Fig.6 A flow chart for computation of a sampling control system



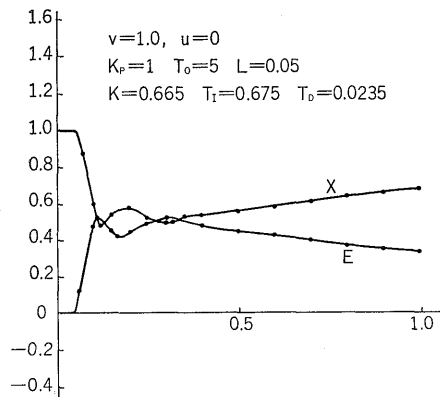
第7図 連続制御

Fig.7. Continuous control



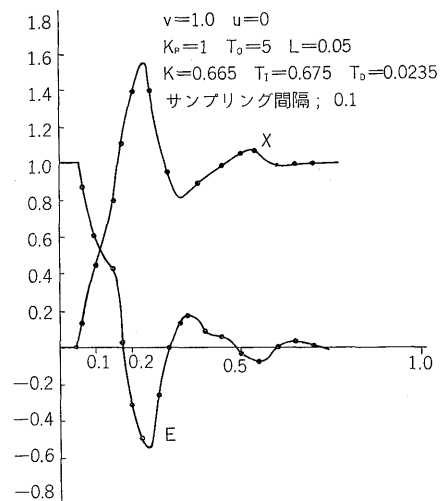
第8図 サンプルコントロール

Fig.8. Sampling control



第9図 連続制御

Fig.9. Continuous control



第10図 サンプルコントロール

Fig.10. Sampling control

場合も、まったく同様に計算する。

この計算のジェネラルフローチャートを第5図に、さらに詳しいフローチャートを第6図に示す。

なお  $v=1$  のステップ入力が入った場合の計算例を第7図、第9図 (連続制御の場合)、第8図、第10図 サンプルコントロールの場合に示す。第9図の場合は  $K, T_l, T_d$  の設定が不良のため立ち上りのゲインを押えすぎてしまっている。



\*本誌に記載されている会社名および製品名は、それぞれの会社が所有する  
商標または登録商標である場合があります。