1. Linear systems
representations and properties

(Chapters 2, 3)

Torsten Söderström
Department of Systems and Control
Information Technology
Uppsala University
REPRESENTATIONS OF A LINEAR MULTIVARIABLE SYSTEM

\[ u(t) \rightarrow \text{SYSTEM} \rightarrow y(t) \]

- Input \( u(t) \) (\( m \) dimensional)
- Output \( y(t) \) (\( p \) dimensional)
REPRESENTATIONS

- Impulse response (weighting function)
- Transfer function (matrix)
- Transfer operator
- Differential equations in $u(t)$ and $y(t)$
- State-space form

In discrete-time use sampling interval as time-unit.
IMPULSE RESPONSE

Continuous-time

\[ y(t) = \int_0^\infty g(\tau)u(t - \tau) \, d\tau \]

Discrete-time

\[ y(t) = \sum_{k=0}^{\infty} g(k)u(t - k) \]

The weighting function \( g(\cdot) \) weights the influence of old input values.

When \( u(t) \) is an impulse, \( y(t) = g(t) \).
TRANSFER FUNCTION

Continuous-time: Take Laplace transforms:

\[ G(s) = \mathcal{L}(g) = \int_0^\infty e^{-st} g(t) \, dt \]
\[ U(s) = \mathcal{L}(u), \quad Y(s) = \mathcal{L}(y) \]

leading to

\[ Y(s) = G(s)U(s) \]

Note that \( G(s) \) is a \( p \times m \) matrix.
TRANSFER FUNCTION, cont’d

**Discrete-time:** Use Z-transform instead, leading to

\[ G(z) = \mathcal{Z}(g) = \sum_{k=0}^{\infty} z^{-k} g(k) \]

\[ Y(z) = G(z)U(z) \]
TRANSFER OPERATOR

**Continuous-time**: Introduce the differentiation operator

\[ py(t) = \frac{d}{dt} y(t) \]

Then

\[ y(t) = G(p)u(t) \]

**Discrete-time**: Introduce the (forward) shift operator

\[ qy(t) = y(t + 1) \]

Then

\[ y(t) = G(q)u(t) \]
Continuous-time. Assume $G(s)$ rational

$$G(s) = \frac{B(s)}{A(s)}$$

with $A(s), B(s)$ being polynomials in $s$. Then

$$y(t) = G(p)u(t) \Rightarrow A(p)y(t) = B(p)u(t)$$

(a linear differential equation)

In the multivariable case, $A(p)$ and $B(p)$ are matrices, while $u(t)$ and $y(t)$ are vectors.

$$G(p) = A^{-1}(p)B(p)$$

is called a matrix fraction description.
INPUT-OUTPUT DIFFERENCE EQUATIONS

Discrete-time.

Change $p$ to $q$ to get difference equations rather than differential equations.

$$y(t) = \frac{B(q)}{A(q)} u(t)$$

can be written as a linear difference equation

$$A(q)y(t) = B(q)u(t)$$

Causality: $\deg A(q) \geq \deg B(q)$
STATE SPACE FORM

Coupled first order equations

\[
\dot{x}(t) = Ax(t) + Bu(t) \\
y(t) = Cx(t) + Du(t)
\]

The state \( x(t) \) contains all information about the system’s history that has any impact on the future behavior.

In discrete time, change \( px(t) = \dot{x}(t) \) to \( qx(t) = x(t + 1) \).
CONNECTIONS

State-space to transfer function:

\[ G(p) = C(pI - A)^{-1}B + D \]

Transfer function to state space:

- Many possibilities, several special cases
  - Diagonal form
  - Canonical forms (controllable form, observable form)
- Multivariable systems more complex than SISO systems

Discrete-time systems?
CONNECTIONS, cont’d

State-space to weighting function

\[ g(t) = Ce^{At}B + D\delta(t) \]

Discrete-time?
How can we link continuous-time representations with discrete-time models?

Some alternatives:

- Approximate derivatives by differences

\[ py(t) \approx \frac{y(t + h) - y(t)}{h} \]

meaning

\[ p \approx \frac{q - 1}{h} \]

A more sophisticated approximation: (Tustin, bilinear transformation)

\[ p \approx \frac{2q - 1}{h(q + 1)} \]
CONNECTIONS, cont’d

How can we link continuous-time representations with discrete-time models?

- Specify how the input varies between the sampling points. Solve the differential equation exactly.

  *Example:* In a computer-controlled system the input is constant over the sampling intervals, and we get a sampled-data system. (Details will be given later.)
MATLAB commands

- `ss`, `tf`. Create a system in state space form, or as a transfer function
- `ss2tf`. Transformation from state space form to transfer function
- `tf2ss`. Transformation from transfer function to state space form
- `impulse`. Impulse response
- `step`. Step response
Properties of Linear Multivariable System

- Solution of the system equations
- Controllability and observability
- Poles and zeros
- Stability
- Frequency functions

Which concepts are the same as in the SISO case?
SOLUTION OF THE SYSTEM EQUATIONS

Continuous-time. System representation

\[
\begin{align*}
\dot{x}(t) &= Ax(t) + Bu(t) \\
y(t) &= Cx(t) + Du(t)
\end{align*}
\]

Solution

\[
x(t) = e^{A(t-t_o)}x(t_o) + \int_{t_o}^{t} e^{A(t-\tau)}Bu(\tau) \, d\tau
\]

- The first term gives influence of the initial condition
- The second term gives influence of the input actions over the time interval \([t_o, t]\).
Discrete-time. System representation

\[ x(t + 1) = Ax(t) + Bu(t) \]
\[ y(t) = Cx(t) + Du(t) \]

Solution

\[ x(t) = A^{t-t_o}x(t_o) + \sum_{\tau=t_o}^{t-1} A^{t-\tau-1}Bu(\tau) \]

(t, t_o, \tau \text{ integers})
CONTROLLABILITY AND OBSERVABILITY

1. The state $x^*$ is controllable if there exists an input $u(t)$ that moves the system from $x = 0$ to $x = x^*$ in finite time.

2. A system is controllable if all states are controllable.

3. The state $x^*$ is non-observable if $x(0) = x^*$ and $u(t) \equiv 0$ gives $y(t) \equiv 0$.

4. A system is observable if there are no non-observable states.

N.B. The above definitions apply also in discrete-time.
The controllable states span the range of the controllability matrix

\[ S(A, B) = [B \ AB \ A^2B \ \ldots \ A^{n-1}B] \]

The system is controllable \( \iff \) \( S(A, B) \) has full rank. (If there is one input only, \( S \) is a square matrix, and full rank is equivalent to non-singular.)

A system in controller canonical form is always controllable.
• The non-observable states span the null space of the observability matrix

\[ \mathcal{O}(A, C') = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \]

• The system is observable \( \Leftrightarrow \mathcal{O}(C, A) \) has full rank.

• A system in observer canonical form is always observable.
(Pole placement) For given $A$ and $B$ one can find a matrix $L$ so that $A - BL$ has a given set of eigenvalues $\iff$ the matrix $S(A, B)$ has full rank $\iff$ the system $\dot{x} = Ax + Bu$ is controllable

- The system is a *minimal realization* if it is both controllable and observable.

- The system is controllable $\iff$

\[
\begin{bmatrix} A - \lambda I & B \end{bmatrix} \text{ full rank } \forall \lambda
\]

- The system is observable $\iff$

\[
\begin{bmatrix} A - \lambda I \\ C \end{bmatrix} \text{ full rank } \forall \lambda
\]
CONTROLLABILITY AND OBSERVABILITY, cont’d

Adjusted concepts that concern unstable modes.

- The system \((A, B, C)\) is *stabilizable* if there exists a matrix \(L\) such that \(A - BL\) has all eigenvalues in the stability region.

- A system \((A, B, C)\) is stabilizable \(\iff\) all unstable modes are controllable.

- A system \((A, B, C)\) is *detectable* if there exists a matrix \(K\) such that \(A - KC\) has all eigenvalues in the stability region.

- A system \((A, B, C)\) is detectable \(\iff\) all non-observable modes are stable.
STABILITY

The characteristic behavior of the system response is determined by

- $e^{At}$ in continuous-time
- $A^t$ in discrete-time

Mathematical result:
If the matrix $A$ has eigenvalues $p_1, \ldots, p_n$, the matrix exponential $e^{At}$ has eigenvalues in $e^{p_1t}, \ldots, e^{p_nt}$.

For a *continuous-time system* the stability region is the left half plane (without the imaginary axis).

For a *discrete-time system* the stability region is the interior of the unit circle.
STABILITY cont’d

A linear time-invariant system is \textit{input-output stable} \iff all its poles are located in the stability region.

A system with a zero outside the stability region is said to be \textit{non-minimum phase}.
STABILITY, cont’d

Testing stability:

- Direct computation (in Matlab use \texttt{roots}, or \texttt{eig}).
- Root locus
- Nyquist criterion
- Routh (continuous-time) or Jury-Schur-Cohn (discrete-time)
FREQUENCY FUNCTION

What does \( G(i\omega) \) tells us?

Assume the system is stable, and the input is

\[
u_k(t) = \cos(\omega t)
\]

After a transient the output will be

\[
y_j(t) = A \cos(\omega t + \varphi) \quad j = 1, \ldots, p
\]

\[
A = |G_{j,k}(i\omega)|, \quad \varphi = \arg[G_{j,k}(i\omega)]
\]

(Use SISO reasoning and superposition).

Nontrivial how to cope with multivariable systems, and how to treat coupling between signals.

Can we relate the gain to an amplitude curve?
USEFUL MATLAB COMMANDS

- `ss2zp`, `zp2ss`, `tf2zp`, `zp`. Transformations between representations
- `lsim`. Simulation
- `eig`, `roots`. Eigenvalues and poles
- `bode`. Frequency function, Bode plots
- `obsv`, `ctrb`. Observability, controllability