

## 1 2 複合制御系

プロセス制御においては、フィードバック制御が基本である。しかし、プロセスの遅れが大きい場合、あるいは大きなむだ時間要素が含まれる場合には、その制御性はかなり悪くなり、より良好な制御を行うためには工夫が必要となる。そこで、このようなプロセスの制御には、多重の制御経路を持つ複合制御系が採用される。本章では、フィードフォワード・フィードバック制御系、カスケード制御系およびむだ時間補償制御系について解説する。

### 1 2・1 フィードフォワード制御

図7・2で示した加熱タンクの制御系を例にとり、フィードフォワード制御(*feedforward control*)について説明する。プロセスの制御変数はタンク内の温度であり、操作変数はヒータにからの熱量である。主な外乱は流入液の温度変化である。このプロセスに対し、通常のフィードバック制御を行うと、そのブロック線図は図7・1のようになり、ここでプロセス伝達関数  $G_p$  および外乱伝達関数  $G_D$  は、(7・1)式から、いずれも1次遅れ要素となる。

フィードバック制御系においては、外乱の影響を取り除くため、その影響が制御変数に現れてから始めて打ち消しの制御動作が開始される。したがって、プロセスの遅れが大きい場合には、打ち消し動作の効果が現れるまでに時間がかかるため、外乱の影響はかなり大きく出てしまう。いま、このプロセスに入る主要な外乱が明らかの場合、この外乱を直接検出して、プロセスに外乱が入ると同時に打ち消しの制御動作を行えば、その制御性は大いに改善される。こ

のような考えにしたがって、フィードバック制御系に、フィードフォワード制御経路を付け加えることが行われる。加熱タンク系のフィードフォワード・フィードバック制御系のブロック線図を図12・1に示す。ここで、フィードバックループ中の検出部および操作部の伝達関数は定数とした。

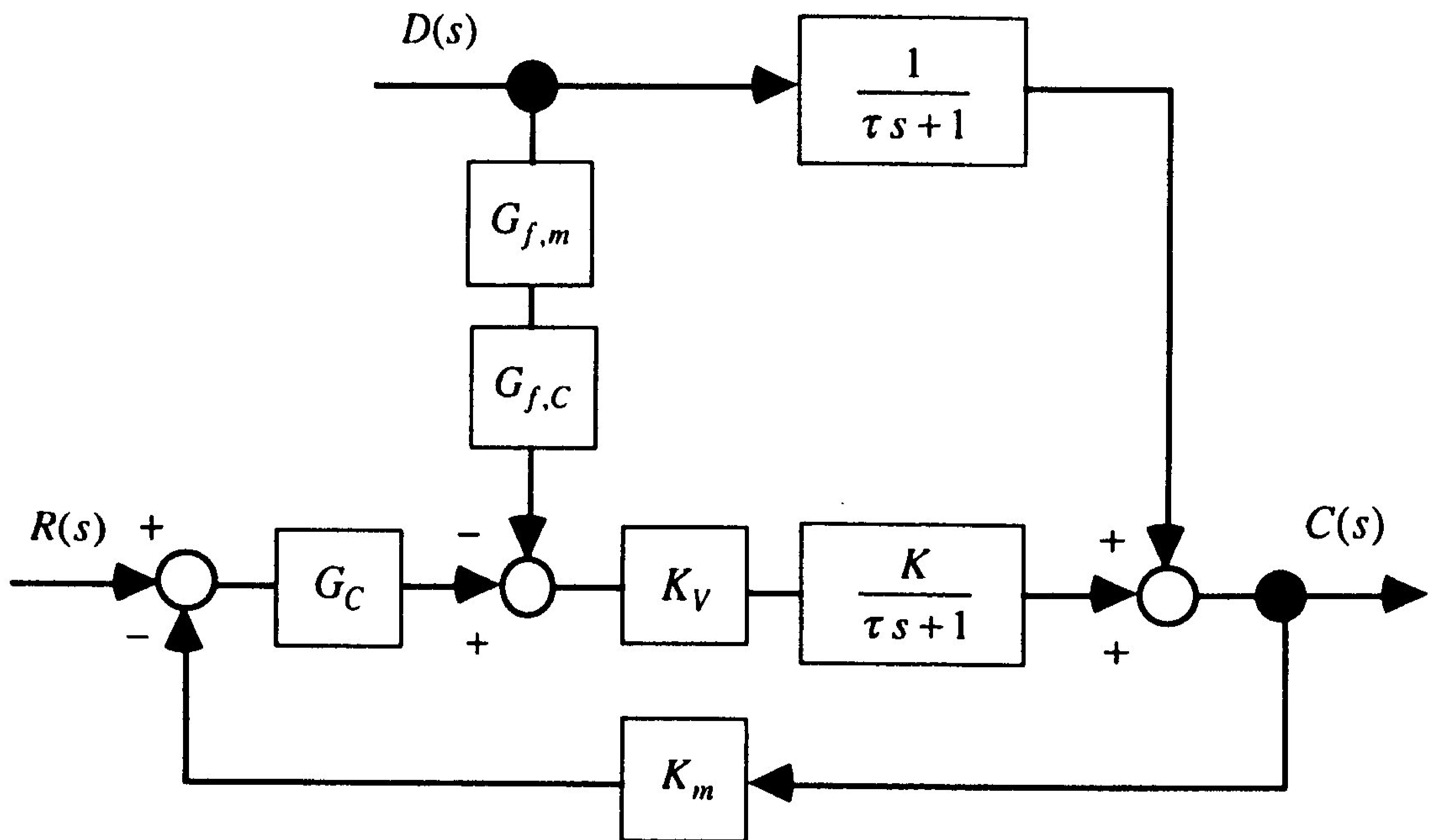


図 12・1 フィードフォワード・フィードバック制御系

上図中の  $G_{f,m}$  は外乱検出部の伝達関数であり、 $G_{f,c}$  はフィードフォワード制御器の伝達関数である。この制御系の外乱と制御変数との間の総括伝達関数を求めると次のようになる。

$$\frac{C(s)}{D(s)} = \frac{\frac{1}{\tau s + 1} - G_{f,m} G_{f,c} K_V \frac{K}{\tau s + 1}}{1 + G_C K_V \frac{K}{\tau s + 1} K_m} \quad (12 \cdot 1)$$

$$= \frac{1 - G_{f,m} G_{f,c} K_V K}{\tau s + 1 + G_C K_V K K_m}$$

ここで、(12・1)式の分子がゼロとなるようなフィードフォワード制御器を設計できれば、外乱の影響を完全に取り除くことが可能となる。すなわち、 $1 - G_{f,m} G_{f,c} K_V K = 0$  から、

$$G_{f,c} = \frac{1}{G_{f,m} K_V K} \quad (12 \cdot 2)$$

が、理想的なフィードフォワード制御器の伝達関数となる。一般的には、外乱伝達要素の伝達関数を  $G_D(s)$  とし、プロセスの伝達関数を  $G_P(s)$ 、操作端の伝達関数を  $G_V(s)$  とすると、

$$G_{f,c}(s) = \frac{G_D(s)}{G_{f,m}(s) G_V(s) G_P(s)} \quad (12 \cdot 3)$$

となる。

いま、外乱検出部の特性が  $G_{f,m} = K_{f,m}$  で与えられるとすると、(12・2)式から、

$$G_{f,c} = \frac{1}{K_{f,m} K_V K} = K_{f,c} \quad (12 \cdot 4)$$

となり、比例制御がフィードフォワード制御器となる。外乱検出部の特性が1次遅れ要素で与えられる場合には、

$$G_{f,m} = \frac{K_{f,m}}{T_f s + 1}$$

として,

$$G_{f,c} = \frac{T_{f,m}s+1}{K_{f,m}K_VK} = K_{f,c}(T_{f,m}s+1) \quad (12\cdot5)$$

となる. これは比例-微分制御動作によって達成され, 位相進み要素といわれる. 外乱検出部の特性がむだ時間要素で与えられる場合には,  $G_{f,m} = K_{f,m}e^{-L_f s}$  とすると,

$$G_{f,c} = \frac{e^{L_f s}}{K_{f,m}K_VK} = K_{f,c}e^{L_f s} \quad (12\cdot6)$$

この伝達関数は時間進み要素で, 今から  $L_f$  時間経過した後に得られる未来の情報をいま知る必要がある. そのため実現不可能となる. しかし, このような場合でも,  $G_{f,c} \approx K_{f,c}$  とすること (動的な遅れの影響を無視して, 静特性のみにて制御動作を決定する) によって, フィードフォワード制御の効果がかなり現れ, 制御性の改善がなされる.

フィードフォワード制御器の伝達関数は, (12·2), (12·3)式のようにして設定されるが, そのためにはプロセスの動特性を正しく把握しておく必要がある. プロセスの特性が, 上記の例と異なり, かなり複雑な場合には, フィードフォワード制御器の伝達関数もまた複雑な形となり, 高次の位相進み要素となる場合がある. いまは, デジタル制御器を使用することが多いので, かなり複雑な制御動作であっても実現することが可能である.

フィードフォワード制御は, 主要な外乱を直接検出することによって制御動作を決定する. したがって, この外乱以外の外乱に対しては無力である. そのため一般には, フィードバック制御系の中に組み込んで使用し, フィードフォワード制御を単独で使用することはほとんどない.

## 12・2 カスケード制御

通常のフィードバック制御系においては、その設定値を、手動にて変動させるか、あるいは時間的に予め決められた様式（パターン）に従って変動させる。しかし、あるプロセスにおいては、複数の制御ループを構成し、その主要制御ループの制御器出力によって、もう一つのフィードバック制御ループ（内部制御ループ）の設定値を変動させる場合がある。このような制御方式をカスケード制御\* (*cascade control*) とよぶ。そのブロック線図を図12・2に示す。  
(操作端の伝達関数はプロセス部に含め省略)

\* カスケードとは、階段状に流れ落ちる滝を意味する。内部ループの制御が主要ループの制御出力に支配されるためにこのように命名された。

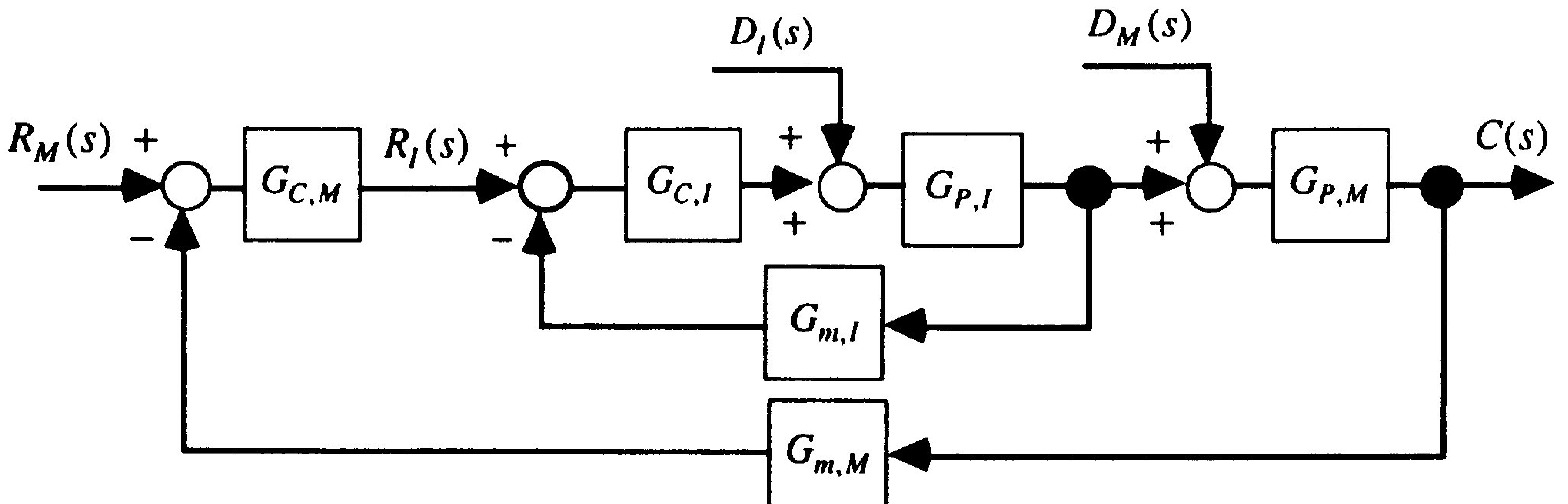


図12・2 カスケード制御系

カスケード制御の一例として、ガス燃料ボイラーの温度制御について説明する（図12・3）。この温度制御では、ボイラーから出る熱媒体の温度  $T_{out}$  [制御変数,  $C(s)$ ] を一定に保つ [設定値,  $R_M(s)$ ] ことが制御目的となる。この系に入る外乱としては熱媒体入口温度  $T_{in}$  [主要外乱,  $D_M(s)$ ] が考えられる。熱媒体の加熱はガスバーナにて行うものとし、実際の制御操作はバーナで燃焼させるガス流量  $F_{gas}$  [操作変数] を加減して行う。バーナのガス流量は、

ガス供給ラインの元圧 [外乱,  $D_I(s)$ ] が変化すると変動する恐れがある。そこで、まずはじめに、ガス流量を検出して一定流量を得るためのフィードバック制御系 (FC, 内部ループ制御器:  $G_{C,I}$ ) を構成する。そして次に、主要制御ループとして、熱媒体出口温度を計測し、その信号を基にガス流量制御系の設定値 [ $R_I(s)$ ] を変化させる制御ループ (TC, 主要ループ制御器:  $G_{C,M}$ ) を構成する。

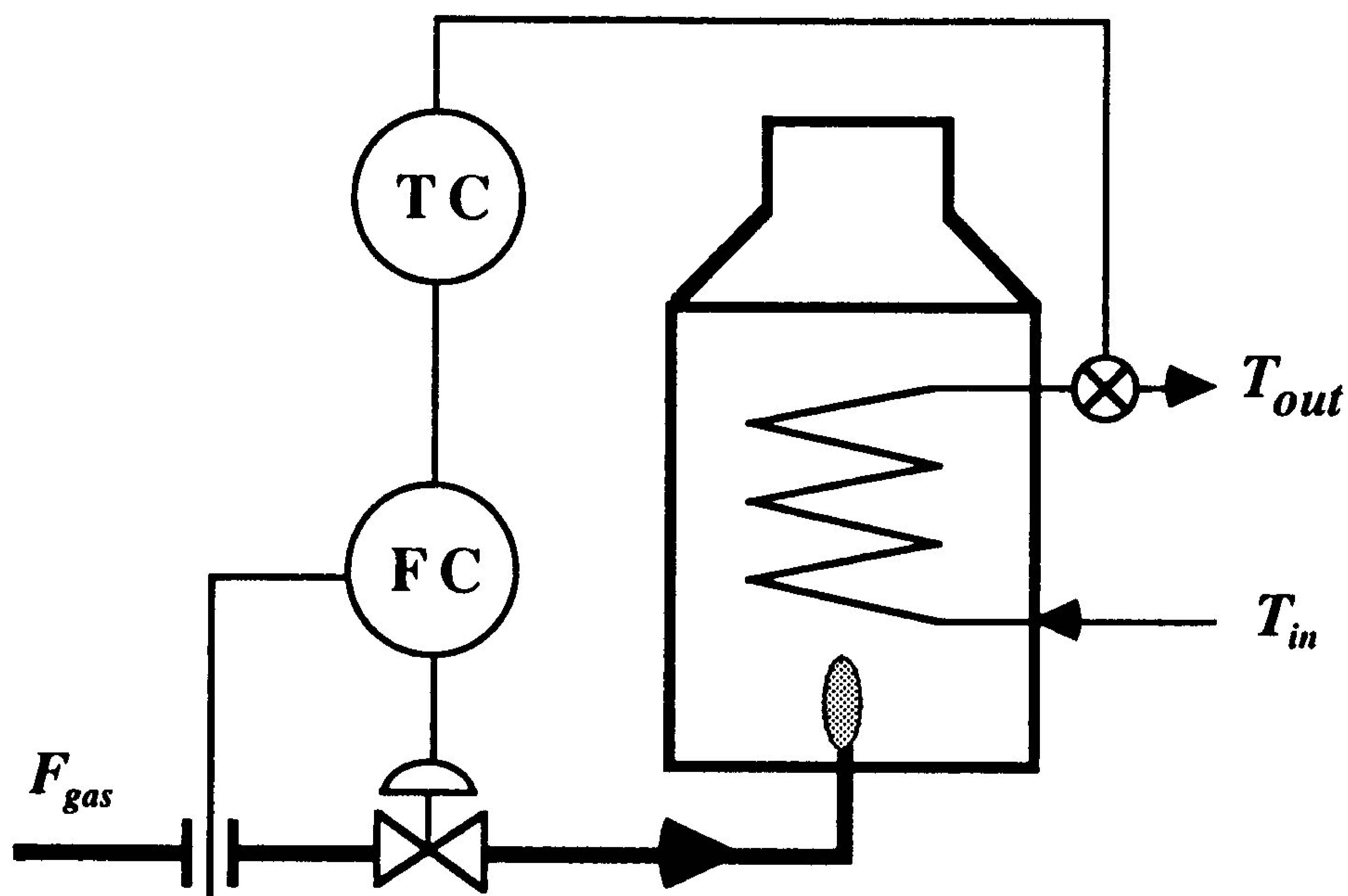


図 12・3 ガス燃料ボイラーのカスケード制御

上記の制御で、流量制御のループがない単一フィードバック制御 ( $T_{out}$  を検出して  $F_{gas}$  流量を操作する) であっても、ガスの供給元圧の変動という外乱の影響を制御することができる。しかしこの場合、元圧の変動によるバーナの熱量変化の影響が、熱媒体の出口温度の変化として現れてから、制御動作が開始される。したがって、時間的な遅れが大きく、良好な制御性を得るのは困難となる。

図12・2のブロック線図から，主要外乱の  $D_M(s)$  と制御変数  $C(s)$  との間の総括伝達関数を求めると次のようになる。

$$\frac{C(s)}{D_M(s)} = G_{OV,M}(s) = \frac{G_{P,M}}{1 + G_{C,M}G_{OV,I}G_{P,M}G_{m,M}} \quad (12\cdot7)$$

ここで，

$$G_{OV,I} = \frac{G_{C,I}G_{P,I}}{1 + G_{C,I}G_{P,I}G_{m,I}} \quad (12\cdot8)$$

この伝達関数  $G_{OV,I}$  は内部ループの総括伝達関数である。ここには，主要ループの伝達関数が含まれていない。そこで，内部ループのみの制御性を考えて，内部ループ制御器  $G_{C,I}$  のパラメータを設定する。いま，内部ループのプロセス動特性  $G_{P,I}$  に含まれる時定数やむだ時間が，主要ループのプロセス動特性  $G_{P,M}$  のそれと比べて，じゅうぶん小さいと見なせる場合， $G_{OV,I}$  の動特性もまた， $G_{P,M}$  と比較して，その時間的遅れを無視することができる。したがってこの場合， $G_{OV,I} \approx K_{OV,I}$  (定数) と扱うことができる。すると，(12・7)式の総括伝達関数に含まれる  $G_{OV,I}$  は定数として扱えるので，主要ループの制御器  $G_{C,M}$  のパラメータは，主要プロセスの動特性のみを考慮して設定することが可能となる。

図12・3に示したボイラーのカスケード制御では，流量制御ループの応答は，燃料ガス流量変化に対する熱媒体の出口温度変化の応答と比べると，じゅうぶんに速い。したがって，FCおよびTCの制御器は，それぞれ独立に，そのパラメータを設定することができる。

### 12・3 むだ時間を含むプロセスの制御

プロセス制御で扱うプロセスの中には、比較的大きな時間遅れを示すものが多い。中でも、むだ時間を含むプロセスでは、その制御性は悪く、系の安定性が損なわれる。むだ時間要素の周波数特性をみると、入力の周波数が高くなるにつれ、いくらでもその位相は遅れてしまう。そのため、ボードの安定性判定基準からも推測できるように、むだ時間が大きい場合、制御系の安定性は非常に悪くなる。このようなプロセスを安定に制御する方法として、スミスによって巧妙な方法（*Smith dead-time compensation*：スミスのむだ時間補償）が提案された。その方法をブロック線図にて以下に説明する。  
[Smith, O.J.M., *Chem. Eng. Prog.*, 53, 217(1957)]

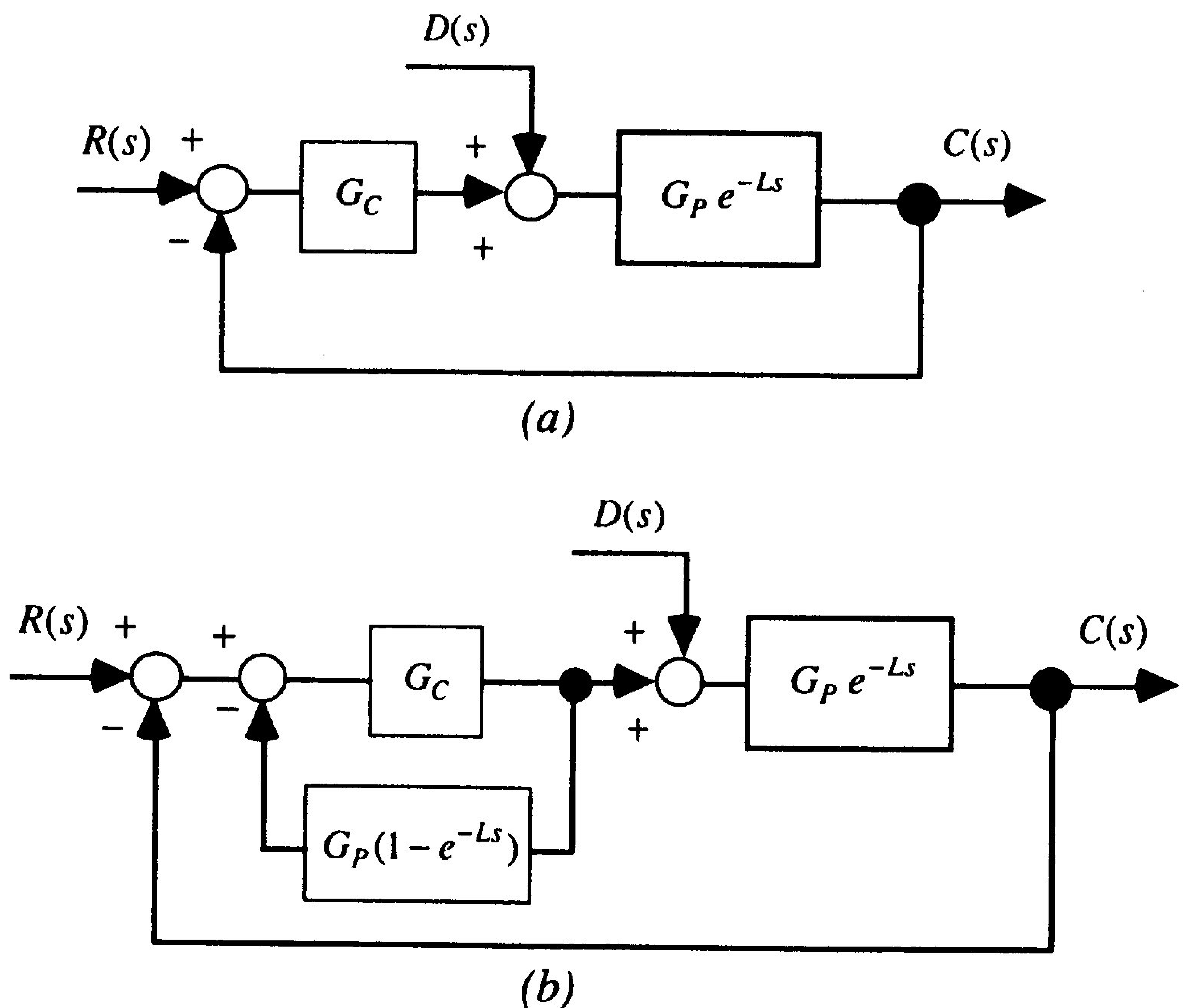


図 12・4 スミスのむだ時間補償型フィードバック制御系



図12・4(a)に、むだ時間を含むプロセスの通常のフィードバック制御系を示す。ここで、プロセスの動特性をむだ時間要素  $e^{-Ls}$  とそれ以外の要素  $G_P$  に分けて表記してある。また、操作端および検出端の伝達関数は省略してある。この制御系に対し、制御器の回りにネガティブ・フィードバックループを付け加え、このループの中に、制御補償要素を挿入する。補償要素として、図の(b)に示すような伝達関数を考える。この補償要素は、プロセスの一要素  $G_P$  とプロセス伝達関数  $G_P e^{-Ls}$  との差を取ったものである。プロセスの動特性が正しく同定されている場合には、このような補償要素を作成することが可能となる。

図12・4(b)において、設定値と制御変数との間の総括伝達関数を求める。

$$\frac{C(s)}{R(s)} = \frac{G_{OV,I} G_P e^{-Ls}}{1 + G_{OV,I} G_P e^{-Ls}} \quad (12\cdot9)$$

ここで、

$$G_{OV,I} = \frac{G_C}{1 + G_C G_P (1 - e^{-Ls})} \quad (12\cdot10)$$

(12・10)式を(12・9)式に代入して整理すると、

$$\frac{C(s)}{R(s)} = \frac{G_C G_P}{1 + G_C G_P} \cdot e^{-Ls} \quad (12\cdot11)$$

となる。

(12・11)式を基に、設定値と制御変数間の等価ブロック線図を描くと次のようになる。

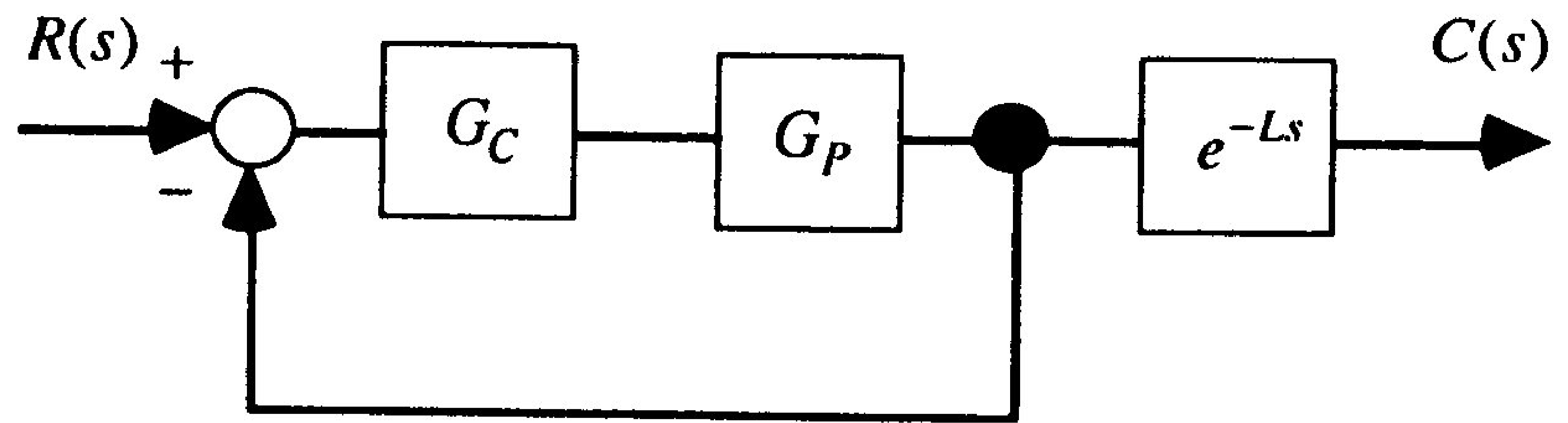


図 12・5 等価変換したむだ時間補償型制御系

図12・5から明らかのように、フィードバックループの中にむだ時間要素が入ってこない。したがって、制御系の安定性にむだ時間要素が関与しなくなり、制御系の安定性・制御性は大いに改善された。この制御系の特性方程式は、(12・11)式から次のようになり、この式の中にはむだ時間要素は含まれていない。

$$1 + G_C G_P = 0 \quad (12 \cdot 12)$$

外乱に対する制御性は、図12・4(b)から総括伝達関数を求めることによって評価できる。この場合にも、その特性方程式は(12・12)式と等しくなり、むだ時間要素が系の安定性に関与しないことがわかる。

スミスのむだ時間補償要素は、プロセスの動特性に基づいて構成される。したがって、プロセスの伝達関数の正確さが、補償要素の有効性に大きく拘わってくる。一般には、実際のプロセスの動特性とその伝達関数との間には、いくぶんかの誤差が含まれる。そこで、このことを考えに入れて、制御系の安定性に余裕をもたせて制御器パラメータを設定することが大切となる。