4・3　一般入力に対する応答

プロセスのインパルス応答が求まると，この応答を基にして，任意の一般入力に対する応答が計算できる．いま，一般入力を $x(t)$ とすると，この関数は次のように近似的に表現することができる．

$$x(t) \approx x(0)\delta(t) + x(t_1)\delta(t-t_1) + x(t_2)\delta(t-t_2) + x(t_3)\delta(t-t_3) + \cdots \tag{4・14}$$

ここで，$t_1, t_2, t_3 \cdots$ は任意に選んだ時間である．そこで，この時間の選択個数を十分多くして各時間の間隔を小さくしていくと，任意の関数を精度よく近似的に表現することができる．

(4・14)式の入力に対する応答は，以下のようにして求まる．
まず，上式をラプラス変換すると次のようになる．

$$X(s) = x(0) + x(t_1)e^{-t_1 s} + x(t_2)e^{-t_2 s} + x(t_3)e^{-t_3 s} + \cdots \tag{4・15}$$

伝達関数 $G(s)$ のプロセスに，この入力が入ると，その出力 $Y(s)$ は次のようになる．

$$Y(s) = G(s)x(0) + G(s)x(t_1)e^{-t_1 s} + G(s)x(t_2)e^{-t_2 s} + G(s)x(t_3)e^{-t_3 s} + \cdots \tag{4・16}$$

そこで，上式を逆ラプラス変換*すると，次のような応答が得られる．

* (3・7)式から，$\mathcal{L}^{-1}\{G(s)e^{-\theta s}\} = g(t-\theta)$

$$y(t) = x(0)g(t) + x(t_1)g(t-t_1) + x(t_2)g(t-t_2) + x(t_3)g(t-t_3) + \cdots \tag{4・17}$$

ここで，$g(t)$ は伝達関数そのものを逆ラプラス変換したインパルス応答である．また，$g(t-t_i)$ はインパルス応答の時間を t_i だけ

ずらした応答を表す．したがって，あるプロセスのインパルス応答がひとたび求まると，この応答の時間をずらしながら，入力の大きさ $x(t_i)$ を重み係数として掛け合わせて，全ての応答を重ね合わせることによって，任意の入力に対する応答が，原理的には求まる．上記のような関係が成立するのは，プロセスが線型で，重ね合わせの原理が成立するためである．

上記の関係はラプラス変換のたたみこみ積分 (*convolution integral*) によって表現できる．これは，ラプラス変換された 2 変数の積を逆変換する公式である．任意の入力 $x(t)$ に対する応答は，次のように書ける．

$$\begin{aligned} y(t) &= L^{-1}\{Y(s)\} = L^{-1}\{G(s)X(s)\} \\ &= \int_0^t g(t-\tau)x(\tau)d\tau \end{aligned} \tag{4・18}$$

ここで，$g(t)$ はプロセスのインパルス応答である．(4・18)式の積分をたたみこみ積分といい，この積分を数値計算した近似式が(4・17)式に相当する．

インパルス応答は，ステップ応答を微分することによって求まる．ステップ応答を実験的に求めることは比較的容易である．そこで，プロセスの動特性をこのステップ応答によって表すことがしばしば行われる．

4・4　過渡応答のデジタルシミュレーション

過渡応答は，逆ラプラス変換によって得られた数式に基づいて計算できる．ここでは，計算機を使って伝達関数の特性を模擬し，そこへ入力を与えて直接過渡応答を求めるデジタルシミュレーションの手法について解説する．シミュレーションに使用する言語はBASICである．BASICのプログラムを組むことのできる（ポケット）コンピュータを用いて，基本的な伝達関数を模擬する方法について述べる．

積分要素

まずはじめに，最も基本的な要素である積分要素のシミュレーションを考える．

$$Y(s) = \frac{1}{s} X(s) \tag{4・19}$$

上式は時間領域で次のように表される．

$$\begin{aligned} y(t) &= \int_0^t x(t)dt \\ &\approx \sum_i x(t_i)\Delta t = y(t - \Delta t) + x(t)\Delta t \end{aligned} \tag{4・20}$$

ここで，入力 $x(t)$ の積分を微小長方形の面積 $x(t_i)\Delta t$ の和で近似している．なお，Δt は時間の刻み幅である．いま，時間 t から1刻み前の時間における積分値 $y(t - \Delta t)$ がわかっていれば，これに1個の微小長方形の面積を加えると現時点での積分値が求まる．この操作の繰り返しによって，時間0から任意の時間までの積分が可能となる．

上記の数値計算法のプログラムの例を次に示す．ただし，この例では入力 X が時間的に変化せず一定値を取る場合である．

プログラムリスト：1

```
110   REM INTEGRAL ELEMENT
120   CLEAR: WAIT
130   INPUT "DT=";DT
140   INPUT "TE=";TE
150   INPUT" X=";X
160   FOR TM=0 TO TE STEP DT
170   PRINT"TM=";TM
180   PRINT" Y=";Y
190   Y=Y+X*DT
200   NEXT TM
210   END
```

ここで，簡単なBASICに関する解説を行う．プログラムの一覧表をリストという．プログラムは各行からできており，行の先頭にある番号を行番号という．計算は原則として行番号の順に行う．

・行番号110の REM は，計算には関係せず，プログラムをわかりやすくするための注釈（ INTEGRAL ELEMENT ）を表示する．

・行番号120の CLEAR は，以後に使用する変数の中身を消去する．WAIT は，以下に出てくる PRINT 命令が実行されたあと，次のプログラムを実行するのを待て，という命令である．リターンキーを押すと，以後のプログラムが実行される．なお，一般のパソコンのBASICにはこの命令はない．

・行番号130は入力命令で，数値を入力してリターンキーを押すと，その値を変数 DT（時間刻み幅）が受け取る．

・行番号160は，行番号200と対になっていて，初めは TM=0 として，行番号200までのプログラムを実行し，次に TM=0+DT として再び行番号200までプログラムを実行する．そして，順次 TM を DT づつ増加させて，TM が TE を超えるようになるまで繰り返し計算を行う．（ TM は時間を，TE は計算打ち切り時間を示す．）

・行番号190の式が(4·20)式の計算を行っている．BASICでは，＝の右辺が

初めに実行され，その結果を左辺の変数に代入する．したがって，右辺の Y と左辺の Y とは異なった値を取る．

・行番号210はプログラムの実行を終了させる．

1次遅れ要素

1 次遅れ要素の伝達関数を次式で与える．

$$\frac{Y(s)}{X(s)}=\frac{K}{Ts+1} \tag{4·21}$$

この式を展開して，

$$\begin{aligned}
Y(s)(Ts+1) &= KX(s) \\
Y(s)Ts &= KX(s)-Y(s) \\
Y(s) &= \left(\frac{1}{s}\right)\left(\frac{1}{T}\right)\{KX(s)-Y(s)\} \\
&= \left(\frac{1}{s}\right)U(s)
\end{aligned} \tag{4·22}$$

ここで $U(s)$ は，

$$U(s)=\left(\frac{1}{T}\right)\{KX(s)-Y(s)\} \tag{4·23}$$

したがって，(4·19)式の入力 $X(s)$ の代わりに，$U(s)$を入力としたものが，1次遅れ要素である．ただし，数値計算では(4·23)式の右辺にある$Y(s)$として，現時点の出力 $y(t)$の代わりに 1 刻み前の時間の出力 $y(t-\Delta t)$ を使用する．

これらの数式の関係を図で示すと次のようになる．

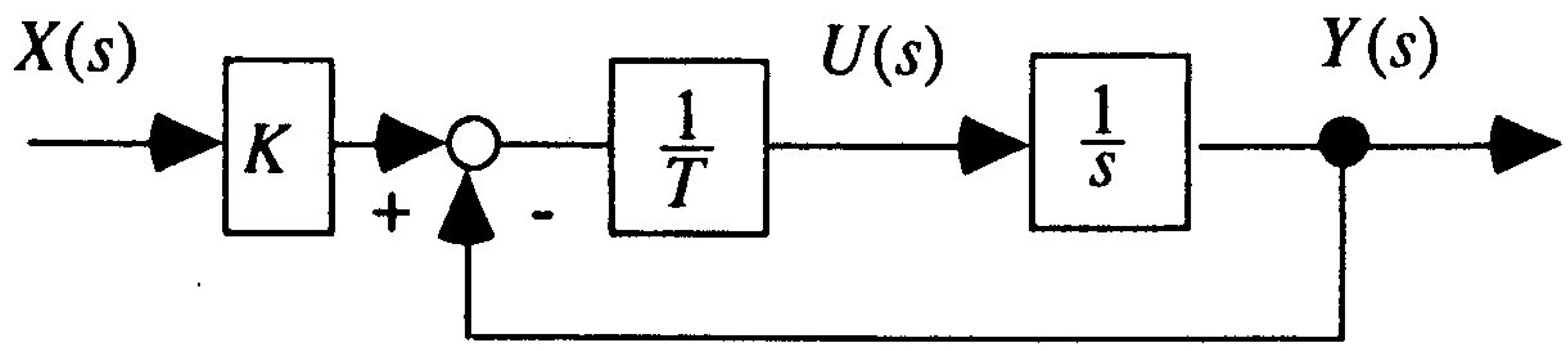

図 4·4　1 次遅れ要素の伝達関数

一定入力 X に対する出力 Y を求めるプログラムのリストを以下に示す.

プログラムリスト：2

```
300   REM 1ST-ORDER ELEMENT
310   CLEAR: WAIT
320   INPUT "DT=";DT
330   INPUT" T=";T
340   INPUT" K=";K
350   INPUT "TE=";TE
360   INPUT" X=";X
370   FOR TM=0 TO TE STEP DT
380   PRINT"TM=";TM
390   PRINT" Y=";Y
400   U=(1/T)*(K*X-Y)
410   Y=Y+U*DT
420   NEXT TM
430   END
```

・行番号400で，積分要素に入る入力 U を，1次要素への入力 X と出力 Y およびゲイン定数 K，時定数 T から，(4·23)式に基づいて計算している.

2次遅れ要素

2 次遅れ要素の伝達関数は次式で与えられる．

$$\frac{Y(s)}{X(s)}=\frac{K}{T^2s^2+2T\zeta s+1} \tag{4·24}$$

この式を展開して，

$$\begin{aligned}
&Y(s)(T^2s^2+2T\zeta s+1)=KX(s)\\
&Y(s)(T^2s^2)=KX(s)-Y(s)(2T\zeta s+1)\\
&Y(s)=\left(\frac{1}{T^2s^2}\right)\left\{KX(s)-Y(s)-2T\zeta s\right\}\\
&\qquad=\left(\frac{1}{s}\right)\left[\left(\frac{1}{s}\right)\left(\frac{1}{T^2}\right)\left\{KX(s)-Y(s)\right\}-\left(\frac{2\zeta}{T}\right)Y(s)\right]\\
&\qquad=\left(\frac{1}{s}\right)U(s)
\end{aligned} \tag{4·25}$$

ここで，

$$\left.\begin{aligned}
&U(s)=V(s)-M(s)\\
&M(s)=\left(\frac{2\zeta}{T}\right)Y(s)\\
&V(s)=\left(\frac{1}{s}\right)W(s)\\
&W(s)=\left(\frac{1}{T^2}\right)\left\{KX(s)-Y(s)\right\}
\end{aligned}\right\} \tag{4·26}$$

(4·25)式と(4·26)式中の $V(s)$ の 2 度の積分操作によって，入力 $X(s)$ に対する応答 $Y(s)$ が求まる．これらの展開式の関係をブロック線図にて示すと以下のようになる．

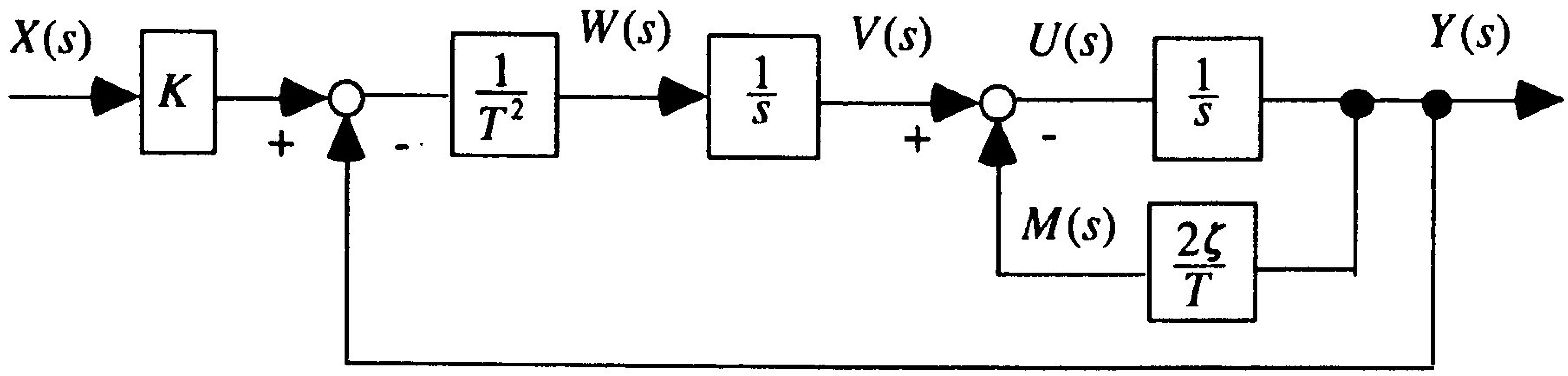

図 4·5　　2 次遅れ要素の伝達関数

上図で$Y(s)$の信号がループを形づくる所，すなわち(4·26)式の$M(s)$, $W(s)$右辺に現れる$Y(s)$は，数値計算では全て$y(t-\Delta t)$の値を用いる．一定入力 X に対する出力 Y を求める数値計算のプログラムリストを以下に示す．

プログラムリスト：3

```
500   REM 2ND-ORDER ELEMENT
510   CLEAR: WAIT
520   INPUT "DT=";DT
530   INPUT" T=";T
540   INPUT" Z=";Z
550   INPUT" K=";K
560   INPUT "TE=";TE
570   INPUT" X=";X
580   FOR TM=0 TO TE STEP DT
590   PRINT"TM=";TM
600   PRINT" Y=";Y
610   W=(1/T^2)*(K*X-Y)
620   M=(2*Z/T)*Y
630   V=V+W*DT
```

```
640   U=V-M
650   Y=Y+U*DT
660   NEXT TM
670   END
```

・行番号630と行番号650にて積分操作を行っている.

・行番号610の ^ はべき乗の演算操作を表している.

むだ時間要素

むだ時間要素の伝達関数は次式で与えられる.

$$\frac{Y(s)}{X(s)}=e^{-Ls} \tag{4·27}$$

この関数では，入力が一定時間 L だけ遅れて出力側に出てくる．この要素のシミュレーションプログラムを以下に示す.

プログラムリスト：4

```
700   REM TIME LAG ELEMENT
710   CLEAR: WAIT
720   INPUT" L=";L
730   INPUT"TE=";TE
740   INPUT "DT=";DT
750   NL=INT(L/DT+0.5)
760   DIM D(NL)
770   FOR TM=0 TO TE STEP DT
780   INPUT" X=";X
790   D(0)=X: Y=D(NL)
800   PRINT "TM=";TM
810   PRINT " Y=";Y
```

```
820   FOR I=NL TO 1 STEP -1
830   D(I)=D(I-1)
840   NEXT I
850   NEXT TM
860   END
```

・行番号750の INT は，整数化関数といい，L を時間刻み DT で割った値の整数部 NL を求める．ここで，L/DT に 0.5 を加えているのは少数点以下の数値を四捨五入するためである．

・行番号760の DIM は，配列と呼ばれる変数 D を定義し，その大きさを決めるものである．配列とは添字付き変数のようなもので，1つの変数名で数個の数値を収納することができる．

・行番号790では，入力 X の値を0番目の D に収納し，また NL 番目の D の数値を出力 Y に代入している．

・行番号820から840の間では，I=NL で (NL-1) 番目の D の値を NL 番目の D に代入し，次に (NL-2) 番目の D の値を (NL-1) 番目の D に代入，この操作を繰り返して，0番目の D の値を1番目の D に代入するまで続ける．この代入の繰り返しは，時間 TM が変わる度に行われる．ここが，むだ時間要素のシミュレーションの要点である．