State-Space:
Controllably & Observability

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

Lecture 23

elec3004@itee.uq.edu.au
http://robotics.itee.uq.edu.au/~elec3004/

Today in Linear Systems…

<table>
<thead>
<tr>
<th></th>
<th>27-Feb Introduction</th>
<th>1-Mar Systems Overview</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>6-Mar Signals &amp; Signal Models</td>
<td>8-Mar System Models</td>
</tr>
<tr>
<td>3</td>
<td>13-Mar Linear Dynamical Systems</td>
<td>15-Mar Sampling &amp; Data Acquisition</td>
</tr>
<tr>
<td>4</td>
<td>20-Mar Time Domain Analysis of Continuous Time Systems</td>
<td>22-Mar System Behavior &amp; Stability</td>
</tr>
<tr>
<td>5</td>
<td>29-Mar Signal Representation</td>
<td>29-Mar Notes</td>
</tr>
<tr>
<td>6</td>
<td>20-Apr Frequency Response</td>
<td>12-Apr Transfer</td>
</tr>
<tr>
<td>7</td>
<td>17-Apr Noise &amp; Filtering</td>
<td>15-Apr Stability Filters</td>
</tr>
<tr>
<td>8</td>
<td>24-Apr Fourier-Time Signals</td>
<td>26-Apr Fourier-Time Systems</td>
</tr>
<tr>
<td>9</td>
<td>1-May Analog Filters &amp; IIR Systems</td>
<td>3-May Fourier Transform &amp; DFT</td>
</tr>
<tr>
<td>10</td>
<td>8-May Introduction to Digital Control</td>
<td>10-May Stability of Digital Systems</td>
</tr>
<tr>
<td>11</td>
<td>15-May PID &amp; Computer Control</td>
<td>17-May Applications in Industry</td>
</tr>
<tr>
<td>12</td>
<td>22-May State-Space</td>
<td>24-May State-Space</td>
</tr>
<tr>
<td>13</td>
<td>29-May Controllability &amp; Observability</td>
<td>31-May Information Theory/Communications &amp; Review</td>
</tr>
</tbody>
</table>
Goals for the Week

Today:
• State-Space
• Compensator Design

Friday:
• Controllability
• Observability

Affairs of state
• Introductory brain-teaser:
  – If you have a dynamic system model with history (ie. integration) how do you represent the instantaneous state of the plant?

Eg. how would you setup a simulation of a step response, mid-step?

\[ t = 0 \]

start
Introduction to state-space

• Linear systems can be written as networks of simple dynamic elements:

\[
H = \frac{s + 2}{s^2 + 7s + 12} = \frac{2}{s + 4} + \frac{-1}{s + 3}
\]

\[
\begin{array}{c}
\Sigma \\
\frac{1}{s} \\
\Sigma \\
-7 \\
-12
\end{array} \rightarrow \begin{array}{c}
\frac{1}{s} \\
1 \\
2 \\
\Sigma
\end{array} \rightarrow y
\]

Introduction to state-space

• We can identify the nodes in the system
  – These nodes contain the integrated time-history values of the system response
  – We call them “states”

\[
\begin{array}{c}
\Sigma \\
\frac{1}{s} \\
\Sigma \\
-7 \\
-12
\end{array} \rightarrow \begin{array}{c}
\frac{1}{s} \\
1 \\
2 \\
\Sigma
\end{array} \rightarrow y
\]
State-Space Terminology

\[ \dot{x}(t) = A(t)x(t) + B(t)u(t) \]
\[ y(t) = C(t)x(t) + D(t)u(t) \]

- **u**: Input \[ u : [0, \infty) \rightarrow \mathbb{R}^k \]
- **x**: State \[ x : [0, \infty) \rightarrow \mathbb{R}^n \]
- **y**: Output \[ y : [0, \infty) \rightarrow \mathbb{R}^m \]
### LTI State-Space

\[
\dot{x}(t) = A(t)x(t) + B(t)u(t) \\
y(t) = C(t)x(t) + D(t)u(t)
\]

- **If** the system is **linear and time invariant**, **then** \( A, B, C, D \) are constant coefficient

\[
\rightarrow \dot{x} = Ax + Bu \\
\rightarrow y =Cx + Du
\]

### Discrete Time State-Space

\[
\dot{x}(t) = A(t)x(t) + B(t)u(t) \\
y(t) = C(t)x(t) + D(t)u(t)
\]

- **If** the system is **discrete**, **then** \( x \) and \( u \) are given by difference equations

\[
\rightarrow x[k + 1] = A[k]x[k] + B[k]u[k] \\
y[k] = C[k]x[k] + D[k]u[k]
\]

\[
\rightarrow x^+ = Ax + Bu \\
y = Cx + Du
\]
Block Diagram Algebra in State Space

• Series:

\[ X(s) \xrightarrow{F(s)} G(s) \xrightarrow{Y(s)} \]

\[
\begin{bmatrix}
\dot{x}_G \\
\dot{x}_F
\end{bmatrix} =
\begin{bmatrix}
A_G & B_G C_F \\
0 & A_F
\end{bmatrix}
\begin{bmatrix}
x_G \\
x_F
\end{bmatrix} +
\begin{bmatrix}
B_G D_F \\
B_F
\end{bmatrix} u
\]

System 1:
\[
x_F' = A_F x_F + B_F u \\
y_F = C_F x_F + D_F u
\]

System 2:
\[
x_G' = A_G x_G + B_G y_F \\
y_G = C_G x_G + D_G y_F
\]

• Parallel:

\[ X(s) \xrightarrow{F(s)} G(s) \xrightarrow{Y(s)} \]

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{bmatrix} =
\begin{bmatrix}
A_1 & 0 \\
0 & A_2
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2
\end{bmatrix} +
\begin{bmatrix}
B_1 \\
B_2
\end{bmatrix} u
\]

\[
y =
\begin{bmatrix}
C_1 & C_2
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2
\end{bmatrix} + (D_1 + D_2) u
\]
State-space representation

- State-space matrices are not necessarily a unique representation of a system
  - There are two common forms

- Control canonical form
  - Each node – each entry in $x$ – represents a state of the system (each order of $s$ maps to a state)

- Modal form
  - Diagonals of the state matrix $A$ are the poles ("modes") of the transfer function

Control canonical form

- CCF matrix representations have the following structure:

$$
\begin{bmatrix}
-a_1 & -a_1 & \cdots & -a_{n-2} & -a_{n-1} & -a_n \\
1 & 0 & 0 & 0 & 0 \\
0 & 1 & \ddots & \cdots & \ddots & \vdots \\
\vdots & \ddots & 1 & 0 & 0 \\
0 & 0 & \cdots & 1 & 0 & 0
\end{bmatrix}
$$

Pretty diagonal!
State variable transformation

- Important note!
  - The states of a control canonical form system are not the same as the modal states
  - They represent the same dynamics, and give the same output, but the vector values are different!

- However we can convert between them:
  - Consider state representations, \( x \) and \( q \) where
    \[
    x = Tq
    \]
    \( T \) is a “transformation matrix”

State variable transformation

- Two homologous representations:
  \[
  \dot{x} = Ax + Bu \quad \text{and} \quad \dot{q} = Fq + Gu \\
  y = Cx + Du \quad \text{and} \quad y = Hq + Ju
  \]

We can write:
\[
\dot{x} = T\dot{q} = ATz + Bu \\
\dot{q} = T^{-1}ATz + T^{-1}Bu
\]

Therefore, \( F = T^{-1}AT \) and \( G = T^{-1}B \)

Similarly, \( C = HT \) and \( D = J \)
Controllability matrix

• To convert an arbitrary state representation in $F$, $G$, $H$ and $J$ to control canonical form $A$, $B$, $C$ and $D$, the “controllability matrix”

$$\mathcal{C} = [G \quad FG \quad F^2G \quad \ldots \quad F^{n-1}G]$$

must be nonsingular.

Why is it called the “controllability” matrix?

Controllability matrix

• If you can write it in CCF, then the system equations must be linearly independent.

• Transformation by any nonsingular matrix preserves the controllability of the system.

• Thus, a nonsingular controllability matrix means $x$ can be driven to any value.
Why is this “Kind of awesome”?

• The controllability of a system depends on the particular set of states you chose

• You can’t tell just from a transfer function whether all the states of $x$ are controllable

• The poles of the system are the Eigenvalues of $F$, ($p_i$).

State evolution

• Consider the system matrix relation:

\[
\dot{x} = Fx + Gu \\
y = Hx +Ju
\]

The time solution of this system is:

\[
x(t) = e^{F(t-t_0)} x(t_0) + \int_{t_0}^{t} e^{F(t-\tau)} Gu(\tau) d\tau
\]

If you didn’t know, the matrix exponential is:

\[
e^{Kt} = I + Kt + \frac{1}{2!} K^2 t^2 + \frac{1}{3!} K^3 t^3 + \ldots
\]
### Stability

- We can solve for the natural response to initial conditions $x_0$:

$$x(t) = e^{pt}x_0$$

$$\therefore \dot{x}(t) = p_t e^{pt}x_0 = Fe^{pt}x_0$$

Clearly, a system will be stable provided $\text{eig}(F) < 0$

---

### Characteristic polynomial

- From this, we can see $Fx_0 = p_l x_0$

or, $(p_l I - F)x_0 = 0$

which is true only when $\det(p_l I - F)x_0 = 0$

Aka. the characteristic equation!

- We can reconstruct the CP in $s$ by writing:

$$\det(sI - F)x_0 = 0$$
Great, so how about control?

• Given \( \dot{x} = Fx + Gu \), if we know \( F \) and \( G \), we can design a controller \( u = -Kx \) such that
  \[
  \text{eig}(F - GK) < 0
  \]

• In fact, if we have full measurement and control of the states of \( x \), we can position the poles of the system in arbitrary locations!
  
  (Of course, that never happens in reality.)

Example: PID control

• Consider a system parameterised by three states:
  – \( x_1, x_2, x_3 \)
  – where \( x_2 = \dot{x}_1 \) and \( x_3 = \dot{x}_2 \)

\[
\begin{align*}
\dot{x} &= \begin{bmatrix} 1 & 1 & -2 \end{bmatrix} x - Ku \\
y &= \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} x + 0u
\end{align*}
\]

\( x_2 \) is the output state of the system;
\( x_1 \) is the value of the integral;
\( x_3 \) is the velocity.
• We can choose $K$ to move the eigenvalues of the system as desired:

$$
\text{det} \begin{bmatrix}
1 - K_1 \\
1 - K_2 \\
-2 - K_3
\end{bmatrix} = 0
$$

All of these eigenvalues must be positive.

It’s straightforward to see how adding derivative gain $K_3$ can stabilise the system.

---

**Just scratching the surface**

• There is a lot of stuff to state-space control

• One lecture (or even two) can’t possibly cover it all in depth

  Go play with Matlab and check it out!
Discretisation FTW!

- We can use the time-domain representation to produce difference equations!

\[
x(kT + T) = e^{FT} x(kT) + \int_{kT}^{kT+T} e^{F(kT+T-\tau)} Gu(\tau) d\tau
\]

Notice \( u(\tau) \) is not based on a discrete ZOH input, but rather an integrated time-series.

We can structure this by using the form:

\[
u(\tau) = u(kT'), \quad kT \leq \tau \leq kT + T
\]

---

Discretisation FTW!

- Put this in the form of a new variable:

\[
\eta = kT + T - \tau
\]

Then:

\[
x(kT + T) = e^{FT} x(kT) + \left( \int_{kT}^{kT+T} e^{F\eta} d\eta \right) Gu(kT')
\]

Let’s rename \( \Phi = e^{FT} \) and \( \Gamma = \left( \int_{kT}^{kT+T} e^{F\eta} d\eta \right) G \)
Discrete state matrices

So,
\[ x(k + 1) = \Phi x(k) + \Gamma u(k) \]
\[ y(k) = H x(k) + J u(k) \]

Again, \( x(k + 1) \) is shorthand for \( x(kT + T) \)

Note that we can also write \( \Phi \) as:
\[ \Phi = I + FT \Psi \]
where
\[ \Psi = I + \frac{FT}{2!} + \frac{F^2T^2}{3!} + \ldots \]

Simplifying calculation

• We can also use \( \Psi \) to calculate \( \Gamma \)
  – Note that:
\[ \Gamma = \sum_{k=0}^{\infty} \frac{F^k T^k}{(k + 1)!} T G \]
\[ = \Psi T G \]
\[ \Psi \] itself can be evaluated with the series:
\[ \Psi \cong I + \frac{FT}{2} \left\{ I + 
\frac{FT}{3} \left[ I + \ldots \frac{FT}{n-1} \left( I + \frac{FT}{n} \right) \right] \right\} \]
State-space z-transform

We can apply the z-transform to our system:

\[(zI - \Phi)X(z) = \Gamma U(k)\]

\[Y(z) = HX(z)\]

which yields the transfer function:

\[\frac{Y(z)}{X(z)} = G(z) = H(zI - \Phi)^{-1}\Gamma\]

State-space control design

- Design for discrete state-space systems is just like the continuous case.
  - Apply linear state-variable feedback:
    \[u = -Kx\]
  such that \(\text{det}(zI - \Phi + \Gamma K) = \alpha_c(z)\)

where \(\alpha_c(z)\) is the desired control characteristic equation

Predictably, this requires the system controllability matrix

\[C = [\Gamma \Phi \Phi^2 \Gamma \ldots \Phi^{n-1}\Gamma]\] to be full-rank.
Announcements:

• Final Exam:
  ➡ Saturday, June 15 at 9:30 AM (sorry!)

• Problem Set 2 is due this Friday!

Next Time in Linear Systems ….

• AKA: I can see that. Yes, I can control that!
\[ u, \quad y, \quad P, \quad y = \frac{P}{1+P} \]

\[ u_1 = u - y, \quad y_1 = y_1 \]

\[ P: \quad x_1 = A_1 x_1 + B_1 u_1, \quad y_1 = c_1 x_1 + d_1 u_1 \]

\[ x_1 = (A_1 - B_1 (I+D)\quad^{-1} c_1) x_1 \\
+ B_1 (I - (I+D)\quad^{-1} D_1) u \]

\[ y = (I+D)\quad^{-1} c_1 x_1 \\
+ (I+D_1)\quad^{-1} D_1 u \]
STATE-SPACE DECOMPOSITION

\[ (1) \quad x = x + u \]
\[ (2) \quad y = x \]
\[ y = S [ x ] \]

\[ U \rightarrow 0 \rightarrow x \rightarrow \square \rightarrow x \rightarrow y \]

\[ U + \rightarrow 0 \rightarrow \square \rightarrow y \]

\[ U \rightarrow \square \rightarrow \square \rightarrow y \]

Compare to:

\[ \rightarrow 0 \rightarrow \square \rightarrow y \]
[Cumulative Sum]

or

\[ \rightarrow 0 \rightarrow \square \rightarrow y \]
[Delay Stage]
Q-Profile for a Resonator

\[ \frac{Q}{\omega_0} \]

No Damping\(\Rightarrow Q = \infty\)

\[ \frac{1}{2} \cdot x^T \cdot m \cdot x \rightarrow \frac{1}{2} M \cdot x^2 \]