Introduction to Digital Control & Stability of Digital Systems

ELEC 3004: Systems: Signals & Controls
Dr. Surya Singh
(Some material adapted from Paul Pounds)

Lecture 19

elec3004@itee.uq.edu.au
http://robotics.itee.uq.edu.au/~elec3004/
May 10, 2013

Today...

<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Lecture Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>27-Feb</td>
<td>Introduction</td>
</tr>
<tr>
<td>1</td>
<td>1-Mar</td>
<td>Systems Overview</td>
</tr>
<tr>
<td>2</td>
<td>6-Mar</td>
<td>Signals &amp; Signal Models</td>
</tr>
<tr>
<td>2</td>
<td>8-Mar</td>
<td>System Models</td>
</tr>
<tr>
<td>3</td>
<td>13-Mar</td>
<td>Linear Dynamical Systems</td>
</tr>
<tr>
<td>3</td>
<td>15-Mar</td>
<td>Sampling &amp; Data Acquisition</td>
</tr>
<tr>
<td>4</td>
<td>20-Mar</td>
<td>Time Domain Analysis of Continuous Time Systems</td>
</tr>
<tr>
<td>5</td>
<td>22-Mar</td>
<td>System Behaviour &amp; Stability</td>
</tr>
<tr>
<td>5</td>
<td>27-Mar</td>
<td>Signal Representation</td>
</tr>
<tr>
<td>6</td>
<td>30-Mar</td>
<td>Holiday</td>
</tr>
<tr>
<td>6</td>
<td>10-Apr</td>
<td>Frequency Response</td>
</tr>
<tr>
<td>7</td>
<td>12-Apr</td>
<td>Transform</td>
</tr>
<tr>
<td>7</td>
<td>17-Apr</td>
<td>Noise &amp; Filtering</td>
</tr>
<tr>
<td>7</td>
<td>19-Apr</td>
<td>Analog Filters</td>
</tr>
<tr>
<td>8</td>
<td>24-Apr</td>
<td>Discrete-Time Signals</td>
</tr>
<tr>
<td>8</td>
<td>26-Apr</td>
<td>Discrete-Time Systems</td>
</tr>
<tr>
<td>9</td>
<td>1-May</td>
<td>Digital Filters &amp; IIR/FIR Systems</td>
</tr>
<tr>
<td>9</td>
<td>3-May</td>
<td>Fourier Transform &amp; DTFT</td>
</tr>
<tr>
<td>10</td>
<td>8-May</td>
<td>Introduction to Digital Control</td>
</tr>
<tr>
<td>10</td>
<td>10-May</td>
<td>Stability of Digital Systems</td>
</tr>
<tr>
<td>11</td>
<td>15-May</td>
<td>PID &amp; Computer Control</td>
</tr>
<tr>
<td>12</td>
<td>17-May</td>
<td>Applications in Industry</td>
</tr>
<tr>
<td>12</td>
<td>22-May</td>
<td>State-Space</td>
</tr>
<tr>
<td>12</td>
<td>24-May</td>
<td>Controllability &amp; Observability</td>
</tr>
<tr>
<td>13</td>
<td>28-May</td>
<td>Information Theory/Communications &amp; Review</td>
</tr>
<tr>
<td>13</td>
<td>31-May</td>
<td>Summary and Course Review</td>
</tr>
</tbody>
</table>
Goals for the Week

- Wrap up Digital Filters (and bookend it with a Pop-Quiz)
- Introduce Digital Systems and Feedback Control
- z-Transform for Digital Control
- Design Using Digital Equivalents ➔ Friday
- Stability of Digital Systems

Announcements:

- Final Exam:
  - 15 Questions (60% Short Answer, 40% Regular Problems)
  - 3 Hours
  - Closed-book
  - Took tutor ~90min to complete
  - Equation sheet will be provided (in addition to your own)
    [See Prac Final – Coming out next week]
  - Yes, it has an unexpected twist at the end, but you’ll like it. 😊

  ➔Saturday, June 15 at 9:30 AM (sorry!)
Digital control

Once upon a time…

- Electromechanical systems were controlled by electromechanical compensators
  - Mechanical flywheel governors, capacitors, inductors, resistors, relays, valves, solenoids (fun!)
  - But also complex and sensitive!

> Idea: Digital computers in real-time control
  - Transform approach (classical control)
    - Root-locus methods (pretty much the same as METR 3200)
    - Bode’s frequency response methods (these change compared to METR 3200)
  - State-space approach (modern control)

> Model Making: Control of frequency response as well as Least Squares Parameter Estimation
Simple Controller Goes Digital

\[ d_i = \text{desiredFront} \quad d_o = \text{distanceFront} \]

\[ \begin{align*}
\text{plant:} & \quad y[n] = y[n-1] - T u[n-1] \\
\text{sensor:} & \quad y'[n] = u'[n-1] \\
\text{controller:} & \quad y[n] = K u[n] \\
\end{align*} \]

Complex system behaviors, depending on \( K \)

Return to the discrete domain

- Recall that continuous signals can be represented by a series of samples with period \( T \)
Zero Order Hold

- An output value of a synthesised signal is held constant until the next value is ready
  - This introduces an effective delay of $T/2$

Digitisation

- Continuous signals sampled with period $T$
- $k$th control value computed at $t_k = kT$
Digitisation

- Continuous signals sampled with period $T$
- $k$th control value computed at $t_k = kT$

![Diagram showing the process of digitisation]

Difference equations

- How to represent differential equations in a computer? Difference equations!
- The output of a difference equation system is a function of current and previous values of the input and output:

$$y(t_k) = D(x(t_k), x(t_{k-1}), \ldots, x(t_{k-n}), y(t_{k-1}), \ldots, y(t_{k-n}))$$

  - We can think of $x$ and $y$ as parameterised in $k$
  - Useful shorthand: $x(t_{k+i}) \equiv x(k + i)$
Euler’s method*

- Dynamic systems can be approximated† by recognising that:

\[
\dot{x} \approx \frac{x(k + 1) - x(k)}{T}
\]

- As \( T \to 0 \), approximation error approaches 0

*Also known as the forward rectangle rule
†Just an approximation — more on this later

---

An example!

Convert the system \( \frac{Y(s)}{X(s)} = \frac{s+2}{s+1} \) into a difference equation with period \( T \), using Euler’s method.

1. Rewrite the function as a dynamic system:

\[
sY(s) + Y(s) = sX(s) + 2X(s)
\]

Apply inverse Laplace transform:

\[
y(t) + y(t) = \dot{x}(t) + 2x(t)
\]

2. Replace continuous signals with their sampled domain equivalents, and differentials with the approximating function

\[
\frac{y(k + 1) - y(k)}{T} + y(k) = \frac{x(k + 1) - x(k)}{T} + 2x(k)
\]
An example!

Simplify:

\[
\begin{align*}
y(k + 1) - y(k) + Ty(k) &= x(k + 1) - x(k) + 2Tx(k) \\
y(k + 1) + (T - 1)y(k) &= x(k + 1) + (2T - 1)x(k) \\
y(k + 1) &= x(k + 1) + (2T - 1)x(k) - (T - 1)y(k)
\end{align*}
\]

We can implement this in a computer.

Cool, let’s try it!

---

Back to the future

A quick note on causality:

- Calculating the “\((k+1)th\)” value of a signal using

\[
y(k + 1) = x(k + 1) + Ax(k) - By(k)
\]

relies on also knowing the next (future) value of \(x(t)\).

- Real systems always run with a delay:

\[
y(k) = x(k) + Ax(k - 1) - By(k - 1)
\]
Back to the example!

T = 0.02; //period of 50 Hz, a number pulled from thin air
A = 2*T-1; //pre-calculated control constants
B = T-1;
...

while(1)
{
    if(interrupt_flag) //this triggers every 20 ms
    {
        x0 = x; //save previous values
        y0 = y;
        x = update_input(); //get latest x value
        y = x + A*x0 - B*y0; //do the difference equations
        update_output(y); //write out current value
    }
}

(The actual maths bit)

Coping with complexity

- Transfer functions help control complexity
  - Recall the Laplace transform:
    \[ \mathcal{L}\{f(t)\} = \int_0^{\infty} f(t)e^{-st}dt = F(s) \]
    where
    \[ \mathcal{L}\{\dot{f}(t)\} = sF(s) \]

\[
\begin{array}{ccc}
x(t) & \xrightarrow{H(s)} & y(t)
\end{array}
\]

Is there a something similar for sampled systems?
The \( z \)-transform

- The discrete equivalent is the \( z \)-Transform†:

\[
Z \{ f(\k) \} = \sum_{k=0}^{\infty} f(k)z^{-k} = F(z)
\]

and

\[
Z \{ f(k-1) \} = z^{-1}F(z)
\]

Convenient!

†This is not an approximation, but approximations are easier to derive

Some useful properties
- **Delay by \( n \) samples:** \( Z \{ f(k-n) \} = z^{-n}F(z) \)
- **Linear:** \( Z \{ af(k) + bg(k) \} = aF(z) + bG(z) \)
- **Convolution:** \( Z \{ f(k) * g(k) \} = F(z)G(z) \)

So, all those block diagram manipulation tools you know and love will work just the same!
The z-transform

- In practice, you’ll use look-up tables or computer tools (ie. Matlab) to find the z-transform of your functions

<table>
<thead>
<tr>
<th>$F(s)$</th>
<th>$F(kt)$</th>
<th>$F(z)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{1}{s}$</td>
<td>$1$</td>
<td>$\frac{z}{z-1}$</td>
</tr>
<tr>
<td>$\frac{1}{s^2}$</td>
<td>$kT$</td>
<td>$\frac{Tz}{(z-1)^2}$</td>
</tr>
<tr>
<td>$\frac{1}{s+a}$</td>
<td>$e^{-akT}$</td>
<td>$\frac{z}{z-e^{-at}}$</td>
</tr>
<tr>
<td>$\frac{1}{(s+a)^2}$</td>
<td>$kTe^{-akT}$</td>
<td>$\frac{Tz e^{-at}}{(z-e^{-at})^2}$</td>
</tr>
<tr>
<td>$\frac{1}{s^2 + a^2}$</td>
<td>$\sin(akT)$</td>
<td>$\frac{z \sin at}{z^2 - (2 \cos at)z + 1}$</td>
</tr>
</tbody>
</table>

Why z-Transform

- Makes it easy to analyse feedback systems governed by difference equations
- For any complex number $z = re^{j\omega}$, $y[n] \xrightarrow{z} Y(z)$

Forward Analysis: $Y(z) = \sum_{k=-\infty}^{\infty} h[k]z^{-k}$

Backward Synthesis
(for any fixed $r > 0$ on which the Z-transform converges):

$$y[n] = \frac{1}{2\pi j} \int_{2\pi} Y(re^{j\omega})(re^{j\omega})^n \, d\omega$$
z-Transforms for Difference Equations

- First-order linear constant coefficient difference equation:

\[
y[n] = ay[n-1] + bu[n]
\]

\[
h[n] = \begin{cases} 
  ba^n & n \geq 0, \\
  0 & \text{otherwise.}
\end{cases}
\]

\[
H(z) = \sum_{k=0}^{\infty} ba^k z^{-k} = b \sum_{k=0}^{\infty} \left(\frac{a}{z}\right)^k = \frac{b}{1 - az^{-1}}, \quad \text{when } |z| > |a|.
\]

z-Transforms for Difference Equations

- First-order linear constant coefficient difference equation:

\[
y[n] = ay[n-1] + bu[n]
\]

\[
y[n] - ay[n-1] = bu[n]
\]

\[
\downarrow
\]

\[
Y(z) - az^{-1}Y(z) = bU(z)
\]

\[
H(z) = \frac{Y(z)}{U(z)} = \frac{b}{1 - az^{-1}}, \quad \text{when does it converge?}
\]
Region of Convergence (ROC) Plots

\[ H(z) = \frac{Y(z)}{U(z)} = \frac{b}{1 - az^{-1}}, \quad |z| > |a| \]

Right-sided signals have “outsided” ROCs.

Left-sided signals have “insided” ROCs.

Properties of the ROC

- The ROC is always defined by circles centered around the origin.
  
  \( h[n] r^{-n} \) is absolutely summable, where \( r = |z| \).

- Right-sided signals have “outsided” ROCs.
  
  If \( \exists n_0 \) such that \( h[n] = 0 \) \( \forall n < n_0 \), then if \( r_0 \in \text{ROC} \), then \( \forall r \) with \( r_0 < r < \infty \) are also in the ROC.

- Left-sided signals have “insided” ROCs.
  
  (with \( \forall r \) within \( 0 < r < r_0 \))
Combinations of Signals

\[ y_1[n] = \begin{cases} ba^n & n \geq 0 \\ 0 & n < 0 \end{cases}, \quad y_2[n] = \begin{cases} 0 & n \geq 0 \\ -ba^n & n < 0 \end{cases} \]

\[
\alpha \text{ ROC for } \alpha_1 y_1[n] + \alpha_2 y_2[n]
\]

Higher-order difference equations

\[ y[n] = a_1 y[n-1] + a_2 y[n-2] + a_3 y[n-3] + b_0 u[n] + b_1 u[n-1] + \ldots \]

Easy to take the Z-transform

\[ Y(z) = a_1 z^{-1} Y(z) + a_2 z^{-2} Y(z) + a_3 z^{-3} Y(z) + b_0 U(z) + \ldots \]

\[ H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \ldots}{1 - a_1 z^{-1} - a_2 z^{-2} - a_3 z^{-3} + \ldots} \]
Final value theorem

- An important question: what is the steady-state output a stable system at \( t = \infty \)?
  - For continuous systems, this is found by:
    \[
    \lim_{t \to \infty} x(t) = \lim_{s \to 0} sX(s)
    \]
  - The discrete equivalent is:
    \[
    \lim_{k \to \infty} x(k) = \lim_{z \to 1} (1 - z^{-1})X(z)
    \]
    (Provided the system is stable)

An example!

- Back to our difference equation:
  \[
  y(k) = x(k) + Ax(k - 1) - By(k - 1)
  \]
  becomes
  \[
  Y(z) = X(z) + Az^{-1}X(z) - Bz^{-1}Y(z)
  \]
  \[
  (z + B)Y(z) = (z + A)X(z)
  \]
  which yields the transfer function:
  \[
  \frac{Y(z)}{X(z)} = \frac{z + A}{z + B}
  \]

Note: It is also not uncommon to see systems expressed as polynomials in \( z^{-n} \).
This looks familiar…

• Compare:

\[
\frac{Y(s)}{X(s)} = \frac{s+2}{s+1} \quad \text{vs} \quad \frac{Y(z)}{X(z)} = \frac{z+A}{z+B}
\]

How are the Laplace and \( z \) domain representations related?

Consider the simplest system

• Take a first-order response:

\[
f(t) = e^{-at} \Rightarrow \mathcal{L}\{f(t)\} = \frac{1}{s + a}
\]

• The discrete version is:

\[
f(kT) = e^{-akT} \Rightarrow \mathcal{Z}\{f(k)\} = \frac{z}{z - e^{-aT}}
\]

The equivalent system poles are related by

\[
z = e^{sT}
\]

That sounds somewhat profound… but what does it mean?
The $z$-Plane

- $z$-domain poles and zeros can be plotted just like $s$-domain poles and zeros:

![Diagram of $z$-Plane](image)

Deep insight #1

The mapping between continuous and discrete poles and zeros acts like a distortion of the plane

![Diagram of distortion](image)
The $z$-plane

- We can understand system response by pole location in the $z$-plane

[Adapted from Franklin, Powell and Emami-Naeini]

Effect of pole positions

- We can understand system response by pole location in the $z$-plane
Effect of pole positions

- We can understand system response by pole location in the $z$-plane

![Graph showing the effect of pole positions](image)

Effect of pole positions

- We can understand system response by pole location in the $z$-plane

![Graph showing the effect of pole positions](image)
Damping and natural frequency

\[ z = e^{\xi T} \text{ where } s = -\xi \omega_n \pm j \omega_n \sqrt{1 - \xi^2} \]

Quick refresher: the root locus

- The transfer function for a closed-loop system can be easily calculated:

\[
\begin{align*}
  y &= CH(r - y) \\
  y + CHy &= CHr \\
  \therefore \frac{y}{r} &= \frac{CH}{1 + CH}
\end{align*}
\]
Quick refresher: the root locus

- We often care about the effect of increasing gain of a control compensator design:

\[ \frac{y}{r} = \frac{kCH}{1 + kCH} \]

Multiplying by denominator:

\[ \frac{y}{r} = \frac{kC_nH_n}{C_dH_d + kCnHn} \]

**Example:**

- Is this system stable?

\[ u(k) = 0.9u(k - 1) - 0.2u(k - 2) \]

- Time-shift it:

\[ u(k + 2) = 0.9u(k + 1) - 0.2u(k) \]

- z-Transform:

\[ (1)z^2 - 0.9z + 0.2 = 0 \]

- Characteristic Roots:

\( z = 0.5, z = 0.4 \) \( \Rightarrow \) STABLE!
Quick refresher: the root locus

- Pole positions change with increasing gain
  - The trajectory of poles on the pole-zero plot with changing $k$ is called the “root locus”
  - This is sometimes quite complex

(In practice you’d plot these with computers)

z-plane stability

- In the $z$-domain, the unit circle is the system stability bound
**z-plane stability**

- In the $z$-domain, the unit circle is the system stability bound.

\[
\begin{align*}
\text{Img}(z) & \quad \text{Re}(z) \\
\text{Img}(s) & \quad \text{Re}(s)
\end{align*}
\]

---

**z-plane stability**

- The $z$-plane root-locus in closed loop feedback behaves just like the $s$-plane:

\[
\begin{align*}
\text{Img}(z) & \quad \text{Re}(z) \\
\text{Img}(s) & \quad \text{Re}(s)
\end{align*}
\]
Deep insight #2

Gains that stabilise continuous systems can actually *destabilise* digital systems!

Quick plug*

- Most of this is based on Chapter 8 of "Feedback Control of Dynamic Systems" by Franklin, Powell and Emami-Naeini.

* No, they’re not paying me – it’s just a really good book!