Stability

Behavior near an Equilibrium Point.
Stability

Behavior near an Equilibrium Point.
Consider the following dynamical system

\[ \dot{x}(t) = f(x(t), t) \]
\[ x(t_0) = x_0 \]  

(1)

Definition: equilibrium point (pg 45) The state \( x_{eq} \) is an equilibrium point of (1) if it satisfies:

\[ f(x_{eq}, t) = 0 \]  

(2)

for all \( t \geq t_0 \).
Stability of LTI Plants

A motivating example: determine the stability of the origin for the following scalar system

\[ \dot{x}(t) = Ax(t) \]

Equilibrium point: \( x = 0 \)
A motivating example: determine the stability of the origin for the following scalar system

\[ \dot{x}(t) = Ax(t) \]

Equilibrium point: \( x = 0 \)
Can determine the stability of the origin by evaluating eigenvalues of \( A \)

\[ x(t) = e^{A(t-t_0)}x(t_0) \]

\[ A = V\Lambda V^{-1}; \quad V : \text{from eigenvector; } \Lambda : diag(\lambda_i) : \text{from eigenvalues} \]

Stability follows if \( Re(\lambda_i) \leq 0 \)
Asymptotic stability follows if \( Re(\lambda_i) < 0 \).
Stability of LTI Plants

A motivating example: determine the stability of the origin for the following scalar system

\[ \dot{x}(t) = Ax(t) \]

Equilibrium point: \( x = 0 \)
Can determine the stability of the origin by evaluating eigenvalues of \( A \)

\[ x(t) = e^{A(t-t_0)}x(t_0) \]

\[ A = V \Lambda V^{-1}; \quad V : \text{from eigenvector}; \quad \Lambda : \text{diag}(\lambda_i) : \text{from eigenvalues} \]

Stability follows if \( Re(\lambda_i) \leq 0 \)
Asymptotic stability follows if \( Re(\lambda_i) < 0. \)

Lyapunov’s methods allow us to determine the stability of an equilibrium for such a system without solving the differential equation!
Lyapunov Stability

For the system

\[ \dot{x} = f(x) \]

Let

(i) \( V(x) > 0, \ \forall x \neq 0, \ \text{and} \ V(0) = 0 \)

(ii) \( \dot{V}(x) = \left( \frac{\partial V}{\partial x} \right)^T f(x) < 0 \)

(ii) \( V(x) \to \infty \ \text{as} \ \|x\| \to \infty \)

Then \( x = 0 \) is asymptotically stable.
Lyapunov Stability

For the system

\[ \dot{x} = f(x) \]

Let

(i) \( V(x) > 0, \; \forall x \neq 0, \; \text{and} \; V(0) = 0 \)

(ii) \( \dot{V}(x) = \left( \frac{\partial V}{\partial x} \right)^T f(x) < 0 \)

(ii) \( V(x) \to \infty \; \text{as} \; \|x\| \to \infty \)

Then \( x = 0 \) is asymptotically stable.

If instead of (ii), we have

(ii’) \( \dot{V} \leq 0 \)

Then \( x = 0 \) is stable.
Error Model 1

Error Model 1 leads to the following

\[ \dot{x}(t) = A(t)x(t) \quad A(t) = -u(t)u^T(t) \]

Equilibrium point: \( x = 0 \)
Error Model 1

Error Model 1 leads to the following

\[ \dot{x}(t) = A(t)x(t) \quad A(t) = -u(t)u^T(t) \]

Equilibrium point: \( x = 0 \)
Choose a quadratic function

\[ V = \frac{1}{2}x^T x \]
\[ \dot{V} = x^T A(t)x = -x^T u(t)u^T(t)x = -\left(x^T u(t)\right)^2 \leq 0 \]
Error Model 1

Error Model 1 leads to the following

\[ \dot{x}(t) = A(t)x(t) \quad A(t) = -u(t)u^T(t) \]

Equilibrium point: \( x = 0 \)

Choose a quadratic function

\[ V = \frac{1}{2} x^T x \]

\[ \dot{V} = x^T A(t)x = -x^T u(t)u^T(t)x = -(x^T u(t))^2 \leq 0 \]

\[ \Rightarrow \text{stability.} \]
Error Model 1

Error Model 1 leads to the following

\[ \dot{x}(t) = A(t)x(t) \quad A(t) = -u(t)u^T(t) \]

Equilibrium point: \( x = 0 \)

Choose a quadratic function

\[ V = \frac{1}{2}x^T x \]

\[ \dot{V} = x^T A(t)x = -x^T u(t)u^T(t)x = - (x^T u(t))^2 \leq 0 \]

\( \Rightarrow \) stability.

A later lecture will show that if \( u(t) \) is "persistently exciting", \( x(t) \to 0 \).
Error Model 1

Error Model 1 leads to the following

\[ \dot{x}(t) = A(t)x(t) \quad A(t) = -u(t)u^T(t) \]

Equilibrium point: \( x = 0 \)

Choose a quadratic function

\[ V = \frac{1}{2}x^T x \]

\[ \dot{V} = x^T A(t)x = -x^T u(t)u^T(t)x = -\left(x^T u(t)\right)^2 \leq 0 \]

\( \Rightarrow \) stability.

A later lecture will show that if \( u(t) \) is "persistently exciting", \( x(t) \rightarrow 0 \).

We therefore conclude that error model 1 leads to a stable parameter estimation. Asymptotic stability will be shown later.
Adaptive Control of a First-Order Plant

Problem:

Plant:

\[ \dot{x}_p = a_p x_p + k_p u \]

Find \( u \) such that \( x_p \) follows a desired command.
A Model-reference Approach

\( x_p \): Output of a first-order system - can only follow 'smooth' signals

Pose the problem as

\[
\dot{x}_m = a_m x_m + k_m r
\]

Set \( a_m = -5 \) and \( k_m = 5 \).
A Model-reference Approach

\( x_p \): Output of a first-order system - can only follow 'smooth' signals
Pose the problem as

\[
\dot{x}_m = a_m x_m + k_m r
\]

Set \( a_m = -5 \) and \( k_m = 5 \). Or \( a_m = -50, \ k_m = 50 \), Choose \( r \) so that \( x_m \approx x_d \)
Statement of the problem

Choose $u$ so that $e(t) \to 0$ as $t \to \infty$. 

\[
\begin{align*}
&\text{Choose } u \text{ so that } e(t) \to 0 \text{ as } t \to \infty. \\
\end{align*}
\]
Choose $u$ so that $e(t) \to 0$ as $t \to \infty$. \( k_p, a_p \) are unknown.
Certainty Equivalence Principle

Step 1: **Algebraic Part**: Find a solution to the problem when parameters are known.

\[
\begin{align*}
u(t) &= \theta_c x_p + k_c r, \quad \text{choose } \theta_c, k_c \text{ so that closed-loop transfer function matches the reference model transfer function.}
\end{align*}
\]

Desired Parameters:

\[
\begin{align*}
\theta_c &= \theta^*, \\
k_c &= k^* \\
\end{align*}
\]

\[
\begin{align*}
\theta^* &= a_m - a_p k_p \\
k^* &= k_m k_p
\end{align*}
\]
Certainty Equivalence Principle

Step 1: **Algebraic Part:** Find a solution to the problem when parameters are known.

Step 2: **Analytic Part:** Replace the unknown parameters by their estimates. Ensure stable and convergent behavior.

\[ u(t) = \theta c x p + k c r, \text{ choose } \theta c, k c \text{ so that closed-loop transfer function matches the reference model transfer function.} \]

**Desired Parameters:**
\[ \theta = \theta^*, k = k^* \]
\[ \theta^* = a m - a p k p, k^* = k m k p \]
Certainty Equivalence Principle

Step 1: **Algebraic Part:** Find a solution to the problem when parameters are known.

Step 2: **Analytic Part:** Replace the unknown parameters by their estimates. Ensure stable and convergent behavior.

Step 1: \( u(t) = \theta_c x_p + k_c r \), choose \( \theta_c, k_c \) so that closed-loop transfer function matches the reference model transfer function.
Certainty Equivalence Principle

Step 1: **Algebraic Part:** Find a solution to the problem when parameters are known.

Step 2: **Analytic Part:** Replace the unknown parameters by their estimates. Ensure stable and convergent behavior.

Step 1: \( u(t) = \theta_c x_p + k_c r \), choose \( \theta_c, k_c \) so that closed-loop transfer function matches the reference model transfer function.

Desired Parameters: \( \theta_c = \theta^* \), \( k_c = k^* \)

\[
\theta^* = \frac{a_m - a_p}{k_p}, \quad k^* = \frac{k_m}{k_p}
\]
Certainty Equivalence Principle - Step 2

Step 2: **Analytic Part:** Replace the unknown parameters by their estimates. Ensure stable and convergent behavior.

From Step 1, we have

\[ u(t) = \theta^* x_p + k^* r, \quad \theta^* = \frac{a_m - a_p}{k_p}, \quad k^* = \frac{k_m}{k_p} \]
Certainty Equivalence Principle - Step 2

Step 2: Analytic Part: Replace the unknown parameters by their estimates. Ensure stable and convergent behavior.

From Step 1, we have

$$u(t) = \theta^* x_p + k^* r, \quad \theta^* = \frac{a_m - a_p}{k_p}, \quad k^* = \frac{k_m}{k_p}$$

Replace $\theta^*$ and $k^*$ by their estimates $\theta(t)$ and $k(t)$. 
Certainty Equivalence Principle - Step 2

Step 2: **Analytic Part:** Replace the unknown parameters by their estimates. Ensure stable and convergent behavior.

From Step 1, we have

\[ u(t) = \theta^* x_p + k^* r, \quad \theta^* = \frac{a_m - a_p}{k_p}, \quad k^* = \frac{k_m}{k_p} \]

Replace \( \theta^* \) and \( k^* \) by their estimates \( \theta(t) \) and \( k(t) \).

\[ u(t) = \theta(t) x_p + k(t) r \]

\[ \dot{\theta}(t) = ?? \quad \dot{k}(t) = ?? \]
Certainty Equivalence Principle - Step 2

Adaptive control input:

\[ u(t) = \theta(t)x_p + k(t)r \]

Closed-loop Equations:

\[
\dot{x}_p = a_p x_p + k_p u(t)
\]
Certainty Equivalence Principle - Step 2

Adaptive control input:

\[ u(t) = \theta(t)x_p + k(t)r \]

Closed-loop Equations:

\[
\begin{align*}
\dot{x}_p &= a_p x_p + k_p u(t) \\
&= a_p x_p + k_p [\theta(t)x_p + k(t)r]
\end{align*}
\]
Certainty Equivalence Principle - Step 2

Adaptive control input:

\[ u(t) = \theta(t)x_p + k(t)r \]

Closed-loop Equations:

\[
\dot{x}_p = a_p x_p + k_p u(t) \\
= a_p x_p + k_p [\theta(t)x_p + k(t)r] \\
= [a_p + k_p \theta^* - k_p \theta^* + k_p \theta(t)] x_p + k_p [k(t) - k^* + k^*] r
\]
Certainty Equivalence Principle - Step 2

Adaptive control input:

\[ u(t) = \theta(t)x_p + k(t)r \]

Closed-loop Equations:

\[
\begin{align*}
\dot{x}_p &= a_p x_p + k_p u(t) \\
&= a_p x_p + k_p \left[ \theta(t)x_p + k(t)r \right] \\
&= \left[ a_p + k_p\theta^* - k_p\theta^* + k_p \theta(t) \right] x_p + k_p \left[ k(t) - k^* + k^* \right] r \\
&= a_m x_p + k_m r + k_p \tilde{\theta}(t)x_p + k_p \tilde{k}(t)r
\end{align*}
\]
Certainty Equivalence Principle - Step 2

Adaptive control input:

\[ u(t) = \theta(t) x_p + k(t) r \]

Closed-loop Equations:

\[
\begin{align*}
\dot{x}_p &= a_p x_p + k_p u(t) \\
&= a_p x_p + k_p \left[ \theta(t) x_p + k(t) r \right] \\
&= \left[ a_p + k_p \theta^* - k_p \theta^* + k_p \theta(t) \right] x_p + k_p \left[ k(t) - k^* + k^* \right] r \\
&= a_m x_p + k_m r + k_p \tilde{\theta}(t) x_p + k_p \tilde{k}(t) r
\end{align*}
\]

Reference Model:

\[ \dot{x}_m = a_m x_m + k_m r \]
Certainty Equivalence Principle - Step 2

Adaptive control input:

\[ u(t) = \theta(t) x_p + k(t) r \]

Closed-loop Equations:

\[ \dot{x}_p = a_p x_p + k_p u(t) \]
\[ = a_p x_p + k_p [\theta(t) x_p + k(t) r] \]
\[ = [a_p + k_p \theta^* - k_p \theta^* + k_p \theta(t)] x_p + k_p [k(t) - k^* + k^*] r \]
\[ = a_m x_p + k_m r + k_p \tilde{\theta}(t) x_p + k_p \tilde{k}(t) r \]

Reference Model:

\[ \dot{x}_m = a_m x_m + k_m r \]

Error model:

\[ \dot{e} = a_m e + k_p \tilde{\theta}(t) x_p + k_p \tilde{k}(t) r \]
Certainty Equivalence Principle - Step 2

Step 2: Analytic Part: Replace the unknown parameters by their estimates. Ensure stable and convergent behavior.

From Step 1, we have

\[ u(t) = \theta^* x_p + k^* r, \quad \theta^* = \frac{a_m - a_p}{k_p}, \quad k^* = \frac{k_m}{k_p} \]
Certainty Equivalence Principle - Step 2

Step 2: Analytic Part: Replace the unknown parameters by their estimates. Ensure stable and convergent behavior.
From Step 1, we have

\[ u(t) = \theta^* x_p + k^* r, \quad \theta^* = \frac{a_m - a_p}{k_p}, \quad k^* = \frac{k_m}{k_p} \]

Replace \( \theta^* \) and \( k^* \) by their estimates \( \theta(t) \) and \( k(t) \).
Certainty Equivalence Principle - Step 2

Step 2: **Analytic Part:** Replace the unknown parameters by their estimates. Ensure stable and convergent behavior.

From Step 1, we have

\[ u(t) = \theta^* x_p + k^* r, \quad \theta^* = \frac{a_m - a_p}{k_p}, \quad k^* = \frac{k_m}{k_p} \]

Replace \( \theta^* \) and \( k^* \) by their estimates \( \theta(t) \) and \( k(t) \).

\[ u(t) = \theta(t) x_p + k(t) r \]

\[ \dot{e} = a_m e + k_p \tilde{\theta}(t) x_p + k_p \tilde{k}(t) r \]
Certainty Equivalence Principle - Step 2

Step 2: **Analytic Part**: Replace the unknown parameters by their estimates. Ensure stable and convergent behavior.

From Step 1, we have

\[ u(t) = \theta^* x_p + k^* r, \quad \theta^* = \frac{a_m - a_p}{k_p}, \quad k^* = \frac{k_m}{k_p} \]

Replace \( \theta^* \) and \( k^* \) by their estimates \( \theta(t) \) and \( k(t) \).

\[
\begin{align*}
\dot{e} &= a_m e + k_p \tilde{\theta}(t) x_p + k_p \tilde{k}(t) r \\
\omega &= \begin{bmatrix} x_p \\ r \end{bmatrix}, \quad \tilde{\theta} = \begin{bmatrix} \tilde{\theta}(t) \\ \tilde{k}(t) \end{bmatrix}
\end{align*}
\]
Error Model 3

\[ \dot{e} = a_m e + k_p \tilde{\theta}(t) x_p + k_p \tilde{k}(t) r \]
Error Model 3

\[ \dot{e} = a_m e + k_p \tilde{\theta}(t) x_p + k_p \tilde{k}(t) r \]

\[ = a_m e + k_p \tilde{\theta}^T(t) \omega \]

\[ \omega = \begin{bmatrix} x_p \\ r \end{bmatrix}, \quad \tilde{\theta} = \begin{bmatrix} \tilde{\theta}(t) \\ \tilde{k}(t) \end{bmatrix} \]
Error Model 3

\[ \dot{e} = a_m e + k_p \tilde{\theta}(t) x_p + k_p \tilde{k}(t) r \]

\[ = a_m e + k_p \tilde{T}^{T}(t) \omega \]

\[ \omega = \begin{bmatrix} x_p \\ r \end{bmatrix}, \quad \tilde{\theta} = \begin{bmatrix} \tilde{\theta}(t) \\ \tilde{k}(t) \end{bmatrix} \]