Course Outline - 1st Half

Virginia Tech

- Review Classical Feedback Control
- Review Vector/Matrix Theory
- State-Space Representations (1)
- LTI Response & Matrix Exponential
- Transfer Functions & Eigenvalues
- Frequency-Domain Analysis
- Harmonic & Impulse Responses
- Pole Placement
- Controllability

Virginia Tech

Mathematical Representations of System Models

- First-Order Equations
- State Representations
- Non-Linear Models
- Linearization

First-Order Equations - I

Virginia Tech

Many, but not all, control design techniques require that we have a mathematical model of the process.

Oftentimes a model is obtained by writing a set of (constitutive) equations that describe the physical process.

For mechanical systems we frequently use Newton's Law:

net force = rate of change of momentum

First-Order Equations - II

Virginia Tech

Assuming a constant mass, this law can be written as the differential equation:

$$\underbrace{\sum F}_{\text{net force}} = \frac{d}{dt}\underbrace{\left(m\,v\right)}_{\text{momentum}} = \underbrace{m}_{\text{mass}}\underbrace{\frac{d^2x}{dt^2}}_{\text{acceleration}}$$

Differentiation with respect to time is often written with an overdot for each time derivative:

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

First-Order Equations - III

Virginia Tech

Models of dynamic processes are written as *differential equations* with time t as the independent variable.

For mechanical systems, we typically have

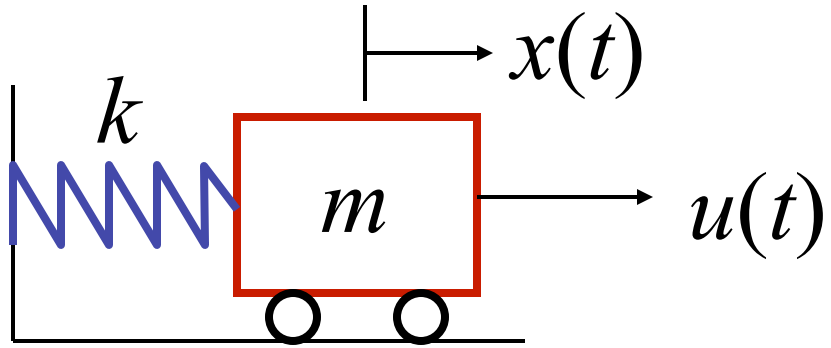
$$x(t) = \text{Position or Displacement}$$

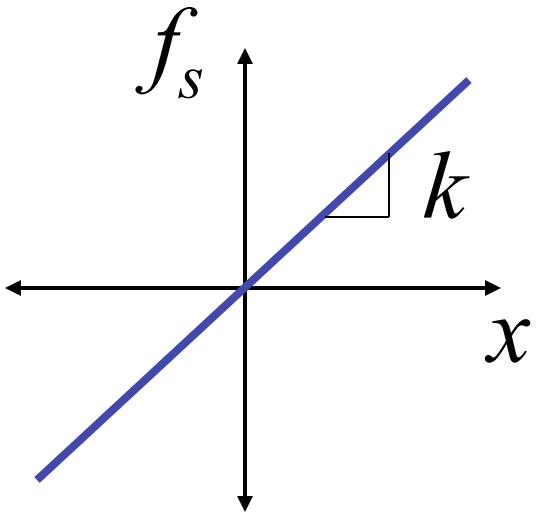
$$\dot{x}(t) = \text{Velocity}$$

$$\ddot{x}(t) = \text{Acceleration}$$

First-Order Equations - IV

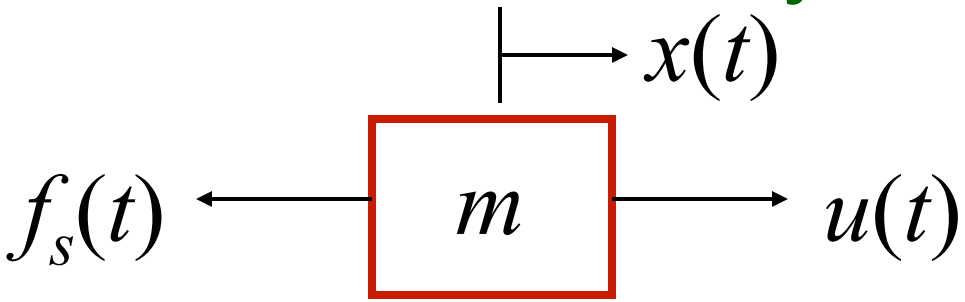
Example: Find the differential equation of motion for a mass-spring oscillator excited by an external force.





First-Order Equations - V

Step 1: Draw the free-body diagram



Step 2: Apply Newton's law

$$m\ddot{x}(t) = \sum F = u(t) - f_s(t) = u(t) - kx(t)$$

The result is a second order diffeq in time:

First-Order Equations - VI

Virginia Tech

Is there any way that we can write this *second-order differential equation* as a set of *first-order differential equations*?

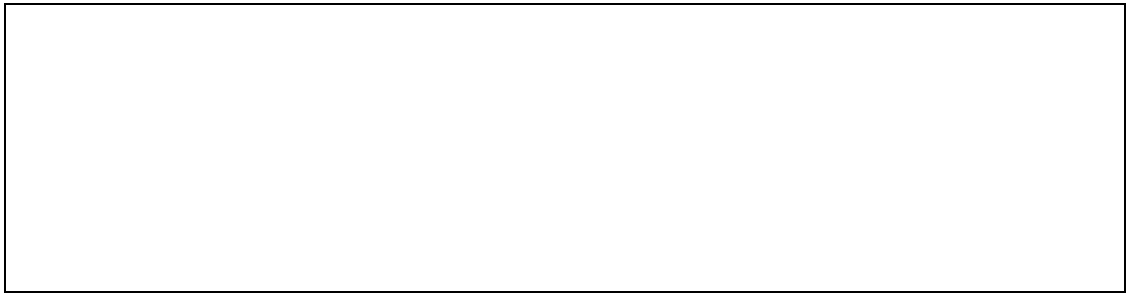
What if we make the following definitions:

First-Order Equations - VII

Substituting these into the original second-order differential equation:

$$m\dot{x}_2(t) + kx_1(t) = u(t)$$

Solving for the first derivative term:



The definition of $x_1(t)$ and $x_2(t)$ provides the second differential equation:

First-Order Equations - VIII

Virginia Tech

Now we can combine these two results into one matrix expression:

This matrix expression is an alternative representation of the original second-order differential equation.

First-Order Equations - IX

Virginia Tech

Why is this important?

Any linear ordinary differential equation or set of differential equations can be written as a set of first-order coupled differential equations.

The original equations and the set of first-order equations represent the same dynamic system.

In-Class Assignment - I

The dynamic equations for a DC motor are

$$\underbrace{J\frac{d^2\theta(t)}{dt^2}}_{\text{rotational inertia}} + \underbrace{b\frac{d\theta(t)}{dt}}_{\text{rotational damping}} = \underbrace{K_T i(t)}_{\text{induced torque}}$$

$$\underbrace{L\frac{di(t)}{dt}}_{\text{electrical inductance}} + \underbrace{Ri(t)}_{\text{electrical resistance}} + \underbrace{K_V\frac{d\theta(t)}{dt}}_{\text{back-EMF}} = \underbrace{V_{in}(t)}_{\text{applied voltage}}$$

Question #1: How many first-order equations will be required to represent this dynamic system?

In-Class Assignment - II

Virginia Tech

Question #2: Write the equations in first-order form using the following variables:

$$i(t),\ \theta(t),\ \dot{\theta}(t)$$

Output Representation - I

Virginia Tech

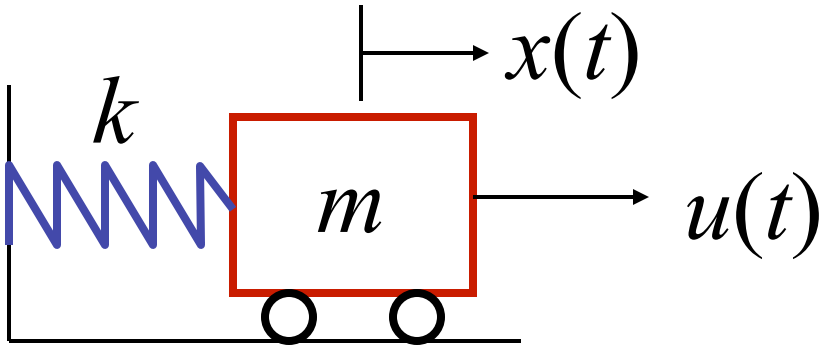
What are outputs?

The Outputs are quantities that we measure either for feedback or for the purpose of quantifying performance.

Outputs can always be written as linear combinations of the Input and States (which haven't been formally defined yet)

Output Representation - II

Let's go back to the spring-mass oscillator example:



What if we were interested in the spring force and the acceleration as outputs?

$$\begin{bmatrix}\text{Spring Force}\\ \text{Acceleration}\end{bmatrix} = \begin{bmatrix}y_1(t)\\ y_2(t)\end{bmatrix} =$$

State Representations - I

We can now write out the full state-space representation for this example:

$$\underbrace{\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix}0 & 1\\ -\frac{k}{m} & 0\end{bmatrix}}_{\mathbf{A}}\underbrace{\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix}0\\ \frac{1}{m}\end{bmatrix}}_{\mathbf{B}}\underbrace{u(t)}_{\mathbf{u}}$$

$$\underbrace{\begin{bmatrix}y_1(t)\\ y_2(t)\end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix}k & 0\\ -\frac{k}{m} & 0\end{bmatrix}}_{\mathbf{C}}\underbrace{\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix}0\\ \frac{1}{m}\end{bmatrix}}_{\mathbf{D}}\underbrace{u(t)}_{\mathbf{u}}$$

State Representations - II

Virginia Tech

As indicated on the previous slide, the standard notational convention for describing a State-Space representation is given by:

$$\underset{[N\times 1]}{\dot{\mathbf{x}}} = \underset{[N\times N][N\times 1]}{\mathbf{A}\ \mathbf{x}} + \underset{[N\times M][M\times 1]}{\mathbf{B}\ \mathbf{u}} \Bigg\} \text{ State Equations}$$

$$\underset{[P\times 1]}{\mathbf{y}} = \underset{[P\times N][N\times 1]}{\mathbf{C}\ \mathbf{x}} + \underset{[P\times M][M\times 1]}{\mathbf{D}\ \mathbf{u}} \Bigg\} \text{ Output Equations}$$

State Representations - III

Conventional Nomenclature:

$\mathbf{x}(t)$	State Vector
$\dot{\mathbf{x}}(t)$	Derivative of State Vector
$\mathbf{A}$	State Matrix
$\mathbf{B}$	Input Matrix
$\mathbf{u}(t)$	Input Vector
$\mathbf{y}(t)$	Output Vector
$\mathbf{C}$	Output Matrix
$\mathbf{D}$	Direct Transmission Matrix

State Representations - IV

What is the “State” of a dynamic system?

The State is a set of variables which, along with the current time, summarizes the current configuration of a system.

States can be positions, velocities, accelerations, forces, momentum, torques, pressure, voltage, current, charge, etc.

State Representations - V

Virginia Tech

Important State Representation facts:

- The choice of state variables is not unique
- A model using a specific choice of state variables is called a Realization
- A system has infinitely many realizations
- The Dimension of a realization is N (the number of states)
- The Order of a system is the minimal number of state variables required to describe it

State Representations - VI

Virginia Tech

Example: Find an alternate realization for the spring-mass oscillator example.

What if we choose momentum and spring force as states? (Remember that we chose position and velocity before)

$$\hat{x}_1(t) = m\dot{x}(t) = \text{Momentum}$$

$$\hat{x}_2(t) = kx(t) = \text{Spring force}$$

State Representations - VII

Substitute into the original diffeq:

$$m\ddot{x}(t) + kx(t) = u(t) = \dot{\hat{x}}_1(t) + \hat{x}_2(t)$$

Solve for the first derivative:

Relate the two new states to find the second differential equation:

State Representations - VIII

In state-space notation we have the following new realization:

$$\begin{bmatrix}\dot{\hat{x}}_1(t)\\ \dot{\hat{x}}_2(t)\end{bmatrix} = \begin{bmatrix}0 & -1\\ \frac{k}{m} & 0\end{bmatrix}\begin{bmatrix}\hat{x}_1(t)\\ \hat{x}_2(t)\end{bmatrix} + \begin{bmatrix}1\\ 0\end{bmatrix}u(t)$$

$$\begin{bmatrix}y_1(t)\\ y_2(t)\end{bmatrix} = \begin{bmatrix}0 & 1\\ 0 & -\frac{1}{m}\end{bmatrix}\begin{bmatrix}\hat{x}_1(t)\\ \hat{x}_2(t)\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{m}\end{bmatrix}u(t)$$

Compare this with the original realization:

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix} = \begin{bmatrix}0 & 1\\ -\frac{k}{m} & 0\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{m}\end{bmatrix}u(t)$$

$$\begin{bmatrix}y_1(t)\\ y_2(t)\end{bmatrix} = \begin{bmatrix}k & 0\\ -\frac{k}{m} & 0\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{m}\end{bmatrix}u(t)$$

State Representations - IX

We can construct any new realization by transforming the states using a non-singular (i.e. full rank) transformation $\mathbf{T}$:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \qquad \mathbf{x} = \mathbf{T}\hat{\mathbf{x}}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \qquad \text{and} \qquad \hat{\mathbf{x}} = \mathbf{T}^{-1}\mathbf{x}$$

State Representations - X

Virginia Tech

Pre-multiply the state equations by $\mathbf{T}^{-1}$

$$\Rightarrow \quad \dot{\hat{\mathbf{x}}} = \mathbf{A}'\hat{\mathbf{x}} + \mathbf{B}'\mathbf{u}$$

$$\mathbf{y} = \mathbf{C}'\hat{\mathbf{x}} + \mathbf{D}\mathbf{u}$$

Even though the state representation is not unique, the input-output relationships are invariant.

General State Representations

A set of first-order equations can be written in the following general form:

$$\left\{\begin{array}{c} \dot{x}_1 = f_1\left(x_1,\ldots,x_N,u_1,\ldots,u_M,t\right) \\ \vdots \\ \dot{x}_N = f_N\left(x_1,\ldots,x_N,u_1,\ldots,u_M,t\right) \end{array}\right\} \quad \Leftrightarrow \quad \dot{\mathbf{x}}(t) = \mathbf{f}\left(\mathbf{x},\mathbf{u},t\right)$$

$$\left\{\begin{array}{c} y_1 = g_1\left(x_1,\ldots,x_N,u_1,\ldots,u_M,t\right) \\ \vdots \\ y_P = g_P\left(x_1,\ldots,x_N,u_1,\ldots,u_M,t\right) \end{array}\right\} \quad \Leftrightarrow \quad \mathbf{y}(t) = \mathbf{g}\left(\mathbf{x},\mathbf{u},t\right)$$

Where the f_n and g_p can be any function.

General State Representations

There are three model classification types:

Linear Time Invariant (LTI) systems

- f_n and g_p are linear combinations of states and inputs with no time-varying coefficients:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

Linear Time Varying (LTV) systems

- f_n and g_p are linear combinations of states and inputs with time-varying coefficients:

$$\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x} + \mathbf{B}(t)\mathbf{u}$$

$$\mathbf{y} = \mathbf{C}(t)\mathbf{x} + \mathbf{D}(t)\mathbf{u}$$

General State Representations

Virginia Tech

Non-Linear (NL) systems

- f_n and/or g_p have terms with non-linear functions of states and/or inputs (i.e. sin, cos)
- f_n and/or g_p have terms with states and/or inputs appearing as powers of something other than 1 or 0
- f_n and/or g_p have terms with cross products of states and/or inputs

Linearization - I

Virginia Tech

Non-linear dynamic systems can be Linearized, but only about a specified operating point $(\mathbf{x}_0, \mathbf{u}_0)$

Typically we choose the operating point $(\mathbf{x}_0, \mathbf{u}_0)$ to be a "fixed point", i.e.

$$\dot{\mathbf{x}} = \mathbf{f}\left(\mathbf{x}_0, \mathbf{u}_0\right) = \mathbf{0}$$

Next, define the deviation variables:

$$\tilde{\mathbf{u}} = \mathbf{u} - \mathbf{u}_0 \qquad \tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}_0 \qquad \dot{\tilde{\mathbf{x}}} = \dot{\mathbf{x}}$$

Linearization - II

The Taylor series expansion about the operating point is:

$$\dot{\tilde{\mathbf{x}}} = \mathbf{f}(\mathbf{x},\mathbf{u}) = \mathbf{f}(\mathbf{x}_0,\mathbf{u}_0) + \mathbf{J}_{\mathbf{f},\mathbf{x}}(\mathbf{x}_0,\mathbf{u}_0)\tilde{\mathbf{x}} + \mathbf{J}_{\mathbf{f},\mathbf{u}}(\mathbf{x}_0,\mathbf{u}_0)\tilde{\mathbf{u}} + (\text{h.o.t.})$$

In the special but very common case where the operating point is at the origin:

$$\mathbf{x}_0 = \mathbf{0} \qquad \mathbf{u}_0 = \mathbf{0}$$

The Taylor series simplifies to:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x},\mathbf{u}) \\ \mathbf{y} &= \mathbf{g}(\mathbf{x},\mathbf{u}) \end{aligned} \quad \Rightarrow \quad \begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(\mathbf{x}_0,\mathbf{u}_0)\mathbf{x} + \mathbf{B}(\mathbf{x}_0,\mathbf{u}_0)\mathbf{u} \\ \mathbf{y} &= \mathbf{C}(\mathbf{x}_0,\mathbf{u}_0)\mathbf{x} + \mathbf{D}(\mathbf{x}_0,\mathbf{u}_0)\mathbf{u} \end{aligned}$$

Linearization - III

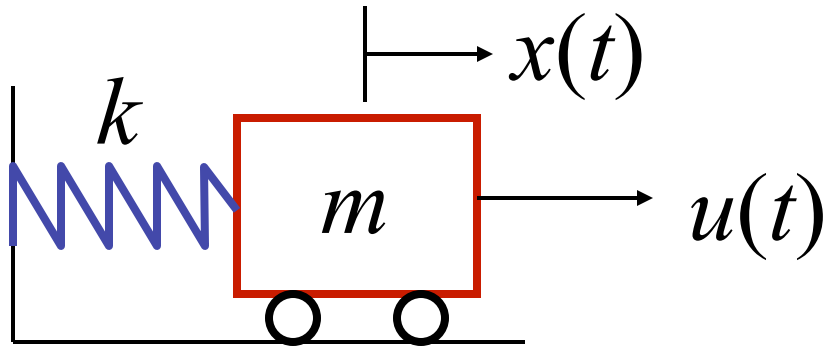
The linearized matrices are the Jacobian matrices of $\mathbf{f}$ and $\mathbf{g}$

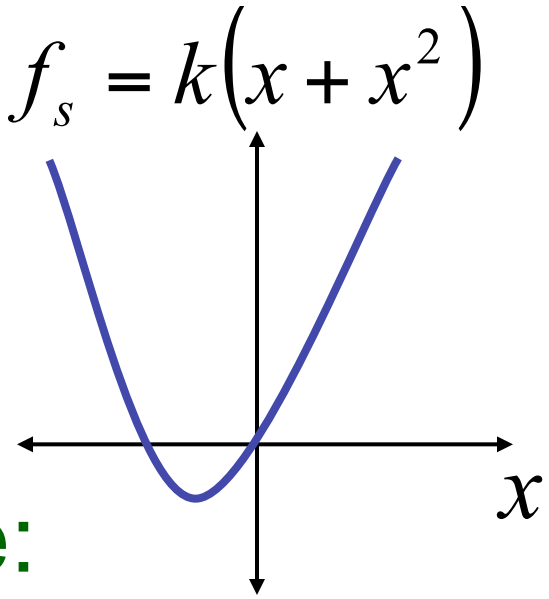
$$\mathbf{A}(\mathbf{x}_0,\mathbf{u}_0)=\left[\begin{array}{ccc}\ddots & \vdots & \iddots \\ \cdots & \dfrac{\partial f_i(\mathbf{x},\mathbf{u})}{\partial x_j} & \cdots \\ \iddots & \vdots & \ddots\end{array}\right]\Bigg|_{\substack{\mathbf{x}=\mathbf{x}_0\\ \mathbf{u}=\mathbf{u}_0}} \qquad \mathbf{B}(\mathbf{x}_0,\mathbf{u}_0)=\left[\begin{array}{ccc}\ddots & \vdots & \iddots \\ \cdots & \dfrac{\partial f_i(\mathbf{x},\mathbf{u})}{\partial u_j} & \cdots \\ \iddots & \vdots & \ddots\end{array}\right]\Bigg|_{\substack{\mathbf{x}=\mathbf{x}_0\\ \mathbf{u}=\mathbf{u}_0}}$$

$$\mathbf{C}(\mathbf{x}_0,\mathbf{u}_0)=\left[\begin{array}{ccc}\ddots & \vdots & \iddots \\ \cdots & \dfrac{\partial g_i(\mathbf{x},\mathbf{u})}{\partial x_j} & \cdots \\ \iddots & \vdots & \ddots\end{array}\right]\Bigg|_{\substack{\mathbf{x}=\mathbf{x}_0\\ \mathbf{u}=\mathbf{u}_0}} \qquad \mathbf{D}(\mathbf{x}_0,\mathbf{u}_0)=\left[\begin{array}{ccc}\ddots & \vdots & \iddots \\ \cdots & \dfrac{\partial g_i(\mathbf{x},\mathbf{u})}{\partial u_j} & \cdots \\ \iddots & \vdots & \ddots\end{array}\right]\Bigg|_{\substack{\mathbf{x}=\mathbf{x}_0\\ \mathbf{u}=\mathbf{u}_0}}$$

Linearization - IV

Example: What if the original spring-mass oscillator now has a non-linear spring?





The first-order diffeq's are:

$$x_1 = x \;\Rightarrow\; \dot{x}_1 = x_2 = f_1(\mathbf{x},\mathbf{u})$$

$$x_2 = \dot{x} \;\Rightarrow\; \dot{x}_2 = -\tfrac{k}{m}\left(x + x^2\right) + \tfrac{1}{m}u = f_2(\mathbf{x},\mathbf{u})$$

Linearization - V

Virginia Tech

The Jacobian matrices for the state equations evaluated at $\mathbf{x}_0 = \mathbf{u}_0 = \mathbf{0}$ are:

$$\mathbf{A}(\mathbf{x}_0, \mathbf{u}_0) = \left[\begin{array}{cc} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{array}\right]\Bigg|_{\mathbf{x}_0 = \mathbf{u}_0 = \mathbf{0}} = \left[\begin{array}{cc} 0 & 1 \\ -\frac{k}{m}(1 + 2x) & 0 \end{array}\right]\Bigg|_{\mathbf{x}_0 = \mathbf{u}_0 = \mathbf{0}} = \left[\begin{array}{cc} 0 & 1 \\ -\frac{k}{m} & 0 \end{array}\right]$$

$$\mathbf{B}(\mathbf{x}_0, \mathbf{u}_0) = \left[\begin{array}{c} \frac{\partial f_1}{\partial u} \\ \frac{\partial f_2}{\partial u} \end{array}\right]\Bigg|_{\mathbf{x}_0 = \mathbf{u}_0 = \mathbf{0}} = \left[\begin{array}{c} 0 \\ \frac{1}{m} \end{array}\right]$$

Note: This result happens to be the same as the original linear result because of a careful choice of nonlinear spring!