Lecture 4 - PID Control

- 90% (or more) of control loops in industry are PID
- Simple control design model $\rightarrow$ simple controller
**P control**

- Integrator plant:
  \[
  \dot{y} = u + d
  \]

- P controller:
  \[
  u = -k_p (y - y_d)
  \]

**Example:**
Utilization control in a video server

Video stream \(i\)
- processing time \(c[i]\), period \(p[i]\)
- CPU utilization: \(U[i] = c[i]/p[i]\)

\[
\begin{align*}
  &u(t) \quad \text{admission rate} = \frac{\Delta U_{new}}{\Delta t} \\
  &y_d(t) \quad \text{server utilization} \\
  &y(t) \quad \text{completion rate} = \frac{\Delta U_{done}}{\Delta t} \\
  &-d(t)
\end{align*}
\]
P control

- Closed-loop dynamics
  \[ \dot{y} + k_P y = k_p y_d + d \]
  \[ y = \frac{k_p}{s + k_p} y_d + \frac{1}{s + k_p} d \]

- Steady-state \((s = 0)\)
  \[ y_{SS} = y_d + \frac{1}{k_p} d_{ss} \]

- Transient
  \[ y(t) = y(0)e^{-t/T} + \left( y_d + \frac{1}{k_p} d_{ss} \right) \cdot \left( 1 - e^{-t/T} \right) \]
  \[ T = 1/k_p \]

- Frequency-domain (bandwidth)
  \[ y_d(t) = \hat{y}_d(i\omega)e^{i\alpha} \]
  \[ d(t) = \hat{d}(i\omega)e^{i\alpha} \]
  \[ |\hat{y}(i\omega)| = \sqrt{\left( \frac{\hat{y}_d(i\omega) + \hat{d}(i\omega)/k_p}{\sqrt{(\omega/k_p)^2 + 1}} \right) \}} \]
I control

\[ y = g \cdot u + d, \]

- Introduce integrator into control
  \[ \dot{u} = v, \]
  \[ v = -k_I (y - y_d) \]
- Closed-loop dynamics
  \[ y = \frac{gk_I}{s + gk_I} y_d + \frac{s}{s + gk_I} d \]

Example:
- Servosystem command
- More:
  - stepper motor
  - flow through a valve
  - motor torque ...
Sampled time I control

- Step to step update:
  \[ y(t) = g \cdot u(t) + d(t) \]
  \[ u(t) = u(t - 1) + v(t - 1) \]
  \[ v(t) = k_I [y(t) - y_d] \]
- Closed-loop dynamics
  \[ y = g \cdot u + d \]
  \[ u = \frac{k_I}{z - 1} [y - y_d] \]

- Deadbeat control: \( g k_I = 1 \)
  \[ y = z^{-1} y_d + (1 - z^{-1})d \]
Run-to-run (R2R) control

- Main APC (Advanced Process Control) approach in semiconductor processes
- Modification of a product recipe between tool "runs"
- Processes:
  - vapor phase epitaxy
  - lithography
  - chemical mechanical planarization (CMP)
  - plasma etch

\[ y(t) = g \cdot u(t) + d(t) \]
\[ u(t) = u(t - 1) + k_1 [y(t) - y_d] \]

Run-to-run control
Cell controller
Metrology
system
Tool
Runtime controller
Process
Tunable recipe parameters
PI control

- First-order system:
  \[ \tau \dot{y} = -y + u + d \]
- P control + integrator for cancelling steady state error
  \[
  e = y - y_d; \\
  \dot{v} = e \\
  u = -k_I v - k_P e
  \]

Example:
- WDM laser-diode temperature control
  \[ \tau \dot{y} = -y + u + d \]
- Other applications
  - ATE
  - EDFA optical amplifiers
  - Fiber optic laser modules
  - Fiber optic network equipment

\[ y(t) = \text{temperature - ambient temperature} \]

heat capacity

heat loss to environment

pumped heat

produced heat
PI control

- P Control + Integrator for cancelling steady state error

\[ e = y - y_d \; ; \]
\[ \dot{v} = e \]
\[ u = -k_I v - k_P e = k_P (e - k_i v) \]

- Velocity form of the controller

\[ \dot{u} = -k_I e - k_P \dot{e} \]
\[ u(t + 1) = u(t) - k_I e(t) - k_P [e(t) - e(t - 1)] \]
PI control

- Closed-loop dynamics
  \[ y = \frac{sk_p + k_i}{s(\tau s + 1) + sk_p + k_i} y_d + \frac{s}{s(\tau s + 1) + sk_p + k_i} d \]

- Steady state \((s = 0)\): \(y_{ss} = y_d\).
  No steady-state error!

- Transient dynamics: look at the characteristic equation
  \[ \tau \lambda^2 + (1+k_p) \lambda + k_i = 0 \]

- Disturbance rejection
  \[ |\hat{y}(i\omega)| = |H_d(i\omega)| \cdot |\hat{d}(i\omega)| \]
PLL Example

- Phase-locked loop is arguably a most prolific feedback system

\[ e = 2K_m \text{LPF} \langle r \times v \rangle \]
\[ = 2K_m \text{LPF} \left\langle A \sin(\omega t + \theta_d) \times \cos(\omega_o t + \theta_o) \right\rangle \]
\[ e \approx AK_m \sin(\omega t - \omega_o t + \theta_d - \theta_o) \]
\[ \dot{\theta}_o = \Delta \omega_o = K_o u \]
PLL Loop Model

• Small-signal model:

\[ \theta = \omega t - \omega_0 t + \theta_d - \theta_o \ll 1 \]

\[ e = K_d \sin(\theta) \approx K_d \theta \]

\[ \dot{\theta} = \omega - \omega_0 + \dot{\theta}_d - K_o u \]

• Loop dynamics:

\[ \ddot{\theta} = d - K_o u \]

\[ e = K_d \theta \]

\[ u = k_p e + k_i \int e \cdot dt \]

\[ \theta = \frac{s}{s^2 + K_o K_d k_p s + k_i} \frac{d}{d} \]
PD control

- 2nd order dynamics
  \[ \ddot{y} = u + d \]

- PD control
  \[ e = y - y_d \]
  \[ u = -k_D \dot{e} - k_P e \]

- Closed-loop dynamics
  \[ \ddot{e} + k_D \dot{e} + k_P e = d \]
  \[ e = \frac{1}{s^2 + k_D s + k_P} d \]

- Optimal gains (critical damping)
  \[ k_D = 2\tau; \quad k_P = \tau^2 \]

Example:
- Disk read-write control

\[ J\ddot{\phi} = T_{VCM} + T_{DISTURB} \]

Voice Coil Motor
PD control

• Derivative (rate of $e$) can be obtained
  – speed sensor (tachometer)
  – low-level estimation logic

• Signal differentiation
  – is noncausal
  – amplifies high-frequency noise

• Causal (low-pass filtered) estimate of the derivative

$$\dot{e} \approx \frac{s}{\tau_D s + 1} e = \frac{1}{\tau_D} e + \frac{1/\tau_D}{\tau_D s + 1} e$$

• Modified PD controller:

$$u = -k_D \frac{s}{\tau_D s + 1} e - k_p e$$
PD control performance

• The performance seems to be infinitely improving for
  \[ k_D = 2\tau; \quad k_p = \tau^2; \quad \tau \rightarrow \infty \]

• This was a simple design model, remember?

• Performance is limited by
  – system being different from the model
    • flexible modes, friction, VCM inductance
  – sampling in a digital controller
  – rate estimation would amplify noise if too aggressive
  – actuator saturation
  – you might really find after you have tried to push the performance

• If high performance is really that important, careful application of more advanced control approaches might help
Plant Type

- Constant gain - I control
- Integrator - P control
- Double integrator - PD control
- Generic second order dynamics - PID control
PID Control

- Generalization of P, PI, PD
- Early motivation: control of first order processes with deadtime

$$y = \frac{ge^{-T_Ds}}{\tau s + 1} u$$

Example:
- Paper machine control
PID Control

- PID: three-term control
  
  \[
e = y - y_d
  \]
  
  \[
u = -k_D \dot{e} - k_P e - k_I \int e \cdot dt
  \]

- Sampled-time PID
  
  \[
u = -k_D (1 - z^{-1}) e + k_P e + k_I \frac{1}{1 - z^{-1}} e
  \]

- Velocity form
  
  - bumpless transfer between manual and automatic

  \[
  \Delta u = -k_D \Delta^2 e - k_P \Delta e - k_I e
  \]

  \[\Delta = 1 - z^{-1}\]

  \[
u(t + 1) = u(t) - k_I e(t) - k_P \left( e(t) - e(t - 1) \right)
  \]

  \[- k_D \left( e(t) - 2e(t - 1) + e(t - 2) \right)\]
Tuning PID Control

- Model-based tuning
- Look at the closed-loop poles
- Numerical optimization
  - For given parameters run a sim, compute performance parameters and a performance index
  - Optimize the performance index over the three PID gains using grid search or Nelder method.
Zeigler-Nichols tuning rule

- Explore the plant:
  - set the plant under P control and start increasing the gain till the loop oscillates
  - note the critical gain $k_C$ and oscillation period $T_C$

- Tune the controller:

<table>
<thead>
<tr>
<th></th>
<th>$k_P$</th>
<th>$k_I$</th>
<th>$k_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>$0.5k_C$</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>PI</td>
<td>$0.45k_C$</td>
<td>$1.2k_P/T_C$</td>
<td>—</td>
</tr>
<tr>
<td>PID</td>
<td>$0.5k_C$</td>
<td>$2k_P/T_C$</td>
<td>$k_P T_C/8$</td>
</tr>
</tbody>
</table>

- Z and N used a Monte Carlo method to develop the rule
- Z-N rule enables tuning if a model and a computer are both unavailable, only the controller and the plant are.
Integrator anti wind-up

- In practice, control authority is always limited:
  \[ u_{\text{MIN}} \leq u(t) \leq u_{\text{MAX}} \]
- Wind up of the integrator:
  - if \[ |u_c| > u_{\text{MAX}} \] the integral \[ \nu \] will keep growing while the control is constant. This results in a heavy overshoot later
- Anti wind-up:
  - switch the integrator off if the control has saturated

\[
\begin{align*}
\dot{v} &= e \\
u_c &= -k_l \nu - k_p e \\
u &= \begin{cases} 
    u_{\text{MAX}}, & u_c > u_{\text{MAX}} \\
u_c, & u_{\text{MIN}} \leq u_c \leq u_{\text{MAX}} \\
u_{\text{MIN}}, & u_c < u_{\text{MIN}}
\end{cases}
\end{align*}
\]

\[
\dot{v} = \begin{cases} 
    e, & \text{for } u_{\text{MIN}} \leq u_c \leq u_{\text{MAX}} \\
0, & \text{if } u_c > u_{\text{MAX}} \text{ or } u_c < u_{\text{MIN}}
\end{cases}
\]
Industrial PID Controller

- A box, not an algorithm
- Auto-tuning functionality:
  - pre-tune
  - self-tune
- Manual/cascade mode switch
- Bumpless transfer between different modes, setpoint ramp
- Loop alarms
- Networked or serial port