Module 08
Observability and State Estimator Design of Dynamical LTI Systems

Ahmad F. Taha

EE 5143: Linear Systems and Control

Email: ahmad.taha@utsa.edu

Webpage: http://engineering.utsa.edu/ataha

November 7, 2017
Introduction to Observability

- **Observability**: The ability to observer what’s happening inside your system (i.e., to know system states $x(t)$)
- Observability: In order to see what is going on inside the system under observation (i.e., output $y(t)$), the system must be observable. Observation: output $y(t)$
- Given this dynamical system:

  \[
  x(k + 1) = Ax(k) + Bu(k), \quad x(0) = x_0,
  \]

  \[
  y(k) = Cx(k) + Du(k),
  \]

  or

  \[
  \dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0,
  \]

  \[
  y(t) = Cx(t) + Du(t)
  \]

  a natural question arises: can we learn anything about $x(t)$ given $y(t)$ and $u(t)$ for a specific time $t$?
- Clearly, if we know $x(0)$ and $u(t)$ for all $t$, we can determine $x(t)$ via

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

  However, if $x(0)$ if unknown, can you obtain $x(t)$ via only $y(t), u(t)$?
Observability — 1

DTLTI system \((n \text{ states, } m \text{ inputs, } p \text{ outputs})\):

\[
\begin{align*}
    x(k + 1) &= Ax(k) + Bu(k), \quad x(0) = x_0, \\
    y(k) &= Cx(k) + Du(k),
\end{align*}
\]

- **Application**: given that \(A, B, C, D, \text{ and } u(k), y(k)\) are known \(\forall k = 0 : 1 : k - 1\), can we determine \(x(0)\)?

- **Solution**:

\[
\begin{bmatrix}
y(0) \\
y(1) \\
\vdots \\
y(k-1)
\end{bmatrix}
= \begin{bmatrix}
    C \\
    CA \\
    \vdots \\
    CA^{k-1}
\end{bmatrix}
\begin{bmatrix}
x(0) \\
\end{bmatrix} +
\begin{bmatrix}
    D & 0 & \cdots & 0 \\
    CB & D & \cdots & \vdots \\
    \vdots & \vdots & \ddots & \vdots \\
    CA^{k-2}B & \cdots & CB & 0
\end{bmatrix}
\begin{bmatrix}
u(0) \\
u(1) \\
\vdots \\
u(k-1)
\end{bmatrix}
\]
Observability — 2

\[
\begin{bmatrix}
  y(0) \\
  y(1) \\
  \vdots \\
  y(k-1)
\end{bmatrix}
= 
\begin{bmatrix}
  C & & & \\
  CA & & & \\
  \vdots & \ddots & \ddots & \\
  CA^{k-1} & \cdots & CB & D
\end{bmatrix}
\begin{bmatrix}
  x(0) \\
  u(0) \\
  \vdots \\
  u(k-1)
\end{bmatrix}
+ 
\begin{bmatrix}
  D & 0 & \cdots & 0 \\
  CB & D & \cdots & \vdots \\
  \vdots & \ddots & \ddots & 0 \\
  CA^{k-2}B & \cdots & CB & 0
\end{bmatrix}
\begin{bmatrix}
  u(0) \\
  u(1) \\
  \vdots \\
  u(k-1)
\end{bmatrix}
\]

\[Y(k-1) = O_k x(0) + T_k U(k-1) \implies O_k x(0) = Y(k-1) - T_k U(k-1)\]

- Since \(O_k, T_k, Y(k-1), U(k-1)\) are all known quantities, then we can find a unique \(x(0)\) iff \(O_k\) is full rank

**Observability Definition**

DTLTI system **is observable at time** \(k\) if the initial state \(x(0)\) can be uniquely determined from any given

\[u(0), \ldots, u(k-1), y(0), \ldots, y(k-1)\].
Quantifying Observability

Observability Test

For a system with \( n \) states and \( p \) outputs, the test for observability is that the matrix

\[
O = \begin{bmatrix}
C \\
CA \\
\vdots \\
CA^{n-1}
\end{bmatrix} \in \mathbb{R}^{np \times n}
\]

has full column rank (i.e., \( \text{rank}(C) = n \)).

The test is equivalent for DTLTI and CTTLTI systems.

Theorem

The following statements are equivalent:

1. \( O \) is full rank, system is observable
2. PBH Test: for any \( \lambda \in \mathbb{C} \), \( \text{rank} \left[ \begin{array}{c} \lambda I - A \\ C \end{array} \right] = n \)
3. Eigenvector Test: for any right eigenvector of \( A \), \( Cv_i \neq 0 \)
4. The following matrices are nonsingular

\[
\sum_{i=0}^{n-1} (A^T)^i C^T CA^i \quad \text{(DTLIT)} \quad \& \quad \int_{0}^{t} e^{A^T \tau} C^T Ce^{A \tau} \, d\tau \quad \text{(CTTLTI)}
\]
Example 1

Consider a dynamical system defined by:

\[
A = \begin{bmatrix}
1 & -1 & 0 \\
1 & -1 & 0 \\
0 & 0 & 0
\end{bmatrix},
B = \begin{bmatrix}
1 \\
0 \\
0
\end{bmatrix},
C = \begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

Is this system controllable?

Is this system observable?

**Answers:** Yes, Yes!

MATLAB commands: ctrb, obsv
Example 2

Determine whether the following system is observable or not:

\[
\begin{align*}
    x(k + 1) &= \begin{bmatrix}
        -1 & 1 & 0 & 0 & 0 & 0 & 0 \\
        0 & -1 & 0 & 0 & 0 & 0 & 0 \\
        0 & 0 & -1 & 0 & 0 & 0 & 0 \\
        0 & 0 & -1 & -1 & 0 & 0 & 0 \\
        0 & 0 & 0 & 0 & 0 & 0 & 0 \\
        0 & 0 & 0 & 0 & 1 & 0 & 0 \\
        0 & 0 & 0 & 0 & 0 & 1 & 0 \\
    \end{bmatrix} x(k) + \begin{bmatrix}
        0 \\
        1 \\
        -1 \\
        1 \\
        1 \\
        0 \\
        2
    \end{bmatrix} u(k) \\

    y(k) &= \begin{bmatrix}
        1 & 0 & 2 & 0 & 0 & 0 & 0 \\
        0 & 0 & 0 & 2 & 3 & 0 & 0
    \end{bmatrix} x(k).
\]

The challenge here is to be able to figure out which test should be used. Clearly, \( A \) has 7 evales as follows: \( \lambda_A = \{-1, -1, -1, -1, 0, 0, 0\} \). Test 2 is the easiest test to use here. Applying the test, you’ll see that the PBH test fails for the zero eigenvalue, which means that the system is not observable.
Unobservable Subspace

- **Unobservable subspace: null-space of** $\mathcal{O}_k = \mathcal{N}(\mathcal{O}_k)$
- It is basically the space (i.e., set of states $x \in \mathcal{X}$ that you cannot estimate or observer.
- Notice that if $x(0) \in \text{Null}(\mathcal{O}_k)$, and $u(k) = 0$, then the output is going to zero from $[0, k - 1]$.
- Notice that input $u(k)$ does not impact the ability to determine $x(0)$.
- The unobservable subspace $\mathcal{N}(\mathcal{O}_k)$ is $A$-invariant: if $z \in \mathcal{N}(\mathcal{O}_k)$, then $Az \in \mathcal{N}(\mathcal{O}_k)$.

Unobservable Space

The null spaces $\text{Null}(\mathcal{O}_k) = \mathcal{N}(\mathcal{O}_k)$ satisfy

$$\mathcal{N}(\mathcal{O}_0) \supseteq \mathcal{N}(\mathcal{O}_1) \supseteq \cdots \supseteq \mathcal{N}(\mathcal{O}_n) = \mathcal{N}(\mathcal{O}_{n+1}) = \cdots$$

This means that the more output measurements you have, the smaller the unobservable subspace.
It also implies that you cannot get more information if you go above $k > n$. You can prove this by C-H theorem ($A^n = \sum_{i=0}^{n-1} \alpha_i A^i$).
Detectability

Detectability Definition

DTLTI or CTLIT system, defined by \((A, C)\), is detectable if there exists a matrix \(L\) such that \(A - LC\) is stable.

Detectability Theorem

DTLTI or CTLIT system, defined by \((A, C)\) is detectable if all its unobservable modes correspond to stable eigenvalues of \(A\).

Facts:

- \(A\) is stable \(\Rightarrow\) \((A, C)\) is detectable
- \((A, C)\) is observable \(\Rightarrow\) \((A, C)\) is detectable as well
- \((A, B)\) is not observable \(\Rightarrow\) it could still be detectable
- If system has some unobservable modes that are unstable, then no gain \(L\) can make \(A - LC\) stable
- \(\Rightarrow\) Observer will fail to track system state
Observability for CT Systems

- The previous derivation for observability was for DT LTI systems.
- What if we have a CT LTI system? Do we obtain the same observability testing conditions?
- Yes, we do!
- First, note that the control input $u(t)$ plays no role in observability, just like how the output $y(t)$ plays no role in controllability.
- To see that, consider the following system with $n$ states, $p$ outputs, where (again) we want to obtain $x(t_0)$ (unknown):

$$\dot{x}(t) = Ax(t), \quad y(t) = Cx(t) \quad x(t_0) = x_0 \implies$$

$$y(t_0) = Cx(t_0)$$
$$\dot{y}(t_0) = C\dot{x}(t_0) = CAx(t_0)$$
$$\ddot{y}(t_0) = C\ddot{x}(t_0) = CA^2x(t_0)$$
$$\vdots$$
$$y^{(n-1)}(t_0) = Cx^{(n-1)}(t_0) = CA^{n-1}x(t_0)$$
We can write the previous equation as:

\[
\begin{bmatrix}
y(t_0) \\
y'(t_0) \\
y''(t_0) \\
\vdots \\
y^{(n-1)}(t_0)
\end{bmatrix}
= \begin{bmatrix}
C \\
CA \\
\vdots \\
CA^{n-1}
\end{bmatrix}
\Rightarrow
\begin{bmatrix}
x(t_0)
\end{bmatrix}
\]

\[Y(t_0) = \begin{bmatrix}
C \\
CA \\
\vdots \\
CA^{n-1}
\end{bmatrix} x(t_0) = \mathcal{O} \in \mathbb{R}^{np \times n}
\]

\[x(t_0) = \mathcal{O}^\dagger Y(t_0) = (\mathcal{O}^\top \mathcal{O})^{-1} \mathcal{O} Y(t_0)
\]

Hence, the initial conditions can be determined if the observability matrix is full column rank.

This condition is identical to the DT case where we also wanted to obtain \(x(k = 0)\) from a set of output measurements.

The difference here is that we had to obtain derivatives of the output at \(t_0\).

Can you rederive the equations if \(u(t) \neq 0\)? It won’t make an impact on whether a solution exists, but it’ll change \(x(t_0)\).
Controllability-Observability Duality, Minimality

**Duality**

The CT LTI system with state-space matrices \((\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D})\) is called the **dual** of another CT LTI system with state-space matrices \((A, B, C, D)\) if

\[
\tilde{A} = A^T, \quad \tilde{B} = C^T, \quad \tilde{C} = B^T, \quad \tilde{D} = D^T.
\]

**Controllability-Observability Duality**

CT system \((A, B, C, D)\) is observable (controllable) if and only if its dual system \((\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D})\) is controllable (observable).

**Minimality**

A system \((A, B, C, D)\) is called minimal if and only if it is both controllable and observable.
Observer Design

**Original system** with unknown $x(0)$:

\[
\dot{x} = Ax, \\
y =Cx
\]

**Simulator** with linear feedback:

\[
\dot{x} = A\hat{x} + L(y - \hat{y}), \quad \hat{x}(0) = 0 \\
\hat{y} = C\hat{x}
\]

- Objective here is to estimate (in real-time) the state of the actual system $x(t)$ given that ICs $x(0)$ are not known.
- To do that, we design an observer—dynamic state estimator (DSE)
- Define dynamic estimation error: $e(t) = x(t) - \hat{x}(t)$
- Error dynamics:

\[
\dot{e}(t) = \dot{x}(t) - \dot{\hat{x}}(t) = (A - LC)(x(t) - \hat{x}(t)) = (A - LC)e(t)
\]

- Hence, $e(t) \to 0$, as $t \to \infty$ if $\text{eig}(A - LC) < 0$
- **Objective**: design observer/estimator gain $L$ such that $\text{eig}(A - LC) < 0$ or at a certain location
Example — Controller Design

- Given a system characterized by $A = \begin{bmatrix} 1 & 3 \\ 3 & 1 \end{bmatrix}$, $B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

- Is the system stable? What are the eigenvalues?

**Solution:** unstable, $\text{eig}(A) = 4, -2$

- Find linear state-feedback gain $K$ (i.e., $u = -Kx$), such that the poles of the closed-loop controlled system are $-3$ and $-5$

- Characteristic polynomial: $\lambda^2 + (k_1 - 2)\lambda + (3k_2 - k_1 - 8) = 0$

**Solution:** $u = -Kx = -[10 \ 11] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = -10x_1 - 11x_2$

- MATLAB command: $K = \text{place}(A,B,\text{eig}\_\text{desired})$

- What if $A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $B = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, can we stabilize the system?
Given a system characterized by \( A = \begin{bmatrix} 1 & 3 \\ 3 & 1 \end{bmatrix}, C = [0.5 \ 1] \)

Find linear state-observer gain \( L = [l_1 \ l_2]^\top \) such that the poles of the estimation error are \(-5\) and \(-3\)

Characteristic polynomial:
\[
\lambda^2 + (-2 + l_2 + 0.5l_1)\lambda + (-8 + 0.5l_2 + 2.5l_1) = 0
\]

**Solution:** \( L = \begin{bmatrix} 8 \\ 6 \end{bmatrix} \)

MATLAB command: \( L = \text{place}(A',C',\text{eig}_\text{desired}) \)
Observer, Controller Design for DT Systems—Summary

- For CT system
  \[ \dot{x}(t) = Ax(t) + Bu(t), \quad y(t) =Cx(t) + Du(t) \]
  - To design a stabilizing controller, find \( K \) such that
    \[ \text{eig}(A_{cl}) = \text{eig}(A - BK) < 0 \]
    or at a prescribed location
  - To design a converging estimator (observer), find \( L \) such that
    \[ \text{eig}(A_{cl}) = \text{eig}(A - LC) < 0 \]
    or at a prescribed location
- What if the system is DT?
  \[ x(k+1) = Ax(k) + Bu(k), \quad y(k) = Cx(k) + Du(k) \]
  - To design a stabilizing controller, find \( K \) such that
    \[ -1 < \text{eig}(A_{cl}) = \text{eig}(A - BK) < 1 \] or at a prescribed location
  - To design a converging estimator (observer), find \( L \) such that
    \[ -1 < \text{eig}(A_{cl}) = \text{eig}(A - LC) < 1 \] or at a prescribed location
What if the system dynamics are:

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

The observer dynamics will then be:

\[ \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L(y(t) - \hat{y}(t)) \]

Hence, the control input shouldn’t impact the estimation error.

Why? Because the input \( u(t) \) is know!

Estimation error:

\[ e(t) = x(t) - \hat{x}(t) \implies \dot{e}(t) = \dot{x}(t) = \dot{\hat{x}}(t) = (A - LC)(x(t) - \hat{x}(t)) \]

\[ \implies \dot{e}(t) = (A - LC)e(t) \]
MATLAB Example

```matlab
A=[1 -0.8; 1 0];
B=[0.5; 0];
C=[1 -1];
% Selecting desired poles
eig_desired=[.5 .7];
L=place(A',C',eig_desired)';
% Initial state
x=[-10;10];
% Initial estimate
xhat=[0;0];
% Dynamic Simulation
XX=x;
XXhat=xhat;
T=10;
% Constant Input Signal
UU=.1*ones(1,T);
for k=0:T-1,
    u=UU(k+1);
y=C*x;
yhat=C*xhat;
x=A*x+B*u;
xhat=A*xhat+B*u+L*(y-yhat);
XX=[XX,x];
XXhat=[XXhat,xhat];
end
% Plotting Results
subplot(2,1,1)
plot(0:T,[XX(1,:);XXhat(1,:)]);
subplot(2,1,2)
plot(0:T,[XX(2,:);XXhat(2,:) ]);```

©Ahmad F. Taha
Observer-Based Control — 1

- Recall that for LSF control: \( u(t) = -Kx(t) \)
- What if \( x(t) \) is not available, i.e., it can only be estimated?
- **Solution:** get \( \hat{x} \) by designing \( L \)
- Apply LSF control using \( \hat{x} \) with a LSF matrix \( K \) to both the original system and estimator
- **Question:** how to design \( K \) and \( L \) simultaneously? Poles of the closed-loop system?
- This is called an observer-based controller (OBC)
- Design questions: how shall we design \( K \) and \( L \)? Are these designs independent?
Observer-Based Control — 2

Notice that \( u(t) = -K\hat{x}(t) \)
Observer-Based Control — 3

- Closed-loop dynamics:
  \[
  \dot{x}(t) = Ax(t) - BK\hat{x}(t) \\
  \dot{\hat{x}}(t) = A\hat{x}(t) + L(y(t) - \hat{y}(t)) - BK\hat{x}(t)
  \]

- The overall system (observer + controller) can be written as follows:
  \[
  \begin{bmatrix}
  \dot{x}(t) \\
  \dot{\hat{x}}(t)
  \end{bmatrix}
  =
  \begin{bmatrix}
  A & -BK \\
  LC & A - LC - BK
  \end{bmatrix}
  \begin{bmatrix}
  x(t) \\
  \hat{x}(t)
  \end{bmatrix}
  
  \]

- Transformation:
  \[
  \begin{bmatrix}
  x(t) \\
  e(t)
  \end{bmatrix}
  =
  \begin{bmatrix}
  x(t) \\
  x(t) - \hat{x}(t)
  \end{bmatrix}
  =
  \begin{bmatrix}
  I & 0 \\
  I & -I
  \end{bmatrix}
  \begin{bmatrix}
  x(t) \\
  \hat{x}(t)
  \end{bmatrix}
  
  \]

- Hence, we can write:
  \[
  \begin{bmatrix}
  \dot{x}(t) \\
  \dot{e}(t)
  \end{bmatrix}
  =
  \begin{bmatrix}
  A - BK & BK \\
  0 & A - LC
  \end{bmatrix}
  \begin{bmatrix}
  x(t) \\
  e(t)
  \end{bmatrix}
  \]

  \[
  A_{cl}
  \]

- If the system is controllable & observable ⇒ \text{eig}(A_{cl}) can be arbitrarily assigned by proper \( K \) and \( L \).
Separation Principle

\[
\begin{bmatrix}
\dot{x}(t) \\
\dot{e}(t)
\end{bmatrix} = \begin{bmatrix}
A - BK & BK \\
0 & A - LC
\end{bmatrix} \begin{bmatrix}
x(t) \\
e(t)
\end{bmatrix} \equiv \begin{bmatrix}
\dot{x}(t) \\
\dot{\hat{x}}(t)
\end{bmatrix} = \begin{bmatrix}
A & -BK \\
LC & A - LC - BK
\end{bmatrix} \begin{bmatrix}
x(t) \\
\hat{x}(t)
\end{bmatrix}
\]

Notice the above dynamics for the OBC are equivalent.

What are the evals of the closed loop system above?

Since \( A_{cl} \) is block diagonal, the evals of \( A_{cl} \) are

\[\text{eig}(A - BK) \cup \text{eig}(A - LC)\]

- \( \text{eig}(A - BK) \) characterizes the state control dynamics
- \( \text{eig}(A - BK) \) characterizes the state estimation dynamics
- If the system is obsv. AND cont. \( \implies \) evals(\( A_{cl} \)) can be arbitrarily assigned by properly designing \( K \) and \( L \)
- If the system is detect. AND stab. \( \implies \) evals(\( A_{cl} \)) can be stabilized via properly designing \( K \) and \( L \)
OBC Example

Design an OBC (i.e., \( u(t) = -K\hat{x}(t) \)) for the following SISO system

\[
\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t), \quad y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t)
\]

1. Before doing anything, check whether system is cont. (or stab.) and obs. (or det.): **system is cont. AND obs.**
2. First, design a stabilizing state feedback control, i.e., find \( K \) s.t.

\[
eig(A-BK) < 0, A-BK = \begin{bmatrix} 0 & 1 \\ -k_1 & -k_2 \end{bmatrix} \Rightarrow K = \begin{bmatrix} 4 \\ 2 \end{bmatrix} \text{ does the job}
\]

3. Second, design a stabilizing observer (estimator), i.e., find \( L \) s.t.

\[
eig(A-LC) < 0, A-LC = \begin{bmatrix} -l_1 & 1 \\ -l_2 & 0 \end{bmatrix} \Rightarrow L = \begin{bmatrix} 10 \\ 100 \end{bmatrix} \text{ does the job}
\]

4. Finally, overall system design:

\[
\begin{align*}
u(t) &= -K\hat{x}(t) = -4\hat{x}_1(t) - 2\hat{x}_2(t) \\
\dot{\hat{x}}_1(t) &= \hat{x}_2(t) + 10(y(t) - \hat{x}_1(t)) \\
\dot{\hat{x}}_2(t) &= u(t) + 100(y(t) - \hat{x}_1(t))
\end{align*}
\]
Questions And Suggestions?

Any questions?

Thank You!

Please visit engineering.utsa.edu/ataha

IFF you want to know more 😊