

9 制御器の特性

プロセス制御の目的は、制御量を目標値に一致させることである。そのため制御器では、目標値と検出値とを比較して偏差があれば、すなわち誤差信号があれば、これを入力としある制御動作を演算し、その結果を出力として発信する。基本的な制御動作は、比例、積分、微分動作でこの三動作を行う制御器・調節計をPID制御器と呼ぶ。プロセスはそれぞれ固有の動特性を示すが、制御器には汎用に作られたPID制御器を使用しても、比較的望ましい制御結果が得られる。

制御動作の演算を行うためにアナログの電気信号あるいは空気圧信号を用いる制御器もある。しかし現在では、デジタル信号を用いてマイクロプロセッサとメモリを使用して制御動作を演算するデジタルタイプのものが大部分となっている。

プロセス制御技術の発展には、ソフト技術である制御理論の進歩と、ハード面での計装技術の進歩が必要である。このハード面では、コンピュータ技術の発達によって大きな展開を見せ、現在では、**FA** (*factory automation*) や、**CIM** (*computer integrated manufacturing*) などと呼ばれる新技術が開発、使用されるようになってきた。

この章では、制御器の構成およびその機能について解説し、「最も望ましい制御」とは、について説明する。

9.1 制御器の構成

制御器の基本的な構成を図9・1に示す。なお、この図には検出部および操作部の構成についても併せて示した。

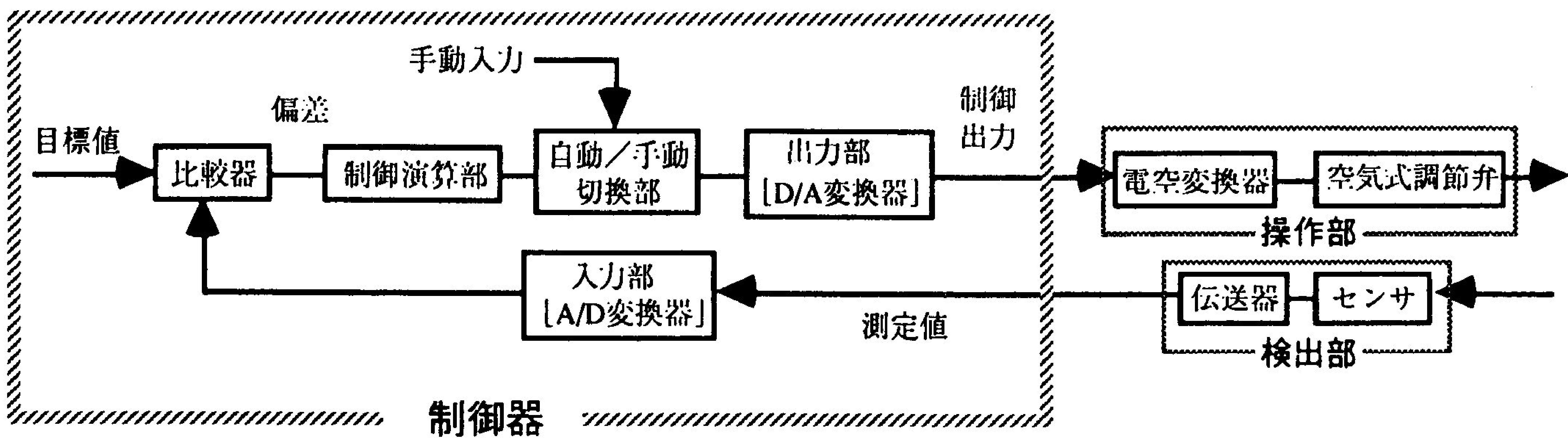


図9・1 制御器および検出部、操作部の構成

まず、プロセスの制御変数をセンサによって検出する。これを伝送器を通じて規格化した電気信号4～20mADC（または1～5VDC）に直し制御器に伝送する。この信号はアナログ信号（連続信号）で、これを制御器の入力部でデジタル信号（離散信号）に変換する。この機構はサンプラ (*sampler*) と呼ばれる。連続信号をサンプリングする周期は非常に短いので（0.05～0.2秒）得られたデジタル信号とアナログ信号との差異は無視できる。

この測定値と制御系の目標値とを比較して偏差信号を求め、これを基に制御動作の演算を行う。この計算はマイクロプロセッサ（コンピュータ）で行う。計算結果は、出力部にてアナログ信号に直し、操作部に送られる。

出力部のデジタル／アナログ変換器は、ホールド機構 (*hold element*) と呼ばれる。これは、デジタル信号として出力された値を1サンプリング周期の間保持している機構で、この一定に保持された信号がアナログ信号となる。出力の範囲（レンジ）は規格

化されていて、4~20mA DC（または1~5V DC）の値を取る。制御器から出た電気信号を空気圧の信号（0.2~1.0kgf/cm²）に変換する装置を電空変換器 (*electric - pneumatic transducer*) という。そして、この空気圧によって制御弁を駆動する。

計測，制御演算，制御操作を自動的に行うのが自動制御 (*automatic control*) であるが，この自動制御ループを切断し，手動にて制御操作を行うのが手動制御 (*manual control*) である。通常の制御器には，この自動/手動制御の切換部が設置されており，手動にて入力信号を加えることができるようになっている。

最も一般的な調節計（制御器）は，1入力・1出力タイプのもので，一つの制御系に1台使用される。これに対し，多数個の制御ループあるいは一つのプラント全体を一台のコンピュータ（バックアップのコンピュータは備えてある）で制御する方式が検討された。この方式を DDC (*Direct Digital Control*) という。しかし DDC の場合，コンピュータに故障が発生するとプラントの安全性が大きく損なわれ，危険度が非常に大きい。そこで，1台のコンピュータでは比較的小さい数の制御ループ（中・大規模の計装用では，8, 16, 40 あるいは80ループ）を扱う分散形制御システム (*distributed control system: DCS*) が採用されるようになった。このシステムでは，CRT（ブラウン管使用のモニター）と入力装置を使用して監視・操作が行える。ただし，個々の制御機能は，通常の1入力・1出力タイプのものと大差はない。

9. 2 制御器の機能

図 9・1に示した制御演算部の働き，すなわち，制御偏差を入力として，この偏差を打ち消すための制御出力を算出する方法について説明する．最も基本的な制御動作は，偏差信号の大きさに比例した訂正動作信号を出す方式で，比例制御 (*proportional control, P-control*) と呼ばれている．

$$p(t) = K_C \varepsilon(t) + p_S \quad (9 \cdot 1)$$

ここで， $\varepsilon(t)$ は入力の偏差信号， $p(t)$ は制御出力信号で， p_S はバイアス信号で偏差がゼロのときの制御器の出力信号である． K_C は比例ゲイン (*proportional gain*) あるいは比例感度 (*proportional sensitivity*) と呼ばれる定数である．

いま，一例としてガスバーナで加熱する加熱炉の制御について考える．この制御系では，炉内の温度を測ってバーナに供給するガス量を制御する．温度の計測範囲（入力レンジ）は $0 \sim 500^\circ\text{C}$ で，炉温度の設定値は 250°C とする．バーナのガス量を制御する制御弁は，*air to close* 弁で制御器の出力が $0 \sim 100\%$ 変化して弁が開閉するものとする．また，定常状態での炉内温度 250°C （設定値）では，制御弁が半分（ 50% ）開いているものとする．比例制御を行い制御器の比例ゲインが $1.67 (= 100/60)$ とする．これらの関係を図にて示すと，図 9・2 のようになる．図中の黒丸点が定常状態の値で，図中の太線が操作線となる．入力 $\varepsilon(t)$ が $\pm 30\%$ 変化すると出力が 100% 変化する．したがって，入・出力の関係は次式のようなになる．

$$p(t) = 1.67 \varepsilon(t) + 0.5 \quad (9 \cdot 2)$$

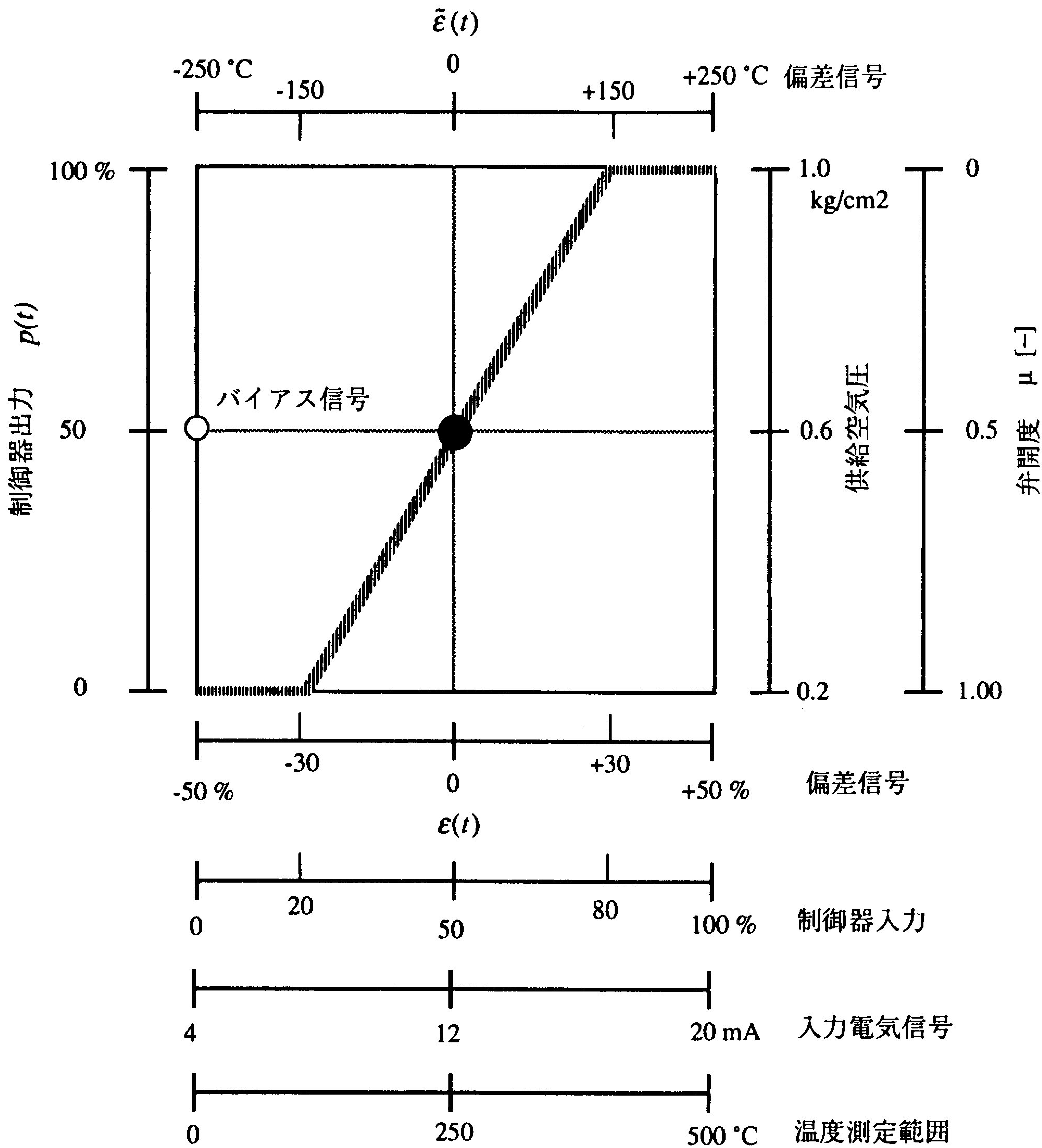


図 9・2 比例制御器の制御動作

図9・2の横軸下部には、温度センサのレンジ、伝送器の出力、制御器の入力範囲および偏差信号を示した。縦軸の左側には制御器の出力範囲を示し、右側に電空変換器の出力範囲、制御弁の開度を示した。偏差信号として温度の単位℃を付けた変数 $\tilde{e}(t)$ を用いると、横軸上部のようになる。

比例制御器の比例ゲインを、単位を取り込んだ入力と出力の比で表しているテキストもある。本例では、入力を $\tilde{e}(t)$ [℃]で、出力を空気圧 $\tilde{p}(t)$ [kgf/cm²]で表すと、 $K_C = 0.8/300[(\text{kgf/cm}^2)/^\circ\text{C}]$ となる。

比例ゲインの代わりに、比例帯 (*proportional band* : PB) を用いて、比例制御の動作を表すことがある。これは、比例ゲインの逆数をパーセントにて表したもので、制御器の出力を100%変化させるために必要な入力の範囲を%にて表したものである。

$$PB = (1/K_C) \times 100 \quad (9\cdot3)$$

図9・2の場合、出力が0~100%変化するのに入力が±30%変化する必要がある。したがって比例帯は60%となる。通常のP-制御では、 $0 \leq PB \leq 500\%$ の範囲である。

比例帯が0%すなわち比例ゲインが無限大の特殊な制御方式がある。この方式では、偏差信号が(-)であれば制御出力はゼロで、偏差が(+)であれば出力は100%となる。この制御方式をオンオフ制御 (*on-off control*) あるいは二位置制御 (*two position control*) という。実際の制御では、偏差が $+d\%$ 以上で出力100%、偏差が $-d\%$ 以下で出力0%となるように、制御出力の切換えに動作すきま (*differential gap*) $\pm d\%$ を持たせるようにする。この様子を図9・3に示す。

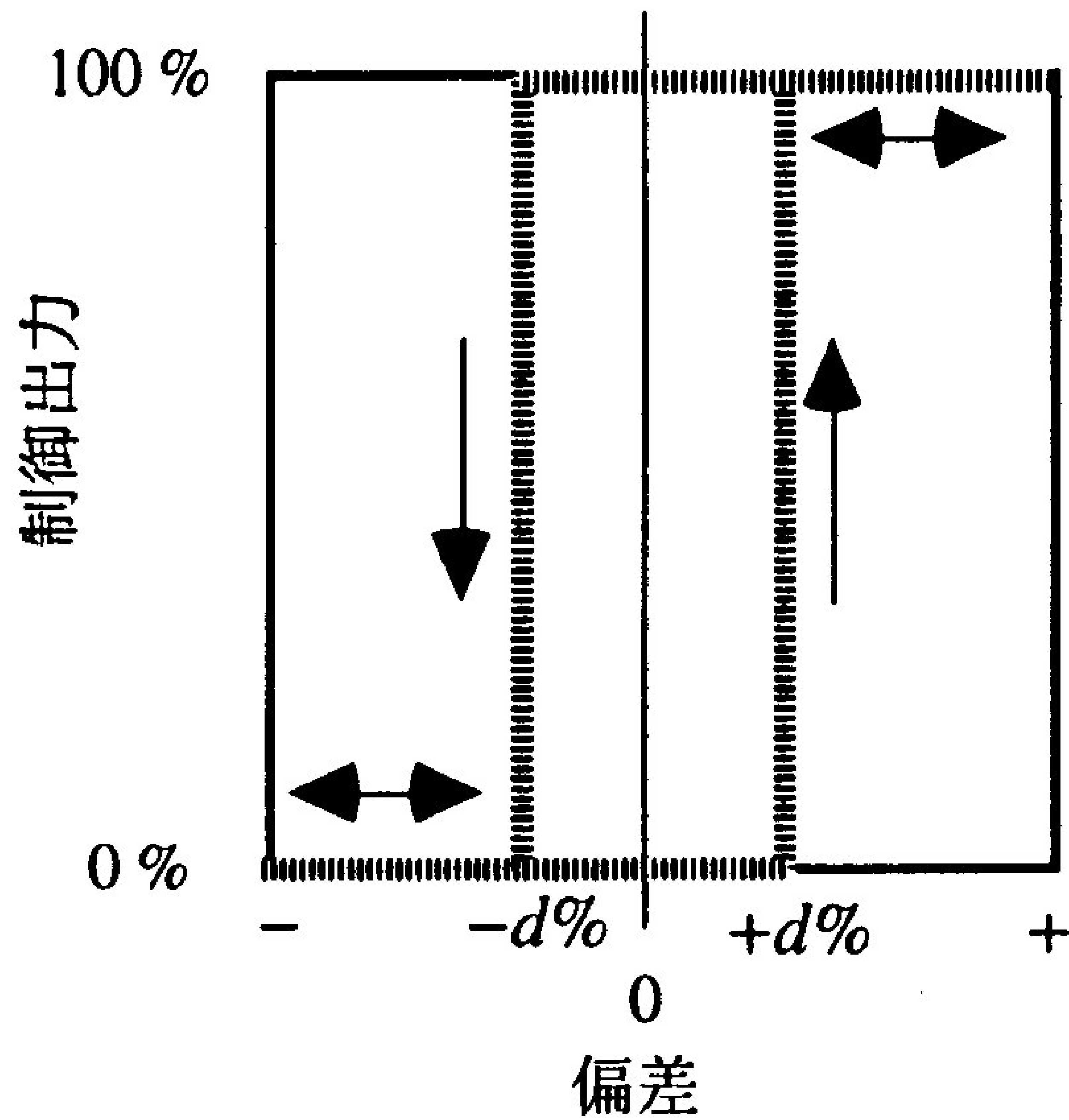


図 9.3 動作すきまのあるオンオフ制御

オンオフ制御は、比較的単純な制御方式であるため広く用いられている。この方式では、制御弁は常に全開か全閉のいずれかになっている。いま、弁を全開にして制御変数の測定値が上昇していくプロセスを考えると、偏差が $-d\%$ になったとき弁が閉じる。しかし、弁が閉じても制御変数は直ちに下降はしない。プロセスの応答に遅れがあるためである。偏差が $-d\%$ より大きくなってしばらくしてから、制御変数は減少し始め偏差も小さくなる。同様に、偏差が $+d\%$ を越す場合にも、制御変数の行過ぎ現象が起こる。その結果、制御変数は、設定値からの偏差が $\pm d\%$ より更に大きい行過ぎ量を示しながら振動する。この現象をサイクリング (cycling) あるいはハンチング* (hunting) と呼ぶ

* このましくないサイクリングをハンチングという。特に、制御系が不安定となったときに起きる振動をいう。

比例制御は制御動作の基本で、オンオフ制御は比例ゲインが無限大の場合に相当する。制御系に、ある外乱（例えばステップ状の外乱）が入ってきた場合を想定すると、比例制御のみでこの外乱の影

響を完全に取り除くことはできない。制御器の出力は、外乱によって生じた偏差を基にして(9・1)式で算出され、この出力によって偏差がゼロとなるよう打ち消し動作を行う。しかし、偏差がゼロとなると $p(t) = p_s$ となる。これは外乱の挿入がない定常状態での制御器出力であり、この出力では外乱の影響を消去できない。したがって、制御系はある偏差を持った状態で収束し定常となる。このように定常状態になって制御変数が一定値を示したとき、その値と設定値との間に生じた定常偏差をオフセット (*offset*) と呼ぶ。比例制御のみでは、このオフセットを取り除くことはできない。オフセットが生じた場合、制御器のバイアス信号 p_s を修正すればこの定常偏差を取り除くことができる。オフセットの大きさを見ながら、この操作を人が行うことを手動リセットという。

リセットの操作を自動的に行う方法が積分制御 (*integral control, I-control*) である。通常、積分動作は比例制御と組み合わせて、PI-制御として用いられる。その制御動作は次式で与えられる。

$$p(t) = K_C \left\{ \varepsilon(t) + \frac{1}{T_I} \int_0^t \varepsilon(t) dt \right\} + p_s \quad (9 \cdot 4)$$

上式右辺の第二項が積分動作を表す。ここで、 T_I は積分時間 (*integral time*) と呼ばれ時間の次元を持つ定数である。積分動作を大きく働かせたいときには、積分時間を小さくすればよい。積分時間が十分に大きければ、(9・4)式の右辺第二項は無視されP-制御となる。通常プロセス制御では、積分時間は $0.1 \leq T_I \leq 50 \text{ min}$ の範囲にセットされる。

積分時間は次のような物理的意味を持つ。いま $\varepsilon(t)$ が一定値 ε_s となっているとすれば、(9・4)式の右辺第一項は $K_C \varepsilon_s$ となり、第二項は $K_C (1/T_I) \varepsilon_s \cdot t$ となる。そこで、 $t = T_I$ のとき第二項と第一項との値は等しくなる。すなわち、積分時間 T_I は、積分動作の出力が比例動作の出力と

等しくなるまでの時間を表し，リセット時間(reset time)と呼ばれることもある。積分時間の逆数($1/T_I$)をリセット率(reset rate)という。

比例制御に積分動作を付け加えると，制御器出力は偏差が存在する限り必ず大きくなっていく。そのため，オフセットを取り除くことが可能となる。一方，偏差を取り除くのに時間がかかる場合や積分動作をかなり強く働かせすぎた場合には，制御器出力が100%あるいは0%（飽和状態）となってしまう。このような状態を積分ワインドアップ(integral windup)あるいはリセットワインドアップ(reset windup)という。この現象は，手動制御から自動制御に切り替えたとき，回分プロセスを運転開始したとき，あるいは設定値を大きく変更した場合に起こる。ワインドアップが起こると，制御変数が設定値から外れて大きなオーバーシュートが現れ，また正常な制御動作に移るまでに時間がかかる。マイクロプロセッサを用いたデジタル制御器では，このワインドアップの影響を取り除くよう工夫されている。

制御偏差が増加あるいは減少する傾向にあれば，その偏差が検出された後に訂正動作を行うよりも，偏差の増加・減少割合を求め，その値からこれから生じる偏差を予想して訂正動作を起こすことが考えられる。通常，人間が行う制御では，上記のようにな動作がされている。この制御動作が微分制御* (derivative control, D-control)である。D-制御は，比例制御あるいは比例-積分制御と共に，PD-制御，PID-制御として用いられる。PID-制御器の制御動作を次式に示す。

* 微分制御を anticipatory control (予測制御) と呼ぶことがある。

$$p(t) = K_C \left\{ \varepsilon(t) + \frac{1}{T_I} \int_0^t \varepsilon(t) dt + T_D \frac{d\varepsilon(t)}{dt} \right\} + p_S \quad (9.5)$$

ここで， T_D は微分時間(derivative time) あるいはレートタイム

(rate time) と呼ばれ，時間の次元を持つ． T_D の値として， $T_D = (0.1 \sim 0.5) \times T_I$ の範囲にセットされる．

微分動作は未来の偏差を予測しながら制御を行うため，制御性がより向上することが期待できる．しかし，プロセス変数の測定値がノイズを含むとき，偏差の平均値はほとんどゼロでも，その微係数は時間的に激しく変動する．そのため，微分動作による制御出力も激しく変動する．また，設定値をステップ状に変更した場合，微分動作の出力が瞬時ではあるが非常に大きくなる（*derivative kick* という）．デジタル制御器ではこれらの影響を取り除くよう工夫されている．

いま，(9.5)式を次のように書き直す．

$$\begin{aligned} \Delta p(t) &= p(t) - p_S \\ &= K_C \left\{ \varepsilon(t) + \frac{1}{T_I} \int_0^t \varepsilon(t) dt + T_D \frac{d\varepsilon(t)}{dt} \right\} \end{aligned} \quad (9 \cdot 10)$$

ここで， $\Delta p(t)$ は制御器出力の変動を表す．上式をラプラス変換する．

$$\begin{aligned} P(s) &= \mathcal{L} \{ \Delta p(t) \} = \mathcal{L} \left[K_C \left\{ \varepsilon(t) + \frac{1}{T_I} \int_0^t \varepsilon(t) dt + T_D \frac{d\varepsilon(t)}{dt} \right\} \right] \\ &= K_C \left\{ E(s) + \frac{1}{T_I} \frac{E(s)}{s} + T_D s E(s) \right\} \end{aligned}$$

偏差入力 $E(s)$ と制御出力 $P(s)$ との比を取ると，これが制御器の伝達関数となる．

$$G_C(s) = \frac{P(s)}{E(s)} = K_C \left(1 + \frac{1}{T_I s} + T_D s \right) \quad (9 \cdot 11)$$

制御器の伝達関数には三つのパラメータ，比例ゲイン，積分時間および微分時間が含まれる。これらの値をどのように設定するか，という問題が制御系の設計問題の一つである。これらのパラメータの最適値は取り扱うプロセスによって異なる。そこで，プロセスの動特性と制御器の最適パラメータとの間にある定まった関係式が成立すればよいことになる。しかし，一般的には，この関係は決まらない。その理由は，「最も望ましい制御」という要求が，それぞれのプロセスによって異なるからである。

* 蒸留塔の底部から抜き出した液を，伝熱媒体によって加熱し，蒸気を発生させてる蒸発装置

** 蒸留塔の塔頂から出た蒸気を凝縮し，その液を塔頂へ再び戻すのがリフレックス（還流）である。この液を溜めておくタンクをリフレックスドラムという。

例えば，タンクの液面制御について考えると，蒸留塔のリボイラー*の液面制御であれば，発生する蒸気量が一定となるようにするため，液面を常に一定に保つことが望ましい。しかし，蒸留塔のリフレックスドラム**の液面制御では，液面がある範囲で変動し，タンクが空または満杯にならない範囲内におさまるよう制御する。これは，このタンクがバッファータンクの役目をはたし，一定流量の留出液およびリフレックスを取り出すのに必要なためである。他の例として，発熱量の大きな化学反応器の温度制御を考える。ここで，温度設定値を変更する場合，反応器の操作安定性のために，温度の追従性はいくぶん遅くとも，大きなオーバーシュートが絶対生じないような制御方法が望まれる。一方，熱交換器の温度制御では，設定値変動に対しその追従性が高いことが望ましい。

制御器のパラメータを決定する方法としては，プロセスの動特性が伝達関数の形で与えられている場合には，制御系の過渡応答および周波数応答を基に行う。プロセスの動特性が事前に求まっていない場合には，実験的手法によってプロセスの動特性を推定し，制御器のパラメータをチューニングする。その詳細は，以後の章にて説明する。