Feedback Control Systems
Lecture 3 State Feedback

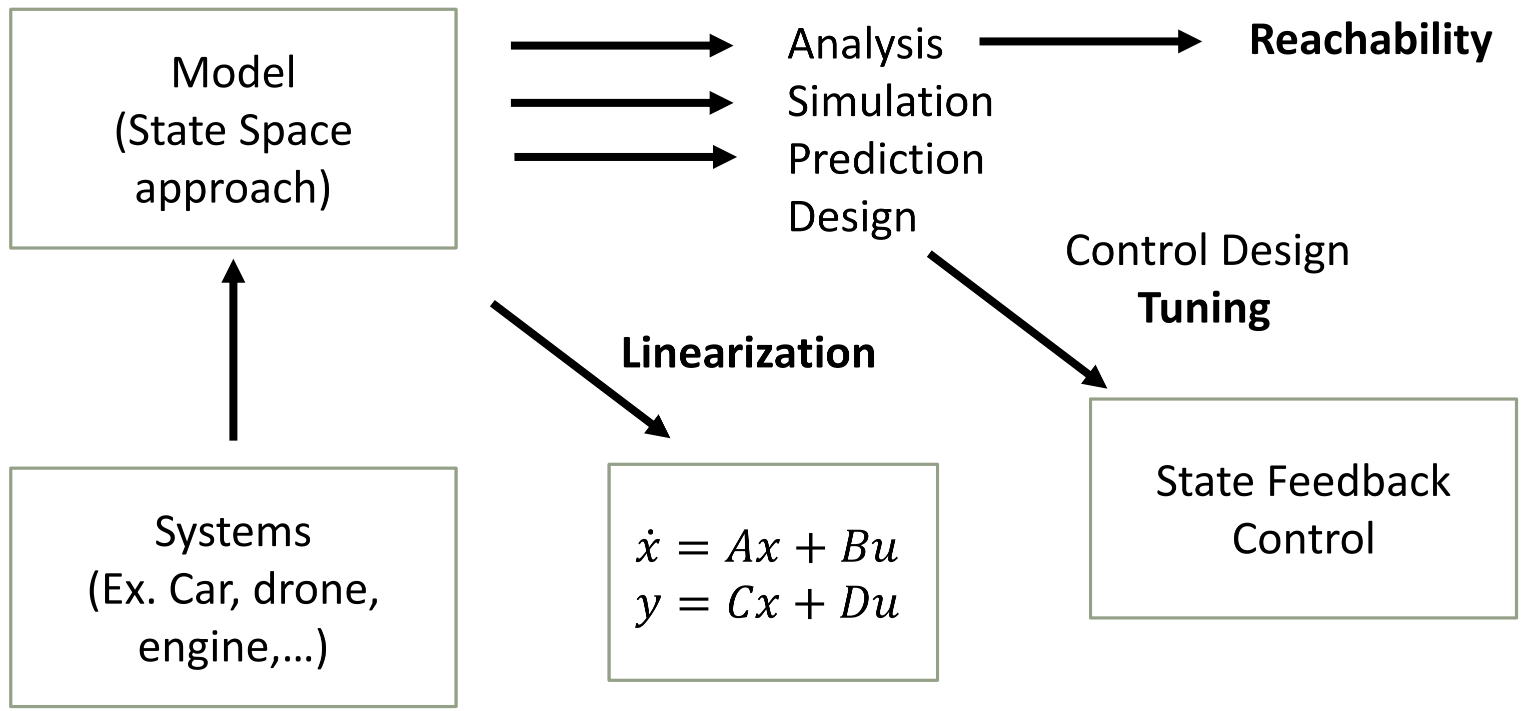
Instructor: Daro VAN

Laboratory of Dynamics and Control
Department of Industrial and Mechanical Engineering
Institute of Technology of Cambodia

Content

- State Feedback Control Design
- Linearization
- Reachability
- Tuning
- References

State Feedback Control Design



State Feedback Control Design

Concepts

- We made an assumption that the model is linear which is not always the case. However, we can transform a nonlinear model to a linear one using **linearization**.
- The system might not be **reachable**. In order to do the design, we need to make sure that the system is reachable.
- **Tuning** is the important part of the design process since the system's behavior depends on the tuning.

State Feedback Control Design

Consider a linear time invariant state space model given by

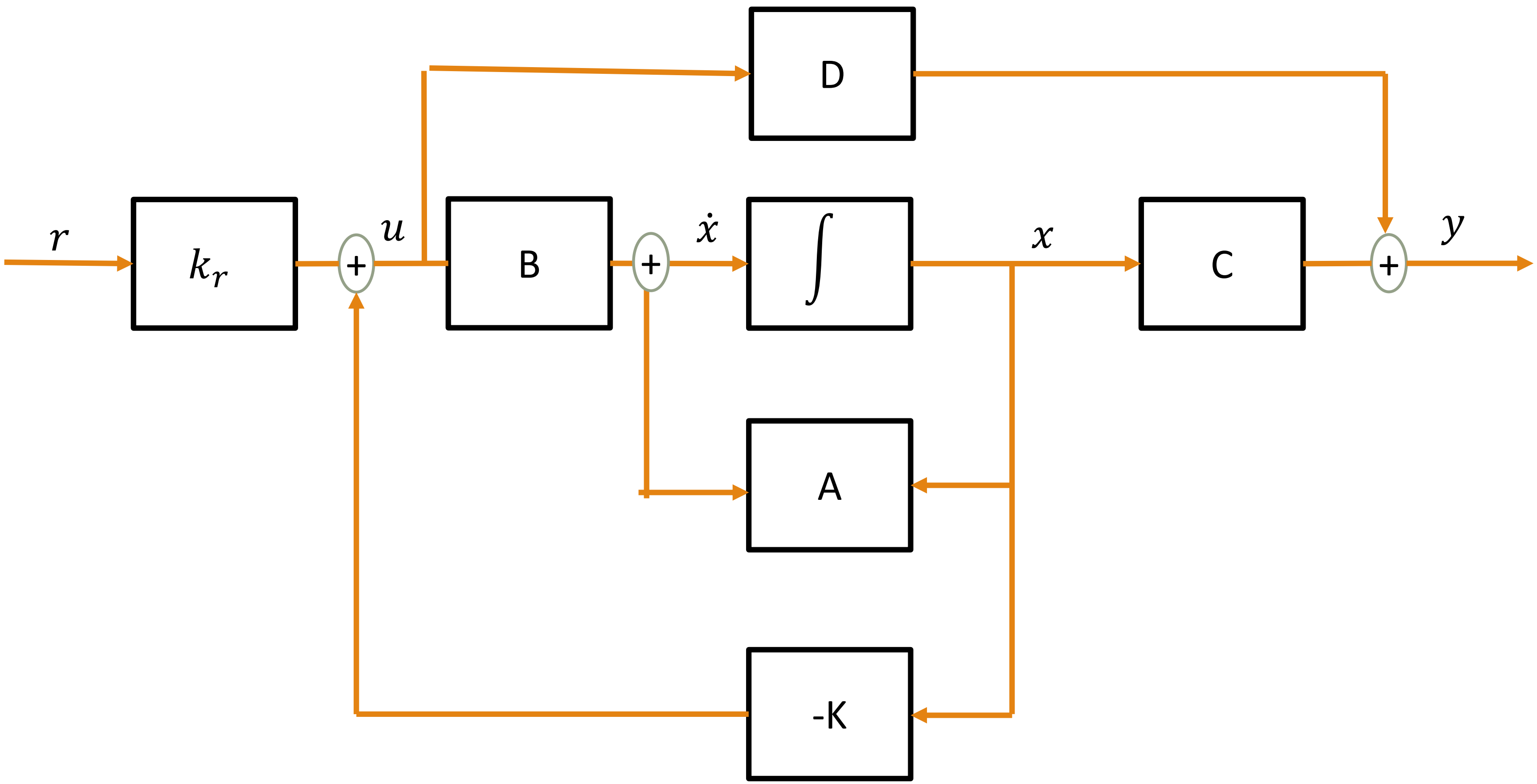
$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y = Cx(t) + Du(t)$$

Where $x(t) \in \mathbb{R}^n$ are state vector, $u(t) \in \mathbb{R}^p$ is the control input, $y \in \mathbb{R}^q$ is the output signal.

Idea with control design: Modify the eigenvalues of A by using the input $u(t)$

State feedback controller : $u(t) = -Kx(t) + k_r r(t)$
Where $K \in \mathbb{R}^{p\times n}$ is feedback gain, $k_r \in \mathbb{R}^{p\times r}$ is the steady state reference gain, and $r(t) \in \mathbb{R}^r$ is the reference input.

State Feedback Control Design



State Feedback Control Design

Using the state feedback controller, the closed loop dynamics becomes:

$$\dot{x}(t) = Ax(t) + B\big(-Kx(t) + k_r r(t)\big)$$
$$= (A - BK)x(t) + Bk_r r(t)$$

Control objective : Choose K such that the closed loop systems $(A - BK)$ get desired properties: Fulfill the specification and stabilize the system.

Note: The steady state reference gain does not affect the stability, but it affect the steady state solution.

The steady state gain is usually chosen such that

$$y(t) \approx r(t) as\ t\ \rightarrow \infty$$

State Feedback Control Design

At steady state, the time derivate of the state variable is $\dot{x}(t) \equiv 0$. Assume that $D = 0$, then

$$\left.\begin{aligned} 0 &= (A - BK)x(t) + Bk_r r(t) \\ y(t) &= Cx(t) \end{aligned}\right\} \quad y = \boxed{-C(A - BK)^{-1}B} k_r r$$

If $y(t) \approx r(t)$ as $t \to \infty$, then k_r should be chosen as

$$k_r = -(C(A - BK)^{-1}B)^{-1} \text{ or } k_r = -1/C(A - BK)^{-1}B$$

Design of the control law becomes 2 steps:

- **Shape the dynamics** by choosing K.
- **Set the steady state level** by choosing k_r. If we have perfect knowledge of the system, we don't get the steady state error (very rare). Unfortunately, most often then not, there is an error since we made some assumption when we do modelling.

State Feedback Control Design

Integral action

Using the steady state feedback gain, k_r, we can achieve zero steady state error, but it does depend on the model parameters, as

$$k_r = -(C(A - BK)^{-1}B)^{-1} \text{ or } k_r = -1/C(A - BK)^{-1}B$$

We introduce integral action to remove the steady state error.

Approach: introduce an additional state variable in our system which computes the integral of the error

$$\dot{z}(t) = y(t) - r(t)$$

The augmented state space model becomes:

$$\begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} Ax + Bu \\ y - r \end{bmatrix} = \begin{bmatrix} Ax + Bu \\ Cx - r \end{bmatrix} = \begin{bmatrix} A & 0 \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ -1 \end{bmatrix} r$$

Given the new state-space model, we design a controller in the usual fashion and the resulting controller becomes

$$u(t) = -Kx(t) - K_I z(t) + k_r r(t)$$

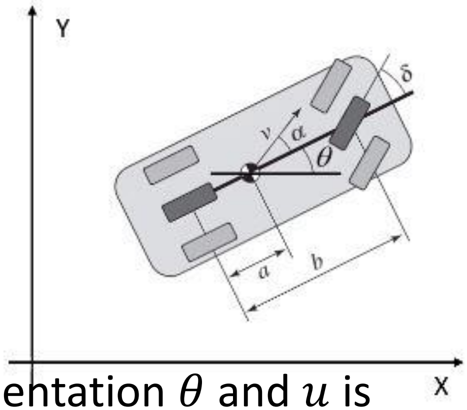
State Feedback Control Design

Example

Consider the following system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & v_0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} a v_0 / b \\ v_0 / b \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0] u$$



Where x_1 is the lateral position Y, x_2 is the heading orientation θ and u is the steering angle δ

The idea is to design a controller that **stabilizes** the dynamics and **tracks** a given lateral position of the vehicle.

Specification: Desired characteristic polynomial

$$p_{des}(\lambda) = \lambda^2 + 2\zeta\omega_n\lambda + \omega_n^2$$

Note: The details of the vehicle steering model can be found in the in the reference book (Ex. 2.8, Ex. 5.12, Ex. 6.4, Ex. 7.4)

State Feedback Control Design

Example

State feedback control:

$$u = -Kx + k_r r = -k_1 x_1 - k_2 x_2 + k_r r$$

The closed loop system dynamics becomes

$$\begin{aligned}\dot{x} &= (A - BK)x + Bk_r r \\ &= \begin{bmatrix} -k_1 a v_0 / b & v_0 - k_2 a v_0 / b \\ -k_1 v_0 / b & -k_2 v_2 / b \end{bmatrix} x + \begin{bmatrix} k_r a v_0 / b \\ k_r\, v_0 / b \end{bmatrix} r \\ y &= Cx + Du = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\end{aligned}$$

The closed loop system has the characteristic polynomial

$$\det(\lambda I - A + BK) = \cdots = \lambda^2 + \frac{v_0}{b}(ak_1 + k_2)\lambda + \frac{k_1 v_0^2}{b}$$

Matching with desired characteristic polynomial gives:

$$k_1 = \frac{b\omega_n^2}{v_0^2}, k_2 = \frac{2\zeta\omega_n b}{v_0} - \frac{ab\omega_n^2}{v_0^2}$$

State Feedback Control Design Example

The steady-state gain can be determined by

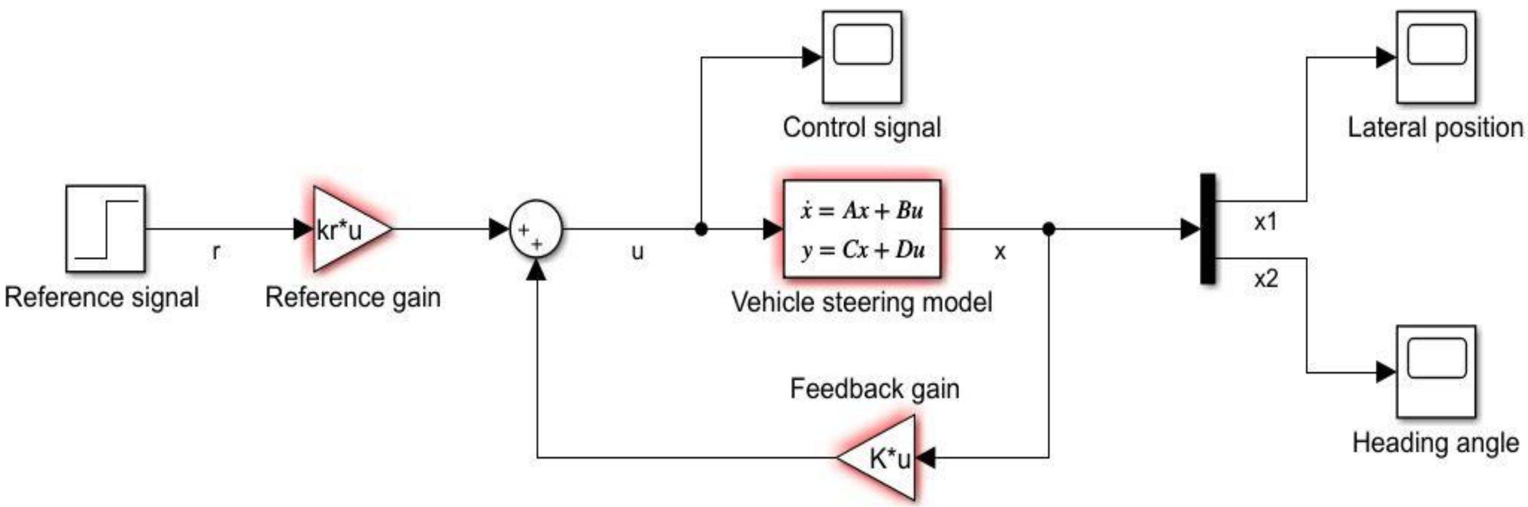
$$k_r = -\frac{1}{C(A-BK)^{-1}B} = \cdots = k_1 = \frac{b\omega_n^2}{v_0^2}$$

Inserting these control design parameters into the feedback controller gives:

$$u = -k_1x_1 - k_2x_2 + k_rr = \boxed{-\frac{b\omega_n^2}{v_0^2}}x_1 \boxed{-\left(\frac{2\zeta\omega_n b}{v_0} - \frac{ab\omega_n^2}{v_0^2}\right)}x_2 + \boxed{\frac{b\omega_n^2}{v_0^2}}r$$

State Feedback Control Design

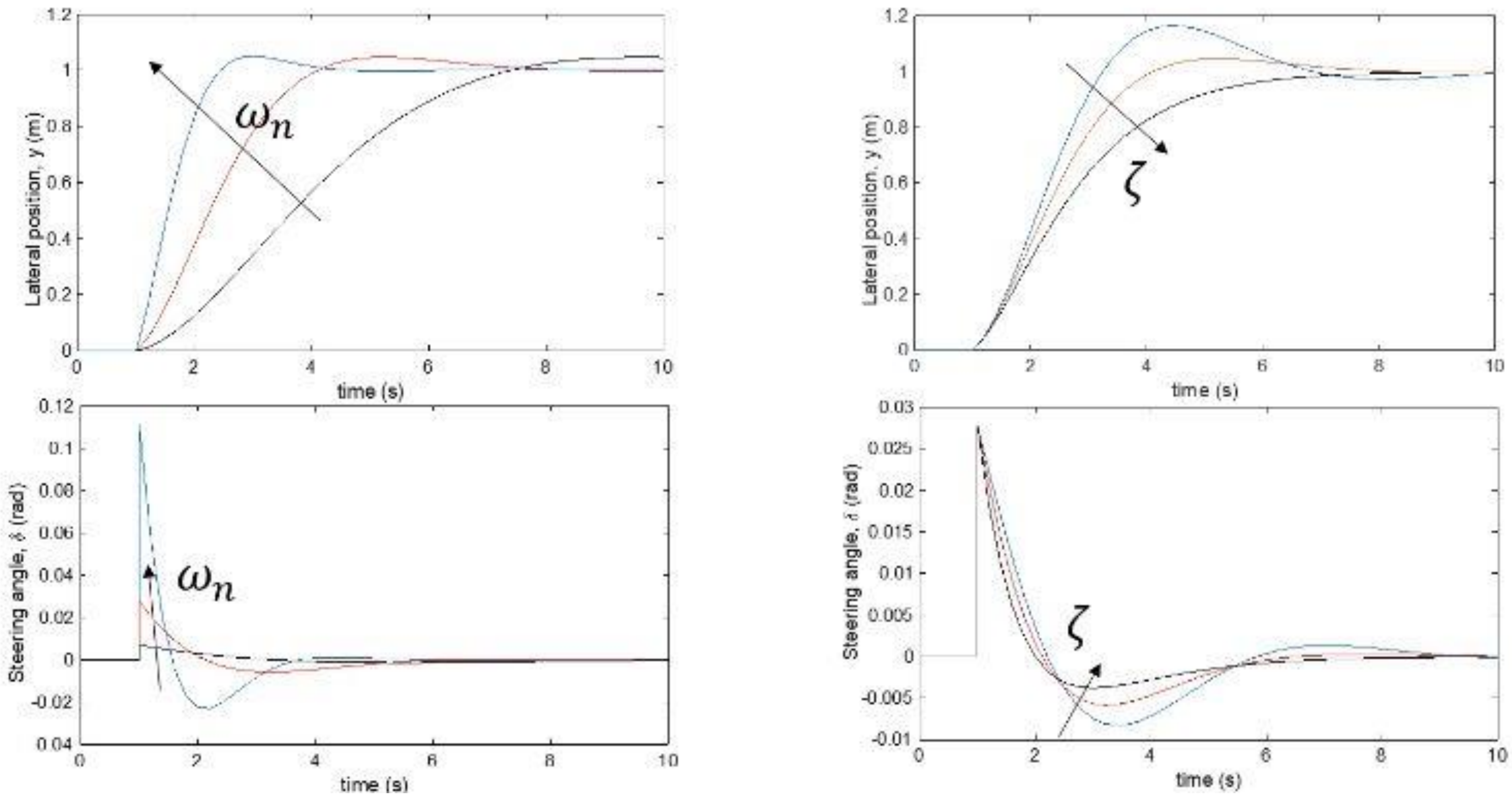
Example



In the given zip file, run the *Example_VehicleSteering_PP.m.* This will load all the parameters used in the Simulink. Then run the *VehicleSteering_sim. Tune the value of* ζ *and* ω_n. What can you observe?

State Feedback Control Design

Example



State Feedback Control Design
Example

- If we increase ω_n, the system response faster, the rise time decrease and the settling time also decreases. However, it increases the control action.

- If we increase ζ, the overshoot decreases, but the rise time increases. The settling time is approximately the same.

Note: Both ω_n and ζ affect the rising time. But the overshoot is only controlled by ζ.

State Feedback Control Design

Example

Consider the following system (similar to Example 1 with one difference)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]u$$

The closed loop system has the characteristic polynomial

$$\det(\lambda I - A + BK) = \det\begin{pmatrix} \lambda + k_1 & k_2 - 1 \\ 0 & \lambda \end{pmatrix} = \lambda^2 + k_1\lambda = \underbrace{\lambda}_{\text{unstable}}(\lambda + k_1)$$

While matching with the desired characteristic polynomial

$$\lambda^2 + 2\zeta\omega_n\lambda + \omega_n^2 = \lambda(\lambda + k_1)$$

We notice that it is not possible to make them identical to each other.
It is not possible to control all the eigenvalues. We say that the system is not **controllable** or not **reachable**. Furthermore, one eigenvalue is always 0, which means that the closed loop system is unstable.

State Feedback Control Design

Model
State Space
approach

$$\dot{x} = f(x, u, d)$$
$$y = h(x, u, d)$$

Linearization

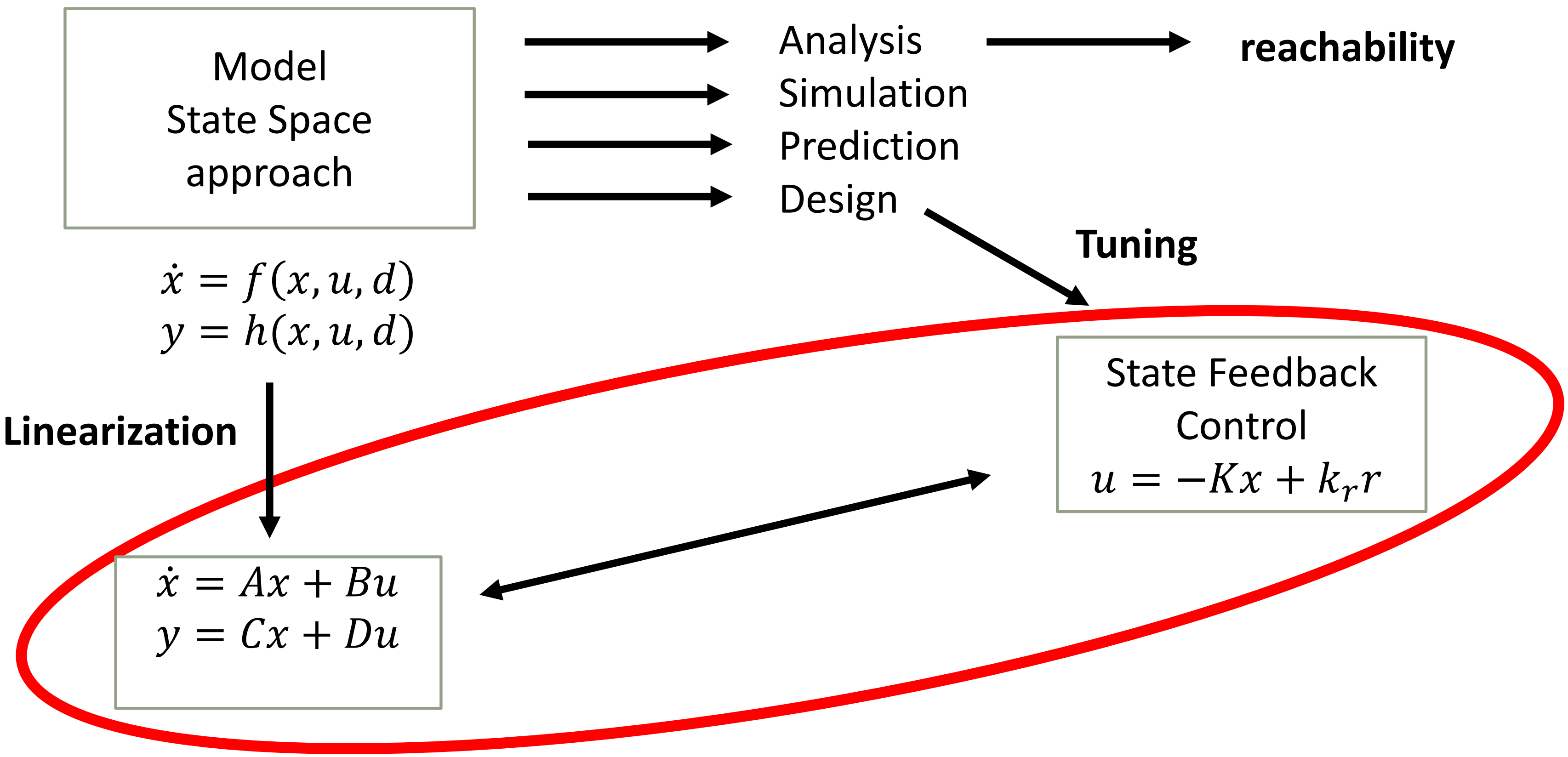
$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

Analysis → **reachability**
Simulation
Prediction
Design

Tuning

State Feedback
Control
$u = -Kx + k_r r$

Linearization



Linearization

From modeling, we often get a nonlinear state-space model

$$\dot{x} = f(x, u)$$
$$y = h(x, u)$$

Where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^p$ is the input or control signal, and $y \in \mathbb{R}^q$ is the output signal. (For SISO case, p = 1, q=1)

Idea: Study the local behavior of the system around the equilibrium point (x_e, u_e).

An **equilibrium point** is the steady state solution to the differential equation:

$$\dot{x} = f(x_e, u_e) = 0$$

ie. If the system starts at x_e (with u_e as input signal) it will remain there.

Linearization

Consider a point close to the equilibrium point $(x = x_e + \Delta x, u = u_e + \Delta u)$

The dynamics close to the equilibrium point is described by the differential equation:

$$\frac{d}{dt}(x_e + \Delta x) = f(x_e + \Delta x, u_e + \Delta u)$$

The linear model can now be developed from a Taylor series expansion of $f(.)$ around the equilibrium point (ignoring higher order terms):

$$\frac{d}{dt}(x_e + \Delta x) = f(x_e + \Delta x, u_e + \Delta u) \approx f(x_e, u_e) + \left.\frac{\partial f}{\partial x}\right|_{x_e,u_e} \Delta x + \left.\frac{\partial f}{\partial u}\right|_{x_e,u_e} \Delta u$$

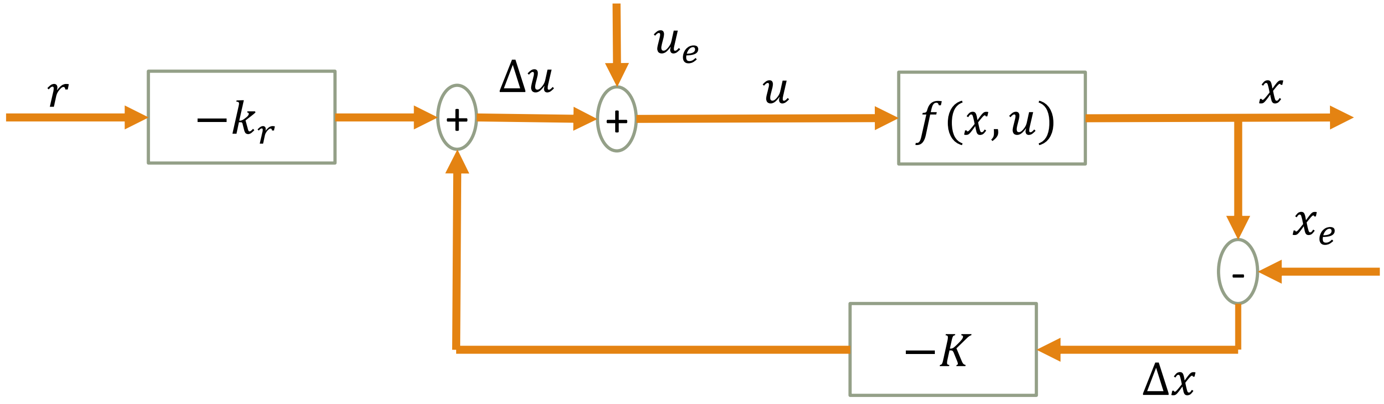
$$\frac{d}{dt}\Delta x = \underbrace{\left.\frac{\partial f}{\partial x}\right|_{x_e,u_e}}_{A} \Delta x + \underbrace{\left.\frac{\partial f}{\partial u}\right|_{x_e,u_e}}_{B} \Delta u$$

Linearization

So, to conclude, the linear system becomes:

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x,u) \end{aligned} \quad \Rightarrow \quad \begin{aligned} \Delta\dot{x} &= A\Delta x + B\Delta u \\ \Delta y &= C\Delta x + D\Delta u \end{aligned}$$

Where $\Delta x = x - x_e$ is the state, $\Delta u = u - u_e$ is the input or control signal
And $\Delta y = y - y_e$ is the output signal.



Linearization

A leaf spring system is often used on heavy vehicles.

https://en.wikipedia.org/wiki/Leaf_spring

A model of a leaf spring system is:

$$m\ddot{x} = -k_1 x \; - k_2 x^3$$

The model can be written in state-space form as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ -\dfrac{k_1}{m} x_1 - \dfrac{k_2}{m} x_1^3 \end{bmatrix} = \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{bmatrix}$$

Where x_1 is the horizontal position of the chassis and x_2 is its velocity.

Note: leaf spring made of a number of strip of metal curved slightly upwards and clamped together one above anther. It is commonly used in suspension systems of trucks, van, commercial vehicles.

Linearization

The equilibrium point can be found by solving the algebraic equation:

$$\begin{aligned} f_1(x_1, x_2) &= 0 \\ f_2(x_1, x_2) &= 0 \end{aligned} \quad \Rightarrow \quad \begin{aligned} &x_{2e} = 0 \\ &-\frac{k_1}{m} x_{1e} - \frac{k_2}{m} x_{1e}^3 = 0 \end{aligned}$$

Which gives the following equilibrium points:

$$\begin{aligned} &x_{2e} = 0 \\ &x_{1e} = 0, and\ x_{1e} = \pm\sqrt{-k_1/k_2} \end{aligned}$$

Compute the partial derivatives at the equilibrium point $(x_{1e}, x_{2e}) = (0{,}0)$

$$\left.\frac{\partial f_1}{\partial x_1}\right|_{x_e} = 0 \quad , \qquad \left.\frac{\partial f_1}{\partial x_2}\right|_{x_e} = 1$$

$$\left.\frac{\partial f_2}{\partial x_1}\right|_{x_e} = -\frac{k_1}{m} - \frac{3k_2}{m} x_{1e}^2 = -\frac{k_1}{m}, \qquad \left.\frac{\partial f_2}{\partial x_2}\right|_{x_e} = 0$$

Linearization

So, our linear model becomes

$$\begin{bmatrix} \Delta\dot{x}_1 \\ \Delta\dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k_1/m & 0 \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \end{bmatrix}$$

Linearization

Model
State Space
approach

Analysis → **reachability**
Simulation
Prediction
Design

Tuning

$$\dot{x} = f(x, u, d)$$
$$y = h(x, u, d)$$

Linearization

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

State Feedback
Control
$$u = -Kx + k_r r$$

Reachability

Consider again the following system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]u$$

State feedback control: $u = -Kx + k_r r = -k_1 x_1 - k_2 x_2 + k_r r$

The closed loop system dynamics becomes

$$\dot{x} = (A - BK)x + Bk_r r = \begin{bmatrix} -k_1 & 1 - k_2 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} k_r \\ 0 \end{bmatrix} r$$

And the characteristic polynomial:

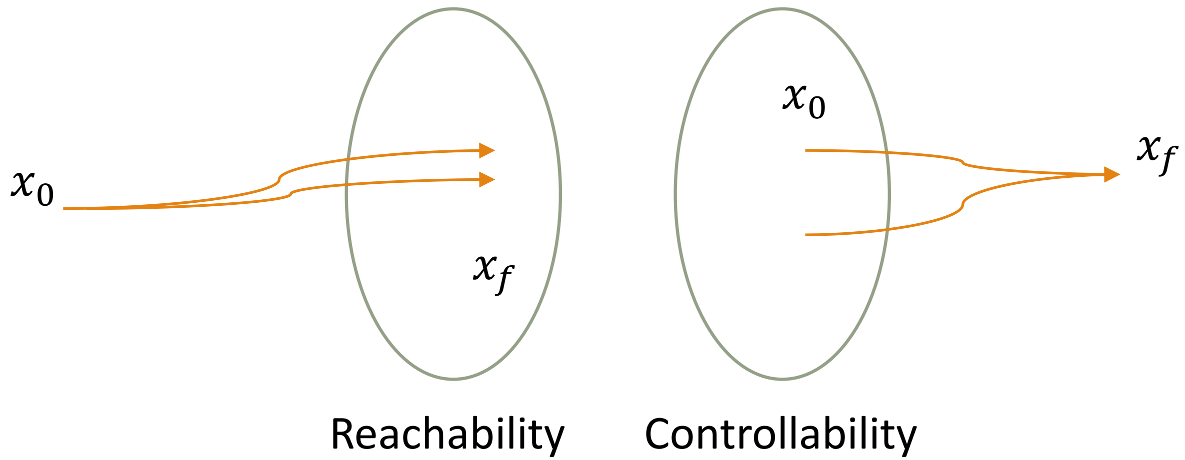
Controllable

$$\det(\lambda I - A + BK) = \lambda^2 + k_1 \lambda = \lambda(\lambda + k_1)$$

Uncontrollable

Reachability

Definition (Reachability): A linear system is **reachable** if for any $x_0, x_f \in \mathbb{R}^n$, there exists a $T > 0$ and u: $[0, T] \to \mathbb{R}$ such that if $x(0) = x_0$, then the corresponding solution satisfies $x(T) = x_f$.



Note: The definition doesn't tell us the time it takes to reach or the amount of control action needed.

Reachability

To see that an arbitrary point can be reached, we can use the convolution equation.

Assume that the system starts from zero, the state of a linear system is given by

$$x(t) = \int_0^t e^{A(t-\tau)} B u(\tau) d\tau = \int_0^t e^{A\tau} B u(t-\tau) d\tau$$

From linear theory, it can be shown that

$$e^{A\tau} = I\alpha_0(\tau) + A\alpha_1(\tau) + \cdots + A^{n-1}\alpha_{n-1}(\tau)$$

Where $\alpha_i(t)$ are scalar functions, so that

$$x(t) = B\int_0^t \alpha_0(\tau) u(t-\tau) d\tau + AB\int_0^t \alpha_1(\tau) u(t-\tau) d\tau + \cdots + A^{n-1}B\int_0^t \alpha_{n-1}(\tau) u(t-\tau) d\tau$$

Reachability

By writing it in vector form, we get

$$x(t) = \underbrace{\begin{bmatrix} B & AB & \dots & A^{n-1}B \end{bmatrix}}_{W_r} \begin{bmatrix} \int_0^t \alpha_0(\tau)u(t-\tau)d\tau \\ \int_0^t \alpha_1(\tau)u(t-\tau)d\tau \\ \vdots \\ \int_0^t \alpha_{n-1}(\tau)u(t-\tau)d\tau \end{bmatrix}$$

To reach an arbitrary point, we require that W_r is nonsingular. The matrix W_r is called the **reachability matrix**.

Theorem (Reachability rank condition): A linear system is reachable if and only if the reachability matrix is invertible (has full rank).

Reachability

Return to our example, with the following systems:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]u$$

Reachability matrix

$$W_r = [B \quad AB] = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

The system Is not reachable.

Note: a square matrix $M(n \times n)$ has full rank n iff the det(M)$\neq 0$

Reachability

Consider the following system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & v_0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} av_0/b \\ v_0/b \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]u$$

Reachability matrix

$$W_r = \begin{bmatrix} B & AB \end{bmatrix} = \begin{bmatrix} av_0/b & v_0^2/b \\ v_0/b & 0 \end{bmatrix}$$

Compute the determinant:

$$\det(W_r) = \begin{vmatrix} av_0/b & v_0^2/b \\ v_0/b & 0 \end{vmatrix} = -v_0^3/b^2 \neq 0$$

The system is reachable, as long as $v_0 \neq 0$.

Reachability

Model
State Space
approach

$$\dot{x} = f(x, u, d)$$
$$y = h(x, u, d)$$

Linearization

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

Analysis → **reachability**
Simulation
Prediction
Design

Tuning

State Feedback
Control
$u = -Kx + k_r r$

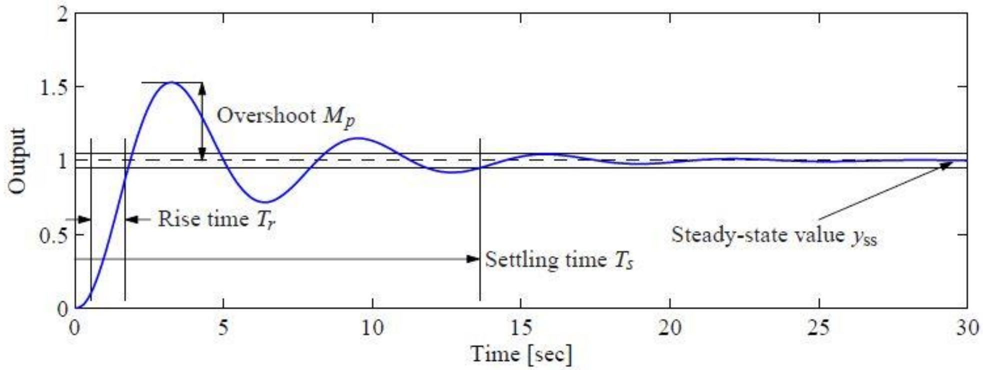
Tuning
Pole Placement

So far we have learnt how a state feedback looks like and when it is possible to design a state feedback controllers to stabilize a system

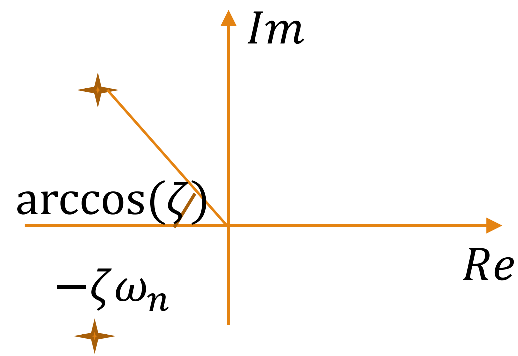
$$\dot{x} = Ax + Bu$$
$$u = -Kx + k_r r$$

$$\dot{x} = (A - BK)x + Bk_r r$$

The questions that remains are: How do we design a state feedback controller and where do we place the closed loop system's poles



$\Leftrightarrow$

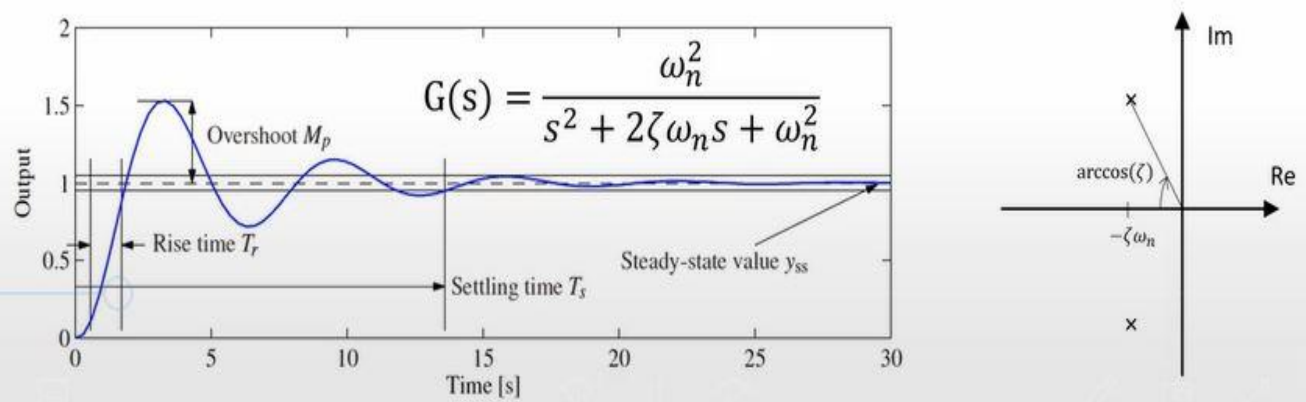


Note: The **rise time** is related to the **real part of the poles**. The further into the negative left half plane the poles are placed, the faster the system will be. The overshoot is related to the imaginary part of the poles. The closer to the imaginary axis, the more it oscillates.

Tuning

Specifications and Pole Placement of Second Order System

Property	Value	$\zeta = 0.5$	$\zeta = 1/\sqrt{2}$	$\zeta = 1$
Rise time	$T_r \approx 1/\omega_n \cdot e^{arccos\zeta/\tan(arccos\zeta)}$	$1.8/\omega_n$	$2.2/\omega_n$	$2.7/\omega_n$
Overshoot	$M_p \approx e^{-\pi\zeta\sqrt{1-\zeta^2}}$	16%	4%	0%
Settling time (2%)	$T_s \approx 4/\zeta\omega_n$	$8.0/\omega_n$	$5.9/\omega_n$	$4.0/\omega_n$



For second order system, the relationship between the poles represented as ζ and ω_n and the time domain specification can be approximately determined.

Tuning

Pole Placement

Where do we place the closed loop system's poles?

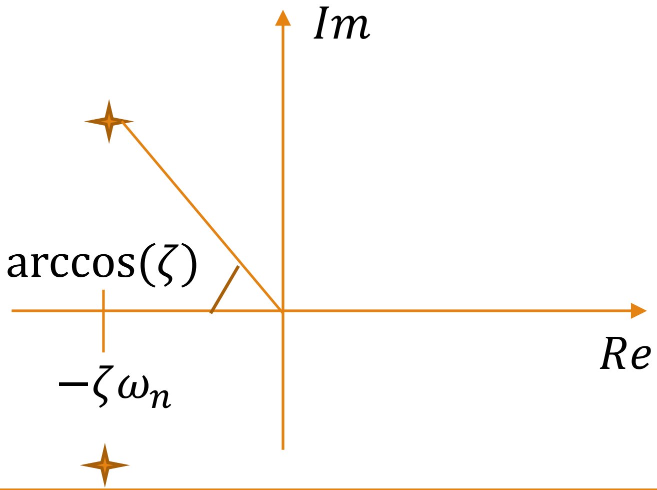
Idea:

- Use time domain specifications to place the dominant poles, as a second order system, $s^2 + 2\zeta\omega_n s + \omega_n^2$.
- Place the rest of the poles so they become faster than the dominant poles (2 or 3 times faster).

(The dominant second order poles, guarantee the time specification.)

Usually you end up with some zeros as well:

- Zeros in the left half plane give additional overshoot.
- Zero in the right half plane give a negative undershoot.



Tuning
Pole placement(Ackermann's formula)

Pole placement is performed by matching the desired characteristic polynomial with the closed loop system's characteristic polynomial.

From earlier example (vehicle steering) we have seen:

$$\lambda^2 + 2\zeta\omega_n\lambda + \omega_n^2 = \lambda^2 + \frac{v_0}{b}(ak_1 + k_2)\lambda + \frac{k_1 v_0^2}{b}$$

$$k_1 = \frac{b\omega_n^2}{v_0^2} \qquad k_2 = \frac{2\zeta\omega_n b}{v_0} - \frac{ab\omega_n^2}{v_0^2}$$

For low order systems, it is fine, but for larger systems, this is a boring task.

Ackermann's formula offers us a method to do this in one computational step.

Tuning

Pole placement(Ackermann's formula)

Consider a system $\dot{x} = Ax + Bu$ with the characteristic polynomial

$$a(s) = s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n$$

If the system is reachable, then there exist a control law, $u = -Kx$, that gives a closed loop system with the characteristic polynomial

$$p(s) = s^n + p_1 s^{n-1} + \cdots + p_{n-1} s + p_n$$

Tuning
Pole placement(Ackermann’s formula)

The feedback gain is given by

$$K = [p_1 - a_1 \quad p_2 - a_2 \quad \dots \quad p_n - a_n]\widetilde{W}_r W_r^{-1}$$

Where W_r is the reachability matrix

$$W_r = [B \quad AB \quad \dots \quad A^{n-1}B]$$

And

$$\widetilde{W}_r = \begin{bmatrix} 1 & a_1 & a_2 & \cdots & a_{n-1} \\ 0 & 1 & a_1 & \cdots & a_{n-2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & a_1 \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}^{-1}$$

This is called **Ackermann’s formula**.

Tuning

Pole placement(Ackermann's formula)

Consider the vehicle steering example again. The system is given as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & v_0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} av_0/b \\ v_0/b \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]u$$

Determine the characteristic polynomial for the system:

$$\det(sI - A) = \begin{vmatrix} s & -v_0 \\ 0 & s \end{vmatrix} = s^2 = s^2 + 0s + 0$$

Desired characteristic polynomial for the closed loop system:

$$p(s) = s^2 + 2\zeta\omega_n s + \omega_n^2$$

Tuning

Pole placement(Ackermann's formula)

The feedback gain is given by

$$K = [p_1 - a_1 \quad p_2 - a_2]\widetilde{W}_r W_r^{-1} = [2\zeta\omega_n \;\; \omega_n^2]\widetilde{W}_r W_r^{-1}$$

Where W_r is the reachability matrix

$$W_r = [B \quad AB] = \begin{bmatrix} \frac{av_0}{b} & \frac{v_0^2}{b} \\ \frac{v_0}{b} & 0 \end{bmatrix} \qquad \widetilde{W}_r = \begin{bmatrix} 1 & a_1 \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The feedback gain can be computed as

$$K = [2\zeta\omega_n \quad \omega_n^2]\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} \frac{av_0}{b} & \frac{v_0^2}{b} \\ \frac{v_0}{b} & 0 \end{bmatrix}^{-1} = \begin{array}{c} \begin{array}{cc} k_1 & \qquad\qquad k_2 \end{array} \\ \begin{bmatrix} \frac{b\omega_n^2}{v_0^2} & \frac{2\zeta\omega_n b}{v_0} - \frac{ab\omega_n^2}{v_0^2} \end{bmatrix} \end{array}$$

Tuning

Pole placement(Ackermann’s formula)

Note

- Pole placement design is suitable for SISO systems since the number of degree of freedom is the same as that of the parameters.
- For MIMO systems, the number of parameters will be more than that of the degree of freedom. So, we might end up with suboptimal solutions.

Tuning

Linear Quadratic Regulator

An alternative method to pole placement is to place the poles so that the closed loop system optimizes a cost function

$$J = \int_0^\infty (x^T Q_x x + u^T Q_u u) dt$$

Where $x^T Q_x x$ is the **state cost** and $u^T Q_u u$ is the **control cost**. The matrices Q_x and Q_u are symmetric, positive (semi-) definite matrices. This is called the **linear quadratic regulator (LQR) problem**.

The solution to the LQR problem is given by

$$u = -Kx, \qquad K = Q_u^{-1} B^T S$$

Where S is a solution of

$$A^T S + SA - SBQ_u^{-1} B^T S + Q_x = 0$$

This equation is called the **algebraic Riccati equation**

Tuning

Linear Quadratic Regulator

The tuning of the LQR is to choose the weighted matrices Q_x and Q_u. To guarantee that a solution exists, the system **must be reachable** and that $Q_x \succcurlyeq 0$ and $Q_u \succ 0$.

1. Simplest choice: $Q_x = I$ and $Q_u = \rho I$

$$J = \int_0^\infty (x^T x + \rho u^T u) dt \;\Rightarrow\; \text{trade} - \text{off} \quad \|x\|^2 \text{ vs } \rho\|u\|^2$$

This reduces the tuning to select ρ, which then becomes a trade-off between state cost and control cost. If ρ is large, then we have small control action and vice versa.

2. Output weighting . Let $z = C_z x$ be the output you want to keep small. choose $Q_x = C_z^T C_z$, and $Q_u = \rho I$. $\Rightarrow$ trade-off $\Rightarrow \|z\|^2 \; vs \; \rho\|u\|^2$

Tuning

Linear Quadratic Regulator

3. Diagonal weighting

$$Q_x = \begin{bmatrix} q_1 & & 0 \\ & \ddots & \\ 0 & & q_n \end{bmatrix}, Q_u = \begin{bmatrix} \rho_1 & & 0 \\ & \ddots & \\ 0 & & \rho_p \end{bmatrix}$$

Choose the individual diagonal elements based on how each state or input signal should contribute to the overall cost.

Alternative, (Bryson's rule) choose the diagonal weight as $q_i = \alpha_i^2 / x_{i,max}^2$ and $\rho_i = \beta_i^2 / u_{i,max}^2$, where $x_{i,max}$ and $u_{i,max}$ represents the largest response. α and β are used for additional individual weighting of the state and control cost.

$$\sum_{i=1}^{n} \alpha_i^2 = 1 \qquad , \qquad \sum_{i=1}^{p} \beta_i^2 = 1$$

4. Trail and error

Tuning

Linear Quadratic Regulator

Consider again the following system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & v_0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} av_0/b \\ v_0/b \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]u$$

Vehicle data: $v_0 = 12\ m/s$, $a = 2m, b = 4m$.

Place the poles so that the closed loop system optimizes the cost function:

$$J = \int_0^{\infty} (x^T x + \rho u^T u) dt$$

Where $Q_x = \begin{bmatrix} q_1 & 0 \\ 0 & q_2 \end{bmatrix}, Q_u = \rho$

Tuning
Linear Quadratic Regulator

For the case when

$$Q_x = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, Q_u = 10$$

The solution to the algebraic Ricatti equation

$$A^T S + SA - SBQ_u^{-1}B^T S + Q_x = 0$$

Is

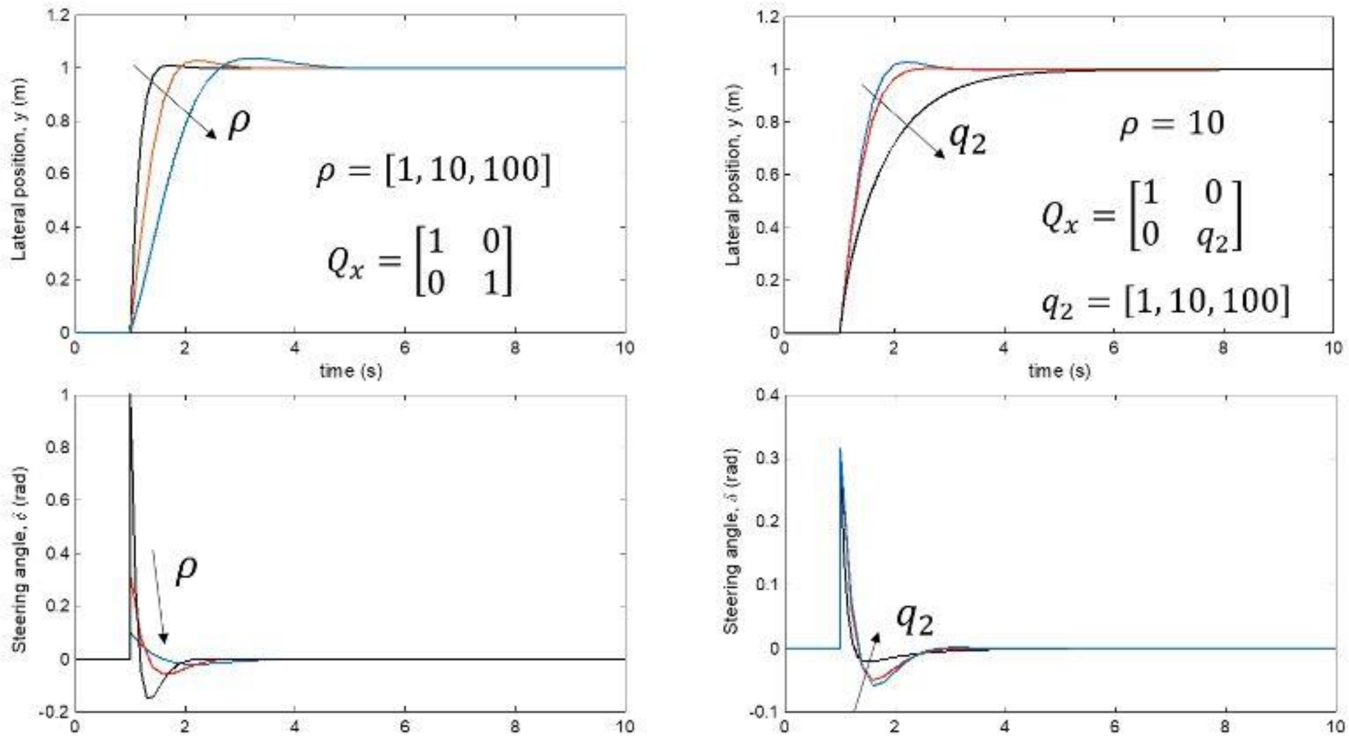
$$S = \begin{bmatrix} 0.292 & 0.470 \\ 0.470 & 2.754 \end{bmatrix}$$

And the corresponding control law becomes

$$u = -Kx, K = Q_u^{-1}B^T S = [0.316 \quad 1.108]$$

In MATLAB: The LQR problem can be solved using the lqr command

Tuning



When we increase the value of ρ, the response is slower. If we keep the value of ρ and change the weight of the heading angle, the response become slower and slower. You can play around with the Simulink and tune the weighting.

Tuning

The closed loop system poles are

$$E = \begin{bmatrix} -2.6110 + 2.1371i \\ -2.6110 - 2.1371i \end{bmatrix}$$

Compared to the pole placement design, this corresponds to $\zeta = 0.77$ and $\omega_n = 3.44$.

Tuning

Model
State Space
approach

$$\dot{x} = f(x, u, d)$$
$$y = h(x, u, d)$$

Linearization

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

Analysis
Simulation
Prediction
Design

reachability

Tuning

Control Design
(State Feedback)
$u = -Kx + k_r r$

Reference

Materials used in this lecture draws heavily from:

- The Lectures on Model-Based Automotive Systems Engineering taught professor Jonas Fredriksson, Chalmers University of Technology.
- The book, entitled Feedback Systems: An Introduction for Scientists and Engineering by Karl Johan Astrom and Richard M. Murray.