Systems Analysis and Control

Matthew M. Peet
Arizona State University

Lecture 2: Systems Defined by Differential Equations

Introduction

In this Lecture, you will learn:

1. Quantitative illustration of the benefits of Feedback
2. Some basics of modeling using differential equations.
3. The State-Space Framework.

PART 1: An Example of Control without Dynamics

Just for Motivation: Makes it easier to solve for the output

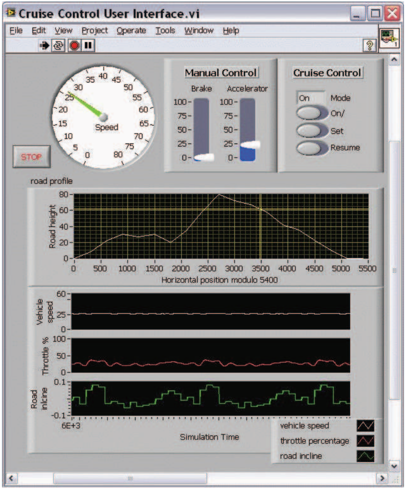
CRUISE CONTROL

Plant: The Automobile (Car)

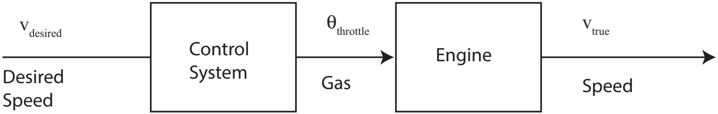
- **Input:** Throttle Position, $\theta_{throttle}$.
- **Output:** Real Velocity, $\mathbf{v_{true}}$.
- **Dynamics:** No Dynamics! Speed is proportional to throttle (proportional gain).

$$\underbrace{\mathbf{v_{true}}}_{\text{output}} = \underbrace{10}_{\text{Plant}} \cdot \underbrace{\theta_{throttle}}_{\text{input}}$$

The *gain factor* is $10\ mph/°$.



First lets start with open loop control



Actuator: Throttle

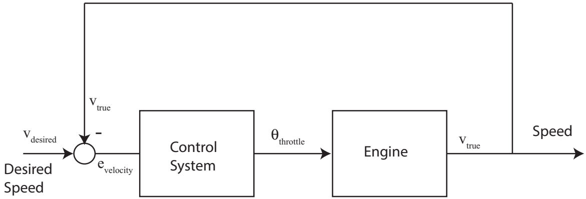
Controller:

- **Input**: Desired Velocity, $\mathbf{v_{desired}}$.
- **Output**: Throttle, $\theta_{throttle}$.

Open Loop Controller We use a simple controller based on our knowledge of the plant.

$$\underbrace{\theta_{throttle}}_{\text{output}} = \underbrace{\frac{1}{10}}_{\text{controller}} \underbrace{\mathbf{v_{desired}}}_{\text{input}}$$

Now lets try using closed loop control



Actuator: Throttle

Sensor: Speedometer

Look at the **CONTROLLER:**

- **Input:** Error in Velocity, $e_{velocity} = \mathbf{v_{true}} - \mathbf{v_{desired}}$.
- **Output:** Throttle, $\theta_{throttle}$.

Proportional Feedback: Amplify the error signal by a scalar *gain* k.

$$\underbrace{\theta_{throttle}}_{\text{output}} = -k \cdot e_{velocity} = \underbrace{-k}_{\text{control}} \cdot \underbrace{(\mathbf{v_{true}} - \mathbf{v_{desired}})}_{\text{input}}$$

Closed Loop vs. Open Loop (Solving for $\mathbf{v_{true}}$)

Open Loop: Two equations:

$$\mathbf{v_{true}} = 10 \cdot \theta_{throttle} \qquad \text{and} \qquad \theta_{throttle} = \frac{1}{10}\mathbf{v_{desired}}$$

Combining, we get

$$\mathbf{v_{true}} = 10\frac{1}{10}\mathbf{v_{desired}} = \mathbf{v_{desired}}.$$

Open-loop control has **No Error!**

Closed Loop: Two equations:

$$\mathbf{v_{true}} = 10 \cdot \theta_{throttle} \qquad \text{and} \qquad \theta_{throttle} = -k\left(\mathbf{v_{true}} - \mathbf{v_{desired}}\right).$$

Combining these, we get $\mathbf{v_{true}} = -10 \cdot k(\mathbf{v_{true}} - \mathbf{v_{desired}})$.
Lets **Choose** $k = 10$.

$$\mathbf{v_{true}} = \frac{10 \cdot k}{1 + 10 \cdot k}\mathbf{v_{desired}} = \frac{100}{101}\mathbf{v_{desired}} = .99\mathbf{v_{desired}}.$$

Closed-loop control has **1% Error**.

Impact of Error and Disturbances

Comparison:

- Open Loop: No error
- Closed Loop: Small error
  - Error gets very small if $k \to \infty$, since

$$\mathbf{v_{true}} = \frac{10 \cdot k}{1 + 10 \cdot k}\mathbf{v_{desired}} \to \mathbf{v_{desired}}.$$

Question: Why use feedback?

- **Answer:** Life is Messy.

Problems:

- Modeling Errors: What if our model is wrong by 10%, so

$$\mathbf{v_{true}} = 11 \cdot \theta_{throttle}$$

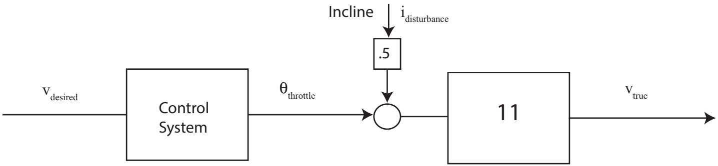
- Disturbances: An Incline of $i_{disturbance}$ degrees will reduce the throttle by $.5/^\circ$.

$$\Delta\theta_{throttle} = -.5 \cdot i_{disturbance}$$

Impact of Error and Disturbances

Open Loop

Let $\mathbf{v_{desired}} = 50mph$, $i_{disturbance} = -1°$.
Now Recalculate the *Open Loop* Output : $\mathbf{V_{true}}$



$$\mathbf{v_{true}} = 11(\theta_{throttle} - .5 \cdot i_{disturbance})$$

$$\theta_{throttle} = \frac{1}{10}\mathbf{v_{desired}} = 5$$

we have

$$\mathbf{v_{true}} = 11(5 + .5) = 60.5mph$$

Which is **NOT ACCEPTABLE!!!**.

Impact of Error and Disturbances

Closed Loop

Recalculate the *Closed Loop* output : $\mathbf{V_{true}}$

- **Plant with Disturbance:** $\mathbf{v_{true}} = 11 \cdot (\theta_{throttle} - .5 \cdot i_{disturbance})$
- **Controller:** $\theta_{throttle} = -k\left(\mathbf{v_{true}} - \mathbf{v_{desired}}\right) = -k(\mathbf{v_{true}} - 50)$

Combine these equations and solve for $\mathbf{v_{true}}$.

$$\mathbf{v_{true}} = 11(-k \cdot \mathbf{v_{true}} + 50 \cdot k + .5) = -11 \cdot k \cdot \mathbf{v_{true}} + 11 \cdot 50 \cdot k + 5.5$$

Solving for $\mathbf{v_{true}}$ yields

$$\mathbf{v_{true}} = \frac{11k + .11}{1 + 11k} 50 = \frac{110.11}{111} 50 = .991 \cdot 50 = 49.6mph$$

PART 2: A Brief Review of Modeling

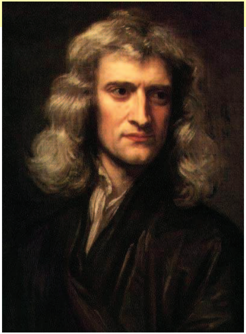
Ordinary Differential Equations (ODEs)

Models can be

- static ($x = Ku$).
- dynamic ($\dot{x} = -x + u$).

Physics-based Modeling

- Mechanics and circuits define ODEs
- Identify *states* (position, voltage, et c.)
- Identify governing differential equations.

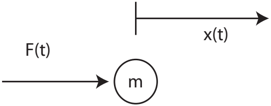


Newton invented ODE models in 1684:

- Newton's Second Law: (x is position)

$$\frac{d^2}{dt^2}x(t) = F(t)/m$$

- $x(t)$ is a state (can also be a signal (output))
- $F(t)$ is a signal (input)



Nonlinear Differential Equations:

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

Where

- This is a first-order differential equation
- $u(t)$ is the Input
- $y(t)$ is the Output
- $x(t)$ is the *state variable*.
  - position, heading, velocity, etc.
- f, g are functions (possibly vector-valued).

Usually, our equations of motion will be Linear. e.g.

$$\ddot{x}(t) = b\dot{x}(t) + ax(t)$$

where

- a and b are constants.

Linear equations are better because

- The Laplace Transform exists (Lecture 4)
- Stability is easy
  - $\dot{x} = ax$ is stable if $a < 0$ and unstable if $a \geq 0$.

Nonlinear Equations should be Linearized (Lecture 3)!!!

Review: Equations of Motion

Higher Orders or Multiple Variables

Types of EOM:

Coupled Differential Equations:

$$\dot{x} = ax + bz$$
$$\dot{z} = cx + dz$$

- The motion of x affects the motion of z and vice-versa.

Higher Order Derivatives:

$$\dddot{x} = a\ddot{x} + b\dot{x} + cx$$

- Commonly obtained from Newton's Second law.
  - Or anything with inertia....

$$F = m\ddot{x}$$

AKA

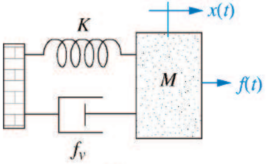
$$\ddot{x}(t) = \frac{1}{m}F(t).$$

Dynamic Model: Suspension System

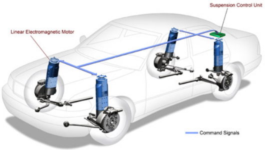
Mass-Spring Model

We wish to study the motion of the vehicle subject to disturbances.

- Model the car as a solid mass
- Control the vertical motion of the car ($x(t)$)



(a)



Inputs: Force, $f(t)$.
Outputs: Displacement, $y(t) = x(t)$.

Definition 1.

A system with one input and one output is Single-Input, Single-Output (**SISO**). A system with more than one input *or* more than one output is Multi-Input Multi-Output (**MIMO**)

Plant Dynamics: Equations of Motion

- Spring Force: Opposes motion in x with spring constant K.

$$F_s(t) = -Kx(t)$$

- Damper Force: Opposes motion in $\dot{x}$ with damping coefficient c

$$F_d(t) = -c\dot{x}(t)$$

- Newton's Second Law:

$$m\ddot{x}(t) = F_s(t) + F_d(t) + f(t)$$

System Model:

$$\begin{aligned}\ddot{x}(t) &= -\frac{K}{m}x(t) - \frac{c}{m}\dot{x}(t) + \frac{1}{m}f(t)\\ y(t) &= x(t)\end{aligned}$$

Standard Forms

Frequency Domain

Once we have our dynamic model

$$\ddot{x}(t) = -\frac{K}{m}x(t) - \frac{c}{m}\dot{x}(t) + \frac{1}{m}f(t) \qquad \text{Differential Equations}$$

$$y(t) = x(t) \qquad \text{Output Equation}$$

This model can be expressed in two standard forms

- Transfer Function
- State-Space

We will discuss these in more depth soon. For now:

Transfer Function: Apply the Laplace Transform to both equations and solve for the output.

$$s^2\hat{x}(s) = -\frac{K}{m}\hat{x}(s) - \frac{c}{m}s\hat{x}(s) + \frac{1}{m}\hat{f}(s) \qquad \text{Differential Equations}$$

$$\hat{y}(s) = \hat{x}(s) \qquad \text{Output Equation}$$

which yields

$$\hat{y}(s) = \frac{1}{ms^2 + cs + K}\hat{f}(s)$$

Suspension System with Wheel Dynamics

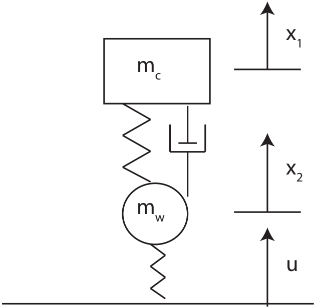
More Detailed Model

Now, we add the dynamics of the wheel.

There are two states:

States:

- Vehicle Position, x_1
- Wheel Position, x_2



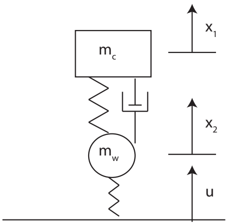
Our Input is the position of the surface of the road.

Inputs:

- Road Surface, u

Suspension Model: Free Body 1

This time we write the dynamics of both the wheel and the car.

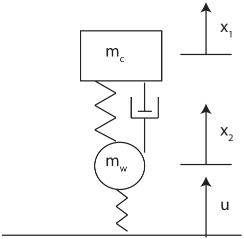


Car Dynamics: Equations of Motion

- Spring 1 Force on Car: $F_{s1,c}(t) = -K_1(x_1(t) - x_2(t))$
- Damper Force on Car: $F_{d,c}(t) = -c(\dot{x}_1(t) - \dot{x}_2(t))$
- Newton's Second Law:

$$\begin{aligned} m_c\ddot{x}_1(t) &= F_{s1,c}(t) + F_{d,c}(t) \\ &= -K_1(x_1(t) - x_2(t)) - c(\dot{x}_1(t) - \dot{x}_2(t)) \end{aligned}$$

Suspension Model: Free Body 2



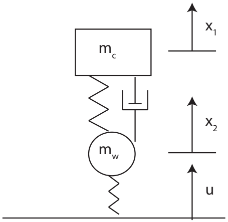
Wheel Dynamics: Equations of Motion

- Spring 1 Force on Wheel: $F_{s1,w}(t) = K_1(x_1(t) - x_2(t))$
- Spring 2 Force on Wheel: $F_{s2,w}(t) = -K_2(x_2(t) - u(t))$
- Damper Force on Wheel: $F_{d,w}(t) = c(\dot{x}_1(t) - \dot{x}_2(t))$
- Newton's Second Law:

$$\begin{aligned} m_w \ddot{x}_2(t) &= F_{s1,w}(t) + F_{s2,w}(t) + F_{d,w}(t) \\ &= K_1(x_1(t) - x_2(t)) - K_2(x_2(t) - u(t)) + c(\dot{x}_1(t) - \dot{x}_2(t)) \end{aligned}$$

Equations of Motion

Combining the dynamics, we get the coupled system dynamics.



$$
\begin{aligned}
m_w \ddot{x}_2(t) &= K_1(x_1(t) - x_2(t)) - K_2(x_2(t) - u(t)) + c(\dot{x}_1(t) - \dot{x}_2(t)) \\
m_c \ddot{x}_1(t) &= -K_1(x_1(t) - x_2(t)) - c(\dot{x}_1(t) - \dot{x}_2(t)) \\
y(t) &= \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}
\end{aligned}
$$

This is quite complicated.

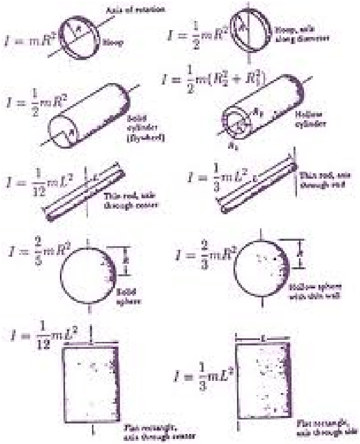
- To simplify, we would like to use a *Standard Form*.

Newton's Second Law Applied to Rigid Bodies

The rate of change of angular momentum is given by

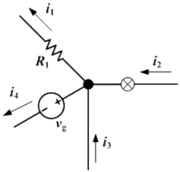
$$\sum M(t) = I\alpha(t) = I\ddot{\theta}(t)$$

- $\alpha(t) = \ddot{\theta}(t)$ is the angular acceleration.
- $\theta(t)$ is the state (angular displacement)
- I is the moment of inertia.
- $M(t)$ is the torque (moment).



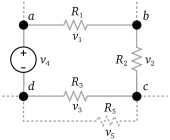
Kirchhoff's Current Law (KCL):
Current is conserved at each junction

$$\sum_k i_k(t) = 0$$



Kirchhoff's Voltage Law (KVL): Net Voltage change around any loop is zero.

$$\sum_k V_k(t) = 0$$



These are combined with standard voltage laws such as voltage drop across a resister, inductor and capacitor:

$$V_r(t) = Ri_r(t) \qquad \frac{d}{dt}i_L(t) = \frac{1}{L}V_L(t) \qquad \frac{d}{dt}V_c(t) = \frac{1}{C}i_c(t)$$

PART 3: State-Space Formulation

A Standard Form for writing Diff. Eqns.

State-Space is a way of writing first order differential equation using matrices.

$$\dot{\vec{x}}(t) = A\vec{x}(t)$$

where $\vec{x}$(t) is a vector and $A \in \mathbb{R}^{n \times n}$ is a square matrix.

Example:

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 1 \\ 2 & 0 & 0 \\ 0 & -1 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

Is equivalent to writing the three differential equations

$$\dot{x}_1 = -x_1 + x_3$$
$$\dot{x}_2 = 2x_1$$
$$\dot{x}_3 = -x_2 + x_3$$

Writing equations in state-space has many advantages

Review: Equations of Motion

Multiple Variables and State-Space

Consider the system

$$\dot{x} = ax + by$$
$$\dot{y} = cx + dy$$

When we have multiple coupled equations: **Convert to State-Space:**

$$\frac{d}{dt}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}$$

Which is easily expressed as

$$\dot{\mathbf{x}} = A\mathbf{x}$$

where

- $\mathbf{x}$ is a vector.
- A is a matrix.

State-Space Form for Systems

Definition 2.

State-Space Form is a convenient way of representing *linear* multivariate or MIMO systems using 4 matrices.

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

- u is the vector of **Inputs**.
- y is the vector of **Outputs**.
- x is the **State**.

$u \in \mathbb{R}^m$, $y \in \mathbb{R}^p$, and $x \in \mathbb{R}^n$ can be vectors of any dimension. However, the matrices must be compatable (the right size):

$$A \in \mathbb{R}^{n \times n} \qquad B \in \mathbb{R}^{n \times m}$$
$$C \in \mathbb{R}^{p \times n} \qquad D \in \mathbb{R}^{p \times m}$$

- $u \in \mathbb{R}^m$ means u is a real vector of length m.
- $C \in \mathbb{R}^{p \times n}$ means C is a matrix with p rows and n columns.

When we have higher order derivatives,

$$\dddot{x}(t) = a\dot{x}(t) + bx(t) + u(t)$$
$$y(t) = x(t) + u(t)$$

we can put it in state-space form by

- Introducing new variables.

Procedure:

- Define a new variable for every derivative term except for the the highest order one.
  - e.g. Let $x_1 = x$, $x_2 = \dot{x}$ and $x_3 = \ddot{x}$.
- Add a new first order differential equation for each new variable.
  - e.g. $\dot{x}_1 = x_2$ and $\dot{x}_2 = x_3$
- Then put in state-space form.

So, for example, we would have 3 equations

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = x_3(t)$$
$$\dot{x}_3(t) = ax_2(t) + bx_1(t) + u(t)$$

Putting Things in State-Space Form

Using our first-order equations:

$$\dot{x}_1(t) = x_2(t); \qquad \dot{x}_2(t) = x_3(t)$$
$$\dot{x}_3(t) = a x_2(t) + b x_1(t) + u(t) \qquad y(t) = x_1(t) + u(t)$$

We construct the matrix representation:

$$\dot{x}(t) = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ b & a & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(t)$$
$$y(t) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 1 \end{bmatrix} u(t)$$

So that

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ b & a & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$
$$C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \qquad D = \begin{bmatrix} 1 \end{bmatrix}$$

State-Space Form: Suspension System

Recall the dynamics:

$$m_w\ddot{x}_2(t) = K_1(x_1(t) - x_2(t)) - K_2(x_2(t) - u(t)) + c(\dot{x}_1(t) - \dot{x}_2(t))$$
$$m_c\ddot{x}_1(t) = -K_1(x_1(t) - x_2(t)) - c(\dot{x}_1(t) - \dot{x}_2(t))$$
$$y(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

Define the new variables z_i

$$z_1(t) = x_1(t) \qquad z_2(t) = \dot{x}_1(t) \qquad z_3(t) = x_2(t) \qquad z_4(t) = \dot{x}_2(t)$$

Which yields the following set of equations: $y(t) = \begin{bmatrix} z_1(t) \\ z_3(t) \end{bmatrix}$,

$$\dot{z}_1(t) = z_2(t)$$
$$\dot{z}_2(t) = -\frac{K_1}{m_c}(z_1(t) - z_3(t)) - \frac{c}{m_c}(z_2(t) - z_4(t))$$
$$\dot{z}_3(t) = z_4(t)$$
$$\dot{z}_4(t) = \frac{K_1}{m_w}(z_1(t) - z_3(t)) - \frac{K_2}{m_w}(z_3(t) - u(t)) + \frac{c}{m_w}(z_2(t) - z_4(t))$$

Constructing State-Space Systems

$$\dot{z}_1(t) = z_2(t)$$

$$\dot{z}_2(t) = -\frac{K_1}{m_c} z_1(t) - \frac{c}{m_c} z_2(t) + \frac{K_1}{m_c} z_3(t) + \frac{c}{m_c} z_4(t)$$

$$\dot{z}_3(t) = z_4(t)$$

$$\dot{z}_4(t) = \frac{K_1}{m_w} z_1(t) + \frac{c}{m_w} z_2(t) - \left(\frac{K_1}{m_w} + \frac{K_2}{m_w}\right) z_3(t) - \frac{c}{m_w} z_4(t) + \frac{K_2}{m_w} u(t)$$

$$y(t) = \begin{bmatrix} z_1(t) \\ z_3(t) \end{bmatrix}$$

$$\begin{bmatrix} \dot{z}_1(t) \\ \dot{z}_2(t) \\ \dot{z}_3(t) \\ \dot{z}_4(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{K_1}{m_c} & -\frac{c}{m_c} & \frac{K_1}{m_c} & \frac{c}{m_c} \\ 0 & 0 & 0 & 1 \\ \frac{K_1}{m_w} & \frac{c}{m_w} & -\left(\frac{K_1}{m_w} + \frac{K_2}{m_w}\right) & -\frac{c}{m_w} \end{bmatrix} \begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \\ z_4(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{K_2}{m_w} \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \\ z_4(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u(t)$$

Summary

What have we learned today?

A Static Model of Cruise-Control

- Simple static model and Control
- Open Loop Control
- Closed Loop Control
- Benefits of Feedback

Dynamic Models

- Including Inputs and Outputs
- Using Newton's Laws
- MIMO and SISO systems
- Other sources of models (Kirchhoff's Laws)

State-Space

- State-Space Form