10-proportional-integral-derivative

goal: techniques for analysis, design, & implementation of the most ubiquitous control architecture

Based on a survey of over eleven thousand controllers in the refining, chemicals and pulp and paper industries, 97% of regulatory controllers utilize a PID feedback control algorithm.

L. Desborough and R. Miller, 2002 [DM02a].

1°. essentials of feedback control [AM\u2 Ch 11] [Nu7 Ch 9.4]
  1¹. a simple controller
  1². implementation issues

1°. essentials of feedback control

- let's take a step back and reflect on (a) what feedback controllers do and (b) how they do it

(a) starting in the first week of lecture & extending to last week, we've studied how feedback can be used to:
  - make output (y) track reference (r)
  - reject disturbances to

inputs (v) and outputs (w)

- provide robustness to model inaccuracies, e.g. unmodeled dynamics or uncertain parameters

(b) feedback controllers can use

- prior knowledge, e.g. of process dynamics, disturbance statistics, or model uncertainty
- the measured signal y and commanded signals u, r, including their time histories and (predictions about their) futures

ex: the full-state observer/controller processes the time history of the output and predicts the future:

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - \hat{y})$$
$$\hat{y} = C\hat{x} + Du$$

- requires substantial prior knowledge (i.e. A, B, C, D), and is only reliable if uncertainty is low

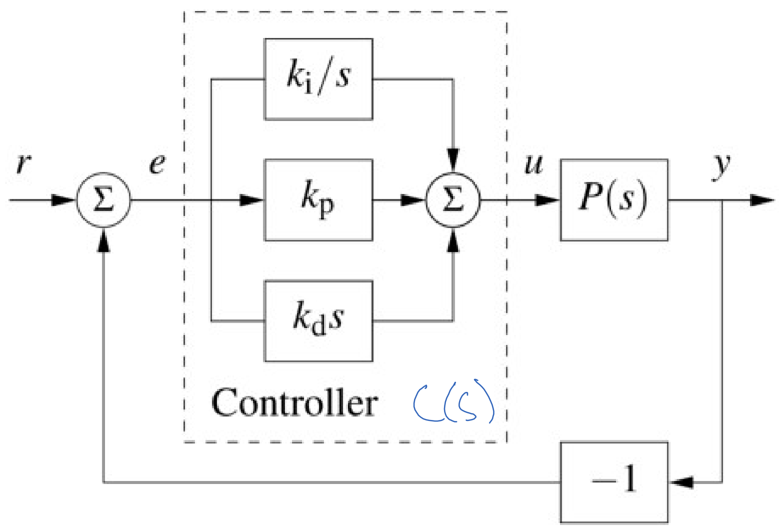
---

1'. a simple controller

- at another extreme of complexity, consider the simplest controller, obtained using a linear combination of: P – present error

I – (integral of) past error

D – (prediction of) future error (using linear extrapolation)



– as a transfer function:

$$C(s) = k_P + \frac{k_I}{s} + k_D s$$

– as a function of time:

$$u(t) = k_P e(t) + k_I \int_0^t e(\tau)\,d\tau + k_D \frac{d}{dt} e(t)$$

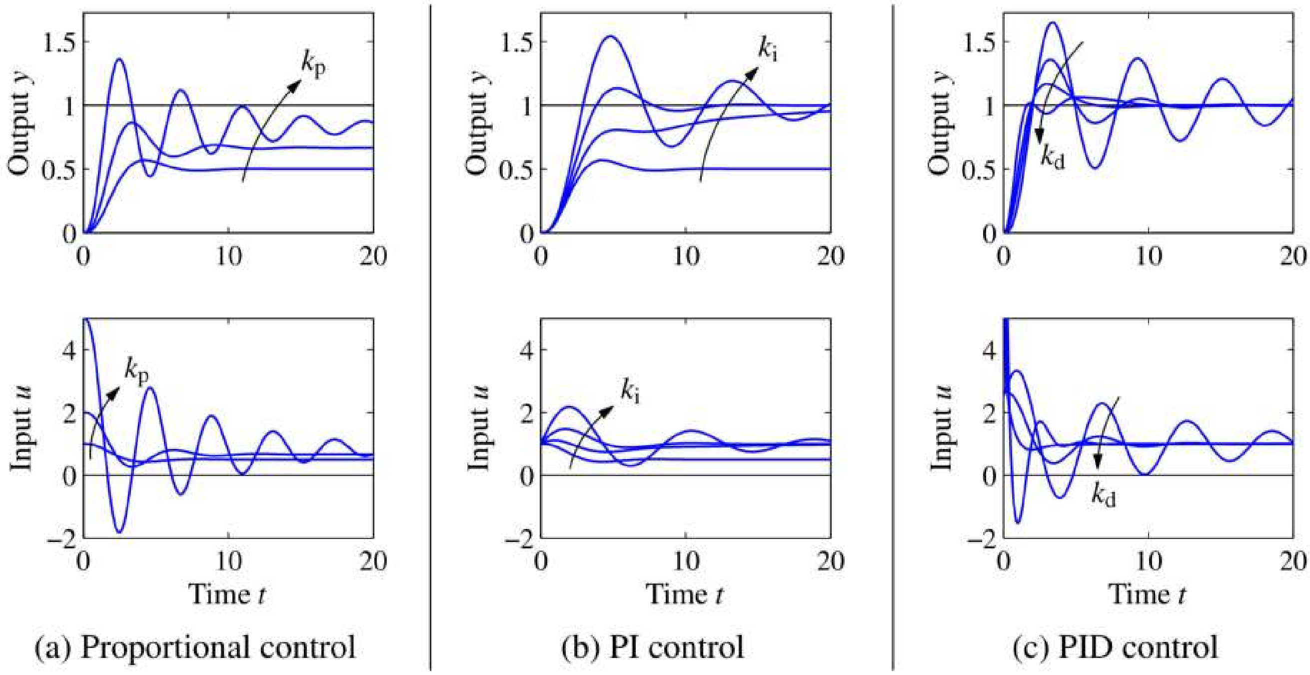


Figure 11.2: Responses to step changes in the reference value for a system with a proportional controller (a), PI controller (b) and PID controller (c). The process has the transfer function $P(s) = 1/(s+1)^3$, the proportional controller has parameters $k_p = 1$, 2 and 5, the PI controller has parameters $k_p = 1$, $k_i = 0$, 0.2, 0.5, and 1, and the PID controller has parameters $k_p = 2.5$, $k_i = 1.5$ and $k_d = 0$, 1, 2, and 4.

→ describe trends in transient & steady-state error as k_P, k_I, k_D vary

- integral feedback eliminates steady-state error (assuming stable closed-loop system)

claim: if $\lim_{t\to\infty} u(t) = \bar{u}$, $\lim_{t\to\infty} e(t) = \bar{e}$, then $\bar{e} = 0$

→ prove this claim using the expression for $u(t)$

→ prove this claim using the transfer function $C(s)$

- derivative feedback predicts future error
  - proportional + derivative terms are

$$k_P e(t) + k_D \frac{d}{dt} e(t) = k_P \left(e(t) + \Delta \dot{e}(t)\right) \Big\} \; \Delta = \frac{k_D}{k_P}$$
$$\simeq k_P\, e(t+\Delta)$$

Q: why is PID ubiquitous?

A: it's (often) good enough

* if there are ≤ 2 unstable poles, (e.g. if the system is already stable) then PID is a simple, intuitive way to shape closed-loop performance

---

1.2. implementation issues

* we'll sample a few issues – read [AMv2 Ch 11.5] for more!

- we have an elegant formula for

$$u(t) = k_P e(t) \qquad \text{proportional}$$
$$+ k_I \int_0^t e(\tau)\, d\tau \qquad \text{integral}$$
$$+ k_D \frac{d}{dt} e(t) \qquad \text{derivative}$$

control, with nice analytical properties

→ what practical issues could arise when this formula is implemented on actual hardware, eg: digital microcontroller;

analogue RLC/op-amp circuit?

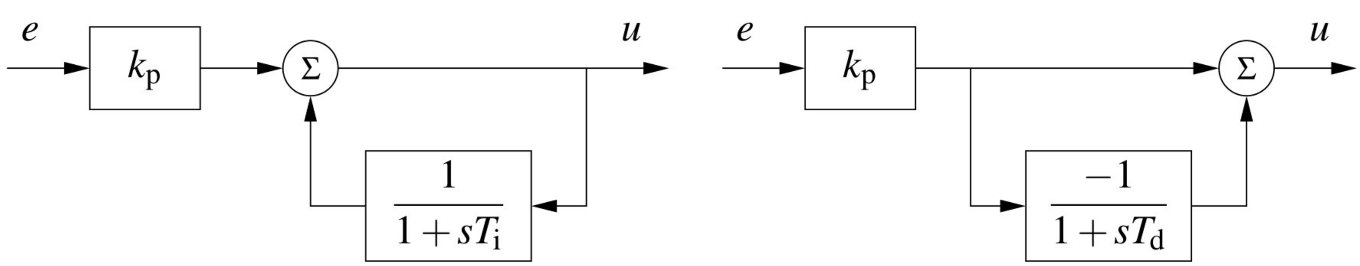
(i) integral term is unbounded, derivative term amplifies noise

* can become large, saturate actuators, cause instability

(ii) if we chose PID control to avoid relying on a model, how do we choose gains k_P, k_I, k_D?

(i) practically speaking, implementing actual integrator/differentiator is a bad idea

– can instead use low-pass filters in feedback or feedforward



(a) Integral action (automatic reset)

(b) Derivative action

Figure 11.3: Implementation of integral and derivative action. The block diagram in (a) shows how integral action is implemented using *positive feedback* with a first-order system, sometimes called automatic reset. The block diagram in (b) shows how derivative action can be implemented by taking differences between a static system and a first-order system.

→ compute transfer function G_{ue} for (a), (b)

(a) G_{ue}: $u = k_p e + \frac{1}{1+sT_I} u$

$$\Leftrightarrow u\left(1 - \frac{1}{1+sT_I}\right) = k_p e$$

$$\Leftrightarrow u\left(\frac{sT_I}{1+sT_I}\right) = k_p e$$

$$\Leftrightarrow u = \underbrace{k_p\left(1 + \frac{1}{sT_I}\right)}_{= G_{ue}} e$$

$$= k_p e + \frac{k_I}{s} e, \quad k_I = \frac{k_p}{T_I}$$

(b) G_{ue}: $u = k_p e - \frac{k_p}{1+sT_D} e$

$$= k_p\left(1 - \frac{1}{1+sT_D}\right) e$$

$$= \underbrace{k_p\left(\frac{sT_D}{1+sT_D}\right)}_{= G_{ue}} e$$

note: implementation in (a) via low-pass filter feedback yields integral control, but implementation in (b) via low-pass filter feedforward yields filtered derivative control, i.e. derivative control based on filtered error signal

(ii) to tune PID controller gains:

- you can use a model for the system, e.g.
  - write down the DE and estimate free parameters, compute controller gains
  - empirically estimate transfer function's Bode plot, apply frequency-domain design methods

OR:

- tune the gains until it works!

- the first (and still most widely-known) rules were developed by Zeigler & Nichols in the 1940s
  - heuristic rules hand-designed to work on systems Z & N had access to
  - assumes (i.e. is only guaranteed to work) for systems with transfer function

    $\frac{e^{-sT}}{(a+s)}$, i.e. one pole at $-a$ & delay of duration T
  - a critical step leverages the Nyquist plot & stability criterion:

    1. set $k_I, k_D = 0$

2. increase k_p until system oscillates at gain k_c w/ period T_c
3. Nyquist implies loop transfer function $L = k_c P(s)$ passes through critical point $-1 \in \mathbb{C}$ at frequency $\omega_c = \frac{2\pi}{T_c}$

* this knowledge of Nyquist plot is used to select controller gains:

Type	k_p	T_i	T_d
P	$0.5k_c$		
PI	$0.4k_c$	$0.8T_c$	
PID	$0.6k_c$	$0.5T_c$	$0.125T_c$