State-Space Models

(Section 2.4-2.5 in Ogata)

**n**th order linear ODEs can be written as a set of coupled, first-order ODEs. For example, consider

\[ y^{(3)} + 8y^{(2)} + 9y + 7y = u \]

**IC:**
\[ y(0) = y_0, \quad y'(0) = y_0, \quad y''(0) = y_0 \]

Define

\[ X_1 = y, \quad X_2 = y', \quad X_3 = y'' \]

We can rewrite this 3rd order ODE as:

\[
\begin{align*}
\dot{X}_1 &= X_2 \\
\dot{X}_2 &= X_3 \\
\dot{X}_3 &= -8X_3 - 9X_2 - 7X_1 + u
\end{align*}
\]

This can be written in matrix form as:

\[
\begin{bmatrix}
\dot{X}_1 \\
\dot{X}_2 \\
\dot{X}_3
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
-8 & -9 & -7
\end{bmatrix}
\begin{bmatrix}
X_1 \\
X_2 \\
X_3
\end{bmatrix} +
\begin{bmatrix}
0 \\
0 \\
1
\end{bmatrix} u
\]

\[
\begin{bmatrix}
y_1 \\
y_2 \\
y_3
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
X_1 \\
X_2 \\
X_3
\end{bmatrix}
\]

If we define \( X(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} \) and matrices \( A, B, C \), then this can be compactly expressed as a vector-valued, first-order differential equation:

\[
\dot{X} = AX + Bu \\
y = CX
\]

There is a single, vector-valued initial condition \( X(0) = \begin{bmatrix} y_0 \\ y_0' \\ y_0'' \end{bmatrix} \)
A generic \( n \)-th order ODE can always be converted to a state-space model of the form

\[
\begin{align*}
\dot{x}(t) &= Ax(t) + Bu(t) \\
y(t) &= Cx(t) + Du(t)
\end{align*}
\]

IC: \( x(0) = x_0 \).

This is a collection of \( n \) coupled first-order differential equations. This is known as a "state-space" model. \( x(t) \) is the "state" at time \( t \).

The state at time \( t \) along with the value of the input is all that is required to compute future values of the output. Thus, the state \( x(t) \) encapsulates all the information about the system condition at time \( t \).

The state-space model is equivalent to the \( n \)-th order ODE representation. The state-space representation has some advantages. Most importantly, the matrix representation allows us to use tools from linear algebra.
Ex 1) State-space model for RLC Circuit

\[ \begin{align*}
R & \quad C & \quad L \\
\frac{d^2 v_c}{dt^2} + \left( \frac{R}{L} \right) \frac{dv_c}{dt} + \left( \frac{1}{LC} \right) v_c = \left( \frac{1}{LC} \right) v_i \\
\text{IC:} & \quad v_c(0) = v_{c0}, \quad \dot{v}_c(0) = \frac{v_{i0}}{C}
\end{align*} \]

Define \( x_1 = v_c \)
\( x_2 = \dot{v}_c \)
\( y = v_c \)

\[ \begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= v_c = -\frac{R}{L} x_2 - \frac{1}{LC} x_1 + \frac{1}{LC} v_i
\end{align*} \]

\[ \begin{align*}
\dot{x} &= \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{1}{LC} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{LC} \end{bmatrix} v_i \\
y &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}
\end{align*} \]
When the linear ODE does not depend on \( \dot{u}, \ddot{u}, \) etc then the state-space model can be constructed as follows:

\[
y \frac{d}{dt} x_1 \text{d}t + a_1 x_1 \text{d}t + \cdots + a_n y(t) = u(t)
\]

Define the states as

\[
x_1 = y \\
x_2 = \dot{y} \\
\vdots \\
x_n = y^{(n-1)}
\]

Define

\[
X = \begin{bmatrix}
x_1 \\
x_2 \\
\vdots \\
x_n
\end{bmatrix}
\]

It is easy to show that (1) is equivalent to the state-space model

\[
\dot{X} = AX + Bu \\
y = CX
\]

where

\[
A = \begin{bmatrix}
0 & 1 & 0 & \cdots & 0 \\
0 & 0 & 1 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & 1 \\
-a_n & a_{n-1} & -a_{n-2} & \cdots & -a_1
\end{bmatrix} \\
B = \begin{bmatrix}
0 \\
0 \\
\vdots \\
0 \\
1
\end{bmatrix} \\
C = \begin{bmatrix}
1 & 0 & \cdots & 0
\end{bmatrix}
\]

Notes:
1) The state-space realization is not unique. Different definitions of the state \( X \) lead to different state-space matrices. All these different realizations represent the same dynamics.

2) If the linear ODE depends on \( \ddot{u}, \dddot{u}, \) etc (i.e., the right side of (1) is not just \( u(t) \)) then deriving a state-space realization is more complicated (see §25 of Ogata).
up till now we've only considered linear ODE models.

While most of the course will focus on linear models, it is important to realize that system dynamics can have significant nonlinear terms. In many cases, linear models can be developed from the nonlinear model through "linearizing" and "linearizing".

Ex) Pendulum

The moment of inertia about the pivot point is $I = ml^2$

By Newton's 2nd law for rotational systems:

$I \ddot{\theta} = u - Mgls \sin \theta$

$\Rightarrow \ddot{\theta} + \frac{g}{l} \sin \theta = \frac{1}{ml^2} u$

This is a nonlinear, second-order ODE.

We can construct a state-space model by defining $X_1 = \theta$

$X_2 = \dot{\theta}$

$\begin{align*}
\dot{X}_1 &= X_2 \\
\dot{X}_2 &= \ddot{\theta} = -\frac{g}{l} \sin(X_1) + \frac{1}{ml^2} u
\end{align*}$

This nonlinear state-space model has the form

$\begin{align*}
\dot{X}_1 &= f(X_2) \\
\dot{X}_2 &= F(X_1, X_2, u)
\end{align*}$
In general an \( n \)-state, single input state-space model has the form
\[
\begin{align*}
    \dot{x}_1 &= f_1(x_1, x_2, \ldots, x_n, u) \\
    \dot{x}_2 &= f_2(x_1, x_2, \ldots, x_n, u) \\
    &\quad \vdots \\
    \dot{x}_n &= f_n(x_1, x_2, \ldots, x_n, u)
\end{align*}
\]
In addition, the output can be a not nonlinear function of the states and input
\[
y = h(x_1, x_2, \ldots, x_n, u)
\]
These equations are more compactly written as
\[
\begin{align*}
    \dot{x} &= F(x, u) \\
    y &= h(x, u)
\end{align*}
\]
(+ an initial condition: \( x(0) = x_0 \))

Equilibrium Points

Consider a nonlinear differential equation
\[
\dot{x} = F(x, u)
\]
A point \( \bar{x} \in \mathbb{R}^n \) is called an "equilibrium point" if \( \bar{u} \) (denotes \( n \)-dimensional, real vector)
there is an input \( \bar{u} \in \mathbb{R} \) (called the equilibrium input) such that \( F(\bar{x}, \bar{u}) = 0 \).
This is called an equilibrium point because if the system is initialized to \( x(0) = \bar{x} \) and the input is held at \( u(t) = \bar{u} \) \( \forall t \geq 0 \) then the solution \( x(t) \) satisfies \( x(t) = \bar{x} \) \( \forall t \geq 0 \).
Recall our notation for a general $n^{th}$ order linear ODE:

$$a_n y^{(n)}(t) + a_{n-1} y^{(n-1)}(t) + \ldots + a_2 y''(t) + a_1 y'(t) + a_0 y(t) = b_m u^{(m)}(t) + b_{m-1} u^{(m-1)}(t) + \ldots + b_1 u'(t) + b_0 u(t)$$

where $y^{(k)}(t) = \frac{d^k y}{dt^k}(t)$ is the $k^{th}$ derivative of $y$.

We can write this ODE in the following way:

$$Y = \left[ \frac{b_m s^m + b_{m-1} s^{m-1} + \ldots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \ldots + a_1 s + a_0} \right] U$$

$$= G(s)$$

$G(s)$ is the transfer function associated with the linear ODE. At this point (2.2) is merely a new notation for the ODE (1.1), i.e., the transfer function is simply a different way of representing the ODE.

Later in the course we'll see that the Laplace Transform can be used to make a more formal connection between the ODE and transfer function representations. Moreover, we'll see that the transfer function representation has several uses beyond being just another notation for the ODE.

Finally, we will typically represent the dynamics in a block diagram as:

$$U \rightarrow G(s) \rightarrow Y$$

\[\text{Ex 1) The transfer function for the ODE:}\]

\[6 \dot{y} + 9 \dot{y} + 2y = 4u + 8u \]

\[G(s) = \frac{4s + 8}{6s^2 + 9s + 2} \]