

10 制御系の過渡応答

制御系の設計にあたっては、先の第7章で述べたように、定値制御における外乱の影響を調べる調節問題と、設定値の変化に対する制御変数の追従性を調べるサーボ問題とがある。本章では、ステップ状の外乱あるいは設定値変化に対する、フィードバック制御の過渡応答を調べ、PID-制御動作の働きについて検討する。次いで、デジタルシミュレーションによって種々の条件下におけるフィードバック制御系の過渡応答を求める。さらに、制御系の安定性に関する検討を行い、過渡応答法によるPID-制御器のパラメータチューニングについて解説する。

10・1 制御系のステップ応答

フィードバック制御系のステップ応答を調べるため、図10・1のような制御系を考える。ここで、検出部および操作部の遅れはプロセスの遅れに比べ十分小さいと見なして、これらの特性を定数のゲイン K_m, K_v のみで表した。

プロセスの伝達関数 $G_p(s)$ および外乱の伝達関数 $G_D(s)$ が次の1次遅れ要素にて与えられる制御系を考える。

$$G_p(s) = \frac{K_P}{Ts+1}, \quad G_D(s) = \frac{K_D}{Ts+1} \quad (10.1)$$

さらに、制御器として最も基本的な比例制御を考える。

$$G_C = K_C \quad (10.2)$$

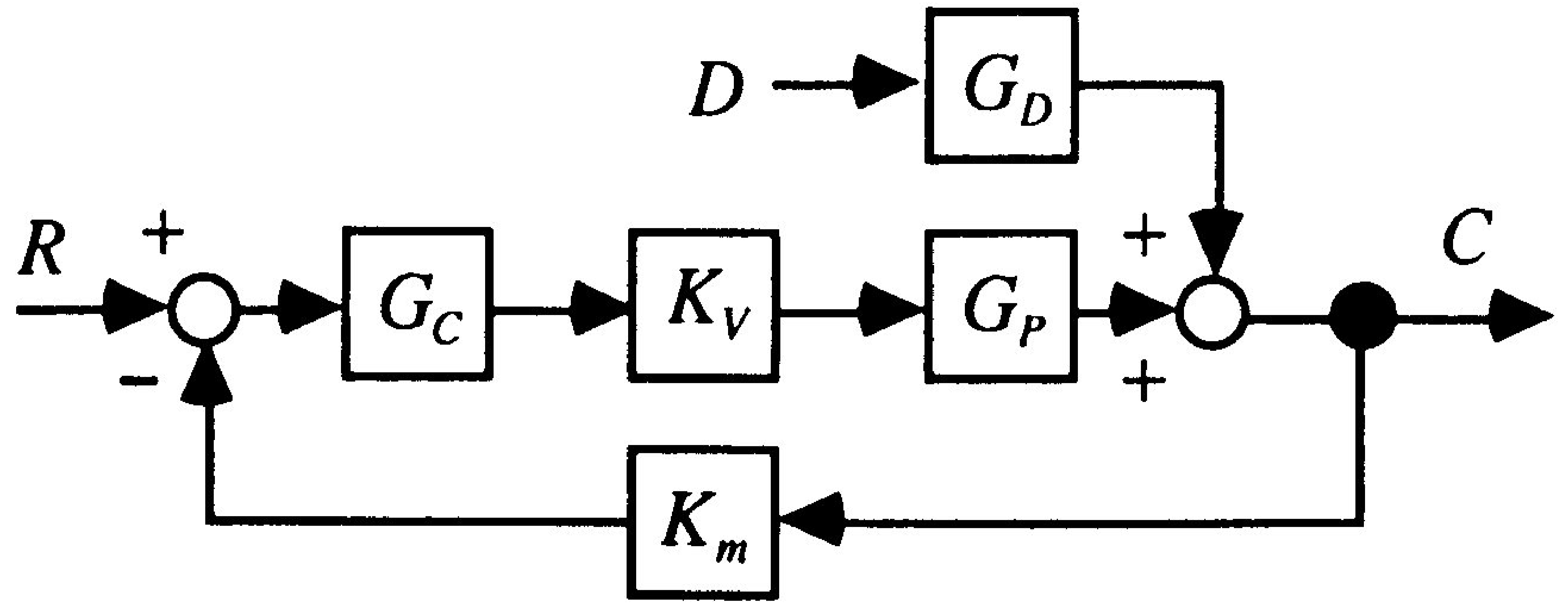


図 10・1 フィードバック制御系

設定値変化に対する制御出力の総括伝達関数は、(7・3)式に従って次のように与えられる。

$$\begin{aligned} \frac{C(s)}{R(s)} = G_{OV,1} &= \frac{K_C K_V \frac{K_P}{Ts+1}}{1 + K_m K_C K_V \frac{K_P}{Ts+1}} \\ &= \frac{K_C K_V K_P}{Ts+1 + K_m K_C K_V K_P} = \frac{K_{OV,1}}{T_{OV}s+1} \end{aligned} \quad (10\cdot3)$$

ここで、

$$K_{OV,1} = \frac{K_C K_V K_P}{1 + K_m K_C K_V K_P} \quad (10\cdot4)$$

$$T_{OV} = \frac{T}{1 + K_m K_C K_V K_P}$$

同様にして、外乱に対する制御出力の総括伝達関数は(7・4)式から次のようになる。

$$\begin{aligned} \frac{C(s)}{D(s)} &= G_{OV,2} = \frac{K_D}{Ts + 1 + K_m K_C K_V K_P} \\ &= \frac{K_{OV,2}}{T_{OV}s + 1} \end{aligned} \quad (10.5)$$

ここで,

$$K_{OV,2} = \frac{K_D}{1 + K_m K_C K_V K_P} \quad (10.6)$$

以上のように、比例制御にて1次遅れプロセスを制御する場合、設定値変化あるいは外乱に対する制御出力の総括伝達関数はいずれも1次遅れ要素となる。ここで、総括伝達関数の時定数 T_{OV} およびゲイン定数 $K_{OV,1}, K_{OV,2}$ は、比例制御器の比例ゲイン K_C によって変化する。

ここで、比例制御の特性を調べるため、プロセスの特性として $T=1, K_P=1, K_D=1$ とし、また検出部、操作部の特性として $K_m=1, K_V=1$ とし、制御器の比例ゲイン K_C を種々変化させて制御系のステップ応答を計算する。1次遅れ要素のステップ応答は、先に示した(4.9)式にて与えられる。すなわち、単位ステップ状の設定値変化、 $R(s)=1/s$ 、に対する応答は次式となる。

$$\begin{aligned} c(t) &= K_{OV,1}(1 - e^{-t/T_{OV}}) \\ &= \left(\frac{K_C}{1 + K_C} \right) \{ 1 - e^{-(1+K_C)t} \} \end{aligned} \quad (10.7)$$

同様に、単位ステップの外乱、 $D(s)=1/s$ 、に対しては、

$$\begin{aligned}
 c(t) &= K_{OV,2}(1 - e^{-t/T_{ov}}) \\
 &= \left(\frac{1}{1 + K_C} \right) \left\{ 1 - e^{-(1+K_C)t} \right\}
 \end{aligned}
 \tag{10\cdot8}$$

となる。

(10·8)式から、比例ゲインをパラメータとして、単位ステップの外乱に対する応答を計算すると、図10·2のようになる。

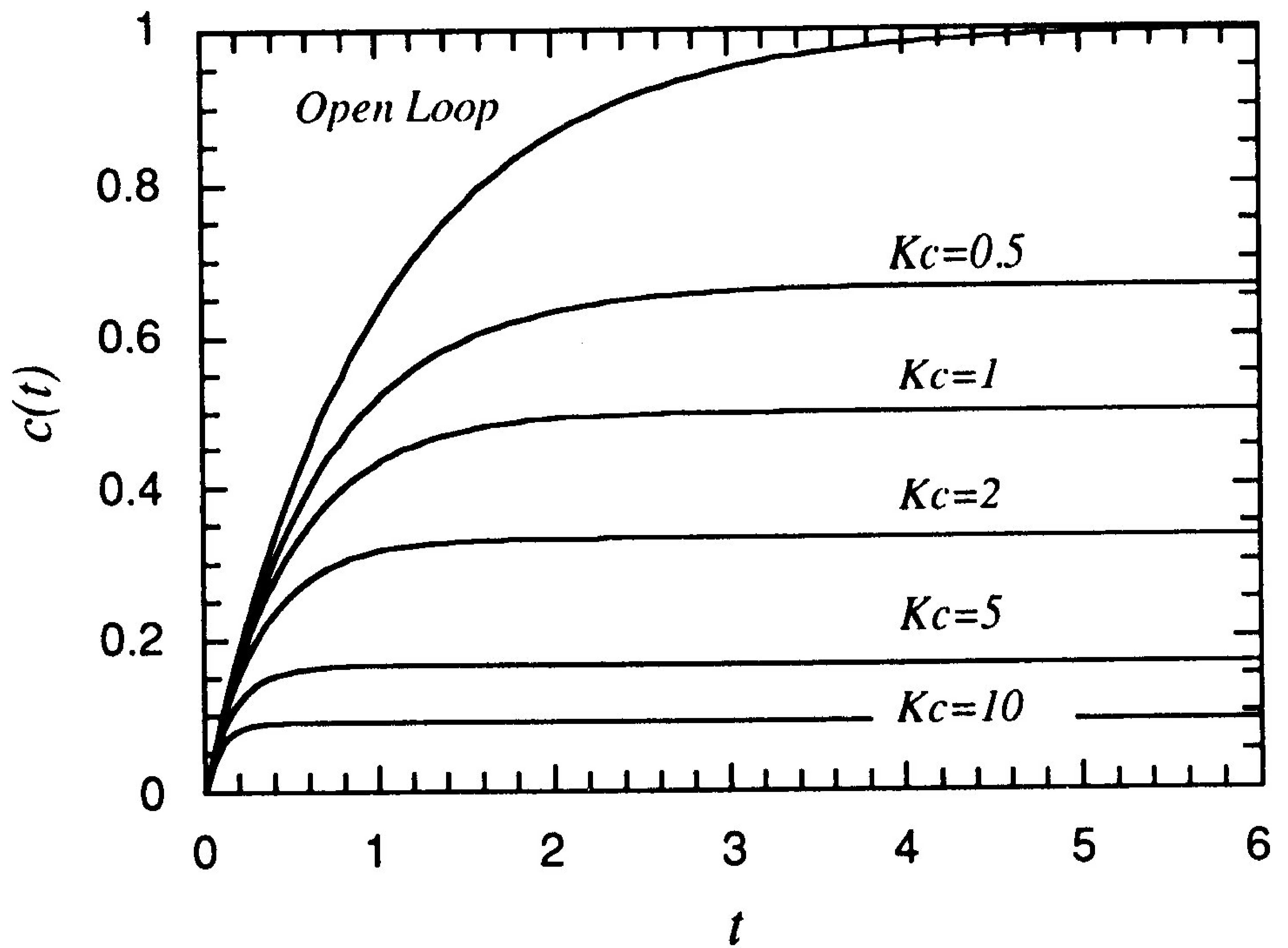


図 10·2 外乱に対するステップ応答 (比例制御)

図10·2にて、 $K_C = 0$ の応答はフィードバック制御を行わない場合の応答で、オープンループの応答という。外乱に対する調節問題では、設定値は変化させていない。そこで、外乱の挿入にもかかわらず、制御系の出力 $c(t)$ が変化しない場合、すなわち、 $c(t) = 0$ となることが最も望ましい制御結果となる。しかし図10·2の結果を見

ると、いずれの比例ゲインの場合も、ステップ応答の最終値はゼロとはならず、あるオフセットを示している。この応答の最終値は(10・8)式から次式となる。

$$\lim_{t \rightarrow \infty} c(t) = c_f = \frac{1}{1 + K_C} \quad (10 \cdot 9)$$

オフセットは、制御系の応答の最終値と設定値との差（差の絶対値）であるから、次のようになる。

$$\text{offset} = |c_f - R| = |c_f - 0| = \frac{1}{1 + K_C} \quad (10 \cdot 10)$$

上記の例では、オフセットを取り除くためには、比例ゲインを無限大にしなくてはならない。比例ゲインが無限大の制御は、オンオフ制御である。しかし、実際のオンオフ制御では、外乱に対する応答はサイクリングを示す。その理由は以下のように説明される。この制御系の解析にあたって、検出部、操作部の遅れは、プロセスの遅れに比べて小さいとして（検出部、操作部の時定数とプロセスの時定数 T を比較すると、 $T_m, T_v \ll T$ ）その特性を定数のゲインのみで表してきた。しかし、制御器の比例ゲインを大きくすると、制御系自身の総括伝達関数の時定数 T_{OV} は、(10・4)式からわかるように小さくなる。そこで、 K_C が大きい場合には、検出部および操作部の遅れも無視できなくなり、制御系の応答はサイクリングを示すようになる。これは、制御系の安定性に関連する問題で、後の節にて詳しく説明する。

次に、制御動作として比例-積分制御を行う場合について検討する。制御器の伝達関数は次式で与えられる。

$$G_C(s) = K_C \left(1 + \frac{1}{T_I s} \right) \quad (10 \cdot 11)$$

プロセスの特性は1次遅れで $T=1, K_P=K_D=1$ とし、検出部、操作部を $K_m=K_v=1$ とすると、外乱に対する制御変数の総括伝達関数は次のようになる。

$$\begin{aligned} \frac{C(s)}{D(s)} &= \frac{\frac{1}{s+1}}{1+K_C\left(1+\frac{1}{T_I s}\right)\left(\frac{1}{s+1}\right)} \\ &= \frac{T_I s}{T_I s^2 + T_I(1+K_C)s + K_C} \end{aligned} \quad (10\cdot12)$$

いま、制御器のパラメータを $K_C=1, T_I=1$ とし、ステップ状の外乱に対する応答を求めると次のようになる。[表3・2 (p26), ラプラス逆変換表参照]

$$\begin{aligned} c(t) &= L^{-1}\left\{\frac{s}{s^2+2s+1} \cdot \frac{1}{s}\right\} \\ &= L^{-1}\left\{\frac{1}{(s+1)^2}\right\} = te^{-t} \end{aligned} \quad (10\cdot13)$$

制御器の積分時間を変えて、 $K_C=1, T_I=0.5$ とすると、外乱に対する応答は次のようになる。[(4・6)式 (p37) 参照]

$$\begin{aligned} c(t) &= L^{-1}\left\{\frac{0.5s}{0.5s^2+s+1} \cdot \frac{1}{s}\right\} \\ &= L^{-1}\left\{\frac{0.5}{0.5s^2+s+1}\right\} = e^{-t} \sin(t) \end{aligned} \quad (10\cdot14)$$

これらの応答を図示すると、図10・3となる。なお、この図には、オープンループの応答および比例制御のみの応答も併せて示した。積分制御を付け加えると、制御系の応答のオフセットはゼロとなった。

積分制御の動作を強くしていくと、すなわち積分時間 T_I を小さくしていくと、ステップ応答には振動成分が現れてくる。しかし、この系では T_I を1から0.5に変えると、その制御性は改善され、外乱の影響をより少なくすることができた。

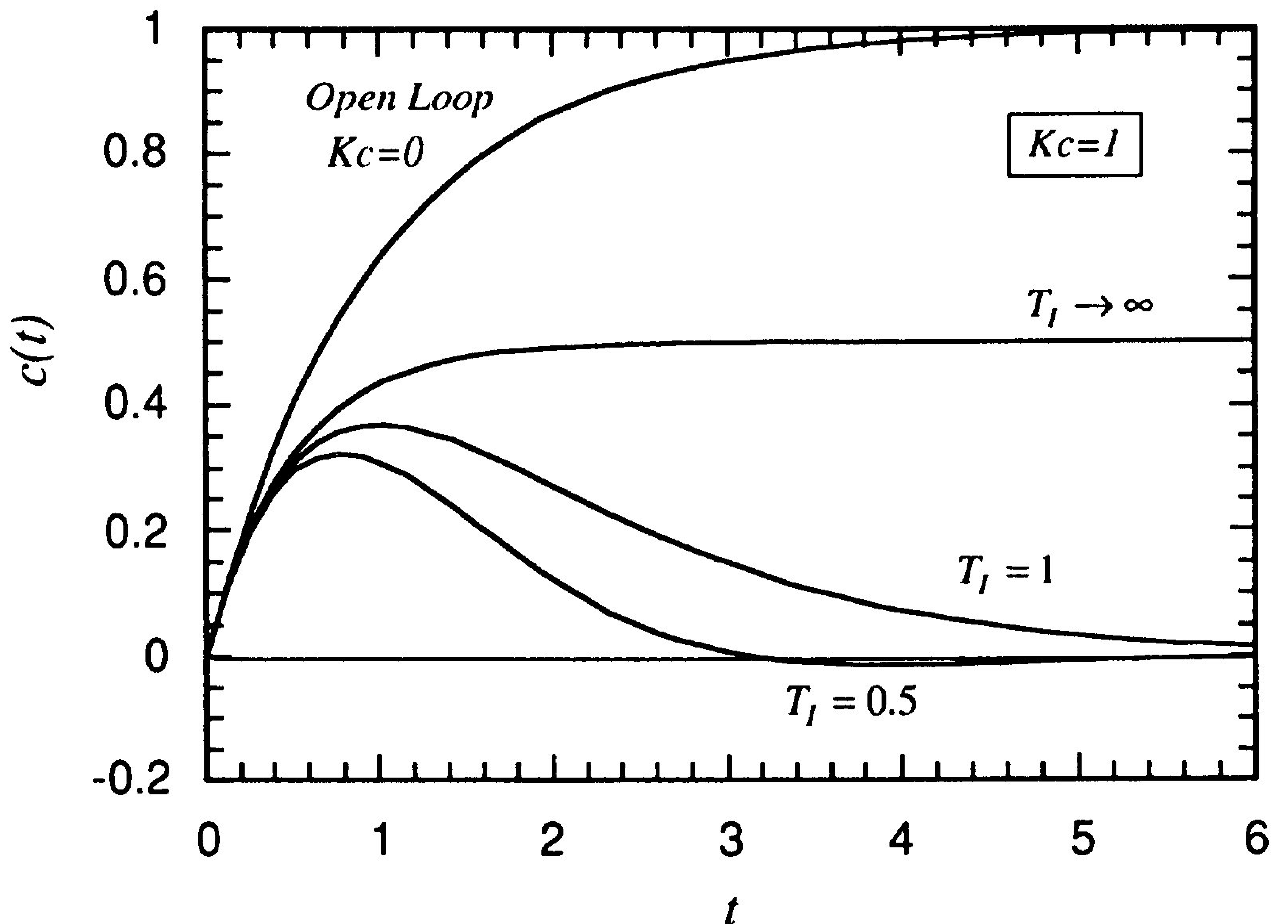


図 10.3 外乱に対するステップ応答
(比例-積分制御)

次に、比例-積分-微分制御動作について検討する。プロセスは1次遅れで $T=1$, $K_P=K_D=1$ とし、検出部、操作部は $K_m=K_v=1$, とする。なお、PID-制御器の伝達関数は(9.11)式で与えられている。

$$G_C = K_C \left(1 + \frac{1}{T_I s} + T_D s \right) \quad (9.11)$$

ここで、制御器のパラメータとして $K_C = 1$, $T_I = 1$ とし、外乱に対する総括伝達関数を求めると次のようになる。

$$\frac{C(s)}{D(s)} = \frac{s}{(1+T_D)s^2 + 2s + 1} \quad (10 \cdot 15)$$

微分時間を $T_D = 0.5$ として、単位ステップ状の外乱に対する応答を求めると次式となる。 [(4・6)式 (p37) 参照]

$$\begin{aligned} c(t) &= L^{-1} \left\{ \frac{s}{1.5s^2 + 2s + 1} \cdot \frac{1}{s} \right\} = L^{-1} \left\{ \frac{1}{1.5s^2 + 2s + 1} \right\} \\ &= \frac{1}{\sqrt{0.5}} e^{(-t/1.5)} \sin \left(\frac{\sqrt{0.5} t}{1.5} \right) \end{aligned} \quad (10 \cdot 16)$$

また、 $T_D = 1$ としたときのステップ応答は次式となる。

$$\begin{aligned} c(t) &= L^{-1} \left\{ \frac{s}{2s^2 + 2s + 1} \cdot \frac{1}{s} \right\} = L^{-1} \left\{ \frac{1}{2s^2 + 2s + 1} \right\} \\ &= e^{(-t/2)} \sin \left(\frac{t}{2} \right) \end{aligned} \quad (10 \cdot 17)$$

これらの応答を図示すると図10・4となる。この図には、オープンループの応答および、PI制御の結果 ($T_D = 0$) も併せて示した。この図を見ると、微分動作を付け加えることによって、応答は穏やかとなり、PI制御の結果よりも遅くなることがわかる。しかし、この1次遅れプロセスの制御系では、微分動作を加えることによって顕著な制御性の改善は見られなかった。

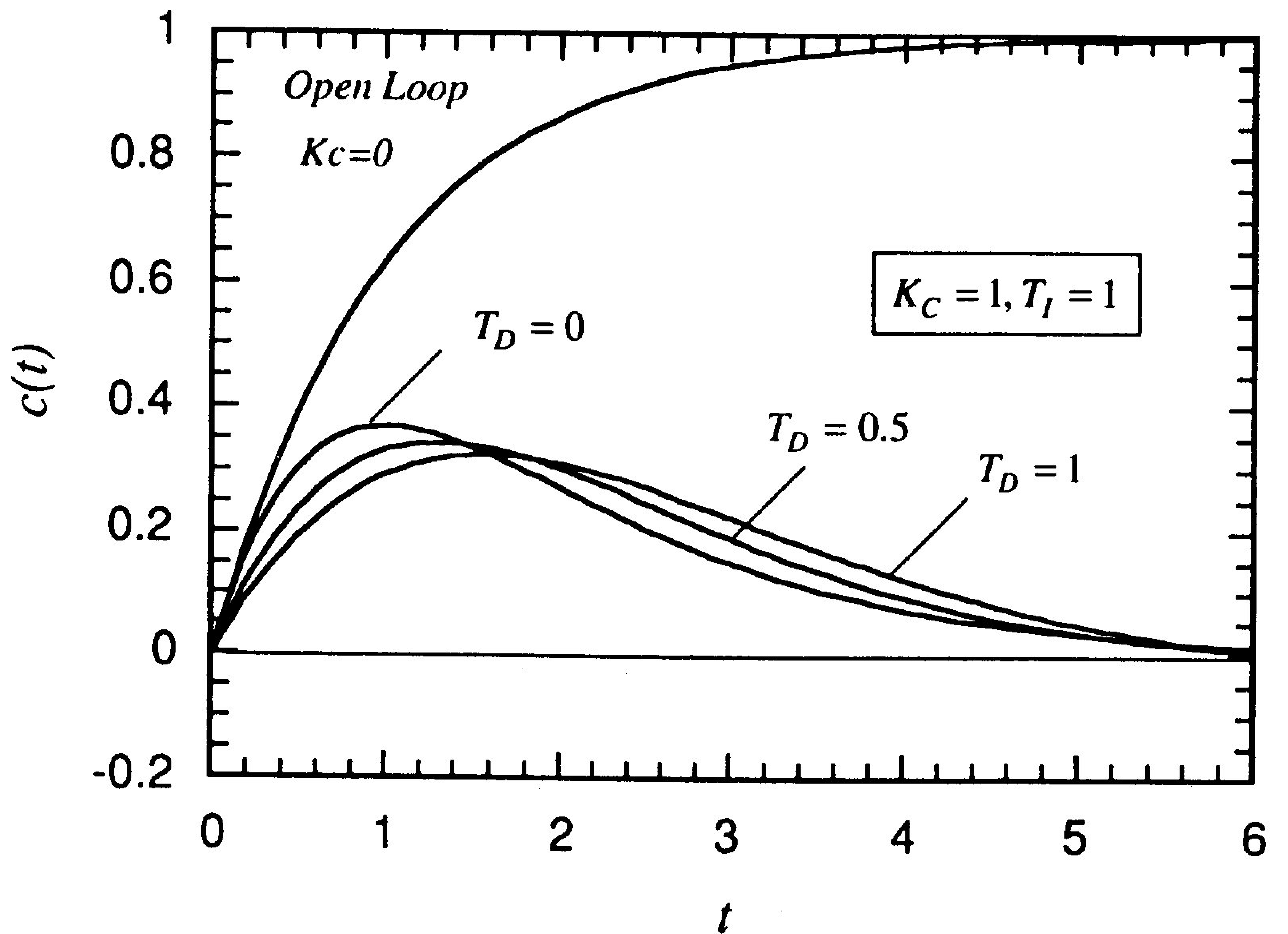


図 10・4 外乱に対するステップ応答
(比例 - 積分 - 微分制御)

制御系のオフセットを求めるためには、系の過渡応答が最終的にどのような値を取るかを知る必要がある。ラプラス変換された関数では、その逆ラプラス変換を求めなくとも、 s -領域での関数からその最終値を求めることができる。すなわち、先の§3・1で示したラプラス変換の最終値の定理 (p22) を応用すればよい。

ステップ入力に対する制御系の応答は次のようになる。

$$C(s) = G_{OV}(s) \cdot \frac{1}{s} \quad (10 \cdot 18)$$

そこで、(3・6)式を適用して最終値を求めると、

$$\begin{aligned}
 c_f &= \lim_{t \rightarrow \infty} c(t) = \lim_{s \rightarrow 0} \{sC(s)\} \\
 &= \lim_{s \rightarrow 0} \left\{ sG_{OV}(s) \frac{1}{s} \right\} = \lim_{s \rightarrow 0} \{G_{OV}(s)\}
 \end{aligned}
 \tag{10\cdot19}$$

先に例示した1次遅れプロセスをPI-制御する系について上式を適用する。外乱に対する総括伝達関数は(10\cdot12)式で与えられる。そこで、ステップ状の外乱に対する応答の最終値は、(10\cdot19)式から、

$$c_f = \lim_{s \rightarrow 0} \left\{ \frac{T_I s}{T_I s^2 + T_I(1 + K_C)s + K_C} \right\} = 0
 \tag{10\cdot20}$$

すなわち、この応答の最終値はゼロとなり、オフセットを取り除くことができる。(10\cdot20)式からわかるように、最終値がゼロとなるのは総括伝達関数の分子に s があるためである。この s は、制御器の伝達関数に含まれている積分動作に由来する。以上のことから、積分制御を加えることによって、オフセットを取り除けることがわかる。